
Etude du comportement de pâturage de différentes races bovines dans différentes formes dégradées de savane arbustive à *Hyparrhenia diplandra* au ranch Nyanga (Gabon)

Auteur : Renard, Gauthier

Promoteur(s) : Bindelle, Jérôme; 8330

Faculté : Gembloux Agro-Bio Tech (GxABT)

Diplôme : Master en bioingénieur : sciences agronomiques, à finalité spécialisée

Année académique : 2018-2019

URI/URL : <http://hdl.handle.net/2268.2/7919>

Avertissement à l'attention des usagers :

Tous les documents placés en accès ouvert sur le site le site MatheO sont protégés par le droit d'auteur. Conformément aux principes énoncés par la "Budapest Open Access Initiative"(BOAI, 2002), l'utilisateur du site peut lire, télécharger, copier, transmettre, imprimer, chercher ou faire un lien vers le texte intégral de ces documents, les disséquer pour les indexer, s'en servir de données pour un logiciel, ou s'en servir à toute autre fin légale (ou prévue par la réglementation relative au droit d'auteur). Toute utilisation du document à des fins commerciales est strictement interdite.

Par ailleurs, l'utilisateur s'engage à respecter les droits moraux de l'auteur, principalement le droit à l'intégrité de l'oeuvre et le droit de paternité et ce dans toute utilisation que l'utilisateur entreprend. Ainsi, à titre d'exemple, lorsqu'il reproduira un document par extrait ou dans son intégralité, l'utilisateur citera de manière complète les sources telles que mentionnées ci-dessus. Toute utilisation non explicitement autorisée ci-avant (telle que par exemple, la modification du document ou son résumé) nécessite l'autorisation préalable et expresse des auteurs ou de leurs ayants droit.

**ETUDE DU COMPORTEMENT DE PÂTURAGE
DE DIFFÉRENTES RACES BOVINES
DANS DIFFÉRENTES FORMES DÉGRADÉES
DE SAVANE ARBUSTIVE À HYPARRHENIA
DIPLANDRA AU RANCH NYANGA (GABON)**

GAUTHIER RENARD

**TRAVAIL DE FIN D'ÉTUDES PRÉSENTÉ EN VUE DE L'OBTENTION DU DIPLÔME DE
MASTER BIOINGÉNIEUR EN SCIENCES AGRONOMIQUES**

ANNÉE ACADÉMIQUE 2018-2019

CO-PROMOTEURS : JERÔME BINDELLE & A. L. HERINAÏNA ANDRIAMANDROSO

Toute reproduction du présent document, par quelque procédé que ce soit,
ne peut être réalisée qu'avec l'autorisation de l'auteur
et de l'autorité académique de Gembloux Agro-Bio Tech.

Le présent document n'engage que son auteur.

**ETUDE DU COMPORTEMENT DE PÂTURAGE
DE DIFFÉRENTES RACES BOVINES
DANS DIFFÉRENTES FORMES DÉGRADÉES
DE SAVANE ARBUSTIVE À HYPARRHENIA
DIPLANDRA AU RANCH NYANGA (GABON)**

GAUTHIER RENARD

**TRAVAIL DE FIN D'ÉTUDES PRÉSENTÉ EN VUE DE L'OBTENTION DU DIPLÔME DE
MASTER BIOINGÉNIEUR EN SCIENCES AGRONOMIQUES**

ANNÉE ACADÉMIQUE 2018-2019

CO-PROMOTEURS : JERÔME BINDELLE & A. L. HERINAINA ANDRIAMANDROSO

Remerciements

Je tiens tout d'abord à remercier la direction de Siat Group et celle du ranch Nyanga ainsi que mes co-promoteurs Jérôme Bindelle et Herinaina Andriamandroso pour m'avoir permis de réaliser ce travail de fin d'études.

Je remercie également Guy Lov et André, sans qui la mise en place de mon dispositif expérimental et la récolte de mes données n'auraient pas été possibles.

Je remercie tout particulièrement Yves Brostaux pour son aide précieuse dans le traitement de mes données ainsi que tous les membres du personnel du ranch Nyanga pour leur accueil sympathique et leurs nombreux coups de main.

J'exprime toute ma reconnaissance à ma mère, mon père et mes sœurs pour leur soutien pendant toute la durée de mes études.

J'aimerais remercier tous mes amis gembloutois pour les moments passés ensemble du jour de mon arrivée au home jusqu'à celui de la remise de mon TFE.

Un grand merci en particulier à mes compagnons de galère Choubi, Stron et Lucas ainsi qu'à mes relecteurs attentionnés, Plum, Botox et Cyril.

Je remercie chaleureusement le Climat de Biture, le Bannissement Accidentel des Régimes et l'Amoindrissement Gériatrique d'avoir rythmé mes années de fac.

Finalement, merci à zébu 2B ainsi qu'à Perceval et Karadoc de m'avoir permis de tenir le coup durant les longues heures passées seul dans mes pâtures.

Résumé

Ce travail de fin d'études a pour objectif l'évaluation de la pertinence d'une gestion différenciée des pâturages et des troupeaux en fonction des différentes races bovines au ranch Nyanga (Gabon). Des techniques classiques d'élevage extensif sont pratiquées afin d'exploiter la végétation naturelle du ranch (savane arbustive à *Hyparrhenia diplandra*). Deux éléments présentent cependant une évolution : l'état de dégradation des pâtures et le mix des races bovines élevées (Ndama, zébu Goudali de l'Adamaoua, Ndama x Senepol et zébu Goudali de l'Adamaoua x Senepol).

Ces différentes races ont été étudiées en saison des pluies dans une pâture partiellement dégradée et une pâture fortement dégradée via un screening de variables liées à leur comportement alimentaire, social et spatial. Les données ont été récoltées au moyen d'observations directes et de capteurs présents dans des iPhones. Une cartographie de la végétation a également été réalisée.

L'analyse statistique des observations directes a permis de conclure que les troupeaux de race homogène devaient être privilégiés à cause des différences de structures sociales entre les races. L'analyse géostatistique de la position des bêtes (capteurs des iPhones) et de ses liens avec les caractéristiques de la végétation (cartographie) a permis de conclure qu'une gestion différenciée des pâturages et des troupeaux en fonction des races n'étaient pas justifiée au vu de l'absence de différence de comportement spatial et alimentaire entre celles-ci. Cependant, les bêtes ont présenté des différences de comportement spatial et alimentaire en fonction de la pâture dans laquelle elles se trouvaient. Des recherches plus approfondies à ce sujet ont donc été suggérées.

Mots-clés : Elevage en ranch, pâturage, Ndama, Goudali, Senepol, comportement animal, comportement alimentaire, comportement social.

Abstract

This master thesis aims to evaluate the relevance of differentiated grazing management regarding various bovine races at the Nyanga ranch (Gabon). Classical methods of extensive breeding are implemented to exploit the ranch natural vegetation (*Hyparrhenia diplandra* shrubby savanna). The ranch shows two evolving elements: land degradation and bovine races mix (Ndama, Adamaoua Goudali zebu, Ndama x Senepol and Adamaoua Goudali zebu x Senepol).

Those different races were studied during the rainy season in a partially degraded pasture and in a highly degraded pasture through a screening of observations about their feeding, social and spatial behaviour. Data were harvested using direct observations and iPhone sensors. A vegetation mapping was carried out.

Statistical analyses of direct observations show that homogeneous breed herds should be preferred because of social structure differences between races. Geostatistical analyses of cattle positions (iPhone sensors) linked to vegetation characteristics (mapping) show that differentiated grazing management regarding breeds is not justified. Indeed, there is a lack of differences in their feeding and spatial behaviour. However, the studied bovines show differences in spatial and feeding behaviour regarding pasture type. Hence, further studies should be carried on this topic.

Keywords : ranching, grazing, Ndama, Goudali, Senepol, animal behaviour, feeding behaviour, social behaviour.

Table des matières

Remerciements	I
Résumé	II
Abstract	III
Table des matières	IV
Liste des figures	VII
Liste des tableaux	VIII
1. Introduction.....	1
2. Contexte du ranching en Afrique tropicale.....	2
3. Contexte du ranching au Gabon.....	9
4. Ranch Nyanga.....	12
a. Géographie et conditions du milieu.....	12
b. Végétation	13
c. Dégradation des pâtures	14
d. Techniques d'élevage	15
e. Races bovines	16
5. Problématique	19
6. Objectif de l'étude.....	21
7. Matériel et méthode	22
a. Dispositif expérimental	22
b. Récolte des données relatives aux animaux	23
c. Récolte des données relatives à la végétation.....	26
d. Traitement statistique des sources de biais et des observations directes du comportement	29
e. Prétraitement des données provenant des iPhones	33
f. Traitement statistique de la vitesse des bêtes et de leur accès à l'eau et au sel	35
g. Traitement géostatistique de la position des bêtes et de ses liens avec la végétation.....	36

8.	Résultats	41
a.	Sources de biais et observations directes du comportement.....	41
b.	Vitesse des bêtes et accès à l'eau et au sel	44
c.	Position des bêtes et liens avec la végétation	45
9.	Discussion	50
a.	Biais, limites et pistes d'améliorations	50
b.	Interprétation des résultats.....	53
10.	Conclusions et perspectives	56
	Références bibliographiques	57
	Annexes	64
	Annexe 1 : carte du ranch Nyanga	64
	Annexe 2 : calendriers de gestion des troupeaux et des pâtures.....	67
	Annexe 3 : photos des races bovines	69
	Annexe 4 : photos des pâtures de l'expérimentation.....	70
	Annexe 5 : chronogramme de l'expérimentation	71
	Annexe 6 : tableau récapitulatif des données récoltées	72
	Annexe 7 : fiches standardisées pour les observations directes du comportement	75
	Annexe 8 : schéma de la protection de l'iPhone et de la batterie.....	78
	Annexe 9 : fiche standardisée pour la caractérisation de la végétation	79
	Annexe 10 : photos des exclos utilisés pour les prélèvements végétaux.....	80
	Annexe 11 : description du script du traitement statistique des sources de biais	81
	Annexe 12 : description du script du traitement statistique des observations directes continues	82
	Annexe 13 : description du script du traitement statistique des observations directes discontinues	84
	Annexe 14 : script du traitement statistique des sources de biais.....	86
	Annexe 15 : script du traitement statistique des observations directes continues	87
	Annexe 16 : script du traitement statistique des observations directes discontinues.....	102
	Annexe 17 : description de la fonction de prétraitement des données provenant des iPhones.....	110
	Annexe 18 : description du script du prétraitement des données provenant des iPhones	111
	Annexe 19 : fonction de prétraitement des données provenant des iPhones.....	112

Annexe 20 : script du prétraitement des données provenant des iPhones	114
Annexe 21 : description du script du traitement statistique de la vitesse des bêtes et de leur accès à l'eau et au sel.....	120
Annexe 22 : script du traitement statistique de la vitesse des bêtes et de leur accès à l'eau et au sel	123
Annexe 23 : variogrammes de la présence des bêtes.....	133
Annexe 24 : rasters de la présence des bêtes	137
Annexe 25 : analyse en composantes principales des caractéristiques de la végétation	145
Annexe 26 : variogrammes des caractéristiques de la végétation	146
Annexe 27 : rasters des caractéristiques de la végétation.....	151
Annexe 28 : description du script du traitement géostatistique de la position des bêtes et de ses liens avec la végétation	154
Annexe 29 : script du traitement géostatistique de la position des bêtes et de ses liens avec la végétation	157
Annexe 30 : tableau récapitulatif des moyennes et des écarts-types de toutes les variables pour tous les groupes d'animaux.....	169

Liste des figures

Figure 1 : carte de l'agencement des pâtures	22
Figure 2 : photo d'un iPhone protégé par une boîte étanche et placé sur un licol.....	25
Figure 3 : photo d'une batterie protégée par une bouteille de soda et placée sur un licol.	25
Figure 4 : carte de la caractérisation de la végétation	27
Figure 5 : carte des prélèvements végétaux.....	28
Figure 6 : carte de la position des bêtes.....	33
Figure 7 : organigramme du processus de traitement de données qui permet d'évaluer la corrélation entre les données relatives à la position des bêtes et celles relatives aux caractéristiques de la végétation	37

Liste des tableaux

Tableau 1 : détermination du seuil d'accélération gravitationnelle selon l'axe X qui permet de séparer les données « grazing » des données « non grazing »	35
Tableau 2 : résultats du traitement statistique des sources de biais.....	41
Tableau 3 : résultats du traitement statistique des observations directes continue.....	41
Tableau 4 : résultats du traitement statistique des observations directes discontinues	42
Tableau 5 : résultats complémentaires du traitement statistique des observations directes du comportement : moyennes et écarts-types par races	43
Tableau 6 : résultats du traitement statistique des vitesses des bêtes et de leur accès à l'eau et au sel pour les données de jour.....	44
Tableau 7 : résultats du traitement statistique des vitesses des bêtes et de leur accès à l'eau et au sel pour les données de nuit	44
Tableau 8 : résultats du traitement statistique des vitesses des bêtes et de leur accès à l'eau et au sel pour les données "grazing"	45
Tableau 9 : résultats du traitement statistique des vitesses des bêtes et de leur accès à l'eau et au sel pour les données "non grazing"	45
Tableau 10 : résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation pour les données de jour	46
Tableau 11 : résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation pour les données de nuit.....	47
Tableau 12 : résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation pour les données "grazing"	47
Tableau 13 : résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation pour les données "non grazing"	48
Tableau 14 : résultats complémentaires du traitement géostatistique de la position des bêtes et de ses liens avec la végétation : moyennes et écarts-types par pâture	48

1. Introduction

La démarche de ce travail de fin d'études fait suite à un stage technique en entreprise au ranch Nyanga, une exploitation du groupe Siat située dans le sud du Gabon. Cette expérience a apporté une vision d'ensemble du fonctionnement interne du ranch, de son potentiel, de ses points faibles et de ses possibilités d'optimisation qui invitent à la réflexion (Renard, 2017). Des pistes d'améliorations se présentent en effet à différentes échelles et dans des domaines d'expertise particulièrement variés (nutrition animale, cultures fourragères, gestion des adventices, gestion de la qualité, modélisation des systèmes d'élevage, santé animale, etc.). Les implications concrètes et immédiates ainsi que les changements profonds qui peuvent en résulter rendent le sujet passionnant.

Cette approche « locale » s'inscrit dans un contexte socio-économique en pleine mutation dont elle ne peut se dissocier. Les relations entre la société et l'élevage évoluent en profondeur partout dans le monde, mêlant les traditions et l'Histoire avec des problématiques et des politiques actuelles. L'image de l'élevage auprès du public, les pratiques zootechniques employées et les modèles économiques qui y sont associés présentent une grande diversité dans le temps et l'espace et demandent à être analysés en profondeur.

Une compréhension du contexte du ranching dans la région s'avère indispensable pour comprendre les enjeux de ce travail de fin d'études. C'est pourquoi il a été analysé en détail, d'abord à l'échelle de l'Afrique tropicale puis à celle du Gabon. La situation et le fonctionnement du ranch Nyanga ont alors pu être étudiés sur la base de ce contexte. Une problématique pertinente et originale a pu en être dégagée. Les objectifs ont ensuite été délimités dans le but d'apporter des éléments nouveaux à cette problématique et des méthodes à la fois rigoureuses et adaptées aux réalités du terrain ont été sélectionnées puis mises en œuvre. La structure de ce travail de fin d'études est basée sur ce processus de réflexion.

2. Contexte du ranching en Afrique tropicale

Les pâturages représentent un quart des terres émergées et deux tiers des terres agricoles. Les herbivores domestiques, et tout particulièrement les ruminants, ont la capacité d'y transformer des végétaux impropres à la consommation humaine en différents produits alimentaires riches en protéines. La grande diversité des systèmes de production et des ressources génétiques animales permet ainsi d'exploiter des milieux où l'élevage est la seule possibilité de production agricole. Les opportunités économiques associées aux ruminants domestiques et leurs apports à la sécurité alimentaire jouent ainsi un rôle clé dans de nombreuses régions (Mottet et al., 2018).

Dans les pays en développement, l'élevage est amené à se développer en réponse à l'accroissement de la demande en protéines animales. En effet, la croissance démographique, l'augmentation progressive des revenus et l'urbanisation croissante de ces régions du monde modifient les habitudes alimentaires et demandent une évolution des systèmes d'élevage (Besse & Toutain, 2012).

En 2017, la population africaine de ruminants comptait 346 millions de bovins, 381 millions d'ovins et 422 millions de caprins, ce qui correspond à 23%, 32% et 41% de leur population mondiale respective (FAO, 2018). La production annuelle de viande était de 6,5 millions de tonnes pour les bovins, de 1,8 millions de tonnes pour les ovins et de 1,4 millions de tonnes pour les caprins, ce qui correspond à 10%, 19% et 23% de leur production mondiale respective (FAO, 2019a). La population humaine atteignait dans un même temps les 1,25 milliards de personnes en Afrique (FAO, 2019b).

La diversité des systèmes d'élevage liés aux pâturages étant particulièrement importante, la contextualisation de ce travail de fin d'études se concentre sur l'élevage bovin dans un système de ranching en Afrique tropicale (Classification climatique A¹ dans le système Köppen-Geiger), excluant donc les zones arides (Classification B²) et tempérées (Classification C³) (Tsalefac et al., 2015 ; Peel et al., 2007). Cette région couvre plusieurs centres régionaux d'endémisme qui présentent des caractéristiques différentes au niveau de leurs systèmes

¹ Classification A : température du mois le plus froid supérieure à 18°C.

² Classification B : précipitations annuelles moyenne en mm inférieures à 20 fois la température moyenne en °C (si moins de 30% des précipitations tombent en été), augmentée de 140 si 30% à 70% des précipitations tombent en été ou augmentée 280 si plus de 70% des précipitations tombent en été.

³ Classification C : température du mois le plus froid comprise entre 0 et 18°C et température du mois le plus chaud supérieure à 10°C.

d'élevage. Les plus importants sont le domaine soudano-sahélien, qui couvre la majorité de l'Afrique de l'ouest et la partie nord de l'Afrique centrale, et le domaine guinéo-congolais, qui couvre la partie sud de l'Afrique centrale et l'extrémité sud-ouest de l'Afrique de l'ouest (Boutrais, 1983 ; White, 1983).

Le ranching est un système d'élevage sédentaire basé sur le pâturage de prairies permanentes sans intégration avec des productions végétales. La conduite des animaux est simplifiée, les apports de fourrages et de concentrés sont absents et les surfaces exploitées sont importantes mais délimitées. Les infrastructures sont généralement limitées à des clôtures, des points d'abreuvement et des centres de soins. Les intrants sont également peu nombreux, principalement constitués de médicaments vétérinaires. Le ranching est pratiqué dans les régions naturellement recouvertes de prairies, de steppes et de savanes mais également dans d'anciennes régions forestières enherbées (Dufumier, 2012). Si la strate herbacée constitue donc la principale source de nourriture des bovins, le potentiel fourragé des ligneux ne doit cependant pas être négligé dans les régions les plus sèches (Sèwadé et al., 2016).

Les méthodes de ranching permettent de produire de la viande, du lait, de la laine ou du cuir à partir de différentes espèces animales et de mettre en place des programmes de sélection. Elles sont cependant utilisées dans la grande majorité des cas pour la production de viande bovine. Des ranchs naisseurs, engraisseurs et naisseurs-engraisseurs coexistent (Otte & Chilonda, 2002). Le système de ranching, en se basant sur la modification du territoire plutôt que sur le déplacement des troupeaux, présente l'avantage de demander peu de main d'œuvre par rapport aux systèmes pastoraux traditionnels (Boutonnet, 2005).

Même si la situation semble s'inverser (Otte & Chilonda, 2002), Les ranchs africains ont historiquement présenté des résultats très inégaux et souvent inférieurs aux élevages traditionnels (Lhoste et al., 1993). Pour comprendre cette situation, un retour sur l'histoire des ranchs africains est nécessaire.

D'origine anglosaxonne, le ranching a été introduit progressivement par les européens dans les colonies d'Afrique tropicale à partir de la première guerre mondiale. Les ranchs ont été mis en place par des entreprises privées car les administrations coloniales ne souhaitent pas investir dans des projets aussi risqués. Les négociations entre les entreprises et les administrations ont souvent été longues et compliquées. En effet, au vu de l'ampleur des investissements financiers, les entreprises privées ont demandé d'importantes concessions (jusqu'à plusieurs centaines de milliers d'hectares) afin d'atteindre leurs objectifs de rentabilité.

Les administrations coloniales se sont souvent montrées réticentes car elles reprochaient aux ranchs de concurrencer les éleveurs locaux sans leur apporter d'innovations techniques. Malgré ces problèmes, les ranchs de l'époque coloniale ont globalement montré de bons résultats et se sont révélés être des structures économiquement viables (Boutrais, 1990).

A la suite de l'indépendance des pays africains, le contexte politique a évolué et l'appui aux ranchs privés européens a été retiré. Dans les années 1970, les ranchs d'Etat ont été créés afin de diminuer la dépendance alimentaire alarmante de l'Afrique tropicale et de servir de modèle à l'élevage traditionnel. Leur mise en place a été longue et coûteuse et s'est accompagnée d'erreurs techniques et stratégiques (décisions hasardeuses dues au manque d'expérience, absence ou changements d'orientations stratégiques, etc.). Les ranchs étatiques ont globalement été un échec et, avec la perte de soutien de la Banque mondiale, les états n'ont progressivement plus été en mesure de les maintenir. Ceux-ci ont donc fait faillite et ont été privatisés selon différents modèles (Boutrais, 1990). Ces différents modèles expliquent une partie de la grande diversité observée dans les systèmes d'élevage et dans leurs performances. La situation varie cependant également en fonction des régions.

La situation dans le domaine soudano-sahélien est la plus complexe à analyser, tant par la diversité des modèles pratiqués que par l'organisation sociale de sa population. En effet, l'élevage traditionnel et le nomadisme, liés aux conditions du milieu, y sont très présents et leur cohabitation avec le système de ranching est particulièrement difficile (Gallais, 1979). On peut distinguer trois principaux modèles ayant coexistés ou succédés aux ranchs d'Etat : les ranchs « capitalistes », les ranchs « familiaux » et les ranchs « collectifs ».

La mise en place de ranchs « capitalistes », de grande taille, et le plus souvent détenus par de riches citadins ou par des européens, a permis l'utilisation des techniques d'élevage acquises avec l'expérience des précédents modèles. En effet, les investissements liés à ces techniques étant conséquents et ne permettant qu'une rentabilité à long terme, ils sont uniquement à la portée de grosses entreprises. Ce modèle a cependant rarement été privilégié car il nécessitait une mise à l'écart de la communauté pastorale et l'expropriation des éleveurs présents sur le territoire. Ces pratiques, en opposition totale avec le droit coutumier, ont entraîné un bouleversement et un clivage profond de la société ainsi que de nombreuses tensions dans les régions où elles ont été mises en place (Boutrais 1983). De plus, la main d'œuvre nécessaire à ce modèle a été engagée parmi les anciens éleveurs traditionnels dépossédés de leurs terres, créant ainsi une nouvelle classe sociale, appauvrie et privée de son statut (Boutrais, 1987).

Une alternative aux grands ranchs « capitalistes » est leur morcellement en petits ranchs privés « familiaux ». Cependant, les conflits entre ranchs ainsi que ceux entre les ranchs et les éleveurs pastoraux, le détournement des moyens mis en place et le manque de rentabilité des techniques de ranching à cette échelle ont considérablement freiné leur développement (Boutrais, 1990). Ces problèmes prennent leurs racines dans la profonde incompatibilité entre, d'un côté, le changement dans les techniques d'élevage et la privatisation du foncier et, de l'autre, la structure sociale (droit coutumier, structure familiale, dot, héritage, etc.) et les traditions pastorales (nomadisme et transhumance, gardiennage traditionnel, etc.) (Boutrais, 1987).

Le modèle des ranchs « collectifs » est une alternative qui a présenté des résultats plus satisfaisants et a contenté les participants aux projets. Ces ranchs devaient, en théorie, concilier les avancées techniques et la cohésion sociale en mutualisant les efforts des éleveurs dans un système de ranching collaboratif de grande échelle (Boutrais, 1990). Ils n'ont cependant pas pu se maintenir dans le temps à cause de problèmes structurels, organisationnels et techniques. En effet, les décisions se sont rarement appuyées sur des expertises agronomiques solides et les objectifs qui ont été fixés ont manqué de stabilité, de cohérence et de précision. De plus, des tensions se sont développées entre les éleveurs inclus dans le projet et ceux qui n'en faisaient pas partie (Boutrais, 1987). Ni les conditions sociales, ni les conditions environnementales, n'étaient alors réunies pour l'essor de ce modèle (Bougler et al., 2005).

Le bilan des systèmes de ranching dans le domaine soudano-sahélien a donc présenté peu de réussites, notamment à cause de politiques peu judicieuses et de la mise en œuvre de modèles inadaptés. Cela a engendré des tensions sociales et des conflits entre ranchs et éleveurs traditionnels, principalement autour de problèmes fonciers (Gonne et al., 2010). Cependant, le secteur du commerce de bovins sur pied s'est depuis développé et présente un caractère particulièrement dynamique (Duteurtre et al., 2003). Le secteur de l'élevage invite ainsi à mettre en place des politiques plus adéquates afin d'adapter et de moderniser les techniques de production. Le ranching fait partie intégrante des différents modèles d'élevage qui peuvent être mis en avant via ces politiques (Vall et al., 2003).

La situation dans le domaine guinéo-congolais est très différente et également moins complexe. L'élevage bovin traditionnel y est beaucoup moins répandu (Otte & Chilonda, 2002) et moins documenté. Si la contrainte de la sécheresse dans le domaine soudano-sahélien a pu être contournée dans l'élevage traditionnel par la mobilité des troupeaux, les contraintes

sanitaires des régions plus humides n'ont pas permis le développement de celui-ci (Gallais, 1979). Les vecteurs de maladies tels que les glossines et, dans une moindre mesure, les tiques ont été la principale raison du rôle minime joué historiquement par l'élevage bovin dans les sociétés guinéo-congolaises. Il a donc fallu attendre la diffusion des races trypanotolérantes (Ndama et Baoulé principalement) à travers le domaine guinéo-congolais et leur importation dans des ranchs dits « modernes » ou « industriels » nouvellement créés pour voir se développer l'élevage bovin (Lecomte & Compère, 2005 ; Moula et al., 2014 ; Shaw & Hoste, 1991).

Les modèles d'élevage y sont souvent basés sur de très grands ranchs « capitalistes » (jusqu'à plusieurs centaines de milliers d'hectares), ceux-ci étant particulièrement bien adaptés aux conditions sociales, économiques et environnementales. Ceux-ci présentent généralement d'assez bons résultats pour être économiquement viables. Les principales difficultés rencontrées avec ces modèles sont liées à la taille des exploitations et à la grande variabilité spatio-temporelle de la qualité des pâtures (Lecomte & Compère, 2005). Le fait que les spécificités du travail d'éleveur soient étrangères à la culture des employés peut également être un frein au vu de la masse salariale employée (Boutrais, 1987).

Bien que l'élevage se soit déjà fortement développé dans la région, les perspectives restent particulièrement importantes au vu du potentiel de production inexploité (Lecomte & Compère, 2005). Le développement de la filière permettrait, en outre, de répondre à la demande croissante en protéines animales et, de ce fait, de diminuer l'ampleur du problème de surexploitation de la viande de brousse (Moula et al., 2014). Une augmentation de la production locale permettrait également de diminuer les importations régionales de viande, qui sont souvent vendues à des prix beaucoup plus élevés que dans les pays producteurs (Duteurtre et al., 2003).

En Afrique de l'est, en bordure des domaines soudano-sahélien et guinéo-congolais, les systèmes de ranching, tout comme l'élevage de manière générale, sont beaucoup plus développés. En effet, en 2017, la densité de population bovine en Afrique de l'est était trois fois plus importante qu'en Afrique de l'ouest et presque six fois plus qu'en Afrique centrale (FAO, 2018). Si les conditions initiales, qu'elles soient environnementales ou sociales, restent proches de celle du domaine soudano-sahélien, l'évolution des modèles de ranching et les résultats obtenus par ceux-ci en sont à l'opposé.

Les ranchs « capitalistes » et, dans certains cas, les ranchs d'Etat ont généralement montré de bonnes conditions de production et des performances stables et satisfaisantes. Cela

s'explique par des méthodes de gestion plus réfléchies et une meilleure maîtrise technique et stratégique que dans les régions soudano-sahéliennes (Barnes, 1979). En effet, l'expérience accumulée depuis l'introduction du ranching, ainsi que la stabilité des politiques agricoles mises en place, ont permis de mettre au point des méthodes d'élevage appropriées. Le ranching a ainsi pu prendre un rôle important dans l'économie (Skovlin, 1971).

L'Afrique de l'est présente également un élevage traditionnel très ancré dans la société. Cependant, son encadrement progressif dès l'époque coloniale, appuyé par des politiques cohérentes et efficaces après les indépendances, a permis une transition progressive vers des techniques d'élevage plus modernes. Cet encadrement a tout de même régulièrement suscité incompréhension et méfiance mais n'a que rarement amené à des tensions sociales (Ng'ethe, 1993 ; Skovlin, 1971).

Des essais de ranchs collectifs ont ainsi pu être menés avec des résultats bien supérieurs à ceux obtenus dans le domaine soudano-sahélien. La mise à disposition de moyens et de conseils techniques inspirés des élevages « capitalistes » a permis la transition vers les méthodes de ranching malgré une certaine réticence des éleveurs. Les résultats sont cependant restés mitigés pour des raisons principalement culturelles et structurelles (Lhoste et al., 1993).

De manière générale, les conditions sociales et environnementales, les structures et les modèles, ainsi que les résultats et les performances des ranchs d'Afrique tropicale présentent une grande diversité. Cependant, les techniques d'élevage mises au point et utilisées dans les différentes régions restent très souvent semblables et ont peu évolué depuis les années 1970 (Labonne et al., 2003). Il s'agit d'un système éprouvé qui, lorsque les conditions sont réunies, se montre tout à fait viable. Il ne faut cependant pas perdre de vue que différentes techniques ont été pratiquées en Afrique tropicale afin d'intensifier la production.

Les techniques qui avaient pour objet l'amélioration des pâtures ont principalement poursuivi deux objectifs. Le premier était la modification de la composition floristique des espèces de graminées et de fabacées ainsi que celle de la strate arbustive. Les méthodes utilisées reposaient alors sur la mise en place d'espèces plus productives, plus nutritives ou mieux appréciées, et sur la réduction de la concurrence des espèces présentant peu d'intérêt fourrager. Le second objectif visait quant à lui l'optimisation de l'accès aux nutriments et à l'eau pour la végétation (Barnes, 1979). Ces techniques d'intensification ont généralement impliqué l'augmentation des recours aux intrants tels que les herbicides ou les engrais (Dufumier, 2012).

D'autres techniques ont visé l'amélioration de la génétique animale. Les plus anciennes se sont basées sur le croisement de races trypanotolérantes et de zébus afin d'améliorer les performances zootechniques (gabarit, croissance, reproduction, etc.) des premiers, tout en conservant la possibilité de les élever dans des régions infestées par les glossines (Bougler et al., 2005). Plus récemment, l'amélioration des races s'est généralement concentrée sur des croisements par absorption. Dans ce cadre, des races locales, bien adaptées aux conditions du milieu, ont été absorbées par des races modernes particulièrement productives, parfois au moyen de l'insémination artificielle (Dufumier, 2012).

Dans le cas d'un système d'élevage déjà bien maîtrisé, la production a également pu être améliorée via une gestion plus fine et rigoureuse des conduites de pâturage et des charges en bétail. La complémentation s'est également présentée comme un outil intéressant en cas de déficit qualitatif ou quantitatif dans l'alimentation des animaux. La cohérence de l'ensemble des techniques d'élevage utilisées reste cependant une condition nécessaire à toute tentative d'intensification de la production (Besse & Toutain, 2012).

3. Contexte du ranching au Gabon

La description générale du ranching en Afrique tropicale permet de situer parfaitement la situation du Gabon dans ce contexte. Le Gabon est situé en zone guinéo-congolaise. La pression glossinaire y est importante et la population ne présente pas de tradition pastorale. Le ranching, absent durant l'époque coloniale, s'y est développé sous la forme de ranchs d'Etat, qui ont par la suite été privatisés sous la forme de ranchs « capitalistes ».

Le Gabon est un état rentier dont l'économie est basée sur l'exploitation du pétrole, du bois et des minerais (manganèse, fer, uranium, etc.). L'agriculture, qu'elle soit vivrière ou commerciale, y est peu développée pour la région (Feumetio, 2008 ; Gouvernement de la République gabonaise, 2005). Elle ne représentait, en effet, que 5,3% du PIB en 2017 (World Bank, 2019a). Le Gabon présente une faible densité de population d'environ 8 habitants par km² (World Bank, 2019b), ainsi qu'un écart particulièrement important entre la richesse du pays et le niveau de vie réel des habitants (Feumetio, 2008 ; Gouvernement de la République gabonaise, 2005). En effet, en 2017, le pays était classé 74^{ème} en ce qui concerne le produit intérieur brut par habitant à parité du pouvoir d'achat (World Bank, 2019c), alors qu'il n'était que 110^{ème} pour l'indice de développement humain (United Nations Development Programme, 2018).

Le coût de la main d'œuvre, les difficultés liées au transport de marchandises, l'importance de l'urbanisation et de l'exode rurale ainsi que la densité de population peu élevée sont des éléments qui peuvent expliquer la faible production agricole du Gabon (Feumetio, 2008). Cependant, la raison principale de cette situation est l'abandon du secteur agricole par l'Etat dans les années 1960 et 1970 (Magnagna Nguema, 2005). La faiblesse du secteur entraîne un sérieux problème de sécurité alimentaire au Gabon. Le recours aux importations de denrées alimentaires (Gouvernement de la République gabonaise, 2005) et la pression de la chasse sur la biodiversité des écosystèmes (Moula et al., 2014) sont particulièrement préoccupants.

Seul un refinancement en profondeur du secteur rural et agricole permettrait de renverser la situation de dépendance alimentaire du Gabon (Gouvernement de la République gabonaise, 2005). Ce refinancement devrait être accompagné d'une adaptation du cadre institutionnel et de la formation massive de personnel technique et scientifique compétent, en particulier pour le secteur de l'élevage (Ministère de l'agriculture de l'élevage et du développement rural (Gabon), 2003).

Les statistiques permettant de rendre compte de la situation sont anciennes et relativement rares. On peut cependant estimer que plus de la moitié des produits alimentaires consommés proviennent des importations (Gouvernement de la République gabonaise, 2005 ; Ministère de l'agriculture de l'élevage et du développement rural, 2003). Ces importations de produits alimentaires, évaluées à 235 milliards de francs CFA en 2010, représentaient 15% du total des produits importés. Les importations de produits d'origine animale, évaluées à 82 milliards de francs CFA, en constituaient donc plus d'un tiers (Ministère de l'économie du commerce de l'industrie et du tourisme (Gabon), 2011).

La situation est cependant très variable selon la catégorie de produit. En effet, si la couverture en tubercules, en produits de la pêche ou en œufs est correcte (ordre de grandeur du déficit inférieur à 25%), elle est beaucoup plus problématique pour de nombreux aliments de consommation courante comme le riz, les légumes ou la viande (ordre de grandeur du déficit supérieur à 75%) (Feumetio, 2008, Gouvernement de la République gabonaise, 2005 ; Ministère de l'agriculture de l'élevage et du développement rural (Gabon), 2003). La viande bovine importée provient majoritairement du Cameroun, de l'Union européenne et d'Amérique du sud (Labonne et al., 2003 ; Vall et al., 2003). Paradoxalement, la plupart des produits alimentaires importés présentent d'excellentes possibilités de production au Gabon (Feumetio, 2008).

La production de viande bovine par des structures familiales ou de petites entreprises peut être considérée comme anecdotique au Gabon. Ce sont donc les grands ranchs qui assurent la plus grande partie de la production. Celui de la Nyanga, d'une capacité de plus de 20 000 têtes, en est le principal représentant (Magnagna Nguema, 2005).

C'est à la fin des années 1950, peu avant son indépendance, que le gouvernement gabonais a pris conscience du déficit important de production de viande dans le pays et a entamé des missions prospectives. En 1961, il a commandé une étude afin d'évaluer les possibilités pastorales dans la vallée de la Nyanga dans le but d'y développer l'élevage bovin. Les conclusions ont été positives : les conditions étaient réunies pour la mise en place d'une exploitation de grande envergure. La qualité et la quantité de la flore fourragère ont été jugées suffisantes et les accès à l'eau n'ont pas présenté de difficultés. De plus, ce rapport a décrit une occupation humaine faible et des voies de communication qui pouvaient facilement être tracées ou restaurées (Descoings, 1961).

Ce n'est cependant qu'en 1977 qu'une société publique, la Sogadel (appelée Agrogabon-Elevage dans un premier temps) a été créée afin de développer des élevages industriels de bovins. Trois ranchs ont été mis en place au début des années 1980. Le projet a toutefois pris du retard et il a fallu revoir les objectifs à la baisse. Les nouvelles prévisions envisageaient, à terme, un cheptel de 40 000 têtes pour une production de viande de 1200 tonnes par an. Le projet s'est développé relativement correctement jusqu'en 1994, atteignant les 33 000 têtes. Le nombre de bêtes a ensuite commencé à chuter. En effet, la dévaluation du franc CFA ayant entraîné le doublement du prix des soins vétérinaires, le suivi des troupeaux a été progressivement délaissé. De plus, des difficultés de trésorerie liées à la masse salariale ont entraîné des retards de salaires croissants, des tensions sociales et finalement des grèves « illimitées » (Magnagna Nguema, 2005).

La Sogadel a été placée en liquidation en 2001 (Magnagna Nguema, 2005), dans un contexte général de désengagement de l'Etat dans les entreprises du secteur public (Diambounambatsi, 2015). En 2004, la multinationale de l'agroalimentaire Siat Group a créé une filiale au Gabon et les actifs du ranch Nyanga, ainsi que ceux de deux autres sociétés publiques, Agrogabon (production d'huile de palme) et Hevegab (production de caoutchouc) lui ont été cédés (Diambounambatsi, 2015 ; Feumetio, 2008). Depuis 2016, Siat Gabon limite ses activités aux secteurs de l'hévéa et de l'élevage bovin, les actifs du secteur palmier ayant été cédés au groupe Olam à la suite de problèmes de trésorerie (Siat Gabon, 2017).

Le ranch Nyanga est actuellement encore en phase de développement : le troupeau est composé d'environ 6000 têtes et l'objectif est d'atteindre un effectif de croisière de 20 000 têtes d'ici 2025. Siat Gabon souhaite, à terme, maîtriser l'entièreté de la chaîne de production de viande bovine fraîche afin d'approvisionner le marché gabonais. La priorité actuelle est cependant aux importations de bovins sur pieds et aux programmes de reproduction (Siat Gabon, 2019). Un des principaux freins à l'augmentation du cheptel a été l'épidémie de péripneumonie contagieuse bovine qui a sévit sur le ranch de 2010 à 2014 (Siat Gabon, 2017). Cette maladie est en effet reconnue pour entraîner des impacts économiques importants et être particulièrement difficile à éradiquer (Lefèvre, 2003). Depuis 2018, l'abattage, la découpe et la vente de viande fraîche par le ranch Nyanga ont été suspendus pour une durée indéterminée. Des bovins continuent cependant à être vendus sur pied.

4. Ranch Nyanga

a. Géographie et conditions du milieu

Le ranch Nyanga, d'une superficie d'environ 100 000 hectares, est situé dans le sud du Gabon (province de la Nyanga, département de Mongo). Il prend place dans la vallée de la Nyanga, une plaine schisto-calcaire recouverte de savane. Le ranch est délimité au nord-est par le fleuve Nyanga et au sud-ouest par le massif montagneux du Mayombe. Celui-ci est recouvert de forêt et sépare la plaine littorale de la vallée de la Nyanga et de son prolongement au Congo, la vallée du Niari (Chatelin, 1968). Les savanes de la vallée de la Nyanga ne sont pas d'origine anthropique mais d'origine paléoclimatique, même si l'Homme semble avoir joué un rôle dans leur maintien (Schwartz et al., 1990).

Le ranch présente des sols latéritiques, avec des horizons supérieurs argilo-sableux exploitables par la végétation. De manière générale, l'épaisseur de ces horizons varie selon un gradient perpendiculaire au fleuve, tandis que leur proportion de sable le fait parallèlement. La qualité des sols n'est donc pas totalement homogène (Descoings, 1961).

De nombreux lacs, marais et rivières parsèment la vallée de la Nyanga. Les plus importants d'entre eux ne s'assèchent pas durant la saison sèche (Descoings, 1961).

Les précipitations annuelles sont de l'ordre de 1600 mm par an (Descoings, 1961). Elles sont cependant presque totalement absentes de mi-mai à mi-octobre (grande saison sèche) et présentent une légère diminution de décembre à février (petite saison sèche). Les températures moyennes varient de 23°C en juillet à 27°C en avril. La variabilité spatiale des précipitations et des températures est cependant importante à l'échelle de la vallée (Hellemans, 1984). Le climat est donc de type équatorial à transition australe (Hellemans, 1984 ; Lecomte & Compère, 2005).

Le climat chaud et humide, la présence de galeries forestières le long des cours d'eau et celle du couvert forestier sur les reliefs entraînent une pression en glossine (*Glossina spp.* Wiedemann) particulièrement importante. Le ranch Nyanga abrite également deux espèces responsables d'une pression de prédation sur les veaux : les pythons (*Python sebae* Gmelin) et les potamochères (*Potamochoerus porcus* L.). Finalement, les tiques et les pique-bœufs (*Buphagus erythrorhynchus* Stanley) ont, eux aussi, un impact négatif sur le bétail (Okouyi, 2018).

b. Végétation

La végétation présente sur le ranch prend la forme d'une savane arbustive relativement homogène. Les espèces dominantes et caractéristiques sont *Hyparrhenia diplandra* (Hack.) Stapf et, dans une moindre mesure, *Panicum phragmitoides* Stapf. La végétation est composée de trois strates : une première strate herbacée allant de 40 cm à 1 m, une seconde allant de 1 m à plus de 2 m et une strate arbustive pouvant atteindre les 4 m. Le recouvrement végétal est généralement de 100%. (Descoings, 1961 ; Hellemans, 1984). Les savanes à *Hyparrhenia diplandra* et à *Panicum phragmitoides* sont des formations courantes en région guinéenne. Elles témoignent d'un sol de bonne qualité et correctement approvisionné en eau (Lecomte & Compère, 2005). Cependant, certains sols du ranch sont moins profonds, plus pentus ou dégradés, un gradient de végétation y est alors observé. La végétation devient moins haute, moins dense et moins intéressante au niveau fourragé. *H. diplandra* perd du terrain et *P. phragmitoides* est remplacé par d'autres espèces telles que *Pobeguinea arrecta* (Stapf) Jacq.-Fél. (Descoings, 1961 ; Hellemans, 1984).

En conditions naturelles, le cycle végétatif de *H. diplandra* dépend du cycle des feux. Après avoir subi un feu en fin de saison sèche, les repousses se développent durant la saison des pluies. La floraison a lieu en mai-juin et la maturation des graines en juillet-août. La plante se dessèche alors rapidement et peut facilement prendre feu (Hellemans, 1984). En l'absence de toute intervention humaine, l'intervalle entre les feux est généralement supérieur à un an (Alvarado et al., 2018). Les feux présentent cependant une importance pour la gestion pérenne des savanes à *Hyparrhenia diplandra*. Si ceux-ci sont trop rares ou pas assez puissants, le cycle des nutriments ne fonctionne plus correctement et le sol s'appauvrit. Dans le cas contraire, c'est la reproduction des plantes qui voit son efficacité diminuer (Garnier & Dajoz, 2001). De manière générale dans les milieux tropicaux qui dépendent des feux, une gestion inadaptée de ceux-ci entraîne quasi systématiquement des répercussions néfastes sur les écosystèmes et les activités humaines, en plus de dégager des quantités importantes de gaz à effet de serre (Shlisky et al., 2009).

H. diplandra présente une productivité et des valeurs nutritionnelles satisfaisantes (protéines brutes : 14,8% de la matière sèche ; énergie nette : 1540 kcal par kg de matière sèche). Il s'agit donc d'une plante fourragère de bonne qualité lorsqu'elle est consommée au stade de développement approprié (Descoings, 1961 ; Hellemans, 1984 ; Musco et al., 2016 ; Okouyi, 2018). Celui-ci correspond à des repousses de 30 à 40 cm, âgées d'environ un mois.

c. Dégradation des pâtures

Les risques de dégradation des pâtures tropicales liés à l'élevage extensif les plus classiquement rencontrés sont la diminution du couvert végétal, la dénudation et le compactage des sols, la modification et l'appauvrissement de la flore, le développement d'espèces végétales envahissantes et l'embroussaillage des pâtures (Besse & Toutain, 2012). Tous ces phénomènes sont observables sur le ranch Nyanga, de façon non homogène et à des échelles diverses. En effet, certaines pâtures sont momentanément ou durablement surpâturées. Cela s'explique le plus souvent par des contraintes logistiques, mais parfois également par une analyse trop approximative de la qualité fourragère de la végétation. Certaines pâtures, notamment celles où les bovins se rassemblent, sont des zones « sacrifiées » particulièrement touchées. Elles peuvent donc servir de zones témoins pour évaluer les risques de dégradations futures dans les autres pâtures.

La menace la plus importante qui pèse actuellement sur les pâtures est le développement massif et rapide de l'espèce invasive *Chromolaena odorata* (L.) R.M. King & Robinson à travers tout le ranch. Cette plante buissonnante forme des massifs denses, est assez résistante au feu, n'est aucunement consommée par le bétail et étouffe progressivement toute autre forme de végétation.

Le développement de *C. odorata* est particulièrement difficile à maîtriser pour plusieurs raisons dont sa croissance rapide et sa reproduction précoce. La production de graines est importante et celles-ci forment rapidement des stocks conséquents dans les sols, tout en présentant un taux de germination élevé. Enfin, les graines peuvent être transportées sur de longues distances par le vent, les animaux ou les véhicules (Witkowski & Wilson, 2001).

Différentes méthodes de lutte existent mais elles sont coûteuses et doivent être répétées inlassablement. Un contrôle chimique peut être mis en place au moyen d'herbicides appliqués manuellement ou mécaniquement sur les feuilles, la souche ou le sol. Le contrôle mécanique n'est, quant à lui, efficace que lorsque la plante est complètement déracinée (Goodall & Erasmus, 1996). Les graines ne résistent cependant pas à la chaleur et la plante peut être éliminée par des feux particulièrement intenses déclenchés lorsque les réserves de la plante sont faibles, entre la période de floraison et la libération des graines (Witkowski & Wilson, 2001).

La lutte biologique est une piste particulièrement prometteuse. Si elle a connu beaucoup d'échecs, certains programmes ont tout de même montré des résultats particulièrement encourageants. Ils ont en effet permis une diminution drastique de la couverture de *C. odorata*,

tout en permettant une recolonisation du territoire par les autres espèces végétales. De plus, il s'agit actuellement de la seule méthode de contrôle applicable à grande échelle (Timbilla & Braimah, 2002).

d. Techniques d'élevage

La structure du ranch Nyanga est relativement simple. Il comprend trois divisions qui possèdent chacune leurs propres infrastructures techniques (services généraux, parc de véhicules, locaux de stockage, etc.), socio-économiques (infirmerie, superette, etc.) et administratives (bureaux). Trois à quatre sections d'élevage sont rattachées à chaque division. L'organisation des sections d'élevage se base sur un centre de soin autour duquel viennent se greffer différents parcs. Chaque parc est clôturé et associé à un troupeau généralement homogène. Les parcs sont eux-mêmes composés de plusieurs parcelles séparées par des pare-feux afin de permettre la rotation des feux et du bétail. Une carte du ranch est présentée en annexe 1.

Les bêtes se nourrissent de la végétation naturellement présente sur le ranch. Seul un complément minéral (76,4% NaCl ; 19,6% CaHPO₄ ; 0,6% CuSO₄ ; 1,7% ZnCO₃ ; 0,03% CoSO₄ ; 1,67% liant) est mis à disposition à volonté sous forme de poudre. La charge en bétail qui doit théoriquement être appliquée est de 4 hectares par unité de bétail tropical mais des différences existent entre les pâtures. Dans la plupart des parcs, les points d'eau naturellement présents sont nombreux et suffisent largement à abreuver les bêtes.

Les parcelles ne sont pas brûlées chaque année à la même période mais selon un cycle de quatre feux en cinq ans. Le gardiennage des bêtes par les employés du ranch permet de maintenir le troupeau sur la parcelle qui se situe à la bonne étape du cycle et qui présente donc le niveau de repousse de la végétation adéquat. Une fois par semaine, les employés procèdent à un ramassage du troupeau et l'amènent au centre de soins. Les bêtes sont comptées, inspectées et soignées. Les passages du troupeau au centre de soins permettent également de mettre en œuvre les programmes zootechniques et vétérinaires. Ceux-ci comprennent notamment : la mise au mâle ou l'insémination des femelles, le marquage et l'écornage des veaux, la castration des mâles, les bains acaricides, les vaccinations et les traitements trypano-préventifs. Les calendriers de gestion des troupeaux et des pâtures qui sont appliqués sur le ranch sont présentés en annexe 2.

Toutes les bêtes du ranch Nyanga sont identifiées. Cependant, le suivi des troupeaux est réalisé sur papier et non via un système informatisé, ce qui représente un frein à la réalisation

de statistiques. Les possibilités d'étude et d'interprétation de l'évolution des paramètres démographiques sont donc très limitées.

La monte naturelle et l'insémination artificielle sont toutes les deux pratiquées sur le ranch Nyanga. La monte naturelle constitue la base du programme de reproduction : elle est la méthode la plus ancienne et la plus utilisée. En dépit d'une saison de monte complémentaire organisée après la saison principale sur les femelles non gravides, le taux de gestation moyen reste anormalement faible pour la région. L'insémination artificielle est, quant à elle, utilisée pour réaliser des croisements dans le but d'améliorer les performances du bétail. Les coûts sont importants : importations d'hormones et de semences, honoraires de l'inséminateur, etc. Cette technique permet cependant l'accès à des taureaux de haut potentiel génétique ainsi que l'uniformisation, la synchronisation et le raccourcissement de la période de reproduction. (Okouyi, 2018).

A l'exception du recours à l'insémination artificielle, les techniques d'élevage utilisées sur le ranch Nyanga sont tout à fait classiques et courantes dans la région (Lecomte et Compère, 2005).

e. Races bovines

La population bovine du ranch Nyanga n'est pas homogène. La plupart des bêtes actuellement présentes appartiennent à la race taurine Ndama, à la race zébu Goudali de l'Adamaoua, ou sont issues de croisements F1 obtenus par l'insémination artificielle des femelles de ces deux races par des mâles de la race taurine Senepol. Ces deux catégories de bêtes sont respectivement nommées « Ndapols » et « Zépols ». Chacune de ces races est illustrée par une photo en annexe 3.

L'élevage étant très peu développé au Gabon, les animaux présents sur le ranch Nyanga sont tous originaires de l'extérieur du pays. La race Ndama est la première à avoir été introduite dans les années 1980 et reste encore aujourd'hui majoritaire. La population de Ndamas du bassin du Congo a principalement été importée de Sénégal (Shaw & Hoste, 1991), et la race s'est particulièrement bien implantée au Zaïre (Moula et al., 2014). Les importations de bétail de la Sogadel ont donc été réalisées par avion à partir de ces deux pays (Dehoux, 1990b). En 2010, Siat importait toujours des Ndamas depuis la République démocratique du Congo (Siat Gabon, 2010).

Le berceau de la race Ndama est le plateau du Fouta-Djalou, situé en actuelle Guinée-Conakry. Son aire de distribution s'est cependant étendue à la plupart des pays d'Afrique de l'ouest dès le début du XX^e siècle (Coulomb, 1976 ; Joshi et al., 1957). La rusticité et la trypanotolérance des Ndamas les rendent particulièrement bien adaptés aux milieux chauds, humides et infestés de glossines (Akouango et al., 2010 ; Sokouri et al., 2010). Leur trypanotolérance n'est cependant pas absolue et l'utilisation d'un traitement prophylactique s'avère utile pour éviter les baisses de leurs performances en condition de stress (Dehoux, 1990a). Les Ndamas sont des animaux de petit format mais leurs qualités bouchères sont intéressantes. Ils mesurent en moyenne 115 cm au garrot et peuvent dépasser les 350 kg, tandis que le rendement en viande de leur carcasse atteint les 60 % (Coulomb, 1976). Ils présentent une bonne conformation et leur viande est maigre et de qualité (Joshi et al., 1957).

Les zébus ont été introduits sur le ranch Nyanga beaucoup plus récemment que les Ndamas. Ils ont été importés en bétailière à partir du Cameroun par Siat afin d'accélérer la croissance du troupeau. Les importations ont été suspendues depuis qu'elles ont entraîné l'introduction de la péripneumonie contagieuse bovine sur le ranch (Okouyi, 2018).

Les zébus Goudalis de l'Adamaoua proviennent, comme leur nom l'indique, du plateau de l'Adamaoua, à cheval sur le Cameroun et le Nigeria (Joshi et al., 1957). Ils ne doivent pas être confondus avec les zébus Goudalis de Sokoto auxquels ils sont apparentés. Différentes variétés de zébus Goudalis de l'Adamaoua coexistent et présentent différents phénotypes (Assani Seidou, 2013 ; Messine et al., 1995). De manière générale, la classification et la caractérisation des zébus de cette région sont assez confuses (Tawah & Rege, 1996). Les zébus ne présentent pas de trypanotolérance et doivent donc suivre un traitement prophylactique systématique et poussé pour pouvoir être élevés dans les régions à forte pression glossinaire (Sokouri et al., 2010 ; Tawah & Rege, 1996). La croissance des zébus Goudalis de l'Adamaoua est plus lente que celle des Ndamas. Leur fécondité est également plus faible et plus tardive. (Messine et al., 1995 ; Sokouri et al., 2010). Ils présentent cependant un plus gros gabarit : les variétés les mieux conformées mesurent jusqu'à 130 cm (Tawah & Rege, 1996) et atteignent les 550 kg (Messine et al., 1995). En revanche, leurs qualités bouchères sont moindres, tout comme leur rendement en carcasse d'environ 50 % (Joshi et al., 1957). Leur démarche n'est également pas la même : les zébus Goudalis de l'Adamaoua sont plus hauts sur pattes que les Ndamas, même si la différence est moins importante qu'avec d'autres races de zébus (Assani Seidou, 2013). Finalement, les individus de la race zébu Goudali présentent un tempérament beaucoup plus lent et docile que ceux de la race taurine Ndama.

Dès les années 1920, des essais de croisements entre la race Ndama et des races plus productives ont été entrepris afin d'accroître le gabarit des animaux. Les résultats ont généralement été encourageants, même si les performances sont restées assez inégales (Coulomb, 1976 ; Joshi et al., 1957). Le risque de perte du caractère trypanotolérant des animaux doit cependant être pris en compte lors de l'élaboration des programmes de croisement (Sokouri et al., 2010). Des croisements entre des zébus Goudalis de l'Adamaoua et des individus de races plus productives ont également été réalisés. Les performances obtenues étaient intéressantes : les animaux, et tout particulièrement les F1, ont atteint des poids et des rendements de carcasse nettement supérieurs. Cependant, leur fragilité et leur manque d'adaptation aux milieux exigeants ont rendu difficile leur maintien dans des systèmes d'élevage extensif (Lhoste, 1980).

Des inséminations artificielles à partir de semence de Senepol ont pour la première fois eu lieu sur le ranch Nyanga en 2009 (Siat Gabon, 2010). Elles avaient pour but d'augmenter le poids moyen des animaux et donc la productivité du cheptel. Le nombre d'inséminations et leur taux de réussite s'accroît continuellement depuis. Les semences sont importées du Brésil et les hormones le sont principalement de Belgique (Okouyi, 2018).

La race taurine Senepol a été créée au début du XX^e siècle sur l'île Sainte-Croix dans les Antilles (Iles Vierges des Etats-Unis) à partir de Ndamas importés d'Afrique de l'ouest et de Red Polls importés d'Europe. L'objectif était de développer une race adaptée aux conditions tropicales tout en conservant des rendements du niveau des bovins européens. Les pedigrees et les performances des taureaux ont par ailleurs rapidement été enregistrés. (Williams et al., 1988). Les Senepols sont particulièrement bien adaptés à la chaleur (Hernandez et al., 2002) et résistent bien aux maladies et aux parasites (Alvarez et al., 2000). En revanche, des analyses génétiques ont montré que les apports du Ndama à la race Senepol ne sont finalement que très limités et que celle-ci n'a pas conservé le caractère trypanotolérant (Flori, 2012). La croissance des Senepols est particulièrement rapide (Guimarães et al., 2017) et les bêtes dépassent les 500 kg en pâture sans apport de concentré. Leur rendement carcasse est de l'ordre de 60 % et les qualités organoleptiques de la viande sont élevées (Cianzo, 2002). De nombreux croisements ont déjà été pratiqués afin d'augmenter les performances ou la résistance à la chaleur de différentes races. De plus, les individus F1 présentent des résultats particulièrement intéressants grâce à l'effet d'hétérosis (Chase et al., 1998).

5. Problématique

On peut pointer deux évolutions majeures du ranch Nyanga. Premièrement, les changements observés dans la végétation des pâtures (modifications des proportions des différentes strates et des différentes espèces herbacées, développement d'espèces végétales invasives, embroussaillage, etc.) semblent indiquer une dégradation de son état. Deuxièmement, le mix des races bovines élevées sur le ranch a été et continue d'être modifié. Cette situation amène à se poser des questions sur les implications de ces changements.

Si l'élevage n'est pas la priorité du département recherche et développement de Siat Group, différentes études ont tout de même régulièrement lieu sur le ranch, principalement via des travaux de fin d'études et des thèses de doctorat. Ces études sont cependant presque exclusivement axées sur les thèmes de la reproduction (Okouyi, 2018) et de la santé animale (Dibanganga, 2015). Les thématiques liées à l'évaluation de la qualité des pâtures, à l'interface plante-animale et à l'alimentation animale, si elles ont été abordées par le passé (Hellemans, 1984), ne sont aujourd'hui explorées que de manière très superficielle (Renard, 2017). Le sujet, particulièrement vaste, ne manque pourtant pas d'intérêt pour le développement du ranch.

La qualité et la quantité des aliments ingérés par le bétail sont des éléments clés pour comprendre et améliorer la productivité des systèmes d'élevage. Or les conditions environnementales (ressources fourragères, contraintes physiques, etc.) ne peuvent à elles seules expliquer la variabilité de l'alimentation des animaux. C'est pourquoi l'étude du comportement animal et celle des relations que le bétail entretient avec son milieu font partie intégrante de cette problématique et ne peuvent être négligées. De plus, cette approche éthologique se montre particulièrement pertinente dans le cadre du ranching, du fait de son caractère extensif. (Manteca i Vilanova & Smith, 2014).

Si l'étude du comportement alimentaire proprement dit est le point de départ de la démarche, il ne saurait être analysé isolément d'autres composantes telles que le cycle du jour et de la nuit ou l'alternance de périodes de repos et d'activité (Manteca i Vilanova & Smith, 2014). La sélectivité alimentaire, que ce soit entre les espèces ou les strates de la végétation, est en lien étroit avec les stratégies d'utilisation de l'espace et les déplacements des animaux. Ceux-ci doivent en effet s'adapter à la variabilité spatio-temporelle des ressources alimentaires, de l'accès à l'eau ou de la présence des abris contre les aléas climatiques (Fourcassié, 2015 ; Manteca i Vilanova & Smith, 2014). L'exploitation de la végétation naturellement présente

dans les pâtures, associée à l'irrégularité des microclimats et de la qualité des sols, peut par ailleurs expliquer l'importante variabilité du milieu au sein du ranch Nyanga.

De plus, on ne peut pas écarter l'influence des structures sociales des troupeaux sur les comportements individuels. En effet, les interactions entre les animaux, tels que les communications, les stimulations ou les agressions, entraînent des relations sociales entre individus ainsi que des réseaux sociaux au niveau du groupe. Les phénomènes d'affinité et de hiérarchie modifient profondément l'occupation de l'espace des individus et donc leur comportement alimentaire (Dumont & Boissi, 1999 ; Thierry, 2015).

Une étude qui a pour objectif de faire le lien entre le comportement animal, l'état de dégradation des pâtures et les différences entre les races, se présente donc à la fois comme pertinente et originale dans le contexte du ranch Nyanga.

6. Objectif de l'étude

L'objectif de l'étude est d'évaluer la pertinence d'une gestion différenciée des pâturages et des troupeaux en fonction des différentes races bovines dans les conditions actuelles du ranch Nyanga.

Le principe de l'expérimentation repose sur le « screening » d'un maximum de variables liées au comportement animal, notamment dans ses aspects spatiaux, sociaux et alimentaires. Les mesures doivent être effectuées sur les différentes races bovines présentes sur le ranch et dans des pâtures qui ne présentent pas le même état de dégradation de la végétation afin de pouvoir évaluer l'influence de ces deux facteurs sur l'ensemble des variables testées.

Les résultats de ce « screening » doivent permettre d'obtenir une vision globale de la situation et d'apporter des suggestions quant à la gestion des pâtures et des troupeaux. Par extension, ce travail doit également permettre d'évaluer l'intérêt d'entamer de plus amples recherches sur ce thème et proposer des pistes de réflexions.

7. Matériel et méthode

a. Dispositif expérimental

L'expérimentation s'est déroulée dans la section Kouri de la division 2 du ranch Nyanga du 5 mars au 13 mai 2018 (saison des pluies). Deux pâtures adjacentes au centre de soin de la section ont été mises à disposition par la direction du ranch : la « petite pâture » du parc 1, d'une superficie de 6,2 hectares et la « petite pâture » du parc 8, d'une superficie de 7,6 hectares. Celles-ci sont respectivement nommées « pâture 1 » et « pâture 2 » dans la suite du document. Chacune des deux pâtures est illustrée par une photo en annexe 4.

Les deux pâtures ont été brûlées plus de six mois avant le début de l'expérimentation et disposaient d'abreuvoirs et de bacs à sel contenant un complément minéral durant celle-ci. Une carte des pâtures (figure 1) a été réalisée à partir de points GPS et au moyen du logiciel QGIS afin d'illustrer le dispositif expérimental.

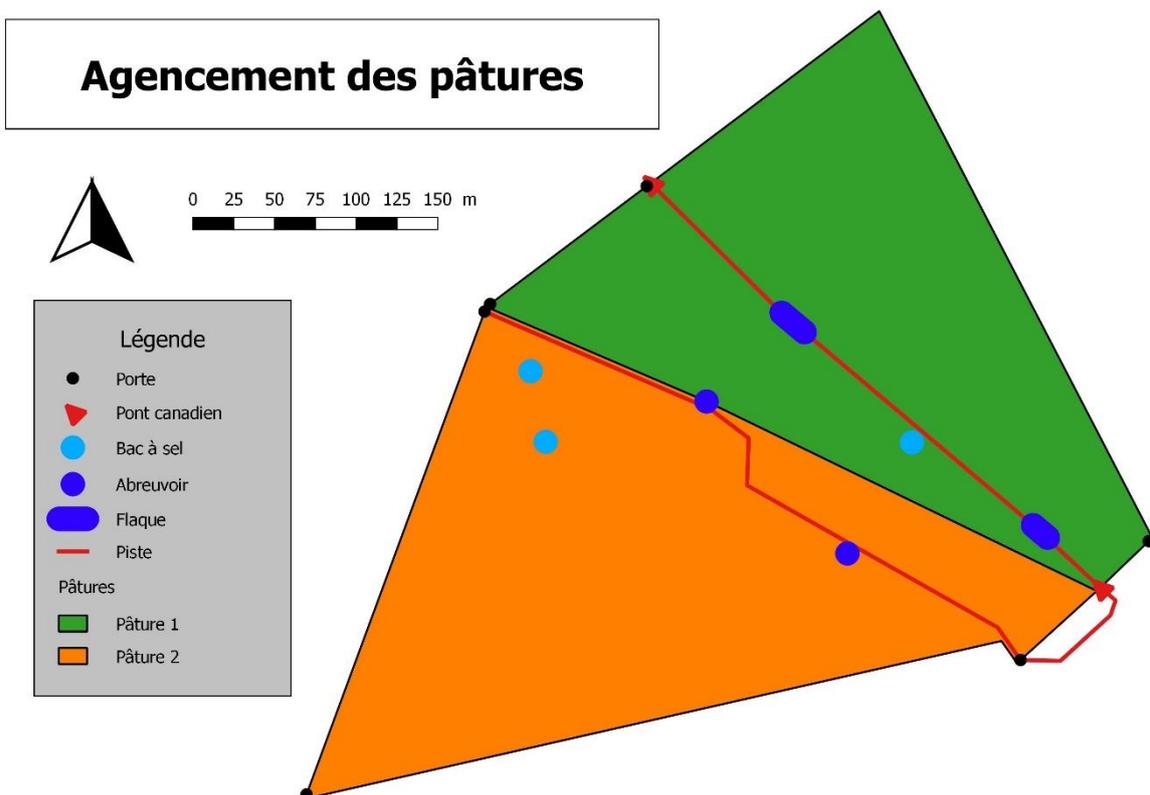


Figure 1 : carte de l'agencement des pâtures

La pâture 1 présente visuellement les caractéristiques d'une dégradation de la végétation plus avancée que dans la pâture 2 : la couverture de la strate herbacée de plus de 1 m est moins importante, la flore comporte moins de graminées et de légumineuses, et l'embroussaillage ainsi que le développement de *Chromolaena odorata* (L.) R.M. King & Robinson sont plus avancés.

Seize bêtes ont été mises à disposition par la direction du ranch Nyanga. Toutes sont nées durant l'année 2014 et proviennent du même troupeau. La sélection était composée de quatre taurillons de la race taurine Ndama, de quatre taurillons de la race zébu Goudali de l'Adamaoua, de quatre bouvillons croisés « Ndapols » et de quatre bouvillons croisés « Zépol ». Les bêtes sélectionnées avaient des caractéristiques physiques dans la moyenne de ce qui peut être observé sur le ranch. Les animaux possédant un tempérament calme et docile ont cependant été privilégiés afin de faciliter les manipulations et les observations.

Les bêtes ont été réparties en deux troupeaux comportant chacun deux bêtes de chaque race (individu A et individu B). Chacun des troupeaux a été attribué à une pâture pour toute la durée de l'expérimentation. Une semaine a été consacrée à l'acclimatation des bêtes aux conditions expérimentales. Les données ont ensuite été récoltées durant neuf semaines consécutives. L'organisation de l'expérimentation a été basée sur un cycle de 16 jours qui a été répété quatre fois. Un chronogramme de ce cycle et un tableau récapitulatif des données récoltées sont respectivement présentés en annexe 5 et 6.

b. Récolte des données relatives aux animaux

Les données relatives aux animaux peuvent être divisées en deux catégories en fonction de l'outil utilisé pour leur acquisition. La première correspond à des observations directes du comportement effectuées au moyen de formulaires standardisés, tandis que la seconde a été obtenue grâce à l'utilisation des capteurs présents dans des iPhones.

Les observations directes ont été effectuées de manière à observer le comportement d'un seul animal durant 12 heures consécutives, de 06h30 à 18h30. Chaque bête s'est vu consacrer une journée pendant la durée de l'expérimentation, en respectant une alternance entre les différentes races et les pâtures. Les observations n'ont cependant pas pu être réalisées durant certaines plages horaires pour des raisons pratiques (grosses intempéries, etc.). Celles-ci ont donc dû être effectuées lors de journées de « rattrapage ». L'utilisation d'une paire de jumelles a permis d'éviter de perturber le comportement des animaux durant les observations.

Les comportements liés notamment au broutage des ligneux et aux interactions entre animaux ont été étudiés au moyen d'observations continues. Chacune des apparitions de ces comportements a ainsi été notifiée et décrite au moment où elle se produisait. A contrario, les variables comportementales liées à la structure sociale du troupeau ont été relevées au moyen d'observations discontinues, réalisées toutes les dix minutes.

Les observations directes continues et les variables qui y sont associées sont les suivantes :

- Les bouchées prélevées sur des ligneux : l'heure, l'espèce végétale (*Gardenia ternifolia* subsp. *jovis-tonantis* Welw. - *Sarcocephalus latifolius* Sm. - *Alchornea cordifolia* Schumach. & Thonn. - *Bridelia ferruginea* Benth. – autre ligneux), le nombre de bouchées et la hauteur des bouchées (cm) ;
- Les attaques perpétrées sur d'autres animaux : l'heure, l'identité de la bête ciblée et l'intensité de l'attaque (bousculade – combat à un round – combat à plusieurs rounds) ;
- Le léchage d'autres animaux : l'heure, l'identité de la bête ciblée et le nombre de coups de langue donnés ;
- Les meuglements : l'heure et le nombre de cris poussés ;
- Les flehmens⁴ : l'heure ;
- Les périodes passées allongé : l'heure et la durée (minutes).

Les observations directes discontinues réalisées toutes les dix minutes sont les suivantes :

- L'identité des bêtes présentes dans le groupe ;
- L'identité de la bête la plus proche ;
- La distance à laquelle se trouve la bête la plus proche (m).

Un relevé de toutes les bêtes présentes dans la pâture a été réalisé au début de chaque journée d'observation car les fuites de bêtes du protocole et les intrusions de bêtes hors protocole ont été régulières. Les bêtes ont été identifiées au moyen d'un code à trois chiffres correspondant à la pâture, à la race et à l'individu (A ou B) pour faciliter le traitement informatique des données. Les fiches standardisées utilisées pour réaliser les observations directes sont disponibles en annexe 7.

Quatre iPhones 4s ont été mis à disposition par l'unité de zootechnie de Gembloux Agro-Bio Tech (Université de Liège) dans le but d'utiliser leurs capteurs pour récolter des

⁴ Le flehmen est un comportement sexuel et social au cours duquel l'animal retousse sa lèvre supérieure afin de mieux percevoir les phéromones (McDonnell, 2002)

informations supplémentaires sur le comportement des bêtes. Les données provenant de l'accéléromètre, du gyroscope, du GPS et du chronomètre ont été enregistrées à une fréquence de 100 Hz au moyen de l'application « Sensor Data 1.23 ». Des batteries externes de 15 000 mAh ont été branchées aux iPhones afin de prolonger leur autonomie jusqu'à 60 heures. Les iPhones et les batteries ont cependant dû être préalablement protégés des chocs, des pressions, de l'humidité et de la surchauffe due au soleil. Un schéma du dispositif de protection est présenté en annexe 8. Afin de procéder aux mesures, les iPhones (figure 2) et les batteries (figure 3) ont été fixés à des licols ajustés sur la tête des animaux



*Figure 2 : photo d'un iPhone protégé par une boîte étanche et placé sur un licol.
Gauthier Renard, 2018*



*Figure 3 : photo d'une batterie protégée par une bouteille de soda et placée sur un licol.
Gauthier Renard, 2018*

Les sessions de mesures au moyen des iPhones se sont déroulées selon un cycle de quatre jours. Le premier jour à l'aube, les iPhones et les batteries étaient placés sur quatre bêtes de races différentes. Ils étaient récupérés au crépuscule du troisième jour, tout comme les licols auxquels ils étaient fixés. Le quatrième jour, pendant que les quatre bêtes suivantes s'acclimataient au port du licol, les batteries étaient rechargées. Les iPhones étaient également redémarrés et leur mémoire vidée afin de minimiser les risques de crash de l'application. Les câbles qui reliaient les batteries aux iPhones ont été régulièrement remplacés du fait de leur usure particulièrement rapide.

Après chaque pose d'un iPhone sur la tête d'un animal, celui-ci a été filmé durant 30 minutes afin de mettre en parallèle son comportement avec les données récoltées. Les séquences vidéo ont été tournées de façon à permettre l'observation de l'emploi du temps de la bête tout en évitant de la perturber dans ses activités.

En complément de ces données, une pesée de l'ensemble des bêtes a été organisée tous les 16 jours à l'aide d'une balance à bestiaux afin de pouvoir comparer leur poids et leur gain quotidien moyen.

c. Récolte des données relatives à la végétation

Les données relatives à la végétation peuvent être divisées en deux catégories en fonction des techniques utilisées pour leur acquisition. Les premières ont été obtenues par une description visuelle du couvert végétal tandis que les secondes proviennent de prélèvements effectués dans la végétation, avant et après la mise en place d'exclos.

Dix critères ont été retenus afin d'évaluer visuellement le couvert végétal. Ils ont été choisis de manière à permettre une caractérisation de la végétation la plus complète possible tout en étant observables rapidement, debout et à l'œil nu. Ces critères sont les suivants :

- La présence de graminées rampantes ou gazonnantes ;
- La présence de graminées cespitueuses de moins de 1 m de haut ;
- La présence de graminées cespitueuses de plus de 1 m de haut ;
- La présence de *Calopogonium mucunoides* Desv. ;
- La présence de *Sida cf. acuta* Burm.f. ;
- La présence de *Mitracarpus hirtus* L. ;
- La présence de végétation herbacée non reprise dans les 6 critères précédents ;
- La présence de végétation arbustive à l'exception de *C. odorata* ;
- La présence de *C. odorata* ;
- La présence d'abris contre le soleil et la pluie.

Lors de chacune des observations, les critères ont été évalués sur une surface de 9 m² correspondant à un carré de 3 m par 3 m autour de l'observateur. Une échelle allant de zéro à trois a été utilisée afin de rendre les observations les plus objectives possibles et de permettre leur traitement statistique. L'attribution de scores sur cette échelle a été effectuée selon la méthode suivante :

- Score de 0 : couvert inférieur à 1 m² ;
- Score de 1 : couvert estimé à 1, 2 ou 3 m² ;
- Score de 2 : couvert estimé à 4, 5 ou 6 m² ;
- Score de 3 : couvert estimé à 7, 8 ou 9 m².

Un GPS a été utilisé afin d'arpenter l'entièreté des deux pâtures selon des transects parallèles orientés nord-sud et distants de $0,0001^\circ$ (11,1 m). Sur ces transects, une observation a été effectuée tous les $0,0001^\circ$ (11,1 m) et retranscrite sur une fiche standardisée. Cette fiche est présentée en annexe 9. Une carte reprenant les points d'observation utilisés pour la caractérisation de la végétation (figure 4) a été réalisée au moyen du logiciel QGIS afin de permettre leur visualisation.

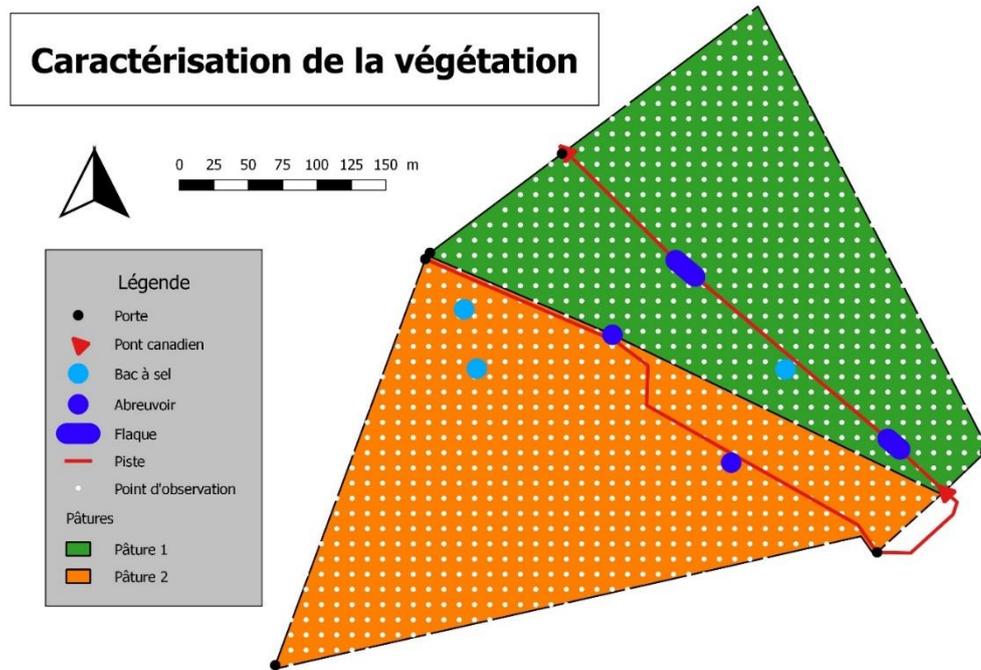


Figure 4 : carte de la caractérisation de la végétation

Les prélèvements végétaux qui constituent la seconde catégorie de données relatives à la végétation ont nécessité la fabrication d'un quadrat et de 12 exclos. Le matériel utilisé pour leur réalisation a été mis à disposition par la direction du ranch. Une photo de chaque modèle d'exclos mis au point est présentée en annexe 10.

Chacune des deux pâtures a été divisée en six parties de superficie équivalente. Dans chaque partie, quatre couples de prélèvements végétaux ont successivement été effectués avec un intervalle de 16 jours. Cette configuration a induit l'étude de 48 zones de prélèvement et a permis de rendre compte de l'évolution, à la fois spatiale et temporelle, de la végétation. Ces zones de prélèvement ont été choisies de façon à présenter une végétation homogène et représentative de la partie dans laquelle elles se trouvent. Une carte des zones de prélèvements de la végétation (figure 5) a été réalisée au moyen du logiciel QGIS afin de permettre leur visualisation.

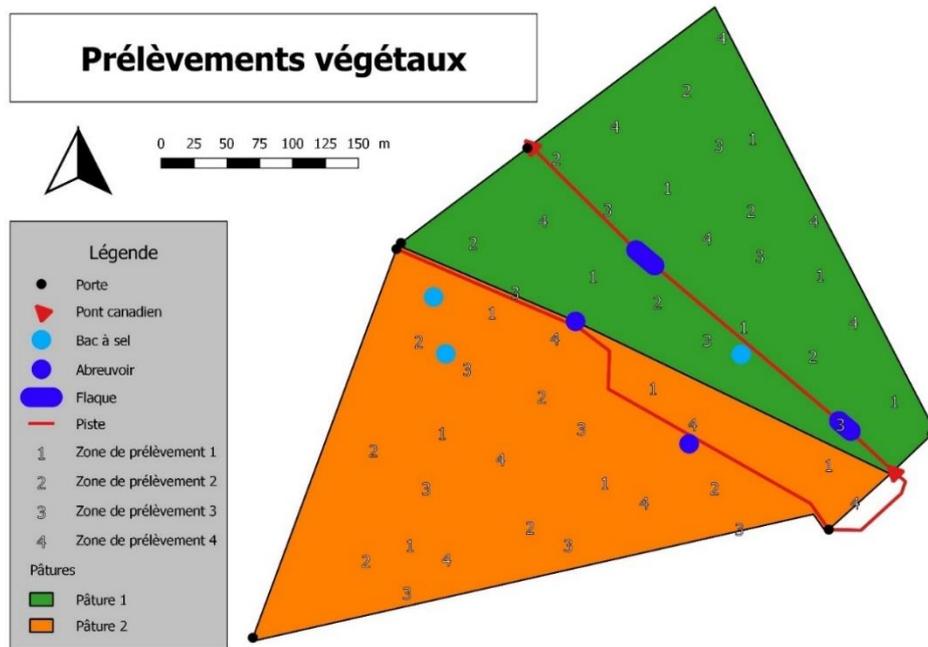


Figure 5 : carte des prélèvements végétaux

Dans chacune des zones, un premier prélèvement a été effectué à l'aide d'un couteau et d'un quadrat afin de récolter le matériel végétal de la strate herbacée présent sur un mètre carré. Un exclos a été placé à proximité immédiate pendant 16 jours afin d'empêcher l'accès du bétail à la végétation puis un second prélèvement a été effectué selon la même méthode. Le premier prélèvement a permis d'analyser la biomasse présente dans la pâture tandis que sa comparaison avec le second a permis d'étudier la production de biomasse.

Les prélèvements ont été pesés, séchés à l'air libre pendant 16 jours et à nouveau pesés. Un échantillon sec de 25 g a également été récupéré dans chaque prélèvement et ramené en Belgique. L'analyse en laboratoire de ces échantillons devait permettre d'évaluer leur teneur en protéines et leur valeur énergétique, mais elle n'a finalement pas été réalisée. Les données associées à chaque zone de prélèvements sont donc les suivantes :

- La date
- Les coordonnées géographiques
- La biomasse fraîche disponible (g/m^2) ;
- La biomasse sèche disponible (g/m^2) ;
- La teneur en matière sèche de la biomasse disponible (%)
- La production de biomasse fraîche ($\text{g/m}^2 \text{ j}$) ;
- La production de biomasse sèche ($\text{g/m}^2 \text{ j}$).
- La teneur en matière sèche de la production de biomasse (%)

Les données récoltées lors des prélèvements végétaux n'ont cependant pas fait l'objet d'un traitement statistique et sont les seules à ne pas avoir été exploitées dans le cadre de ce travail de fin d'études.

En complément des données relatives à la végétation, la pluviométrie a également été enregistrée pendant toute la durée de l'expérimentation. Les données ont été transmises par le personnel du ranch à partir du relevé journalier d'un pluviomètre situé à proximité immédiate des pâtures étudiées.

d. Traitement statistique des sources de biais et des observations directes du comportement

Une différence de gain quotidien moyen des bêtes ou de pluviométrie lors de la journée consacrée à leur observation pourrait être une source de variabilité dans les données relatives au comportement des animaux. Or leur influence ne fait pas partie de l'objet de ce travail de fin d'études et représente donc une source de biais. C'est pourquoi il a fallu vérifier que ces deux variables ne présentaient pas, en moyenne, de différences en fonction des variables explicatives étudiées dans le cadre de ce travail : la race de la bête et la pâture dans laquelle elle se trouve. Une base de données a été créée dans laquelle chaque bête de l'expérimentation s'est vu attribuer une valeur pour les variables suivantes :

- La pluviométrie lors de la journée d'observation en millimètres ;
- Le gain quotidien moyen en kilogrammes par jour (moyenne des mesures effectuées).

L'influence de la race de la bête et de la pâture dans laquelle elle se trouve sur les observations directes continues de son comportement a ensuite été évaluée. Pour ce faire, les données brutes ont été remaniées afin de créer une nouvelle base de données et chaque bête de l'expérimentation s'est vu attribuer une valeur pour les variables suivantes :

- Le nombre total d'attaques perpétrées sur d'autres animaux durant une journée type (comptage des observations) ;
- Le nombre d'attaques perpétrées sur des Ndamas durant une journée type (comptage des observations) ;
- Le nombre d'attaques perpétrées sur des zébus Goudalis durant une journée type (comptage des observations) ;
- Le nombre d'attaques perpétrées sur des Ndapols durant une journée type (comptage des observations) ;

- Le nombre d'attaques perpétrées sur des Zépols durant une journée type (comptage des observations) ;
- Le nombre d'attaques perpétrées sur des bêtes hors protocole durant une journée type (comptage des observations) ;
- L'intensité des attaques perpétrées sur une échelle de 0 à 3 (moyenne des observations)
- Le nombre total de ligneux broutés durant une journée type (comptage des observations) ;
- Le nombre de *Gardenia ternifolia subsp. jovis-tonantis* Welw. broutés durant une journée type (comptage des observations) ;
- Le nombre de *Sarcocephalus latifolius* Sm. broutés durant une journée type (comptage des observations) ;
- Le nombre de *Alchornea cordifolia* Schumach. & Thonn. broutés durant une journée type (comptage des observations) ;
- Le nombre de *Bridelia ferruginea* Benth. broutés durant une journée type (comptage des observations) ;
- Le nombre de ligneux appartenant à une autre espèce broutés durant une journée type (comptage des observations) ;
- Le nombre total de bouchées prélevées sur des ligneux durant une journée type (somme des observations) ;
- La hauteur des bouchées prélevées sur des ligneux en centimètres (moyenne des observations)
- Le nombre de flehms durant une journée type (comptage des observations) ;
- Le nombre d'animaux léchés durant une journée type (comptage des observations) ;
- Le nombre total de coups de langue donnés aux animaux léchés durant une journée type (somme des observations) ;
- Le nombre de périodes de meuglements durant une journée type (comptage des observations) ;
- Le nombre total de cris lors des périodes de meuglements durant une journée type (somme des observations) ;
- Le nombre de périodes passées allongé durant une journée type (comptage des observations) ;
- Le nombre total de minutes passées allongé durant une journée type (somme des observations).

Enfin et de façon similaire, l'influence de la race de la bête et de la pâture dans laquelle elle se trouve sur les observations directes discontinues de son comportement a été évaluée. Une nouvelle base de données a de nouveau été créée et chaque bête de l'expérimentation s'est également vu attribuer une valeur pour les variables suivantes :

- Le nombre total de bêtes présentes dans le groupe (moyenne des observations) ;
- La distance à laquelle se trouve la bête la plus proche en mètres (moyenne des observations) ;
- Le nombre d'heures passées seul durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec la bête qui est le plus souvent la bête la plus proche durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec au moins un Ndama durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec tous les Ndamas durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec au moins un zébu Goudali durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec tous les zébus Goudalis durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec au moins un Ndapol durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec tous les Ndapols durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec au moins un Zépol durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec tous les Zépols durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec au moins une bête ne faisant pas partie de l'expérimentation durant une journée type (comptage des observations) ;
- Le nombre d'heures passées avec toutes les bêtes ne faisant pas partie de l'expérimentation durant une journée type (comptage des observations) ;
- Le nombre d'heures pendant lequel la bête qui est le plus souvent la bête la plus proche est la bête la plus proche durant une journée type (comptage des observations) ;

- Le nombre d'heures pendant lequel un Ndama est la bête la plus proche durant une journée type (comptage des observations) ;
- Le nombre d'heures pendant lequel un zébu Goudali est la bête la plus proche durant une journée type (comptage des observations) ;
- Le nombre d'heures pendant lequel un Ndapol est la bête la plus proche durant une journée type (comptage des observations) ;
- Le nombre d'heures pendant lequel un Zépol est la bête la plus proche durant une journée type (comptage des observations) ;
- Le nombre d'heures pendant lequel une bête hors protocole est la bête la plus proche durant une journée type (comptage des observations).

Les moyennes et les écarts-types de chacune des variables citées ont été calculés pour les sept groupes d'animaux suivants :

- L'ensemble des animaux ;
- Les bêtes de la pâture 1 ;
- Les bêtes de la pâture 2 ;
- Les Ndamas ;
- Les zébus Goudalis ;
- Les Ndapols ;
- Les Zépols.

Une analyse de la variance a également été effectuée pour chacune de ces variables selon les modalités suivantes :

- Modèle : une ANOVA à deux facteurs croisés fixes ;
- Hypothèses nulles : l'absence d'interaction entre le facteur « pâture » et le facteur « race », l'égalité des moyennes entre les pâtures et l'égalité des moyennes entre les races ;
- Unité expérimentale : une bête ;
- Facteurs : la pâture (facteur fixe et qualitatif à deux objets) et la race (facteur fixe et qualitatif à quatre objets).

Cette partie a été réalisée dans l'environnement de développement MATLAB. Les descriptions des scripts du traitement statistique des sources de biais, des observations directes

continues et des observations directes discontinues sont respectivement disponibles en annexe 11, 12 et 13. Les scripts en eux-mêmes le sont en annexes 14 et 15 et 16.

e. Prétraitement des données provenant des iPhones

Les fichiers bruts contenant les données des capteurs des iPhones présentaient une masse de données trop importante et n'étaient pas exploitables en l'état. Une étape de prétraitement leur a donc été appliquée. Celle-ci a principalement répondu à deux objectifs. Tout d'abord, les données qui n'ont pas été utilisées pour le traitement statistique et géostatistique ont été supprimées : le temps, la latitude, la longitude, la vitesse et l'accélération gravitationnelle selon l'axe X sont les seules variables à avoir été conservées. Ensuite, les données, enregistrées à une fréquence de 100 Hz, ont été réduites à une seule mesure par seconde. Une carte (figure 6) a été réalisée au moyen du logiciel QGIS afin de faciliter la visualisation de ces données : chaque point correspond à la présence d'une bête pendant une seconde, toutes bêtes et toutes sessions d'enregistrement confondues.

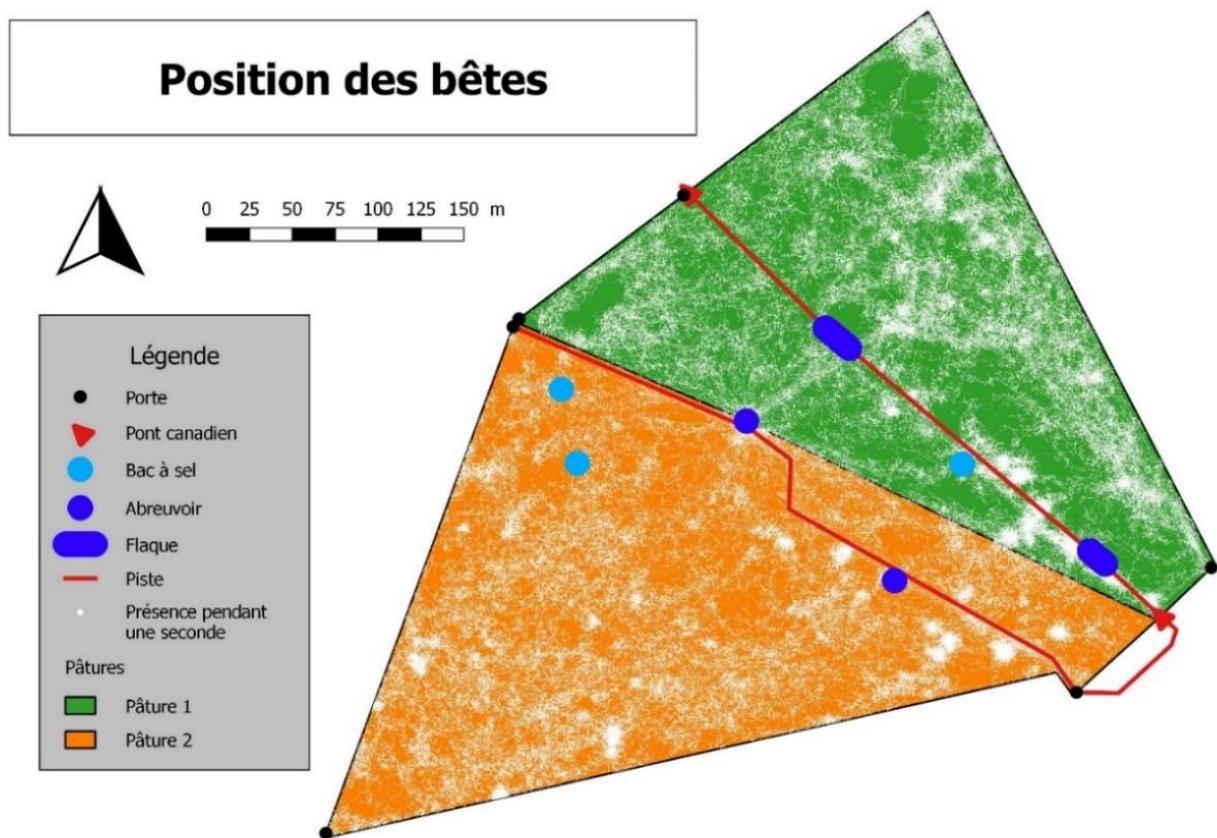


Figure 6 : carte de la position des bêtes

L'ensemble des données des différentes sessions d'enregistrement a été fusionné en une seule base de données. Celle-ci a ensuite été divisée en deux selon deux méthodes différentes.

La première a permis de séparer les données enregistrées le jour de celles enregistrées la nuit tandis que la seconde a permis de séparer les données relatives à des bêtes en train de se nourrir de celles relatives à des bêtes en train de ruminer, de se déplacer ou de se reposer.

Cette étape a également été réalisée dans l'environnement de développement MATLAB. Les descriptions de la fonction et du script qui ont été utilisés sont présentées en annexe 17 et 18. La fonction et le script en eux-mêmes le sont en annexe 19 et 20.

Le critère qui a été utilisé pour séparer les données enregistrées le jour de celles enregistrées la nuit est tout simplement l'heure associée à la mesure : de 06h30 à 18h30 pour le jour et de 18h30 à 06h30 pour la nuit.

Le critère qui a été utilisé pour séparer les données relatives à des bêtes en train de se nourrir (nommées données « grazing » dans la suite du document) de celles relatives à des bêtes en train de ruminer, de se déplacer ou de se reposer (nommées données « non grazing » dans la suite du document) est plus complexe. Il est basé sur l'accélération gravitationnelle selon l'axe X, qui correspond à la position de la tête de l'animal sur l'axe « dressé-baissé ». Le seuil qui a permis de séparer les données a été calculé à partir de la comparaison des valeurs prises par cette variable et des comportements des bêtes observables sur les séquences vidéo.

Les fichiers vidéo et les fichiers contenant les données des capteurs ont tout d'abord dû être synchronisés. Cette opération a été réalisée en deux temps. Une précision d'une à trois secondes a été obtenue en comparant les vitesses des bêtes sur les deux supports. Elle a ensuite été améliorée jusqu'au dixième de seconde en analysant les mouvements de tête des animaux.

L'accélération gravitationnelle selon l'axe X a alors été standardisée (moyenne égale à zéro et écart-type égal à un) pour chaque fichier afin de minimiser l'influence des différences de position de l'iPhone sur la tête de l'animal. En parallèle, les séquences vidéo ont été visionnées et l'emploi du temps de la bête (« grazing » – « non grazing » – bête non visible) a été déterminé et retranscrit seconde par seconde.

Différents seuils d'accélération gravitationnelle selon l'axe X ont été successivement testés avec un pas de 0,1 g. Pour chacun des seuils, la prédiction de « grazing » ou de « non grazing » donnée par l'iPhone pour chaque seconde de vidéo a été comparée au comportement réel de l'animal. Les pourcentages de vrais positifs par rapport aux faux positifs et de vrais négatifs par rapport aux faux négatifs ont ainsi été calculés pour chaque seuil et chaque animal. Finalement, les médianes de ces deux pourcentages ont été calculées pour chaque seuil. Le seuil

associé au maximum atteint par la moyenne des deux médianes a été retenu comme le seuil d'accélération gravitationnelle selon l'axe X commun à toutes les bêtes et a permis de départager les données « grazing » des données « non grazing ». Ce seuil est de 0,5 g. Le tableau 1 présente la moyenne de la médiane des pourcentages de vrais positifs et de la médiane des pourcentages de vrais négatifs associée à chaque seuil.

Tableau 1 : détermination du seuil d'accélération gravitationnelle selon l'axe X qui permet de séparer les données « grazing » des données « non grazing »

Seuil (g)	0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1,0
Prédictions correctes (%)	77,7	80,0	81,3	83,3	83,7	83,9	83,4	81,8	78,7	75,9	71,6

Vert : seuil d'accélération gravitationnelle selon l'axe X qui a été retenu pour le prétraitement des données

f. Traitement statistique de la vitesse des bêtes et de leur accès à l'eau et au sel

L'influence de la race de la bête et de la pâture dans laquelle elle se trouve sur la vitesse des bêtes a été évaluée. Pour ce faire, les données obtenues grâce aux iPhones puis modifiées lors de l'étape de prétraitement ont été remaniées afin de créer une nouvelle base de données. Chaque bête de l'expérimentation s'est ainsi vu attribuer quatre valeurs pour chaque nouvelle variable : une pour le jour, une pour la nuit, une pour les périodes de « grazing » et une pour les périodes de « non grazing ». Les variables suivantes ont été créées :

- La vitesse moyenne en mètres par heure (moyenne des données) ;
- Le pourcentage du temps passé entre 0 et 0,5 m/s (comptage des données) ;
- Le pourcentage du temps passé entre 0,5 et 2 m/s (comptage des données) ;
- Le pourcentage du temps passé à plus de 2 m/s (comptage des données).

De façon similaire, l'influence de la race de la bête et de la pâture dans laquelle elle se trouve sur la distance la séparant des points d'eau et des bacs à sel a été évaluée. Les positions des bêtes, également obtenues grâce au prétraitement des données provenant des iPhones, ont été couplées à celles utilisées pour la carte de l'agencement des pâtures afin d'ajouter deux variables à la base de données :

- Le pourcentage du temps passé à moins de 10 m d'un bac à sel (comptage de données) ;
- Le pourcentage du temps passé à moins de 10 m d'un point d'eau (comptage de données).

Les moyennes et les écarts-types de chaque variable ont été calculés à quatre reprises (pour le jour, pour la nuit, pour les périodes de « grazing » et pour les périodes de « non grazing ») pour les sept groupes d'animaux suivants :

- L'ensemble des animaux ;
- Les bêtes de la pâture 1 ;
- Les bêtes de la pâture 2 ;
- Les Ndamas ;
- Les zébus Goudalis ;
- Les Ndapols ;
- Les Zépols.

Une analyse de la variance a également été effectuée, également à quatre reprises, pour chacune des variables selon les modalités suivantes :

- Modèle : une ANOVA à deux facteurs croisés fixes ;
- Hypothèses nulles : l'absence d'interaction entre le facteur « pâture » et le facteur « race », l'égalité des moyennes entre les pâtures et l'égalité des moyennes entre les races ;
- Unité expérimentale : une bête ;
- Facteurs : la pâture (facteur fixe et qualitatif à deux objets) et la race (facteur fixe et qualitatif à quatre objets).

Cette partie a été réalisée dans l'environnement de développement MATLAB. La description du script est présentée en annexe 21 tandis que le script en lui-même l'est en annexe 22.

g. Traitement géostatistique de la position des bêtes et de ses liens avec la végétation

Les liens entre la position des bêtes et la description du couvert végétal ont également été étudiés dans le cadre de ce travail de fin d'études. Cette thématique a impliqué la création d'une nouvelle base de données. Quatre valeurs (une pour le jour, une pour la nuit, une pour les périodes de « grazing » et une pour les périodes de « non grazing ») ont ainsi été attribuées à chaque bête pour les variables suivantes :

- La corrélation entre la présence des bêtes et celle des graminées rampantes ou gazonnantes ;
- La corrélation entre la présence des bêtes et celle des graminées cespitueuses de moins de 1 m de haut ;
- La corrélation entre la présence des bêtes et celle des graminées cespitueuses de plus de 1 m de haut ;

- La corrélation entre la présence des bêtes et celle de *Calopogonium mucunoides* Desv. ;
- La corrélation entre la présence des bêtes et celle de *Sida cf. acuta* Burm.f. ;
- La corrélation entre la présence des bêtes et celle de *Mitracarpus hirtus* L. ;
- La corrélation entre la présence des bêtes et celle de la végétation herbacée non reprise dans les 6 critères précédents ;
- La corrélation entre la présence des bêtes et celle de la végétation arbustive à l'exception de *C. odorata* ;
- La corrélation entre la présence des bêtes et celle de *C. odorata* ;
- La corrélation entre la présence des bêtes et celle des abris contre le soleil et la pluie.

Le calcul de ces corrélations a nécessité un traitement géostatistique composé de plusieurs étapes. La structure du raisonnement a été présentée au moyen d'un organigramme (figure 7) avant que chaque étape ne soit détaillée.

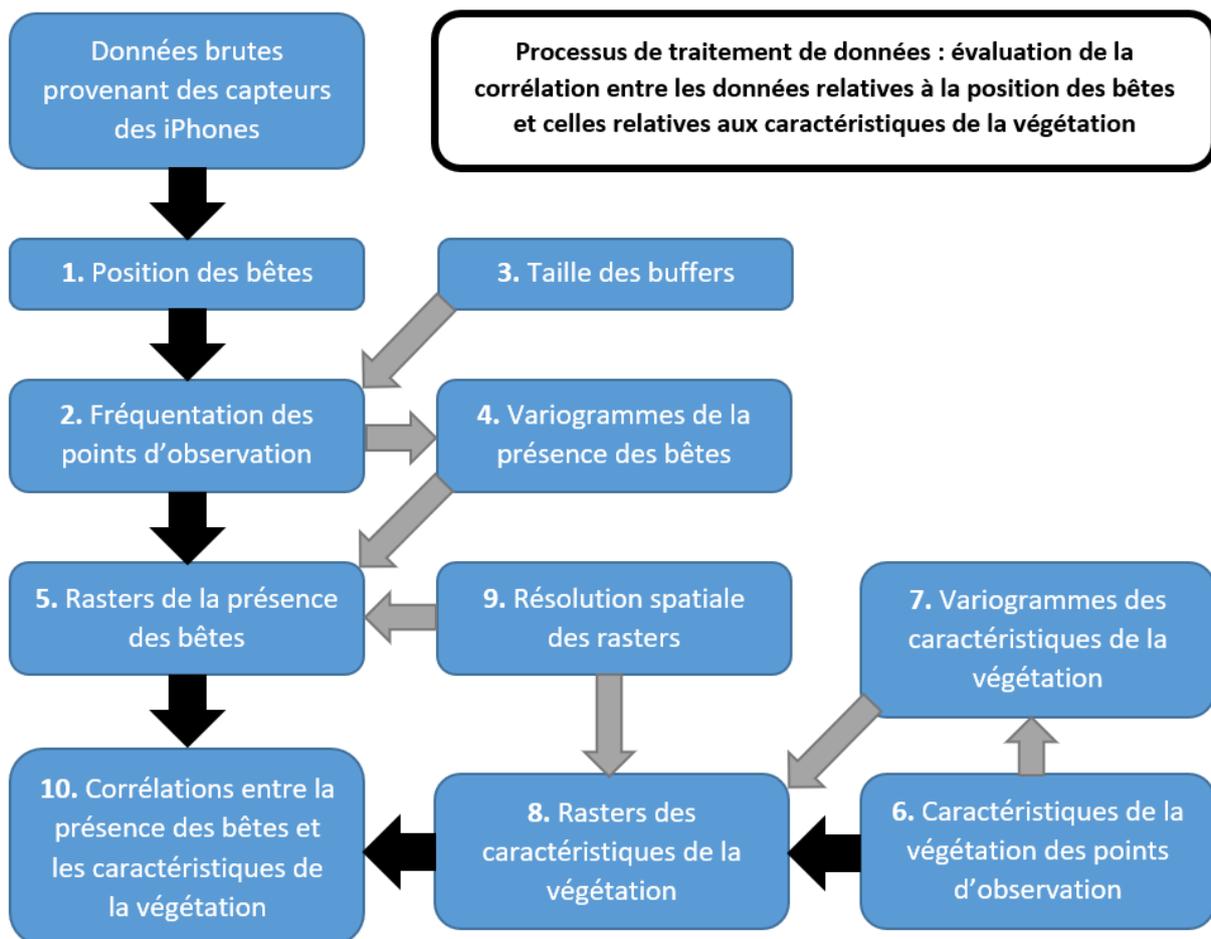


Figure 7 : organigramme du processus de traitement de données qui permet d'évaluer la corrélation entre les données relatives à la position des bêtes et celles relatives aux caractéristiques de la végétation

Les positions des bêtes (1), obtenues via le prétraitement des données des capteurs des iPhones, sont constituées d'une liste de coordonnées géographiques et ne peuvent pas être exploitées en l'état. Afin de pouvoir comparer la répartition des bêtes sur la pâture à celle du couvert végétal, le nombre de secondes passées dans une zone tampon (le buffer) autour de chacun des points utilisés pour la caractérisation de la végétation a été comptabilisé. Ces nombres ont ensuite été inscrits dans une nouvelle base de données nommée « fréquentation des points d'observation par les bêtes (2) ».

La taille des buffers (3) utilisés pour réaliser ce comptage a été déterminée de la façon suivante. Le buffer correspondant à une grille sans vide ni recouvrement autour des points a été choisi comme buffer « a priori ». L'effet de la diminution et de l'augmentation de sa taille sur les corrélations obtenues à la fin du processus du traitement des données a été analysé. L'objectif était de rechercher une taille de buffer réaliste autour de laquelle les valeurs des corrélations présentaient une évolution stable pour un maximum de jeux de données. C'est finalement le buffer initialement envisagé qui a été utilisé pour le traitement des données.

Avant de pouvoir continuer à traiter les données de la fréquentation des points d'observation par les bêtes, il a fallu évaluer leur autocorrélation spatiale⁵ au moyen de variogrammes (4). Les variogrammes expérimentaux correspondant aux données ont été ajustés au moyen d'une méthode de lissage automatique afin de les rendre exploitables. Différents essais ont été réalisés afin de fixer les paramètres de la fonction de lissage. Ceux qui entraînaient des incohérences entre les variogrammes expérimentaux et variogrammes ajustés ou entre les données brutes et les rasters⁶ obtenus à la fin du processus de traitement des données ont été éliminés. Les variogrammes expérimentaux se sont cependant montrés trop irréguliers pour être lissés individuellement pour chaque bête. Ils ont donc été groupés en huit catégories : les données de jour, de nuit, « grazing » et « non grazing » pour la pâture 1 et ces mêmes catégories de données pour la pâture 2. Les variogrammes associés à chacune des catégories sont présentés en annexe 23.

⁵ L'autocorrélation spatiale est un concept qui permet de quantifier le fait que des points proches présentent des caractéristiques similaires. Un variogramme expérimental décrit l'autocorrélation spatiale tandis qu'un variogramme ajusté la modélise (ArcGIS, 2016a).

⁶ Les rasters sont des matrices de pixels utilisées pour présenter des informations géographiques (ArcGIS, 2016b).

Les variogrammes ont ensuite été utilisés afin de réaliser un krigeage universel⁷ des données de la fréquentation des points d'observation par les bêtes. Ce krigeage a permis d'obtenir des rasters (5) qui présentent les niveaux de probabilité de présence des bêtes de manière continue sur l'entièreté de la pâture tout en prenant en compte les biais liés à l'autocorrélation spatiale. Les rasters de présence associés à chaque bête sont présentés en annexe 24.

Les dix critères utilisés pour caractériser la végétation des points d'observation (6) ont été directement utilisés pour le traitement géostatistique. En effet, une analyse en composantes principales a été effectuée sur ces critères mais ne s'est pas révélée pertinente. Les données brutes ont donc été préférées aux résultats de celle-ci. La matrice de corrélation des critères de la végétation ainsi que la table du pourcentage d'information reprise par chaque axe de l'analyse en composantes principales sont présentées en annexe 25.

La même méthode que celle utilisée pour la présence des bêtes a été appliquée pour définir les paramètres de la méthode de lissage automatique des variogrammes expérimentaux des caractéristiques de la végétation (7). Ensuite, un krigeage universel a également été réalisé pour chaque critère d'observation grâce à l'utilisation des variogrammes ajustés. Ce krigeage a permis d'obtenir les rasters des caractéristiques de la végétation (8). Les variogrammes sont présentés en annexe 26 tandis que les rasters le sont en annexe 27.

La réalisation d'un krigeage implique de déterminer au préalable la résolution spatiale des rasters (9) qui en résultent. Celle-ci doit être identique pour les rasters de la présence des bêtes et pour les rasters des caractéristiques de la végétation afin de permettre leur comparaison. Tout une gamme de résolution a été testée sur différents jeux de données afin d'observer l'influence de l'augmentation et de la diminution de celle-ci sur les corrélations obtenues à la fin du processus de traitement de ces données. Une résolution dont les variations n'entraînaient pas de comportement instable des corrélations et qui demandait une puissance de calcul raisonnable a été sélectionnée.

Le coefficient de corrélation de Pearson a enfin pu être calculé pour chaque couple de rasters. La série de rasters associés à la présence d'une bête pour une catégorie de donnée (jour, nuit, « grazing » ou « non grazing ») et celle des rasters associés à un critère d'observation du

⁷ Le krigeage est une procédure géostatistique qui permet d'estimer une variable sur une surface à partir des valeurs de cette variable associées à des points dispersés. Le krigeage universel est une forme de krigeage qui part du principe que la moyenne de la variable n'est pas constante et peut être modélisée : la surface estimée prend ainsi en compte l'effet de la tendance de la variable tout en évitant celui de l'autocorrélation (ArcGIS 2016a).

couvert végétal ont ainsi été croisées. Ce sont donc 640 corrélations entre la présence des bêtes et les caractéristiques de la végétation (10) qui ont été calculées.

Cette nouvelle base de données a été utilisée pour évaluer l'influence de la race de la bête et de la pâture dans laquelle elle se trouve sur les liens entre sa position et la végétation de la pâture. Les moyennes et les écarts-types de chaque variable « corrélation entre la présence des bêtes et une des caractéristiques de la végétation » ont été calculés à quatre reprises (pour le jour, pour la nuit, pour les périodes de « grazing » et pour les périodes de « non grazing ») pour les sept groupes d'animaux suivants :

- L'ensemble des animaux ;
- Les bêtes de la pâture 1 ;
- Les bêtes de la pâture 2 ;
- Les Ndamas ;
- Les zébus Goudalis ;
- Les Ndapols ;
- Les Zépols.

Une analyse de la variance a alors été effectuée, également à quatre reprises, pour ces mêmes variables selon les modalités suivantes :

- Modèle : une ANOVA à deux facteurs croisés fixes ;
- Hypothèses nulles : l'absence d'interaction entre le facteur « pâture » et le facteur « race », l'égalité des moyennes entre les pâtures et l'égalité des moyennes entre les races ;
- Unité expérimentale : une bête ;
- Facteurs : la pâture (facteur fixe et qualitatif à deux objets) et la race (facteur fixe et qualitatif à quatre objets).

La totalité de la partie « traitement géostatistique de la position des bêtes et de ses liens avec la végétation » a été réalisée dans l'environnement de développement RStudio. La description du script qui a été utilisé est présentée en annexe 28 tandis que le script en lui-même l'est en annexe 29.

8. Résultats

a. Sources de biais et observations directes du comportement

Les tableaux 2, 3 et 4 présentent respectivement les résultats du traitement statistique des sources de biais, des observations directes continues et des observations directes discontinues du comportement pour toutes les variables abordées dans la partie « matériel et méthode » de ce travail de fin d'études. Les valeurs suivantes sont associées à chacune des variables :

- La moyenne pour l'ensemble des bêtes de l'expérimentation ;
- L'écart-type pour l'ensemble des bêtes de l'expérimentation ;
- La p-valeur associée à l'hypothèse nulle « absence d'interaction entre le facteur pâture et le facteur race » ;
- La p-valeur associée à l'hypothèse nulle « égalité des moyennes selon le facteur pâture » ;
- La p-valeur associée à l'hypothèse nulle « égalité des moyennes selon le facteur race ».

Tableau 2 : résultats du traitement statistique des sources de biais

Sources de biais	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Pluviométrie lors de la journée d'observation (mm)	7,7 ± 13,8	0,48	0,43	0,62
Gain quotidien moyen (kg/j)	0,071 ± 0,385	0,59	0,66	0,50

Tableau 3 : résultats du traitement statistique des observations directes continue

Observations directes continues du comportement	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Attaques perpétrées sur d'autres animaux :				
• Nombre total	2,1 ± 2,2	0,76	0,86	0,44
• Nombre de Ndamas	0,6 ± 0,7	0,60	1,00	0,01
• Nombre de Zébus Goudalis	0,5 ± 1,1	0,31	0,65	0,37
• Nombre de Ndapols	0,4 ± 0,9	0,91	0,68	0,30
• Nombre de Zépols	0,4 ± 1,0	0,99	0,87	0,33
• Nombre de bêtes hors protocole	0,2 ± 0,4	0,80	0,58	0,10
• Intensité des attaques (0-3)	1,4 ± 0,4	0,62	0,75	0,14
Broutage de ligneux :				
• Nombre total	2,7 ± 2,3	0,65	0,08	0,19
• Nombre de <i>Gardenia ternifolia</i>	1,4 ± 1,5	0,83	0,24	0,42
• Nombre de <i>Sarcocephalus latifolius</i>	0,3 ± 0,4	0,89	0,67	0,35
• Nombre de <i>Alchornea cordifolia</i>	0,1 ± 0,3	0,60	0,20	0,60
• Nombre de <i>Bridelia ferruginea</i>	0,9 ± 1,1	0,62	0,28	0,11

• Nombre de ligneux d'une autre espèce	0,0 ± 0,0			
• Nombre total de bouchées	9,8 ± 11,2	0,49	0,15	0,50
• Hauteur des bouchées (cm)	77,2 ± 27,1	0,44	0,94	0,11
Nombre de flehms	1,1 ± 2,7	0,21	0,26	0,03
Léchage d'autres animaux :				
• Nombre d'animaux léchés	1,8 ± 2,3	0,86	0,23	0,02
• Nombre total de coups de langue donnés	76,8 ± 153,4	0,67	0,23	0,16
Périodes de meuglements :				
• Nombre de périodes	2,0 ± 4,6	0,25	0,23	0,03
• Nombre total de cris	19,8 ± 54,4	0,30	0,33	0,17
Périodes passées allongé :				
• Nombre de périodes	2,3 ± 1,9	0,52	0,77	0,00
• Temps total (minutes)	35,3 ± 34,2	0,81	0,74	0,00

Vert : p-valeur Race < 0,1 ; Jaune : p-valeur Pâtture < 0,1

Tableau 4 : résultats du traitement statistique des observations directes discontinues

Observations directes discontinues du comportement	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâtture	p-valeur Race
Nombre total de bêtes présentes dans le groupe	5,0 ± 1,0	0,41	0,81	0,06
Distance à laquelle se trouve la bête la plus proche (m)	4,4 ± 1,3	0,45	0,34	0,00
Temps passé seul (heures)	1,5 ± 2,1	0,19	0,27	0,00
Temps passé avec chaque catégorie de bêtes (heures) :				
• La bête le plus souvent la plus proche	8,4 ± 2,5	0,23	0,39	0,00
• Au moins un Ndama	5,3 ± 1,5	0,66	0,86	0,66
• Tous les Ndamas	3,9 ± 1,6	0,42	0,71	0,08
• Au moins un zébu Goudali	8,4 ± 2,7	0,45	0,49	0,00
• Tous les zébus Goudalis	7,6 ± 2,7	0,29	0,81	0,00
• Au moins un Ndapol	7,3 ± 1,9	0,86	0,20	0,12
• Tous les Ndapols	5,2 ± 1,5	0,70	0,19	0,15
• Au moins un Zépol	8,2 ± 2,7	0,52	0,80	0,03
• Tous les Zépols	6,9 ± 2,5	0,58	0,88	0,04
• Au moins une bête hors protocole	3,0 ± 2,6	0,86	0,50	0,77
• Toutes les bêtes hors protocole	1,6 ± 2,2	0,82	0,19	0,55
Temps pendant lequel une bête de chaque catégorie est la bête la plus proche (heures) :				
• La bête le plus souvent la plus proche	3,5 ± 1,5	0,43	0,10	0,00
• Un Ndama	1,3 ± 1,0	0,67	0,79	0,18
• Un zébu Goudali	3,3 ± 2,0	0,46	0,93	0,01
• Un Ndapol	1,9 ± 1,1	0,47	0,65	0,00
• Un Zépol	3,2 ± 1,3	0,58	0,77	0,15
• Une bête hors protocole	0,8 ± 0,8	0,48	0,92	0,35

Vert : p-valeur Race < 0,1

Le tableau 5 contient les moyennes et les écarts-types associés à chacune des races pour les variables dont l'hypothèse nulle « égalité des moyennes selon le facteur race » a été rejetée avec une p-valeur de 0,1 (en vert dans les tableaux précédents).

Tableau 5 : résultats complémentaires du traitement statistique des observations directes du comportement : moyennes et écarts-types par races

Variabiles pour lesquelles l'hypothèse nulle « égalité des moyennes selon le facteur race » a été rejetée	Ndama : Moyenne ± Ecart-type	Zébu Goudali : Moyenne ± Ecart-type	Ndapol : Moyenne ± Ecart-type	Zépol : Moyenne ± Ecart-type
Attaques perpétrées sur d'autres animaux :				
• Nombre de Ndamas	0,5 ± 0,5	0,0 ± 0,0	0,5 ± 0,5	1,3 ± 0,8
• Nombre de bêtes hors protocole	0,5 ± 0,5	0,0 ± 0,0	0,0 ± 0,0	0,3 ± 0,4
Nombre de flehmens	3,8 ± 4,5	0,0 ± 0,0	0,5 ± 0,5	0,3 ± 0,4
Léchage d'autres animaux :				
• Nombre d'animaux léchés	0,0 ± 0,0	4,8 ± 1,9	0,5 ± 0,9	1,8 ± 1,5
Périodes de meuglements :				
• Nombre de périodes	6,5 ± 7,4	0,0 ± 0,0	1,5 ± 1,5	0,0 ± 0,0
Périodes passées allongé :				
• Nombre de périodes	0,8 ± 0,8	3,8 ± 0,8	1,5 ± 1,7	3,3 ± 1,9
• Temps total (minutes)	17,5 ± 19,2	52,0 ± 23,8	8,3 ± 8,3	63,3 ± 39,8
Nombre total de bêtes présentes dans le groupe	4,6 ± 1,5	5,7 ± 0,4	4,6 ± 0,5	5,3 ± 0,3
Distance à laquelle se trouve la bête la plus proche (m)	5,9 ± 0,4	3,3 ± 1,2	5,2 ± 0,3	3,4 ± 0,4
Temps passé seul (heures)	3,1 ± 3,2	0,1 ± 0,1	2,5 ± 1,0	0,3 ± 0,2
Temps passé avec chaque catégorie de bêtes (heures) :				
• La bête le plus souvent la plus proche	5,9 ± 2,6	10,9 ± 0,8	7,0 ± 0,8	9,9 ± 0,8
• Au moins un zébu Goudali	5,8 ± 2,4	10,5 ± 1,2	6,6 ± 1,3	10,6 ± 0,3
• Tous les zébus Goudalis	4,9 ± 2,3	10,5 ± 1,2	6,0 ± 1,2	9,1 ± 0,8
• Au moins un Zépol	5,7 ± 2,8	11,0 ± 0,9	6,8 ± 1,2	9,1 ± 1,6
• Tous les Zépols	4,4 ± 2,1	8,5 ± 1,0	5,4 ± 1,0	9,1 ± 1,6
Temps pendant lequel une bête de chaque catégorie est la bête la plus proche (heures) :				
• La bête le plus souvent la plus proche	2,2 ± 0,8	5,5 ± 0,7	2,2 ± 0,2	3,9 ± 0,6
• Un zébu Goudali	1,7 ± 1,3	4,7 ± 1,8	2,0 ± 1,0	4,9 ± 1,2
• Un Ndapol	3,0 ± 1,5	1,2 ± 0,4	1,5 ± 0,4	2,0 ± 0,8

Les moyennes et les écarts-types associés aux autres variables et aux autres groupes d'animaux sont disponibles en annexe 30.

b. Vitesse des bêtes et accès à l'eau et au sel

Les tableaux 6, 7, 8 et 9 présentent respectivement les résultats du traitement statistique de la vitesse des bêtes et de leur accès à l'eau et au sel pour les données de jour, de nuit, lorsque les bêtes se nourrissent (grazing) et lorsqu'elles ruminent, se déplacent ou se reposent (non grazing). Les valeurs suivantes ont été associées à chacune des variables abordées dans la partie « matériel et méthode » de ce travail de fin d'études :

- La moyenne pour l'ensemble des bêtes de l'expérimentation ;
- L'écart-type pour l'ensemble des bêtes de l'expérimentation ;
- La p-valeur associée à l'hypothèse nulle « absence d'interaction entre le facteur pâture et le facteur race » ;
- La p-valeur associée à l'hypothèse nulle « égalité des moyennes selon le facteur pâture » ;
- La p-valeur associée à l'hypothèse nulle « égalité des moyennes selon le facteur race ».

Tableau 6 : résultats du traitement statistique des vitesses des bêtes et de leur accès à l'eau et au sel pour les données de jour

DONNEES DE JOUR	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Vitesse moyenne (m/h)	427 ± 85	0,53	0,31	0,34
Temps passé entre 0 et 0,5 m/s (%)	91,32 ± 1,59	0,88	0,14	0,20
Temps passé entre 0,5 et 2 m/s (%)	8,64 ± 1,57	0,88	0,14	0,20
Temps passé à plus de 2 m/s (%)	0,04 ± 0,04	0,96	0,99	0,23
Temps passé à moins de 10 m d'un bac à sel (%)	1,04 ± 1,19	0,69	0,17	0,67
Temps passé à moins de 10 m d'un point d'eau (%)	2,77 ± 1,73	0,58	0,77	0,56

Tableau 7 : résultats du traitement statistique des vitesses des bêtes et de leur accès à l'eau et au sel pour les données de nuit

DONNEES DE NUIT	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Vitesse moyenne (m/h)	96 ± 40	0,19	0,44	0,24
Temps passé entre 0 et 0,5 m/s (%)	98,32 ± 0,65	0,20	0,36	0,49
Temps passé entre 0,5 et 2 m/s (%)	1,67 ± 0,64	0,20	0,35	0,48
Temps passé à plus de 2 m/s (%)	0,01 ± 0,01	0,40	0,89	0,76
Temps passé à moins de 10 m d'un bac à sel (%)	1,40 ± 3,86	1,00	0,93	0,23
Temps passé à moins de 10 m d'un point d'eau (%)	4,47 ± 9,18	0,95	0,47	0,26

Tableau 8 : résultats du traitement statistique des vitesses des bêtes et de leur accès à l'eau et au sel pour les données "grazing"

DONNEES « GRAZING »	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Vitesse moyenne (m/h)	307 ± 95	0,49	0,33	0,32
Temps passé entre 0 et 0,5 m/s (%)	95,25 ± 1,71	0,58	0,21	0,27
Temps passé entre 0,5 et 2 m/s (%)	4,73 ± 1,70	0,58	0,21	0,27
Temps passé à plus de 2 m/s (%)	0,02 ± 0,01	0,55	0,35	0,37
Temps passé à moins de 10 m d'un bac à sel (%)	1,10 ± 1,54	0,60	0,32	0,55
Temps passé à moins de 10 m d'un point d'eau (%)	1,37 ± 0,86	0,77	0,46	0,46

Tableau 9 : résultats du traitement statistique des vitesses des bêtes et de leur accès à l'eau et au sel pour les données "non grazing"

DONNEES « NON GRAZING »	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Vitesse moyenne (m/h)	256 ± 58	0,46	0,12	0,17
Temps passé entre 0 et 0,5 m/s (%)	94,14 ± 1,17	0,71	0,10	0,45
Temps passé entre 0,5 et 2 m/s (%)	5,82 ± 1,16	0,73	0,11	0,50
Temps passé à plus de 2 m/s (%)	0,03 ± 0,03	0,82	0,92	0,16
Temps passé à moins de 10 m d'un bac à sel (%)	1,32 ± 2,46	1,00	0,93	0,22
Temps passé à moins de 10 m d'un point d'eau (%)	4,84 ± 7,14	0,94	0,49	0,27

L'annexe 30 comprend les moyennes et les écarts-types de chaque variable pour tous les groupes d'animaux.

c. Position des bêtes et liens avec la végétation

Les tableaux 10, 11, 12 et 13 présentent respectivement les résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation pour les données de jour, de nuit, lorsque les bêtes se nourrissent (grazing) et lorsqu'elles ruminent, se déplacent ou se reposent (non grazing). Les valeurs suivantes ont été associées à chacune des variables abordées dans la partie « matériel et méthode » de ce travail de fin d'études :

- La moyenne pour l'ensemble des bêtes de l'expérimentation ;
- L'écart-type pour l'ensemble des bêtes de l'expérimentation ;
- La p-valeur associée à l'hypothèse nulle « absence d'interaction entre le facteur pâture et le facteur race » ;
- La p-valeur associée à l'hypothèse nulle « égalité des moyennes selon le facteur pâture » ;
- La p-valeur associée à l'hypothèse nulle « égalité des moyennes selon le facteur race ».

Cependant, l'hypothèse nulle « absence d'interaction entre le facteur pâture et le facteur race » a été rejetée pour quatre des variables (en rouge dans les tableaux) :

- La corrélation entre la présence des bêtes et celle des graminées rampantes ou gazonnantes de nuit ;
- La corrélation entre la présence des bêtes et *C. odorata* de nuit ;
- La corrélation entre la présence des bêtes et celle des graminées rampantes ou gazonnantes pour les données « non grazing » ;
- La corrélation entre la présence des bêtes et *C. odorata* pour les données « non grazing ».

La p-valeur associée à l'hypothèse nulle « égalité des moyennes selon le facteur race » n'est, dans ce cas, pas présentée et les moyennes et les écarts-types ainsi que les p-valeurs associées à l'hypothèse nulle « égalité des moyennes selon le facteur pâture » sont présentées indépendamment pour chacune des races.

Tableau 10 : résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation pour les données de jour

DONNEES DE JOUR	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation :				
• Graminées rampantes ou gazonnantes	0,17 ± 0,09	0,24	0,35	0,79
• Graminées cespiteuses de moins de 1 m de haut	-0,02 ± 0,13	0,80	0,00	0,96
• Graminées cespiteuses de plus de 1 m de haut	-0,03 ± 0,12	0,57	0,00	0,84
• <i>Calopogonium mucunoides</i>	-0,12 ± 0,08	0,39	0,95	0,49
• <i>Sida cf. acuta</i>	-0,05 ± 0,04	0,98	0,85	0,87
• <i>Mitracarpus hirtus</i>	-0,03 ± 0,09	0,64	0,00	0,51
• Végétation herbacée hors 6 critères précédents	0,00 ± 0,10	0,41	0,06	0,13
• Végétation arbustive à l'exception de <i>C. odorata</i>	0,06 ± 0,06	0,44	0,44	0,35
• <i>C. odorata</i>	-0,03 ± 0,12	0,84	0,53	0,29
• Abris contre le soleil et la pluie	0,06 ± 0,09	0,69	0,28	0,32

Vert : p-valeur Pâture < 0,1

Tableau 11 : résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation pour les données de nuit

DONNEES DE NUIT	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation :				
• Graminées rampantes ou gazonnantes	0,04 ± 0,06	0,05		
→ Pour la race Ndama	0,05 ± 0,03		0,07	
→ Pour la race zébu Goudali	0,04 ± 0,06		0,34	
→ Pour la race Ndapol	0,03 ± 0,07		0,06	
→ Pour la race Zépol	0,05 ± 0,08		0,03	
• Graminées cespiteuses de moins de 1 m de haut	-0,01 ± 0,05	0,43	0,00	0,85
• Graminées cespiteuses de plus de 1 m de haut	0,01 ± 0,06	0,76	0,01	0,52
• <i>Calopogonium mucunoides</i>	0,00 ± 0,05	0,14	0,04	0,08
• <i>Sida cf. acuta</i>	-0,05 ± 0,03	0,50	0,00	0,98
• <i>Mitracarpus hirtus</i>	-0,06 ± 0,10	0,12	0,00	0,44
• Végétation herbacée hors 6 critères précédents	0,00 ± 0,05	0,94	0,01	0,74
• Végétation arbustive à l'exception de <i>C. odorata</i>	0,02 ± 0,04	0,82	0,33	0,71
• <i>C. odorata</i>	-0,05 ± 0,04	0,04		
→ Pour la race Ndama	-0,04 ± 0,02		0,87	
→ Pour la race zébu Goudali	-0,06 ± 0,03		0,17	
→ Pour la race Ndapol	-0,04 ± 0,05		0,01	
→ Pour la race Zépol	-0,04 ± 0,06		0,21	
• Abris contre le soleil et la pluie	-0,03 ± 0,03	0,60	0,03	0,40

Rouge : p-valeur Interaction < 0,1 ; Vert : p-valeur Pâture < 0,1 ; Jaune : p-valeur Race < 0,1

Tableau 12 : résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation pour les données "grazing"

DONNEES « GRAZING »	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation :				
• Graminées rampantes ou gazonnantes	0,16 ± 0,13	0,38	0,09	0,84
• Graminées cespiteuses de moins de 1 m de haut	-0,01 ± 0,13	0,30	0,01	0,85
• Graminées cespiteuses de plus de 1 m de haut	0,00 ± 0,12	0,91	0,04	0,90
• <i>Calopogonium mucunoides</i>	-0,07 ± 0,09	0,28	0,67	0,45
• <i>Sida cf. acuta</i>	-0,09 ± 0,06	0,77	0,63	0,66
• <i>Mitracarpus hirtus</i>	-0,05 ± 0,11	0,78	0,32	0,83
• Végétation herbacée hors 6 critères précédents	0,00 ± 0,11	0,95	0,13	0,39
• Végétation arbustive à l'exception de <i>C. odorata</i>	0,07 ± 0,07	0,26	0,95	0,99
• <i>C. odorata</i>	-0,03 ± 0,10	0,75	0,66	0,61
• Abris contre le soleil et la pluie	0,00 ± 0,07	0,96	0,12	0,89

Vert : p-valeur Pâture < 0,1

Tableau 13 : résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation pour les données "non grazing"

DONNEES « NON GRAZING »	Moyenne ± Ecart-type	p-valeur Interaction	p-valeur Pâture	p-valeur Race
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation :				
• Graminées rampantes ou gazonnantes	0,06 ± 0,06	0,04		
→ Pour la race Ndama	0,07 ± 0,03		0,45	
→ Pour la race zébu Goudali	0,06 ± 0,07		0,26	
→ Pour la race Ndapol	0,04 ± 0,06		0,05	
→ Pour la race Zépol	0,07 ± 0,08		0,05	
• Graminées cespiteuses de moins de 1 m de haut	-0,01 ± 0,04	0,25	0,05	0,95
• Graminées cespiteuses de plus de 1 m de haut	0,02 ± 0,05	0,51	0,03	0,53
• <i>Calopogonium mucunoides</i>	-0,01 ± 0,05	0,28	0,03	0,12
• <i>Sida cf. acuta</i>	-0,05 ± 0,03	0,72	0,00	0,91
• <i>Mitracarpus hirtus</i>	-0,07 ± 0,10	0,41	0,00	0,51
• Végétation herbacée hors 6 critères précédents	0,00 ± 0,05	0,66	0,02	0,97
• Végétation arbustive à l'exception de <i>C. odorata</i>	0,02 ± 0,05	0,78	0,32	0,60
• <i>C. odorata</i>	-0,05 ± 0,04	0,03		
→ Pour la race Ndama	-0,05 ± 0,02		0,69	
→ Pour la race zébu Goudali	-0,07 ± 0,03		0,05	
→ Pour la race Ndapol	-0,04 ± 0,05		0,02	
→ Pour la race Zépol	-0,05 ± 0,06		0,15	
• Abris contre le soleil et la pluie	-0,02 ± 0,03	0,96	0,03	0,33

Rouge : p-valeur Interaction < 0,1 ; Vert : p-valeur Pâture < 0,1

Le tableau 14 contient les moyennes et les écarts-types associés à chaque pâture pour les variables dont l'hypothèse nulle « égalité des moyennes selon le facteur pâture » a été rejetée avec une p-valeur de 0,1 (en vert dans les tableaux précédents).

Tableau 14 : résultats complémentaires du traitement géostatistique de la position des bêtes et de ses liens avec la végétation : moyennes et écarts-types par pâture

Variables pour lesquelles l'hypothèse nulle « égalité des moyennes selon le facteur pâture » a été rejetée	Pâture 1 : Moyenne ± Ecart-type	Pâture 2 : Moyenne ± Ecart-type
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation pour les données de jour :		
• Graminées cespiteuses de moins de 1 m de haut	-0,14 ± 0,03	0,09 ± 0,08
• Graminées cespiteuses de plus de 1 m de haut	-0,13 ± 0,04	0,07 ± 0,08
• <i>Mitracarpus hirtus</i>	0,03 ± 0,06	-0,10 ± 0,07
• Végétation herbacée hors 6 critères précédents	-0,05 ± 0,08	0,05 ± 0,10

Corrélation entre la présence des bêtes et celle de chaque critère de la végétation pour les données de nuit :		
• Graminées rampantes ou gazonnantes		
→ Pour la race Ndama	0,08 ± 0,01	0,02 ± 0,01
→ Pour la race Ndapol	0,09 ± 0,01	-0,04 ± 0,03
→ Pour la race Zépol	0,13 ± 0,02	-0,03 ± 0,02
• Graminées cespiteuses de moins de 1 m de haut	0,02 ± 0,02	-0,04 ± 0,04
• Graminées cespiteuses de plus de 1 m de haut	0,05 ± 0,05	-0,02 ± 0,04
• <i>Calopogonium mucunoides</i>	0,02 ± 0,01	-0,02 ± 0,06
• <i>Sida cf. acuta</i>	-0,07 ± 0,02	-0,03 ± 0,01
• <i>Mitracarpus hirtus</i>	-0,15 ± 0,03	0,04 ± 0,03
• Végétation herbacée hors 6 critères précédents	0,03 ± 0,03	-0,03 ± 0,04
• <i>C. odorata</i>		
→ Pour la race Ndapol	-0,09 ± 0,01	0,01 ± 0,01
• Abris contre le soleil et la pluie	-0,04 ± 0,02	-0,01 ± 0,02
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation pour les données "grazing" :		
• Graminées rampantes ou gazonnantes	0,22 ± 0,06	0,10 ± 0,15
• Graminées cespiteuses de moins de 1 m de haut	-0,09 ± 0,07	0,08 ± 0,11
• Graminées cespiteuses de plus de 1 m de haut	-0,07 ± 0,08	0,06 ± 0,11
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation pour les données "non grazing" :		
• Graminées rampantes ou gazonnantes		
→ Pour la race Ndapol	0,10 ± 0,01	-0,02 ± 0,02
→ Pour la race Zépol	0,15 ± 0,02	-0,01 ± 0,03
• Graminées cespiteuses de moins de 1 m de haut	0,01 ± 0,03	-0,03 ± 0,04
• Graminées cespiteuses de plus de 1 m de haut	0,05 ± 0,05	-0,01 ± 0,03
• <i>Calopogonium mucunoides</i>	0,01 ± 0,01	-0,03 ± 0,05
• <i>Sida cf. acuta</i>	-0,07 ± 0,03	-0,03 ± 0,01
• <i>Mitracarpus hirtus</i>	-0,16 ± 0,03	0,03 ± 0,03
• Végétation herbacée hors 6 critères précédents	0,03 ± 0,04	-0,02 ± 0,03
• <i>C. odorata</i>		
→ Pour la race zébu Goudali	-0,10 ± 0,01	-0,04 ± 0,01
→ Pour la race Ndapol	-0,09 ± 0,01	0,00 ± 0,00
• Abris contre le soleil et la pluie	-0,04 ± 0,03	0,00 ± 0,03

Les moyennes et les écarts-types associés aux autres variables et aux autres groupes d'animaux sont disponibles en annexe 30.

9. Discussion

a. Biais, limites et pistes d'améliorations

Avant de procéder à l'interprétation des résultats, il convient d'évaluer leur fiabilité et de mettre en lumière les sources de biais potentielles ainsi que les limites de ce travail de fin d'études. Cette démarche a pour but d'aborder les résultats d'un point de vue objectif et d'arriver à des conclusions mesurées et réalistes. L'expérience acquise au cours de ce travail permet également de proposer des pistes d'améliorations du protocole qui a été mis en place.

Premièrement, l'analyse du dispositif expérimental amène à poser plusieurs limites à l'interprétation des résultats. En effet, l'étude a été réalisée sur une surface, un nombre de bêtes et une durée particulièrement limités. De plus, les pâtures mises à disposition présentaient un microclimat relativement sec et des caractéristiques de dégradation très supérieures à celles rencontrées couramment dans le ranch Nyanga. Les conditions expérimentales étaient donc fort différentes des conditions de terrain réellement rencontrées dans le système d'élevage du ranch. Il semble de ce fait difficile d'extrapoler les observations qui ont été réalisées dans le cadre de ce travail à la conduite d'un troupeau de 200 têtes sur des pâtures de plusieurs centaines d'hectares. De plus, il n'a pas été possible de prendre en compte la variabilité importante de la qualité nutritionnelle de la végétation et des conditions climatiques au cours du cycle annuel des températures, des pluies, des feux et des rotations de pâtures. Une analyse du comportement des animaux dans leurs conditions habituelles de pâturage serait cependant plus difficile à mettre en œuvre à cause des échelles de temps et d'espace qu'elles impliquent. Elle demanderait la mise en place d'un protocole beaucoup plus conséquent que celui qui était envisageable dans le cadre de ce travail.

En outre, l'application du dispositif expérimental n'a pas permis de contrôler un certain nombre d'éléments qui peuvent impacter la fiabilité des données récoltées. Le problème le plus important a été la perméabilité des clôtures des pâtures mises à disposition : la charge en bétail n'a pas pu être maîtrisée du fait des intrusions régulières de bêtes qui ne faisaient pas partie du protocole et les bêtes étudiées se sont échappées à de multiples reprises, parfois pendant plusieurs jours. Les interférences avec les activités du ranch ont également été nombreuses, ce qui a entraîné le parcage provisoire de troupeaux entiers dans les pâtures de l'expérimentation ainsi que de nombreux passages au centre de soin et de longues périodes d'attente dans des paddocks pour les bêtes du protocole. De plus, il n'a pas été possible de disposer pour l'expérimentation d'un troupeau homogène au niveau de la castration ni de pâtures

suffisamment nutritives pour ne pas avoir à libérer régulièrement les bêtes afin de préserver leur état de santé. De manière générale, toutes ces difficultés sont inerrantes à la mise en place d'un protocole expérimental qui ne s'inscrit pas au sein même des activités du ranch Nyanga. En dehors de ce cadre, il semble difficile d'envisager un contrôle efficace des conditions expérimentales sans moyens plus conséquents que ceux qui étaient à disposition.

Le deuxième point à prendre compte dans cette analyse est la récolte des données. Pour les données relatives aux animaux, la fiabilité des observations directes du comportement réalisées aux jumelles doit être nuancée. En effet, le développement important de la végétation arbustive et le fait que certaines bêtes étaient particulièrement farouches les a rendues difficiles. De plus, ces observations impliquaient une certaine part de subjectivité et donc un manque de reproductibilité.

L'utilisation des capteurs des iPhones était donc particulièrement indiquée dans ce contexte et a permis de pallier ces défauts. Cependant, les difficultés techniques rencontrées ont réduit la quantité de données à 30 pourcents de ce qui était prévu dans le protocole. La base de données qui résulte de ces mesures est donc moins fiable que ce qui était envisagé. Si le recours à cette technique reste particulièrement pertinent, des améliorations matérielles et logicielles sont donc indispensables à son utilisation dans des conditions similaires.

Pour les données relatives à la végétation, la caractérisation visuelle de celle-ci a permis une spatialisation rigoureuse des observations qui s'est révélée précieuse. Si son principal avantage était sa rapidité de mise en œuvre par rapport à la quantité de données récoltées, les critères qui ont été utilisés étaient relativement peu précis et leur évaluation assez subjective. Une possibilité d'amélioration de cette méthode, qui ne rentrait pas dans le cadre de ce travail de fin d'études, serait l'identification et le comptage ou l'évaluation sur une échelle d'abondance-dominance de toutes les espèces végétales sur une surface donnée pour chacun des points d'observation (Daget et al., 2010). A contrario, pour la caractérisation de la végétation à une plus grande échelle, l'utilisation de drones et le recours à la télédétection pourraient s'avérer très pertinents (Cunliffe et al., 2016).

Les prélèvements végétaux qui n'ont pas été exploités auraient pu apporter des informations intéressantes sous certaines conditions : ils auraient dû être beaucoup plus nombreux, être séchés à l'étuve et non à l'air libre, et être analysés en laboratoire. Dans de telles circonstances, les données récoltées au moyen de cette méthode pourraient être mises en parallèle avec la présence des bêtes et la caractérisation de la végétation.

Finally, the discussion of the methods used for the treatment of the data proves to be equally relevant. Several remarks can be made about the variance analyses that were carried out. The most problematic element is the very small number of observations for each combination of the factor « pasture » and the factor « race », so that the standard deviations of most of the variables studied are particularly important. This finding implies a non-negligible probability that the statistical tests did not allow for the detection of all the differences in means present. In addition, this small number of observations has led to the impossibility of verifying whether the populations were normal and of the same variance. This assumption has been taken as true, as it is the main condition for the application of a variance analysis. The increase in the number of animals studied thus presents itself as an indispensable element for any improvement of the protocol.

Parallèlement, un nombre très important de tests statistiques (plus de 300) ont été réalisés en se basant sur une p-valeur de 0,1. Or cette p-valeur correspond à la probabilité d'observer les données analysées dans le cas où l'hypothèse nulle est vraie et donc, par extension, à la probabilité de commettre une erreur en rejetant l'hypothèse nulle. Les différences de moyennes qui ont été déduites des analyses de la variance ne correspondent donc probablement pas toutes à la réalité. Cela implique que les résultats doivent être analysés dans leur globalité dans l'optique de déceler des tendances et non indépendamment les uns des autres. On peut toutefois observer que cette contrainte n'est pas problématique dans le cadre de ce travail de fin d'études au vu des objectifs poursuivis.

Two additional elements also raise questions. Firstly, the reliability of the process that allowed for the correlations between the position of the animals and the characteristics of the vegetation is particularly difficult to estimate. In fact, the implications of the large number of steps in the treatment of the data, the choices of parameters envisaged and the non-respect of certain conditions for the application of the models (notably their use for the study of ordinal variables) on the reliability of the data are not fully mastered. A theoretical analysis of this process would be relevant and would probably allow for improvements.

Secondly, the criterion used to separate the « grazing » data from the « non grazing » data could be improved. In fact, a common acceleration threshold was used for all the files, whereas the position of the iPhones on the animal

n'était pas parfaitement identique pour chacune des mesures ni même pendant la durée de celles-ci. Les difficultés à maintenir les iPhones dans la position adéquate sont en partie liées à la forme de la nuque des bovins tropicaux, plus anguleuse que celle des bovins européens. Plusieurs pistes existent afin d'améliorer la détection des périodes pendant lesquelles les bêtes se nourrissent : un seuil différent pourrait être utilisé pour chaque fichier, le système de fixation des iPhones pourrait être amélioré et des techniques de détections plus fines pourraient être utilisées. L'adaptation de la méthode mise au point par Andriamandroso et al. (2017) aux conditions tropicales permettrait, par exemple, d'accroître la précision de la détection en utilisant une plus grande diversité de données en provenance des capteurs.

En conclusion, les résultats doivent être analysés avec prudence à cause de l'effectif réduit des analyses de la variance et des inconnues concernant la fiabilité de certaines données. De plus, leur interprétation doit se baser sur des tendances générales et non sur des variables précises du fait du nombre de tests statistiques réalisés. Enfin, les conditions expérimentales rendent difficile la généralisation des résultats à l'ensemble du ranch.

b. Interprétation des résultats

Les résultats du traitement statistique des sources de biais (pluviométrie et gain quotidien moyen) n'ont montré aucune différence de moyenne entre les différents groupes d'animaux. Il n'est donc pas nécessaire de prendre en compte ces éléments pour analyser le reste des résultats.

Le traitement statistique des observations directes du comportement a, quant à lui, permis de mettre en évidence de nombreuses différences entre les races. De manière générale, les Ndamas et, dans une moindre mesure, les Ndapols semblent se montrer relativement indépendants et interagissent assez peu avec leurs congénères. Au contraire, les zébus Goudalis de l'Adamaoua et les Zépols présentent un comportement grégaire beaucoup plus marqué : les bêtes sont moins souvent seules, forment des groupes plus importants et restent plus proches les unes des autres.

L'apport de la race Senepol dans les croisements, si elle influe beaucoup sur les caractéristiques physiques des animaux, semble donc ne jouer qu'un rôle mineur dans leur comportement social. Il faut toutefois nuancer cette affirmation. En effet, les liens que les animaux ont tissés avec leur mère lorsqu'ils étaient veaux doivent être pris en compte et le fait que les jeunes Ndapols aient été sociabilisés auprès de Ndamas et les jeunes Zépols auprès de zébus a plus que probablement une influence sur leur comportement à l'âge adulte (Galef,

1988 ; Howery et al., 1998). Une différence notable est cependant présente entre les zébus et les Zépols : les premiers sont en effet plus calmes et moins agressifs que les seconds.

Les résultats de ce traitement statistique n'ont en revanche pas mis en évidence de différences entre les structures sociales observées dans chacune des deux pâtures. La dégradation de la végétation ne semble donc pas influencer sur les relations entre les animaux. De manière générale, la consommation de ligneux ne semble pas non plus présenter de différences, que ce soit au niveau de la race des bêtes ou de la pâture dans laquelle elles se trouvent. Des observations plus précises et plus nombreuses seraient toutefois nécessaires pour étayer ces affirmations.

Si l'interprétation des observations directes du comportement ne permet pas de suggérer une gestion différenciée des troupeaux en fonction de la végétation, il semble pertinent de préciser que la formation de troupeaux composés d'un mélange de races peut présenter des difficultés. En effet, si les différentes races cohabitent sans problème, elles ne forment pas un groupe homogène et cohérent. Cet élément peut amener à des complications dans la gestion et la conduite des troupeaux dans le système d'élevage pratiqué au ranch Nyanga. En outre, l'hypothèse que, dans le cadre de troupeaux mixtes, certaines races pourraient être désavantagées dans leur stratégie d'utilisation de l'espace et que l'exploitation de la végétation pourrait être moins uniforme mériterait d'être approfondie. Il semble donc judicieux de maintenir les animaux de races différentes au sein de troupeaux installés dans des pâtures distinctes.

Le traitement statistique de la vitesse des bêtes et de leur accès à l'eau et au sel n'a pas mis en évidence de différences pour les variables observées. Toutefois, au vu de la taille des pâtures dans lesquelles s'est déroulée l'expérimentation, la généralisation de cette conclusion pour les conditions d'élevage pratiquées au ranch Nyanga est délicate. C'est pourquoi l'étude des déplacements des bêtes ainsi que celle de leurs passages aux points d'abreuvement et de complémentation en sel conserve un intérêt certain à cette échelle. En effet, ces éléments peuvent avoir des conséquences importantes sur les performances des animaux et ne doivent pas être négligés.

L'interprétation des résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation a permis de mettre en évidence des différences entre les troupeaux des deux pâtures. Les données de jour et les données « grazing » relatives aux trois catégories de graminées, qui composent par ailleurs la majeure partie du régime alimentaire du bétail,

sont particulièrement intéressantes. Elles montrent que les bêtes ne consomment pas les différentes strates de la végétation de graminées de manière proportionnelle à leur présence dans le couvert végétal. En effet, les bêtes adaptent leur comportement alimentaire en fonction de la pâture dans laquelle elles se trouvent et présentent une appétence différente pour chaque strate en fonction de leur environnement. Si les autres éléments de la végétation ne semblent pas influencer sur leur comportement alimentaire, il ne faut pas exclure le fait qu'une modification du protocole de récolte des données ou une optimisation du traitement de celles-ci pourrait affiner les résultats et amener à d'autres conclusions. Il en va de même pour l'absence d'influence de la race des animaux sur les adaptations du comportement alimentaire qui peut être déduite de ces résultats.

Les données de nuit et les données « non-grazing » ont montré que les stratégies d'occupation de l'espace sont liées à un nombre important de facteurs et ne dépendent pas des mêmes critères dans les deux pâtures. Les différences dans la composition de la végétation modifient donc le comportement non alimentaire du troupeau (choix des zones de repos, etc.). De plus, certaines de ces modifications ne semblent pas se faire de façon identique en fonction de la race de l'animal, notamment celles vis-à-vis de la végétation herbacée rase (graminées rampantes ou gazonnantes) ou des zones envahies par *Chromolaena odorata* (L.) R.M. King & Robinson. Cette dernière affirmation devrait cependant être corroborée par de nouvelles données avant de pouvoir en tirer quelque conclusion. De plus, rien ne permet de faire le lien entre ces modifications de l'occupation de l'espace et les performances zootechniques des animaux.

Les résultats du traitement géostatistique de la position des bêtes et de ses liens avec la végétation ne semblent donc pas justifier une gestion différenciée des pâturages en fonction de la race. Cependant, les modifications du comportement alimentaire des animaux en fonction de l'état de dégradation de la végétation amènent différentes pistes de réflexions sur les protocoles à mettre en place pour quantifier ces modifications et pour mettre en évidence leurs relations avec les performances zootechniques. Ces protocoles demanderaient dans tous les cas des échelles de temps et de surface importantes ainsi que la création de bases de données conséquentes pour mener à des conclusions rigoureuses. De plus, un suivi plus rigoureux des performances du bétail, un état des lieux de la qualité de l'ensemble des pâtures du ranch et une évaluation de la variabilité des dégradations de la végétation seraient nécessaires pour mettre ces résultats en perspective et évaluer leurs implications dans le cadre de la gestion des troupeaux et des pâtures du ranch Nyanga.

10. Conclusions et perspectives

Ce travail de fin d'études permet d'arriver à deux conclusions principales dans le contexte actuel du ranch Nyanga et des techniques d'élevage extensif qui y sont pratiquées. Ces conclusions permettent d'apporter de nouveaux éléments de réflexion pour la gestion des troupeaux et des pâtures du ranch, ainsi que pour l'adaptation de celle-ci à l'évolution de la végétation et à l'introduction de nouvelles races.

Premièrement, le comportement social des bovins présents sur le ranch diffère en fonction de leur race. Cela implique que les troupeaux de race homogène devraient être privilégiés. En effet, leur gestion, tout comme leur conduite, est moins complexe à mettre en œuvre que celle de troupeaux composés de groupes d'animaux de races différentes qui ne présentent pas de comportement cohérent entre eux.

Deuxièmement, les bovins des différentes races adaptent leur comportement alimentaire à la végétation de la pâture dans laquelle ils se trouvent. Leur appétence pour les différentes parties du couvert végétal se trouve donc modifiée par les conditions de leur milieu. Cette conclusion n'implique pas de gestion différenciée des différentes races en fonction de l'état de dégradation des pâtures. Elle amène cependant à suggérer des recherches complémentaires afin de quantifier ces modifications du comportement alimentaire et d'étudier leur incidence sur les performances des bovins.

Deux éléments clés sont à prendre en compte si de telles recherches étaient envisagées : le nombre de bêtes sur lesquelles porteraient les observations devrait être beaucoup plus conséquent que dans le cadre de ce travail de fin d'études et les observations devraient être réalisées dans les conditions réelles de pâturage pratiquées sur le ranch. Diverses observations pourraient être analysées avec pertinence si un tel protocole était mis en place. Le couplage de données en provenance des capteurs des iPhones avec celles issue de processus de télédétection sur une durée d'un an semble, par exemple, être une piste particulièrement intéressante.

Dans tous les cas, il semble judicieux de mettre en place un suivi précis, complet et informatisé des performances zootechniques, de procéder à un état des lieux de l'état de dégradation des pâtures du ranch Nyanga et d'évaluer leur évolution dans l'espace et dans le temps. Sans ces informations, des études plus poussées au sujet du comportement alimentaire des bovins du ranch ne pourraient en effet que difficilement être mises en perspective.

Références bibliographiques

- Akouango F., Ngokaka C., Ewomango P. & Kimbembe E., 2010. Caractérisation morphométrique et reproductive des taureaux et vaches N'Dama du Congo. *Animal Genetic Resources*, 46, 41–47.
- Alvarado S.T., Silva T.S.F. & Archibald S., 2018. Management impacts on fire occurrence: A comparison of fire regimes of African and South American tropical savannas in different protected areas. *Journal of Environmental Management*, 218, 79–87.
- Alvarez P. et al., 2000. Ovarian and endocrine characteristics during an estrous cycle in Angus, Brahman, and Senepol cows in a subtropical environment. *Journal of Animal Science*, 78, 1291–1302.
- Andriamandroso A.L.H. et al., 2017. Development of an open-source algorithm based on inertial measurement units (IMU) of a smartphone to detect cattle grass intake and ruminating behaviors. *Computers and Electronics in Agriculture*, 139, 126–137.
- ArcGIS, 2016a. Fonctionnement de l'outil Krigeage. <http://desktop.arcgis.com/fr/arcmap/10.3/tools/3d-analyst-toolbox/how-kriging-works.htm>, (13/08/19).
- ArcGIS, 2016b. Que sont les données raster ? <http://desktop.arcgis.com/fr/arcmap/10.3/manage-data/raster-and-images/what-is-raster-data.htm>, (13/08/19).
- Assani Seidou A., 2013. *Typologie et productivité des élevages de Zébu Goudali situés dans les communes de Malanville et de Karimama à l'extrême Nord du Bénin*. Thèse de doctorat : Faculté d'agronomie, Université de Parakou.
- Barnes D.L., 1979. Cattle ranching in the semi-arid savannas of East and Southern Africa. In : Walker B. H., ed. *Management of semi-arid ecosystems*. Amsterdam : Elsevier scientific publishing company, 9–54.
- Besse F. & Toutain B., 2012. L'élevage et l'environnement. In : CIRAD, GRET & Ministère des Affaires étrangères, eds. *Mémento de l'agronome*. Versailles : Editions Quae, 1313–1324.
- Bougler J., Duplan J.M. & Zongo D., 2005. Caractéristiques et répartition mondiale des espèces animales exploitées par l'homme. Espèces domestiques à grands effectifs ou cosmopolites. In : Thewis A. et al., eds. *Manuel de zootechnie comparée Nord-Sud*. Paris : INRA Editions, 45–57.
- Boutonnet J.P., 2005. L'économie des productions animales. In : Thewis A. et al., eds. *Manuel de zootechnie comparée Nord-Sud*. Paris : INRA Editions, 519–544.
- Boutrais J., 1983. *L'élevage soudanien : des parcours de savanes aux ranchs, Cameroun-Nigeria*. Paris : ORSTOM.

- Boutrais J., 1987. *Ranchs ou coopératives d'élevage ? L'alternative pastorale au Cameroun*. Paris : ORSTOM.
- Boutrais J., 1990. Derrière les clôtures... essai d'histoire comparée de ranchs africains. *Cahiers Des Sciences Humaines*, 26(1-2), 73-95.
- Chase C.C. et al., 1998. Preweaning Growth Traits for Senepol, Hereford, and Reciprocal Crossbred Calves and Feedlot Performance and Carcass Characteristics of Steers. *Journal of Animal Science*, 76, 2967-2975.
- Chatelin Y., 1968. Notes de pédologie gabonaise : Géomorphologie et pédologie dans le Sud Gabon, des monts Birougou au littoral. *Cahiers ORSTOM Série Pédologie*, 6(1), 3-20.
- Cianzo D., 2002. Brief history of the Senepol cattle in Puerto Rico. In : *Senepol Symposium, St. Croix, USVI, 8-10 November 2002*. Kingshill : University of the Virgin Islands, 1-5.
- Coulomb J., 1976. La race N'Dama. Quelques caractéristiques zootechniques. *Revue d'élevage et de Médecine Vétérinaire Des Pays Tropicaux*, 29(4), 367-380.
- Cunliffe A.M., Brazier R.E. & Anderson K., 2016. Ultra-fine grain landscape-scale quantification of dryland vegetation structure with drone-acquired structure-from-motion photogrammetry. *Remote Sensing of Environment*, 183, 129-143.
- Daget P., Poissonet J. & Huguenin J., 2010. *Prairies & Pâturages. Méthodes d'étude de terrain et Interprétation*. Paris : Cnrs ; Montpellier : Cirad.
- Dehoux J.P., 1990a. Chimio prophylaxie antitrypanosomienne de bovins N'Dama importés de Sénégal et du Zaïre au Gabon. *Revue d'élevage et de Médecine Vétérinaire Des Pays Tropicaux*, 43(3), 337-341.
- Dehoux J.P., 1990b. Note à propos de l'importation de bétail de race N'dama provenant du Zaïre et de Sénégal au Gabon, sur le ranch de la Ngounie, en 1986 et en 1987. *TROPICULTURA*, 8(2), 78-81.
- Descoings B., 1961. *Les Savanes de la vallée de la Nyanga (République Gabonaise) : phytosociologie et possibilités pastorales*. Brazzaville : Institut d'Etudes Centre-Africaines.
- Diambounambatsi J., 2015. *Privatisation et emploi au Gabon : Analyse de la politique de privatisations sur l'emploi et sa structure de trois entreprises publiques rendues privées au Gabon*. Thèse de doctorat : Ecole Doctorale Sciences Humaines et Sociale, Université Rennes 2.
- Dibanganga G.L., 2015. *Effet de l'infection trypanosomienne sur les bovins trypanotolérants élevés au ranch SIAT-Nyanga, au sud du Gabon*. Mémoire de fin de cycle : Institut national supérieur d'agronomie et de biotechnologies, Université des sciences et techniques de Masuku.

- Dufumier M., 2012. Les modes de mise en valeur et leurs évolutions. In : CIRAD, GRET & Ministère des Affaires étrangères, eds. *Mémento de l'agronome*. Versailles : Editions Quae, 411–432.
- Dumont B. & Boissy A., 1999. Relations sociales et comportement alimentaire au pâturage. *Productions Animales*, 12(1), 3–10.
- Duteurtre G., Koussou M.O., Essang T. & Kadekoy-Tigague D., 2003. Le commerce de bétail dans les savanes d'Afrique centrale : réalités et perspectives. In : Jamin J.Y., Seiny-Boukar L. & Floret C., eds. *Savanes africaines : des espaces en mutation, des acteurs face à de nouveaux défis, 27-31 May 2002, Garoua, Cameroon*. N'Djaména : Prasac ; Montpellier : Cirad.
- FAO, 2018. FAOSTAT : élevage. <http://www.fao.org/faostat/fr/#data/QA>, (18/07/19).
- FAO, 2019a. FAOSTAT : élevage primaire. <http://www.fao.org/faostat/fr/#data/QL>, (18/07/19).
- FAO, 2019b. FAOSTAT : séries temporelles annuelles. <http://www.fao.org/faostat/fr/#data/OA>, (13/08/19).
- Feumetio B., 2008. *Le Gabon, un pays si riche ... mais très pauvre. Décryptage d'un authentique paradoxe socio-économique*. Paris : Publibook.
- Flori L. et al., 2012. A Quasi-Exclusive European Ancestry in the Senepol Tropical Cattle Breed Highlights the Importance of the slick Locus in Tropical Adaptation. *PLoS ONE*, 7(5): e36133.
- Fourcassié V., 2015. L'utilisation de l'espace. In : Darmaillacq A.S. & Lévy F., eds. *Ethologie animale. Une approche biologique du comportement*. Louvain-la-Neuve : De Boeck Supérieur, 45–64.
- Galef B.G., 1988. Imitations in animals: History, definitions and interpretations of data from the psychological laboratory. In : Zentall T.R. & Galef B.G. eds. *Social Learning: Psychological and Biological Perspectives*. Hillsdale, New Jersey : Lawrence Erlbaum Associates, 3–27.
- Gallais J., 1979. La situation de l'élevage bovin et le problème des éleveurs en Afrique occidentale et centrale. *Cahiers d'Outre-Mer*, 126, 113–138.
- Garnier L.K.M. & Dajoz I., 2001. The influence of fire on the demography of a dominant grass species of West African savannas, *Hyparrhenia diplandra*. *Journal of Ecology*, 89, 200–208.
- Gonne B., Sougnabe P. & Ngana F., 2010. Le champ et le bœuf en savane d'Afrique centrale complémentarité, antagonisme ou intégration ? In : Seiny-Boukar L. & Boumard P., eds. *Savanes africaines en développement : innover pour durer, 20-23 April 2009, Garoua, Cameroun*. N'Djaména : Prasac ; Montpellier : Cirad.

- Goodall J.M. & Erasmus D.J., 1996. Review of the status and integrated control of the invasive alien weed, *Chromolaena odorata*, in South Africa. *Agriculture, Ecosystems and Environment*, 56, 151–164.
- Gouvernement de la République gabonaise, 2005. *Appui à la mise en œuvre du NEPAD-PDDA. TCP/GAB/2907(I) (NEPAD Ref. O5/09 F). Volume I de V. Programme national d'investissement à moyen terme (PNIMT)*.
- Guimarães A.L. et al., 2017. Phenotypic association between feed efficiency and feeding behavior, growth and carcass traits in Senepol cattle. *Revista Brasileira de Zootecnia*, 46(1), 47–55.
- Hellemans P., 1984. *Influence du système d'exploitation sur la flore et les caractéristiques de la savane à Hyparrhenia diplandra et Panicum phragmitoides (Ranch de la Nyanga, Gabon)*. Travail de fin d'études : Faculté des sciences agronomiques de l'Etat (Gembloux).
- Hernandez A.I., Cianzio D. & Olson T.A., 2002. Physiological performance and grazing behavior of Senepol, Brahman and Holstein heifers in Puerto Rico. In : *Senepol Symposium, St. Croix, USVI, 8-10 November 2002*. Kingshill : University of the Virgin Islands.
- Howery L.D., Provenza F.D., Banner R.E. & Scott C.B., 1998. Social and environmental factors influence cattle distribution on rangeland. *Applied Animal Behaviour Science*, 55, 231–244.
- Joshi N.R., McLaughlin E.A. & Phillips R.W., 1957. *Les bovins d'Afrique. Types et races*. Rome : FAO.
- Labonne M., Magrong P. & Oustalet Y., 2003. Le secteur de l'élevage au Cameroun et dans les provinces du grand Nord : situation actuelle, contraintes, enjeux et défis. In : Jamin J.Y., Seiny-Boukar L. & Floret C., eds. *Savanes africaines : des espaces en mutation, des acteurs face à de nouveaux défis, 27-31 May 2002, Garoua, Cameroon*. N'Djaména : Prasac ; Montpellier : Cirad.
- Lecomte P. & Compère R., 2005. Systèmes d'élevage sur parcours. Ranches bovins en Afrique tropicale humide. In : Thewis A. et al., eds. *Manuel de zootechnie comparée Nord-Sud*. Paris : INRA Editions, 133–140.
- Lefèvre P.C., 2003. Péripleurmonie contagieuse bovine. In : Lefèvre P.C., Blancou J. & Chermette R., eds. *Principales maladies infectieuses et parasitaires du bétail. Europe et régions chaudes*. Paris : Tec & Doc Lavoisier, 775–794.
- Lhoste P., Dollé V., Rousseau J. & Soltner D., 1993. *Manuel de zootechnie des régions chaudes. Les systèmes d'élevage*. Paris : Ministère de la Coopération.
- Lhoste P., 1980. L'amélioration génétique des zébus de l'Adamaoua (Cameroun) pour la production de la viande. In : *Premier colloque international : Recherches sur l'élevage bovin en zone tropicale humide, Bouake, Côte d'Ivoire, 18-22 April 1977*. Maisons-Alfort : Institut d'Élevage et de Médecine Vétérinaire des Pays Tropicaux, 761–769.

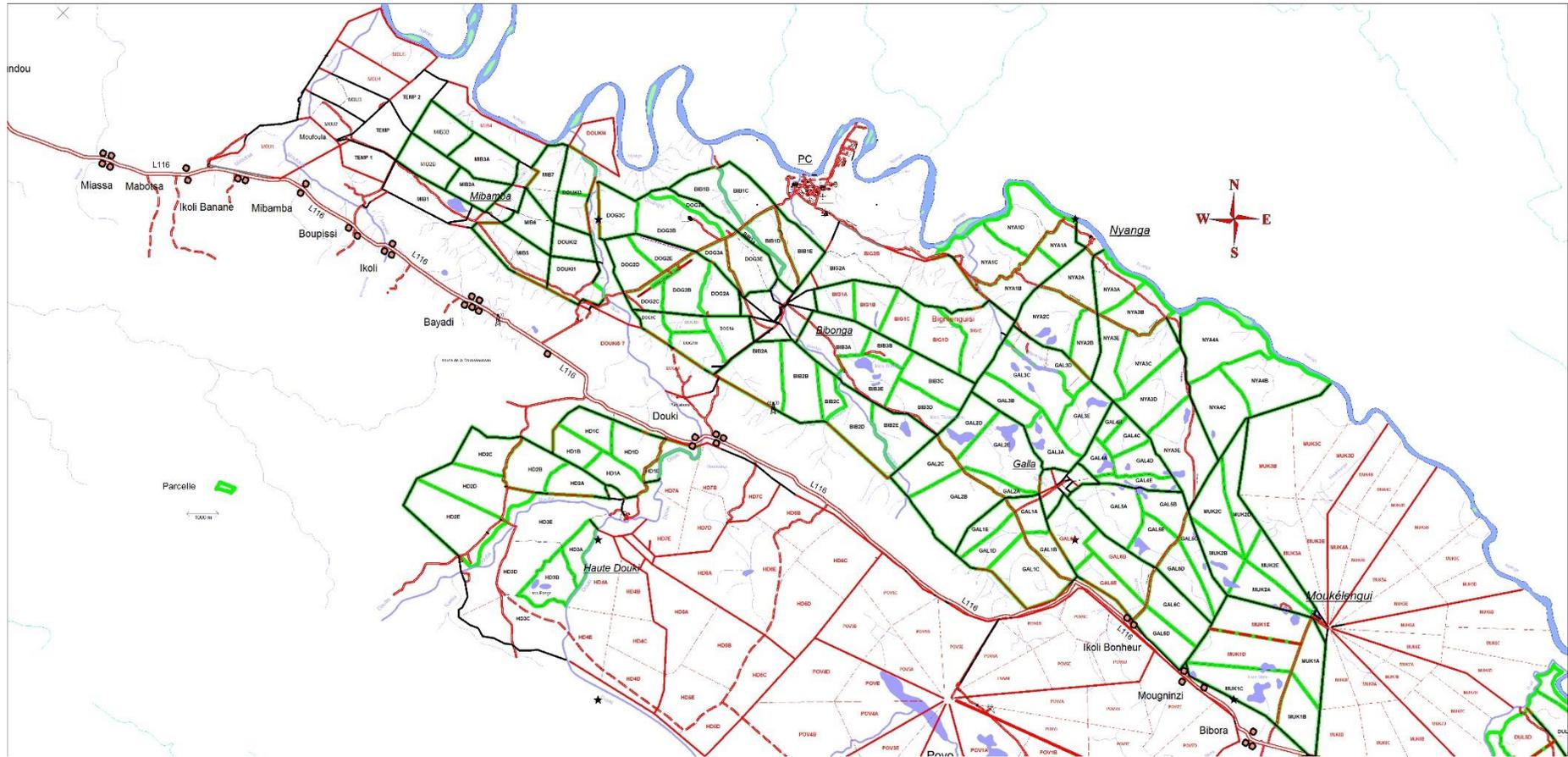
- Magnagna Nguema V., 2005. *L'agriculture du Gabon. Entre décolonisation et ajustements structurels (1960-2000)*. Paris : Editions Karthala.
- Manteca i Vilanova X. & Smith A.J., 2014. *Comportement, conduite et bien-être animal*. Gembloux : Presses agronomiques de Gembloux ; Versailles : Editions Quae ; Wageningen : CTA.
- McDonnell S.M., 2002. Behaviour of Cattle. In : Jenson P., ed. *The Ethology of Domestic Animal*. New York : CABI Publishing.
- Messine O., Tanya V.N., Mbah D.A. & Tawah C.L., 1995. Ressources génétiques animales du Cameroun. Passé, présent et avenir : Le cas des ruminants. *Animal Genetic Resources Information*, 16, 47–63.
- Ministère de l'agriculture de l'élevage et du développement rural (Gabon), 2003. *Rapport national sur l'état des ressources génétiques animales du Gabon*.
- Ministère de l'économie du commerce de l'industrie et du tourisme (Gabon), 2011. *Bulletin des statistiques du commerce extérieur. Années 2008, 2009 et 2010*.
- Mottet A. et al., 2018. Review : Domestic herbivores and food security : current contribution , trends and challenges for a sustainable development. *Animal*, 12(s2), s188–s198.
- Moula N. et al., 2014. Production of animal protein in the Congo Basin, a challenge for the future of people and wildlife. In : Rammeloo J., Van de Vijver B. & Goyens P., eds. *International Conference "Nutrition and Food Production in the Congo Basin", 30 October - 1 November 2013, Brussels, Belgium*. Brussels : Royal Academy for Overseas Sciences.
- Musco N. et al., 2016. Nutritional characteristics of forage grown in south of Benin. *Asian-Australasi. J. Anim. Sci.*, 29(1), 51–61.
- Ng'ethe J.C., 1993. Group ranch concept and practice in Kenya with special emphasis on Kajiado District. In : Kategile J.A. & Mubi S., eds. *Future of livestock industries in East and Southern Africa, Kadoma Ranch Hotel, Zimbabwe 20-23 July 1992*. Addis Ababa : International Livestock Centre for Africa, 187–200.
- Okouyi M., 2018. *Application des progestagènes à la gestion hormonale de la reproduction chez la femelle de race n'dama au Gabon*. Thèse de doctorat : Faculté de médecine vétérinaire, Université de Liège.
- Otte M.J. & Chilonda P. (2002). *Cattle and small ruminant production systems in sub-Saharan Africa. A systematic review*. Rome : FAO.
- Peel M.C., Finlayson B.L. & McMahon T.A., 2007. Updated world map of the Köppen-Geiger climate classification. *Hydrology and Earth System Sciences*, 11, 1633–1644.
- Renard G., 2017. *Stage technique en entreprise : Ranch "Nyanga", Gabon, du 26 juin au 4 septembre 2017*. Rapport de stage : Gembloux Agro-Bio Tech, Université de Liège.

- Schwartz D., Lanfranchi R. & Mariotti A., 1990. Origine et évolution des savanes intramayombiennes (R.P. du Congo). I. Apports de la pédologie et de la biogéochimie isotopique (14C et 13C). In : Lanfranchi R. & Schwartz D., eds. *Paysages quaternaires de l'Afrique centrale atlantique*. Paris : ORSTOM, 314–325.
- Sèwadé C. et al., 2016. Diversité, priorité pastorale et de conservation des ligneux fourragers des terres de parcours en zone soudano- guinéenne du Bénin. *Biotechnology, Agronomy and Society and Environment*, 20(2), 113–129.
- Shaw A.P.M. & Hoste C.H., 1991. Les échanges internationaux de bovins trypanotolérants. I. Historique et synthèse. *Revue d'élevage et de Médecine Vétérinaire Des Pays Tropicaux*, 44(2), 221–228.
- Shlisky A., Alencar A., Manta M. & Curran L., 2009. Overview: Global fire conditions, threats, and opportunities for fire management in tropics. In : Cochrane M.A., ed. *Tropical fire ecology. Climate change, land use and ecosystem dynamics*. Berlin : Springer ; Chichester : Praxis Publishing, 65–83.
- Siat Gabon, 2010. Siat Info. Journal d'information de Siat Gabon. Volume 9. Décembre 2010. <http://www.siatgabon.com/telechargements/siat-infos-nc2b0-9/>, (31/07/19).
- Siat Gabon, 2017. Siat Infos. Journal d'information de Siat Gabon. Volume 19. Septembre 2017. <http://www.siatgabon.com/telechargements/siat-infos-n-19/>, (25/07/19).
- Siat Gabon, 2019. Le ranch Nyanga. <http://www.siatgabon.com/le-ranch-nyanga/>, (25/07/19).
- Skovlin J., 1971. Ranching in East Africa: A Case Study. *Journal of Range Management*, 24(4), 263–270.
- Sokouri D. et al., 2010. Performances de reproduction des races bovines locales de Côte d'Ivoire. *Journal of Applied Biosciences*, 36, 2353–2369.
- Tawah C.L. & Rege J.E.O., 1996. Gudali cattle of west and central Africa. In : Boyazoglu J. & Chupin D., eds. *Bulletin d'information sur les ressources génétiques animales*. Rome : FAO ; Nairobi : UNEP, 147–156.
- Thierry B., 2015. Les systèmes sociaux. In : Darmaillacq A.-S. & Lévy F., eds. *Ethologie animale. Une approche biologique du comportement*. Louvain-la-Neuve : De Boeck Supérieur, 83–97.
- Timbilla J.A. & Braimah H., 2002. Successful biological control of *Chromolaena odorata* in Ghana : The potential for a regional programme in Africa. In : Zachariades C., Muniappan R. & Strathie L.W., eds. *Proceedings of the fifth international workshop on biological control and management of Chromolaena odorata, 23-25 October 2000, Durban, South Africa*. Pretoria : ARC-PPRI, 66–70.
- Tsalefac M. et al., 2015. Climat de l'Afrique centrale : passé, présent et futur. In : De Wasseige C., Tadoum M., Eba'a Atyi R. & Doumenge C., eds. *Les forêts du Bassin du Congo - Forêts et changement climatique*. Neufchâteau : Weyrich, 37–52.

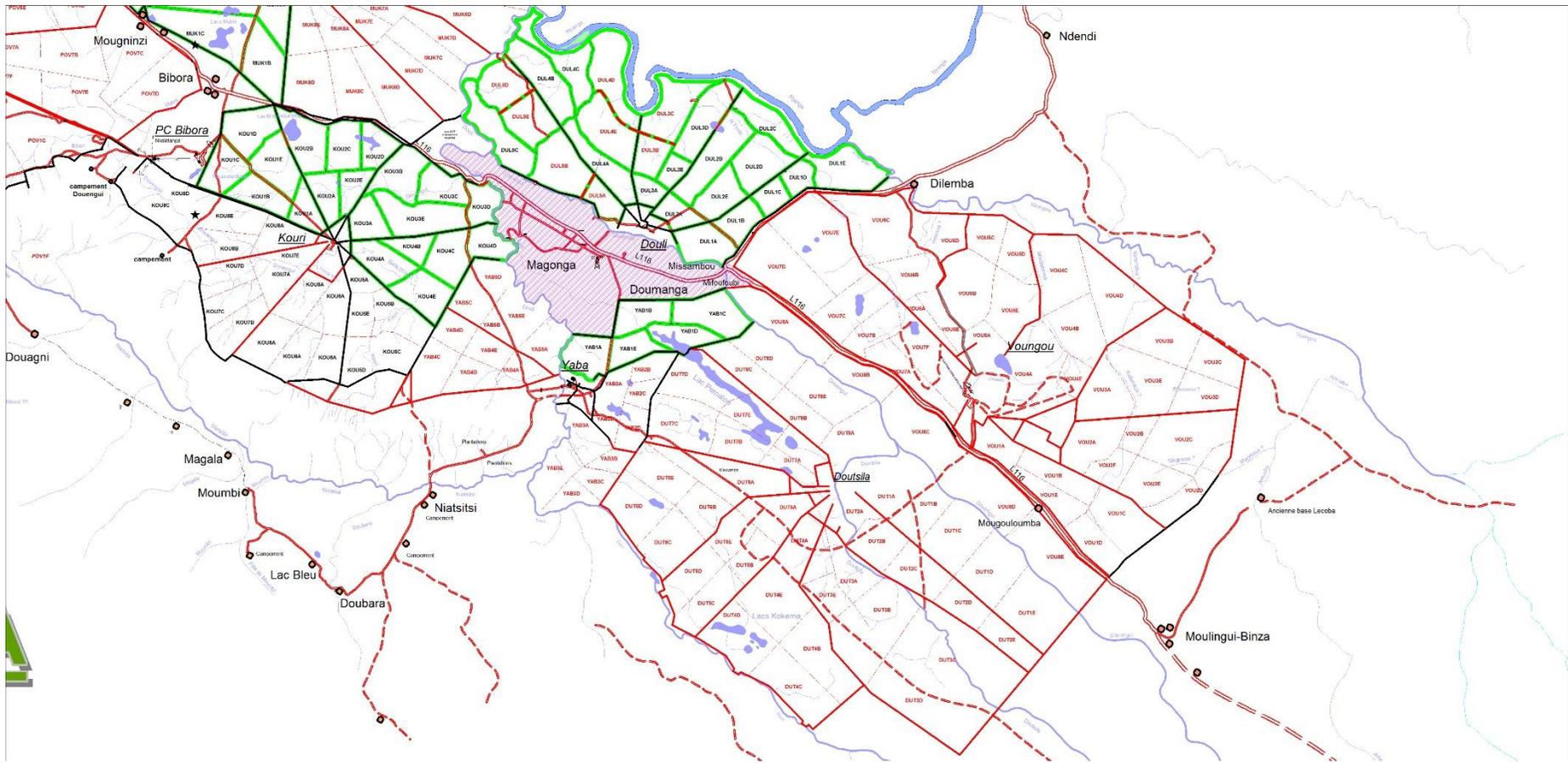
- United Nations Development Programme, 2018. *Indices et indicateurs de développement humain 2018. Mise à jour statistique*.
http://hdr.undp.org/sites/default/files/2018_human_development_statistical_update_fr.pdf, (24/07/19).
- Vall E., Njoya A. & Guérin H., 2003. L'animal et l'élevage dans les espaces agraires, l'alimentation des villes et les politiques agricoles : spécificités et complémentarité avec les autres activités rurales. Synthèse des communications du thème 3. In : Jamin J.Y., Seiny-Boukar L. & Floret C., eds. *Savanes africaines : des espaces en mutation, des acteurs face à de nouveaux défis, 27-31 May 2002, Garoua, Cameroon*. N'Djaména : Prasad ; Montpellier : Cirad.
- White F., 1983. *Vegetation of Africa. A descriptive memoir to accompany the Unesco/AETFAT/UNSO vegetation map of Africa*. Paris : UNESCO.
- Williams A.R., Hupp H.D., Thompson C.E. & Grimes L.W., 1988. Breed structure of Senepol Cattle. *Journal of Heredity*, 79, 9–13.
- Witkowski E.T.F. & Wilson M., 2001. Changes in density, biomass, seed production and soil seed banks of the non-native invasive plant, *Chromolaena odorata*, along a 15 year chronosequence. *Plant Ecology*, 152, 13–27.
- World Bank, 2019a. Agriculture, forestry, and fishing, value added (% of GDP).
<https://data.worldbank.org/indicator/NV.AGR.TOTL.ZS>, (24/07/19).
- World Bank, 2019b. Population density (people per sq. km of land area).
<https://data.worldbank.org/indicator/EN.POP.DNST>, (24/07/19).
- World Bank, 2019c. GDP Indicators 2017. <http://statisticstimes.com/economy/gdp-indicators-2017.php>, (24/07/19).

Annexes

Annexe 1 : carte du ranch Nyanga



Partie gauche : division 1. Siat Group



Partie droite : division 3. Siat Group

Annexe 2 : calendriers de gestion des troupeaux et des pâtures

1

Planning annuel des activités de reproduction au ranch Nyanga

Tâches	Fréquence	Mois												Responsable	Superviseur								
		J	F	M	A	M	J	J	A	S	O	N	D										
Mise au mâle • Monte principale : 90 j • Monte complémentaire (Vides SMP) : 45 j • 1 ^{er} services : 65 j	2 saisons	x	x	x						x	x	x							x	x	x	Chef de division	Responsable Elevage
Diagnostic de gestation • Monte principale • Monte compl. – 1 ^{er} Service	2 fois l'année																					Vétérinaire	Directeur

2

Planning annuel des opérations zootechniques au ranch Nyanga

Tâches	Fréquence	Mois												Responsable	Superviseur								
		J	F	M	A	M	J	J	A	S	O	N	D										
Comptage des troupeaux	Jours de soins	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Brigadier / Berger / Chef de section	Chef de division / Responsable Elevage
Reporting et Identification des naissances de veaux	Chaque jour	x	x	x	x	x							x	x	x	x	x	x	x	x	x	Brigadier	Chef de section / division
Marquage – écornage des veaux	Jours de soins			x	x	x	x	x	x					x	x	x	x	x	x	x	x	Chef de section / division	Responsable Elevage
Sevrage veaux	2 fois l'année																			x		Chef de section / division	Responsable Elevage
Castration des taurillons	Jours de soins	x	x	x	x	x	x													x	x	Chef de section / division	Responsable Elevage
Rapport d'activité (Stock - Elevage – Santé animale – Reproduction)	Chaque mois / semestre		x			x		x		x		x		x		x		x		x		Chef de section / Chef de division / Vétérinaire	Directeur
Inventaire du bétail	Chaque semestre																				x	Chef de section / division / Responsable Elevage	Directeur

3

Planning annuel des opérations sanitaires au ranch Nyanga

Tâches	Fréquence	Mois												Responsable	Superviseur								
		J	F	M	A	M	J	J	A	S	O	N	D										
Reporting des cas de maladies, de mortalités, d'avortements et tout événement inhabituel	Chaque jour	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Brigadier / Berger	Chef de section / division
Constat des mortalités et autopsie	Chaque jour	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Chef de section / division	Vétérinaire
Soins des troupeaux	Jours de soins	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Brigadier / Berger / Chef de section	Chef de division / Vétérinaire
Bain acaricide	Chaque 2 semaines	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Chef de section	Chef de division
Prophylaxie médicale • Anthelminthique • Trypano-prévention • Vaccinations (pasteurellose, PPCB)	Jours de soins																					Chef de division	Vétérinaire

Planning annuel de la gestion des pâtures au ranch Nyanga

Tâches	Fréquence	Mois												Responsable	Superviseur				
		J	F	M	A	M	J	J	A	S	O	N	D						
Réparation des clôtures des parcs	Chaque jour	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Brigadier	Chef de section / division
Gyro-broyage des pare-feu	Chaque jour	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Conducteur traceur / Chef de section	Chef de division
Entretien des pare-feu par la niveleuse	Jours fixés							x	x	x	x	x	x				Conducteur niveleuse / Chef de section	Chef de division	
Mise à feu des parcelles <ul style="list-style-type: none"> Contre-saison Précoces 1 et 2 Tardifs 	4 feux	x	x						x	x							Chef de section / division	Responsable Elevage	
Mise à charge des parcelles <ul style="list-style-type: none"> Contre-saison Précoces 1 et 2 Tardifs 	4 parcelles		x	x						x	x					x	x	Chef de section / division	Responsable Elevage

Plannings annuels au ranch Nyanga. Siat Group

Programmation générale

Années	Parcelle A	Parcelle B	Parcelle C	Parcelle D	Parcelle E
<u>1</u>	Contre saison	Précoce 1	Précoce 2	Tardif	Défens
<u>2</u>	Précoce 1	Précoce 2	Tardif	Défens	Contre saison
<u>3</u>	Précoce 2	Tardif	Défens	Contre saison	Précoce 1
<u>4</u>	Tardif	Défens	Contre saison	Précoce 1	Précoce 2
<u>5</u>	Défens	Contre saison	Précoce 1	Précoce 2	Tardif

Préparation des parcelles

	Feux	Engins utilisés
Contre saison	Précoce 1	Gyrobroyeur
Précoce 2	Tradif	Niveleuse / Bulldozer

Mises à feu

	Feux	Personnel
Contre saison	Précoce 1	Elément du parc suffisent
Précoce 2	Tradif	Présence du Chef de Section + suffisement du personnel

Mise à charge

Brûlis de feu	Mise à charge	Périodes
Contre saison	28 à 32 jours	Petite saison sèche: mi-décembre à mi-janvier
Précoce 1	40 jours si brûlage précoce	Début grande saison sèche: fin mai
Précoce 2	45 jours minimum	Début grande saison sèche: juin
Tardif	28 à 32 jours après (la première pluie)	Fin grande saison sèche: fin septembre

Gestion des feux. Siat Group

Annexe 3 : photos des races bovines



Ndama. Gauthier Renard, 2018



Zébu Goudali de l'Adamaoua.
Gauthier Renard, 2018



Ndapol. Gauthier Renard, 2018



Zépol. Gauthier Renard, 2018

Annexe 4 : photos des pâtures de l'expérimentation



Pâturation 1 : « petite pâture » du parc 1 de la section Kouri (division 2).
Gauthier Renard, 2018



Pâturation 2 : « petite pâture » du parc 8 de la section Kouri (division 2).
Gauthier Renard, 2018

Annexe 5 : chronogramme de l'expérimentation

<p>JOUR 1</p> <ul style="list-style-type: none"> - iPhones : recharge des batteries - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Action cyclique : prélèvements végétaux et déplacement des exclos en pâture 2 - Manipulations du bétail : ramassage du troupeau 2 	<p>JOUR 2</p> <ul style="list-style-type: none"> - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Manipulations du bétail : mise en place des iPhones sur le troupeau 2A - iPhones : début des mesures sur troupeau 2A - Prises vidéo : troupeau 2A - Progression du travail de bureau et de la caractérisation de la végétation 	<p>JOUR 3</p> <ul style="list-style-type: none"> - iPhones : mesures en cours sur le troupeau 2A - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Observations aux jumelles : troupeau 2A 	<p>JOUR 4</p> <ul style="list-style-type: none"> - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Progression du travail de bureau et de la caractérisation de la végétation - iPhones : fin des mesures sur le troupeau 2A - Manipulations du bétail : ramassage des troupeaux 1 et 2, récupération du matériel sur le troupeau 2A et mise en place des licols sur le troupeau 1A - Action cyclique : pesée des bêtes
<p>JOUR 5</p> <ul style="list-style-type: none"> - iPhones : recharge des batteries - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Action cyclique : prélèvements végétaux et déplacement des exclos en pâture 1 - Manipulations du bétail : ramassage du troupeau 1 	<p>JOUR 6</p> <ul style="list-style-type: none"> - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Manipulations du bétail : mise en place des iPhones sur le troupeau 1A - iPhones : début des mesures sur le troupeau 1A - Prises vidéo : troupeau 1A - Progression du travail de bureau et de la caractérisation de la végétation 	<p>JOUR 7</p> <ul style="list-style-type: none"> - iPhones : mesures en cours sur le troupeau 1A - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Observations aux jumelles : troupeau 1A 	<p>JOUR 8</p> <ul style="list-style-type: none"> - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Progression du travail de bureau et de la caractérisation de la végétation - iPhones : fin des mesures sur le troupeau 1A - Manipulations du bétail : ramassage des troupeaux 1 et 2, récupération du matériel sur le troupeau 1A et mise en place des licols sur le troupeau 2B
<p>JOUR 9</p> <ul style="list-style-type: none"> - iPhones : recharge des batteries - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Progression du travail de bureau et de la caractérisation de la végétation - Manipulations du bétail : ramassage du troupeau 2 	<p>JOUR 10</p> <ul style="list-style-type: none"> - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Manipulations du bétail : mise en place des iPhones sur le troupeau 2B - iPhones : début des mesures sur le troupeau 2B - Prises vidéo : troupeau 2B - Progression du travail de bureau et de la caractérisation de la végétation 	<p>JOUR 11</p> <ul style="list-style-type: none"> - iPhones : mesures en cours sur le troupeau 2B - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Observations aux jumelles : troupeau 2B 	<p>JOUR 12</p> <ul style="list-style-type: none"> - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Progression du travail de bureau et de la caractérisation de la végétation - iPhones : fin des mesures sur le troupeau 2B - Manipulations du bétail : ramassage des troupeaux 1 et 2, récupération du matériel sur le troupeau 2B et mise en place des licols sur le troupeau 1B
<p>JOUR 13</p> <ul style="list-style-type: none"> - iPhones : recharge des batteries - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Progression du travail de bureau et de la caractérisation de la végétation - Manipulations du bétail : ramassage du troupeau 1 	<p>JOUR 14</p> <ul style="list-style-type: none"> - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Mise en place des iPhones sur le troupeau 1B - iPhones : début des mesures sur le troupeau 1B - Prises vidéo : troupeau 1B - Progression du travail de bureau et de la caractérisation de la végétation 	<p>JOUR 15</p> <ul style="list-style-type: none"> - iPhones : mesures en cours sur le troupeau 1B - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Observations aux jumelles : troupeau 1B 	<p>JOUR 16</p> <ul style="list-style-type: none"> - Check-up journalier : vérification de l'état des clôtures et des exclos, de la présence des bêtes et du niveau d'eau de l'abreuvoir - Progression du travail de bureau et de la caractérisation de la végétation - iPhones : fin des mesures sur le troupeau 1B - Manipulations du bétail : ramassage des troupeaux 1 et 2, récupération du matériel sur le troupeau 1B et mise en place des licols sur le troupeau 2A

Annexe 6 : tableau récapitulatif des données récoltées

Données	Variabes	Utilisation
Carte de la pâture : - Angles de clôtures - Angles des pistes - Ponts canadiens - Portes - Abreuvoirs - Extrémités des flaques - Bacs à sels	- Référence (texte) - Latitude (degrés décimaux) - Longitude (degrés décimaux)	- Traitement statistique des vitesses et des distances
Bouchées prélevées sur des ligneux	- Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm) - Espèce végétale (code) - Nombre de bouchées - Hauteur des bouchées (cm)	- Traitement statistique des observations directes continues
Attaques perpétrées sur d'autres animaux	- Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm) - Identité de la bête ciblée (code) - Intensité de l'attaque (variable ordinale)	- Traitement statistique des observations directes continues
Léchages d'autres animaux	- Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm) - Identité de la bête ciblée - Nombre de coups de langue	- Traitement statistique des observations directes continues
Meuglements	- Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm) - Nombre de cris	- Traitement statistique des observations directes continues
Flehms	- Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm)	- Traitement statistique des observations directes continues
Périodes passées allongé	- Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm) - Durée (minutes)	- Traitement statistique des observations directes continues
Bêtes présentes dans le groupe	- Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm) - Nombre de bêtes hors protocole - Identité des Ndamas (code) - Identité des zébus Goudalis (code) - identité des Ndapol (code) - identité des Zépol (code)	- Traitement statistique des observations directes discontinues

Bête la plus proche	<ul style="list-style-type: none"> - Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm) - Identité de la bête (code) - Distance (m) 	- Traitement statistique des observations directes discontinues
Position (iPhone)	<ul style="list-style-type: none"> - Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm:ss.SS) - Latitude (degrés décimaux) - Longitude (degrés décimaux) 	<ul style="list-style-type: none"> - Prétraitement des données provenant des iPhones - Traitement statistique des accès à l'eau et au sel - Traitement géostatistique de la position des bêtes et de ses liens avec la végétation
Vitesse (iPhone)	<ul style="list-style-type: none"> - Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm:ss.SS) - Latitude (degrés décimaux) - Longitude (degrés décimaux) - Vitesse (m/s) 	<ul style="list-style-type: none"> - Prétraitement des données provenant des iPhones - Traitement statistique de la vitesse des bêtes
Accélération gravitationnelle selon l'axe X (iPhone)	<ul style="list-style-type: none"> - Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm:ss.SS) - Latitude (degrés décimaux) - Longitude (degrés décimaux) - Accélération (g) 	- Prétraitement des données provenant des iPhones
Autres données provenant des capteurs de l'iPhone	<ul style="list-style-type: none"> - Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm:ss.SS) - Latitude (degrés décimaux) - Longitude (degrés décimaux) - Divers 	Aucune
Fichiers vidéo	<ul style="list-style-type: none"> - Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd'T'HH:mm) 	- Prétraitement des données provenant des iPhones
Pesées	<ul style="list-style-type: none"> - Pâture (code) - Race (code) - Individu (code) - Temps (MM-dd) - Poids (kg) - Gain quotidien moyen depuis la dernière pesée (kg/j) 	- Traitement statistique des sources de biais
Caractérisation de la végétation	<ul style="list-style-type: none"> - Pâture (code) - Latitude (degrés décimaux) - Longitude (degrés décimaux) - Présence de graminées rampantes ou gazonnantes (variable ordinale) - Présence de graminées cespiteuses de moins de 1 m de haut (variable ordinale) - Présence de graminées cespiteuses de plus de 1 m de haut (variable ordinale) 	- Traitement géostatistique de la position des bêtes et de ses liens avec la végétation

	<ul style="list-style-type: none"> - Présence de <i>Calopogonium mucunoides</i> Desv. (variable ordinale) - Présence de <i>Sida cf. acuta</i> Burm.f. (variable ordinale) - Présence de <i>Mitracarpus hirtus</i> L. (variable ordinale) - Présence de végétation herbacée non reprise dans les 6 critères précédents (variable ordinale) - Présence de végétation arbustive à l'exception de <i>C. odorata</i> (variable ordinale) - Présence de <i>C. odorata</i> (variable ordinale) - Présence d'abris contre le soleil et la pluie (variable ordinale) 	
Biomasse disponible	<ul style="list-style-type: none"> - Pâture (code) - Latitude (degrés décimaux) - Longitude (degrés décimaux) - Temps (MM-dd) - Matière fraîche (g/m²) - Matière sèche (g/m²) - Teneur en matière sèche (%) 	Aucune
Production de biomasse	<ul style="list-style-type: none"> - Pâture (code) - Latitude (degrés décimaux) - Longitude (degrés décimaux) - Temps initial (MM-dd) - Temps final (MM-dd) - Matière fraîche (g/m² j) - Matière sèche (g/m² j) - Teneur en matière sèche (%) 	Aucune
Echantillon végétal	<ul style="list-style-type: none"> - Pâture (code) - Latitude (degrés décimaux) - Longitude (degrés décimaux) - Temps (MM-dd) - Exclos (oui/non) 	Aucune
Pluviométrie	<ul style="list-style-type: none"> - Temps (MM-dd) - Pluviométrie (mm) 	- Traitement statistique des sources de biais

DATE :						ANIMAL :					Distance
H	Bêtes présentes dans le groupe (A/B/AB)					Bête la plus proche (A/B)					
	HP (Nb)	Ndama	Zébu	Ndapol	Zépol	HP (X)	Ndama	Zébu	Ndapol	Zépol	
6H30											
6H40											
6H50											
7H00											
7H10											
7H20											
7H30											
7H40											
7H50											
8H00											
8H10											
8H20											
8H30											
8H40											
8H50											
9H00											
9H10											
9H20											
9H30											
9H40											
9H50											
10H00											
10H10											
10H20											
10H30											
10H40											
10H50											
11H00											
11H10											
11H20											
11H30											
11H40											
11H50											
12H00											
12H10											
12H20											

DATE :						ANIMAL :					Distance
H	Bêtes présentes dans le groupe (A/B/AB)					Bête la plus proche (A/B)					
	HP (Nb)	Ndama	Zébu	Ndapol	Zépol	HP (X)	Ndama	Zébu	Ndapol	Zépol	
12H30											
12H40											
12H50											
13H00											
13H10											
13H20											
13H30											
13H40											
13H50											
14H00											
14H10											
14H20											
14H30											
14H40											
14H50											
15H00											
15H10											
15H20											
15H30											
15H40											
15H50											
16H00											
16H10											
16H20											
16H30											
16H40											
16H50											
17H00											
17H10											
17H20											
17H30											
17H40											
17H50											
18H00											
18H10											
18H20											

Annexe 8 : schéma de la protection de l'iPhone et de la batterie

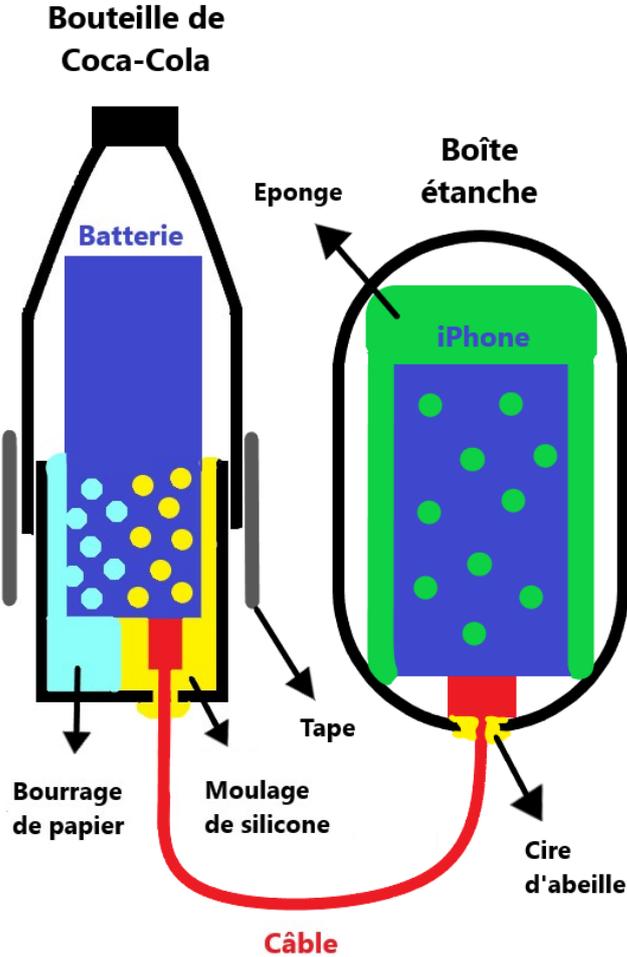


Schéma de la protection de l'iPhone et de la batterie. Gauthier Renard, 2019

Annexe 10 : photos des exclos utilisés pour les prélèvements végétaux.



Modèle d'exclos numéro 1. Gauthier Renard, 2018



Modèle d'exclos numéro 2. Gauthier Renard, 2018

Annexe 11 : description du script du traitement statistique des sources de biais

1. *Méthode utilisée pour les deux variables :*
 - 1.1. Une matrice à 3 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus) est remplie manuellement avec les données.
 - 1.2. Les moyennes par pâture, par race et « globale » sont réalisées et stockées dans un vecteur.
 - 1.3. Les écarts-types par pâture, par race et « global » sont réalisés et stockés dans un vecteur.
 - 1.4. L'analyse de la variance est réalisée et les p-values « pâture », « race » et « interaction » sont stockées dans un vecteur.
 - 1.5. Les vecteurs « moyennes », « écarts-types » et « p-values » sont fusionnés et réarrangés pour former un vecteur « statistiques ».
2. Les vecteurs « statistiques » des deux variables sont compilés en une matrice et celle-ci est écrite au format « xls ».

Annexe 12 : description du script du traitement statistique des observations directes continues

1. Le fichier « xlsx » contenant l'ensemble des données est lu.
2. *Méthode utilisée pour toutes les données de type « comptage par bête » (exemple type : le nombre de ligneux broutés) :*
 - 2.1. Une matrice à 3 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus) est remplie de zéros.
 - 2.2. Une boucle est créée afin de lire les données ligne par ligne. A chaque ligne, si une valeur est présente pour la variable considérée, la cellule correspondant à la bête à laquelle se réfère la ligne est incrémentée de 1.
 - 2.3. Les moyennes par pâture, par race et « globale » sont réalisées et stockées dans un vecteur.
 - 2.4. Les écarts-types par pâture, par race et « global » sont réalisés et stockés dans un vecteur.
 - 2.5. L'analyse de la variance est réalisée et les p-values « pâture », « race » et « interaction » sont stockées dans un vecteur.
 - 2.6. Les vecteurs « moyennes », « écarts-types » et « p-values » sont fusionnés et réarrangés pour former un vecteur « statistiques »
3. *Méthode utilisée pour toutes les données de type « somme par bête » (exemple type : le nombre total de bouchées prélevées sur des ligneux) :*
 - 3.1. Une matrice à 3 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus) est remplie de zéros.
 - 3.2. Une boucle est créée afin de lire les données ligne par ligne. A chaque ligne, si une valeur est présente pour la variable considérée, la cellule correspondant à la bête à laquelle se réfère la ligne est incrémentée de cette valeur.
 - 3.3. Les moyennes par pâture, par race et « globale » sont réalisées et stockées dans un vecteur.
 - 3.4. Les écarts-types par pâture, par race et « global » sont réalisés et stockés dans un vecteur.
 - 3.5. L'analyse de la variance est réalisée et les p-values « pâture », « race » et « interaction » sont stockées dans un vecteur.
 - 3.6. Les vecteurs « moyennes », « écarts-types » et « p-values » sont fusionnés et réarrangés pour former un vecteur « statistiques »
4. *Méthode utilisée pour toutes les données de type « moyenne par bête » (exemple type : la hauteur des bouchées prélevées sur des ligneux) :*
 - 4.1. Une matrice à 4 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus, dimension 4 : somme/comptage) est remplie de zéros.
 - 4.2. Une boucle est créée afin de lire les données ligne par ligne. A chaque ligne, si une valeur est présente pour la variable considérée, la première position de la dimension 4 correspondant à la bête (dimensions 1 à 3) est incrémentée de cette valeur et la seconde position de la dimension 4 correspondant à la bête (dimensions 1 à 3) est incrémentée de 1.
 - 4.3. Les cellules en position 1 de la dimension 4 sont divisées par les cellules en position 2 de la dimension 4. Cette dernière position est ensuite supprimée.

- 4.4. Les moyennes par pâture, par race et « globale » sont réalisées et stockées dans un vecteur.
- 4.5. Les écarts-types par pâture, par race et « global » sont réalisés et stockés dans un vecteur.
- 4.6. L'analyse de la variance est réalisée et les p-values « pâture », « race » et « interaction » sont stockées dans un vecteur.
- 4.7. Les vecteurs « moyennes », « écarts-types » et « p-values » sont fusionnés et réarrangés pour former un vecteur « statistiques ».
5. Les vecteurs « statistiques » de l'ensemble des variables sont compilés en une matrice et celle-ci est écrite au format « xls ».
6. Un fichier « xls » comprenant le nom des lignes dans le bon ordre est écrit afin d'éviter toute erreur lors de la mise en forme de la matrice de l'étape 5.

Annexe 13 : description du script du traitement statistique

des observations directes discontinues

1. Le fichier « xlsx » contenant l'ensemble des données et lu et remanié.
2. La bête le plus souvent la plus proche est identifiée pour chaque bête :
 - 2.1. Une matrice à 2 dimensions (dimension 1 : bêtes, dimension 2 : bêtes pouvant potentiellement être la plus proche) est remplie de zéro.
 - 2.2. Une boucle est créée afin de lire les données de départ ligne par ligne. A chaque ligne, la cellule correspondant au couple « bête – bête la plus proche » en présence est incrémentée de 1.
 - 2.3. Une matrice à 2 dimensions (dimension 1 : bêtes, dimension 2 : bêtes pouvant potentiellement être la plus proche) est remplie de « NaN ».
 - 2.4. Une boucle est créée afin de lire la matrice résultant de l'étape 2.2 ligne par ligne. A chaque ligne, une valeur de 1 est associée au couple « bête – bête la plus proche » dans le ou les cas où il correspond au maximum d'occurrence.
 - 2.5. Afin de départager les « bête la plus proche » ayant la même occurrence, celle se trouvant à la distance moyenne la plus petite est sélectionnée. Pour ce faire, une boucle est créée afin de lire la matrice résultant de l'étape 2.3 ligne par ligne. Lorsque le cas se présente, les valeurs « 1 » sont remplacées par la somme des distances, déterminée au moyen d'une boucle lisant les données de départ. La plus petite des valeurs est remplacée par un « 1 » et la plus grande par un « NaN ».
 - 2.6. La matrice résultant de l'étape 2.5 est transformée en une nouvelle matrice à 2 dimensions (dimension 1 : bêtes, dimension 2 : pâture, race (bête), individu (bête), race (bête la plus proche), individu (bête la plus proche)).
3. *Méthode utilisée pour les données de type « moyenne par bête » (exemples types : le nombre total de bêtes présentes dans le groupe et la distance à laquelle se trouve la bête la plus proche) :*
 - 3.1. Une matrice à 4 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus, dimension 4 : somme/comptage) est remplie de zéros.
 - 3.2. Une boucle est créée afin de lire les données ligne par ligne.
 - Dans le cas d'une moyenne de comptages :

A chaque ligne, la première position de la dimension 4 correspondant à la bête (dimension 1 à 3) est incrémentée du résultat du comptage à effectuer et la seconde position de la dimension 4 correspondant à la bête (dimensions 1 à 3) est incrémentée de 1.
 - Dans le cas d'une moyenne de valeurs :

A chaque ligne, si une valeur est présente pour la variable considérée, la première position de la dimension 4 correspondant à la bête (dimensions 1 à 3) est incrémentée de cette valeur et la seconde position de la dimension 4 correspondant à la bête (dimensions 1 à 3) est incrémentée de 1.
 - 3.3. Les cellules en position 1 de la dimension 4 sont divisées par les cellules en position 2 de la dimension 4. Cette dernière position est ensuite supprimée..
 - 3.4. Les moyennes par pâture, par race et « globale » sont réalisées et stockées dans un vecteur.

- 3.5. Les écarts-types par pâture, par race et « global » sont réalisés et stockés dans un vecteur.
- 3.6. L'analyse de la variance est réalisée et les p-values « pâture », « race » et « interaction » sont stockées dans un vecteur.
- 3.7. Les vecteurs « moyennes », « écarts-types » et « p-values » sont fusionnés et réarrangés pour former un vecteur « statistiques ».
4. *Méthode utilisée pour les données de type « comptage par bête » (exemples types : le temps passé seul et le temps passé avec au moins un ndama) :*
 - 4.1. Une matrice est remplie de zéros.
 - Dans le cas d'une somme qui n'est pas associée à une catégorie de bête :
La matrice créée possède 3 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus).
 - Dans le cas d'une somme qui est associée à une catégorie de bête :
La matrice créée possède 4 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus, dimension 4 : catégories de bête).
 - 4.2. Une boucle est créée afin de lire les données de départ ligne par ligne. A chaque ligne, les conditions correspondant à l'objet du comptage sont posées et, dans le cas où celles-ci se vérifient, la cellule correspondant à la ligne à laquelle se réfère la bête est incrémentée de 1. Dans le cas des sommes associées à une catégorie de bêtes, les comptages associés aux 5 catégories peuvent être réalisés simultanément grâce à la dimension 4.
 - 4.3. Les comptages ayant pour objet des périodes de 10 minutes, ils sont donc divisés par 6 afin d'obtenir un nombre d'heures.
 - 4.4. Les moyennes par pâture, par race et « globale » sont réalisées et stockées dans un vecteur (somme non associée à une catégorie de bête) ou une matrice (somme associée à une catégorie de bête).
 - 4.5. Les écarts-types par pâture, par race et « global » sont réalisés et stockés dans un vecteur (somme non associée à une catégorie de bête) ou une matrice (somme associée à une catégorie de bête).
 - 4.6. L'analyse de la variance est réalisée et les p-values « pâture », « race » et « interaction » sont stockées dans un vecteur (somme non associée à une catégorie de bête) ou une matrice (somme associée à une catégorie de bête).
 - 4.7. Les vecteurs ou les matrices « moyennes », « écarts-types » et « p-values » sont fusionnés et réarrangés pour former un vecteur ou une matrice « statistiques ».
5. Les vecteurs ou matrices « statistiques » de l'ensemble des variables sont compilés en une matrice et celle-ci est écrite au format « xls ».
6. Un fichier « xls » comprenant le nom des lignes dans le bon ordre est écrit afin d'éviter toute erreur lors de la mise en forme de la matrice de l'étape 5.

Annexe 14 : script du traitement statistique des sources de biais

```

1.      %%% biais
2.      %% entrer donnees gqm
3.      gqm(:,1) = [-0.20 -0.30 -0.16 0.71 ; -0.84 0.09 0.18 -0.07] ;
4.      gqm(:,2) = [0.11 0.46 -0.05 -0.02 ; 0.75 -0.18 0.36 0.29] ;
5.      %% stat gqm
6.      % moyenne
7.      gqm_global = mean(mean(mean(gqm)));
8.      gqm_race = mean(mean(gqm,3),1) ;
9.      gqm_pature = mean(mean(gqm,3),2) ;
10.     gqm_moy = [gqm_global gqm_pature gqm_race] ;
11.     % ecart-type
12.     gqm_global_M = ones(2,4,2)*gqm_global ;
13.     gqm_global_DM = gqm - gqm_global_M ;
14.     gqm_global_DM2 = gqm_global_DM.*gqm_global_DM ;
15.     gqm_global_V = mean(mean(mean(gqm_global_DM2)));
16.     gqm_global_ET = sqrt(gqm_global_V) ;
17.     gqm_race_M = ones(2,4,2) ;
18.     for i_race1 = 1:2
19.     for i_race2 = 1:2
20.     gqm_race_M(i_race1, :, i_race2) = gqm_race ;
21.     end
22.     end
23.     gqm_race_DM = gqm - gqm_race_M ;
24.     gqm_race_DM2 = gqm_race_DM.*gqm_race_DM ;
25.     gqm_race_V = mean(mean(gqm_race_DM2,3),1) ;
26.     gqm_race_ET = sqrt(gqm_race_V) ;
27.     gqm_pature_M = ones(2,4,2) ;
28.     for i_pature1 = 1:4
29.     for i_pature2 = 1:2
30.     gqm_pature_M(:, i_pature1, i_pature2) = gqm_pature ;
31.     end
32.     end
33.     gqm_pature_DM = gqm - gqm_pature_M ;
34.     gqm_pature_DM2 = gqm_pature_DM.*gqm_pature_DM ;
35.     gqm_pature_V = mean(mean(gqm_pature_DM2,3),2) ;
36.     gqm_pature_ET = sqrt(gqm_pature_V) ;
37.     gqm_ET = [gqm_global_ET gqm_pature_ET gqm_race_ET] ;
38.     % ANOVA
39.     gqm_AV = [gqm(1, :, 1) gqm(1, :, 2) gqm(2, :, 1) gqm(1, :, 2)] ;
40.     f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
41.     f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
42.     gqm_pvalue = anovan(gqm_AV, {f1, f2}, 'full') ;
43.     gqm_pvalue = flipr(gqm_pvalue) ;
44.     close all force
45.     % fusion
46.     gqm_stat = [gqm_moy gqm_ET gqm_pvalue] ;
47.     clearvars -except gqm gqm_stat
48.     %% entrer donnees pluvio
49.     pluvio(:,1) = [12 12 0 0 ; 20.5 0 53.5 0] ;

```

```

50.     pluvio(:,2) = [0 0 0 0 ; 0 20 4.5 0] ;
51.     %% stat pluvio
52.     % moyenne
53.     pluvio_global = mean(mean(mean(pluvio)));
54.     pluvio_race = mean(mean(pluvio,3),1) ;
55.     pluvio_pature = mean(mean(pluvio,3),2) ;
56.     pluvio_moy = [pluvio_global pluvio_pature pluvio_race] ;
57.     % ecart-type
58.     pluvio_global_M = ones(2,4,2)*pluvio_global ;
59.     pluvio_global_DM = pluvio - pluvio_global_M ;
60.     pluvio_global_DM2 = pluvio_global_DM.*pluvio_global_DM ;
61.     pluvio_global_V = mean(mean(mean(pluvio_global_DM2)));
62.     pluvio_global_ET = sqrt(pluvio_global_V) ;
63.     pluvio_race_M = ones(2,4,2) ;
64.     for i_race1 = 1:2
65.     for i_race2 = 1:2
66.     pluvio_race_M(i_race1, :, i_race2) = pluvio_race ;
67.     end
68.     end
69.     pluvio_race_DM = pluvio - pluvio_race_M ;
70.     pluvio_race_DM2 = pluvio_race_DM.*pluvio_race_DM ;
71.     pluvio_race_V = mean(mean(pluvio_race_DM2,3),1) ;
72.     pluvio_race_ET = sqrt(pluvio_race_V) ;
73.     pluvio_pature_M = ones(2,4,2) ;
74.     for i_pature1 = 1:4
75.     for i_pature2 = 1:2
76.     pluvio_pature_M(:, i_pature1, i_pature2) = pluvio_pature ;
77.     end
78.     end
79.     pluvio_pature_DM = pluvio - pluvio_pature_M ;
80.     pluvio_pature_DM2 = pluvio_pature_DM.*pluvio_pature_DM ;
81.     pluvio_pature_V = mean(mean(pluvio_pature_DM2,3),2) ;
82.     pluvio_pature_ET = sqrt(pluvio_pature_V) ;
83.     pluvio_ET = [pluvio_global_ET pluvio_pature_ET pluvio_race_ET] ;
84.     % ANOVA
85.     pluvio_AV = [pluvio(1, :, 1) pluvio(1, :, 2) pluvio(2, :, 1) pluvio(1, :, 2)] ;
86.     f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
87.     f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
88.     pluvio_pvalue = anovan(pluvio_AV, {f1, f2}, 'full') ;
89.     pluvio_pvalue = flipr(pluvio_pvalue) ;
90.     close all force
91.     % fusion
92.     pluvio_stat = [pluvio_moy pluvio_ET pluvio_pvalue] ;
93.     clearvars -except gqm gqm_stat pluvio pluvio_stat
94.     %% sortie fichier
95.     C = [pluvio_stat ; gqm_stat] ;
96.     stat_gqm_pluvio = [C(:,1) C(:,8) C(:,15) C(:,2) C(:,9) C(:,3) ...
97.     C(:,10) C(:,16) C(:,4) C(:,11) C(:,5) C(:,12) C(:,6) C(:,13) ...
98.     C(:,7) C(:,14) C(:,17)] ;
99.     xlswrite('stat_gqm_pluvio3', stat_gqm_pluvio)
100.    clearvars -except stat_gqm_pluvio

```

Annexe 15 : script du traitement statistique

des observations directes continues

```
101.  %%% observations continues
102.  %% lecture donnees
103.  comport = xlsread('compil_comport.xlsx');
104.  %% nombre d attaques
105.  % comptage par bete
106.  attaque_nb = zeros(2,4,2);
107.  i=1;
108.  while i <= length(comport)
109.  for x = 1:2
110.  for y = 1:4
111.  for z = 1:2
112.  if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
113.  attaque_nb(x,y,z)=attaque_nb(x,y,z)+(~isnan(comport(i,7)));
114.  end
115.  end
116.  end
117.  end
118.  i=i+1;
119.  end
120.  % moyenne
121.  attaque_nb_global = mean(mean(mean(attaque_nb)));
122.  attaque_nb_race = mean(mean(attaque_nb,3),1);
123.  attaque_nb_pature = mean(mean(attaque_nb,3),2);
124.  attaque_nb_moy = [attaque_nb_global attaque_nb_pature' attaque_nb_race];
125.  % ecart type
126.  attaque_nb_global_M = ones(2,4,2)*attaque_nb_global;
127.  attaque_nb_global_DM = attaque_nb - attaque_nb_global_M;
128.  attaque_nb_global_DM2 = attaque_nb_global_DM.*attaque_nb_global_DM;
129.  attaque_nb_global_V = mean(mean(mean(attaque_nb_global_DM2)));
130.  attaque_nb_global_ET = sqrt(attaque_nb_global_V);
131.  attaque_nb_race_M = ones(2,4,2);
132.  for i_race1 = 1:2
133.  for i_race2 = 1:2
134.  attaque_nb_race_M(i_race1, :, i_race2) = attaque_nb_race;
135.  end
136.  end
137.  attaque_nb_race_DM = attaque_nb - attaque_nb_race_M;
138.  attaque_nb_race_DM2 = attaque_nb_race_DM.*attaque_nb_race_DM;
139.  attaque_nb_race_V = mean(mean(attaque_nb_race_DM2,3),1);
140.  attaque_nb_race_ET = sqrt(attaque_nb_race_V);
141.  attaque_nb_pature_M = ones(2,4,2);
142.  for i_pature1 = 1:4
143.  for i_pature2 = 1:2
144.  attaque_nb_pature_M(:, i_pature1, i_pature2) = attaque_nb_pature;
145.  end
146.  end
147.  attaque_nb_pature_DM = attaque_nb - attaque_nb_pature_M;

148.  attaque_nb_pature_DM2 = attaque_nb_pature_DM.*attaque_nb_pature_DM;
149.  attaque_nb_pature_V = mean(mean(attaque_nb_pature_DM2,3),2);
150.  attaque_nb_pature_ET = sqrt(attaque_nb_pature_V);
151.  attaque_nb_ET = [attaque_nb_global_ET attaque_nb_pature_ET' ...
152.  attaque_nb_race_ET];
153.  % ANOVA
154.  attaque_nb_AV = [attaque_nb(1, :, 1) attaque_nb(1, :, 2) ...
155.  attaque_nb(2, :, 1) attaque_nb(1, :, 2)];
156.  f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
157.  f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
158.  attaque_nb_pvalue = anovan(attaque_nb_AV, {f1, f2}, 'full');
159.  attaque_nb_pvalue = fliplr(attaque_nb_pvalue);
160.  close all force
161.  % fusion
162.  attaque_nb_stat = [attaque_nb_moy attaque_nb_ET attaque_nb_pvalue];
163.  clearvars -except comport attaque_nb attaque_nb_stat ...
164.  attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
165.  ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
166.  flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
167.  lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
168.  meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
169.  allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
170.  %% nombre de broutages
171.  % comptage par bete
172.  ligneux_nb = zeros(2,4,2);
173.  i=1;
174.  while i <= length(comport)
175.  for x = 1:2
176.  for y = 1:4
177.  for z = 1:2
178.  if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
179.  ligneux_nb(x,y,z)=ligneux_nb(x,y,z)+(~isnan(comport(i,8)));
180.  end
181.  end
182.  end
183.  end
184.  i=i+1;
185.  end
186.  % moyenne
187.  ligneux_nb_global = mean(mean(mean(ligneux_nb)));
188.  ligneux_nb_race = mean(mean(ligneux_nb,3),1);
189.  ligneux_nb_pature = mean(mean(ligneux_nb,3),2);
190.  ligneux_nb_moy = [ligneux_nb_global ligneux_nb_pature' ligneux_nb_race];
191.  % ecart type
192.  ligneux_nb_global_M = ones(2,4,2)*ligneux_nb_global;
193.  ligneux_nb_global_DM = ligneux_nb - ligneux_nb_global_M;
194.  ligneux_nb_global_DM2 = ligneux_nb_global_DM.*ligneux_nb_global_DM;
195.  ligneux_nb_global_V = mean(mean(mean(ligneux_nb_global_DM2)));
196.  ligneux_nb_global_ET = sqrt(ligneux_nb_global_V);
197.  ligneux_nb_race_M = ones(2,4,2);
198.  for i_race1 = 1:2
199.  for i_race2 = 1:2
```

```

200. ligneux_nb_race_M(i_race1, :, i_race2) = ligneux_nb_race ;
201. end
202. end
203. ligneux_nb_race_DM = ligneux_nb - ligneux_nb_race_M ;
204. ligneux_nb_race_DM2 = ligneux_nb_race_DM.*ligneux_nb_race_DM ;
205. ligneux_nb_race_V = mean(mean(ligneux_nb_race_DM2,3),1) ;
206. ligneux_nb_race_ET = sqrt(ligneux_nb_race_V) ;
207. ligneux_nb_pature_M = ones(2,4,2) ;
208. for i_pature1 = 1:4
209. for i_pature2 = 1:2
210. ligneux_nb_pature_M(:,i_pature1,i_pature2) = ligneux_nb_pature ;
211. end
212. end
213. ligneux_nb_pature_DM = ligneux_nb - ligneux_nb_pature_M ;
214. ligneux_nb_pature_DM2 = ligneux_nb_pature_DM.*ligneux_nb_pature_DM ;
215. ligneux_nb_pature_V = mean(mean(ligneux_nb_pature_DM2,3),2) ;
216. ligneux_nb_pature_ET = sqrt(ligneux_nb_pature_V) ;
217. ligneux_nb_ET = [ligneux_nb_global_ET ligneux_nb_pature_ET' ...
218. ligneux_nb_race_ET] ;
219. % ANOVA
220. ligneux_nb_AV = [ligneux_nb(1, :, 1) ligneux_nb(1, :, 2) ...
221. ligneux_nb(2, :, 1) ligneux_nb(1, :, 2)] ;
222. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
223. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
224. ligneux_nb_pvalue = anovan(ligneux_nb_AV, {f1, f2}, 'full') ;
225. ligneux_nb_pvalue = fliplr(ligneux_nb_pvalue) ;
226. close all force
227. % fusion
228. ligneux_nb_stat = [ligneux_nb_moy ligneux_nb_ET ligneux_nb_pvalue] ;
229. clearvars -except comport attaque_nb attaque_nb_stat ...
230. attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
231. ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
232. flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
233. lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
234. meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
235. allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
236. %% nombre de flehmen
237. % comptage par bete
238. flehmen_nb = zeros(2,4,2) ;
239. i=1 ;
240. while i <= length(comport)
241. for x = 1:2
242. for y = 1:4
243. for z = 1:2
244. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
245. flehmen_nb(x,y,z)=flehmen_nb(x,y,z)+(~isnan(comport(i,11))) ;
246. end
247. end
248. end
249. end
250. i=i+1 ;
251. end
252. % moyenne
253. flehmen_nb_global = mean(mean(mean(flehmen_nb))) ;
254. flehmen_nb_race = mean(mean(flehmen_nb,3),1) ;
255. flehmen_nb_pature = mean(mean(flehmen_nb,3),2) ;
256. flehmen_nb_moy = ...
257. [flehmen_nb_global flehmen_nb_pature' flehmen_nb_race] ;
258. % ecart type
259. flehmen_nb_global_M = ones(2,4,2)*flehmen_nb_global ;
260. flehmen_nb_global_DM = flehmen_nb - flehmen_nb_global_M ;
261. flehmen_nb_global_DM2 = flehmen_nb_global_DM.*flehmen_nb_global_DM ;
262. flehmen_nb_global_V = mean(mean(mean(flehmen_nb_global_DM2))) ;
263. flehmen_nb_global_ET = sqrt(flehmen_nb_global_V) ;
264. flehmen_nb_race_M = ones(2,4,2) ;
265. for i_race1 = 1:2
266. for i_race2 = 1:2
267. flehmen_nb_race_M(i_race1, :, i_race2) = flehmen_nb_race ;
268. end
269. end
270. flehmen_nb_race_DM = flehmen_nb - flehmen_nb_race_M ;
271. flehmen_nb_race_DM2 = flehmen_nb_race_DM.*flehmen_nb_race_DM ;
272. flehmen_nb_race_V = mean(mean(flehmen_nb_race_DM2,3),1) ;
273. flehmen_nb_race_ET = sqrt(flehmen_nb_race_V) ;
274. flehmen_nb_pature_M = ones(2,4,2) ;
275. for i_pature1 = 1:4
276. for i_pature2 = 1:2
277. flehmen_nb_pature_M(:,i_pature1,i_pature2) = flehmen_nb_pature ;
278. end
279. end
280. flehmen_nb_pature_DM = flehmen_nb - flehmen_nb_pature_M ;
281. flehmen_nb_pature_DM2 = ...
282. flehmen_nb_pature_DM.*flehmen_nb_pature_DM ;
283. flehmen_nb_pature_V = mean(mean(flehmen_nb_pature_DM2,3),2) ;
284. flehmen_nb_pature_ET = sqrt(flehmen_nb_pature_V) ;
285. flehmen_nb_ET = [flehmen_nb_global_ET flehmen_nb_pature_ET' ...
286. flehmen_nb_race_ET] ;
287. % ANOVA
288. flehmen_nb_AV = [flehmen_nb(1, :, 1) flehmen_nb(1, :, 2) ...
289. flehmen_nb(2, :, 1) flehmen_nb(1, :, 2)] ;
290. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
291. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
292. flehmen_nb_pvalue = anovan(flehmen_nb_AV, {f1, f2}, 'full') ;
293. flehmen_nb_pvalue = fliplr(flehmen_nb_pvalue) ;
294. close all force
295. % fusion
296. flehmen_nb_stat = [flehmen_nb_moy flehmen_nb_ET flehmen_nb_pvalue] ;
297. clearvars -except comport attaque_nb attaque_nb_stat ...
298. attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
299. ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
300. flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
301. lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
302. meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
303. allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat

```

```

304. %% nombre de lechages
305. % comptage par bete
306. lechage_nb = zeros(2,4,2);
307. i=1;
308. while i <= length(comport)
309.     for x = 1:2
310.         for y = 1:4
311.             for z = 1:2
312.                 if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
313.                     lechage_nb(x,y,z)=lechage_nb(x,y,z)+(~isnan(comport(i,13)));
314.                 end
315.             end
316.         end
317.     end
318.     i=i+1;
319. end
320. % moyenne
321. lechage_nb_global = mean(mean(mean(lechage_nb)));
322. lechage_nb_race = mean(mean(lechage_nb,3),1);
323. lechage_nb_pature = mean(mean(lechage_nb,3),2);
324. lechage_nb_moy = [lechage_nb_global lechage_nb_pature' lechage_nb_race];
325. % ecart type
326. lechage_nb_global_M = ones(2,4,2)*lechage_nb_global;
327. lechage_nb_global_DM = lechage_nb - lechage_nb_global_M;
328. lechage_nb_global_DM2 = lechage_nb_global_DM.*lechage_nb_global_DM;
329. lechage_nb_global_V = mean(mean(mean(lechage_nb_global_DM2)));
330. lechage_nb_global_ET = sqrt(lechage_nb_global_V);
331. lechage_nb_race_M = ones(2,4,2);
332. for i_race1 = 1:2
333.     for i_race2 = 1:2
334.         lechage_nb_race_M(i_race1,;,i_race2) = lechage_nb_race;
335.     end
336. end
337. lechage_nb_race_DM = lechage_nb - lechage_nb_race_M;
338. lechage_nb_race_DM2 = lechage_nb_race_DM.*lechage_nb_race_DM;
339. lechage_nb_race_V = mean(mean(lechage_nb_race_DM2,3),1);
340. lechage_nb_race_ET = sqrt(lechage_nb_race_V);
341. lechage_nb_pature_M = ones(2,4,2);
342. for i_pature1 = 1:4
343.     for i_pature2 = 1:2
344.         lechage_nb_pature_M(:,i_pature1,i_pature2) = lechage_nb_pature;
345.     end
346. end
347. lechage_nb_pature_DM = lechage_nb - lechage_nb_pature_M;
348. lechage_nb_pature_DM2 = lechage_nb_pature_DM.*lechage_nb_pature_DM;
349. lechage_nb_pature_V = mean(mean(lechage_nb_pature_DM2,3),2);
350. lechage_nb_pature_ET = sqrt(lechage_nb_pature_V);
351. lechage_nb_ET = [lechage_nb_global_ET lechage_nb_pature_ET' ...
352.                 lechage_nb_race_ET];
353. % ANOVA
354. lechage_nb_AV = [lechage_nb(1,;,1) lechage_nb(1,;,2) ...
355.                 lechage_nb(2,;,1) lechage_nb(1,;,2)];
356. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
357. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
358. lechage_nb_pvalue = anovan(lechage_nb_AV,{f1,f2},'full');
359. lechage_nb_pvalue = flipr(lechage_nb_pvalue');
360. close all force
361. % fusion
362. lechage_nb_stat = [lechage_nb_moy lechage_nb_ET lechage_nb_pvalue];
363. clearvars -except comport attaque_nb attaque_nb_stat ...
364. attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
365. ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
366. flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
367. lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
368. meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
369. allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
370. %% nombre de meuglements
371. % comptage par bete
372. meuh_nb = zeros(2,4,2);
373. i=1;
374. while i <= length(comport)
375.     for x = 1:2
376.         for y = 1:4
377.             for z = 1:2
378.                 if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
379.                     meuh_nb(x,y,z)=meuh_nb(x,y,z)+(~isnan(comport(i,14)));
380.                 end
381.             end
382.         end
383.     end
384.     i=i+1;
385. end
386. % moyenne
387. meuh_nb_global = mean(mean(mean(meuh_nb)));
388. meuh_nb_race = mean(mean(meuh_nb,3),1);
389. meuh_nb_pature = mean(mean(meuh_nb,3),2);
390. meuh_nb_moy = [meuh_nb_global meuh_nb_pature' meuh_nb_race];
391. % ecart type
392. meuh_nb_global_M = ones(2,4,2)*meuh_nb_global;
393. meuh_nb_global_DM = meuh_nb - meuh_nb_global_M;
394. meuh_nb_global_DM2 = meuh_nb_global_DM.*meuh_nb_global_DM;
395. meuh_nb_global_V = mean(mean(mean(meuh_nb_global_DM2)));
396. meuh_nb_global_ET = sqrt(meuh_nb_global_V);
397. meuh_nb_race_M = ones(2,4,2);
398. for i_race1 = 1:2
399.     for i_race2 = 1:2
400.         meuh_nb_race_M(i_race1,;,i_race2) = meuh_nb_race;
401.     end
402. end
403. meuh_nb_race_DM = meuh_nb - meuh_nb_race_M;
404. meuh_nb_race_DM2 = meuh_nb_race_DM.*meuh_nb_race_DM;
405. meuh_nb_race_V = mean(mean(meuh_nb_race_DM2,3),1);
406. meuh_nb_race_ET = sqrt(meuh_nb_race_V);
407. meuh_nb_pature_M = ones(2,4,2);

```

```

408. for i_pature1 = 1:4
409. for i_pature2 = 1:2
410. meuh_nb_pature_M(:,i_pature1,i_pature2) = meuh_nb_pature ;
411. end
412. end
413. meuh_nb_pature_DM = meuh_nb - meuh_nb_pature_M ;
414. meuh_nb_pature_DM2 = meuh_nb_pature_DM.*meuh_nb_pature_DM ;
415. meuh_nb_pature_V = mean(mean(meuh_nb_pature_DM2,3),2) ;
416. meuh_nb_pature_ET = sqrt(meuh_nb_pature_V) ;
417. meuh_nb_ET = [meuh_nb_global_ET meuh_nb_pature_ET' meuh_nb_race_ET] ;
418. % ANOVA
419. meuh_nb_AV = [meuh_nb(1,;,1) meuh_nb(1,;,2) ...
420.              meuh_nb(2,;,1) meuh_nb(1,;,2)] ;
421. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
422. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
423. meuh_nb_pvalue = anovan(meuh_nb_AV,{f1,f2},'full') ;
424. meuh_nb_pvalue = fliplr(meuh_nb_pvalue') ;
425. close all force
426. % fusion
427. meuh_nb_stat = [meuh_nb_moy meuh_nb_ET meuh_nb_pvalue] ;
428. clearvars -except comport attaque_nb attaque_nb_stat ...
429. attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
430. ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
431. flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
432. lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
433. meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
434. allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
435. %% nombre de periodes passees allonge
436. % comptage par bete
437. allonge_nb = zeros(2,4,2) ;
438. i=1 ;
439. while i <= length(comport)
440. for x = 1:2
441. for y = 1:4
442. for z = 1:2
443. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
444. allonge_nb(x,y,z)=allonge_nb(x,y,z)+(~isnan(comport(i,15))) ;
445. end
446. end
447. end
448. end
449. i=i+1 ;
450. end
451. % moyenne
452. allonge_nb_global = mean(mean(mean(allonge_nb))) ;
453. allonge_nb_race = mean(mean(allonge_nb,3),1) ;
454. allonge_nb_pature = mean(mean(allonge_nb,3),2) ;
455. allonge_nb_moy = [allonge_nb_global allonge_nb_pature' allonge_nb_race] ;
456. % ecart type
457. allonge_nb_global_M = ones(2,4,2)*allonge_nb_global ;
458. allonge_nb_global_DM = allonge_nb - allonge_nb_global_M ;
459. allonge_nb_global_DM2 = allonge_nb_global_DM.*allonge_nb_global_DM ;
460. allonge_nb_global_V = mean(mean(mean(allonge_nb_global_DM2))) ;
461. allonge_nb_global_ET = sqrt(allonge_nb_global_V) ;
462. allonge_nb_race_M = ones(2,4,2) ;
463. for i_race1 = 1:2
464. for i_race2 = 1:2
465. allonge_nb_race_M(i_race1,;,i_race2) = allonge_nb_race ;
466. end
467. end
468. allonge_nb_race_DM = allonge_nb - allonge_nb_race_M ;
469. allonge_nb_race_DM2 = allonge_nb_race_DM.*allonge_nb_race_DM ;
470. allonge_nb_race_V = mean(mean(allonge_nb_race_DM2,3),1) ;
471. allonge_nb_race_ET = sqrt(allonge_nb_race_V) ;
472. allonge_nb_pature_M = ones(2,4,2) ;
473. for i_pature1 = 1:4
474. for i_pature2 = 1:2
475. allonge_nb_pature_M(:,i_pature1,i_pature2) = allonge_nb_pature ;
476. end
477. end
478. allonge_nb_pature_DM = allonge_nb - allonge_nb_pature_M ;
479. allonge_nb_pature_DM2 = allonge_nb_pature_DM.*allonge_nb_pature_DM ;
480. allonge_nb_pature_V = mean(mean(allonge_nb_pature_DM2,3),2) ;
481. allonge_nb_pature_ET = sqrt(allonge_nb_pature_V) ;
482. allonge_nb_ET = [allonge_nb_global_ET allonge_nb_pature_ET' ...
483.                 allonge_nb_race_ET] ;
484. % ANOVA
485. allonge_nb_AV = [allonge_nb(1,;,1) allonge_nb(1,;,2) ...
486.                 allonge_nb(2,;,1) allonge_nb(1,;,2)] ;
487.
488. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
489. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
490. allonge_nb_pvalue = anovan(allonge_nb_AV,{f1,f2},'full') ;
491. allonge_nb_pvalue = fliplr(allonge_nb_pvalue') ;
492. close all force
493. % fusion
494. allonge_nb_stat = [allonge_nb_moy allonge_nb_ET allonge_nb_pvalue] ;
495. clearvars -except comport attaque_nb attaque_nb_stat ...
496. attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
497. ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
498. flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
499. lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
500. meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
501. allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
502. %% intensite des attaques
503. % moyenne par bete
504. attaque_val = zeros(2,4,2,2) ;
505. i=1 ;
506. while i <= length(comport)
507. for x = 1:2
508. for y = 1:4
509. for z = 1:2
510. if isnan(comport(i,7)) == 0
511. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z

```

```

512. attaque_val(x,y,z,1) = attaque_val(x,y,z,1) + (comport(i,7)) ;
513. attaque_val(x,y,z,2) = attaque_val(x,y,z,2) + 1 ;
514. end
515. end
516. end
517. end
518. end
519. i=i+1 ;
520. end
521. attaque_val(:,:,:) = attaque_val(:,:,:) ./ attaque_val(:,:,:) ;
522. attaque_val = attaque_val(:,:,:) ;
523. % moyenne
524. attaque_val_global = nanmean(nanmean(nanmean(attaque_val))) ;
525. attaque_val_race = nanmean(nanmean(attaque_val,3),1) ;
526. attaque_val_pature = nanmean(nanmean(attaque_val,3),2) ;
527. attaque_val_moy = [attaque_val_global attaque_val_pature' ...
528. attaque_val_race] ;
529. % ecart type
530. attaque_val_global_M = ones(2,4,2)*attaque_val_global ;
531. attaque_val_global_DM = attaque_val - attaque_val_global_M ;
532. attaque_val_global_DM2 = attaque_val_global_DM.*attaque_val_global_DM ;
533. attaque_val_global_V = ...
534. nanmean(nanmean(nanmean(attaque_val_global_DM2))) ;
535. attaque_val_global_ET = sqrt(attaque_val_global_V) ;
536. attaque_val_race_M = ones(2,4,2) ;
537. for i_race1 = 1:2
538. for i_race2 = 1:2
539. attaque_val_race_M(i_race1, :, i_race2) = attaque_val_race ;
540. end
541. end
542. attaque_val_race_DM = attaque_val - attaque_val_race_M ;
543. attaque_val_race_DM2 = attaque_val_race_DM.*attaque_val_race_DM ;
544. attaque_val_race_V = nanmean(nanmean(attaque_val_race_DM2,3),1) ;
545. attaque_val_race_ET = sqrt(attaque_val_race_V) ;
546. attaque_val_pature_M = ones(2,4,2) ;
547. for i_pature1 = 1:4
548. for i_pature2 = 1:2
549. attaque_val_pature_M(:, i_pature1, i_pature2) = attaque_val_pature ;
550. end
551. end
552. attaque_val_pature_DM = attaque_val - attaque_val_pature_M ;
553. attaque_val_pature_DM2 = attaque_val_pature_DM.*attaque_val_pature_DM ;
554. attaque_val_pature_V = nanmean(nanmean(attaque_val_pature_DM2,3),2) ;
555. attaque_val_pature_ET = sqrt(attaque_val_pature_V) ;
556. attaque_val_ET = [attaque_val_global_ET attaque_val_pature_ET' ...
557. attaque_val_race_ET] ;
558. % ANOVA
559. attaque_val_AV = [attaque_val(1, :, 1) attaque_val(1, :, 2) ...
560. attaque_val(2, :, 1) attaque_val(1, :, 2)] ;
561. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
562. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
563. attaque_val_pvalue = anovan(attaque_val_AV, {f1, f2}, 'full') ;

```

```

564. attaque_val_pvalue = flipr(attaque_val_pvalue') ;
565. close all force
566. % fusion
567. attaque_val_stat = [attaque_val_moy attaque_val_ET attaque_val_pvalue] ;
568. clearvars -except comport attaque_nb attaque_nb_stat ...
569. attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
570. ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
571. flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
572. lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
573. meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
574. allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
575. %% nombre total de bouchees
576. % somme par bete
577. ligneux_val = zeros(2,4,2) ;
578. i=1 ;
579. while i <= length(comport)
580. for x = 1:2
581. for y = 1:4
582. for z = 1:2
583. if isnan(comport(i,9)) == 0
584. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
585. ligneux_val(x,y,z) = ligneux_val(x,y,z) + (comport(i,9)) ;
586. end
587. end
588. end
589. end
590. end
591. i=i+1 ;
592. end
593. % moyenne
594. ligneux_val_global = nanmean(nanmean(nanmean(ligneux_val))) ;
595. ligneux_val_race = nanmean(nanmean(ligneux_val,3),1) ;
596. ligneux_val_pature = nanmean(nanmean(ligneux_val,3),2) ;
597. ligneux_val_moy = [ligneux_val_global ligneux_val_pature' ...
598. ligneux_val_race] ;
599. % ecart type
600. ligneux_val_global_M = ones(2,4,2)*ligneux_val_global ;
601. ligneux_val_global_DM = ligneux_val - ligneux_val_global_M ;
602. ligneux_val_global_DM2 = ligneux_val_global_DM.*ligneux_val_global_DM ;
603. ligneux_val_global_V = ...
604. nanmean(nanmean(nanmean(ligneux_val_global_DM2))) ;
605. ligneux_val_global_ET = sqrt(ligneux_val_global_V) ;
606. ligneux_val_race_M = ones(2,4,2) ;
607. for i_race1 = 1:2
608. for i_race2 = 1:2
609. ligneux_val_race_M(i_race1, :, i_race2) = ligneux_val_race ;
610. end
611. end
612. ligneux_val_race_DM = ligneux_val - ligneux_val_race_M ;
613. ligneux_val_race_DM2 = ligneux_val_race_DM.*ligneux_val_race_DM ;
614. ligneux_val_race_V = nanmean(nanmean(ligneux_val_race_DM2,3),1) ;
615. ligneux_val_race_ET = sqrt(ligneux_val_race_V) ;

```

```

616. ligneux_val_pature_M = ones(2,4,2);
617. for i_pature1 = 1:4
618.     for i_pature2 = 1:2
619.         ligneux_val_pature_M(:,i_pature1,i_pature2) = ligneux_val_pature;
620.     end
621. end
622. ligneux_val_pature_DM = ligneux_val - ligneux_val_pature_M;
623. ligneux_val_pature_DM2 = ligneux_val_pature_DM.*ligneux_val_pature_DM;
624. ligneux_val_pature_V = nanmean(nanmean(ligneux_val_pature_DM2,3),2);
625. ligneux_val_pature_ET = sqrt(ligneux_val_pature_V);
626. ligneux_val_ET = [ligneux_val_global_ET ligneux_val_pature_ET' ...
627.     ligneux_val_race_ET];
628. % ANOVA
629. ligneux_val_AV = [ligneux_val(1,;,1) ligneux_val(1,;,2) ...
630.     ligneux_val(2,;,1) ligneux_val(1,;,2)];
631.
632. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
633. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
634. ligneux_val_pvalue = anovan(ligneux_val_AV,{f1,f2},'full');
635. ligneux_val_pvalue = fliplr(ligneux_val_pvalue');
636. close all force
637. % fusion
638. ligneux_val_stat = [ligneux_val_moy ligneux_val_ET ligneux_val_pvalue];
639. clearvars -except comport attaque_nb attaque_nb_stat ...
640.     attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
641.     ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
642.     flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
643.     lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
644.     meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
645.     allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
646. %% nombre total de coups de langue
647. % somme par bete
648. lechage_val = zeros(2,4,2);
649. i=1;
650. while i <= length(comport)
651.     for x = 1:2
652.         for y = 1:4
653.             for z = 1:2
654.                 if isnan(comport(i,13)) == 0
655.                     if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
656.                         lechage_val(x,y,z) = lechage_val(x,y,z) + (comport(i,13));
657.                     end
658.                 end
659.             end
660.         end
661.     end
662.     i=i+1;
663. end
664. % moyenne
665. lechage_val_global = nanmean(nanmean(nanmean(lechage_val)));
666. lechage_val_race = nanmean(nanmean(lechage_val,3),1);
667. lechage_val_pature = nanmean(nanmean(lechage_val,3),2);

```

```

668. lechage_val_moy = [lechage_val_global lechage_val_pature' ...
669.     lechage_val_race];
670. % ecart type
671. lechage_val_global_M = ones(2,4,2).*lechage_val_global;
672. lechage_val_global_DM = lechage_val - lechage_val_global_M;
673. lechage_val_global_DM2 = lechage_val_global_DM.*lechage_val_global_DM;
674. lechage_val_global_V = ...
675.     nanmean(nanmean(nanmean(lechage_val_global_DM2)));
676. lechage_val_global_ET = sqrt(lechage_val_global_V);
677. lechage_val_race_M = ones(2,4,2);
678. for i_race1 = 1:2
679.     for i_race2 = 1:2
680.         lechage_val_race_M(i_race1,;,i_race2) = lechage_val_race;
681.     end
682. end
683. lechage_val_race_DM = lechage_val - lechage_val_race_M;
684. lechage_val_race_DM2 = lechage_val_race_DM.*lechage_val_race_DM;
685. lechage_val_race_V = nanmean(nanmean(lechage_val_race_DM2,3),1);
686. lechage_val_race_ET = sqrt(lechage_val_race_V);
687. lechage_val_pature_M = ones(2,4,2);
688. for i_pature1 = 1:4
689.     for i_pature2 = 1:2
690.         lechage_val_pature_M(:,i_pature1,i_pature2) = lechage_val_pature;
691.     end
692. end
693. lechage_val_pature_DM = lechage_val - lechage_val_pature_M;
694. lechage_val_pature_DM2 = lechage_val_pature_DM.*lechage_val_pature_DM;
695. lechage_val_pature_V = nanmean(nanmean(lechage_val_pature_DM2,3),2);
696. lechage_val_pature_ET = sqrt(lechage_val_pature_V);
697. lechage_val_ET = [lechage_val_global_ET lechage_val_pature_ET' ...
698.     lechage_val_race_ET];
699. % ANOVA
700. lechage_val_AV = [lechage_val(1,;,1) lechage_val(1,;,2) ...
701.     lechage_val(2,;,1) lechage_val(1,;,2)];
702. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
703. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
704. lechage_val_pvalue = anovan(lechage_val_AV,{f1,f2},'full');
705. lechage_val_pvalue = fliplr(lechage_val_pvalue');
706. close all force
707. % fusion
708. lechage_val_stat = [lechage_val_moy lechage_val_ET lechage_val_pvalue];
709. clearvars -except comport attaque_nb attaque_nb_stat ...
710.     attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
711.     ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
712.     flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
713.     lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
714.     meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
715.     allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
716. %% nombre total de cris
717. % somme par bete
718. meuh_val = zeros(2,4,2);
719. i=1;

```

```

720. while i <= length(comport)
721. for x = 1:2
722. for y = 1:4
723. for z = 1:2
724. if isnan(comport(i,14)) == 0
725. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
726. meuh_val(x,y,z) = meuh_val(x,y,z) + (comport(i,14)) ;
727. end
728. end
729. end
730. end
731. end
732. i=i+1 ;
733. end
734. % moyenne
735. meuh_val_global = nanmean(nanmean(nanmean(meuh_val)));
736. meuh_val_race = nanmean(nanmean(meuh_val,3),1) ;
737. meuh_val_pature = nanmean(nanmean(meuh_val,3),2) ;
738. meuh_val_moy = [meuh_val_global meuh_val_pature' meuh_val_race] ;
739. % ecart type
740. meuh_val_global_M = ones(2,4,2)*meuh_val_global ;
741. meuh_val_global_DM = meuh_val - meuh_val_global_M ;
742. meuh_val_global_DM2 = meuh_val_global_DM.*meuh_val_global_DM ;
743. meuh_val_global_V = nanmean(nanmean(nanmean(meuh_val_global_DM2)));
744. meuh_val_global_ET = sqrt(meuh_val_global_V) ;
745. meuh_val_race_M = ones(2,4,2) ;
746. for i_race1 = 1:2
747. for i_race2 = 1:2
748. meuh_val_race_M(i_race1, :, i_race2) = meuh_val_race ;
749. end
750. end
751. meuh_val_race_DM = meuh_val - meuh_val_race_M ;
752. meuh_val_race_DM2 = meuh_val_race_DM.*meuh_val_race_DM ;
753. meuh_val_race_V = nanmean(nanmean(meuh_val_race_DM2,3),1) ;
754. meuh_val_race_ET = sqrt(meuh_val_race_V) ;
755. meuh_val_pature_M = ones(2,4,2) ;
756. for i_pature1 = 1:4
757. for i_pature2 = 1:2
758. meuh_val_pature_M(:, i_pature1, i_pature2) = meuh_val_pature ;
759. end
760. end
761. meuh_val_pature_DM = meuh_val - meuh_val_pature_M ;
762. meuh_val_pature_DM2 = meuh_val_pature_DM.*meuh_val_pature_DM ;
763. meuh_val_pature_V = nanmean(nanmean(meuh_val_pature_DM2,3),2) ;
764. meuh_val_pature_ET = sqrt(meuh_val_pature_V) ;
765. meuh_val_ET = [meuh_val_global_ET meuh_val_pature_ET' ...
766. meuh_val_race_ET] ;
767. % ANOVA
768. meuh_val_AV = [meuh_val(1, :, 1) meuh_val(1, :, 2) ...
769. meuh_val(2, :, 1) meuh_val(1, :, 2)] ;
770. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
771. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;

```

```

772. meuh_val_pvalue = anovan(meuh_val_AV, {f1, f2}, 'full') ;
773. meuh_val_pvalue = flipplr(meuh_val_pvalue') ;
774. close all force
775. % fusion
776. meuh_val_stat = [meuh_val_moy meuh_val_ET meuh_val_pvalue] ;
777. clearvars -except comport attaque_nb attaque_nb_stat ...
778. attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
779. ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
780. flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
781. lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
782. meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
783. allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
784. %% duree totale passe allonge
785. % somme par bete
786. allonge_val = zeros(2,4,2) ;
787. i=1 ;
788. while i <= length(comport)
789. for x = 1:2
790. for y = 1:4
791. for z = 1:2
792. if isnan(comport(i,15)) == 0
793. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
794. allonge_val(x,y,z) = allonge_val(x,y,z) + (comport(i,15)) ;
795. end
796. end
797. end
798. end
799. end
800. i=i+1 ;
801. end
802. % moyenne
803. allonge_val_global = nanmean(nanmean(nanmean(allonge_val)));
804. allonge_val_race = nanmean(nanmean(allonge_val,3),1) ;
805. allonge_val_pature = nanmean(nanmean(allonge_val,3),2) ;
806. allonge_val_moy = [allonge_val_global allonge_val_pature' ...
807. allonge_val_race] ;
808. % ecart type
809. allonge_val_global_M = ones(2,4,2)*allonge_val_global ;
810. allonge_val_global_DM = allonge_val - allonge_val_global_M ;
811. allonge_val_global_DM2 = allonge_val_global_DM.*allonge_val_global_DM ;
812. allonge_val_global_V = ...
813. nanmean(nanmean(nanmean(allonge_val_global_DM2)));
814. allonge_val_global_ET = sqrt(allonge_val_global_V) ;
815. allonge_val_race_M = ones(2,4,2) ;
816. for i_race1 = 1:2
817. for i_race2 = 1:2
818. allonge_val_race_M(i_race1, :, i_race2) = allonge_val_race ;
819. end
820. end
821. allonge_val_race_DM = allonge_val - allonge_val_race_M ;
822. allonge_val_race_DM2 = allonge_val_race_DM.*allonge_val_race_DM ;
823. allonge_val_race_V = nanmean(nanmean(allonge_val_race_DM2,3),1) ;

```

```

824. allonge_val_race_ET = sqrt(allonge_val_race_V);
825. allonge_val_pature_M = ones(2,4,2);
826. for i_pature1 = 1:4
827. for i_pature2 = 1:2
828. allonge_val_pature_M(:,i_pature1,i_pature2) = allonge_val_pature;
829. end
830. end
831. allonge_val_pature_DM = allonge_val - allonge_val_pature_M;
832. allonge_val_pature_DM2 = allonge_val_pature_DM.*allonge_val_pature_DM;
833. allonge_val_pature_V = nanmean(nanmean(allonge_val_pature_DM2,3),2);
834. allonge_val_pature_ET = sqrt(allonge_val_pature_V);
835. allonge_val_ET = [allonge_val_global_ET allonge_val_pature_ET' ...
836. allonge_val_race_ET];
837. % ANOVA
838. allonge_val_AV = [allonge_val(1,;,1) allonge_val(1,;,2) ...
839. allonge_val(2,;,1) allonge_val(1,;,2)];
840. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
841. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
842. allonge_val_pvalue = anovan(allonge_val_AV,{f1,f2},'full');
843. allonge_val_pvalue = flipr(allonge_val_pvalue');
844. close all force
845. % fusion
846. allonge_val_stat = [allonge_val_moy allonge_val_ET allonge_val_pvalue];
847. clearvars -except comport attaque_nb attaque_nb_stat ...
848. attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
849. ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
850. flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
851. lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
852. meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
853. allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
854. %% hauteur du broutage
855. % moyenne par bete
856. ligneux_val2 = zeros(2,4,2,2);
857. i=1;
858. while i <= length(comport)
859. for x = 1:2
860. for y = 1:4
861. for z = 1:2
862. if isnan(comport(i,10)) == 0
863. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
864. ligneux_val2(x,y,z,1) = ligneux_val2(x,y,z,1) + (comport(i,10));
865. ligneux_val2(x,y,z,2) = ligneux_val2(x,y,z,2) + 1;
866. end
867. end
868. end
869. end
870. end
871. i=i+1;
872. end
873. ligneux_val2(:,;,1) = ligneux_val2(:,;,1)./ligneux_val2(:,;,2);
874. ligneux_val2 = ligneux_val2(:,;,1);
875. % moyenne

```

```

876. ligneux_val2_global = nanmean(nanmean(nanmean(ligneux_val2)));
877. ligneux_val2_race = nanmean(nanmean(ligneux_val2,3),1);
878. ligneux_val2_pature = nanmean(nanmean(ligneux_val2,3),2);
879. ligneux_val2_moy = [ligneux_val2_global ligneux_val2_pature' ...
880. ligneux_val2_race];
881. % ecart type
882. ligneux_val2_global_M = ones(2,4,2)*ligneux_val2_global;
883. ligneux_val2_global_DM = ligneux_val2 - ligneux_val2_global_M;
884. ligneux_val2_global_DM2 = ...
885. ligneux_val2_global_DM.*ligneux_val2_global_DM;
886. ligneux_val2_global_V = ...
887. nanmean(nanmean(nanmean(ligneux_val2_global_DM2)));
888. ligneux_val2_global_ET = sqrt(ligneux_val2_global_V);
889. ligneux_val2_race_M = ones(2,4,2);
890. for i_race1 = 1:2
891. for i_race2 = 1:2
892. ligneux_val2_race_M(i_race1,;,i_race2) = ligneux_val2_race;
893. end
894. end
895. ligneux_val2_race_DM = ligneux_val2 - ligneux_val2_race_M;
896. ligneux_val2_race_DM2 = ligneux_val2_race_DM.*ligneux_val2_race_DM;
897. ligneux_val2_race_V = nanmean(nanmean(ligneux_val2_race_DM2,3),1);
898. ligneux_val2_race_ET = sqrt(ligneux_val2_race_V);
899. ligneux_val2_pature_M = ones(2,4,2);
900. for i_pature1 = 1:4
901. for i_pature2 = 1:2
902. ligneux_val2_pature_M(:,i_pature1,i_pature2) = ligneux_val2_pature;
903. end
904. end
905. ligneux_val2_pature_DM = ligneux_val2 - ligneux_val2_pature_M;
906. ligneux_val2_pature_DM2 = ...
907. ligneux_val2_pature_DM.*ligneux_val2_pature_DM;
908. ligneux_val2_pature_V = nanmean(nanmean(ligneux_val2_pature_DM2,3),2);
909. ligneux_val2_pature_ET = sqrt(ligneux_val2_pature_V);
910. ligneux_val2_ET = [ligneux_val2_global_ET ligneux_val2_pature_ET' ...
911. ligneux_val2_race_ET];
912. % ANOVA
913. ligneux_val2_AV = [ligneux_val2(1,;,1) ligneux_val2(1,;,2) ...
914. ligneux_val2(2,;,1) ligneux_val2(1,;,2)];
915. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
916. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
917. ligneux_val2_pvalue = anovan(ligneux_val2_AV,{f1,f2},'full');
918. ligneux_val2_pvalue = flipr(ligneux_val2_pvalue');
919. close all force
920. % fusion
921. ligneux_val2_stat = [ligneux_val2_moy ligneux_val2_ET ...
922. ligneux_val2_pvalue];
923. clearvars -except comport attaque_nb attaque_nb_stat ...
924. attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
925. ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
926. flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
927. lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...

```

```

928. meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
929. allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
930. %% nombre d attaques sur ndama
931. % comptage par bete
932. attaque_nb_N = zeros(2,4,2) ;
933. i=1 ;
934. while i <= length(comport)
935. for x = 1:2
936. for y = 1:4
937. for z = 1:2
938. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
939. if comport(i,6)==111 || comport(i,6)==112 || comport(i,6)==211 ...
940. || comport(i,6)==212
941. attaque_nb_N(x,y,z)=attaque_nb_N(x,y,z)+(~isnan(comport(i,6))) ;
942. end
943. end
944. end
945. end
946. end
947. i=i+1 ;
948. end
949. % moyenne
950. attaque_nb_N_global = mean(mean(mean(attaque_nb_N)));
951. attaque_nb_N_race = mean(mean(attaque_nb_N,3),1);
952. attaque_nb_N_pature = mean(mean(attaque_nb_N,3),2);
953. attaque_nb_N_moy = [attaque_nb_N_global attaque_nb_N_pature' ...
954. attaque_nb_N_race];
955. % nombre d attaques sur zebu
956. % comptage par bete
957. attaque_nb_Z = zeros(2,4,2) ;
958. i=1 ;
959. while i <= length(comport)
960. for x = 1:2
961. for y = 1:4
962. for z = 1:2
963. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
964. if comport(i,6)==121 || comport(i,6)==122 || comport(i,6)==221 ...
965. || comport(i,6)==222
966. attaque_nb_Z(x,y,z)=attaque_nb_Z(x,y,z)+(~isnan(comport(i,6))) ;
967. end
968. end
969. end
970. end
971. end
972. i=i+1 ;
973. end
974. % moyenne
975. attaque_nb_Z_global = mean(mean(mean(attaque_nb_Z)));
976. attaque_nb_Z_race = mean(mean(attaque_nb_Z,3),1);
977. attaque_nb_Z_pature = mean(mean(attaque_nb_Z,3),2);
978. attaque_nb_Z_moy = [attaque_nb_Z_global attaque_nb_Z_pature' ...
979. attaque_nb_Z_race];

```

```

980. % nombre d attaques sur ndapol
981. % comptage par bete
982. attaque_nb_Np = zeros(2,4,2) ;
983. i=1 ;
984. while i <= length(comport)
985. for x = 1:2
986. for y = 1:4
987. for z = 1:2
988. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
989. if comport(i,6)==131 || comport(i,6)==132 || comport(i,6)==231 ...
990. || comport(i,6)==232
991. attaque_nb_Np(x,y,z)=attaque_nb_Np(x,y,z)+(~isnan(comport(i,6))) ;
992. end
993. end
994. end
995. end
996. end
997. i=i+1 ;
998. end
999. % moyenne
1000. attaque_nb_Np_global = mean(mean(mean(attaque_nb_Np)));
1001. attaque_nb_Np_race = mean(mean(attaque_nb_Np,3),1);
1002. attaque_nb_Np_pature = mean(mean(attaque_nb_Np,3),2);
1003. attaque_nb_Np_moy = [attaque_nb_Np_global attaque_nb_Np_pature' ...
1004. attaque_nb_Np_race];
1005. % nombre d attaques sur zepol
1006. % comptage par bete
1007. attaque_nb_Zp = zeros(2,4,2) ;
1008. i=1 ;
1009. while i <= length(comport)
1010. for x = 1:2
1011. for y = 1:4
1012. for z = 1:2
1013. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
1014. if comport(i,6)==141 || comport(i,6)==142 || comport(i,6)==241 ...
1015. || comport(i,6)==242
1016. attaque_nb_Zp(x,y,z)=attaque_nb_Zp(x,y,z)+(~isnan(comport(i,6))) ;
1017. end
1018. end
1019. end
1020. end
1021. end
1022. i=i+1 ;
1023. end
1024. % moyenne
1025. attaque_nb_Zp_global = mean(mean(mean(attaque_nb_Zp)));
1026. attaque_nb_Zp_race = mean(mean(attaque_nb_Zp,3),1);
1027. attaque_nb_Zp_pature = mean(mean(attaque_nb_Zp,3),2);
1028. attaque_nb_Zp_moy = [attaque_nb_Zp_global attaque_nb_Zp_pature' ...
1029. attaque_nb_Zp_race];
1030. % nombre d attaques sur bete hors protocole
1031. % comptage par bete

```

```

1032. attaque_nb_HP = zeros(2,4,2);
1033. i=1;
1034. while i <= length(comport)
1035. for x = 1:2
1036. for y = 1:4
1037. for z = 1:2
1038. if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
1039. if comport(i,6)==0
1040. attaque_nb_HP(x,y,z)=attaque_nb_HP(x,y,z)+(~isnan(comport(i,6)));
1041. end
1042. end
1043. end
1044. end
1045. end
1046. i=i+1;
1047. end
1048. % moyenne
1049. attaque_nb_HP_global = mean(mean(mean(attaque_nb_HP)));
1050. attaque_nb_HP_race = mean(mean(attaque_nb_HP,3),1);
1051. attaque_nb_HP_pature = mean(mean(attaque_nb_HP,3),2);
1052. attaque_nb_HP_moy = [attaque_nb_HP_global attaque_nb_HP_pature' ...
1053.     attaque_nb_HP_race];
1054. % ecart type du nombre d attaques sur ndama
1055. attaque_nb_N_global_M = ones(2,4,2)*attaque_nb_N_global;
1056. attaque_nb_N_global_DM = attaque_nb_N - attaque_nb_N_global_M;
1057. attaque_nb_N_global_DM2 = ...
1058.     attaque_nb_N_global_DM.*attaque_nb_N_global_DM;
1059. attaque_nb_N_global_V = mean(mean(mean(attaque_nb_N_global_DM2)));
1060. attaque_nb_N_global_ET = sqrt(attaque_nb_N_global_V);
1061. attaque_nb_N_race_M = ones(2,4,2);
1062. for i_race1 = 1:2
1063. for i_race2 = 1:2
1064. attaque_nb_N_race_M(i_race1, :, i_race2) = attaque_nb_N_race;
1065. end
1066. end
1067. attaque_nb_N_race_DM = attaque_nb_N - attaque_nb_N_race_M;
1068. attaque_nb_N_race_DM2 = ...
1069.     attaque_nb_N_race_DM.*attaque_nb_N_race_DM;
1070. attaque_nb_N_race_V = mean(mean(attaque_nb_N_race_DM2,3),1);
1071. attaque_nb_N_race_ET = sqrt(attaque_nb_N_race_V);
1072. attaque_nb_N_pature_M = ones(2,4,2);
1073. for i_pature1 = 1:4
1074. for i_pature2 = 1:2
1075. attaque_nb_N_pature_M(:, i_pature1, i_pature2) = attaque_nb_N_pature;
1076. end
1077. end
1078. attaque_nb_N_pature_DM = attaque_nb_N - attaque_nb_N_pature_M;
1079. attaque_nb_N_pature_DM2 = ...
1080.     attaque_nb_N_pature_DM.*attaque_nb_N_pature_DM;
1081. attaque_nb_N_pature_V = mean(mean(attaque_nb_N_pature_DM2,3),2);
1082. attaque_nb_N_pature_ET = sqrt(attaque_nb_N_pature_V);
1083. attaque_nb_N_ET = [attaque_nb_N_global_ET attaque_nb_N_pature_ET' ...

```

```

1084.     attaque_nb_N_race_ET];
1085. % ecart type du nombre d attaques sur zebu
1086. attaque_nb_Z_global_M = ones(2,4,2)*attaque_nb_Z_global;
1087. attaque_nb_Z_global_DM = attaque_nb_Z - attaque_nb_Z_global_M;
1088. attaque_nb_Z_global_DM2 = ...
1089.     attaque_nb_Z_global_DM.*attaque_nb_Z_global_DM;
1090. attaque_nb_Z_global_V = mean(mean(mean(attaque_nb_Z_global_DM2)));
1091. attaque_nb_Z_global_ET = sqrt(attaque_nb_Z_global_V);
1092. attaque_nb_Z_race_M = ones(2,4,2);
1093. for i_race1 = 1:2
1094. for i_race2 = 1:2
1095. attaque_nb_Z_race_M(i_race1, :, i_race2) = attaque_nb_Z_race;
1096. end
1097. end
1098. attaque_nb_Z_race_DM = attaque_nb_Z - attaque_nb_Z_race_M;
1099. attaque_nb_Z_race_DM2 = attaque_nb_Z_race_DM.*attaque_nb_Z_race_DM;
1100. attaque_nb_Z_race_V = mean(mean(attaque_nb_Z_race_DM2,3),1);
1101. attaque_nb_Z_race_ET = sqrt(attaque_nb_Z_race_V);
1102. attaque_nb_Z_pature_M = ones(2,4,2);
1103. for i_pature1 = 1:4
1104. for i_pature2 = 1:2
1105. attaque_nb_Z_pature_M(:, i_pature1, i_pature2) = attaque_nb_Z_pature;
1106. end
1107. end
1108. attaque_nb_Z_pature_DM = attaque_nb_Z - attaque_nb_Z_pature_M;
1109. attaque_nb_Z_pature_DM2 = ...
1110.     attaque_nb_Z_pature_DM.*attaque_nb_Z_pature_DM;
1111. attaque_nb_Z_pature_V = mean(mean(attaque_nb_Z_pature_DM2,3),2);
1112. attaque_nb_Z_pature_ET = sqrt(attaque_nb_Z_pature_V);
1113. attaque_nb_Z_ET = [attaque_nb_Z_global_ET attaque_nb_Z_pature_ET' ...
1114.     attaque_nb_Z_race_ET];
1115. % ecart type du nombre d attaques sur ndapol
1116. attaque_nb_Np_global_M = ones(2,4,2)*attaque_nb_Np_global;
1117. attaque_nb_Np_global_DM = attaque_nb_Np - attaque_nb_Np_global_M;
1118. attaque_nb_Np_global_DM2 = ...
1119.     attaque_nb_Np_global_DM.*attaque_nb_Np_global_DM;
1120. attaque_nb_Np_global_V = mean(mean(mean(attaque_nb_Np_global_DM2)));
1121. attaque_nb_Np_global_ET = sqrt(attaque_nb_Np_global_V);
1122. attaque_nb_Np_race_M = ones(2,4,2);
1123. for i_race1 = 1:2
1124. for i_race2 = 1:2
1125. attaque_nb_Np_race_M(i_race1, :, i_race2) = attaque_nb_Np_race;
1126. end
1127. end
1128. attaque_nb_Np_race_DM = attaque_nb_Np - attaque_nb_Np_race_M;
1129. attaque_nb_Np_race_DM2 = ...
1130.     attaque_nb_Np_race_DM.*attaque_nb_Np_race_DM;
1131. attaque_nb_Np_race_V = mean(mean(attaque_nb_Np_race_DM2,3),1);
1132. attaque_nb_Np_race_ET = sqrt(attaque_nb_Np_race_V);
1133. attaque_nb_Np_pature_M = ones(2,4,2);
1134. for i_pature1 = 1:4
1135. for i_pature2 = 1:2

```

```

1136. attaque_nb_Np_pature_M(:,i_pature1,i_pature2) = attaque_nb_Np_pature ;
1137. end
1138. end
1139. attaque_nb_Np_pature_DM = attaque_nb_Np - attaque_nb_Np_pature_M ;
1140. attaque_nb_Np_pature_DM2 = ...
1141.     attaque_nb_Np_pature_DM.*attaque_nb_Np_pature_DM ;
1142. attaque_nb_Np_pature_V = mean(mean(attaque_nb_Np_pature_DM2,3),2) ;
1143. attaque_nb_Np_pature_ET = sqrt(attaque_nb_Np_pature_V) ;
1144. attaque_nb_Np_ET = [attaque_nb_Np_global_ET attaque_nb_Np_pature_ET' ...
1145.     attaque_nb_Np_race_ET] ;
1146. % ecart type du nombre d attaques sur zepol
1147. attaque_nb_Zp_global_M = ones(2,4,2)*attaque_nb_Zp_global ;
1148. attaque_nb_Zp_global_DM = attaque_nb_Zp - attaque_nb_Zp_global_M ;
1149. attaque_nb_Zp_global_DM2 = ...
1150.     attaque_nb_Zp_global_DM.*attaque_nb_Zp_global_DM ;
1151. attaque_nb_Zp_global_V = mean(mean(mean(attaque_nb_Zp_global_DM2))) ;
1152. attaque_nb_Zp_global_ET = sqrt(attaque_nb_Zp_global_V) ;
1153. attaque_nb_Zp_race_M = ones(2,4,2) ;
1154. for i_race1 = 1:2
1155.     for i_race2 = 1:2
1156.         attaque_nb_Zp_race_M(i_race1, :, i_race2) = attaque_nb_Zp_race ;
1157.     end
1158. end
1159. attaque_nb_Zp_race_DM = attaque_nb_Zp - attaque_nb_Zp_race_M ;
1160. attaque_nb_Zp_race_DM2 = ...
1161.     attaque_nb_Zp_race_DM.*attaque_nb_Zp_race_DM ;
1162. attaque_nb_Zp_race_V = mean(mean(attaque_nb_Zp_race_DM2,3),1) ;
1163. attaque_nb_Zp_race_ET = sqrt(attaque_nb_Zp_race_V) ;
1164. attaque_nb_Zp_pature_M = ones(2,4,2) ;
1165. for i_pature1 = 1:4
1166.     for i_pature2 = 1:2
1167.         attaque_nb_Zp_pature_M(:, i_pature1, i_pature2) = attaque_nb_Zp_pature ;
1168.     end
1169. end
1170. attaque_nb_Zp_pature_DM = attaque_nb_Zp - attaque_nb_Zp_pature_M ;
1171. attaque_nb_Zp_pature_DM2 = ...
1172.     attaque_nb_Zp_pature_DM.*attaque_nb_Zp_pature_DM ;
1173. attaque_nb_Zp_pature_V = mean(mean(attaque_nb_Zp_pature_DM2,3),2) ;
1174. attaque_nb_Zp_pature_ET = sqrt(attaque_nb_Zp_pature_V) ;
1175. attaque_nb_Zp_ET = [attaque_nb_Zp_global_ET attaque_nb_Zp_pature_ET' ...
1176.     attaque_nb_Zp_race_ET] ;
1177. % ecart type du nombre d attaques sur bete hors protocole
1178. attaque_nb_HP_global_M = ones(2,4,2)*attaque_nb_HP_global ;
1179. attaque_nb_HP_global_DM = attaque_nb_HP - attaque_nb_HP_global_M ;
1180. attaque_nb_HP_global_DM2 = ...
1181.     attaque_nb_HP_global_DM.*attaque_nb_HP_global_DM ;
1182. attaque_nb_HP_global_V = mean(mean(mean(attaque_nb_HP_global_DM2))) ;
1183. attaque_nb_HP_global_ET = sqrt(attaque_nb_HP_global_V) ;
1184. attaque_nb_HP_race_M = ones(2,4,2) ;
1185. for i_race1 = 1:2
1186.     for i_race2 = 1:2
1187.         attaque_nb_HP_race_M(i_race1, :, i_race2) = attaque_nb_HP_race ;
1188.     end
1189. end
1190. attaque_nb_HP_race_DM = attaque_nb_HP - attaque_nb_HP_race_M ;
1191. attaque_nb_HP_race_DM2 = ...
1192.     attaque_nb_HP_race_DM.*attaque_nb_HP_race_DM ;
1193. attaque_nb_HP_race_V = mean(mean(attaque_nb_HP_race_DM2,3),1) ;
1194. attaque_nb_HP_race_ET = sqrt(attaque_nb_HP_race_V) ;
1195. attaque_nb_HP_pature_M = ones(2,4,2) ;
1196. for i_pature1 = 1:4
1197.     for i_pature2 = 1:2
1198.         attaque_nb_HP_pature_M(:, i_pature1, i_pature2) = attaque_nb_HP_pature ;
1199.     end
1200. end
1201. attaque_nb_HP_pature_DM = attaque_nb_HP - attaque_nb_HP_pature_M ;
1202. attaque_nb_HP_pature_DM2 = ...
1203.     attaque_nb_HP_pature_DM.*attaque_nb_HP_pature_DM ;
1204. attaque_nb_HP_pature_V = mean(mean(attaque_nb_HP_pature_DM2,3),2) ;
1205. attaque_nb_HP_pature_ET = sqrt(attaque_nb_HP_pature_V) ;
1206. attaque_nb_HP_ET = [attaque_nb_HP_global_ET attaque_nb_HP_pature_ET' ...
1207.     attaque_nb_HP_race_ET] ;
1208. % ANOVA du nombre d attaques sur ndama
1209. attaque_nb_N_AV = [attaque_nb_N(1, :, 1) attaque_nb_N(1, :, 2) ...
1210.     attaque_nb_N(2, :, 1) attaque_nb_N(1, :, 2)] ;
1211. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
1212. f2 = [1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2] ;
1213. attaque_nb_N_pvalue = anovan(attaque_nb_N_AV, {f1, f2}, 'full') ;
1214. attaque_nb_N_pvalue = flipr(attaque_nb_N_pvalue) ;
1215. close all force
1216. % ANOVA du nombre d attaques sur zebu
1217. attaque_nb_Z_AV = [attaque_nb_Z(1, :, 1) attaque_nb_Z(1, :, 2) ...
1218.     attaque_nb_Z(2, :, 1) attaque_nb_Z(1, :, 2)] ;
1219. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
1220. f2 = [1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2] ;
1221. attaque_nb_Z_pvalue = anovan(attaque_nb_Z_AV, {f1, f2}, 'full') ;
1222. attaque_nb_Z_pvalue = flipr(attaque_nb_Z_pvalue) ;
1223. close all force
1224. % ANOVA du nombre d attaques sur ndapol
1225. attaque_nb_Np_AV = [attaque_nb_Np(1, :, 1) attaque_nb_Np(1, :, 2) ...
1226.     attaque_nb_Np(2, :, 1) attaque_nb_Np(1, :, 2)] ;
1227. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
1228. f2 = [1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2] ;
1229. attaque_nb_Np_pvalue = anovan(attaque_nb_Np_AV, {f1, f2}, 'full') ;
1230. attaque_nb_Np_pvalue = flipr(attaque_nb_Np_pvalue) ;
1231. close all force
1232. % ANOVA du nombre d attaques sur zepol
1233. attaque_nb_Zp_AV = [attaque_nb_Zp(1, :, 1) attaque_nb_Zp(1, :, 2) ...
1234.     attaque_nb_Zp(2, :, 1) attaque_nb_Zp(1, :, 2)] ;
1235. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
1236. f2 = [1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2] ;
1237. attaque_nb_Zp_pvalue = anovan(attaque_nb_Zp_AV, {f1, f2}, 'full') ;
1238. attaque_nb_Zp_pvalue = flipr(attaque_nb_Zp_pvalue) ;
1239. close all force

```

```

1240. % ANOVA du nombre d attaques sur bete hors protocole
1241. attaque_nb_HP_AV = [attaque_nb_HP(1,;,1) attaque_nb_HP(1,;,2) ...
1242.     attaque_nb_HP(2,;,1) attaque_nb_HP(1,;,2)];
1243. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
1244. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
1245. attaque_nb_HP_pvalue = anovan(attaque_nb_HP_AV,{f1,f2},'full');
1246. attaque_nb_HP_pvalue = fliplr(attaque_nb_HP_pvalue');
1247. close all force
1248. % fusion du nombre d attaques par cible
1249. attaque_cible_stat = [
1250.     attaque_nb_N_moy attaque_nb_N_ET attaque_nb_N_pvalue
1251.     attaque_nb_Z_moy attaque_nb_Z_ET attaque_nb_Z_pvalue
1252.     attaque_nb_Np_moy attaque_nb_Np_ET attaque_nb_Np_pvalue
1253.     attaque_nb_Zp_moy attaque_nb_Zp_ET attaque_nb_Zp_pvalue
1254.     attaque_nb_HP_moy attaque_nb_HP_ET attaque_nb_HP_pvalue];
1255. clearvars -except comport attaque_nb attaque_nb_stat ...
1256.     attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
1257.     ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
1258.     flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
1259.     lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
1260.     meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
1261.     allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
1262. %% nombre de broutages de ligneux 1
1263. % comptage par bete
1264. ligneux_nb_1 = zeros(2,4,2);
1265. i=1;
1266. while i <= length(comport)
1267.     for x = 1:2
1268.         for y = 1:4
1269.             for z = 1:2
1270.                 if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
1271.                     if comport(i,8)==1
1272.                         ligneux_nb_1(x,y,z)=ligneux_nb_1(x,y,z)+(~isnan(comport(i,8)));
1273.                     end
1274.                 end
1275.             end
1276.         end
1277.     end
1278.     i=i+1;
1279. end
1280. % moyenne
1281. ligneux_nb_1_global = mean(mean(mean(ligneux_nb_1)));
1282. ligneux_nb_1_race = mean(mean(ligneux_nb_1,3),1);
1283. ligneux_nb_1_pature = mean(mean(ligneux_nb_1,3),2);
1284. ligneux_nb_1_moy = [ligneux_nb_1_global ligneux_nb_1_pature' ...
1285.     ligneux_nb_1_race];
1286. % nombre de broutages de ligneux 2
1287. % comptage par bete
1288. ligneux_nb_2 = zeros(2,4,2);
1289. i=1;
1290. while i <= length(comport)
1291.     for x = 1:2

```

```

1292.         for y = 1:4
1293.             for z = 1:2
1294.                 if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
1295.                     if comport(i,8)==2
1296.                         ligneux_nb_2(x,y,z)=ligneux_nb_2(x,y,z)+(~isnan(comport(i,8)));
1297.                     end
1298.                 end
1299.             end
1300.         end
1301.     end
1302.     i=i+1;
1303. end
1304. % moyenne
1305. ligneux_nb_2_global = mean(mean(mean(ligneux_nb_2)));
1306. ligneux_nb_2_race = mean(mean(ligneux_nb_2,3),1);
1307. ligneux_nb_2_pature = mean(mean(ligneux_nb_2,3),2);
1308. ligneux_nb_2_moy = [ligneux_nb_2_global ligneux_nb_2_pature' ...
1309.     ligneux_nb_2_race];
1310. % nombre de broutages de ligneux 3
1311. % comptage par bete
1312. ligneux_nb_3 = zeros(2,4,2);
1313. i=1;
1314. while i <= length(comport)
1315.     for x = 1:2
1316.         for y = 1:4
1317.             for z = 1:2
1318.                 if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
1319.                     if comport(i,8)==3
1320.                         ligneux_nb_3(x,y,z)=ligneux_nb_3(x,y,z)+(~isnan(comport(i,8)));
1321.                     end
1322.                 end
1323.             end
1324.         end
1325.     end
1326.     i=i+1;
1327. end
1328. % moyenne
1329. ligneux_nb_3_global = mean(mean(mean(ligneux_nb_3)));
1330. ligneux_nb_3_race = mean(mean(ligneux_nb_3,3),1);
1331. ligneux_nb_3_pature = mean(mean(ligneux_nb_3,3),2);
1332. ligneux_nb_3_moy = [ligneux_nb_3_global ligneux_nb_3_pature' ...
1333.     ligneux_nb_3_race];
1334. % nombre de broutages de ligneux 4
1335. % comptage par bete
1336. ligneux_nb_4 = zeros(2,4,2);
1337. i=1;
1338. while i <= length(comport)
1339.     for x = 1:2
1340.         for y = 1:4
1341.             for z = 1:2
1342.                 if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
1343.                     if comport(i,8)==4

```

```

1344. ligneux_nb_4(x,y,z)=ligneux_nb_4(x,y,z)+(~isnan(comport(i,8)));
1345. end
1346. end
1347. end
1348. end
1349. end
1350. i=i+1 ;
1351. end
1352. % moyenne
1353. ligneux_nb_4_global = mean(mean(mean(ligneux_nb_4)));
1354. ligneux_nb_4_race = mean(mean(ligneux_nb_4,3),1);
1355. ligneux_nb_4_pature = mean(mean(ligneux_nb_4,3),2);
1356. ligneux_nb_4_moy = [ligneux_nb_4_global ligneux_nb_4_pature' ...
1357.     ligneux_nb_4_race];
1358. % nombre de broutages de ligneux autre
1359. % comptage par bete
1360. ligneux_nb_0 = zeros(2,4,2) ;
1361. i=1 ;
1362. while i <= length(comport)
1363.     for x = 1:2
1364.         for y = 1:4
1365.             for z = 1:2
1366.                 if comport(i,1)==x && comport(i,2)==y && comport(i,3)==z
1367.                     if comport(i,8)==0
1368.                         ligneux_nb_0(x,y,z)=ligneux_nb_0(x,y,z)+(~isnan(comport(i,8)));
1369.                     end
1370.                 end
1371.             end
1372.         end
1373.     end
1374.     i=i+1 ;
1375. end
1376. % moyenne
1377. ligneux_nb_0_global = mean(mean(mean(ligneux_nb_0)));
1378. ligneux_nb_0_race = mean(mean(ligneux_nb_0,3),1);
1379. ligneux_nb_0_pature = mean(mean(ligneux_nb_0,3),2);
1380. ligneux_nb_0_moy = [ligneux_nb_0_global ligneux_nb_0_pature' ...
1381.     ligneux_nb_0_race];
1382. % ecart type du nombre de broutages de ligneux 1
1383. ligneux_nb_1_global_M = ones(2,4,2)*ligneux_nb_1_global ;
1384. ligneux_nb_1_global_DM = ligneux_nb_1 - ligneux_nb_1_global_M ;
1385. ligneux_nb_1_global_DM2 = ...
1386.     ligneux_nb_1_global_DM.*ligneux_nb_1_global_DM ;
1387. ligneux_nb_1_global_V = mean(mean(mean(ligneux_nb_1_global_DM2)));
1388. ligneux_nb_1_global_ET = sqrt(ligneux_nb_1_global_V) ;
1389. ligneux_nb_1_race_M = ones(2,4,2) ;
1390. for i_race1 = 1:2
1391.     for i_race2 = 1:2
1392.         ligneux_nb_1_race_M(i_race1,;,i_race2) = ligneux_nb_1_race ;
1393.     end
1394. end
1395. ligneux_nb_1_race_DM = ligneux_nb_1 - ligneux_nb_1_race_M ;

```

```

1396. ligneux_nb_1_race_DM2 = ligneux_nb_1_race_DM.*ligneux_nb_1_race_DM ;
1397. ligneux_nb_1_race_V = mean(mean(ligneux_nb_1_race_DM2,3),1) ;
1398. ligneux_nb_1_race_ET = sqrt(ligneux_nb_1_race_V) ;
1399. ligneux_nb_1_pature_M = ones(2,4,2) ;
1400. for i_pature1 = 1:4
1401.     for i_pature2 = 1:2
1402.         ligneux_nb_1_pature_M(:,i_pature1,i_pature2) = ligneux_nb_1_pature ;
1403.     end
1404. end
1405. ligneux_nb_1_pature_DM = ligneux_nb_1 - ligneux_nb_1_pature_M ;
1406. ligneux_nb_1_pature_DM2 = ...
1407.     ligneux_nb_1_pature_DM.*ligneux_nb_1_pature_DM ;
1408. ligneux_nb_1_pature_V = mean(mean(ligneux_nb_1_pature_DM2,3),2) ;
1409. ligneux_nb_1_pature_ET = sqrt(ligneux_nb_1_pature_V) ;
1410. ligneux_nb_1_ET = [ligneux_nb_1_global_ET ligneux_nb_1_pature_ET' ...
1411.     ligneux_nb_1_race_ET] ;
1412. % ecart type du nombre de broutages de ligneux 2
1413. ligneux_nb_2_global_M = ones(2,4,2)*ligneux_nb_2_global ;
1414. ligneux_nb_2_global_DM = ligneux_nb_2 - ligneux_nb_2_global_M ;
1415. ligneux_nb_2_global_DM2 = ...
1416.     ligneux_nb_2_global_DM.*ligneux_nb_2_global_DM ;
1417. ligneux_nb_2_global_V = mean(mean(mean(ligneux_nb_2_global_DM2)));
1418. ligneux_nb_2_global_ET = sqrt(ligneux_nb_2_global_V) ;
1419. ligneux_nb_2_race_M = ones(2,4,2) ;
1420. for i_race1 = 1:2
1421.     for i_race2 = 1:2
1422.         ligneux_nb_2_race_M(i_race1,;,i_race2) = ligneux_nb_2_race ;
1423.     end
1424. end
1425. ligneux_nb_2_race_DM = ligneux_nb_2 - ligneux_nb_2_race_M ;
1426. ligneux_nb_2_race_DM2 = ligneux_nb_2_race_DM.*ligneux_nb_2_race_DM ;
1427. ligneux_nb_2_race_V = mean(mean(ligneux_nb_2_race_DM2,3),1) ;
1428. ligneux_nb_2_race_ET = sqrt(ligneux_nb_2_race_V) ;
1429. ligneux_nb_2_pature_M = ones(2,4,2) ;
1430. for i_pature1 = 1:4
1431.     for i_pature2 = 1:2
1432.         ligneux_nb_2_pature_M(:,i_pature1,i_pature2) = ligneux_nb_2_pature ;
1433.     end
1434. end
1435. ligneux_nb_2_pature_DM = ligneux_nb_2 - ligneux_nb_2_pature_M ;
1436. ligneux_nb_2_pature_DM2 = ...
1437.     ligneux_nb_2_pature_DM.*ligneux_nb_2_pature_DM ;
1438. ligneux_nb_2_pature_V = mean(mean(ligneux_nb_2_pature_DM2,3),2) ;
1439. ligneux_nb_2_pature_ET = sqrt(ligneux_nb_2_pature_V) ;
1440. ligneux_nb_2_ET = [ligneux_nb_2_global_ET ligneux_nb_2_pature_ET' ...
1441.     ligneux_nb_2_race_ET] ;
1442. % ecart type du nombre de broutages de ligneux 3
1443. ligneux_nb_3_global_M = ones(2,4,2)*ligneux_nb_3_global ;
1444. ligneux_nb_3_global_DM = ligneux_nb_3 - ligneux_nb_3_global_M ;
1445. ligneux_nb_3_global_DM2 = ...
1446.     ligneux_nb_3_global_DM.*ligneux_nb_3_global_DM ;
1447. ligneux_nb_3_global_V = mean(mean(mean(ligneux_nb_3_global_DM2)));

```

```

1448. ligneux_nb_3_global_ET = sqrt(ligneux_nb_3_global_V);
1449. ligneux_nb_3_race_M = ones(2,4,2);
1450. for i_race1 = 1:2
1451. for i_race2 = 1:2
1452. ligneux_nb_3_race_M(i_race1, :, i_race2) = ligneux_nb_3_race;
1453. end
1454. end
1455. ligneux_nb_3_race_DM = ligneux_nb_3 - ligneux_nb_3_race_M;
1456. ligneux_nb_3_race_DM2 = ligneux_nb_3_race_DM.*ligneux_nb_3_race_DM;
1457. ligneux_nb_3_race_V = mean(mean(ligneux_nb_3_race_DM2,3),1);
1458. ligneux_nb_3_race_ET = sqrt(ligneux_nb_3_race_V);
1459. ligneux_nb_3_pature_M = ones(2,4,2);
1460. for i_pature1 = 1:4
1461. for i_pature2 = 1:2
1462. ligneux_nb_3_pature_M(:, i_pature1, i_pature2) = ligneux_nb_3_pature;
1463. end
1464. end
1465. ligneux_nb_3_pature_DM = ligneux_nb_3 - ligneux_nb_3_pature_M;
1466. ligneux_nb_3_pature_DM2 = ...
1467.     ligneux_nb_3_pature_DM.*ligneux_nb_3_pature_DM;
1468. ligneux_nb_3_pature_V = mean(mean(ligneux_nb_3_pature_DM2,3),2);
1469. ligneux_nb_3_pature_ET = sqrt(ligneux_nb_3_pature_V);
1470. ligneux_nb_3_ET = [ligneux_nb_3_global_ET ligneux_nb_3_pature_ET' ...
1471.     ligneux_nb_3_race_ET];
1472. % ecart type du nombre de broutages de ligneux 4
1473. ligneux_nb_4_global_M = ones(2,4,2)*ligneux_nb_4_global;
1474. ligneux_nb_4_global_DM = ligneux_nb_4 - ligneux_nb_4_global_M;
1475. ligneux_nb_4_global_DM2 = ...
1476.     ligneux_nb_4_global_DM.*ligneux_nb_4_global_DM;
1477. ligneux_nb_4_global_V = mean(mean(mean(ligneux_nb_4_global_DM2))) ;
1478. ligneux_nb_4_global_ET = sqrt(ligneux_nb_4_global_V);
1479. ligneux_nb_4_race_M = ones(2,4,2);
1480. for i_race1 = 1:2
1481. for i_race2 = 1:2
1482. ligneux_nb_4_race_M(i_race1, :, i_race2) = ligneux_nb_4_race;
1483. end
1484. end
1485. ligneux_nb_4_race_DM = ligneux_nb_4 - ligneux_nb_4_race_M;
1486. ligneux_nb_4_race_DM2 = ligneux_nb_4_race_DM.*ligneux_nb_4_race_DM;
1487. ligneux_nb_4_race_V = mean(mean(ligneux_nb_4_race_DM2,3),1);
1488. ligneux_nb_4_race_ET = sqrt(ligneux_nb_4_race_V);
1489. ligneux_nb_4_pature_M = ones(2,4,2);
1490. for i_pature1 = 1:4
1491. for i_pature2 = 1:2
1492. ligneux_nb_4_pature_M(:, i_pature1, i_pature2) = ligneux_nb_4_pature;
1493. end
1494. end
1495. ligneux_nb_4_pature_DM = ligneux_nb_4 - ligneux_nb_4_pature_M;
1496. ligneux_nb_4_pature_DM2 = ...
1497.     ligneux_nb_4_pature_DM.*ligneux_nb_4_pature_DM;
1498. ligneux_nb_4_pature_V = mean(mean(ligneux_nb_4_pature_DM2,3),2);
1499. ligneux_nb_4_pature_ET = sqrt(ligneux_nb_4_pature_V);

1500. ligneux_nb_4_ET = [ligneux_nb_4_global_ET ligneux_nb_4_pature_ET' ...
1501.     ligneux_nb_4_race_ET];
1502. % ecart type du nombre de broutages de ligneux autre
1503. ligneux_nb_0_global_M = ones(2,4,2)*ligneux_nb_0_global;
1504. ligneux_nb_0_global_DM = ligneux_nb_0 - ligneux_nb_0_global_M;
1505. ligneux_nb_0_global_DM2 = ...
1506.     ligneux_nb_0_global_DM.*ligneux_nb_0_global_DM;
1507. ligneux_nb_0_global_V = mean(mean(mean(ligneux_nb_0_global_DM2))) ;
1508. ligneux_nb_0_global_ET = sqrt(ligneux_nb_0_global_V);
1509. ligneux_nb_0_race_M = ones(2,4,2);
1510. for i_race1 = 1:2
1511. for i_race2 = 1:2
1512. ligneux_nb_0_race_M(i_race1, :, i_race2) = ligneux_nb_0_race;
1513. end
1514. end
1515. ligneux_nb_0_race_DM = ligneux_nb_0 - ligneux_nb_0_race_M;
1516. ligneux_nb_0_race_DM2 = ligneux_nb_0_race_DM.*ligneux_nb_0_race_DM;
1517. ligneux_nb_0_race_V = mean(mean(ligneux_nb_0_race_DM2,3),1);
1518. ligneux_nb_0_race_ET = sqrt(ligneux_nb_0_race_V);
1519. ligneux_nb_0_pature_M = ones(2,4,2);
1520. for i_pature1 = 1:4
1521. for i_pature2 = 1:2
1522. ligneux_nb_0_pature_M(:, i_pature1, i_pature2) = ligneux_nb_0_pature;
1523. end
1524. end
1525. ligneux_nb_0_pature_DM = ligneux_nb_0 - ligneux_nb_0_pature_M;
1526. ligneux_nb_0_pature_DM2 = ...
1527.     ligneux_nb_0_pature_DM.*ligneux_nb_0_pature_DM;
1528. ligneux_nb_0_pature_V = mean(mean(ligneux_nb_0_pature_DM2,3),2);
1529. ligneux_nb_0_pature_ET = sqrt(ligneux_nb_0_pature_V);
1530. ligneux_nb_0_ET = [ligneux_nb_0_global_ET ligneux_nb_0_pature_ET' ...
1531.     ligneux_nb_0_race_ET];
1532. % ANOVA du nombre de broutages de ligneux 1
1533. ligneux_nb_1_AV = [ligneux_nb_1(1, :, 1) ligneux_nb_1(1, :, 2) ...
1534.     ligneux_nb_1(2, :, 1) ligneux_nb_1(1, :, 2)];
1535. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
1536. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2];
1537. ligneux_nb_1_pvalue = anovan(ligneux_nb_1_AV, {f1, f2}, 'full');
1538. ligneux_nb_1_pvalue = fliplr(ligneux_nb_1_pvalue);
1539. close all force
1540. % ANOVA du nombre de broutages de ligneux 2
1541. ligneux_nb_2_AV = [ligneux_nb_2(1, :, 1) ligneux_nb_2(1, :, 2) ...
1542.     ligneux_nb_2(2, :, 1) ligneux_nb_2(1, :, 2)];
1543. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
1544. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2];
1545. ligneux_nb_2_pvalue = anovan(ligneux_nb_2_AV, {f1, f2}, 'full');
1546. ligneux_nb_2_pvalue = fliplr(ligneux_nb_2_pvalue);
1547. close all force
1548. % ANOVA du nombre de broutages de ligneux 3
1549. ligneux_nb_3_AV = [ligneux_nb_3(1, :, 1) ligneux_nb_3(1, :, 2) ...
1550.     ligneux_nb_3(2, :, 1) ligneux_nb_3(1, :, 2)];
1551. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];

```

```

1552. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2];
1553. ligneux_nb_3_pvalue = anovan(ligneux_nb_3_AV,{f1,f2},'full');
1554. ligneux_nb_3_pvalue = fliplr(ligneux_nb_3_pvalue);
1555. close all force
1556. % ANOVA du nombre de broutages de ligneux 4
1557. ligneux_nb_4_AV = [ligneux_nb_4(1,,:), ligneux_nb_4(1,,:),2] ...
1558.     ligneux_nb_4(2,,:), ligneux_nb_4(1,,:),2];
1559. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
1560. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
1561. ligneux_nb_4_pvalue = anovan(ligneux_nb_4_AV,{f1,f2},'full');
1562. ligneux_nb_4_pvalue = fliplr(ligneux_nb_4_pvalue);
1563. close all force
1564. % ANOVA du nombre de broutages de ligneux autre
1565. ligneux_nb_0_AV = [ligneux_nb_0(1,,:), ligneux_nb_0(1,,:),2] ...
1566.     ligneux_nb_0(2,,:), ligneux_nb_0(1,,:),2];
1567. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
1568. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
1569. ligneux_nb_0_pvalue = anovan(ligneux_nb_0_AV,{f1,f2},'full');
1570. ligneux_nb_0_pvalue = fliplr(ligneux_nb_0_pvalue);
1571. close all force
1572. % fusion du nombre de broutages de ligneux par type
1573. ligneux_cible_stat = [
1574. ligneux_nb_1_moy ligneux_nb_1_ET ligneux_nb_1_pvalue
1575. ligneux_nb_2_moy ligneux_nb_2_ET ligneux_nb_2_pvalue
1576. ligneux_nb_3_moy ligneux_nb_3_ET ligneux_nb_3_pvalue
1577. ligneux_nb_4_moy ligneux_nb_4_ET ligneux_nb_4_pvalue
1578. ligneux_nb_0_moy ligneux_nb_0_ET ligneux_nb_0_pvalue];
1579. clearvars -except comport attaque_nb attaque_nb_stat ...
1580.     attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
1581.     ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
1582.     flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
1583.     lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
1584.     meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
1585.     allonge_val allonge_val_stat attaque_cible_stat ligneux_cible_stat
1586. %% compilation en une seule matrice
1587. stat_comport_final = [attaque_nb_stat ; attaque_cible_stat ; ...
1588.     attaque_val_stat ; ligneux_nb_stat ; ligneux_cible_stat ; ...
1589.     ligneux_val_stat ; ligneux_val2_stat ; flehmen_nb_stat ; ...
1590.     lechage_nb_stat ; lechage_val_stat ; meuh_nb_stat ; ...
1591.     meuh_val_stat ; allonge_nb_stat ; allonge_val_stat];
1592. C = stat_comport_final;
1593. stat_comport_final = [C(:,1) C(:,8) C(:,15) C(:,2) C(:,9) C(:,3) ...
1594.     C(:,10) C(:,16) C(:,4) C(:,11) C(:,5) C(:,12) C(:,6) C(:,13) ...
1595.     C(:,7) C(:,14) C(:,17)];
1596. %% ecriture fichier
1597. xlsxwrite('stat_comport3',stat_comport_final)
1598. %% ecriture lignes
1599. S1='nombre d attaques';
1600. S2='nombre d attaques de ndama';
1601. S3='nombre d attaques de zebu';
1602. S4='nombre d attaques de ndapol';
1603. S5='nombre d attaques de zepol';
1604. S6='nombre d attaques de betes hors protocole';
1605. S7='intensite des attaques (1-3)';
1606. S8='nombre de broutages';
1607. S9='nombre de broutages de ligneux de type 1';
1608. S10='nombre de broutages de ligneux de type 2';
1609. S11='nombre de broutages de ligneux de type 3';
1610. S12='nombre de broutages de ligneux de type 4';
1611. S13='nombre de broutages d autres ligneux';
1612. S14='nombre total de bouchees lors des broutages';
1613. S15='hauteur des broutages (cm)';
1614. S16='nombre de flehmen';
1615. S17='nombre de lechages';
1616. S18='nombre total de coups de langue lors des lechages';
1617. S19='nombre de meuglements';
1618. S20='nombre total de cris lors des meuglements';
1619. S21='nombre de periodes passees allonge';
1620. S22='temps passe allonge (min)';
1621. lignes={S1;S2;S3;S4;S5;S6;S7;S8;S9;S10; ...
1622.     S11;S12;S13;S14;S15;S16;S17;S18;S19;S20;S21;S22};
1623. xlsxwrite('lignes_stat_comport',lignes)
1624. clearvars -except comport attaque_nb attaque_nb_stat ...
1625.     attaque_val attaque_val_stat ligneux_nb ligneux_nb_stat ...
1626.     ligneux_val ligneux_val_stat ligneux_val2 ligneux_val2_stat ...
1627.     flehmen_nb flehmen_nb_stat lechage_nb lechage_nb_stat ...
1628.     lechage_val lechage_val_stat meuh_nb meuh_nb_stat ...
1629.     meuh_val meuh_val_stat allonge_nb allonge_nb_stat ...
1630.     allonge_val allonge_val_stat attaque_cible_stat ...
1631.     ligneux_cible_stat stat_comport_final

```

Annexe 16 : script du traitement statistique

des observations directes discontinues

```
1632. %%% observations discontinues
1633. %% NB : distance en metre et temps en heures
1634. %% lecture fichier
1635. groupe = xlsread('compil_groupe.xlsx') ;
1636. groupe = [groupe(:,1:6) groupe(:,8:9) groupe(:,11:12) ...
1637.           groupe(:,14:15) groupe(:,17:18) groupe(:,7) ...
1638.           groupe(:,10) groupe(:,13) groupe(:,16) groupe(:,19:29)] ;
1639. %% identification de la bete le plus souvent la plus proche (BF)
1640. % comptage
1641. % NB : lignes = bete
1642. % NB : colonnes = bete la plus proche (HP + race X individu)
1643. BF1 = zeros(16,9) ;
1644. i=1 ;
1645. while i <= length(groupe)
1646.     for x=1:2
1647.         for y=1:4
1648.             for z=1:2
1649.                 if groupe(i,1)==x && groupe(i,2)==y && groupe(i,3)==z
1650.                     for c=1:9
1651.                         BF1(((x-1)*8+(y-1)*2+z),c) = BF1(((x-1)*8+(y-1)*2+z),c) + ...
1652.                             groupe(i,20+c) ;
1653.                     end
1654.                 end
1655.             end
1656.         end
1657.     end
1658.     i=i+1 ;
1659. end
1660. % max de comptage (NaN ou 1)
1661. % NB : lignes = bete
1662. % NB : colonnes = bete la plus proche (HP + race X individu)
1663. BF2 = zeros(16,9) ;
1664. for l=1:16
1665.     for c=1:9
1666.         if BF1(l,c)==max(BF1(l,:))
1667.             BF2(l,c)=1 ;
1668.         else
1669.             BF2(l,c)=NaN ;
1670.         end
1671.     end
1672. end
1673. % choix BF si egalite -> plus petite distance
1674. for l=1:16
1675.     if nansum(BF2(l,:)) > 1
1676.         for c=1:9
1677.             if BF2(l,c) == 1
1678.                 BF2(l,c) = 0 ;
1679.
1680.                 i=1 ;
1681.                 while i <= length(groupe)
1682.                     if i>((l-1)*72) && i<=(l*72) && groupe(i,20+c)==1
1683.                         BF2(l,c) = BF2(l,c) + groupe(i,20) ;
1684.                     end
1685.                     i=i+1 ;
1686.                 end
1687.             end
1688.         end
1689.     for c=1:9
1690.         if BF2(l,c) == min(BF2(l,:))
1691.             BF2(l,c)=1 ;
1692.         else
1693.             BF2(l,c)=NaN ;
1694.         end
1695.     end
1696. end
1697. % remplacement NaN par 0 pour la lisibilite
1698. for l=1:16
1699.     for c=1:9
1700.         if isnan(BF2(l,c))==1
1701.             BF2(l,c)=0 ;
1702.         end
1703.     end
1704. end
1705. % transformation en une nouvelle matrice
1706. % NB : lignes = bete
1707. % NB : colonne = pature + race + individu + race BF + individu BF
1708. BF3 = zeros(16,5) ;
1709. BF3(1:8,1) = repmat(1,8,1) ;
1710. BF3(9:16,1) = repmat(2,8,1) ;
1711. BF3(:,2) = repmat([1;1;2;2;3;3;4;4],2,1) ;
1712. BF3(:,3) = repmat([1;2],8,1) ;
1713. for l=1:16
1714.     if BF2(l,1)==1
1715.         BF3(l,4)=0 ;
1716.         BF3(l,5)=0 ;
1717.     end
1718.     if BF2(l,2)==1 || BF2(l,3)==1
1719.         BF3(l,4)=1 ;
1720.     elseif BF2(l,4)==1 || BF2(l,5)==1
1721.         BF3(l,4)=2 ;
1722.     elseif BF2(l,6)==1 || BF2(l,7)==1
1723.         BF3(l,4)=3 ;
1724.     elseif BF2(l,8)==1 || BF2(l,9)==1
1725.         BF3(l,4)=4 ;
1726.     end
1727.     if BF2(l,2)==1 || BF2(l,4)==1 || BF2(l,6)==1 || BF2(l,8)==1
1728.         BF3(l,5)=1 ;
1729.     elseif BF2(l,3)==1 || BF2(l,5)==1 || BF2(l,7)==1 || BF2(l,9)==1
1730.         BF3(l,5)=2 ;
```

```

1731. end
1732. end
1733. clearvars -except groupe BF2 BF3
1734. %% nombre de betes presentes
1735. % moyenne par bete
1736. n_pres = zeros(2,4,2,2);
1737. i=1;
1738. while i <= length(groupe)
1739. for x = 1:2
1740. for y = 1:4
1741. for z = 1:2
1742. if groupe(i,1)==x && groupe(i,2)==y && groupe(i,3)==z
1743. n_pres(x,y,z,1) = n_pres(x,y,z,1) + sum(groupe(i,6:14));
1744. n_pres(x,y,z,2) = n_pres(x,y,z,2) + 1;
1745. end
1746. end
1747. end
1748. end
1749. i=i+1;
1750. end
1751. n_pres(:,:,:,1) = n_pres(:,:,:,1)/n_pres(:,:,:,2);
1752. n_pres = n_pres(:,:,:,1);
1753. % moyenne
1754. n_pres_global = nanmean(nanmean(nanmean(n_pres)));
1755. n_pres_race = nanmean(nanmean(n_pres,3),1);
1756. n_pres_pature = nanmean(nanmean(n_pres,3),2);
1757. n_pres_moy = [n_pres_global n_pres_pature' n_pres_race];
1758. % ecart type
1759. n_pres_global_M = ones(2,4,2)*n_pres_global;
1760. n_pres_global_DM = n_pres - n_pres_global_M;
1761. n_pres_global_DM2 = n_pres_global_DM.*n_pres_global_DM;
1762. n_pres_global_V = nanmean(nanmean(nanmean(n_pres_global_DM2)));
1763. n_pres_global_ET = sqrt(n_pres_global_V);
1764. n_pres_race_M = ones(2,4,2);
1765. for i_race1 = 1:2
1766. for i_race2 = 1:2
1767. n_pres_race_M(i_race1, :, i_race2) = n_pres_race;
1768. end
1769. end
1770. n_pres_race_DM = n_pres - n_pres_race_M;
1771. n_pres_race_DM2 = n_pres_race_DM.*n_pres_race_DM;
1772. n_pres_race_V = nanmean(nanmean(n_pres_race_DM2,3),1);
1773. n_pres_race_ET = sqrt(n_pres_race_V);
1774. n_pres_pature_M = ones(2,4,2);
1775. for i_pature1 = 1:4
1776. for i_pature2 = 1:2
1777. n_pres_pature_M(:, i_pature1, i_pature2) = n_pres_pature;
1778. end
1779. end
1780. n_pres_pature_DM = n_pres - n_pres_pature_M;
1781. n_pres_pature_DM2 = n_pres_pature_DM.*n_pres_pature_DM;
1782. n_pres_pature_V = nanmean(nanmean(n_pres_pature_DM2,3),2);

```

```

1783. n_pres_pature_ET = sqrt(n_pres_pature_V);
1784. n_pres_ET = [n_pres_global_ET n_pres_pature_ET' n_pres_race_ET];
1785. % ANOVA
1786. n_pres_AV = [n_pres(1, :, 1) n_pres(1, :, 2) n_pres(2, :, 1) n_pres(1, :, 2)];
1787. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
1788. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
1789. n_pres_pvalue = anovan(n_pres_AV, {f1, f2}, 'full');
1790. n_pres_pvalue = flipr(n_pres_pvalue);
1791. close all force
1792. % fusion
1793. n_pres_stat = [n_pres_moy n_pres_ET n_pres_pvalue];
1794. clearvars -except groupe BF2 BF3 n_pres_stat
1795. %% distance a laquelle se trouve la bete la plus proche
1796. % moyenne par bete
1797. dist = zeros(2,4,2,2);
1798. i=1;
1799. while i <= length(groupe)
1800. for x = 1:2
1801. for y = 1:4
1802. for z = 1:2
1803. if isnan(groupe(i,20))==0
1804. if groupe(i,1)==x && groupe(i,2)==y && groupe(i,3)==z
1805. dist(x,y,z,1) = dist(x,y,z,1) + groupe(i,20);
1806. dist(x,y,z,2) = dist(x,y,z,2) + 1;
1807. end
1808. end
1809. end
1810. end
1811. end
1812. i=i+1;
1813. end
1814. dist(:,:,:,1) = dist(:,:,:,1)/dist(:,:,:,2);
1815. dist = dist(:,:,:,1);
1816. % moyenne
1817. dist_global = nanmean(nanmean(nanmean(dist)));
1818. dist_race = nanmean(nanmean(dist,3),1);
1819. dist_pature = nanmean(nanmean(dist,3),2);
1820. dist_moy = [dist_global dist_pature' dist_race];
1821. % ecart type
1822. dist_global_M = ones(2,4,2)*dist_global;
1823. dist_global_DM = dist - dist_global_M;
1824. dist_global_DM2 = dist_global_DM.*dist_global_DM;
1825. dist_global_V = nanmean(nanmean(nanmean(dist_global_DM2)));
1826. dist_global_ET = sqrt(dist_global_V);
1827. dist_race_M = ones(2,4,2);
1828. for i_race1 = 1:2
1829. for i_race2 = 1:2
1830. dist_race_M(i_race1, :, i_race2) = dist_race;
1831. end
1832. end
1833. dist_race_DM = dist - dist_race_M;
1834. dist_race_DM2 = dist_race_DM.*dist_race_DM;

```

```

1835. dist_race_V = nanmean(nanmean(dist_race_DM2,3),1);
1836. dist_race_ET = sqrt(dist_race_V);
1837. dist_pature_M = ones(2,4,2);
1838. for i_pature1 = 1:4
1839. for i_pature2 = 1:2
1840. dist_pature_M(:,i_pature1,i_pature2) = dist_pature;
1841. end
1842. end
1843. dist_pature_DM = dist - dist_pature_M;
1844. dist_pature_DM2 = dist_pature_DM.*dist_pature_DM;
1845. dist_pature_V = nanmean(nanmean(dist_pature_DM2,3),2);
1846. dist_pature_ET = sqrt(dist_pature_V);
1847. dist_ET = [dist_global_ET dist_pature_ET' dist_race_ET];
1848. % ANOVA
1849. dist_AV = [dist(1, :, 1) dist(1, :, 2) dist(2, :, 1) dist(1, :, 2)];
1850. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
1851. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
1852. dist_pvalue = anovan(dist_AV, {f1, f2}, 'full');
1853. dist_pvalue = fliplr(dist_pvalue);
1854. close all force
1855. % fusion
1856. dist_stat = [dist_moy dist_ET dist_pvalue];
1857. clearvars -except groupe BF2 BF3 n_pres_stat dist_stat
1858. %% temps passe seul
1859. % somme par bete
1860. t_seul = zeros(2,4,2);
1861. i=1;
1862. while i <= length(groupe)
1863. for x = 1:2
1864. for y = 1:4
1865. for z = 1:2
1866. if groupe(i,1)==x && groupe(i,2)==y && groupe(i,3)==z
1867. t_seul(x,y,z) = t_seul(x,y,z) + isnan(groupe(i,20));
1868. end
1869. end
1870. end
1871. end
1872. i=i+1;
1873. end
1874. t_seul = t_seul./6;
1875. % moyenne
1876. t_seul_global = nanmean(nanmean(nanmean(t_seul)));
1877. t_seul_race = nanmean(nanmean(t_seul,3),1);
1878. t_seul_pature = nanmean(nanmean(t_seul,3),2);
1879. t_seul_moy = [t_seul_global t_seul_pature' t_seul_race];
1880. % ecart type
1881. t_seul_global_M = ones(2,4,2)*t_seul_global;
1882. t_seul_global_DM = t_seul - t_seul_global_M;
1883. t_seul_global_DM2 = t_seul_global_DM.*t_seul_global_DM;
1884. t_seul_global_V = nanmean(nanmean(nanmean(t_seul_global_DM2)));
1885. t_seul_global_ET = sqrt(t_seul_global_V);
1886. t_seul_race_M = ones(2,4,2);

```

```

1887. for i_race1 = 1:2
1888. for i_race2 = 1:2
1889. t_seul_race_M(i_race1, :, i_race2) = t_seul_race;
1890. end
1891. end
1892. t_seul_race_DM = t_seul - t_seul_race_M;
1893. t_seul_race_DM2 = t_seul_race_DM.*t_seul_race_DM;
1894. t_seul_race_V = nanmean(nanmean(t_seul_race_DM2,3),1);
1895. t_seul_race_ET = sqrt(t_seul_race_V);
1896. t_seul_pature_M = ones(2,4,2);
1897. for i_pature1 = 1:4
1898. for i_pature2 = 1:2
1899. t_seul_pature_M(:,i_pature1,i_pature2) = t_seul_pature;
1900. end
1901. end
1902. t_seul_pature_DM = t_seul - t_seul_pature_M;
1903. t_seul_pature_DM2 = t_seul_pature_DM.*t_seul_pature_DM;
1904. t_seul_pature_V = nanmean(nanmean(t_seul_pature_DM2,3),2);
1905. t_seul_pature_ET = sqrt(t_seul_pature_V);
1906. t_seul_ET = [t_seul_global_ET t_seul_pature_ET' t_seul_race_ET];
1907. % ANOVA
1908. t_seul_AV = [t_seul(1, :, 1) t_seul(1, :, 2) t_seul(2, :, 1) t_seul(1, :, 2)];
1909. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
1910. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
1911. t_seul_pvalue = anovan(t_seul_AV, {f1, f2}, 'full');
1912. t_seul_pvalue = fliplr(t_seul_pvalue);
1913. close all force
1914. % fusion
1915. t_seul_stat = [t_seul_moy t_seul_ET t_seul_pvalue];
1916. clearvars -except groupe BF2 BF3 n_pres_stat dist_stat t_seul_stat
1917. %% temps pendant lequel BF est la bete la plus proche
1918. % somme par bete
1919. t_BF = zeros(2,4,2);
1920. i=1;
1921. while i <= length(groupe)
1922. for x = 1:2
1923. for y = 1:4
1924. for z = 1:2
1925. if groupe(i,1)==x && groupe(i,2)==y && groupe(i,3)==z
1926. for l = 1:16
1927. if BF3(l,1)==x && BF3(l,2)==y && BF3(l,3)==z
1928. t_BF(x,y,z) = t_BF(x,y,z) + groupe(i,(20+find(BF2(l, :)==1)));
1929. end
1930. end
1931. end
1932. end
1933. end
1934. end
1935. i=i+1;
1936. end
1937. t_BF = t_BF./6;
1938. % moyenne

```

```

1939. t_BF_global = nanmean(nanmean(nanmean(t_BF)));
1940. t_BF_race = nanmean(nanmean(t_BF,3),1);
1941. t_BF_pature = nanmean(nanmean(t_BF,3),2);
1942. t_BF_moy = [t_BF_global t_BF_pature' t_BF_race];
1943. % ecart type
1944. t_BF_global_M = ones(2,4,2)*t_BF_global;
1945. t_BF_global_DM = t_BF - t_BF_global_M;
1946. t_BF_global_DM2 = t_BF_global_DM.*t_BF_global_DM;
1947. t_BF_global_V = nanmean(nanmean(nanmean(t_BF_global_DM2)));
1948. t_BF_global_ET = sqrt(t_BF_global_V);
1949. t_BF_race_M = ones(2,4,2);
1950. for i_race1 = 1:2
1951. for i_race2 = 1:2
1952. t_BF_race_M(i_race1,;,i_race2) = t_BF_race;
1953. end
1954. end
1955. t_BF_race_DM = t_BF - t_BF_race_M;
1956. t_BF_race_DM2 = t_BF_race_DM.*t_BF_race_DM;
1957. t_BF_race_V = nanmean(nanmean(t_BF_race_DM2,3),1);
1958. t_BF_race_ET = sqrt(t_BF_race_V);
1959. t_BF_pature_M = ones(2,4,2);
1960. for i_pature1 = 1:4
1961. for i_pature2 = 1:2
1962. t_BF_pature_M(:,i_pature1,i_pature2) = t_BF_pature;
1963. end
1964. end
1965. t_BF_pature_DM = t_BF - t_BF_pature_M;
1966. t_BF_pature_DM2 = t_BF_pature_DM.*t_BF_pature_DM;
1967. t_BF_pature_V = nanmean(nanmean(t_BF_pature_DM2,3),2);
1968. t_BF_pature_ET = sqrt(t_BF_pature_V);
1969. t_BF_ET = [t_BF_global_ET t_BF_pature_ET' t_BF_race_ET];
1970. % ANOVA
1971. t_BF_AV = [t_BF(1,;,1) t_BF(1,;,2) t_BF(2,;,1) t_BF(2,;,2)];
1972. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
1973. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
1974. t_BF_pvalue = anovan(t_BF_AV,{f1,f2},'full');
1975. t_BF_pvalue = fliplr(t_BF_pvalue);
1976. close all force
1977. % fusion
1978. t_BF_stat = [t_BF_moy t_BF_ET t_BF_pvalue];
1979. clearvars -except groupe BF2 BF3 n_pres_stat dist_stat ...
1980. t_seul_stat t_BF_stat
1981. %% temps passe avec BF
1982. % comptage par bete
1983. t_BF2 = zeros(2,4,2);
1984. i=1;
1985. while i <= length(groupe)
1986. for x = 1:2
1987. for y = 1:4
1988. for z = 1:2
1989. if groupe(i,1)==x && groupe(i,2)==y && groupe(i,3)==z
1990. for l = 1:16

```

```

1991. if BF3(l,1)==x && BF3(l,2)==y && BF3(l,3)==z
1992. t_BF2(x,y,z) = t_BF2(x,y,z) + groupe(i,(5+find(BF2(l,;)==1)));
1993. end
1994. end
1995. end
1996. end
1997. end
1998. end
1999. i=i+1;
2000. end
2001. t_BF2 = t_BF2./6;
2002. % moyenne
2003. t_BF2_global = nanmean(nanmean(nanmean(t_BF2)));
2004. t_BF2_race = nanmean(nanmean(t_BF2,3),1);
2005. t_BF2_pature = nanmean(nanmean(t_BF2,3),2);
2006. t_BF2_moy = [t_BF2_global t_BF2_pature' t_BF2_race];
2007. % ecart type
2008. t_BF2_global_M = ones(2,4,2)*t_BF2_global;
2009. t_BF2_global_DM = t_BF2 - t_BF2_global_M;
2010. t_BF2_global_DM2 = t_BF2_global_DM.*t_BF2_global_DM;
2011. t_BF2_global_V = nanmean(nanmean(nanmean(t_BF2_global_DM2)));
2012. t_BF2_global_ET = sqrt(t_BF2_global_V);
2013. t_BF2_race_M = ones(2,4,2);
2014. for i_race1 = 1:2
2015. for i_race2 = 1:2
2016. t_BF2_race_M(i_race1,;,i_race2) = t_BF2_race;
2017. end
2018. end
2019. t_BF2_race_DM = t_BF2 - t_BF2_race_M;
2020. t_BF2_race_DM2 = t_BF2_race_DM.*t_BF2_race_DM;
2021. t_BF2_race_V = nanmean(nanmean(t_BF2_race_DM2,3),1);
2022. t_BF2_race_ET = sqrt(t_BF2_race_V);
2023. t_BF2_pature_M = ones(2,4,2);
2024. for i_pature1 = 1:4
2025. for i_pature2 = 1:2
2026. t_BF2_pature_M(:,i_pature1,i_pature2) = t_BF2_pature;
2027. end
2028. end
2029. t_BF2_pature_DM = t_BF2 - t_BF2_pature_M;
2030. t_BF2_pature_DM2 = t_BF2_pature_DM.*t_BF2_pature_DM;
2031. t_BF2_pature_V = nanmean(nanmean(t_BF2_pature_DM2,3),2);
2032. t_BF2_pature_ET = sqrt(t_BF2_pature_V);
2033. t_BF2_ET = [t_BF2_global_ET t_BF2_pature_ET' t_BF2_race_ET];
2034. % ANOVA
2035. t_BF2_AV = [t_BF2(1,;,1) t_BF2(1,;,2) t_BF2(2,;,1) t_BF2(2,;,2)];
2036. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
2037. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
2038. t_BF2_pvalue = anovan(t_BF2_AV,{f1,f2},'full');
2039. t_BF2_pvalue = fliplr(t_BF2_pvalue);
2040. close all force
2041. % fusion
2042. t_BF2_stat = [t_BF2_moy t_BF2_ET t_BF2_pvalue];

```

```

2043. clearvars -except groupe BF2 BF3 n_pres_stat dist_stat ...
2044.     t_seul_stat t_BF_stat t_BF2_stat
2045. %% temps passe avec tous les representants de chaque categorie
2046. % comptage par bete
2047. t2 = zeros(2,4,2,5);
2048. i=1;
2049. while i <= length(groupe)
2050.     for x = 1:2
2051.         for y = 1:4
2052.             for z = 1:2
2053.                 if groupe(i,1)==x && groupe(i,2)==y && groupe(i,3)==z
2054.                     for j = 1:4
2055.                         if sum([groupe(i,j*2+5) groupe(i,j*2+6)]) == groupe(i,j+15) && ...
2056.                             groupe(i,j+15) ~= 0
2057.                                 t2(x,y,z,j+1) = t2(x,y,z,j+1) + 1;
2058.                             end
2059.                         end
2060.                     if groupe(i,6) == groupe(i,15) && groupe(i,15) ~= 0
2061.                         t2(x,y,z,1) = t2(x,y,z,1) + 1;
2062.                     end
2063.                 end
2064.             end
2065.         end
2066.     end
2067.     i=i+1;
2068. end
2069. t2 = t2./6;
2070. % moyenne
2071. t2_global = zeros(1,5);
2072. t2_race = zeros(5,4);
2073. t2_pature = zeros(2,5);
2074. for j = 1:5
2075.     t2_global(j) = nanmean(nanmean(nanmean(t2(:,:,,j)))));
2076.     t2_race(j,:) = nanmean(nanmean(t2(:,:,,j),3),1);
2077.     t2_pature(:,j) = nanmean(nanmean(t2(:,:,,j),3),2);
2078. end
2079. t2_moy = [t2_global' t2_pature' t2_race];
2080. % ecart type global
2081. t2_global_M = ones(2,4,2,5);
2082. t2_global_O = zeros(1,1,1,5);
2083. for m = 1:5
2084.     t2_global_O(1,1,1,m) = t2_global(m);
2085. end
2086. for k1=1:2
2087.     for k2=1:4
2088.         for k3=1:2
2089.             t2_global_M(k1,k2,k3,:) = t2_global_O;
2090.         end
2091.     end
2092. end
2093. t2_global_DM = t2 - t2_global_M;
2094. t2_global_DM2 = t2_global_DM.*t2_global_DM;

2095. t2_global_V = zeros(1,5);
2096. for j= 1:5
2097.     t2_global_V(j) = nanmean(nanmean(nanmean(t2_global_DM2(:,:,,j)))));
2098. end
2099. t2_global_ET = sqrt(t2_global_V);
2100. % ecart type race
2101. t2_race_M = ones(2,4,2,5);
2102. t2_race_O = zeros(1,4,1,5);
2103. for m = 1:5
2104.     for n = 1:4
2105.         t2_race_O(1,n,1,m) = t2_race(m,n);
2106.     end
2107. end
2108. for i_race1 = 1:2
2109.     for i_race2 = 1:2
2110.         t2_race_M(i_race1,:,i_race2,:) = t2_race_O;
2111.     end
2112. end
2113. t2_race_DM = t2 - t2_race_M;
2114. t2_race_DM2 = t2_race_DM.*t2_race_DM;
2115. t2_race_V = zeros(5,4);
2116. for j = 1:5
2117.     t2_race_V(j,:) = nanmean(nanmean(t2_race_DM2(:,:,,j),3),1);
2118. end
2119. t2_race_ET = sqrt(t2_race_V);
2120. % ecart type pature
2121. t2_pature_M = ones(2,4,2,5);
2122. t2_pature_O = zeros(2,1,1,5);
2123. for m = 1:2
2124.     for n = 1:5
2125.         t2_pature_O(m,1,1,n) = t2_pature(m,n);
2126.     end
2127. end
2128. for i_pature1 = 1:4
2129.     for i_pature2 = 1:2
2130.         t2_pature_M(:,i_pature1,i_pature2,:) = t2_pature_O;
2131.     end
2132. end
2133. t2_pature_DM = t2 - t2_pature_M;
2134. t2_pature_DM2 = t2_pature_DM.*t2_pature_DM;
2135. t2_pature_V = zeros(2,5);
2136. for j = 1:5
2137.     t2_pature_V(:,j) = nanmean(nanmean(t2_pature_DM2(:,:,,j),3),2);
2138. end
2139. t2_pature_ET = sqrt(t2_pature_V);
2140. % ecart type fusion
2141. t2_ET = [t2_global_ET' t2_pature_ET' t2_race_ET];
2142. % ANOVA
2143. t2_AV = zeros(5,16);
2144. t2_pvalue = zeros(5,3);
2145. for m = 1:5
2146.     t2_AV(m,:) = [t2(1,:,1,m) t2(1,:,2,m) t2(2,:,1,m) t2(1,:,2,m)];

```

```

2147. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
2148. f2 = [1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2];
2149. t2_pvalue(m,:) = fliplr((anovan(t2_AV(m,:),{f1,f2},'full')));
2150. close all force
2151. end
2152. % fusion
2153. t2_stat = [t2_moy t2_ET t2_pvalue];
2154. clearvars -except groupe BF2 BF3 n_pres_stat dist_stat ...
2155. t_seul_stat t_BF_stat t_BF2_stat t2 t2_stat
2156. %% temps passe avec au moins un representant de chaque categorie
2157. % comptage par bete
2158. t1 = zeros(2,4,2,5);
2159. i=1;
2160. while i <= length(groupe)
2161. for x = 1:2
2162. for y = 1:4
2163. for z = 1:2
2164. if groupe(i,1)==x && groupe(i,2)==y && groupe(i,3)==z
2165. for j = 1:4
2166. t1(x,y,z,j+1) = t1(x,y,z,j+1) + max([groupe(i,j*2+5) groupe(i,j*2+6)]);
2167. end
2168. t1(x,y,z,1) = t1(x,y,z,1) + all(groupe(i,6));
2169. end
2170. end
2171. end
2172. end
2173. i=i+1;
2174. end
2175. t1 = t1./6;
2176. for i1 = 1:2
2177. for i2 = 1:4
2178. for i3 = 1:2
2179. for i4 = 1:5
2180. if t1(i1,i2,i3,i4) == 12
2181. t1(i1,i2,i3,i4) = t2(i1,i2,i3,i4);
2182. end
2183. end
2184. end
2185. end
2186. end
2187. % moyenne
2188. t1_global = zeros(1,5);
2189. t1_race = zeros(5,4);
2190. t1_pature = zeros(2,5);
2191. for j = 1:5
2192. t1_global(j) = nanmean(nanmean(nanmean(t1(:,:,:,j))));
2193. t1_race(j,:) = nanmean(nanmean(t1(:,:,:,j),3),1);
2194. t1_pature(:,j) = nanmean(nanmean(t1(:,:,:,j),3),2);
2195. end
2196. t1_moy = [t1_global' t1_pature' t1_race];
2197. % ecart type global
2198. t1_global_M = ones(2,4,2,5);
2199. t1_global_O = zeros(1,1,1,5);
2200. for m = 1:5
2201. t1_global_O(1,1,1,m) = t1_global(m);
2202. end
2203. for k1=1:2
2204. for k2=1:4
2205. for k3=1:2
2206. t1_global_M(k1,k2,k3,:) = t1_global_O;
2207. end
2208. end
2209. end
2210. t1_global_DM = t1 - t1_global_M;
2211. t1_global_DM2 = t1_global_DM.*t1_global_DM;
2212. t1_global_V = zeros(1,5);
2213. for j = 1:5
2214. t1_global_V(j) = nanmean(nanmean(nanmean(t1_global_DM2(:,:,:,j))));
2215. end
2216. t1_global_ET = sqrt(t1_global_V);
2217. % ecart type race
2218. t1_race_M = ones(2,4,2,5);
2219. t1_race_O = zeros(1,4,1,5);
2220. for m = 1:5
2221. for n = 1:4
2222. t1_race_O(1,n,1,m) = t1_race(m,n);
2223. end
2224. end
2225. for i_race1 = 1:2
2226. for i_race2 = 1:2
2227. t1_race_M(i_race1,:,i_race2,:) = t1_race_O;
2228. end
2229. end
2230. t1_race_DM = t1 - t1_race_M;
2231. t1_race_DM2 = t1_race_DM.*t1_race_DM;
2232. t1_race_V = zeros(5,4);
2233. for j = 1:5
2234. t1_race_V(j,:) = nanmean(nanmean(t1_race_DM2(:,:,:,j),3),1);
2235. end
2236. t1_race_ET = sqrt(t1_race_V);
2237. % ecart type pature
2238. t1_pature_M = ones(2,4,2,5);
2239. t1_pature_O = zeros(2,1,1,5);
2240. for m = 1:2
2241. for n = 1:5
2242. t1_pature_O(m,1,1,n) = t1_pature(m,n);
2243. end
2244. end
2245. for i_pature1 = 1:4
2246. for i_pature2 = 1:2
2247. t1_pature_M(:,i_pature1,i_pature2,:) = t1_pature_O;
2248. end
2249. end
2250. t1_pature_DM = t1 - t1_pature_M;

```

```

2251. t1_pature_DM2 = t1_pature_DM.*t1_pature_DM ;
2252. t1_pature_V = zeros(2,5) ;
2253. for j = 1:5
2254. t1_pature_V(:,j) = nanmean(nanmean(t1_pature_DM2(:, :, j), 3), 2) ;
2255. end
2256. t1_pature_ET = sqrt(t1_pature_V) ;
2257. % ecart type fusion
2258. t1_ET = [t1_global_ET' t1_pature_ET' t1_race_ET] ;
2259. % ANOVA
2260. t1_AV = zeros(5,16) ;
2261. t1_pvalue = zeros(5,3) ;
2262. for m = 1:5
2263. t1_AV(m,:) = [t1(1, :, m) t1(1, :, 2, m) t1(2, :, 1, m) t1(1, :, 2, m)] ;
2264. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
2265. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
2266. t1_pvalue(m,:) = fliplr((anovan(t1_AV(m,:), {f1, f2}, 'full'))') ;
2267. close all force
2268. end
2269. % fusion
2270. t1_stat = [t1_moy t1_ET t1_pvalue] ;
2271. clearvars -except groupe BF2 BF3 n_pres_stat dist_stat ...
2272. t_seul_stat t_BF_stat t_BF2_stat t1_stat t2_stat
2273. %% temps pendant lequel un representant de chaque categorie
2274. %% est la bete la plus proche
2275. % comptage par bete
2276. tpp = zeros(2,4,2,5) ;
2277. i=1 ;
2278. while i <= length(groupe)
2279. for x = 1:2
2280. for y = 1:4
2281. for z = 1:2
2282. if groupe(i,1)==x && groupe(i,2)==y && groupe(i,3)==z
2283. for j = 1:4
2284. tpp(x,y,z,j+1) = tpp(x,y,z,j+1) + sum([groupe(i,j*2+20) ...
2285. groupe(i,j*2+21)]) ;
2286. end
2287. tpp(x,y,z,1) = tpp(x,y,z,1) + groupe(i,21) ;
2288. end
2289. end
2290. end
2291. end
2292. i=i+1 ;
2293. end
2294. tpp = tpp./6 ;
2295. % moyenne
2296. tpp_global = zeros(1,5) ;
2297. tpp_race = zeros(5,4) ;
2298. tpp_pature = zeros(2,5) ;
2299. for j = 1:5
2300. tpp_global(j) = nanmean(nanmean(nanmean(tpp(:, :, j)))) ;
2301. tpp_race(j,:) = nanmean(nanmean(tpp(:, :, j), 3), 1) ;
2302. tpp_pature(:, j) = nanmean(nanmean(tpp(:, :, j), 3), 2) ;

```

```

2303. end
2304. tpp_moy = [tpp_global' tpp_pature' tpp_race] ;
2305. % ecart type global
2306. tpp_global_M = ones(2,4,2,5) ;
2307. tpp_global_O = zeros(1,1,1,5) ;
2308. for m = 1:5
2309. tpp_global_O(1,1,1,m) = tpp_global(m) ;
2310. end
2311. for k1=1:2
2312. for k2=1:4
2313. for k3=1:2
2314. tpp_global_M(k1,k2,k3,:) = tpp_global_O ;
2315. end
2316. end
2317. end
2318. tpp_global_DM = tpp - tpp_global_M ;
2319. tpp_global_DM2 = tpp_global_DM.*tpp_global_DM ;
2320. tpp_global_V = zeros(1,5) ;
2321. for j= 1:5
2322. tpp_global_V(j) = nanmean(nanmean(nanmean(tpp_global_DM2(:, :, j)))) ;
2323. end
2324. tpp_global_ET = sqrt(tpp_global_V) ;
2325. % ecart type race
2326. tpp_race_M = ones(2,4,2,5) ;
2327. tpp_race_O = zeros(1,4,1,5) ;
2328. for m = 1:5
2329. for n = 1:4
2330. tpp_race_O(1,n,1,m) = tpp_race(m,n) ;
2331. end
2332. end
2333. for i_race1 = 1:2
2334. for i_race2 = 1:2
2335. tpp_race_M(i_race1, :, i_race2, :) = tpp_race_O ;
2336. end
2337. end
2338. tpp_race_DM = tpp - tpp_race_M ;
2339. tpp_race_DM2 = tpp_race_DM.*tpp_race_DM ;
2340. tpp_race_V = zeros(5,4) ;
2341. for j = 1:5
2342. tpp_race_V(j, :) = nanmean(nanmean(tpp_race_DM2(:, :, j), 3), 1) ;
2343. end
2344. tpp_race_ET = sqrt(tpp_race_V) ;
2345. % ecart type pature
2346. tpp_pature_M = ones(2,4,2,5) ;
2347. tpp_pature_O = zeros(2,1,1,5) ;
2348. for m = 1:2
2349. for n = 1:5
2350. tpp_pature_O(m,1,1,n) = tpp_pature(m,n) ;
2351. end
2352. end
2353. for i_pature1 = 1:4
2354. for i_pature2 = 1:2

```

```

2355. tpp_pature_M(:,i_pature1,i_pature2,:) = tpp_pature_O ;
2356. end
2357. end
2358. tpp_pature_DM = tpp - tpp_pature_M ;
2359. tpp_pature_DM2 = tpp_pature_DM.*tpp_pature_DM ;
2360. tpp_pature_V = zeros(2,5) ;
2361. for j = 1:5
2362. tpp_pature_V(:,j) = nanmean(nanmean(tpp_pature_DM2(:,:,j),3),2) ;
2363. end
2364. tpp_pature_ET = sqrt(tpp_pature_V) ;
2365. % ecart type fusion
2366. tpp_ET = [tpp_global_ET' tpp_pature_ET' tpp_race_ET] ;
2367. % ANOVA
2368. tpp_AV = zeros(5,16) ;
2369. tpp_pvalue = zeros(5,3) ;
2370. for m = 1:5
2371. tpp_AV(m,:) = [tpp(1, :, m) tpp(1, :, 2, m) tpp(2, :, 1, m) tpp(1, :, 2, m)] ;
2372. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
2373. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
2374. tpp_pvalue(m,:) = flipr(anovan(tpp_AV(m,:),{f1,f2},'full')) ;
2375. close all force
2376. end
2377. % fusion
2378. tpp_stat = [tpp_moy tpp_ET tpp_pvalue] ;
2379. clearvars -except groupe BF2 BF3 n_pres_stat dist_stat ...
2380. t_seul_stat t_BF_stat t_BF2_stat t1_stat t2_stat tpp_stat
2381. %% compilation en une seule matrice
2382. t1_stat = [t1_stat(2:5,:); t1_stat(1,:)] ;
2383. t2_stat = [t2_stat(2:5,:); t2_stat(1,:)] ;
2384. tpp_stat = [tpp_stat(2:5,:); tpp_stat(1,:)] ;
2385. stat_groupe_final = [n_pres_stat ; dist_stat ; t_seul_stat ; ...
2386. t_BF_stat ; t_BF2_stat ; t1_stat ; t2_stat ; tpp_stat] ;
2387. C = stat_groupe_final ;
2388. stat_groupe_final = [C(:,1) C(:,8) C(:,15) C(:,2) C(:,9) C(:,3) ...
2389. C(:,10) C(:,16) C(:,4) C(:,11) C(:,5) C(:,12) C(:,6) C(:,13) ...
2390. C(:,7) C(:,14) C(:,17)] ;
2391. %% ecriture fichier
2392. xlswrite('stat_groupe3',stat_groupe_final)
2393. %% ecriture lignes
2394. S1 ='nombre de betes presentes' ;
2395. S2 ='distance a laquelle se trouve la bete la plus proche (m)' ;
2396. S3 ='temps passe seul (h)' ;
2397. S4 ='temps pendant lequel la bete le plus souvent la PP est la bete la PP (h)' ;
2398. S5 ='temps pendant lequel la bete le plus souvent la PP est presente (h)' ;
2399. S6 ='temps passe avec au moins un ndama (h)' ;
2400. S7 ='temps passe avec au moins un zebu (h)' ;
2401. S8 ='temps passe avec au moins un ndapol (h)' ;
2402. S9 ='temps passe avec au moins un zepol (h)' ;
2403. S10 ='temps passe avec au moins une bete hors protocole (h)' ;
2404. S11 ='temps passe avec tous les ndamas (h)' ;
2405. S12 ='temps passe avec tous les zebras (h)' ;
2406. S13 ='temps passe avec tous les ndapols (h)' ;

```

```

2407. S14 ='temps passe avec tous les zepols (h)' ;
2408. S15 ='temps passe avec toutes les betes hors protocole (h)' ;
2409. S16 ='temps pendant lequel la bete la PP est un ndama (h)' ;
2410. S17 ='temps pendant lequel la bete la PP est un zebu (h)' ;
2411. S18 ='temps pendant lequel la bete la PP est un ndapol (h)' ;
2412. S19 ='temps pendant lequel la bete la PP est un zepol (h)' ;
2413. S20 ='temps pendant lequel la bete la PP est une bete hors protocole (h)' ;
2414. lignes={S1;S2;S3;S4;S5;S6;S7;S8;S9;S10; ...
2415. S11;S12;S13;S14;S15;S16;S17;S18;S19;S20} ;
2416. xlswrite('lignes_stat_groupe',lignes)
2417. clearvars -except groupe BF3 n_pres_stat dist_stat ...
2418. t_seul_stat t_BF_stat t_BF2_stat t1_stat t2_stat ...
2419. tpp_stat stat_groupe_final

```

Annexe 17 : description de la fonction de prétraitement des données provenant des iPhones

1. Le jour et le mois (entrées) sont transformés en jours de l'expérimentation, numérotés à partir du 12/03/18 et les heures, minutes et secondes (entrées) sont transformées en secondes.
2. Les colonnes temps, latitude, longitude, accélération gravitationnelle selon l'axe X et vitesse sont extraites de la matrice (entrées) pour former une nouvelle matrice.
3. La fréquence de mesure est passée de 100 Hz à 1 Hz en moyennant, pour chaque colonne, les valeurs groupées par 100 lignes. La colonne temps est arrondie à la seconde.
4. Les lignes se référant à des positions situées hors des pâtures 1 et 2 sont éliminées en comparant les colonnes latitude et longitude avec la latitude et la longitude des angles des clôtures.
5. Une nouvelle matrice est créée, comprenant 9 colonnes : le code de la pâture (entrées), le code de la race (entrées), le code de l'individu (entrées), le jour (étape 1), l'heure (colonne « temps » de l'étape 2, additionnée de la valeur « secondes » de l'étape 1, puis transformée en heures décimales), la latitude, la longitude, l'accélération gravitationnelle selon l'axe X et la vitesse.
6. Les colonnes jour et heure sont transformées afin que l'heure ne dépasse pas 24h.
7. L'accélération gravitationnelle selon l'axe X est standardisée (moyenne égale à 0 et écart-type égal à 1).

Annexe 18 : description du script du prétraitement des données provenant des iPhones

1. Le nombre de chiffres significatifs utilisés dans l'environnement MATLAB est augmenté par rapport à la valeur par défaut afin de pouvoir travailler avec des degrés décimaux.
2. Chaque fichier « csv » provenant de l'application « Sensor Data » est lu. La fonction de prétraitement des données des iPhones est alors appliquée à la matrice contenant les données, à laquelle a été jointe une retranscription des informations reprises dans le nom du fichier.
3. L'ensemble des sorties de la fonction est fusionné dans une matrice « compilation ». Celle-ci est écrite en format « csv ».
4. La matrice « compilation » est divisée en 16 matrices, en fonction de la bête à laquelle les données se rapportent. Celles-ci sont écrites en format « csv ».
5. La matrice « compilation » (étape 3) est divisée en 16 matrices, en fonction de la bête à laquelle les données se rapportent. Chacune de ces matrices est divisée en deux matrices : l'une comprenant les lignes pour lesquelles l'heure est comprise entre 6h30 et 18h30 et l'autre comprenant les lignes pour lesquelles l'heure est comprise entre 18h30 et 6h30. Ces matrices sont ensuite privées de leurs colonnes « accélération gravitationnelle selon l'axe X » et « vitesse » et sont écrites en format « csv ».
6. La matrice « compilation » (étape 3) est divisée en 16 matrices, en fonction de la bête à laquelle les données se rapportent. Chacune de ces matrices est divisée en deux matrices : l'une comprenant les lignes pour lesquelles l'accélération gravitationnelle selon l'axe X est plus grande que 0,5 et l'autre comprenant les lignes pour lesquelles l'accélération gravitationnelle selon l'axe X est plus petite que 0,5. Ces matrices sont ensuite privées de leurs colonnes « accélération gravitationnelle selon l'axe X » et « vitesse » et sont écrites en format « csv ».
7. La matrice « compilation » (étape 3) est divisée en 2 matrices, en fonction de la pâture où se trouve la bête à laquelle les données se rapportent. Chacune de ces matrices est divisée en deux matrices : l'une comprenant les lignes pour lesquelles l'heure est comprise entre 6h30 et 18h30 et l'autre comprenant les lignes pour lesquelles l'heure est comprise entre 18h30 et 6h30. Ces matrices sont ensuite privées de leurs colonnes « accélération gravitationnelle selon l'axe X » et « vitesse » et sont écrites en format « csv ».
8. La matrice « compilation » (étape 3) est divisée en 2 matrices, en fonction de la pâture où se trouve la bête à laquelle les données se rapportent. Chacune de ces matrices est divisée en deux matrices : l'une comprenant les lignes pour lesquelles l'accélération gravitationnelle selon l'axe X est plus grande que 0,5 et l'autre comprenant les lignes pour lesquelles l'accélération gravitationnelle selon l'axe X est plus petite que 0,5. Ces matrices sont ensuite privées de leurs colonnes « accélération gravitationnelle selon l'axe X » et « vitesse » et sont écrites en format « csv ».

Annexe 19 : fonction de prétraitement

des données provenant des iPhones

```
2420. %% fonction pretraitement
2421. fonction[S] = compil_iphone_fct(E, pature, race, individu, ...
2422.     mois, jour, heure, minute, seconde)
2423. % date -> jour de l'experimentation
2424. if mois == 3
2425.     j = jour - 11 ;
2426. elseif mois == 4
2427.     j = jour + 20 ;
2428. elseif mois == 5
2429.     j = jour + 50 ;
2430. end
2431. t = 3600*heure + 60*minute + seconde ;
2432. % creer matrice
2433. f_1 = [E(:,1),E(:,36),E(:,37),E(:,21),E(:,40)] ;
2434. L1 = length(f_1) ;
2435. clearvars -except f_1 j t pature race individu L1
2436. % 100 Hz -> 1 Hz
2437. f_2 = zeros(L1,5) ;
2438. i1 = 0 ;
2439. while 100*i1+100 <= L1
2440.     f_2(i1+1,1) = f_1(100*i1+100,1) ;
2441.     f_2(i1+1,1) = round(f_2(i1+1,1)) ;
2442.     for i2 = 1:100
2443.         f_2(i1+1,2) = f_2(i1+1,2) + f_1(100*i1+i2,2) ;
2444.     end
2445.     f_2(i1+1,2) = f_2(i1+1,2)/100 ;
2446.     for i2 = 1:100
2447.         f_2(i1+1,3) = f_2(i1+1,3) + f_1(100*i1+i2,3) ;
2448.     end
2449.     f_2(i1+1,3) = f_2(i1+1,3)/100 ;
2450.     for i2 = 1:100
2451.         f_2(i1+1,4) = f_2(i1+1,4) + f_1(100*i1+i2,4) ;
2452.     end
2453.     f_2(i1+1,4) = f_2(i1+1,4)/100 ;
2454.     for i2 = 1:100
2455.         f_2(i1+1,5) = f_2(i1+1,5) + f_1(100*i1+i2,5) ;
2456.     end
2457.     f_2(i1+1,5) = f_2(i1+1,5)/100 ;
2458.     i1 = i1+1 ;
2459. end
2460. f_2 = f_2(1:i1,:) ;
2461. L2 = length(f_2) ;
2462. clearvars -except f_2 j t pature race individu L2
2463. % supprimer hors carte
2464. f_3 = zeros(L2,5) ;
2465. angles = [11.518189 -3.380525
2466.     11.519188 -3.377799
```

```
2467.     11.521352 -3.37616
2468.     11.522025 -3.37965
2469.     11.522111 -3.379773
2470.     11.522856 -3.379078] ;
2471. sud_1 = [((angles(4,2)-angles(1,2))/(angles(4,1)-angles(1,1))) ...
2472.     (angles(1,2)-angles(1,1))*((angles(4,2)-angles(1,2))/ ...
2473.     (angles(4,1)-angles(1,1))))] ;
2474. sud_2 = sud_1 ;
2475. sud_3 = sud_1 ;
2476. sud_4 = [((angles(5,2)-angles(4,2))/(angles(5,1)-angles(4,1))) ...
2477.     (angles(4,2)-angles(4,1))*((angles(5,2)-angles(4,2))/ ...
2478.     (angles(5,1)-angles(4,1))))] ;
2479. sud_5 = [((angles(6,2)-angles(5,2))/(angles(6,1)-angles(5,1))) ...
2480.     (angles(5,2)-angles(5,1))*((angles(6,2)-angles(5,2))/ ...
2481.     (angles(6,1)-angles(5,1))))] ;
2482. nord_1 = [((angles(2,2)-angles(1,2))/(angles(2,1)-angles(1,1))) ...
2483.     (angles(1,2)-angles(1,1))*((angles(2,2)-angles(1,2))/ ...
2484.     (angles(2,1)-angles(1,1))))] ;
2485. nord_2 = [((angles(3,2)-angles(2,2))/(angles(3,1)-angles(2,1))) ...
2486.     (angles(2,2)-angles(2,1))*((angles(3,2)-angles(2,2))/ ...
2487.     (angles(3,1)-angles(2,1))))] ;
2488. nord_3 = [((angles(6,2)-angles(3,2))/(angles(6,1)-angles(3,1))) ...
2489.     (angles(3,2)-angles(3,1))*((angles(6,2)-angles(3,2))/ ...
2490.     (angles(6,1)-angles(3,1))))] ;
2491. nord_4 = nord_3 ;
2492. nord_5 = nord_3 ;
2493. i3 = 1 ;
2494. i4 = 1 ;
2495. while i3 <= L2
2496.     if angles(1,1) <= f_2(i3,3) && f_2(i3,3) < angles(2,1)
2497.         if f_2(i3,2) - (sud_1(1) * f_2(i3,3) + sud_1(2)) >= 0 && ...
2498.             f_2(i3,2) - (nord_1(1) * f_2(i3,3) + nord_1(2)) <= 0
2499.             f_3(i4,:) = f_2(i3,:);
2500.             i4 = i4 + 1 ;
2501.         end
2502.     elseif angles(2,1) <= f_2(i3,3) && f_2(i3,3) < angles(3,1)
2503.         if f_2(i3,2) - (sud_2(1) * f_2(i3,3) + sud_2(2)) >= 0 && ...
2504.             f_2(i3,2) - (nord_2(1) * f_2(i3,3) + nord_2(2)) <= 0
2505.             f_3(i4,:) = f_2(i3,:);
2506.             i4 = i4 + 1 ;
2507.         end
2508.     elseif angles(3,1) <= f_2(i3,3) && f_2(i3,3) < angles(4,1)
2509.         if f_2(i3,2) - (sud_3(1) * f_2(i3,3) + sud_3(2)) >= 0 && ...
2510.             f_2(i3,2) - (nord_3(1) * f_2(i3,3) + nord_3(2)) <= 0
2511.             f_3(i4,:) = f_2(i3,:);
2512.             i4 = i4 + 1 ;
2513.         end
2514.     elseif angles(4,1) <= f_2(i3,3) && f_2(i3,3) < angles(5,1)
2515.         if f_2(i3,2) - (sud_4(1) * f_2(i3,3) + sud_4(2)) >= 0 && ...
2516.             f_2(i3,2) - (nord_4(1) * f_2(i3,3) + nord_4(2)) <= 0
2517.             f_3(i4,:) = f_2(i3,:);
2518.             i4 = i4 + 1 ;
```

```

2519.     end
2520. elseif angles(5,1) <= f_2(i3,3) && f_2(i3,3) < angles(6,1)
2521.     if f_2(i3,2) - (sud_5(1) * f_2(i3,3) + sud_5(2)) >= 0 && ...
2522.         f_2(i3,2) - (nord_5(1) * f_2(i3,3) + nord_5(2)) <= 0
2523.         f_3(i4,:) = f_2(i3,:);
2524.         i4 = i4 + 1;
2525.     end
2526. end
2527. i3 = i3 + 1;
2528. end
2529. f_3 = f_3(1:i4-1,:);
2530. L3 = length(f_3);
2531. clearvars -except f_3 j t pature race individu L3
2532. % ajout colonnes
2533. f_4 = [repmat(pature,L3,1) repmat(race,L3,1) repmat(individu,L3,1) ...
2534.       repmat(j,L3,1) (f_3(:,1)+t)/3600 f_3(:,2:5)];
2535. clearvars -except f_4 L3
2536. % correction jour
2537. i5=1;
2538. while i5 <= L3
2539.     if f_4(i5,5) >= 48
2540.         f_4(i5,5) = f_4(i5,5) - 48;
2541.         f_4(i5,4) = f_4(i5,4) + 2;
2542.     elseif f_4(i5,5) >= 24
2543.         f_4(i5,5) = f_4(i5,5) - 24;
2544.         f_4(i5,4) = f_4(i5,4) + 1;
2545.     end
2546.     i5=i5+1;
2547. end
2548. clearvars -except f_4
2549. % normalisation GravAccX
2550. f_4(:,8) = (f_4(:,8)-mean(f_4(:,8)))/std(f_4(:,8));
2551. % sortie
2552. S = f_4;
2553. clearvars f_4
2554. end

```

Annexe 20 : script du prétraitement

des données provenant des iPhones

```
2555.   %%% pretraitement
2556.   %% creer matrice
2557.   Compil = zeros(0,9) ;
2558.   %% augmenter chiffres significatifs
2559.   format long g
2560.   %% repeter fonction pour chaque fichier
2561.   E = csvread('SD_0313_2NA_073948.csv',1,0) ;
2562.   pature = 2 ;
2563.   race = 1 ;
2564.   individu = 1 ;
2565.   mois = 3 ;
2566.   jour = 13 ;
2567.   heure = 7 ;
2568.   minute = 39 ;
2569.   seconde = 48 ;
2570.   S = compil_iphone_fct(E, pature, race, individu, ...
2571.       mois, jour, heure, minute, seconde) ;
2572.   Compil = [Compil ; S] ;
2573.   clearvars -except Compil
2574.   E = csvread('SD_0313_2ZA_074038.csv',1,0) ;
2575.   pature = 2 ;
2576.   race = 2 ;
2577.   individu = 1 ;
2578.   mois = 3 ;
2579.   jour = 13 ;
2580.   heure = 7 ;
2581.   minute = 40 ;
2582.   seconde = 38 ;
2583.   S = compil_iphone_fct(E, pature, race, individu, ...
2584.       mois, jour, heure, minute, seconde) ;
2585.   Compil = [Compil ; S] ;
2586.   clearvars -except Compil
2587.   E = csvread('SD_0313_2ZpA_074233.csv',1,0) ;
2588.   pature = 2 ;
2589.   race = 4 ;
2590.   individu = 1 ;
2591.   mois = 3 ;
2592.   jour = 13 ;
2593.   heure = 7 ;
2594.   minute = 42 ;
2595.   seconde = 33 ;
2596.   S = compil_iphone_fct(E, pature, race, individu, ...
2597.       mois, jour, heure, minute, seconde) ;
2598.   Compil = [Compil ; S] ;
2599.   clearvars -except Compil
2600.   E = csvread('SD_0317_1NA_082047.csv',1,0) ;
2601.   pature = 1 ;
```

```
2602.   race = 1 ;
2603.   individu = 1 ;
2604.   mois = 3 ;
2605.   jour = 17 ;
2606.   heure = 8 ;
2607.   minute = 20 ;
2608.   seconde = 47 ;
2609.   S = compil_iphone_fct(E, pature, race, individu, ...
2610.       mois, jour, heure, minute, seconde) ;
2611.   Compil = [Compil ; S] ;
2612.   clearvars -except Compil
2613.   E = csvread('SD_0317_1NpA_082502.csv',1,0) ;
2614.   pature = 1 ;
2615.   race = 3 ;
2616.   individu = 1 ;
2617.   mois = 3 ;
2618.   jour = 17 ;
2619.   heure = 8 ;
2620.   minute = 25 ;
2621.   seconde = 2 ;
2622.   S = compil_iphone_fct(E, pature, race, individu, ...
2623.       mois, jour, heure, minute, seconde) ;
2624.   Compil = [Compil ; S] ;
2625.   clearvars -except Compil
2626.   E = csvread('SD_0317_1ZA_081140.csv',1,0) ;
2627.   pature = 1 ;
2628.   race = 2 ;
2629.   individu = 1 ;
2630.   mois = 3 ;
2631.   jour = 17 ;
2632.   heure = 8 ;
2633.   minute = 11 ;
2634.   seconde = 40 ;
2635.   S = compil_iphone_fct(E, pature, race, individu, ...
2636.       mois, jour, heure, minute, seconde) ;
2637.   Compil = [Compil ; S] ;
2638.   clearvars -except Compil
2639.   E = csvread('SD_0317_1ZpA_081703.csv',1,0) ;
2640.   pature = 1 ;
2641.   race = 4 ;
2642.   individu = 1 ;
2643.   mois = 3 ;
2644.   jour = 17 ;
2645.   heure = 8 ;
2646.   minute = 17 ;
2647.   seconde = 3 ;
2648.   S = compil_iphone_fct(E, pature, race, individu, ...
2649.       mois, jour, heure, minute, seconde) ;
2650.   Compil = [Compil ; S] ;
2651.   clearvars -except Compil
2652.   E = csvread('SD_0321_2ZB_082305.csv',1,0) ;
2653.   pature = 2 ;
```

```

2654. race = 2 ;
2655. individu = 2 ;
2656. mois = 3 ;
2657. jour = 21 ;
2658. heure = 8 ;
2659. minute = 23 ;
2660. seconde = 5 ;
2661. S = compil_iphone_fct(E, pature, race, individu, ...
2662.     mois, jour, heure, minute, seconde) ;
2663. Compil = [Compil ; S] ;
2664. clearvars -except Compil
2665. E = csvread('SD_0326_2NB_065434.csv',1,0) ;
2666. pature = 2 ;
2667. race = 1 ;
2668. individu = 2 ;
2669. mois = 3 ;
2670. jour = 26 ;
2671. heure = 6 ;
2672. minute = 54 ;
2673. seconde = 34 ;
2674. S = compil_iphone_fct(E, pature, race, individu, ...
2675.     mois, jour, heure, minute, seconde) ;
2676. Compil = [Compil ; S] ;
2677. clearvars -except Compil
2678. E = csvread('SD_0326_2NpB_065408.csv',1,0) ;
2679. pature = 2 ;
2680. race = 3 ;
2681. individu = 2 ;
2682. mois = 3 ;
2683. jour = 26 ;
2684. heure = 6 ;
2685. minute = 54 ;
2686. seconde = 8 ;
2687. S = compil_iphone_fct(E, pature, race, individu, ...
2688.     mois, jour, heure, minute, seconde) ;
2689. Compil = [Compil ; S] ;
2690. clearvars -except Compil
2691. E = csvread('SD_0326_2ZB_065205.csv',1,0) ;
2692. pature = 2 ;
2693. race = 2 ;
2694. individu = 2 ;
2695. mois = 3 ;
2696. jour = 26 ;
2697. heure = 6 ;
2698. minute = 52 ;
2699. seconde = 5 ;
2700. S = compil_iphone_fct(E, pature, race, individu, ...
2701.     mois, jour, heure, minute, seconde) ;
2702. Compil = [Compil ; S] ;
2703. clearvars -except Compil
2704. E = csvread('SD_0326_2ZpB_065319.csv',1,0) ;
2705. pature = 2 ;

```

```

2706. race = 4 ;
2707. individu = 2 ;
2708. mois = 3 ;
2709. jour = 26 ;
2710. heure = 6 ;
2711. minute = 53 ;
2712. seconde = 19 ;
2713. S = compil_iphone_fct(E, pature, race, individu, ...
2714.     mois, jour, heure, minute, seconde) ;
2715. Compil = [Compil ; S] ;
2716. clearvars -except Compil
2717. E = csvread('SD_0329_1NA_093858.csv',1,0) ;
2718. pature = 1 ;
2719. race = 1 ;
2720. individu = 1 ;
2721. mois = 3 ;
2722. jour = 29 ;
2723. heure = 9 ;
2724. minute = 38 ;
2725. seconde = 58 ;
2726. S = compil_iphone_fct(E, pature, race, individu, ...
2727.     mois, jour, heure, minute, seconde) ;
2728. Compil = [Compil ; S] ;
2729. clearvars -except Compil
2730. E = csvread('SD_0329_1NpA_094125.csv',1,0) ;
2731. pature = 1 ;
2732. race = 3 ;
2733. individu = 1 ;
2734. mois = 3 ;
2735. jour = 29 ;
2736. heure = 9 ;
2737. minute = 41 ;
2738. seconde = 25 ;
2739. S = compil_iphone_fct(E, pature, race, individu, ...
2740.     mois, jour, heure, minute, seconde) ;
2741. Compil = [Compil ; S] ;
2742. clearvars -except Compil
2743. E = csvread('SD_0329_1ZA_093958.csv',1,0) ;
2744. pature = 1 ;
2745. race = 2 ;
2746. individu = 1 ;
2747. mois = 3 ;
2748. jour = 29 ;
2749. heure = 9 ;
2750. minute = 39 ;
2751. seconde = 58 ;
2752. S = compil_iphone_fct(E, pature, race, individu, ...
2753.     mois, jour, heure, minute, seconde) ;
2754. Compil = [Compil ; S] ;
2755. clearvars -except Compil
2756. E = csvread('SD_0329_1ZpA_094236.csv',1,0) ;
2757. pature = 1 ;

```

```

2758. race = 4 ;
2759. individu = 1 ;
2760. mois = 3 ;
2761. jour = 29 ;
2762. heure = 9 ;
2763. minute = 42 ;
2764. seconde = 36 ;
2765. S = compil_iphone_fct(E, pature, race, individu, ...
2766.     mois, jour, heure, minute, seconde) ;
2767. Compil = [Compil ; S] ;
2768. clearvars -except Compil
2769. % NB : ! iphone a l'envers !
2770. E = csvread('SD_0402_1NB_092543.csv',1,0) ;
2771. pature = 1 ;
2772. race = 1 ;
2773. individu = 2 ;
2774. mois = 4 ;
2775. jour = 2 ;
2776. heure = 9 ;
2777. minute = 25 ;
2778. seconde = 43 ;
2779. E(:,21) = -1 * E(:,21) ;
2780. S = compil_iphone_fct(E, pature, race, individu, ...
2781.     mois, jour, heure, minute, seconde) ;
2782. Compil = [Compil ; S] ;
2783. clearvars -except Compil
2784. E = csvread('SD_0402_1NpB_093044.csv',1,0) ;
2785. pature = 1 ;
2786. race = 3 ;
2787. individu = 2 ;
2788. mois = 4 ;
2789. jour = 2 ;
2790. heure = 9 ;
2791. minute = 30 ;
2792. seconde = 44 ;
2793. S = compil_iphone_fct(E, pature, race, individu, ...
2794.     mois, jour, heure, minute, seconde) ;
2795. Compil = [Compil ; S] ;
2796. clearvars -except Compil
2797. E = csvread('SD_0402_1ZB_092804.csv',1,0) ;
2798. pature = 1 ;
2799. race = 2 ;
2800. individu = 2 ;
2801. mois = 4 ;
2802. jour = 2 ;
2803. heure = 9 ;
2804. minute = 28 ;
2805. seconde = 4 ;
2806. S = compil_iphone_fct(E, pature, race, individu, ...
2807.     mois, jour, heure, minute, seconde) ;
2808. Compil = [Compil ; S] ;
2809. clearvars -except Compil

```

```

2810. E = csvread('SD_0402_1ZpB_093315.csv',1,0) ;
2811. pature = 1 ;
2812. race = 4 ;
2813. individu = 2 ;
2814. mois = 4 ;
2815. jour = 2 ;
2816. heure = 9 ;
2817. minute = 33 ;
2818. seconde = 15 ;
2819. S = compil_iphone_fct(E, pature, race, individu, ...
2820.     mois, jour, heure, minute, seconde) ;
2821. Compil = [Compil ; S] ;
2822. clearvars -except Compil
2823. E = csvread('SD_0406_2NA_094630.csv',1,0) ;
2824. pature = 2 ;
2825. race = 1 ;
2826. individu = 1 ;
2827. mois = 4 ;
2828. jour = 6 ;
2829. heure = 9 ;
2830. minute = 46 ;
2831. seconde = 30 ;
2832. S = compil_iphone_fct(E, pature, race, individu, ...
2833.     mois, jour, heure, minute, seconde) ;
2834. Compil = [Compil ; S] ;
2835. clearvars -except Compil
2836. E = csvread('SD_0406_2ZA_094457.csv',1,0) ;
2837. pature = 2 ;
2838. race = 2 ;
2839. individu = 1 ;
2840. mois = 4 ;
2841. jour = 6 ;
2842. heure = 9 ;
2843. minute = 44 ;
2844. seconde = 57 ;
2845. S = compil_iphone_fct(E, pature, race, individu, ...
2846.     mois, jour, heure, minute, seconde) ;
2847. Compil = [Compil ; S] ;
2848. clearvars -except Compil
2849. E = csvread('SD_0410_2NpA_065404.csv',1,0) ;
2850. pature = 2 ;
2851. race = 3 ;
2852. individu = 1 ;
2853. mois = 4 ;
2854. jour = 10 ;
2855. heure = 6 ;
2856. minute = 54 ;
2857. seconde = 4 ;
2858. S = compil_iphone_fct(E, pature, race, individu, ...
2859.     mois, jour, heure, minute, seconde) ;
2860. Compil = [Compil ; S] ;
2861. clearvars -except Compil

```

```

2862. E = csvread('SD_0414_1ZB_081459.csv',1,0);
2863. pature = 1;
2864. race = 2;
2865. individu = 2;
2866. mois = 4;
2867. jour = 14;
2868. heure = 8;
2869. minute = 14;
2870. seconde = 59;
2871. S = compil_iphone_fct(E, pature, race, individu, ...
2872.     mois, jour, heure, minute, seconde);
2873. Compil = [Compil; S];
2874. clearvars -except Compil
2875. E = csvread('SD_0418_1NA_170541.csv',1,0);
2876. pature = 1;
2877. race = 1;
2878. individu = 1;
2879. mois = 4;
2880. jour = 18;
2881. heure = 17;
2882. minute = 5;
2883. seconde = 41;
2884. S = compil_iphone_fct(E, pature, race, individu, ...
2885.     mois, jour, heure, minute, seconde);
2886. Compil = [Compil; S];
2887. clearvars -except Compil
2888. E = csvread('SD_0422_1NpB_071441.csv',1,0);
2889. pature = 1;
2890. race = 3;
2891. individu = 2;
2892. mois = 4;
2893. jour = 22;
2894. heure = 7;
2895. minute = 14;
2896. seconde = 41;
2897. S = compil_iphone_fct(E, pature, race, individu, ...
2898.     mois, jour, heure, minute, seconde);
2899. Compil = [Compil; S];
2900. clearvars -except Compil
2901. E = csvread('SD_0422_1ZB_071700.csv',1,0);
2902. pature = 1;
2903. race = 2;
2904. individu = 2;
2905. mois = 4;
2906. jour = 22;
2907. heure = 7;
2908. minute = 17;
2909. seconde = 0;
2910. S = compil_iphone_fct(E, pature, race, individu, ...
2911.     mois, jour, heure, minute, seconde);
2912. Compil = [Compil; S];
2913. clearvars -except Compil

```

```

2914. E = csvread('SD_0426_2ZpA_080615.csv',1,0);
2915. pature = 2;
2916. race = 4;
2917. individu = 1;
2918. mois = 4;
2919. jour = 26;
2920. heure = 8;
2921. minute = 6;
2922. seconde = 15;
2923. S = compil_iphone_fct(E, pature, race, individu, ...
2924.     mois, jour, heure, minute, seconde);
2925. Compil = [Compil; S];
2926. clearvars -except Compil
2927. E = csvread('SD_0430_2ZB_154818.csv',1,0);
2928. pature = 2;
2929. race = 2;
2930. individu = 2;
2931. mois = 4;
2932. jour = 30;
2933. heure = 15;
2934. minute = 48;
2935. seconde = 18;
2936. S = compil_iphone_fct(E, pature, race, individu, ...
2937.     mois, jour, heure, minute, seconde);
2938. Compil = [Compil; S];
2939. clearvars -except Compil
2940. E = csvread('SD_0504_2NB_080741.csv',1,0);
2941. pature = 2;
2942. race = 1;
2943. individu = 2;
2944. mois = 5;
2945. jour = 4;
2946. heure = 8;
2947. minute = 7;
2948. seconde = 41;
2949. S = compil_iphone_fct(E, pature, race, individu, ...
2950.     mois, jour, heure, minute, seconde);
2951. Compil = [Compil; S];
2952. clearvars -except Compil
2953. E = csvread('SD_0506_2ZB_115539.csv',1,0);
2954. pature = 2;
2955. race = 2;
2956. individu = 2;
2957. mois = 5;
2958. jour = 6;
2959. heure = 11;
2960. minute = 55;
2961. seconde = 39;
2962. S = compil_iphone_fct(E, pature, race, individu, ...
2963.     mois, jour, heure, minute, seconde);
2964. Compil = [Compil; S];
2965. clearvars -except Compil

```

```

2966. E = csvread('SD_0508_2NpB_162103.csv',1,0);
2967. pature = 2;
2968. race = 3;
2969. individu = 2;
2970. mois = 5;
2971. jour = 8;
2972. heure = 16;
2973. minute = 21;
2974. seconde = 3;
2975. S = compil_iphone_fct(E, pature, race, individu, ...
2976.     mois, jour, heure, minute, seconde);
2977. Compil = [Compil; S];
2978. clearvars -except Compil
2979. E = csvread('SD_0511_2NpB_112003.csv',1,0);
2980. pature = 2;
2981. race = 3;
2982. individu = 2;
2983. mois = 5;
2984. jour = 11;
2985. heure = 11;
2986. minute = 20;
2987. seconde = 3;
2988. S = compil_iphone_fct(E, pature, race, individu, ...
2989.     mois, jour, heure, minute, seconde);
2990. Compil = [Compil; S];
2991. clearvars -except Compil
2992. %% fin repeter fonction pour chaque fichier
2993. %% ecrire fichier compil
2994. dlmwrite('compil_iphone2.csv',Compil,'precision',8)
2995. %% fichier / bete
2996. for x = 1:2
2997.     for y = 1:4
2998.         for z = 1:2
2999.             Bete = zeros(length(Compil),9);
3000.             l = 1;
3001.             j = 1;
3002.             while l <= length(Compil)
3003.                 if Compil(l,1)==x && Compil(l,2)==y && Compil(l,3)==z
3004.                     Bete(j,:) = Compil(l,:);
3005.                     j = j+1;
3006.                 end
3007.                 l = l+1;
3008.             end
3009.             Bete = Bete(1:j-1,:);
3010.             nom = int2str(Bete(1,1:3));
3011.             nom = strcat('iphone_',nom,'.csv');
3012.             nom = regexprep(nom, ' ');
3013.             dlmwrite(nom,Bete,'precision',8)
3014.             % fichier jour + fichier nuit / bete
3015.             Bete_j = zeros(length(Bete), 9);
3016.             Bete_n = zeros(length(Bete), 9);
3017.             l = 1;

```

```

3018.             j = 1;
3019.             k = 1;
3020.             while l <= length(Bete)
3021.                 if Bete(l,5) >= 6.5 && Bete(l,5) < 18.5
3022.                     Bete_j(j,:) = Bete(l,:);
3023.                     j = j+1;
3024.                 elseif Bete(l,5) < 6.5 || Bete(l,5) >= 18.5
3025.                     Bete_n(k,:) = Bete(l,:);
3026.                     k = k+1;
3027.                 end
3028.                 l = l+1;
3029.             end
3030.             Bete_j = Bete_j(1:j-1,1:7);
3031.             Bete_n = Bete_n(1:k-1,1:7);
3032.             nom_j = int2str(Bete_j(1,1:3));
3033.             nom_j = strcat('iphone_',nom_j,'_j.csv');
3034.             nom_j = regexprep(nom_j, ' ');
3035.             nom_n = int2str(Bete_n(1,1:3));
3036.             nom_n = strcat('iphone_',nom_n,'_n.csv');
3037.             nom_n = regexprep(nom_n, ' ');
3038.             dlmwrite(nom_j,Bete_j,'precision',8)
3039.             dlmwrite(nom_n,Bete_n,'precision',8)
3040.             % fichier grazing + non-grazing / bete
3041.             Bete_g = zeros(length(Bete), 9);
3042.             Bete_ng = zeros(length(Bete), 9);
3043.             l = 1;
3044.             j = 1;
3045.             k = 1;
3046.             while l <= length(Bete)
3047.                 if Bete(l,8) >= 0.5
3048.                     Bete_g(j,:) = Bete(l,:);
3049.                     j = j+1;
3050.                 elseif Bete(l,8) < 0.5
3051.                     Bete_ng(k,:) = Bete(l,:);
3052.                     k = k+1;
3053.                 end
3054.                 l = l+1;
3055.             end
3056.             Bete_g = Bete_g(1:j-1,1:7);
3057.             Bete_ng = Bete_ng(1:k-1,1:7);
3058.             nom_g = int2str(Bete_g(1,1:3));
3059.             nom_g = strcat('iphone_',nom_g,'_g.csv');
3060.             nom_g = regexprep(nom_g, ' ');
3061.             nom_ng = int2str(Bete_ng(1,1:3));
3062.             nom_ng = strcat('iphone_',nom_ng,'_ng.csv');
3063.             nom_ng = regexprep(nom_ng, ' ');
3064.             dlmwrite(nom_g,Bete_g,'precision',8)
3065.             dlmwrite(nom_ng,Bete_ng,'precision',8)
3066.             clearvars -except Compil x y z
3067.         end
3068.     end
3069. end

```

```

3070. clearvars -except Compil
3071. %% 8 fichiers pour variogrammes
3072. % NB : jour, nuit, grazing, non-grazing pour pature 1 et pature 2
3073. % fichiers jour + fichiers nuit
3074. Compil_j1 = zeros(length(Compil), 9) ;
3075. Compil_n1 = zeros(length(Compil), 9) ;
3076. Compil_j2 = zeros(length(Compil), 9) ;
3077. Compil_n2 = zeros(length(Compil), 9) ;
3078. l = 1 ;
3079. j = 1 ;
3080. k = 1 ;
3081. l = 1 ;
3082. m = 1 ;
3083. while l <= length(Compil)
3084. if Compil(l,1) == 1
3085.     if Compil(l,5) >= 6.5 && Compil(l,5) < 18.5
3086.         Compil_j1(j,:) = Compil(l,:) ;
3087.         j = j+1 ;
3088.     elseif Compil(l,5) < 6.5 || Compil(l,5) >= 18.5
3089.         Compil_n1(k,:) = Compil(l,:) ;
3090.         k = k+1 ;
3091.     end
3092. elseif Compil(l,1) == 2
3093.     if Compil(l,5) >= 6.5 && Compil(l,5) < 18.5
3094.         Compil_j2(l,:) = Compil(l,:) ;
3095.         l = l+1 ;
3096.     elseif Compil(l,5) < 6.5 || Compil(l,5) >= 18.5
3097.         Compil_n2(m,:) = Compil(l,:) ;
3098.         m = m+1 ;
3099.     end
3100. end
3101. l = i+1 ;
3102. end
3103. Compil_j1 = Compil_j1(1:j-1,1:7) ;
3104. Compil_n1 = Compil_n1(1:k-1,1:7) ;
3105. Compil_j2 = Compil_j2(1:l-1,1:7) ;
3106. Compil_n2 = Compil_n2(1:m-1,1:7) ;
3107. dlmwrite('compil_j1.csv',Compil_j1,'precision',8)
3108. dlmwrite('compil_n1.csv',Compil_n1,'precision',8)
3109. dlmwrite('compil_j2.csv',Compil_j2,'precision',8)
3110. dlmwrite('compil_n2.csv',Compil_n2,'precision',8)
3111. % fichiers grazing + fichiers non-grazing
3112. Compil_g1 = zeros(length(Compil), 9) ;
3113. Compil_ng1 = zeros(length(Compil), 9) ;
3114. Compil_g2 = zeros(length(Compil), 9) ;
3115. Compil_ng2 = zeros(length(Compil), 9) ;
3116. l = 1 ;
3117. j = 1 ;
3118. k = 1 ;
3119. l = 1 ;
3120. m = 1 ;
3121. while l <= length(Compil)

```

```

3122. if Compil(l,1) == 1
3123.     if Compil(l,8) >= 0.5
3124.         Compil_g1(j,:) = Compil(l,:) ;
3125.         j = j+1 ;
3126.     elseif Compil(l,8) < 0.5
3127.         Compil_ng1(k,:) = Compil(l,:) ;
3128.         k = k+1 ;
3129.     end
3130. elseif Compil(l,1) == 2
3131.     if Compil(l,8) >= 0.5
3132.         Compil_g2(l,:) = Compil(l,:) ;
3133.         l = l+1 ;
3134.     elseif Compil(l,8) < 0.5
3135.         Compil_ng2(m,:) = Compil(l,:) ;
3136.         m = m+1 ;
3137.     end
3138. end
3139. l = i+1 ;
3140. end
3141. Compil_g1 = Compil_g1(1:j-1,1:7) ;
3142. Compil_ng1 = Compil_ng1(1:k-1,1:7) ;
3143. Compil_g2 = Compil_g2(1:l-1,1:7) ;
3144. Compil_ng2 = Compil_ng2(1:m-1,1:7) ;
3145. dlmwrite('compil_g1.csv',Compil_g1,'precision',8)
3146. dlmwrite('compil_ng1.csv',Compil_ng1,'precision',8)
3147. dlmwrite('compil_g2.csv',Compil_g2,'precision',8)
3148. dlmwrite('compil_ng2.csv',Compil_ng2,'precision',8)
3149. clearvars -except Compil

```

Annexe 21 : description du script du traitement statistique de la vitesse des bêtes et de leur accès à l'eau et au sel

1. Le nombre de chiffres significatifs utilisés dans l'environnement MATLAB est augmenté par rapport à la valeur par défaut afin de pouvoir travailler avec des degrés décimaux.
2. Le fichier « csv » contenant l'ensemble des données est lu.
3. *Méthode utilisée pour les données triées en catégories jour et nuit et pour les données triées en catégories grazing et non grazing.*

Le nombre maximal de données disponible pour une bête dans chacune des deux catégories de données est déterminé afin de pouvoir pré-dimensionner les matrices dans les étapes suivantes :

- 3.1. Une matrice à 4 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus, dimension 4 : catégorie de données) est remplie de zéros.
- 3.2. Une boucle est créée afin de lire les données de départ ligne par ligne. A chaque ligne, la cellule correspondant à la bête et aux conditions est incrémentée de 1.
- 3.3. La valeur maximale présente dans la matrice est retenue.
4. Les vitesses sont classées par bête et par catégorie de données :
 - 4.1. Une matrice à 5 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus, dimension 4 : catégorie de données, dimension 5 : données) est remplie de « NaN ».
 - 4.2. Une boucle est créée afin de lire les données de départ ligne par ligne. A chaque ligne, la variable « vitesse » est recopiée dans la première cellule libre (dimension 5) correspondant à la bête et aux conditions (dimensions 1 à 4).
5. Les données relatives à la vitesse sont calculées pour chaque bête :
 - 5.1. Une matrice à 5 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus, dimension 4 : catégorie de données, dimension 5 : vitesse moyenne / comptage des vitesses entre 0 et 0,5 / comptage des vitesses entre 0,5 et 2 / comptage des vitesses supérieures à 2 / comptage des toutes les vitesses) est remplie de 0.
 - 5.2. Les moyennes des données de la matrice résultant de l'étape 4 sont réalisées selon la dimension 5 et est placées en première position dans la dimension 5 de la nouvelle matrice.
 - 5.3. Une boucle est réalisée afin de lire les données de la matrice résultant de l'étape 4 ligne par ligne. A chaque ligne, la position 2, 3 ou 4 de la dimension 5 de la nouvelle matrice est incrémentée de 1 en fonction de la valeur de la vitesse et la position 5 est, dans tous les cas, incrémentée de 1.
 - 5.4. Les positions 2, 3 et 4 de la dimension 5 de la nouvelle matrice sont divisées par la position 5 de la dimension 5 de la nouvelle matrice. Cette dernière position est ensuite supprimée.
6. Les moyennes par pâture, par race et « globale » sont réalisées et stockées dans une matrice.
7. Les écarts-types par pâture, par race et « global » sont réalisés et stockés dans une matrice.
8. L'analyse de la variance est réalisée et les p-values « pâture », « race » et « interaction » sont stockées dans une matrice.
9. Les matrices « moyennes », « écarts-types » et « p-values » sont fusionnées et réarrangées pour former une matrice « statistiques ».

10. Les positions des bêtes sont classées par bête et par catégories de données :
 - 10.1. Une matrice à 6 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus, dimension 4 : catégorie de données, dimension 5 : latitude/longitude, dimension 6 : données) est remplie de « NaN ».
 - 10.2. Une boucle est créée afin de lire les données ligne par ligne. A chaque ligne, les variables « latitude » et « longitude » sont recopiées dans les premières cellules libres (dimension 6) correspondant à la bête et aux conditions (dimensions 1 à 4).
11. Une matrice est remplie manuellement avec les coordonnées géographiques des points ou des extrémités des segments de droite associés aux abreuvoirs, aux flaques et aux bacs à sel.
12. Les distances séparant les bêtes de ces éléments sont calculées :
 - 12.1. Une matrice à 6 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus, dimension 4 : catégorie de données, dimension 5 : éléments, dimension 6 : données) est remplie de « NaN ».
 - 12.2. Toutes les coordonnées sont transformées de degrés décimaux en mètres.
 - 12.3. Les distances sont calculées :
 - Pour les 4 points : Une boucle est créée et le théorème de Pythagore est appliqué à chaque ligne pour chaque point. Les distances obtenues sont recopiées dans les premières cellules libres (dimension 6) correspondant à la bête et aux conditions (dimensions 1 à 4).
 - Pour les 3 segments de droite : Une boucle est créée et, à chaque ligne et pour chacun des segments, les distances par rapport aux deux extrémités sont calculées (théorème de Pythagore), ainsi que l'équation de la droite passant par ces deux points et de celle de la droite perpendiculaire à cette droite et passant par la position de la bête. Le point d'intersection de ces deux droites est déterminé et la distance par rapport à celui-ci est calculée (théorème de Pythagore). Si ce dernier point appartient au segment de droite, cette distance est retenue et, dans le cas contraire, la plus petite des distances par rapport aux extrémités est retenue. Les distances obtenues sont recopiées dans les premières cellules libres (dimension 6) correspondant à la bête et aux conditions (dimensions 1 à 4).
13. Les données relatives aux bacs à sel et points d'eau sont calculées pour chaque bête :
 - 13.1. Une matrice à 5 dimensions (dimension 1 : pâtures, dimension 2 : races, dimension 3 : individus, dimension 4 : catégorie de données, dimension 5 : comptage des positions à moins de 10 m d'un bac à sel, comptage des positions à moins de 10 m d'un point d'eau, comptage des positions) est remplie de 0.
 - 13.2. Une boucle est réalisée afin de lire les données de la matrice résultant de l'étape 11 ligne par ligne. A chaque ligne, la position 1, la position 2 ou aucune de ces deux positions de la dimension 5 de la nouvelle matrice est incrémentée de 1 en fonction de la valeur des distances par rapports aux 7 éléments. La position 3 de la dimension 5 est, dans tous les cas, incrémentée de 1.
 - 13.3. Les positions 1 et 2 de la dimension 5 de la nouvelle matrice sont divisées par la position 3 de la dimension 5 de la nouvelle matrice. Cette dernière position est ensuite supprimée.
14. Les moyennes par pâture, par race et « globale » sont réalisées et stockées dans une matrice.

15. Les écarts-types par pâture, par race et « global » sont réalisés et stockés dans une matrice.
16. L'analyse de la variance est réalisée et les p-values « pâture », « race » et « interaction » sont stockées dans une matrice.
17. Les matrices « moyennes », « écarts-types » et « p-values » sont fusionnées et réarrangées pour former une matrice « statistiques ».
18. Les matrices « statistiques » provenant des étapes 9 et 17 sont compilées en une matrice et celle-ci est écrite au format « xls ».
19. Un fichier « xls » comprenant le nom des lignes dans le bon ordre est écrit afin d'éviter toute erreur lors de la mise en forme de la matrice de l'étape 18.

Annexe 22 : script du traitement statistique

de la vitesse des bêtes et de leur accès à l'eau et au sel

```
3150.  %%% vitesses et acces a l eau et au sel
3151.  %% NB :
3152.  % vitesse en m/h
3153.  % temps en fraction
3154.  % distance en m
3155.  %% augmenter chiffres significatifs
3156.  format long g
3157.  %% lecture fichier
3158.  vitesse = csvread('compil_iphone2.csv') ;
3159.  %% determination de la longueur de la matrice par bete
3160.  %% par jour / nuit
3161.  longueur = zeros(2,4,2,2) ;
3162.  for x = 1:2
3163.  for y = 1:4
3164.  for z = 1:2
3165.  i=1 ;
3166.  while i <= length(vitesse)
3167.  if vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3168.      vitesse(i,5) >= 6.5 && vitesse(i,5) < 18.5
3169.  longueur(x,y,z,1) = longueur(x,y,z,1) + 1 ;
3170.  elseif vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3171.      (vitesse(i,5) < 6.5 || vitesse(i,5) >= 18.5)
3172.  longueur(x,y,z,2) = longueur(x,y,z,2) + 1 ;
3173.  end
3174.  i=i+1 ;
3175.  end
3176.  end
3177.  end
3178.  end
3179.  longueur = max(max(max(max(longueur)))) ;
3180.  %% determination de la longueur de la matrice par bete
3181.  %% par grazing / non grazing
3182.  longueurGR = zeros(2,4,2,2) ;
3183.  for x = 1:2
3184.  for y = 1:4
3185.  for z = 1:2
3186.  i=1 ;
3187.  while i <= length(vitesse)
3188.  if vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3189.      vitesse(i,8) > 0.5
3190.  longueurGR(x,y,z,1) = longueurGR(x,y,z,1) + 1 ;
3191.  elseif vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3192.      vitesse(i,8) <= 0.5
3193.  longueurGR(x,y,z,2) = longueurGR(x,y,z,2) + 1 ;
3194.  end
3195.  i=i+1 ;
3196.  end
```

```
3197.  end
3198.  end
3199.  end
3200.  longueurGR = max(max(max(max(longueurGR)))) ;
3201.  %% classement des vitesses par bete
3202.  %% par jour/nuit
3203.  vitesse2 = NaN(2,4,2,2,longueur) ;
3204.  for x = 1:2
3205.  for y = 1:4
3206.  for z = 1:2
3207.  i=1 ;
3208.  j=1 ;
3209.  k=1 ;
3210.  while i <= length(vitesse)
3211.  if vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3212.      vitesse(i,5) >= 6.5 && vitesse(i,5) < 18.5
3213.  vitesse2(x,y,z,1,j) = vitesse(i,9) ;
3214.  j=j+1 ;
3215.  elseif vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3216.      (vitesse(i,5) < 6.5 || vitesse(i,5) >= 18.5)
3217.  vitesse2(x,y,z,2,k) = vitesse(i,9) ;
3218.  k=k+1 ;
3219.  end
3220.  i=i+1 ;
3221.  end
3222.  end
3223.  end
3224.  end
3225.  clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR
3226.  %% classement des vitesses par bete
3227.  %% par grazing / non grazing
3228.  vitesse2GR = NaN(2,4,2,2,longueurGR) ;
3229.  for x = 1:2
3230.  for y = 1:4
3231.  for z = 1:2
3232.  i=1 ;
3233.  j=1 ;
3234.  k=1 ;
3235.  while i <= length(vitesse)
3236.  if vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3237.      vitesse(i,8) > 0.5
3238.  vitesse2GR(x,y,z,1,j) = vitesse(i,9) ;
3239.  j=j+1 ;
3240.  elseif vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3241.      vitesse(i,8) <= 0.5
3242.  vitesse2GR(x,y,z,2,k) = vitesse(i,9) ;
3243.  k=k+1 ;
3244.  end
3245.  i=i+1 ;
3246.  end
3247.  end
3248.  end
```

```

3249. end
3250. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR
3251. %% vitesses
3252. %% + temps entre 0 et 0.5 m/s
3253. %% + temps entre 0.5 et 2 m/s
3254. %% + temps > 2 m/s
3255. %% pour le jour et pour la nuit
3256. % moyenne par bete (vitesses)
3257. vitesse3 = zeros(2,4,2,2,5) ;
3258. for c = 1:2
3259. vitesse3(:,:,,c,1) = 3600*nanmean(vitesse2(:,:,,c,:),5) ;
3260. for x = 1:2
3261. for y = 1:4
3262. for z = 1:2
3263. % comptage par bete (temps)
3264. i=1 ;
3265. while i <= longueur
3266. if vitesse2(x,y,z,c,i) >= 0 && vitesse2(x,y,z,c,i) < 0.5
3267. vitesse3(x,y,z,c,2) = vitesse3(x,y,z,c,2) + 1 ;
3268. elseif vitesse2(x,y,z,c,i) >= 0.5 && vitesse2(x,y,z,c,i) < 2
3269. vitesse3(x,y,z,c,3) = vitesse3(x,y,z,c,3) + 1 ;
3270. elseif vitesse2(x,y,z,c,i) >= 2
3271. vitesse3(x,y,z,c,4) = vitesse3(x,y,z,c,4) + 1 ;
3272. end
3273. if isnan(vitesse2(x,y,z,c,i)) == 0
3274. vitesse3(x,y,z,c,5) = vitesse3(x,y,z,c,5) + 1 ;
3275. end
3276. i=i+1 ;
3277. end
3278. vitesse3(x,y,z,c,2:4) = vitesse3(x,y,z,c,2:4)/vitesse3(x,y,z,c,5) ;
3279. end
3280. end
3281. end
3282. end
3283. vitesse3 = vitesse3(:,:,,1:4) ;
3284. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR ...
3285. vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR
3286. % moyenne
3287. vitesse_global = zeros(8,1) ;
3288. vitesse_pature = zeros(8,2) ;
3289. vitesse_race = zeros(8,4) ;
3290. for i = 1:2
3291. for j = 1:4
3292. vitesse_global(((j-1)*2+i,:,:) = mean(mean(mean(vitesse3(:,:,,i,j)))) ;
3293. vitesse_race(((j-1)*2+i,:,:) = mean(mean(vitesse3(:,:,,i,j),3),1) ;
3294. vitesse_pature(((j-1)*2+i,:,:) = (mean(mean(vitesse3(:,:,,i,j),3),2))' ;
3295. end
3296. end
3297. vitesse_moy = [vitesse_global vitesse_pature vitesse_race] ;
3298. % ecart type global
3299. vitesse_global_ET = zeros(8,1) ;
3300. vitesse_pature_ET = zeros(8,2) ;

```

```

3301. vitesse_race_ET = zeros(8,4) ;
3302. for i = 1:2
3303. for j = 1:4
3304. vitesse_global_M = ones(2,4,2)*vitesse_global(((j-1)*2+i,:,:) ;
3305. vitesse_global_DM = vitesse3(:,:,,i,j) - vitesse_global_M ;
3306. vitesse_global_DM2 = vitesse_global_DM.*vitesse_global_DM ;
3307. vitesse_global_V = mean(mean(mean(vitesse_global_DM2))) ;
3308. vitesse_global_ET(((j-1)*2+i,:,:) = sqrt(vitesse_global_V) ;
3309. end
3310. end
3311. % ecart type race
3312. for i = 1:2
3313. for j = 1:4
3314. vitesse_race_M = ones(2,4,2) ;
3315. for i_race1 = 1:2
3316. for i_race2 = 1:2
3317. vitesse_race_M(i_race1,:i_race2) = vitesse_race(((j-1)*2+i,:,:) ;
3318. end
3319. end
3320. vitesse_race_DM = vitesse3(:,:,,i,j) - vitesse_race_M ;
3321. vitesse_race_DM2 = vitesse_race_DM.*vitesse_race_DM ;
3322. vitesse_race_V = mean(mean(vitesse_race_DM2,3),1) ;
3323. vitesse_race_ET(((j-1)*2+i,:,:) = sqrt(vitesse_race_V) ;
3324. end
3325. end
3326. % ecart type pature
3327. for i = 1:2
3328. for j = 1:4
3329. vitesse_pature_M = ones(2,4,2) ;
3330. for i_pature1 = 1:4
3331. for i_pature2 = 1:2
3332. vitesse_pature_M(:,i_pature1,i_pature2) = vitesse_pature(((j-1)*2+i,:,:) ;
3333. end
3334. end
3335. vitesse_pature_DM = vitesse3(:,:,,i,j) - vitesse_pature_M ;
3336. vitesse_pature_DM2 = vitesse_pature_DM.*vitesse_pature_DM ;
3337. vitesse_pature_V = mean(mean(vitesse_pature_DM2,3),2) ;
3338. vitesse_pature_ET(((j-1)*2+i,:,:) = (sqrt(vitesse_pature_V))' ;
3339. end
3340. end
3341. % ecart type fusion
3342. vitesse_ET = [vitesse_global_ET vitesse_pature_ET vitesse_race_ET] ;
3343. % ANOVA
3344. vitesse_AV = zeros(8,16) ;
3345. vitesse_pvalue = zeros(8,3) ;
3346. for i = 1:2
3347. for j = 1:4
3348. vitesse_AV(((j-1)*2+i,:,:) = [vitesse3(1,:i,j) vitesse3(1,:i,j) ...
3349. vitesse3(2,:i,j) vitesse3(1,:i,j)] ;
3350. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
3351. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
3352. vitesse_pvalue(((j-1)*2+i,:,:) = ...

```

```

3353.         fliplr((anovan(vitesse_AV(((j-1)*2+i),:),{f1,f2},'full')));
3354.     close all force
3355. end
3356. end
3357. % fusion
3358. vitesse_stat = [vitesse_moy vitesse_ET vitesse_pvalue];
3359. vitesse_stat(3:8,1:14) = vitesse_stat(3:8,1:14)*100;
3360. C = vitesse_stat;
3361. vitesse_stat_final = [C(:,1) C(:,8) C(:,15) C(:,2) C(:,9) C(:,3) ...
3362.     C(:,10) C(:,16) C(:,4) C(:,11) C(:,5) C(:,12) C(:,6) C(:,13) ...
3363.     C(:,7) C(:,14) C(:,17)];
3364. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR ...
3365.     vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR
3366. %% vitesses
3367. %% + temps entre 0 et 0.5 m/s
3368. %% + temps entre 0.5 et 2 m/s
3369. %% + temps > 2 m/s
3370. %% pour grazing et non grazing
3371. % moyenne par bete (vitesses)
3372. vitesse3GR = zeros(2,4,2,5);
3373. for c = 1:2
3374.     vitesse3GR(:,:,,c,1) = 3600*nanmean(vitesse2GR(:,:,,c,:),5);
3375. for x = 1:2
3376.     for y = 1:4
3377.         for z = 1:2
3378.             % comptage par bete (temps)
3379.             i=1;
3380.             while i <= longueurGR
3381.                 if vitesse2GR(x,y,z,c,i) >= 0 && vitesse2GR(x,y,z,c,i) < 0.5
3382.                     vitesse3GR(x,y,z,c,2) = vitesse3GR(x,y,z,c,2) + 1;
3383.                 elseif vitesse2GR(x,y,z,c,i) >= 0.5 && vitesse2GR(x,y,z,c,i) < 2
3384.                     vitesse3GR(x,y,z,c,3) = vitesse3GR(x,y,z,c,3) + 1;
3385.                 elseif vitesse2GR(x,y,z,c,i) >= 2
3386.                     vitesse3GR(x,y,z,c,4) = vitesse3GR(x,y,z,c,4) + 1;
3387.                 end
3388.                 if isnan(vitesse2GR(x,y,z,c,i)) == 0
3389.                     vitesse3GR(x,y,z,c,5) = vitesse3GR(x,y,z,c,5) + 1;
3390.                 end
3391.                 i=i+1;
3392.             end
3393.             vitesse3GR(x,y,z,c,2:4) = vitesse3GR(x,y,z,c,2:4)/vitesse3GR(x,y,z,c,5);
3394.         end
3395.     end
3396. end
3397. end
3398. vitesse3GR = vitesse3GR(:,:,,1:4);
3399. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR ...
3400.     vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR
3401. % moyenne
3402. vitesseGR_global = zeros(8,1);
3403. vitesseGR_pature = zeros(8,2);
3404. vitesseGR_race = zeros(8,4);

```

```

3405. for i = 1:2
3406.     for j = 1:4
3407.         vitesseGR_global(((j-1)*2+i,:) = mean(mean(mean(vitesse3GR(:,:,,i,j))));
3408.         vitesseGR_race(((j-1)*2+i,:) = mean(mean(vitesse3GR(:,:,,i,j),3),1);
3409.         vitesseGR_pature(((j-1)*2+i,:) = (mean(mean(vitesse3GR(:,:,,i,j),3),2))';
3410.     end
3411. end
3412. vitesseGR_moy = [vitesseGR_global vitesseGR_pature vitesseGR_race];
3413. % ecart type global
3414. vitesseGR_global_ET = zeros(8,1);
3415. vitesseGR_pature_ET = zeros(8,2);
3416. vitesseGR_race_ET = zeros(8,4);
3417. for i = 1:2
3418.     for j = 1:4
3419.         vitesseGR_global_M = ones(2,4,2)*vitesseGR_global(((j-1)*2+i,:);
3420.         vitesseGR_global_DM = vitesse3GR(:,:,,i,j) - vitesseGR_global_M;
3421.         vitesseGR_global_DM2 = vitesseGR_global_DM.*vitesseGR_global_DM;
3422.         vitesseGR_global_V = mean(mean(mean(vitesseGR_global_DM2)));
3423.         vitesseGR_global_ET(((j-1)*2+i,:) = sqrt(vitesseGR_global_V);
3424.     end
3425. end
3426. % ecart type race
3427. for i = 1:2
3428.     for j = 1:4
3429.         vitesseGR_race_M = ones(2,4,2);
3430.         for i_race1 = 1:2
3431.             for i_race2 = 1:2
3432.                 vitesseGR_race_M(i_race1,:,i_race2) = vitesseGR_race(((j-1)*2+i,:);
3433.             end
3434.         end
3435.         vitesseGR_race_DM = vitesse3GR(:,:,,i,j) - vitesseGR_race_M;
3436.         vitesseGR_race_DM2 = vitesseGR_race_DM.*vitesseGR_race_DM;
3437.         vitesseGR_race_V = mean(mean(vitesseGR_race_DM2,3),1);
3438.         vitesseGR_race_ET(((j-1)*2+i,:) = sqrt(vitesseGR_race_V);
3439.     end
3440. end
3441. % ecart type pature
3442. for i = 1:2
3443.     for j = 1:4
3444.         vitesseGR_pature_M = ones(2,4,2);
3445.         for i_pature1 = 1:4
3446.             for i_pature2 = 1:2
3447.                 vitesseGR_pature_M(:,i_pature1,i_pature2) = vitesseGR_pature(((j-1)*2+i,:);
3448.             end
3449.         end
3450.         vitesseGR_pature_DM = vitesse3GR(:,:,,i,j) - vitesseGR_pature_M;
3451.         vitesseGR_pature_DM2 = vitesseGR_pature_DM.*vitesseGR_pature_DM;
3452.         vitesseGR_pature_V = mean(mean(vitesseGR_pature_DM2,3),2);
3453.         vitesseGR_pature_ET(((j-1)*2+i,:) = (sqrt(vitesseGR_pature_V))';
3454.     end
3455. end
3456. % ecart type fusion

```

```

3457. vitesseGR_ET = [vitesseGR_global_ET vitesseGR_pature_ET vitesseGR_race_ET] ;3509.
3458. % ANOVA
3459. vitesseGR_AV = zeros(8,16) ;
3460. vitesseGR_pvalue = zeros(8,3) ;
3461. for i = 1:2
3462. for j = 1:4
3463. vitesseGR_AV(((j-1)*2+i,:)) = [vitesse3GR(1,,:,1,i,j) vitesse3GR(1,,:,2,i,j) ...
3464. vitesse3GR(2,,:,1,i,j) vitesse3GR(1,,:,2,i,j)] ;
3465. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4] ;
3466. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2] ;
3467. vitesseGR_pvalue(((j-1)*2+i,:)) = ...
3468. fliplr(anovan(vitesseGR_AV(((j-1)*2+i,:)),{f1,f2},'full')) ;
3469. close all force
3470. end
3471. end
3472. % fusion
3473. vitesseGR_stat = [vitesseGR_moy vitesseGR_ET vitesseGR_pvalue] ;
3474. vitesseGR_stat(3:8,1:14) = vitesseGR_stat(3:8,1:14)*100 ;
3475. CGR = vitesseGR_stat ;
3476. vitesse_stat_final_GR = [CGR(:,1) CGR(:,8) CGR(:,15) CGR(:,2) CGR(:,9) ...
3477. CGR(:,3) CGR(:,10) CGR(:,16) CGR(:,4) CGR(:,11) CGR(:,5) CGR(:,12) ...
3478. CGR(:,6) CGR(:,13) CGR(:,7) CGR(:,14) CGR(:,17)] ;
3479. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR ...
3480. vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR
3481. %% classement des positions par bete et par jour/nuit
3482. position = NaN(2,4,2,2,2,longueur) ;
3483. for x = 1:2
3484. for y = 1:4
3485. for z = 1:2
3486. i=1 ;
3487. j=1 ;
3488. k=1 ;
3489. while i <= length(vitesse)
3490. if vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3491. vitesse(i,5) >= 6.5 && vitesse(i,5) < 18.5
3492. position(x,y,z,1,1,j) = vitesse(i,6) ;
3493. position(x,y,z,1,2,j) = vitesse(i,7) ;
3494. j=j+1 ;
3495. elseif vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3496. (vitesse(i,5) < 6.5 || vitesse(i,5) >= 18.5)
3497. position(x,y,z,2,1,k) = vitesse(i,6) ;
3498. position(x,y,z,2,2,k) = vitesse(i,7) ;
3499. k=k+1 ;
3500. end
3501. i=i+1 ;
3502. end
3503. end
3504. end
3505. end
3506. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR ...
3507. vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR ...
3508. position positionGR distance distanceGR ...
sel_eau sel_eauGR sel_eau_stat_final sel_eau_stat_final_GR
3510. %% classement des positions par bete et par grazing / non grazing
3511. positionGR = NaN(2,4,2,2,2,longueurGR) ;
3512. for x = 1:2
3513. for y = 1:4
3514. for z = 1:2
3515. i=1 ;
3516. j=1 ;
3517. k=1 ;
3518. while i <= length(vitesse)
3519. if vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3520. vitesse(i,8) > 0.5
3521. positionGR(x,y,z,1,1,j) = vitesse(i,6) ;
3522. positionGR(x,y,z,1,2,j) = vitesse(i,7) ;
3523. j=j+1 ;
3524. elseif vitesse(i,1)==x && vitesse(i,2)==y && vitesse(i,3)==z && ...
3525. vitesse(i,8) <= 0.5
3526. positionGR(x,y,z,2,1,k) = vitesse(i,6) ;
3527. positionGR(x,y,z,2,2,k) = vitesse(i,7) ;
3528. k=k+1 ;
3529. end
3530. i=i+1 ;
3531. end
3532. end
3533. end
3534. end
3535. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR ...
3536. vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR ...
3537. position positionGR distance distanceGR ...
3538. sel_eau sel_eauGR sel_eau_stat_final sel_eau_stat_final_GR
3539. %% calcul des distances par rapport au sel et a l'eau
3540. %% pour jour et nuit
3541. % NB : 4 points + 3 couples de points (3 segments de droite)
3542. % NB : 7 distances / bete * j/n * longueur
3543. % NB : ! 1ere valeur = lat = y et 2e valeur = long = x !
3544. distance = NaN(2,4,2,2,7,longueur) ;
3545. points = [-3.378550 -3.378155 -3.378546 -3.379165 -3.378310 ...
3546. -3.378336 -3.379106 -3.379031 -3.377952 -3.377826
3547. 11.521532 11.519432 11.519515 11.521177 11.520406 11.520396 ...
3548. 11.522260 11.522177 11.520936 11.520808] ;
3549. % transformation des degres en m pour le calcul des distance
3550. position(:,:,:,1,:) = ...
3551. (position(:,:,:,1,:) + 3.380525) * 110578.160 ;
3552. points(:,1) = (points(:,1) + 3.380525) * 110578.160 ;
3553. position(:,:,:,2,:) = ...
3554. (position(:,:,:,2,:) - 11.518189) * 111127.324 ;
3555. points(:,2) = (points(:,2) - 11.518189) * 111127.324 ;
3556. % distance % sel 1 (point)
3557. for x = 1:2
3558. for y = 1:4
3559. for z = 1:2
3560. for c = 1:2

```

```

3561. i=1 ;
3562. while i <= longueur
3563. distance(x,y,z,c,1,i) = sqrt((points(1,1)- ...
3564.         position(x,y,z,c,1,i))^2+(points(1,2)- ...
3565.         position(x,y,z,c,2,i))^2) ;
3566. i=i+1 ;
3567. end
3568. end
3569. end
3570. end
3571. end
3572. % distance % sel 2 (point)
3573. for x = 1:2
3574. for y = 1:4
3575. for z = 1:2
3576. for c = 1:2
3577. i=1 ;
3578. while i <= longueur
3579. distance(x,y,z,c,2,i) = sqrt((points(2,1)- ...
3580.         position(x,y,z,c,1,i))^2+(points(2,2)- ...
3581.         position(x,y,z,c,2,i))^2) ;
3582. i=i+1 ;
3583. end
3584. end
3585. end
3586. end
3587. end
3588. % distance % sel 3 (point)
3589. for x = 1:2
3590. for y = 1:4
3591. for z = 1:2
3592. for c = 1:2
3593. i=1 ;
3594. while i <= longueur
3595. distance(x,y,z,c,3,i) = sqrt((points(3,1)- ...
3596.         position(x,y,z,c,1,i))^2+(points(3,2)- ...
3597.         position(x,y,z,c,2,i))^2) ;
3598. i=i+1 ;
3599. end
3600. end
3601. end
3602. end
3603. end
3604. % distance % eau 1 (point)
3605. for x = 1:2
3606. for y = 1:4
3607. for z = 1:2
3608. for c = 1:2
3609. i=1 ;
3610. while i <= longueur
3611. distance(x,y,z,c,4,i) = sqrt((points(4,1)- ...
3612.         position(x,y,z,c,1,i))^2+(points(4,2)- ...

```

```

3613.         position(x,y,z,c,2,i))^2) ;
3614. i=i+1 ;
3615. end
3616. end
3617. end
3618. end
3619. end
3620. % distance % eau 2 (segment de droite)
3621. % NB : d1 = distance position - point 1
3622. % NB : d2 = distance position - point 2
3623. % NB : e1 = eq droite passant par points 1 et 2
3624. % NB : e2 = eq droite passant par position
3625. % et perpendiculaire a droite passant par points 1 et 2
3626. % NB : p = point d intersection des deux droites
3627. % NB : d3 = distance position - point d intersection
3628. for x = 1:2
3629. for y = 1:4
3630. for z = 1:2
3631. for c = 1:2
3632. i=1 ;
3633. while i <= longueur
3634. d1 = sqrt((points(5,1)-position(x,y,z,c,1,i))^2+ ...
3635.         (points(5,2)- position(x,y,z,c,2,i))^2) ;
3636. d2 = sqrt((points(6,1)-position(x,y,z,c,1,i))^2+ ...
3637.         (points(6,2)-position(x,y,z,c,2,i))^2) ;
3638. e1 = [((points(6,1)-points(5,1))/(points(6,2)-points(5,2))) ...
3639.         (points(5,1)-points(5,2)*((points(6,1)-points(5,1))/ ...
3640.         (points(6,2)-points(5,2))))] ;
3641. e2 = [(-1/e1(1)) ...
3642.         (position(x,y,z,c,1,i)-position(x,y,z,c,2,i)*(-1/e1(1)))] ;
3643. p = [(e1(1)*((e2(2)-e1(2))/(e1(1)-e2(1)))+e1(2)) ...
3644.         ((e2(2)-e1(2))/(e1(1)-e2(1)))] ;
3645. d3 = sqrt((p(1)-position(x,y,z,c,1,i))^2+ ...
3646.         (p(2)-position(x,y,z,c,2,i))^2) ;
3647. if (p(1)<max([points(5,1) points(6,1)]) && ...
3648.         p(1)>min([points(5,1) points(6,1)])) && ...
3649.         (p(2)<max([points(5,2) points(6,2)]) && ...
3650.         p(2)>min([points(5,2) points(6,2)]))
3651. distance(x,y,z,c,5,i) = d3 ;
3652. else
3653. distance(x,y,z,c,5,i) = min([d1 d2]) ;
3654. end
3655. i=i+1 ;
3656. end
3657. end
3658. end
3659. end
3660. end
3661. % distance eau 3 (segment de droite)
3662. for x = 1:2
3663. for y = 1:4
3664. for z = 1:2

```

```

3665. for c = 1:2
3666. i=1 ;
3667. while i <= longueur
3668. d1 = sqrt((points(7,1)-position(x,y,z,c,1,i))^2 ...
3669.         +(points(7,2)-position(x,y,z,c,2,i))^2) ;
3670. d2 = sqrt((points(8,1)-position(x,y,z,c,1,i))^2 ...
3671.         +(points(8,2)-position(x,y,z,c,2,i))^2) ;
3672. e1 = [((points(8,1)-points(7,1))/(points(8,2)-points(7,2))) ...
3673.        (points(7,1)-points(7,2))*((points(8,1)-points(7,1))/ ...
3674.        (points(8,2)-points(7,2)))] ;
3675. e2 = [(-1/e1(1)) ...
3676.        (position(x,y,z,c,1,i)-position(x,y,z,c,2,i))*(-1/e1(1)))] ;
3677. p = [(e1(1))*((e2(2)-e1(2))/(e1(1)-e2(1)))+e1(2)] ...
3678.        ((e2(2)-e1(2))/(e1(1)-e2(1)))] ;
3679. d3 = sqrt((p(1)-position(x,y,z,c,1,i))^2+ ...
3680.         (p(2)-position(x,y,z,c,2,i))^2) ;
3681. if (p(1)<max([points(7,1) points(8,1)]) && ...
3682.     p(1)>min([points(7,1) points(8,1)])) && ...
3683.     (p(2)<max([points(7,2) points(8,2)]) && ...
3684.     p(2)>min([points(7,2) points(8,2)]))
3685. distance(x,y,z,c,6,i) = d3 ;
3686. else
3687. distance(x,y,z,c,6,i) = min([d1 d2]) ;
3688. end
3689. i=i+1 ;
3690. end
3691. end
3692. end
3693. end
3694. end
3695. % distance eau 4 (segment de droite)
3696. for x = 1:2
3697. for y = 1:4
3698. for z = 1:2
3699. for c = 1:2
3700. i=1 ;
3701. while i <= longueur
3702. d1 = sqrt((points(9,1)-position(x,y,z,c,1,i))^2 ...
3703.         +(points(9,2)-position(x,y,z,c,2,i))^2) ;
3704. d2 = sqrt((points(10,1)-position(x,y,z,c,1,i))^2 ...
3705.         +(points(10,2)-position(x,y,z,c,2,i))^2) ;
3706. e1 = [((points(10,1)-points(9,1))/(points(10,2)-points(9,2))) ...
3707.        (points(9,1)-points(9,2))*((points(10,1)-points(9,1))/ ...
3708.        (points(10,2)-points(9,2)))] ;
3709. e2 = [(-1/e1(1)) ...
3710.        (position(x,y,z,c,1,i)-position(x,y,z,c,2,i))*(-1/e1(1)))] ;
3711. p = [(e1(1))*((e2(2)-e1(2))/(e1(1)-e2(1)))+e1(2)] ...
3712.        ((e2(2)-e1(2))/(e1(1)-e2(1)))] ;
3713. d3 = sqrt((p(1)-position(x,y,z,c,1,i))^2 ...
3714.         +(p(2)-position(x,y,z,c,2,i))^2) ;
3715. if (p(1)<max([points(9,1) points(10,1)]) && ...
3716.     p(1)>min([points(9,1) points(10,1)])) && ...

```

```

3717.         (p(2)<max([points(9,2) points(10,2)]) && ...
3718.         p(2)>min([points(9,2) points(10,2)]))
3719. distance(x,y,z,c,7,i) = d3 ;
3720. else
3721. distance(x,y,z,c,7,i) = min([d1 d2]) ;
3722. end
3723. i=i+1 ;
3724. end
3725. end
3726. end
3727. end
3728. end
3729. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR ...
3730.         vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR ...
3731.         position positionGR distance distanceGR ...
3732.         sel_eau sel_eauGR sel_eau_stat_final sel_eau_stat_final_GR
3733. %% calcul des distances par rapport au sel et a l'eau
3734. %% pour grazing et non grazing
3735. % NB : 4 points + 3 couples de points (3 segments de droite)
3736. % NB : 7 distances / bete * g/ng * longueur
3737. % NB : ! 1ere valeur = lat = y et 2e valeur = long = x !
3738. distanceGR = NaN(2,4,2,2,7,longueurGR) ;
3739. points = [-3.378550 -3.378155 -3.378546 -3.379165 -3.378310 ...
3740.          -3.378336 -3.379106 -3.379031 -3.377952 -3.377826
3741.          11.521532 11.519432 11.519515 11.521177 11.520406 11.520396 ...
3742.          11.522260 11.522177 11.520936 11.520808] ;
3743. % transformation des degres en m pour le calcul des distance
3744. positionGR(:,:,,1,:) = ...
3745.         (positionGR(:,:,,1,:) + 3.380525) * 110578.160 ;
3746. points(:,1) = (points(:,1) + 3.380525) * 110578.160 ;
3747. positionGR(:,:,,2,:) = ...
3748.         (positionGR(:,:,,2,:) - 11.518189) * 111127.324 ;
3749. points(:,2) = (points(:,2) - 11.518189) * 111127.324 ;
3750. % distance % sel 1 (point)
3751. for x = 1:2
3752. for y = 1:4
3753. for z = 1:2
3754. for c = 1:2
3755. i=1 ;
3756. while i <= longueurGR
3757. distanceGR(x,y,z,c,1,i) = sqrt((points(1,1)- ...
3758.         positionGR(x,y,z,c,1,i))^2+(points(1,2)- ...
3759.         positionGR(x,y,z,c,2,i))^2) ;
3760. i=i+1 ;
3761. end
3762. end
3763. end
3764. end
3765. end
3766. % distance % sel 2 (point)
3767. for x = 1:2
3768. for y = 1:4

```

```

3769. for z = 1:2
3770. for c = 1:2
3771. i=1 ;
3772. while i <= longueurGR
3773. distanceGR(x,y,z,c,2,i) = sqrt((points(2,1)- ...
3774. positionGR(x,y,z,c,1,i))^2+(points(2,2)- ...
3775. positionGR(x,y,z,c,2,i))^2) ;
3776. i=i+1 ;
3777. end
3778. end
3779. end
3780. end
3781. end
3782. % distance % sel 3 (point)
3783. for x = 1:2
3784. for y = 1:4
3785. for z = 1:2
3786. for c = 1:2
3787. i=1 ;
3788. while i <= longueurGR
3789. distanceGR(x,y,z,c,3,i) = sqrt((points(3,1)- ...
3790. positionGR(x,y,z,c,1,i))^2+(points(3,2)- ...
3791. positionGR(x,y,z,c,2,i))^2) ;
3792. i=i+1 ;
3793. end
3794. end
3795. end
3796. end
3797. end
3798. % distance % eau 1 (point)
3799. for x = 1:2
3800. for y = 1:4
3801. for z = 1:2
3802. for c = 1:2
3803. i=1 ;
3804. while i <= longueurGR
3805. distanceGR(x,y,z,c,4,i) = sqrt((points(4,1)- ...
3806. positionGR(x,y,z,c,1,i))^2+(points(4,2)- ...
3807. positionGR(x,y,z,c,2,i))^2) ;
3808. i=i+1 ;
3809. end
3810. end
3811. end
3812. end
3813. end
3814. % distance % eau 2 (segment de droite)
3815. % NB : d1 = distance position - point 1
3816. % NB : d2 = distance position - point 2
3817. % NB : e1 = eq droite passant par points 1 et 2
3818. % NB : e2 = eq droite passant par position
3819. % et perpendiculaire a droite passant par points 1 et 2
3820. % NB : p = point d intersection des deux droites

```

```

3821. % NB : d3 = distance position - point d intersection
3822. for x = 1:2
3823. for y = 1:4
3824. for z = 1:2
3825. for c = 1:2
3826. i=1 ;
3827. while i <= longueurGR
3828. d1 = sqrt((points(5,1)-positionGR(x,y,z,c,1,i))^2+ ...
3829. (points(5,2)- positionGR(x,y,z,c,2,i))^2) ;
3830. d2 = sqrt((points(6,1)-positionGR(x,y,z,c,1,i))^2+ ...
3831. (points(6,2)-positionGR(x,y,z,c,2,i))^2) ;
3832. e1 = [((points(6,1)-points(5,1))/(points(6,2)-points(5,2))) ...
3833. (points(5,1)-points(5,2))*((points(6,1)-points(5,1))/ ...
3834. (points(6,2)-points(5,2)))] ;
3835. e2 = [(-1/e1(1)) ...
3836. (positionGR(x,y,z,c,1,i)-positionGR(x,y,z,c,2,i)*(-1/e1(1)))] ;
3837. p = [(e1(1)*((e2(2)-e1(2))/(e1(1)-e2(1)))+e1(2)) ...
3838. ((e2(2)-e1(2))/(e1(1)-e2(1)))] ;
3839. d3 = sqrt((p(1)-positionGR(x,y,z,c,1,i))^2+ ...
3840. (p(2)-positionGR(x,y,z,c,2,i))^2) ;
3841. if (p(1)<max([points(5,1) points(6,1)]) && ...
3842. p(1)>min([points(5,1) points(6,1)]) && ...
3843. (p(2)<max([points(5,2) points(6,2)]) && ...
3844. p(2)>min([points(5,2) points(6,2)]))
3845. distanceGR(x,y,z,c,5,i) = d3 ;
3846. else
3847. distanceGR(x,y,z,c,5,i) = min([d1 d2]) ;
3848. end
3849. i=i+1 ;
3850. end
3851. end
3852. end
3853. end
3854. end
3855. % distance eau 3 (segment de droite)
3856. for x = 1:2
3857. for y = 1:4
3858. for z = 1:2
3859. for c = 1:2
3860. i=1 ;
3861. while i <= longueurGR
3862. d1 = sqrt((points(7,1)-positionGR(x,y,z,c,1,i))^2 ...
3863. +(points(7,2)-positionGR(x,y,z,c,2,i))^2) ;
3864. d2 = sqrt((points(8,1)-positionGR(x,y,z,c,1,i))^2 ...
3865. +(points(8,2)-positionGR(x,y,z,c,2,i))^2) ;
3866. e1 = [((points(8,1)-points(7,1))/(points(8,2)-points(7,2))) ...
3867. (points(7,1)-points(7,2))*((points(8,1)-points(7,1))/ ...
3868. (points(8,2)-points(7,2)))] ;
3869. e2 = [(-1/e1(1)) ...
3870. (positionGR(x,y,z,c,1,i)-positionGR(x,y,z,c,2,i)*(-1/e1(1)))] ;
3871. p = [(e1(1)*((e2(2)-e1(2))/(e1(1)-e2(1)))+e1(2)) ...
3872. ((e2(2)-e1(2))/(e1(1)-e2(1)))] ;

```

```

3873. d3 = sqrt((p(1)-positionGR(x,y,z,c,1,i))^2+ ...
3874.         (p(2)-positionGR(x,y,z,c,2,i))^2);
3875. if (p(1)<max([points(7,1) points(8,1)]) && ...
3876.     p(1)>min([points(7,1) points(8,1)])) && ...
3877.     (p(2)<max([points(7,2) points(8,2)]) && ...
3878.     p(2)>min([points(7,2) points(8,2)]))
3879. distanceGR(x,y,z,c,6,i) = d3;
3880. else
3881. distanceGR(x,y,z,c,6,i) = min([d1 d2]);
3882. end
3883. i=i+1;
3884. end
3885. end
3886. end
3887. end
3888. end
3889. % distance eau 4 (segment de droite)
3890. for x = 1:2
3891. for y = 1:4
3892. for z = 1:2
3893. for c = 1:2
3894. i=1;
3895. while i <= longueurGR
3896. d1 = sqrt((points(9,1)-positionGR(x,y,z,c,1,i))^2 ...
3897.         +(points(9,2)-positionGR(x,y,z,c,2,i))^2);
3898. d2 = sqrt((points(10,1)-positionGR(x,y,z,c,1,i))^2 ...
3899.         +(points(10,2)-positionGR(x,y,z,c,2,i))^2);
3900. e1 = [(points(10,1)-points(9,1))/(points(10,2)-points(9,2))] ...
3901.       (points(9,1)-points(9,2))*((points(10,1)-points(9,1))/ ...
3902.       (points(10,2)-points(9,2)))]);
3903. e2 = [(-1/e1(1)) ...
3904.       (positionGR(x,y,z,c,1,i)-positionGR(x,y,z,c,2,i)*(-1/e1(1)))]);
3905. p = [(e1(1))*((e2(2)-e1(2))/(e1(1)-e2(1)))+e1(2)] ...
3906.       ((e2(2)-e1(2))/(e1(1)-e2(1)))]);
3907. d3 = sqrt((p(1)-positionGR(x,y,z,c,1,i))^2 ...
3908.         +(p(2)-positionGR(x,y,z,c,2,i))^2);
3909. if (p(1)<max([points(9,1) points(10,1)]) && ...
3910.     p(1)>min([points(9,1) points(10,1)])) && ...
3911.     (p(2)<max([points(9,2) points(10,2)]) && ...
3912.     p(2)>min([points(9,2) points(10,2)]))
3913. distanceGR(x,y,z,c,7,i) = d3;
3914. else
3915. distanceGR(x,y,z,c,7,i) = min([d1 d2]);
3916. end
3917. i=i+1;
3918. end
3919. end
3920. end
3921. end
3922. end
3923. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR ...
3924.         vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR ...

```

```

3925.         position positionGR distance distanceGR ...
3926.         sel_eau sel_eauGR sel_eau_stat_final sel_eau_stat_final_GR
3927. %% temps passe a < 10 m d un bac a sel
3928. %% + temps passe a < 10 m d un point d eau
3929. %% pour le jour et pour la nuit
3930. % comptage par bete
3931. sel_eau = zeros(2,4,2,2,3);
3932. for x = 1:2
3933. for y = 1:4
3934. for z = 1:2
3935. for c = 1:2
3936. i=1;
3937. while i <= longueur
3938. if distance(x,y,z,c,1,i) <= 10 || distance(x,y,z,c,2,i) <= 10 || ...
3939.     distance(x,y,z,c,3,i) <= 10
3940. sel_eau(x,y,z,c,1) = sel_eau(x,y,z,c,1) + 1;
3941. elseif distance(x,y,z,c,4,i) <= 10 || distance(x,y,z,c,5,i) <= 10 || ...
3942.     distance(x,y,z,c,6,i) <= 10 || distance(x,y,z,c,7,i) <= 10
3943. sel_eau(x,y,z,c,2) = sel_eau(x,y,z,c,2) + 1;
3944. end
3945. if isnan(distance(x,y,z,c,1,i)) == 0
3946. sel_eau(x,y,z,c,3) = sel_eau(x,y,z,c,3) + 1;
3947. end
3948. i=i+1;
3949. end
3950. sel_eau(x,y,z,c,1:2) = sel_eau(x,y,z,c,1:2) / sel_eau(x,y,z,c,3);
3951. end
3952. end
3953. end
3954. end
3955. sel_eau = sel_eau(:,:,:,1:2);
3956. clearvars -except vitesse longueurGR vitesse2 vitesse2GR ...
3957.         vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR ...
3958.         position positionGR distance distanceGR ...
3959.         sel_eau sel_eauGR sel_eau_stat_final sel_eau_stat_final_GR
3960. %% moyenne
3961. sel_eau_global = zeros(4,1);
3962. sel_eau_pature = zeros(4,2);
3963. sel_eau_race = zeros(4,4);
3964. for i = 1:2
3965. for j = 1:2
3966. sel_eau_global(((j-1)*2+i,:)) = mean(mean(mean(sel_eau(:,:,:,i,j))));
3967. sel_eau_race(((j-1)*2+i,:)) = mean(mean(sel_eau(:,:,:,i,j),3),1);
3968. sel_eau_pature(((j-1)*2+i,:)) = (mean(mean(sel_eau(:,:,:,i,j),3),2))';
3969. end
3970. end
3971. sel_eau_moy = [sel_eau_global sel_eau_pature sel_eau_race];
3972. % ecart type global
3973. sel_eau_global_ET = zeros(4,1);
3974. sel_eau_pature_ET = zeros(4,2);
3975. sel_eau_race_ET = zeros(4,4);
3976. for i = 1:2

```

```

3977. for j = 1:2
3978. sel_eau_global_M = ones(2,4,2)*sel_eau_global(((j-1)*2+i,:);
3979. sel_eau_global_DM = sel_eau(:,:,i,j) - sel_eau_global_M;
3980. sel_eau_global_DM2 = sel_eau_global_DM.*sel_eau_global_DM;
3981. sel_eau_global_V = mean(mean(mean(sel_eau_global_DM2)));
3982. sel_eau_global_ET(((j-1)*2+i,:)) = sqrt(sel_eau_global_V);
3983. end
3984. end
3985. % ecart type race
3986. for i = 1:2
3987. for j = 1:2
3988. sel_eau_race_M = ones(2,4,2);
3989. for i_race1 = 1:2
3990. for i_race2 = 1:2
3991. sel_eau_race_M(i_race1,:,i_race2) = sel_eau_race(((j-1)*2+i,:));
3992. end
3993. end
3994. sel_eau_race_DM = sel_eau(:,:,i,j) - sel_eau_race_M;
3995. sel_eau_race_DM2 = sel_eau_race_DM.*sel_eau_race_DM;
3996. sel_eau_race_V = mean(mean(sel_eau_race_DM2,3),1);
3997. sel_eau_race_ET(((j-1)*2+i,:)) = sqrt(sel_eau_race_V);
3998. end
3999. end
4000. % ecart type pature
4001. for i = 1:2
4002. for j = 1:2
4003. sel_eau_pature_M = ones(2,4,2);
4004. for i_pature1 = 1:4
4005. for i_pature2 = 1:2
4006. sel_eau_pature_M(:,i_pature1,i_pature2) = sel_eau_pature(((j-1)*2+i,:));
4007. end
4008. end
4009. sel_eau_pature_DM = sel_eau(:,:,i,j) - sel_eau_pature_M;
4010. sel_eau_pature_DM2 = sel_eau_pature_DM.*sel_eau_pature_DM;
4011. sel_eau_pature_V = mean(mean(sel_eau_pature_DM2,3),2);
4012. sel_eau_pature_ET(((j-1)*2+i,:)) = (sqrt(sel_eau_pature_V))';
4013. end
4014. end
4015. % ecart type fusion
4016. sel_eau_ET = [sel_eau_global_ET sel_eau_pature_ET sel_eau_race_ET];
4017. % ANOVA
4018. sel_eau_AV = zeros(4,16);
4019. sel_eau_pvalue = zeros(4,3);
4020. for i = 1:2
4021. for j = 1:2
4022. sel_eau_AV(((j-1)*2+i,:)) = ...
4023.     [sel_eau(1,:,1,i,j) sel_eau(1,:,2,i,j) ...
4024.     sel_eau(2,:,1,i,j) sel_eau(2,:,2,i,j)];
4025. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
4026. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
4027. sel_eau_pvalue(((j-1)*2+i,:)) = ...
4028.     fliplr(anovan(sel_eau_AV(((j-1)*2+i,:),{f1,f2},'full')));

```

```

4029. close all force
4030. end
4031. end
4032. % fusion
4033. sel_eau_stat = [sel_eau_moy sel_eau_ET sel_eau_pvalue];
4034. sel_eau_stat(:,1:14) = sel_eau_stat(:,1:14)*100;
4035. C = sel_eau_stat;
4036. sel_eau_stat_final = [C(:,1) C(:,8) C(:,15) C(:,2) C(:,9) C(:,3) ...
4037.     C(:,10) C(:,16) C(:,4) C(:,11) C(:,5) C(:,12) C(:,6) C(:,13) ...
4038.     C(:,7) C(:,14) C(:,17)];
4039. clearvars -except vitesse longueurGR vitesse2 vitesse2GR ...
4040.     vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR ...
4041.     position positionGR distance distanceGR ...
4042.     sel_eau sel_eauGR sel_eau_stat_final sel_eau_stat_final_GR
4043. %% temps passe a < 10 m d un bac a sel
4044. %% + temps passe a < 10 m d un point d eau
4045. %% pour grazing et non grazing
4046. % comptage par bete
4047. sel_eauGR = zeros(2,4,2,2,3);
4048. for x = 1:2
4049. for y = 1:4
4050. for z = 1:2
4051. for c = 1:2
4052. i=1;
4053. while i <= longueurGR
4054. if distanceGR(x,y,z,c,1,i) <= 10 || distanceGR(x,y,z,c,2,i) <= 10 || ...
4055.     distanceGR(x,y,z,c,3,i) <= 10
4056. sel_eauGR(x,y,z,c,1) = sel_eauGR(x,y,z,c,1) + 1;
4057. elseif distanceGR(x,y,z,c,4,i) <= 10 || distanceGR(x,y,z,c,5,i) <= 10 || ...
4058.     distanceGR(x,y,z,c,6,i) <= 10 || distanceGR(x,y,z,c,7,i) <= 10
4059. sel_eauGR(x,y,z,c,2) = sel_eauGR(x,y,z,c,2) + 1;
4060. end
4061. if isnan(distanceGR(x,y,z,c,1,i)) == 0
4062. sel_eauGR(x,y,z,c,3) = sel_eauGR(x,y,z,c,3) + 1;
4063. end
4064. i=i+1;
4065. end
4066. sel_eauGR(x,y,z,c,1:2) = sel_eauGR(x,y,z,c,1:2) / sel_eauGR(x,y,z,c,3);
4067. end
4068. end
4069. end
4070. end
4071. sel_eauGR = sel_eauGR(:,:,,1:2);
4072. clearvars -except vitesse longueurGR vitesse2 vitesse2GR ...
4073.     vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR ...
4074.     position positionGR distance distanceGR ...
4075.     sel_eau sel_eauGR sel_eau_stat_final sel_eau_stat_final_GR
4076. %% moyenne
4077. sel_eauGR_global = zeros(4,1);
4078. sel_eauGR_pature = zeros(4,2);
4079. sel_eauGR_race = zeros(4,4);
4080. for i = 1:2

```

```

4081. for j = 1:2
4082. sel_eauGR_global(((j-1)*2+i,:) = mean(mean(mean(sel_eauGR(:,:;i,j)))));
4083. sel_eauGR_race(((j-1)*2+i,:) = mean(mean(sel_eauGR(:,:;i,j),3),1);
4084. sel_eauGR_pature(((j-1)*2+i,:) = (mean(mean(sel_eauGR(:,:;i,j),3),2))';
4085. end
4086. end
4087. sel_eauGR_moy = [sel_eauGR_global sel_eauGR_pature sel_eauGR_race];
4088. %% ecart type global
4089. sel_eauGR_global_ET = zeros(4,1);
4090. sel_eauGR_pature_ET = zeros(4,2);
4091. sel_eauGR_race_ET = zeros(4,4);
4092. for i = 1:2
4093. for j = 1:2
4094. sel_eauGR_global_M = ones(2,4,2)*sel_eauGR_global(((j-1)*2+i,:);
4095. sel_eauGR_global_DM = sel_eauGR(:,:;i,j) - sel_eauGR_global_M;
4096. sel_eauGR_global_DM2 = sel_eauGR_global_DM.*sel_eauGR_global_DM;
4097. sel_eauGR_global_V = mean(mean(mean(sel_eauGR_global_DM2)));
4098. sel_eauGR_global_ET(((j-1)*2+i,:)) = sqrt(sel_eauGR_global_V);
4099. end
4100. end
4101. %% ecart type race
4102. for i = 1:2
4103. for j = 1:2
4104. sel_eauGR_race_M = ones(2,4,2);
4105. for i_race1 = 1:2
4106. for i_race2 = 1:2
4107. sel_eauGR_race_M(i_race1,:;i_race2) = sel_eauGR_race(((j-1)*2+i,:);
4108. end
4109. end
4110. sel_eauGR_race_DM = sel_eauGR(:,:;i,j) - sel_eauGR_race_M;
4111. sel_eauGR_race_DM2 = sel_eauGR_race_DM.*sel_eauGR_race_DM;
4112. sel_eauGR_race_V = mean(mean(sel_eauGR_race_DM2,3),1);
4113. sel_eauGR_race_ET(((j-1)*2+i,:)) = sqrt(sel_eauGR_race_V);
4114. end
4115. end
4116. %% ecart type pature
4117. for i = 1:2
4118. for j = 1:2
4119. sel_eauGR_pature_M = ones(2,4,2);
4120. for i_pature1 = 1:4
4121. for i_pature2 = 1:2
4122. sel_eauGR_pature_M(i_pature1,i_pature2) = sel_eauGR_pature(((j-1)*2+i,:);
4123. end
4124. end
4125. sel_eauGR_pature_DM = sel_eauGR(:,:;i,j) - sel_eauGR_pature_M;
4126. sel_eauGR_pature_DM2 = sel_eauGR_pature_DM.*sel_eauGR_pature_DM;
4127. sel_eauGR_pature_V = mean(mean(sel_eauGR_pature_DM2,3),2);
4128. sel_eauGR_pature_ET(((j-1)*2+i,:)) = (sqrt(sel_eauGR_pature_V))';
4129. end
4130. end
4131. %% ecart type fusion
4132. sel_eauGR_ET = [sel_eauGR_global_ET sel_eauGR_pature_ET

```

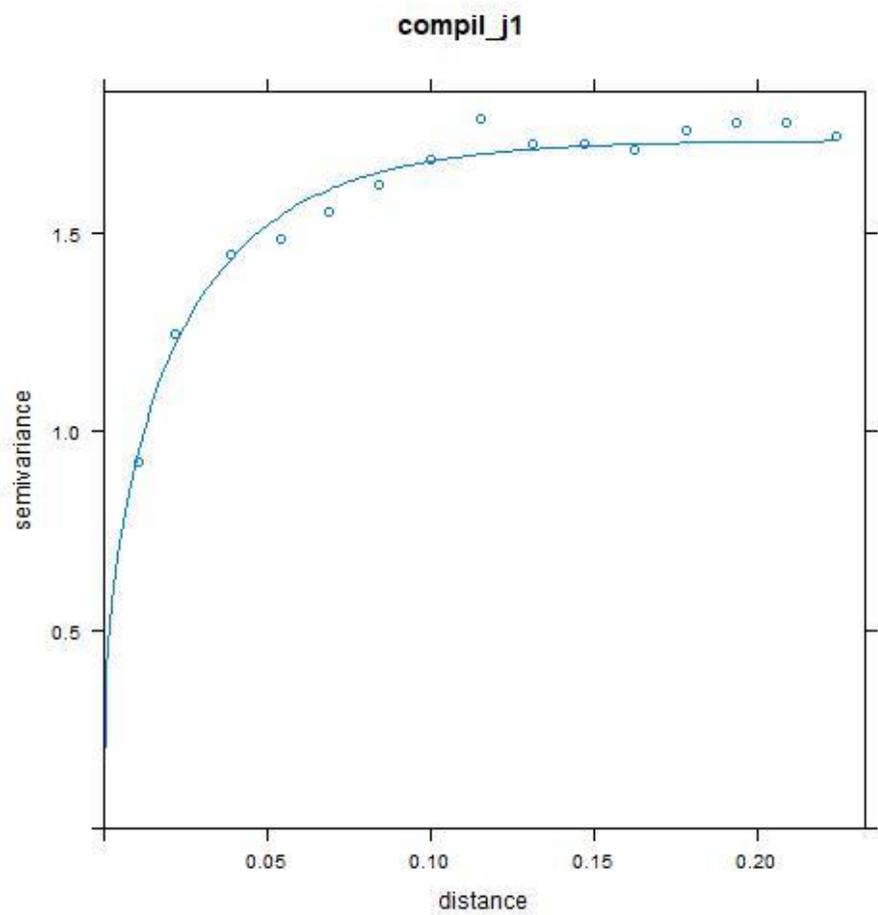
```

sel_eauGR_race_ET];
4133. %% ANOVA
4134. sel_eauGR_AV = zeros(4,16);
4135. sel_eauGR_pvalue = zeros(4,3);
4136. for i = 1:2
4137. for j = 1:2
4138. sel_eauGR_AV(((j-1)*2+i,:)) = ...
4139. [sel_eauGR(1,:;1,i,j) sel_eauGR(1,:;2,i,j) ...
4140. sel_eauGR(2,:;1,i,j) sel_eauGR(1,:;2,i,j)];
4141. f1 = [1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4];
4142. f2 = [1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2];
4143. sel_eauGR_pvalue(((j-1)*2+i,:)) = ...
4144. fliplr((anovan(sel_eauGR_AV(((j-1)*2+i,:)),{f1,f2},'full')));
4145. close all force
4146. end
4147. end
4148. %% fusion
4149. sel_eauGR_stat = [sel_eauGR_moy sel_eauGR_ET sel_eauGR_pvalue];
4150. sel_eauGR_stat(:,1:14) = sel_eauGR_stat(:,1:14)*100;
4151. CGR = sel_eauGR_stat;
4152. sel_eau_stat_final_GR = [CGR(:,1) CGR(:,8) CGR(:,15) CGR(:,2) CGR(:,9) ...
4153. CGR(:,3) CGR(:,10) CGR(:,16) CGR(:,4) CGR(:,11) CGR(:,5) CGR(:,12) ...
4154. CGR(:,6) CGR(:,13) CGR(:,7) CGR(:,14) CGR(:,17)];
4155. clearvars -except vitesse longueur longueurGR vitesse2 vitesse2GR ...
4156. vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR ...
4157. position positionGR distance distanceGR ...
4158. sel_eau sel_eauGR sel_eau_stat_final sel_eau_stat_final_GR
4159. %% compilation en une seule matrice
4160. stat_final = [vitesse_stat_final; sel_eau_stat_final; ...
4161. vitesse_stat_final_GR; sel_eau_stat_final_GR];
4162. F = stat_final;
4163. stat_final = [F(1,:); F(3,:); F(5,:); F(7,:); F(9,:); F(11,:); ...
4164. F(2,:); F(4,:); F(6,:); F(8,:); F(10,:); F(12,:); ...
4165. F(13,:); F(15,:); F(17,:); F(19,:); F(21,:); F(23,:); ...
4166. F(14,:); F(16,:); F(18,:); F(20,:); F(22,:); F(24,:);];
4167. %% ecriture fichier
4168. xlswrite('stat_vitesse5',stat_final)
4169. %% ecriture lignes
4170. S1 = 'vitesse moyenne (m/h)';
4171. S2 = 'temps entre 0 et 0,5 m/s (%)';
4172. S3 = 'temps entre 0,5 et 2 m/s (%)';
4173. S4 = 'temps a plus de 2 m/s (%)';
4174. S5 = 'temps a moins de 10 m d un bac a sel (%)';
4175. S6 = 'temps a moins de 10 m d un point d eau le jour (%)';
4176. SS = 'repete pour le jour la nuit grazing et non grazing';
4177. lignes={S1;S2;S3;S4;S5;S6;SS};
4178. xlswrite('lignes_stat_vitesse',lignes)
4179. clearvars -except vitesse longueurGR vitesse2 vitesse2GR ...
4180. vitesse3 vitesse3GR vitesse_stat_final vitesse_stat_final_GR ...
4181. position positionGR distance distanceGR ...
4182. sel_eau sel_eauGR sel_eau_stat_final sel_eau_stat_final_GR stat_final

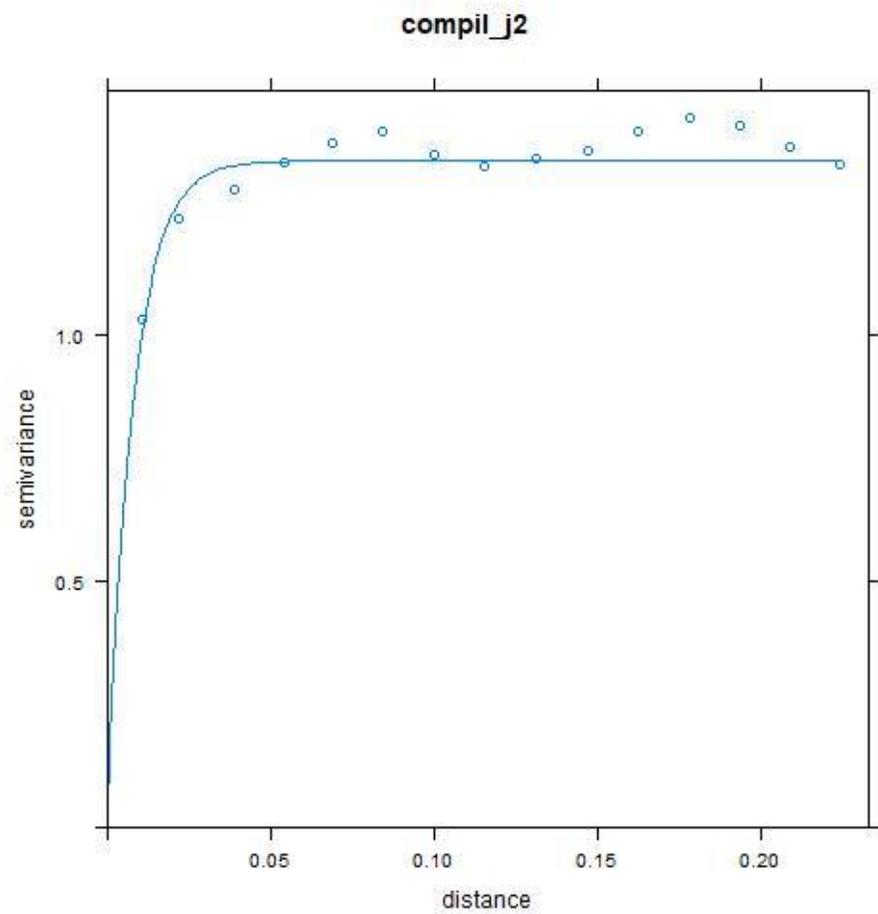
```

Annexe 23 : variogrammes de la présence des bêtes

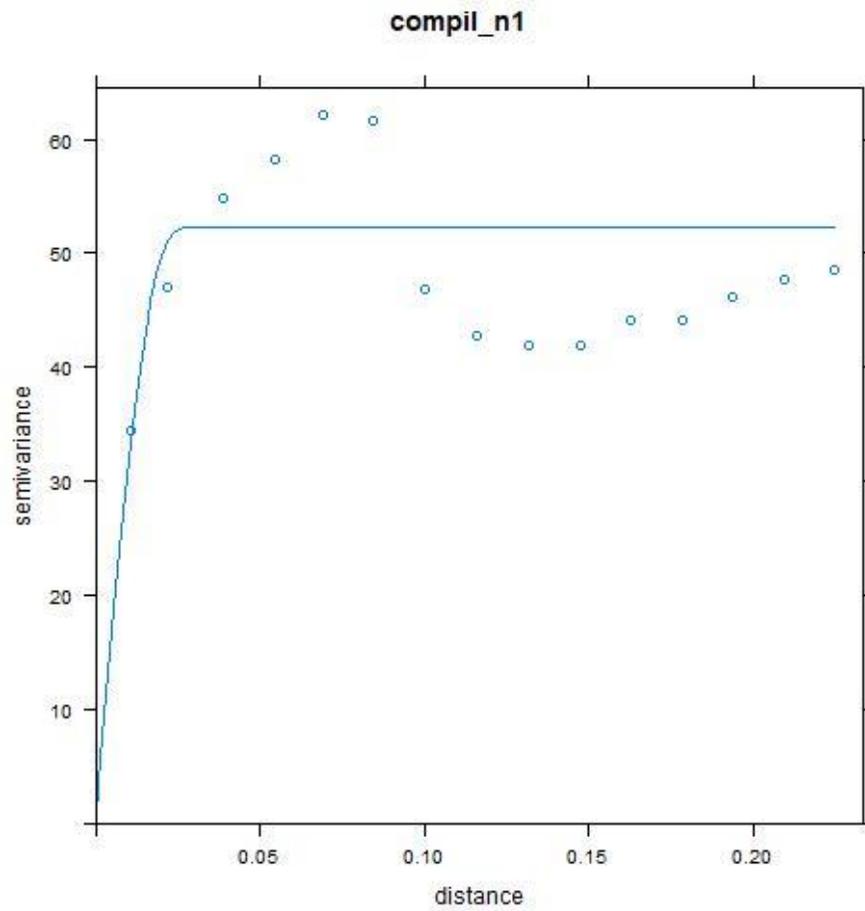
Groupe 1 : bêtes de jour en pâture 1



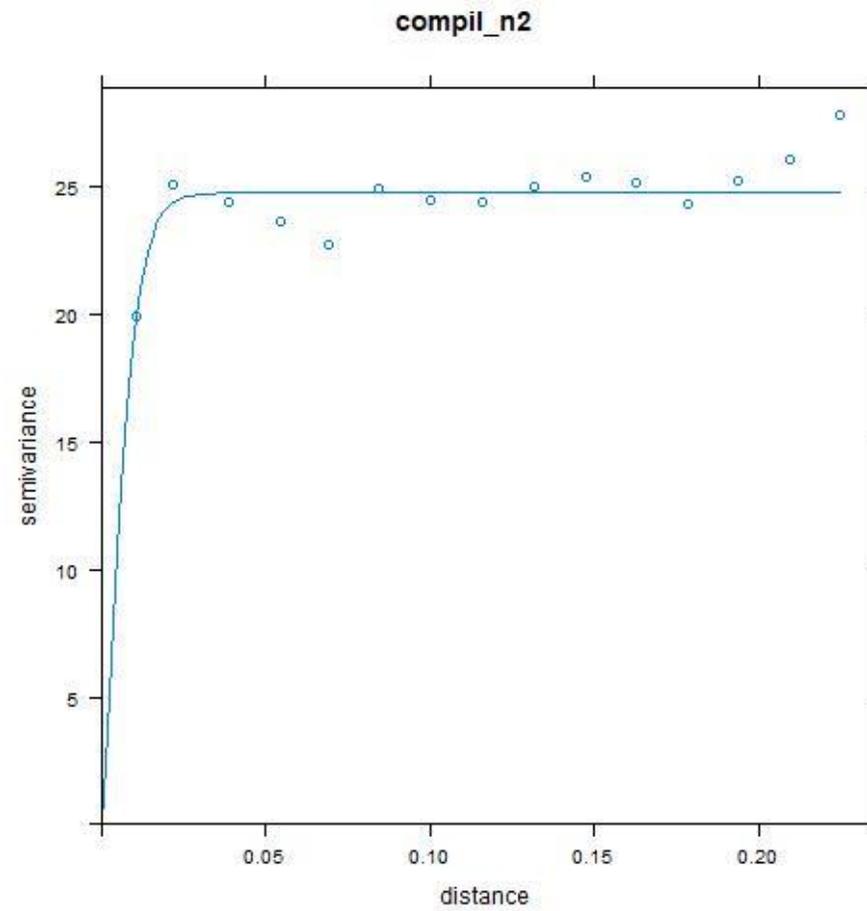
Groupe 2 : bêtes de jour en pâture 2



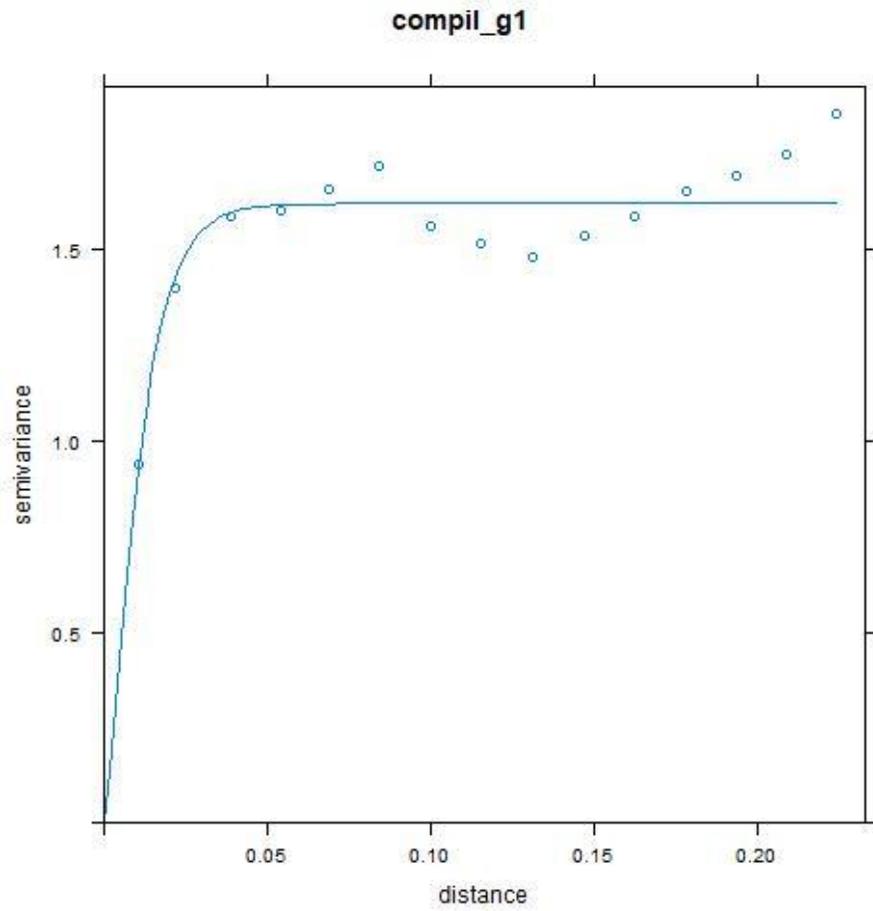
Groupe 3 : bêtes de nuit en pâture 1



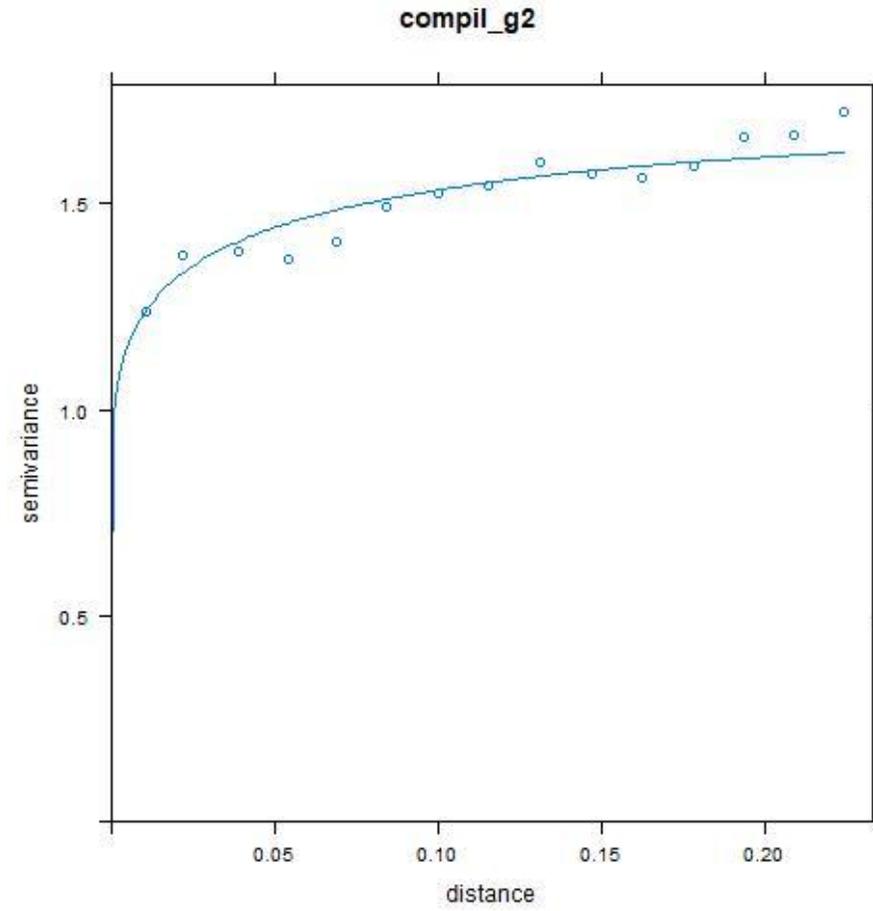
Groupe 4 : bêtes de nuit en pâture 2



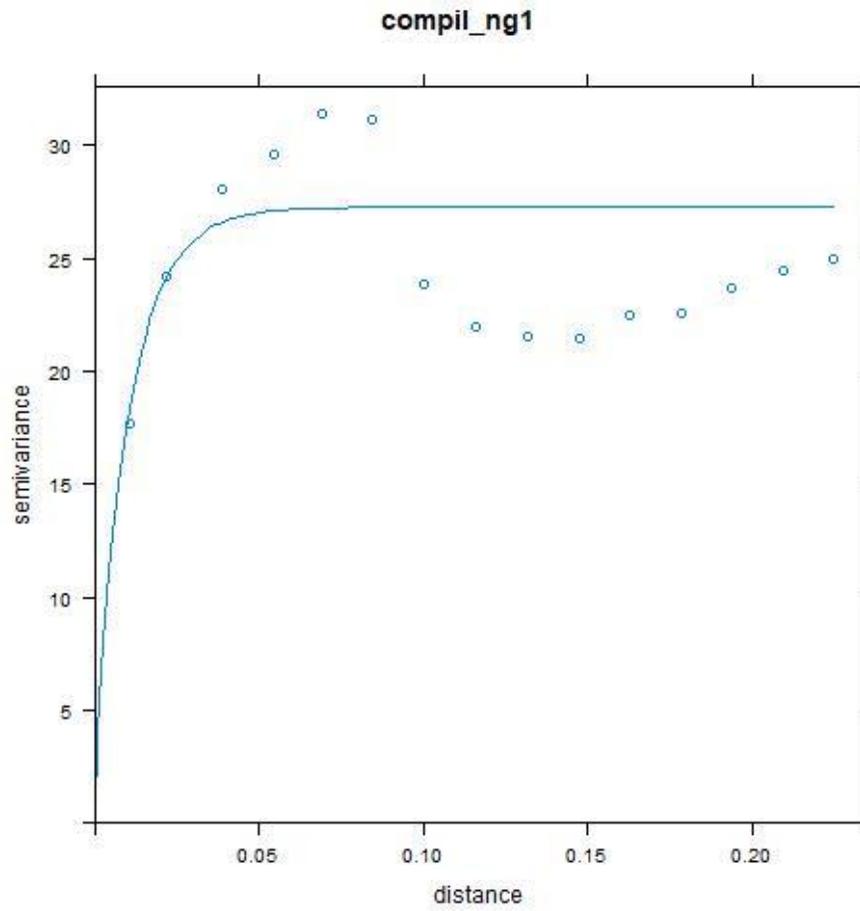
Groupe 5 : bêtes « grazing » en pâture 1



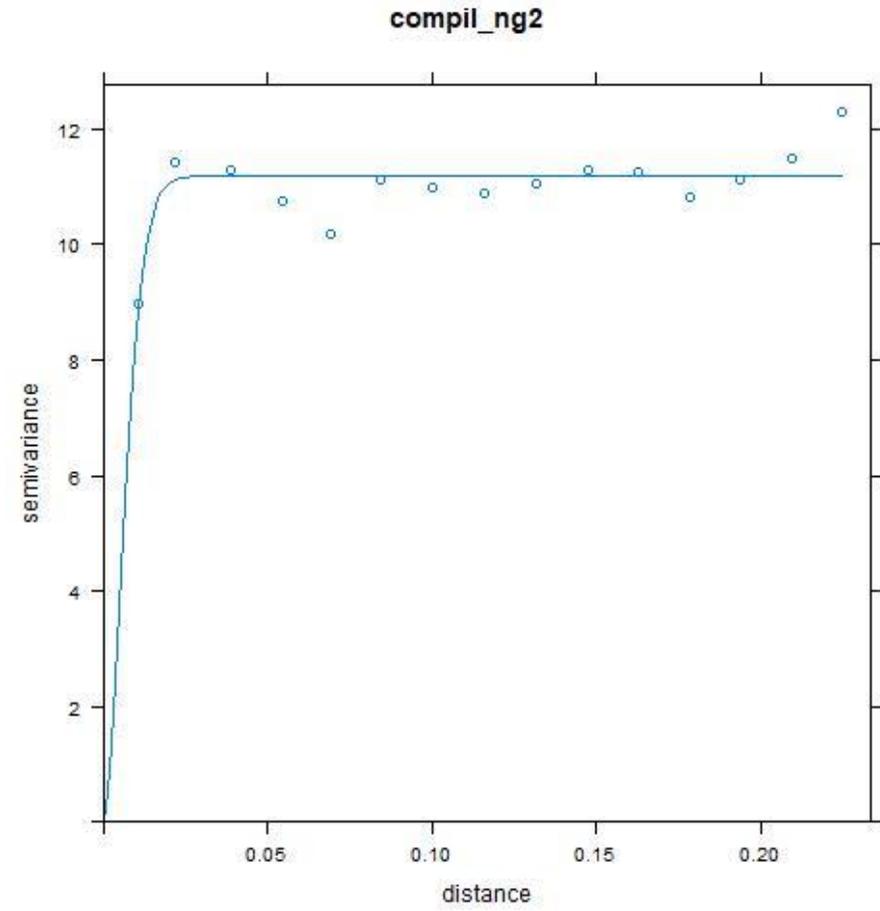
Groupe 6 : bêtes « grazing » en pâture 2



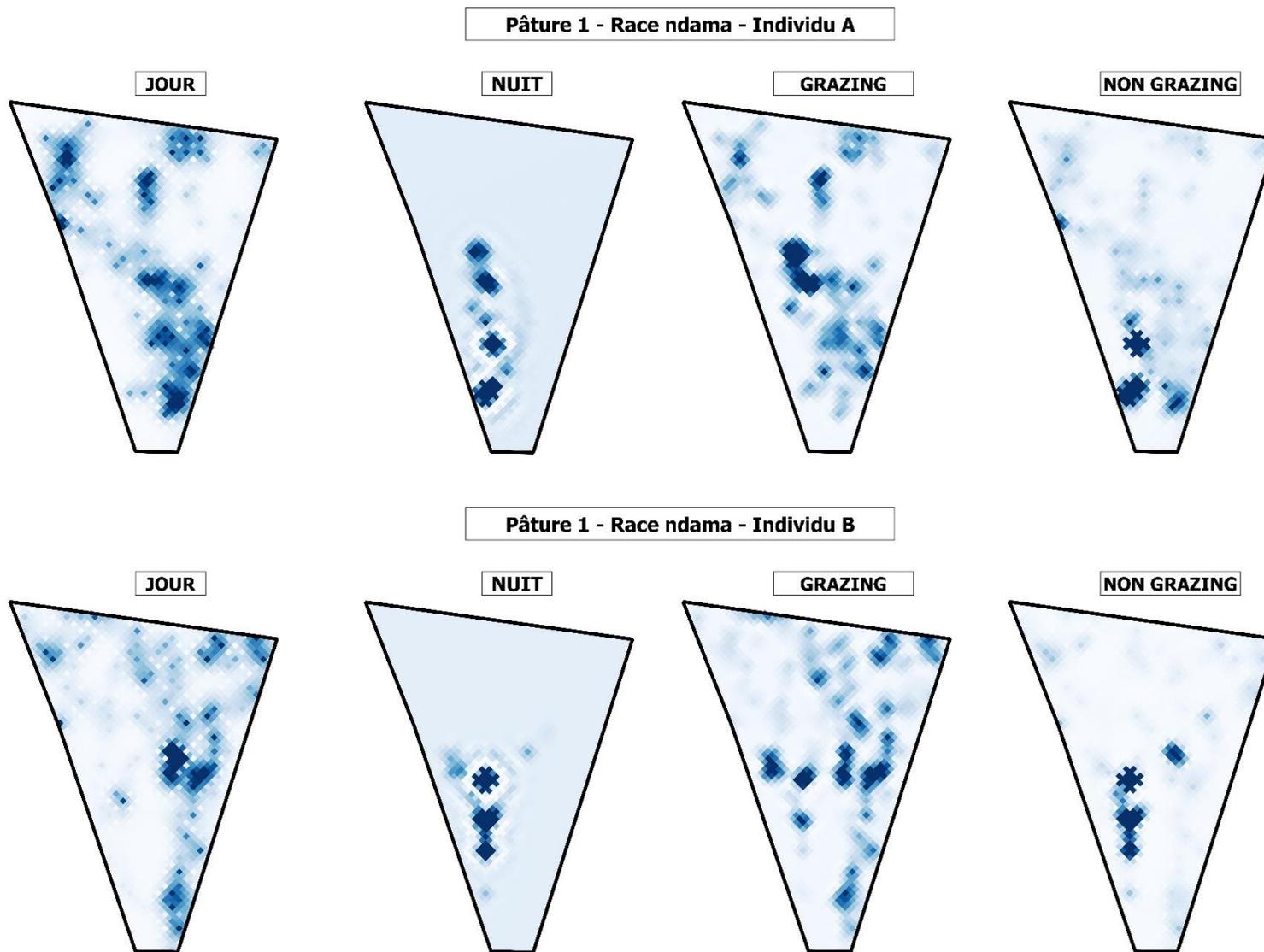
Groupe 7 : bêtes « non grazing » en pâture 1



Groupe 8 : bêtes « non grazing » en pâture 2

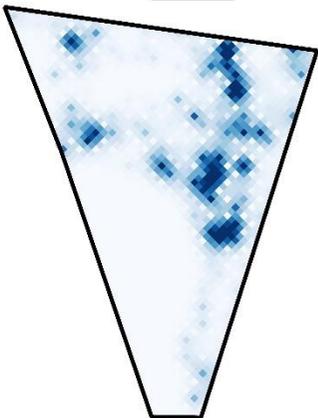


Annexe 24 : rasters de la présence des bêtes

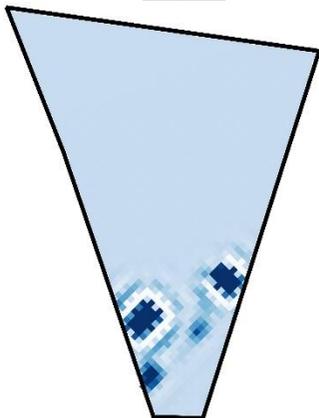


Pâtûre 1 - Race zébu - Individu A

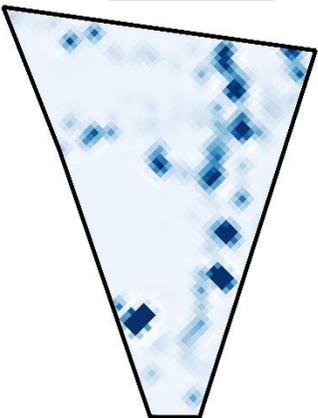
JOUR



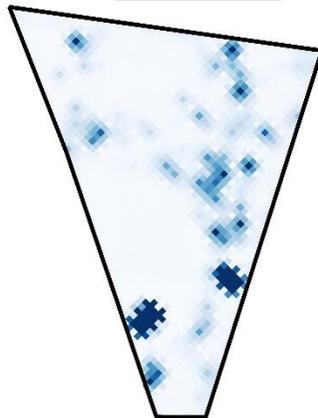
NUIT



GRAZING

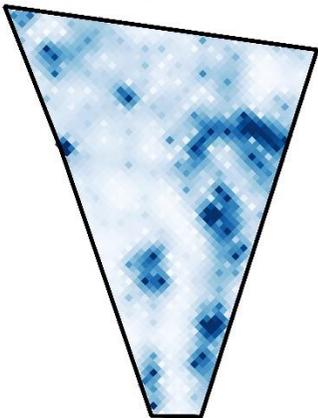


NON GRAZING

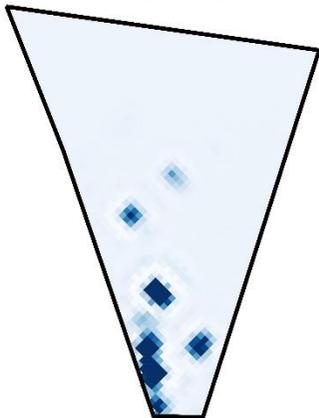


Pâtûre 1 - Race zébu - Individu B

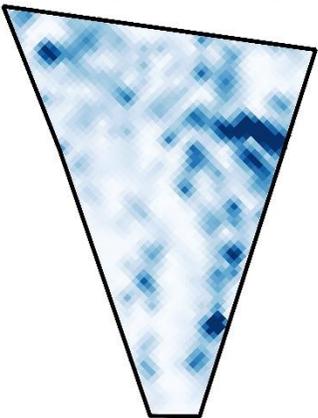
JOUR



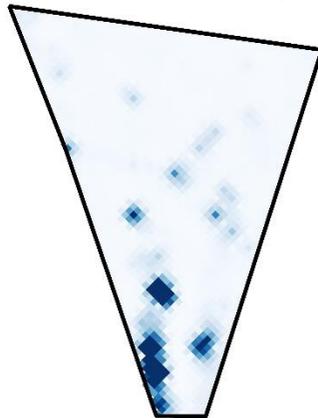
NUIT



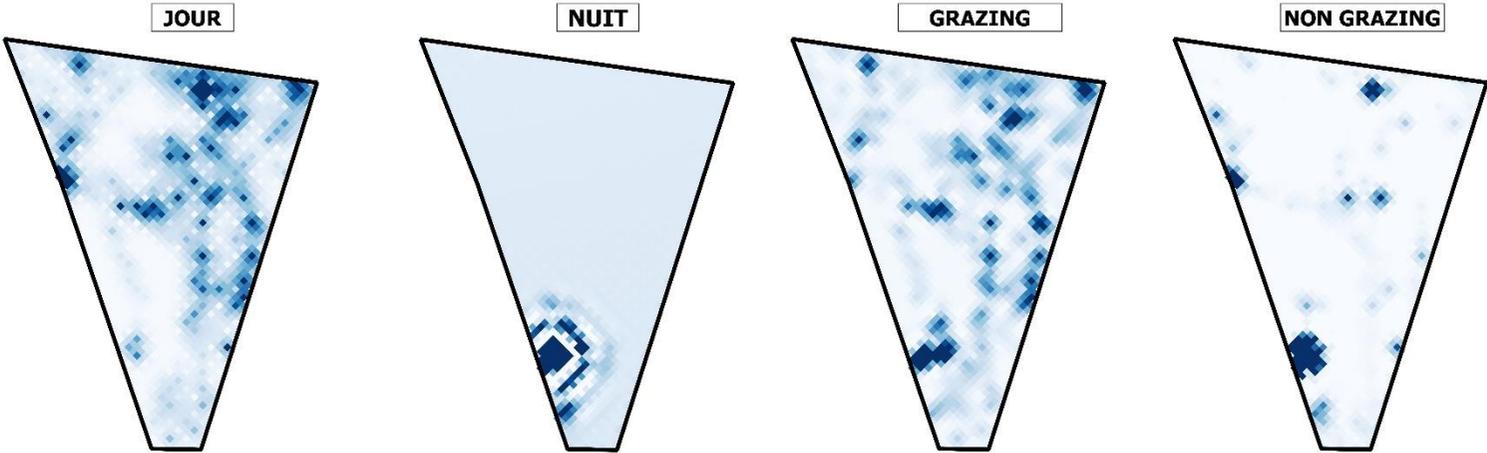
GRAZING



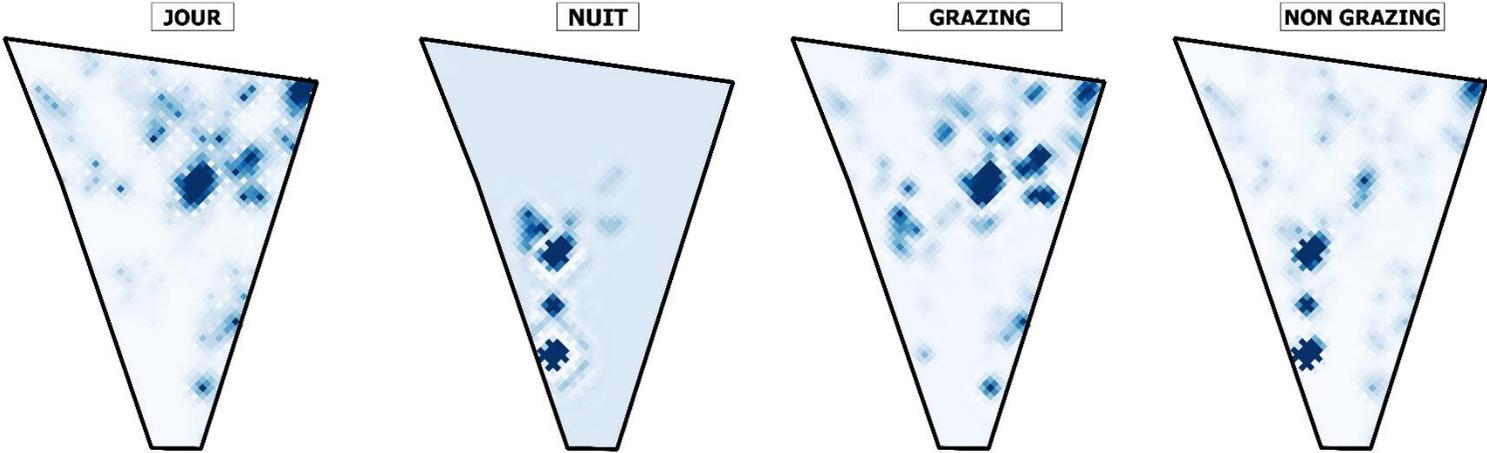
NON GRAZING



Pâtûre 1 - Race ndapol - Individu A

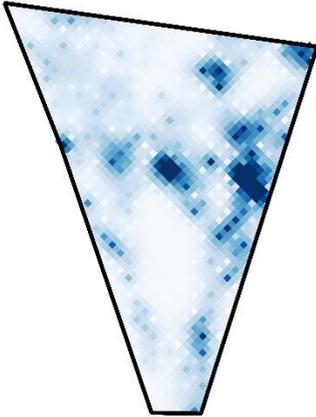


Pâtûre 1 - Race ndapol - Individu B

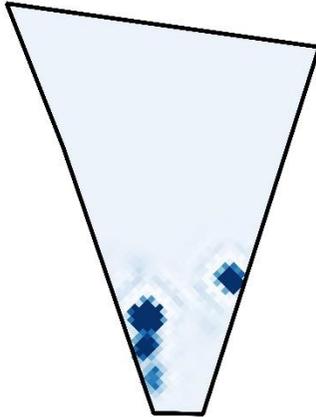


Pâtûre 1 - Race zépol - Individu A

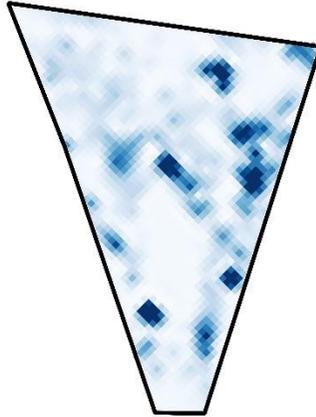
JOUR



NUIT



GRAZING

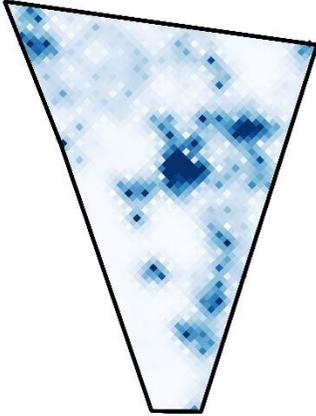


NON GRAZING

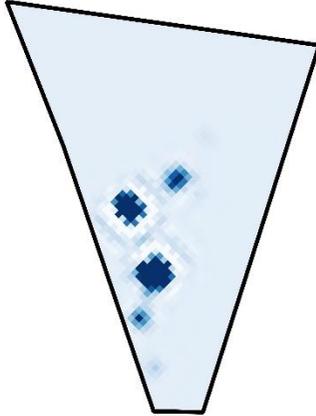


Pâtûre 1 - Race zépol - Individu B

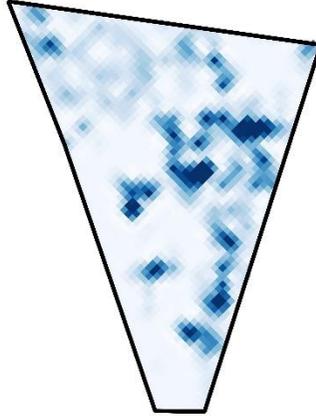
JOUR



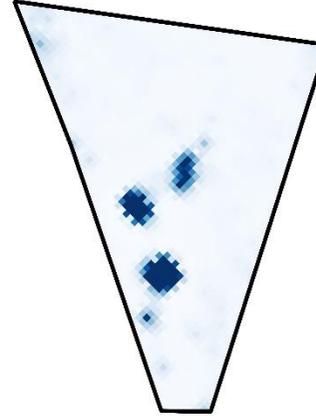
NUIT



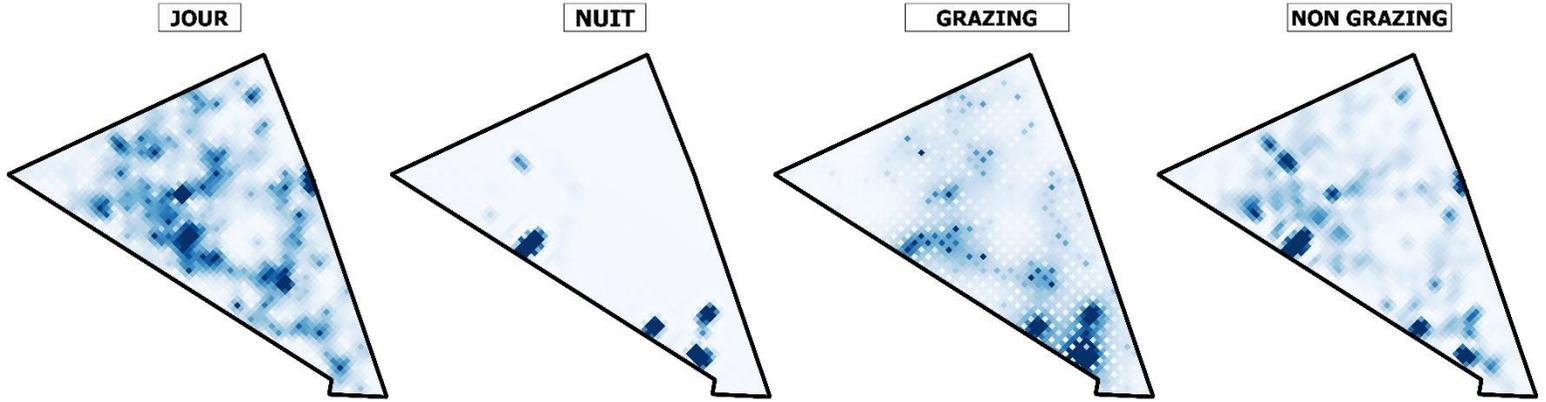
GRAZING



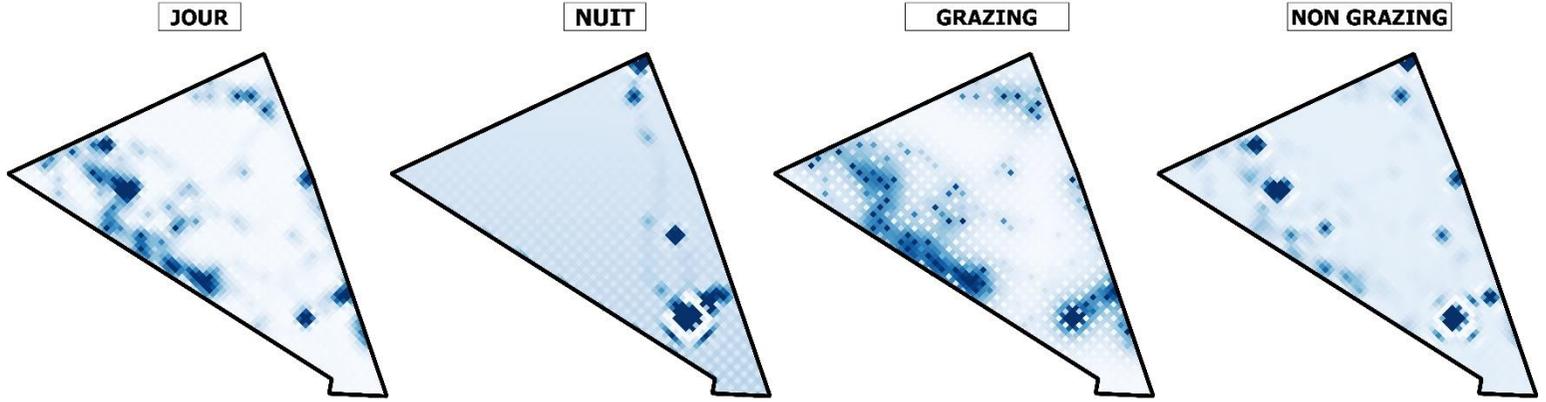
NON GRAZING



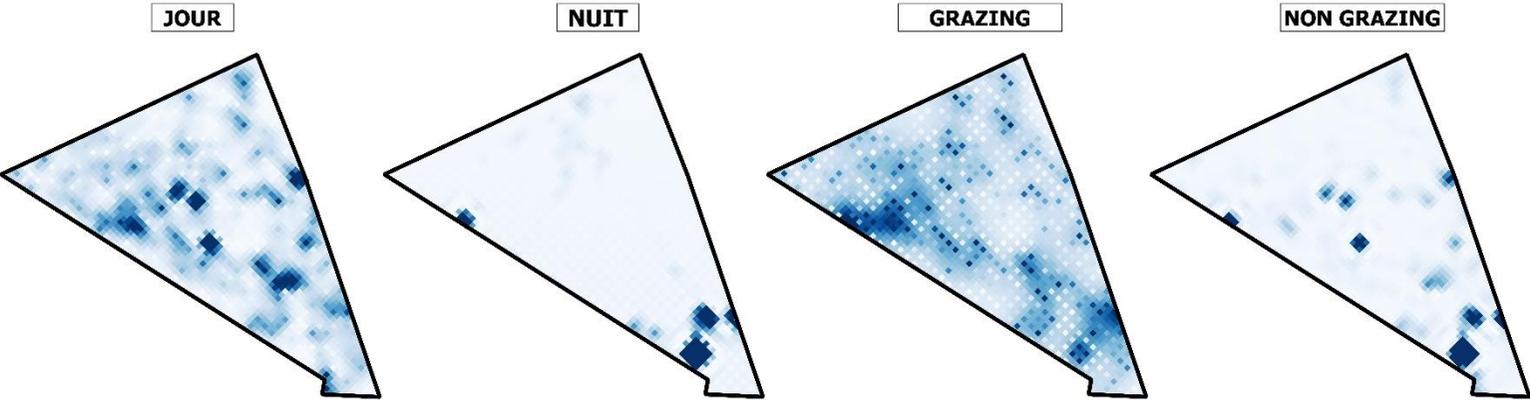
Pâtûre 2 - Race ndama - Individu A



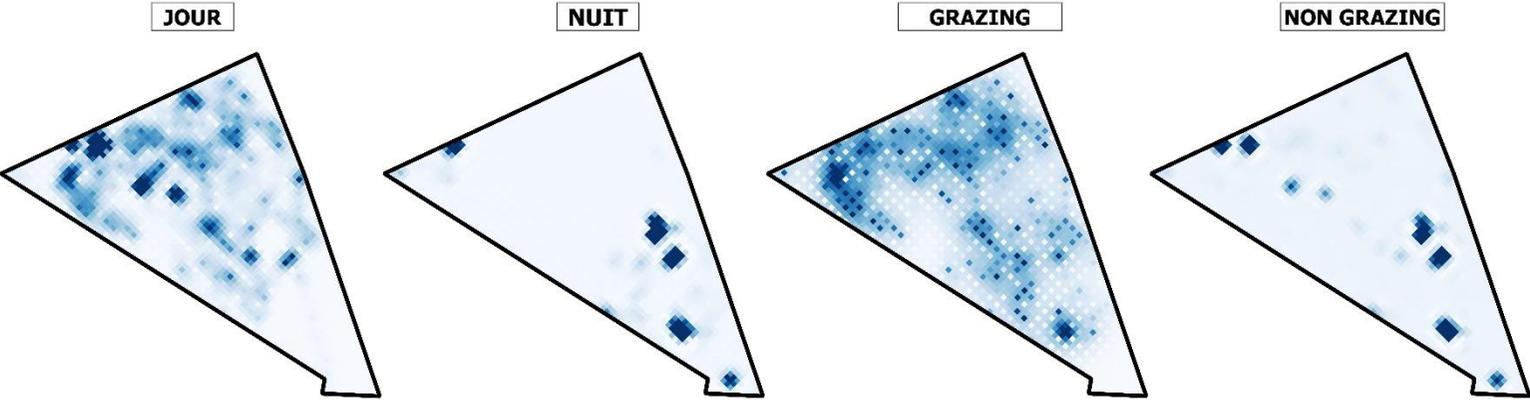
Pâtûre 2 - Race ndama - Individu B



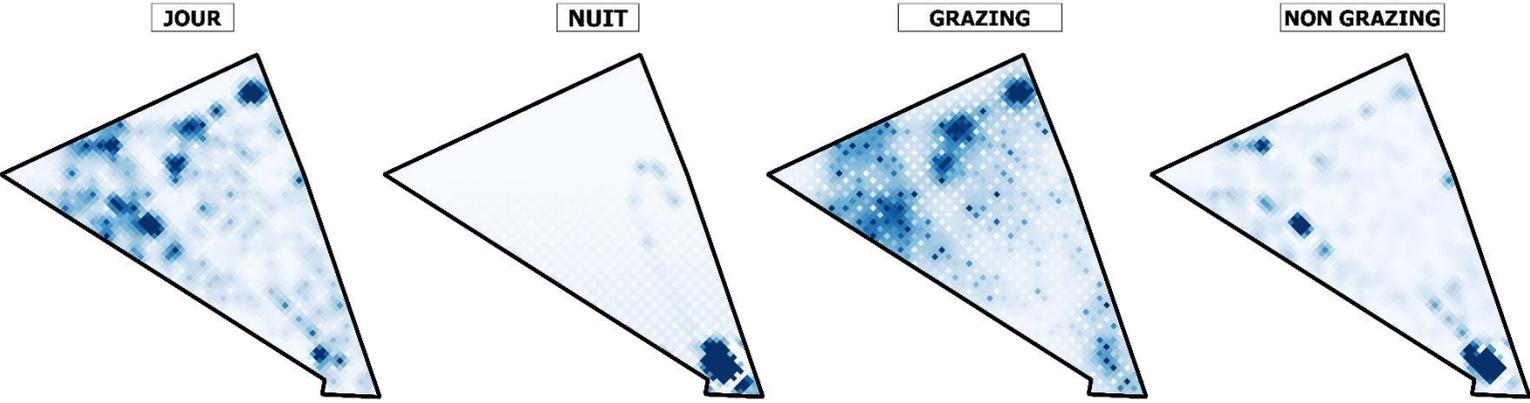
Pâtûre 2 - Race zébu - Individu A



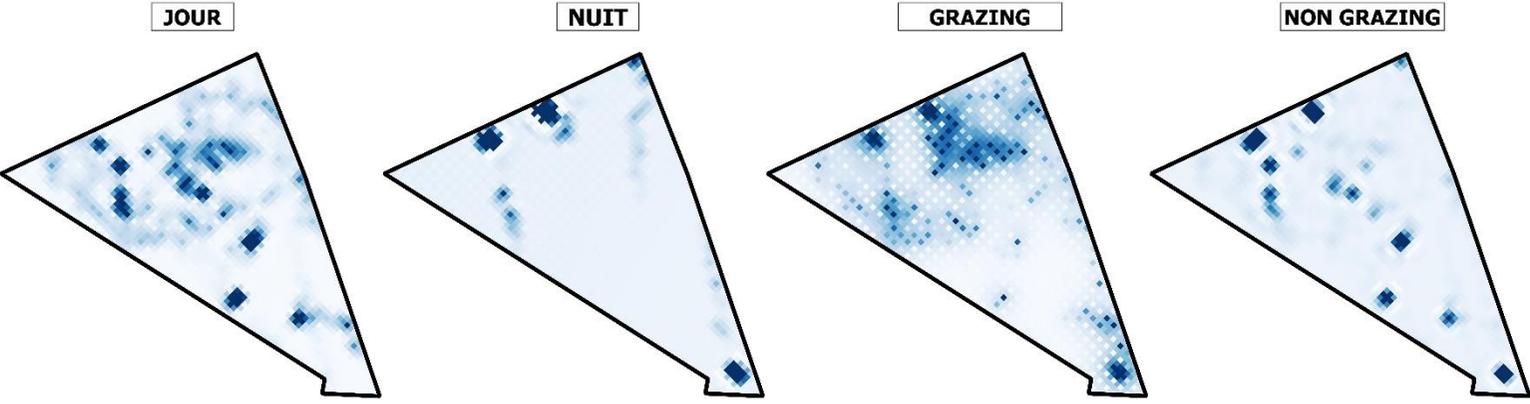
Pâtûre 2 - Race zébu - Individu B



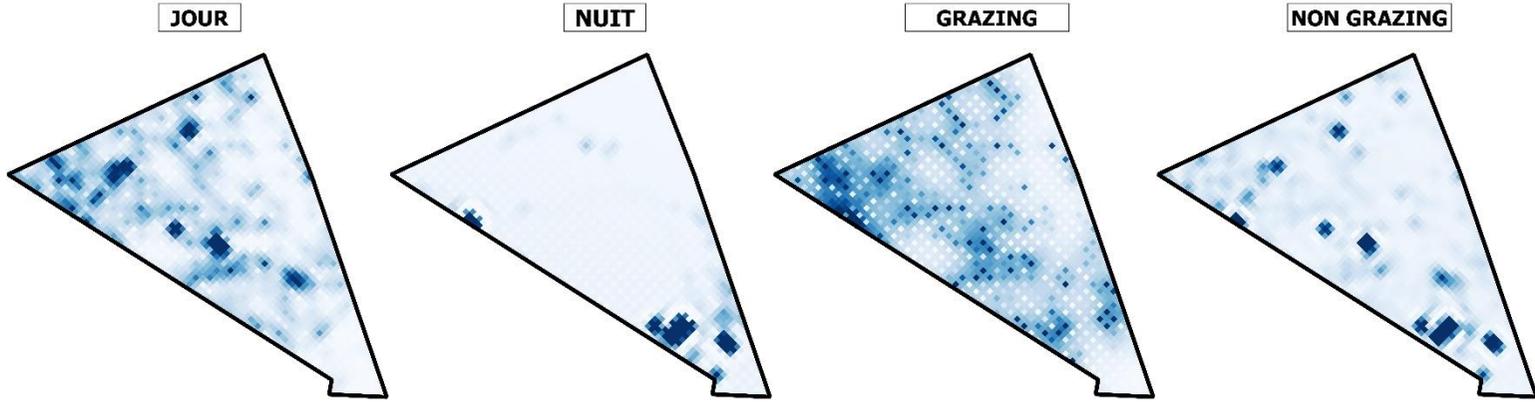
Pâtûre 2 - Race ndapol - Individu A



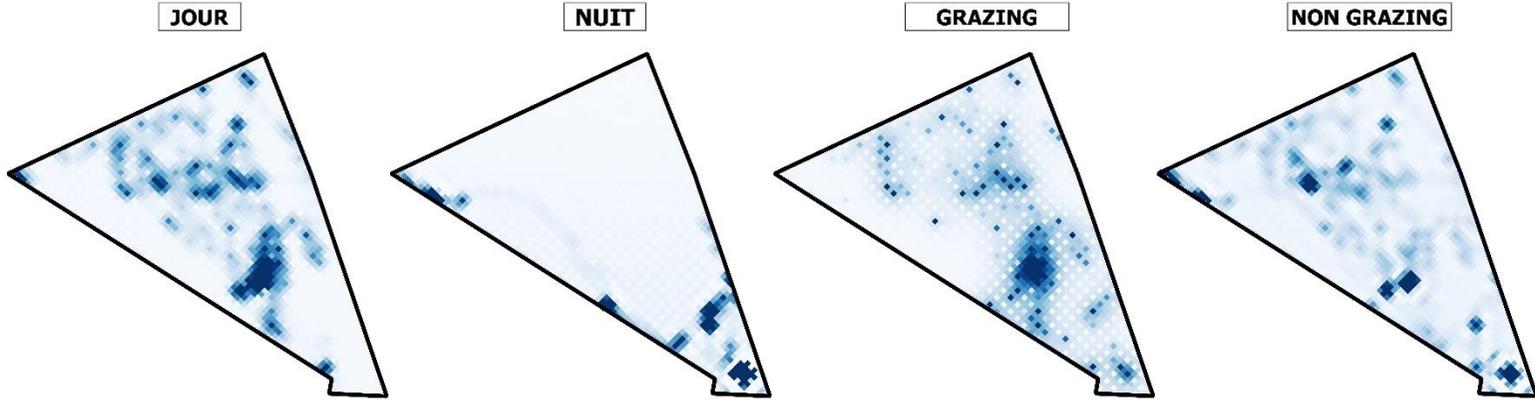
Pâtûre 2 - Race ndapol - Individu B



Pâture 2 - Race zépol - Individu A



Pâture 2 - Race zépol - Individu B



Annexe 25 : analyse en composantes principales des caractéristiques de la végétation

Matrice de corrélations des critères utilisés pour la caractérisation de la végétation

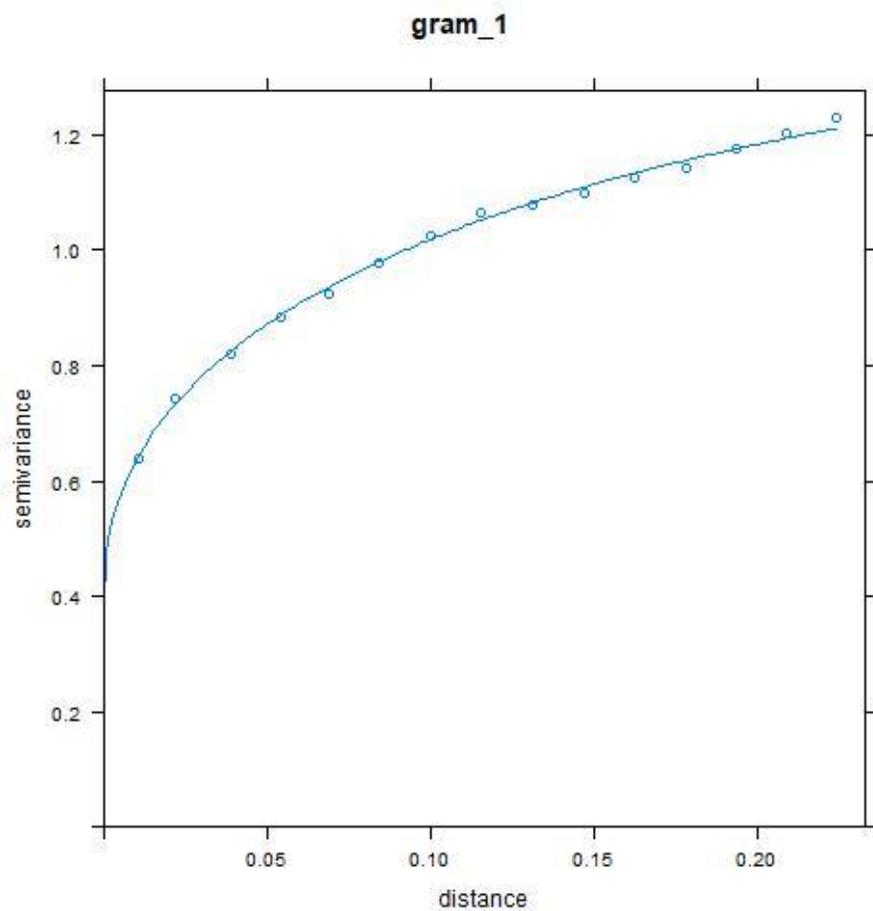
	Critère 1	Critère 2	Critère 3	Critère 4	Critère 5	Critère 6	Critère 7	Critère 8	Critère 9	Critère 10
Critère 1	1,00	-0,27	-0,18	-0,23	-0,20	-0,24	-0,04	0,22	-0,08	-0,04
Critère 2	-0,27	1,00	0,18	0,05	-0,09	-0,03	0,20	-0,15	-0,26	-0,21
Critère 3	-0,18	0,18	1,00	0,15	-0,28	-0,11	0,15	0,01	0,16	0,03
Critère 4	-0,23	0,05	0,15	1,00	0,11	0,18	0,00	-0,14	0,07	-0,02
Critère 5	-0,20	-0,09	-0,28	0,11	1,00	0,11	-0,01	-0,19	-0,15	0,03
Critère 6	-0,24	-0,03	-0,11	0,18	0,11	1,00	-0,15	-0,08	0,01	-0,11
Critère 7	-0,04	0,20	0,15	0,00	-0,01	-0,15	1,00	-0,07	0,02	-0,04
Critère 8	0,22	-0,15	0,01	-0,14	-0,19	-0,08	-0,07	1,00	0,10	0,46
Critère 9	-0,08	-0,26	0,16	0,07	-0,15	0,01	0,02	0,10	1,00	0,30
Critère 10	-0,04	-0,21	0,03	-0,02	0,03	-0,11	-0,04	0,46	0,30	1,00

Tableau du pourcentage cumulé d'information des critères utilisés pour la caractérisation de la végétation reprise par chaque axe de l'analyse en composantes principales

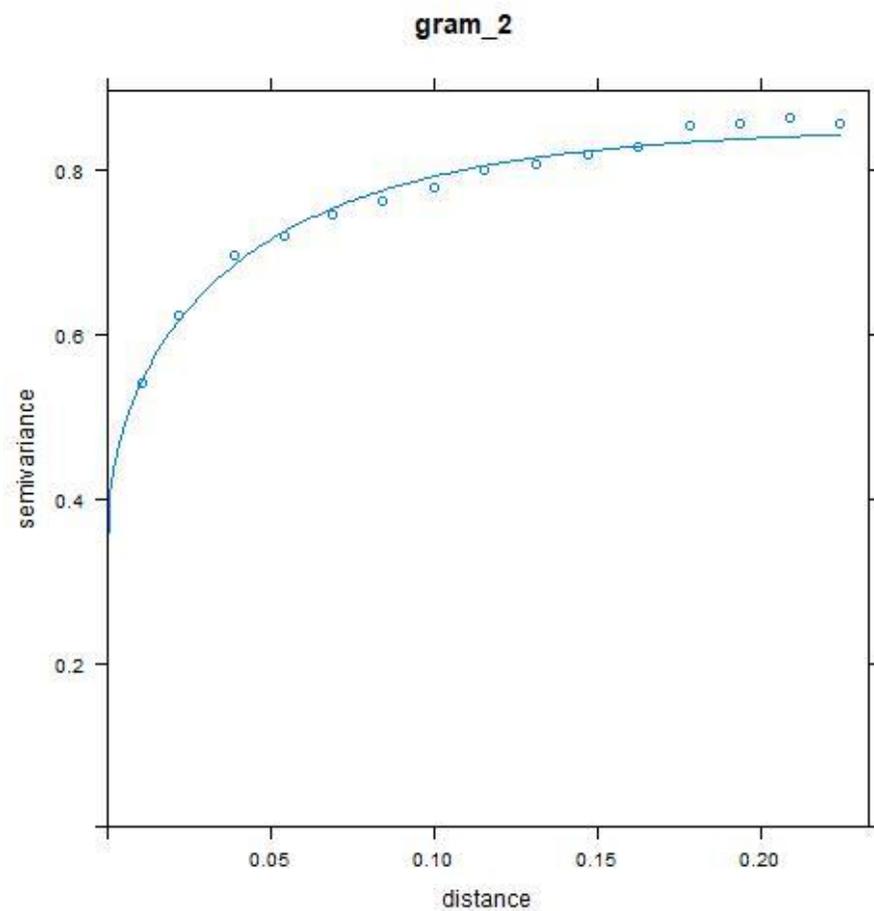
	Critère 1	Critère 2	Critère 3	Critère 4	Critère 5	Critère 6	Critère 7	Critère 8	Critère 9	Critère 10	Moyenne
Axe 1	50,5	43,7	5,1	16,8	3,6	11,4	2,3	27,6	6,2	17,2	18,4
Axe 2	73,6	55,4	11,5	33,9	4,2	19,7	3,0	31,5	52,7	50,6	33,6
Axe 3	73,7	76,8	54,3	35,3	33,2	50,9	13,6	38,0	56,8	55,5	48,8
Axe 4	84,2	82,1	66,3	51,0	48,3	51,7	13,6	48,4	63,7	81,0	59,0
Axe 5	84,5	85,3	66,5	58,4	72,0	94,0	18,0	61,3	63,7	81,1	68,5
Axe 6	90,5	87,0	66,5	96,9	72,0	94,1	18,6	77,2	78,7	86,3	76,8
Axe 7	93,3	96,8	89,4	98,7	72,0	94,1	25,5	77,5	96,3	86,5	83,0
Axe 8	95,0	97,1	93,6	99,4	80,0	98,9	84,1	77,8	96,5	86,6	90,9
Axe 9	99,8	99,9	99,8	100,0	92,8	99,5	99,1	81,3	97,8	88,3	95,8
Axe 10	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0	100,0

Annexe 26 : variogrammes des caractéristiques de la végétation

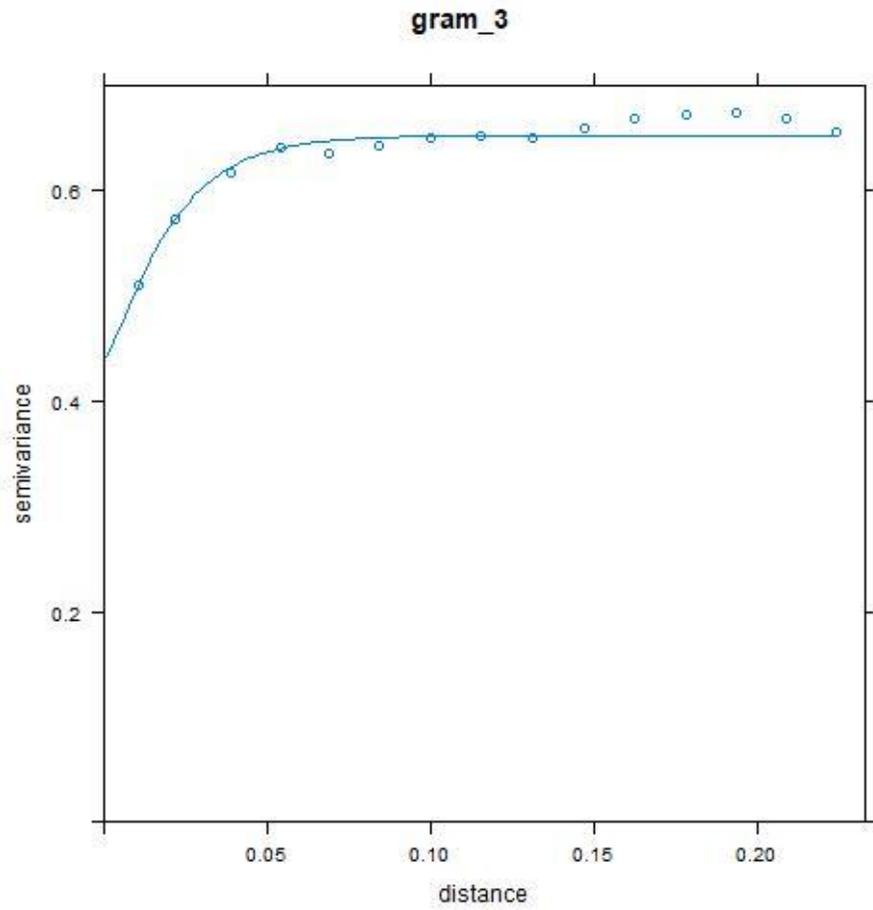
Critère 1 : graminées rampantes ou gazonnantes



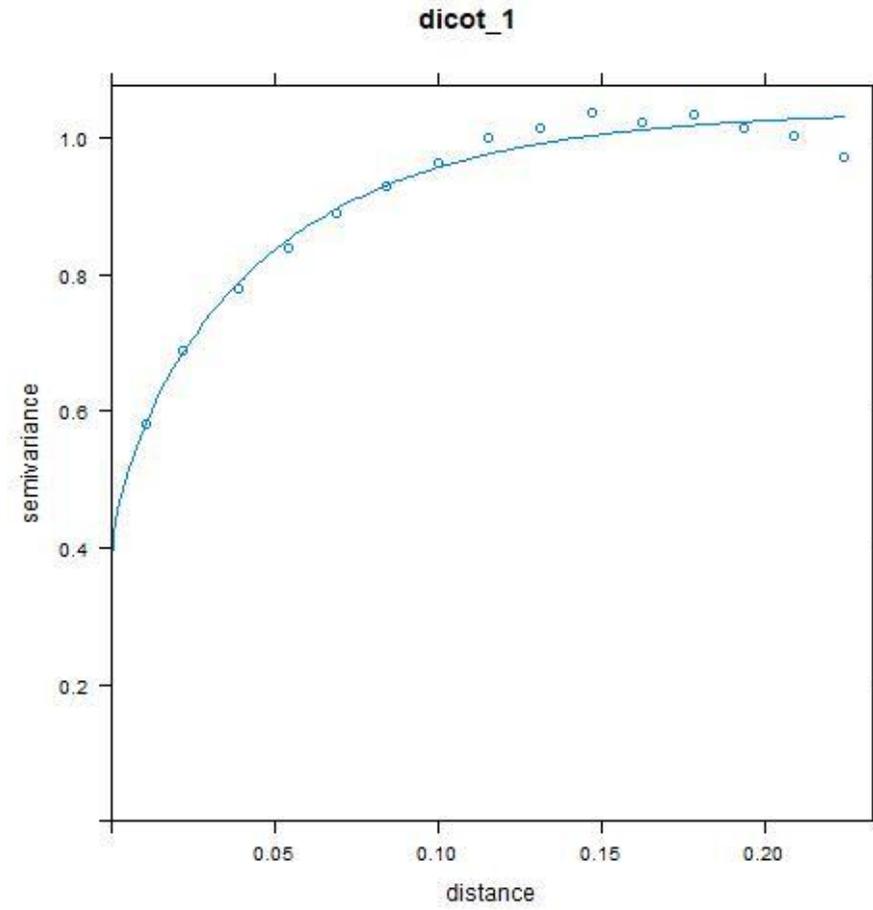
Critère 2 : graminées cespitueuses de moins de 1 m de haut



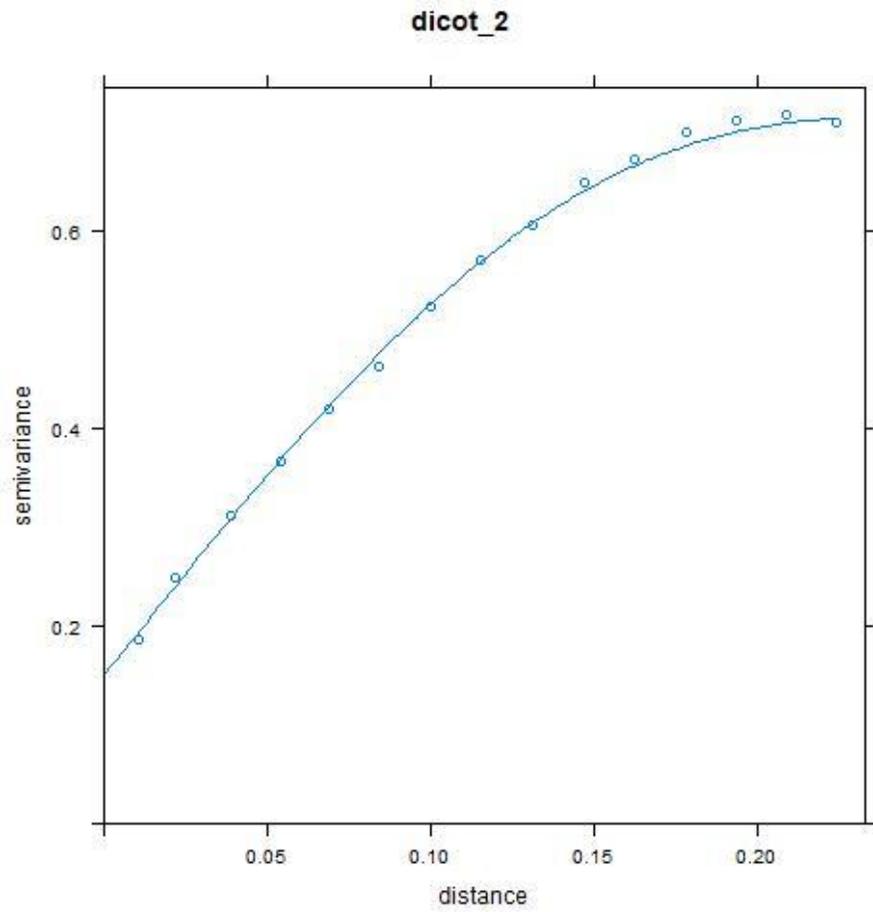
Critère 3 : graminées cespiteuses de plus de 1m de haut



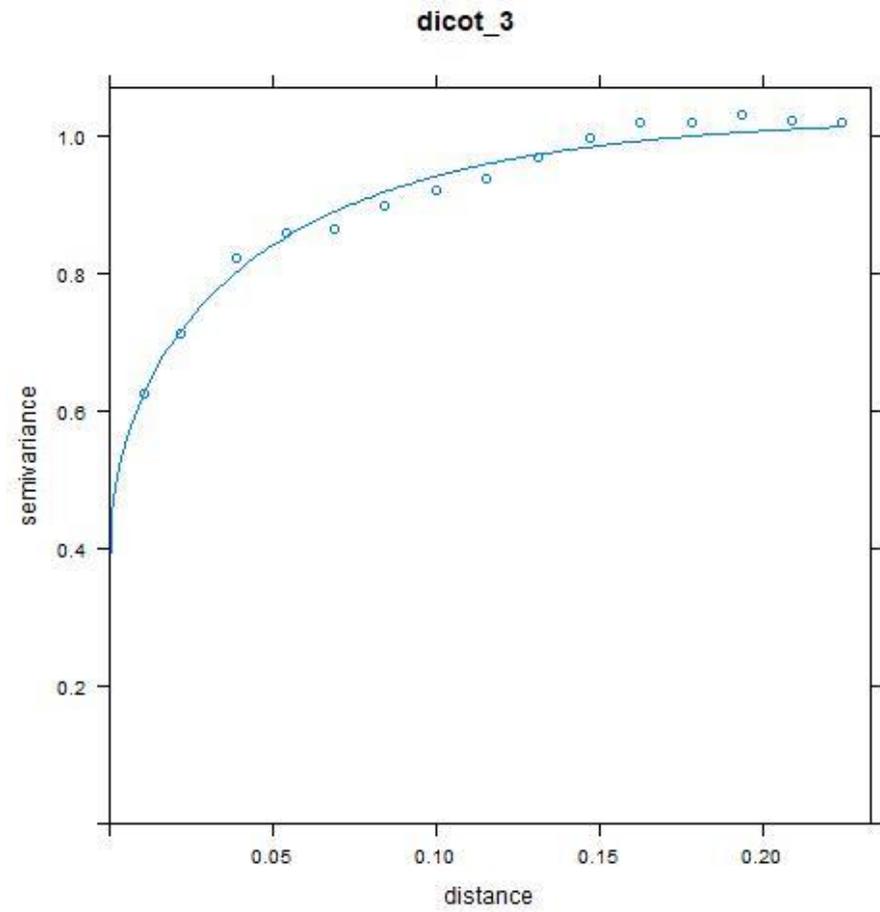
Critère 4 : *Calopogonium mucunoides*



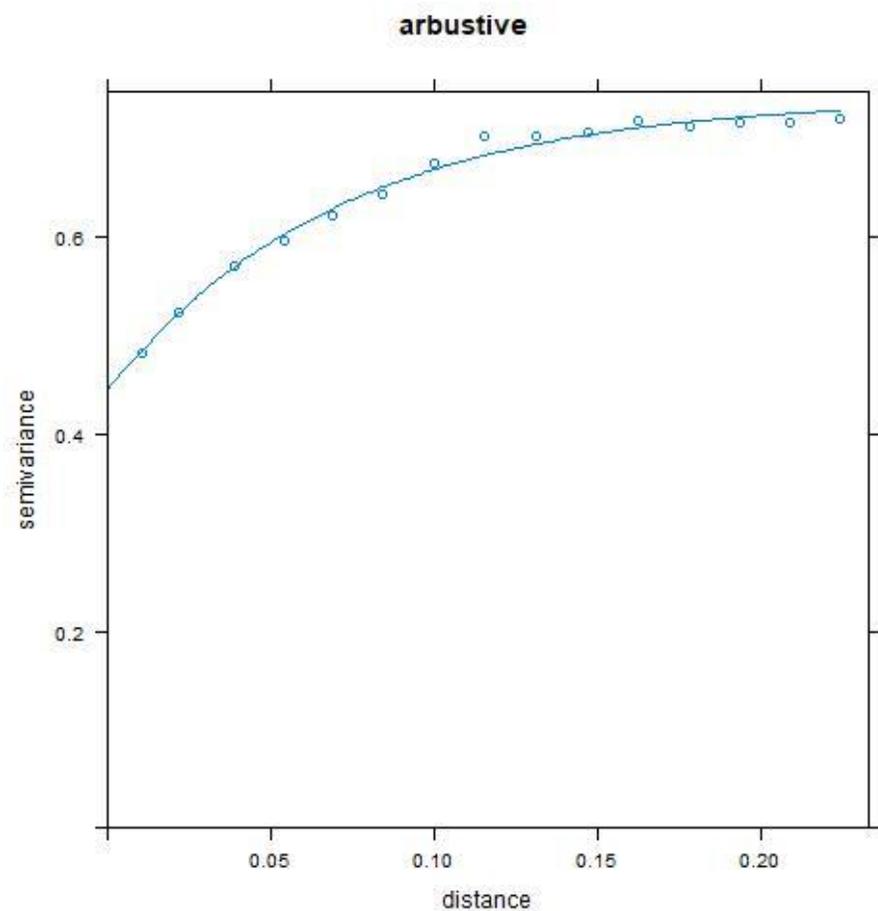
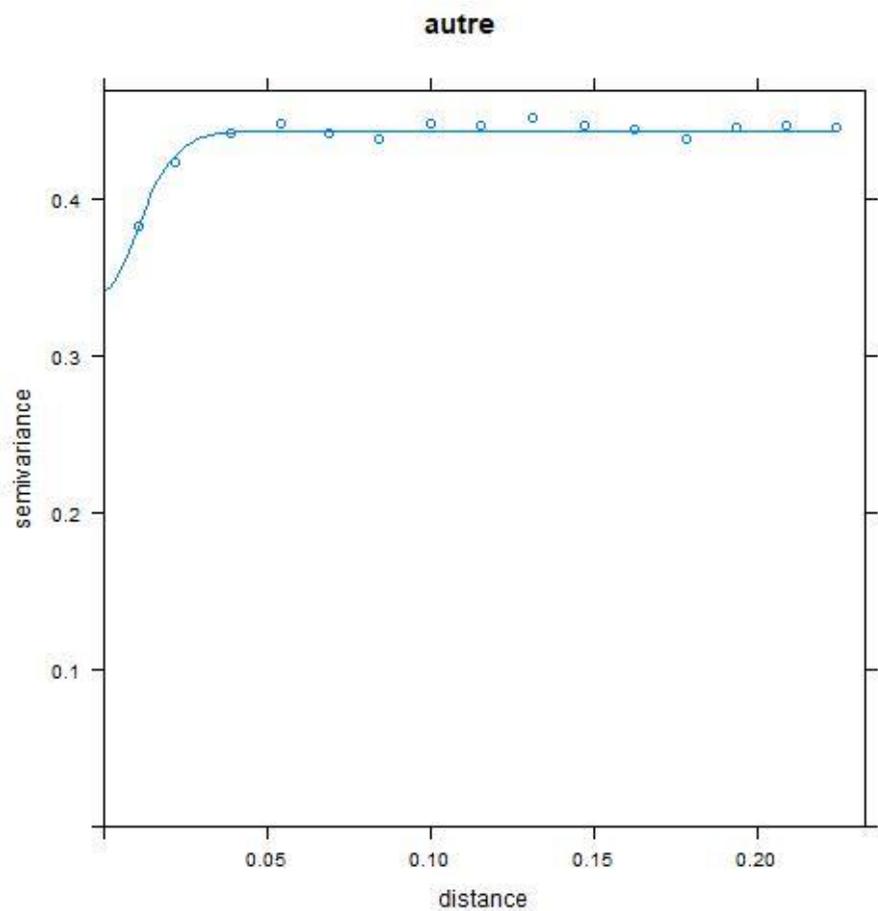
Critère 5 : *Sida cf. acuta*



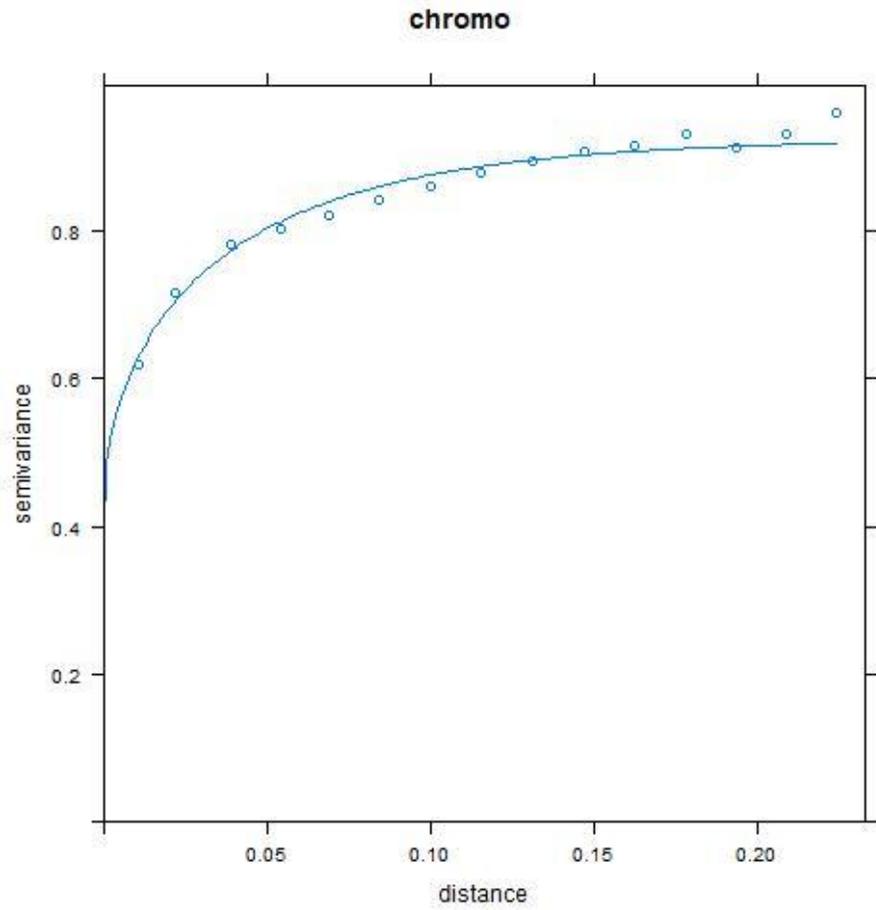
Critère 6 : *Mitracarpus hirtus*



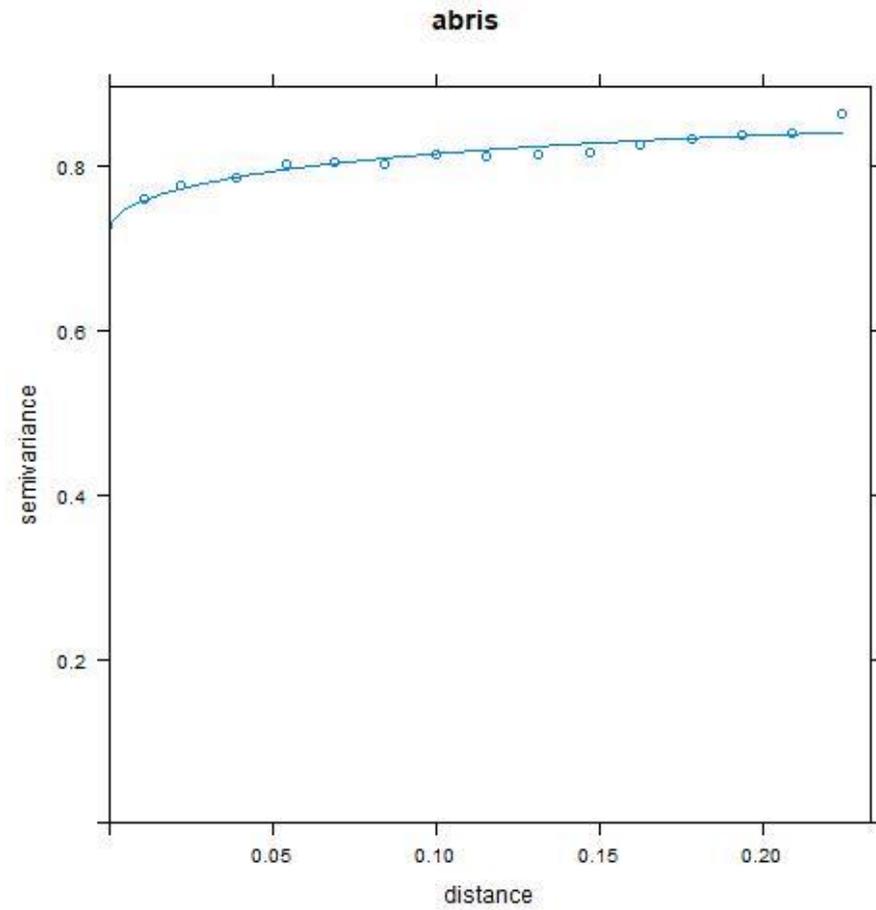
Critère 7 : végétation herbacée non reprise dans les 6 critères précédents Critère 8 : végétation arbustive à l'exception de *Chromolaena odorata*



Critère 9 : *Chromolaena odorata*

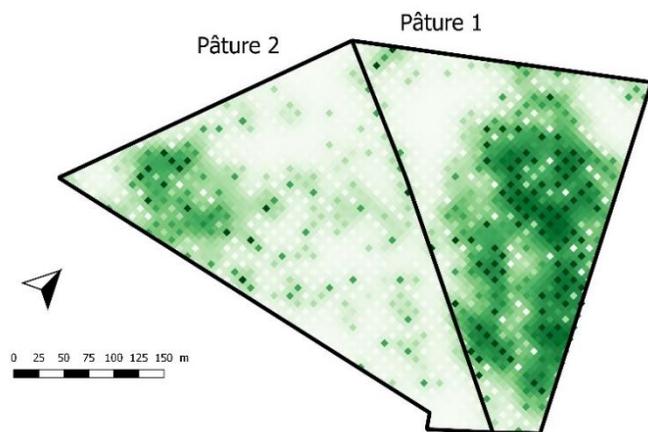


Critère 10 : abris contre le soleil et la pluie

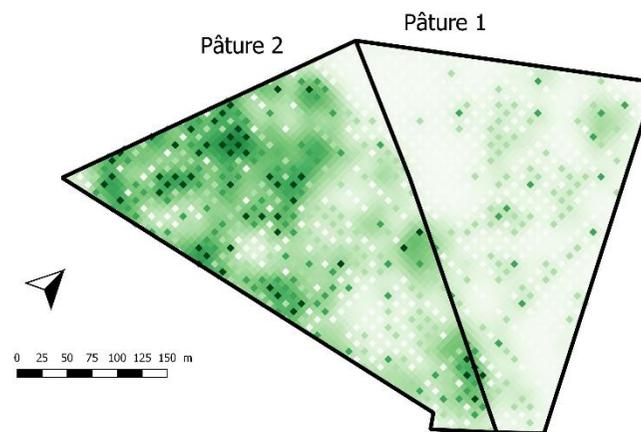


Annexe 27 : rasters des caractéristiques de la végétation

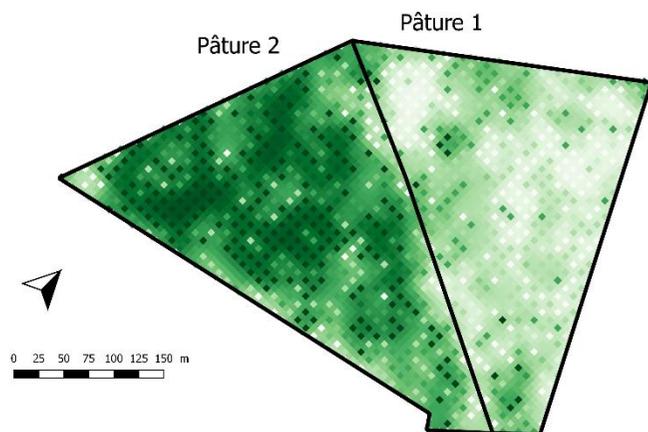
**Critère 1 :
graminées rampantes ou gazonnantes**



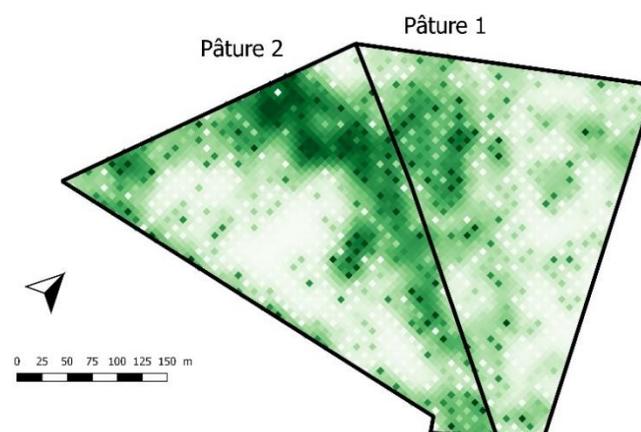
**Critère 3 :
graminées cespitueuses de plus de 1m**



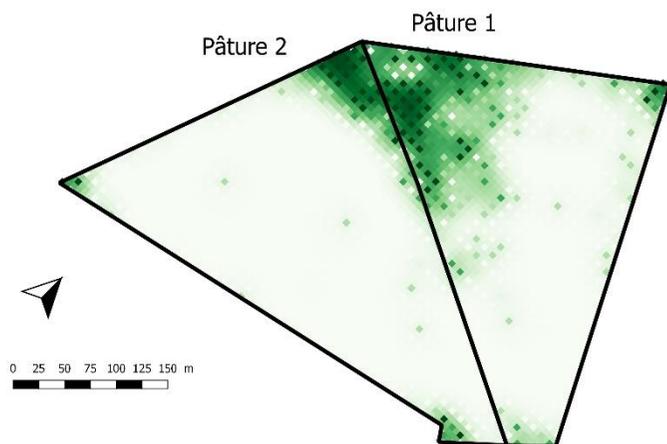
**Critère 2 :
graminées cespitueuses de moins de 1m**



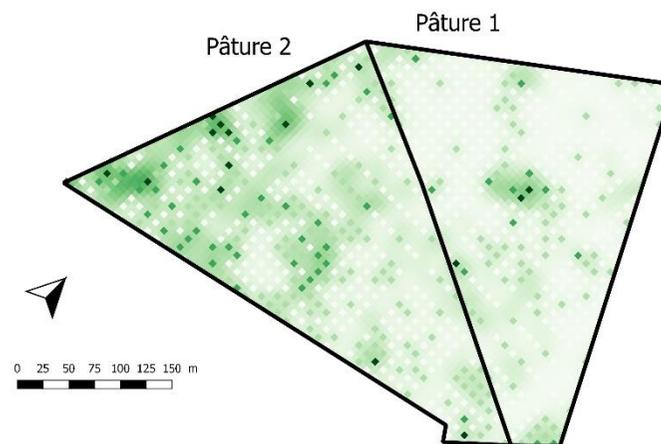
**Critère 4 :
Calopogonium mucunoides**



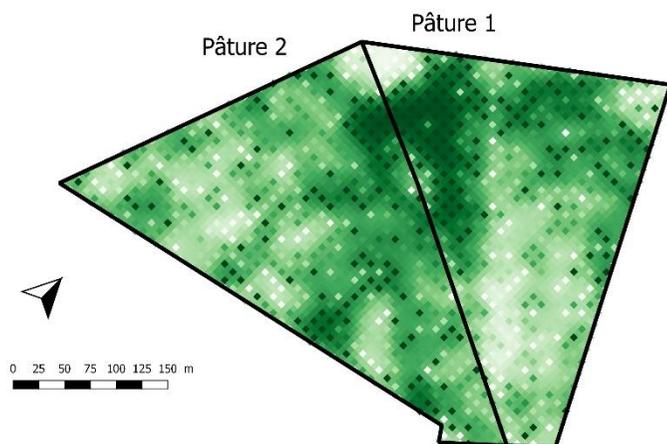
**Critère 5 :
*Sida cf. acuta***



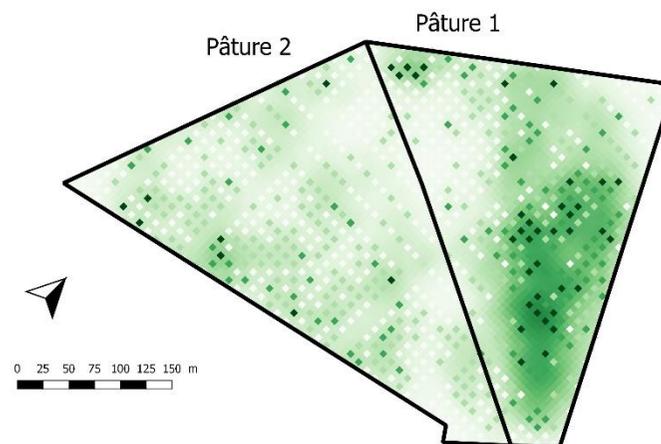
**Critère 7 :
végétation herbacée non reprise dans les 6 critères précédents**



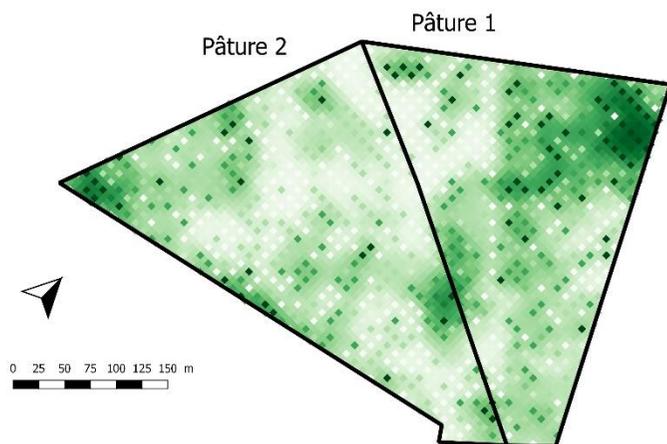
**Critère 6 :
*Mitracarpus hirtus***



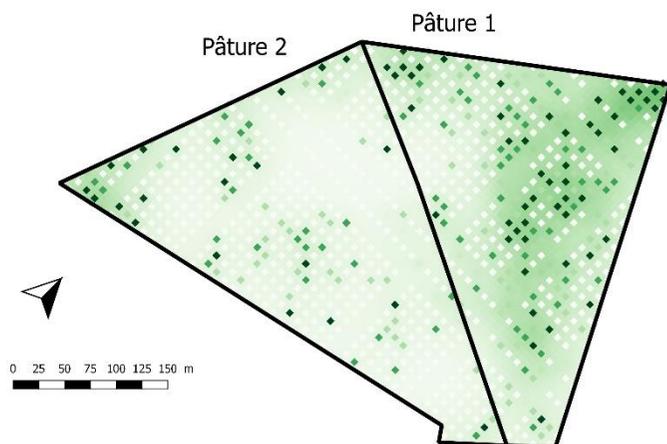
**Critère 8 :
végétation arbustive à l'exception de *Chromolaena odorata***



**Critère 9 :
Chromolaena odorata**



**Critère 10 :
abris contre le soleil et la pluie**



Annexe 28 : description du script du traitement géostatistique de la position des bêtes et de ses liens avec la végétation

1. Le répertoire est défini et les packages nécessaires au bon déroulement du script sont chargés.
2. Trois polygones sont définis après lecture de fichiers provenant de QGIS : un pour la pâture 1, un pour la pâture 2 et un pour la pâture totale.
3. Une grille est créée afin d'accueillir la sortie des krigeages. Elle est découpée à trois reprises, au moyen des polygones de l'étape 2, afin d'obtenir trois grilles : une pour la pâture 1, une pour la pâture 2 et une pour la pâture totale.
4. Les données relatives à la caractérisation de la végétation sont lues à partir d'un fichier provenant du logiciel QGIS et sont remaniées.
5. La matrice de corrélation des critères utilisés pour la caractérisation de la végétation est réalisée, suivie de l'analyse en composantes principales. Une table comprenant le pourcentage cumulé d'information reprise par chaque axe est créée. Cette table et la matrice de corrélation sont écrites au format « xlsx ».
6. Une fonction « krigeage de la végétation » est créée. Celle-ci accepte, en entrée, une variable caractérisant la végétation (format « variable » et format « chaîne de caractère »). Elle renvoie, en sortie, un tracé du variogramme utilisé pour krigeage, un tracé du raster résultant pour l'entièreté de la pâture, le raster résultant pour la pâture 1 et le raster résultant pour la pâture 2. Une fonction de lissage automatique du variogramme expérimental « variable en fonction de la latitude et de la longitude » est utilisée avec les modèles « Matern », « Matern avec les paramètres de M. Stein », « sphérique » et « penta-sphérique » afin d'obtenir le variogramme. Le variogramme obtenu est tracé. La fonction effectue ensuite un krigeage universel à partir des données relatives à la végétation, du variogramme, et de la grille de sortie « pâture totale » provenant de l'étape 3. Le résultat est transformé en raster. Ce raster est écrit puis est découpé et masqué une première fois pour la pâture 1 et une seconde fois pour la pâture 2.
7. La fonction est appliquée à chacune des variables caractérisant la végétation.
8. Les tracés des variogrammes sont écrits au format « jpg ».
9. Un buffer est réalisé autour de chaque point utilisé pour la caractérisation de la pâture afin d'effectuer un comptage de la présence des bêtes pendant une seconde.
10. Une fonction « détermination des variogrammes pour le krigeage de la présence des bêtes » est créée. Elle accepte, en entrée, le jeu de données à utiliser (format « chaîne de caractères ») et renvoie, en sortie, un variogramme et le tracé de ce variogramme. La fonction commence par lire le jeu de données. Elle effectue ensuite un comptage des points au moyen du jeu de buffers précédemment réalisé et réinjecte ces données en tant que nouvelle variable dans le jeu de données caractérisant la végétation. Une fonction de lissage automatique du variogramme expérimental « variable en fonction de la latitude et de la longitude » est utilisée avec les modèles « Matern », « Matern avec les paramètres de M. Stein », « sphérique » et « penta-sphérique » et un « nugget » fixé zéro afin d'obtenir le variogramme. Le variogramme obtenu est tracé.
11. Cette fonction est appliquée à 8 fichiers : compilation des positions des bêtes de jour en pâture 1 et en pâture 2, compilation des positions des bêtes de nuit en pâture 1 et en pâture

- 2, compilation des positions des bêtes « grazing » en pâture 1 et en pâture 2, compilation des positions des bêtes « non grazing » en pâture 1 et en pâture 2.
12. Les tracés des variogrammes sont écrits au format « jpg ».
 13. Une fonction « krigeage de la présence des bêtes » est créée. Elle accepte, en entrée, le jeu de données (format « chaine de caractères »), la grille de sortie et le variogramme à utiliser et renvoie, en sortie, le raster résultant et un tracé du raster résultant. La fonction commence par lire le jeu de données. Elle effectue, ensuite, un comptage des points au moyen du jeu de buffer précédemment réalisé et réinjecte ces données en tant que nouvelle variable dans le jeu de données caractérisant la végétation. La fonction effectue ensuite un krigeage universel à partir de cette variable, du variogramme et de la grille de sortie. Le résultat est alors transformé en raster et écrit.
 14. Cette fonction est appliquée à chacun des 64 fichiers correspondant à la position de chacune des 16 bêtes, de jour, de nuit, « grazing » et « non grazing ». Les fichiers sont accompagnés de la grille correspondante (pâture 1 ou 2), provenant de l'étape 3, et du variogramme correspondant, provenant de l'étape 11.
 15. Une trame de données vide est créée. Les lignes sont nommées en référence aux 16 bêtes de l'expérimentation et les trois premières colonnes sont nommées en référence à la pâture, à la race et à l'individu considéré. Les colonnes suivantes sont nommées selon les variables caractérisant la végétation, une première fois associées à la mention « jour », une deuxième fois à la mention « nuit », une troisième fois à la mention « grazing » et une quatrième fois à la mention « non-grazing ».
 16. Une fonction « corrélation » est créée. Elle accepte, en entrée, le raster associé à la position de la bête et le raster associé à la variable caractérisant la végétation et renvoie, en sortie, la corrélation de Pearson entre ces deux rasters.
 17. Cette fonction est appliquée à toutes les combinaisons d'un raster associé à la position d'une bête et d'un raster associé à une variable caractérisant la végétation. Les résultats obtenus sont attribués à l'emplacement adéquat dans la trame de données vide (étape 15).
 18. La trame de données des corrélations est écrite au format « xls ».
 19. Une trame de données vide est créée. Les lignes sont nommées selon les variables caractérisant la végétation, une première fois associées à la mention « jour », une deuxième fois à la mention « nuit », une troisième fois à la mention « grazing » et une quatrième fois à la mention « non-grazing ». Les colonnes sont nommées « moyenne globale », « écart-type global », « p-value pour l'interaction », « moyenne pour la pâture 1 », « écart-type pour la pâture 1 », « moyenne pour la pâture 2 », « écart-type pour la pâture 2 », « p-value pour la pâture », « moyenne pour la race ndama », « écart-type pour la race ndama », « moyenne pour la race zébu », « écart-type pour la race zébu », « moyenne pour la race ndapol », « écart-type pour la race ndapol », « moyenne pour la race zépol », « écart-type pour la race zépol », « p-value pour la race ».
 20. Une boucle est créée afin de lire la trame de données des corrélations (étape 17), variable à expliquer par variable à expliquer.
 21. Les moyennes et les écart-types sont calculés et, après avoir réalisé les analyses de la variance en fonction des facteurs fixes « pâture » et « race », les p-values sont récupérées.

Toutes ces valeurs sont attribuées à l'emplacement adéquat dans la nouvelle trame de données (étape 19)

22. La trame de données est écrite au format « xlsx ».
23. En cas d'interaction significative entre le facteur « pâture » et le facteur « race », le jeu de données est splitté selon le facteur jugé le moins pertinent et les moyennes, les écarts-types et les analyses de la variance à un facteur fixe sont réalisées en fonction de facteur jugé le plus pertinent. Les trames de données résultant de cette étape sont écrites au format « xlsx ».

Annexe 29 : script du traitement géostatistique de la position des bêtes et de ses liens avec la végétation

```

4183. ### traitement geostatistique
4184. ## intro
4185. # directory
4186. setwd("C:/Users/gauth/Desktop/TFE O/MAJ/R")
4187. # charger packages
4188. library("sp")
4189. library("ggplot2")
4190. library("tmap")
4191. library("raster")
4192. library("rgdal")
4193. library("ggmap")
4194. library("sf")
4195. library("dplyr")
4196. library("spatstat")
4197. library("gstat")
4198. library("FactoMineR")
4199. library("spdep")
4200. library("rgeos")
4201. library("cartogram")
4202. library("R2BayesX")
4203. library("automap")
4204. library("rasterVis")
4205. library("xlsx")
4206. ## patures
4207. pature <- readOGR("C:/Users/gauth/Desktop/TFE O/MAJ/R", "pature")
4208. pature1 <- subset(pature, id == 1)
4209. pature1 <- geometry(pature1)
4210. pature2 <- subset(pature, id == 2)
4211. pature2 <- geometry(pature2)
4212. patureTOT <- as(st_union(st_as_sf(pature)), "Spatial")
4213. ## grilles
4214. G1 <- GridTopology(round(bbox(patureTOT)[,1,4),
4215.                    c(0.00005,0.00005),c(400,400))
4216. G2 <- SpatialPoints(G1, proj4string = CRS(projection(patureTOT)))
4217. G3T <- crop(G2, patureTOT)
4218. G4T <- SpatialPixels(G3T)
4219. G31 <- crop(G2, pature1)
4220. G41 <- SpatialPixels(G31)
4221. G32 <- crop(G2, pature2)
4222. G42 <- SpatialPixels(G32)
4223. ## charger veg
4224. veg <- readOGR("C:/Users/gauth/Desktop/TFE O/MAJ/R", "veg")
4225. coordnames(veg) <- c("x", "y")
4226. ## ACP veg
4227. cor_veg <- cor(data.matrix(veg@data[4:13]))
4228. acp <- PCA(data.matrix(veg@data[4:13]),
4229.            scale.unit = FALSE, ncp = 10, graph = FALSE)

```

```

4230. veg_ACP <- veg
4231. veg_ACP@data[4:13] <- as.data.frame(acp$ind$coord)
4232. names(veg_ACP@data) <- c("pature", "latitude", "longitude",
4233.                        "dim1", "dim2", "dim3", "dim4", "dim5",
4234.                        "dim6", "dim7", "dim8", "dim9", "dim10")
4235. choix_dim <- apply(acp$var$cos2,1,cumsum)
4236. write.xlsx(cor_veg, "ACP_cor_veg.xlsx")
4237. write.xlsx(choix_dim, "ACP_info_dim.xlsx")
4238. # ! STOP : ACP non pertinente !
4239. ## krigeage veg
4240. # fonction krigeage veg
4241. fct_KV <- function(critere, string) {
4242.   vario1 <- variogram(critere ~ x + y, veg)
4243.   vario2 <- autofitVariogram(critere ~ x + y, veg,
4244.                             model = c("Mat", "Ste", "Sph", "Pen"))
4245.   plot1 <- plot(vario1, model = vario2$var_model, main = string)
4246.   pred <- krige(critere ~ x + y, veg, newdata = G4T, model = vario2$var_model)
4247.   raster <- raster(pred)
4248.   plot2 <- levelplot(raster, margin = FALSE, main = string)
4249.   writeRaster(raster, paste("raster_", string, sep = ""),
4250.              format = "GTiff", overwrite = TRUE)
4251.   raster_P1 <- mask(crop(raster, pature1),pature1)
4252.   raster_P2 <- mask(crop(raster, pature2),pature2)
4253.   return(list(plot1, plot2, raster_P1, raster_P2))
4254. }
4255. # krigeage veg + sortie rasters
4256. gram_1 <- fct_KV(veg$gram_1, "gram_1")
4257. plot_vario_gram_1 <- gram_1[[1]]
4258. plot_raster_gram_1 <- gram_1[[2]]
4259. gram_1_P1 <- gram_1[[3]]
4260. gram_1_P2 <- gram_1[[4]]
4261. gram_2 <- fct_KV(veg$gram_2, "gram_2")
4262. plot_vario_gram_2 <- gram_2[[1]]
4263. plot_raster_gram_2 <- gram_2[[2]]
4264. gram_2_P1 <- gram_2[[3]]
4265. gram_2_P2 <- gram_2[[4]]
4266. gram_3 <- fct_KV(veg$gram_3, "gram_3")
4267. plot_vario_gram_3 <- gram_3[[1]]
4268. plot_raster_gram_3 <- gram_3[[2]]
4269. gram_3_P1 <- gram_3[[3]]
4270. gram_3_P2 <- gram_3[[4]]
4271. dicot_1 <- fct_KV(veg$dicot_1, "dicot_1")
4272. plot_vario_dicot_1 <- dicot_1[[1]]
4273. plot_raster_dicot_1 <- dicot_1[[2]]
4274. dicot_1_P1 <- dicot_1[[3]]
4275. dicot_1_P2 <- dicot_1[[4]]
4276. dicot_2 <- fct_KV(veg$dicot_2, "dicot_2")
4277. plot_vario_dicot_2 <- dicot_2[[1]]
4278. plot_raster_dicot_2 <- dicot_2[[2]]
4279. dicot_2_P1 <- dicot_2[[3]]
4280. dicot_2_P2 <- dicot_2[[4]]
4281. dicot_3 <- fct_KV(veg$dicot_3, "dicot_3")

```

```

4282. plot_vario_dicot_3 <- dicot_3[[1]]
4283. plot_raster_dicot_3 <- dicot_3[[2]]
4284. dicot_3_P1 <- dicot_3[[3]]
4285. dicot_3_P2 <- dicot_3[[4]]
4286. autre <- fct_KV(veg$autre, "autre")
4287. plot_vario_autre <- autre[[1]]
4288. plot_raster_autre <- autre[[2]]
4289. autre_P1 <- autre[[3]]
4290. autre_P2 <- autre[[4]]
4291. arbustive <- fct_KV(veg$arbustive, "arbustive")
4292. plot_vario_arbustive <- arbustive[[1]]
4293. plot_raster_arbustive <- arbustive[[2]]
4294. arbustive_P1 <- arbustive[[3]]
4295. arbustive_P2 <- arbustive[[4]]
4296. chromo <- fct_KV(veg$chromo, "chromo")
4297. plot_vario_chromo <- chromo[[1]]
4298. plot_raster_chromo <- chromo[[2]]
4299. chromo_P1 <- chromo[[3]]
4300. chromo_P2 <- chromo[[4]]
4301. abris <- fct_KV(veg$abris, "abris")
4302. plot_vario_abris <- abris[[1]]
4303. plot_raster_abris <- abris[[2]]
4304. abris_P1 <- abris[[3]]
4305. abris_P2 <- abris[[4]]
4306. # sortie variogrammes
4307. jpeg(filename = "vario_gram_1.jpg")
4308. plot_vario_gram_1
4309. dev.off()
4310. jpeg(filename = "vario_gram_2.jpg")
4311. plot_vario_gram_2
4312. dev.off()
4313. jpeg(filename = "vario_gram_3.jpg")
4314. plot_vario_gram_3
4315. dev.off()
4316. jpeg(filename = "vario_dicot_1.jpg")
4317. plot_vario_dicot_1
4318. dev.off()
4319. jpeg(filename = "vario_dicot_2.jpg")
4320. plot_vario_dicot_2
4321. dev.off()
4322. jpeg(filename = "vario_dicot_3.jpg")
4323. plot_vario_dicot_3
4324. dev.off()
4325. jpeg(filename = "vario_autre.jpg")
4326. plot_vario_autre
4327. dev.off()
4328. jpeg(filename = "vario_arbustive.jpg")
4329. plot_vario_arbustive
4330. dev.off()
4331. jpeg(filename = "vario_chromo.jpg")
4332. plot_vario_chromo
4333. dev.off()

4334. jpeg(filename = "vario_abris.jpg")
4335. plot_vario_abris
4336. dev.off()
4337. ## comptage + krigeage betes
4338. # buffer
4339. veg2 <- st_as_sf(geometry(veg))
4340. buffer1 <- st_buffer(veg2, dist = 0.00005, endCapStyle = "SQUARE")
4341. buffer2 <- st_centroid(buffer1[[1]])
4342. buffer3 <- buffer1
4343. buffer3$geometry <- buffer2
4344. # fonction variogramme
4345. setTimeLimit(cpu = Inf, elapsed = Inf, transient = FALSE)
4346. fct_V <- function(string) {
4347.   bete <- readOGR("C:/Users/gauth/Desktop/TFE O/MAJ/R", string)
4348.   sf <- st_as_sf(geometry(bete))
4349.   compt1 <- st_contains(buffer1, sf)
4350.   compt2 <- lapply(compt1, length)
4351.   tot <- sum(unlist(compt2))
4352.   compt3 <- lapply(compt2, function(x) {1000*x/tot})
4353.   compt4 <- mutate(buffer3, compt3)
4354.   compt5 <- as(compt4, "Spatial")
4355.   veg_bete <- veg
4356.   veg_bete@data$compt <- unlist(compt5@data$compt3)
4357.   vario1 <- variogram(compt ~ x + y, veg_bete)
4358.   vario2 <- autofitVariogram(compt ~ x + y, veg_bete,
4359.     model = c("Mat", "Ste", "Sph", "Pen"), fix.values = c(0, NA, NA))
4360.   plot <- plot(vario1, model = vario2$var_model, main = string)
4361.   return(list(vario2, plot))
4362. }
4363. # variogrammes
4364. vario_j1 <- fct_V("compil_j1")
4365. plot_vario_j1 <- vario_j1[[2]]
4366. vario_j1 <- vario_j1[[1]]
4367. vario_j2 <- fct_V("compil_j2")
4368. plot_vario_j2 <- vario_j2[[2]]
4369. vario_j2 <- vario_j2[[1]]
4370. vario_n1 <- fct_V("compil_n1")
4371. plot_vario_n1 <- vario_n1[[2]]
4372. vario_n1 <- vario_n1[[1]]
4373. vario_n2 <- fct_V("compil_n2")
4374. plot_vario_n2 <- vario_n2[[2]]
4375. vario_n2 <- vario_n2[[1]]
4376. vario_g1 <- fct_V("compil_g1")
4377. plot_vario_g1 <- vario_g1[[2]]
4378. vario_g1 <- vario_g1[[1]]
4379. vario_g2 <- fct_V("compil_g2")
4380. plot_vario_g2 <- vario_g2[[2]]
4381. vario_g2 <- vario_g2[[1]]
4382. vario_ng1 <- fct_V("compil_ng1")
4383. plot_vario_ng1 <- vario_ng1[[2]]
4384. vario_ng1 <- vario_ng1[[1]]
4385. vario_ng2 <- fct_V("compil_ng2")

```

```

4386. plot_vario_ng2 <- vario_ng2[[2]]
4387. vario_ng2 <- vario_ng2[[1]]
4388. # sortie variogrammes
4389. jpeg(filename = "vario_j1.jpg")
4390. plot_vario_j1
4391. dev.off()
4392. jpeg(filename = "vario_j2.jpg")
4393. plot_vario_j2
4394. dev.off()
4395. jpeg(filename = "vario_n1.jpg")
4396. plot_vario_n1
4397. dev.off()
4398. jpeg(filename = "vario_n2.jpg")
4399. plot_vario_n2
4400. dev.off()
4401. jpeg(filename = "vario_g1.jpg")
4402. plot_vario_g1
4403. dev.off()
4404. jpeg(filename = "vario_g2.jpg")
4405. plot_vario_g2
4406. dev.off()
4407. jpeg(filename = "vario_ng1.jpg")
4408. plot_vario_ng1
4409. dev.off()
4410. jpeg(filename = "vario_ng2.jpg")
4411. plot_vario_ng2
4412. dev.off()
4413. # fonction betes
4414. fct_KB <- function(string, grille, vario) {
4415.   bete <- readOGR("C:/Users/gauth/Desktop/TFE O/MAJ/R", string)
4416.   sf <- st_as_sf(geometry(bete))
4417.   compt1 <- st_contains(buffer1, sf)
4418.   compt2 <- lapply(compt1, length)
4419.   tot <- sum(unlist(compt2))
4420.   compt3 <- lapply(compt2, function(x) {1000*x/tot})
4421.   compt4 <- mutate(buffer3, compt3)
4422.   compt5 <- as(compt4, "Spatial")
4423.   veg_bete <- veg
4424.   veg_bete@data$compt <- unlist(compt5@data$compt3)
4425.   pred <- krige(compt ~ x + y, veg_bete,
4426.     newdata = grille, model = vario$var_model)
4427.   raster <- raster(pred)
4428.   plot <- levelplot(raster, margin = FALSE, main = string)
4429.   writeRaster(raster, paste("raster_", string, sep = ""),
4430.     format = "GTiff", overwrite = TRUE)
4431.   return(list(raster, plot))
4432. }
4433. # comptage + krigeage betes
4434. # jour
4435. i111j <- fct_KB("iphone_111_j", G41, vario_j1)
4436. plot_raster_i111j <- i111j[[2]]
4437. i111j <- i111j[[1]]
4438. i112j <- fct_KB("iphone_112_j", G41, vario_j1)
4439. plot_raster_i112j <- i112j[[2]]
4440. i112j <- i112j[[1]]
4441. i121j <- fct_KB("iphone_121_j", G41, vario_j1)
4442. plot_raster_i121j <- i121j[[2]]
4443. i121j <- i121j[[1]]
4444. i122j <- fct_KB("iphone_122_j", G41, vario_j1)
4445. plot_raster_i122j <- i122j[[2]]
4446. i122j <- i122j[[1]]
4447. i131j <- fct_KB("iphone_131_j", G41, vario_j1)
4448. plot_raster_i131j <- i131j[[2]]
4449. i131j <- i131j[[1]]
4450. i132j <- fct_KB("iphone_132_j", G41, vario_j1)
4451. plot_raster_i132j <- i132j[[2]]
4452. i132j <- i132j[[1]]
4453. i141j <- fct_KB("iphone_141_j", G41, vario_j1)
4454. plot_raster_i141j <- i141j[[2]]
4455. i141j <- i141j[[1]]
4456. i142j <- fct_KB("iphone_142_j", G41, vario_j1)
4457. plot_raster_i142j <- i142j[[2]]
4458. i142j <- i142j[[1]]
4459. i211j <- fct_KB("iphone_211_j", G42, vario_j2)
4460. plot_raster_i211j <- i211j[[2]]
4461. i211j <- i211j[[1]]
4462. i212j <- fct_KB("iphone_212_j", G42, vario_j2)
4463. plot_raster_i212j <- i212j[[2]]
4464. i212j <- i212j[[1]]
4465. i221j <- fct_KB("iphone_221_j", G42, vario_j2)
4466. plot_raster_i221j <- i221j[[2]]
4467. i221j <- i221j[[1]]
4468. i222j <- fct_KB("iphone_222_j", G42, vario_j2)
4469. plot_raster_i222j <- i222j[[2]]
4470. i222j <- i222j[[1]]
4471. i231j <- fct_KB("iphone_231_j", G42, vario_j2)
4472. plot_raster_i231j <- i231j[[2]]
4473. i231j <- i231j[[1]]
4474. i232j <- fct_KB("iphone_232_j", G42, vario_j2)
4475. plot_raster_i232j <- i232j[[2]]
4476. i232j <- i232j[[1]]
4477. i241j <- fct_KB("iphone_241_j", G42, vario_j2)
4478. plot_raster_i241j <- i241j[[2]]
4479. i241j <- i241j[[1]]
4480. i242j <- fct_KB("iphone_242_j", G42, vario_j2)
4481. plot_raster_i242j <- i242j[[2]]
4482. i242j <- i242j[[1]]
4483. # nuit
4484. i111n <- fct_KB("iphone_111_n", G41, vario_n1)
4485. plot_raster_i111n <- i111n[[2]]
4486. i111n <- i111n[[1]]
4487. i112n <- fct_KB("iphone_112_n", G41, vario_n1)
4488. plot_raster_i112n <- i112n[[2]]
4489. i112n <- i112n[[1]]

```

```

4490. i121n <- fct_KB("iphone_121_n", G41, vario_n1)
4491. plot_raster_i121n <- i121n[[2]]
4492. i121n <- i121n[[1]]
4493. i122n <- fct_KB("iphone_122_n", G41, vario_n1)
4494. plot_raster_i122n <- i122n[[2]]
4495. i122n <- i122n[[1]]
4496. i131n <- fct_KB("iphone_131_n", G41, vario_n1)
4497. plot_raster_i131n <- i131n[[2]]
4498. i131n <- i131n[[1]]
4499. i132n <- fct_KB("iphone_132_n", G41, vario_n1)
4500. plot_raster_i132n <- i132n[[2]]
4501. i132n <- i132n[[1]]
4502. i141n <- fct_KB("iphone_141_n", G41, vario_n1)
4503. plot_raster_i141n <- i141n[[2]]
4504. i141n <- i141n[[1]]
4505. i142n <- fct_KB("iphone_142_n", G41, vario_n1)
4506. plot_raster_i142n <- i142n[[2]]
4507. i142n <- i142n[[1]]
4508. i211n <- fct_KB("iphone_211_n", G42, vario_n2)
4509. plot_raster_i211n <- i211n[[2]]
4510. i211n <- i211n[[1]]
4511. i212n <- fct_KB("iphone_212_n", G42, vario_n2)
4512. plot_raster_i212n <- i212n[[2]]
4513. i212n <- i212n[[1]]
4514. i221n <- fct_KB("iphone_221_n", G42, vario_n2)
4515. plot_raster_i221n <- i221n[[2]]
4516. i221n <- i221n[[1]]
4517. i222n <- fct_KB("iphone_222_n", G42, vario_n2)
4518. plot_raster_i222n <- i222n[[2]]
4519. i222n <- i222n[[1]]
4520. i231n <- fct_KB("iphone_231_n", G42, vario_n2)
4521. plot_raster_i231n <- i231n[[2]]
4522. i231n <- i231n[[1]]
4523. i232n <- fct_KB("iphone_232_n", G42, vario_n2)
4524. plot_raster_i232n <- i232n[[2]]
4525. i232n <- i232n[[1]]
4526. i241n <- fct_KB("iphone_241_n", G42, vario_n2)
4527. plot_raster_i241n <- i241n[[2]]
4528. i241n <- i241n[[1]]
4529. i242n <- fct_KB("iphone_242_n", G42, vario_n2)
4530. plot_raster_i242n <- i242n[[2]]
4531. i242n <- i242n[[1]]
4532. # grazing
4533. i111g <- fct_KB("iphone_111_g", G41, vario_g1)
4534. plot_raster_i111g <- i111g[[2]]
4535. i111g <- i111g[[1]]
4536. i112g <- fct_KB("iphone_112_g", G41, vario_g1)
4537. plot_raster_i112g <- i112g[[2]]
4538. i112g <- i112g[[1]]
4539. i121g <- fct_KB("iphone_121_g", G41, vario_g1)
4540. plot_raster_i121g <- i121g[[2]]
4541. i121g <- i121g[[1]]

```

```

4542. i122g <- fct_KB("iphone_122_g", G41, vario_g1)
4543. plot_raster_i122g <- i122g[[2]]
4544. i122g <- i122g[[1]]
4545. i131g <- fct_KB("iphone_131_g", G41, vario_g1)
4546. plot_raster_i131g <- i131g[[2]]
4547. i131g <- i131g[[1]]
4548. i132g <- fct_KB("iphone_132_g", G41, vario_g1)
4549. plot_raster_i132g <- i132g[[2]]
4550. i132g <- i132g[[1]]
4551. i141g <- fct_KB("iphone_141_g", G41, vario_g1)
4552. plot_raster_i141g <- i141g[[2]]
4553. i141g <- i141g[[1]]
4554. i142g <- fct_KB("iphone_142_g", G41, vario_g1)
4555. plot_raster_i142g <- i142g[[2]]
4556. i142g <- i142g[[1]]
4557. i211g <- fct_KB("iphone_211_g", G42, vario_g2)
4558. plot_raster_i211g <- i211g[[2]]
4559. i211g <- i211g[[1]]
4560. i212g <- fct_KB("iphone_212_g", G42, vario_g2)
4561. plot_raster_i212g <- i212g[[2]]
4562. i212g <- i212g[[1]]
4563. i221g <- fct_KB("iphone_221_g", G42, vario_g2)
4564. plot_raster_i221g <- i221g[[2]]
4565. i221g <- i221g[[1]]
4566. i222g <- fct_KB("iphone_222_g", G42, vario_g2)
4567. plot_raster_i222g <- i222g[[2]]
4568. i222g <- i222g[[1]]
4569. i231g <- fct_KB("iphone_231_g", G42, vario_g2)
4570. plot_raster_i231g <- i231g[[2]]
4571. i231g <- i231g[[1]]
4572. i232g <- fct_KB("iphone_232_g", G42, vario_g2)
4573. plot_raster_i232g <- i232g[[2]]
4574. i232g <- i232g[[1]]
4575. i241g <- fct_KB("iphone_241_g", G42, vario_g2)
4576. plot_raster_i241g <- i241g[[2]]
4577. i241g <- i241g[[1]]
4578. i242g <- fct_KB("iphone_242_g", G42, vario_g2)
4579. plot_raster_i242g <- i242g[[2]]
4580. i242g <- i242g[[1]]
4581. # non grazing
4582. i111ng <- fct_KB("iphone_111_ng", G41, vario_ng1)
4583. plot_raster_i111ng <- i111ng[[2]]
4584. i111ng <- i111ng[[1]]
4585. i112ng <- fct_KB("iphone_112_ng", G41, vario_ng1)
4586. plot_raster_i112ng <- i112ng[[2]]
4587. i112ng <- i112ng[[1]]
4588. i121ng <- fct_KB("iphone_121_ng", G41, vario_ng1)
4589. plot_raster_i121ng <- i121ng[[2]]
4590. i121ng <- i121ng[[1]]
4591. i122ng <- fct_KB("iphone_122_ng", G41, vario_ng1)
4592. plot_raster_i122ng <- i122ng[[2]]
4593. i122ng <- i122ng[[1]]

```

```

4594. i131ng <- fct_KB("iphone_131_ng", G41, vario_ng1)
4595. plot_raster_i131ng <- i131ng[[2]]
4596. i131ng <- i131ng[[1]]
4597. i132ng <- fct_KB("iphone_132_ng", G41, vario_ng1)
4598. plot_raster_i132ng <- i132ng[[2]]
4599. i132ng <- i132ng[[1]]
4600. i141ng <- fct_KB("iphone_141_ng", G41, vario_ng1)
4601. plot_raster_i141ng <- i141ng[[2]]
4602. i141ng <- i141ng[[1]]
4603. i142ng <- fct_KB("iphone_142_ng", G41, vario_ng1)
4604. plot_raster_i142ng <- i142ng[[2]]
4605. i142ng <- i142ng[[1]]
4606. i211ng <- fct_KB("iphone_211_ng", G42, vario_ng2)
4607. plot_raster_i211ng <- i211ng[[2]]
4608. i211ng <- i211ng[[1]]
4609. i212ng <- fct_KB("iphone_212_ng", G42, vario_ng2)
4610. plot_raster_i212ng <- i212ng[[2]]
4611. i212ng <- i212ng[[1]]
4612. i221ng <- fct_KB("iphone_221_ng", G42, vario_ng2)
4613. plot_raster_i221ng <- i221ng[[2]]
4614. i221ng <- i221ng[[1]]
4615. i222ng <- fct_KB("iphone_222_ng", G42, vario_ng2)
4616. plot_raster_i222ng <- i222ng[[2]]
4617. i222ng <- i222ng[[1]]
4618. i231ng <- fct_KB("iphone_231_ng", G42, vario_ng2)
4619. plot_raster_i231ng <- i231ng[[2]]
4620. i231ng <- i231ng[[1]]
4621. i232ng <- fct_KB("iphone_232_ng", G42, vario_ng2)
4622. plot_raster_i232ng <- i232ng[[2]]
4623. i232ng <- i232ng[[1]]
4624. i241ng <- fct_KB("iphone_241_ng", G42, vario_ng2)
4625. plot_raster_i241ng <- i241ng[[2]]
4626. i241ng <- i241ng[[1]]
4627. i242ng <- fct_KB("iphone_242_ng", G42, vario_ng2)
4628. plot_raster_i242ng <- i242ng[[2]]
4629. i242ng <- i242ng[[1]]
4630. ## test correlation
4631. # creer dataframe
4632. # NB : 16 individus, 3 facteurs + 40 variables
4633. DF_cor <- data.frame(c(rep(1,8),rep(2,8)),
4634.   rep(c(1,1,2,2,3,3,4,4),2), rep(c(1,2),8),
4635.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16),
4636.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16),
4637.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16),
4638.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16),
4639.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16),
4640.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16),
4641.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16),
4642.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16),
4643.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16),
4644.   rep(NA,16), rep(NA,16), rep(NA,16), rep(NA,16))
4645. colnames(DF_cor) <- c("pature", "race", "bete",

```

```

4646.   "gram1_j", "gram2_j", "gram3_j", "dicot1_j", "dicot2_j",
4647.   "dicot3_j", "autre_j", "arbustive_j", "chromo_j", "abris_j",
4648.   "gram1_n", "gram2_n", "gram3_n", "dicot1_n", "dicot2_n",
4649.   "dicot3_n", "autre_n", "arbustive_n", "chromo_n", "abris_n",
4650.   "gram1_g", "gram2_g", "gram3_g", "dicot1_g", "dicot2_g",
4651.   "dicot3_g", "autre_g", "arbustive_g", "chromo_g", "abris_g",
4652.   "gram1_ng", "gram2_ng", "gram3_ng", "dicot1_ng", "dicot2_ng",
4653.   "dicot3_ng", "autre_ng", "arbustive_ng", "chromo_ng", "abris_ng")
4654. rownames(DF_cor) <- c("1NA", "1NB", "1ZA", "1ZB", "1NpA", "1NpB", "1ZpA",
4655.   "1ZpB", "2NA", "2NB", "2ZA", "2ZB", "2NpA", "2NpB", "2ZpA", "2ZpB")
4656. # fonction correlation
4657. fct_cor <- function(bete, critere) {
4658.   critere <- mask(crop(critere, bete), bete)
4659.   brick <- brick(list(bete, critere))
4660.   cor1 <- layerStats(brick, 'pearson', na.rm = TRUE)
4661.   cor2 <- cor1[[1]][1,2]
4662.   return(cor2)
4663. }
4664. # test correlation
4665. # i111
4666. DF_cor[[4]][1] <- fct_cor(i111j, gram_1_P1)
4667. DF_cor[[5]][1] <- fct_cor(i111j, gram_2_P1)
4668. DF_cor[[6]][1] <- fct_cor(i111j, gram_3_P1)
4669. DF_cor[[7]][1] <- fct_cor(i111j, dicot_1_P1)
4670. DF_cor[[8]][1] <- fct_cor(i111j, dicot_2_P1)
4671. DF_cor[[9]][1] <- fct_cor(i111j, dicot_3_P1)
4672. DF_cor[[10]][1] <- fct_cor(i111j, autre_P1)
4673. DF_cor[[11]][1] <- fct_cor(i111j, arbustive_P1)
4674. DF_cor[[12]][1] <- fct_cor(i111j, chromo_P1)
4675. DF_cor[[13]][1] <- fct_cor(i111j, abris_P1)
4676. DF_cor[[14]][1] <- fct_cor(i111n, gram_1_P1)
4677. DF_cor[[15]][1] <- fct_cor(i111n, gram_2_P1)
4678. DF_cor[[16]][1] <- fct_cor(i111n, gram_3_P1)
4679. DF_cor[[17]][1] <- fct_cor(i111n, dicot_1_P1)
4680. DF_cor[[18]][1] <- fct_cor(i111n, dicot_2_P1)
4681. DF_cor[[19]][1] <- fct_cor(i111n, dicot_3_P1)
4682. DF_cor[[20]][1] <- fct_cor(i111n, autre_P1)
4683. DF_cor[[21]][1] <- fct_cor(i111n, arbustive_P1)
4684. DF_cor[[22]][1] <- fct_cor(i111n, chromo_P1)
4685. DF_cor[[23]][1] <- fct_cor(i111n, abris_P1)
4686. DF_cor[[24]][1] <- fct_cor(i111g, gram_1_P1)
4687. DF_cor[[25]][1] <- fct_cor(i111g, gram_2_P1)
4688. DF_cor[[26]][1] <- fct_cor(i111g, gram_3_P1)
4689. DF_cor[[27]][1] <- fct_cor(i111g, dicot_1_P1)
4690. DF_cor[[28]][1] <- fct_cor(i111g, dicot_2_P1)
4691. DF_cor[[29]][1] <- fct_cor(i111g, dicot_3_P1)
4692. DF_cor[[30]][1] <- fct_cor(i111g, autre_P1)
4693. DF_cor[[31]][1] <- fct_cor(i111g, arbustive_P1)
4694. DF_cor[[32]][1] <- fct_cor(i111g, chromo_P1)
4695. DF_cor[[33]][1] <- fct_cor(i111g, abris_P1)
4696. DF_cor[[34]][1] <- fct_cor(i111ng, gram_1_P1)
4697. DF_cor[[35]][1] <- fct_cor(i111ng, gram_2_P1)

```

```

4698. DF_cor[[36]][1] <- fct_cor(i111ng, gram_3_P1)
4699. DF_cor[[37]][1] <- fct_cor(i111ng, dicot_1_P1)
4700. DF_cor[[38]][1] <- fct_cor(i111ng, dicot_2_P1)
4701. DF_cor[[39]][1] <- fct_cor(i111ng, dicot_3_P1)
4702. DF_cor[[40]][1] <- fct_cor(i111ng, autre_P1)
4703. DF_cor[[41]][1] <- fct_cor(i111ng, arbustive_P1)
4704. DF_cor[[42]][1] <- fct_cor(i111ng, chromo_P1)
4705. DF_cor[[43]][1] <- fct_cor(i111ng, abris_P1)
4706. # i112
4707. DF_cor[[4]][2] <- fct_cor(i112j, gram_1_P1)
4708. DF_cor[[5]][2] <- fct_cor(i112j, gram_2_P1)
4709. DF_cor[[6]][2] <- fct_cor(i112j, gram_3_P1)
4710. DF_cor[[7]][2] <- fct_cor(i112j, dicot_1_P1)
4711. DF_cor[[8]][2] <- fct_cor(i112j, dicot_2_P1)
4712. DF_cor[[9]][2] <- fct_cor(i112j, dicot_3_P1)
4713. DF_cor[[10]][2] <- fct_cor(i112j, autre_P1)
4714. DF_cor[[11]][2] <- fct_cor(i112j, arbustive_P1)
4715. DF_cor[[12]][2] <- fct_cor(i112j, chromo_P1)
4716. DF_cor[[13]][2] <- fct_cor(i112j, abris_P1)
4717. DF_cor[[14]][2] <- fct_cor(i112n, gram_1_P1)
4718. DF_cor[[15]][2] <- fct_cor(i112n, gram_2_P1)
4719. DF_cor[[16]][2] <- fct_cor(i112n, gram_3_P1)
4720. DF_cor[[17]][2] <- fct_cor(i112n, dicot_1_P1)
4721. DF_cor[[18]][2] <- fct_cor(i112n, dicot_2_P1)
4722. DF_cor[[19]][2] <- fct_cor(i112n, dicot_3_P1)
4723. DF_cor[[20]][2] <- fct_cor(i112n, autre_P1)
4724. DF_cor[[21]][2] <- fct_cor(i112n, arbustive_P1)
4725. DF_cor[[22]][2] <- fct_cor(i112n, chromo_P1)
4726. DF_cor[[23]][2] <- fct_cor(i112n, abris_P1)
4727. DF_cor[[24]][2] <- fct_cor(i112g, gram_1_P1)
4728. DF_cor[[25]][2] <- fct_cor(i112g, gram_2_P1)
4729. DF_cor[[26]][2] <- fct_cor(i112g, gram_3_P1)
4730. DF_cor[[27]][2] <- fct_cor(i112g, dicot_1_P1)
4731. DF_cor[[28]][2] <- fct_cor(i112g, dicot_2_P1)
4732. DF_cor[[29]][2] <- fct_cor(i112g, dicot_3_P1)
4733. DF_cor[[30]][2] <- fct_cor(i112g, autre_P1)
4734. DF_cor[[31]][2] <- fct_cor(i112g, arbustive_P1)
4735. DF_cor[[32]][2] <- fct_cor(i112g, chromo_P1)
4736. DF_cor[[33]][2] <- fct_cor(i112g, abris_P1)
4737. DF_cor[[34]][2] <- fct_cor(i112ng, gram_1_P1)
4738. DF_cor[[35]][2] <- fct_cor(i112ng, gram_2_P1)
4739. DF_cor[[36]][2] <- fct_cor(i112ng, gram_3_P1)
4740. DF_cor[[37]][2] <- fct_cor(i112ng, dicot_1_P1)
4741. DF_cor[[38]][2] <- fct_cor(i112ng, dicot_2_P1)
4742. DF_cor[[39]][2] <- fct_cor(i112ng, dicot_3_P1)
4743. DF_cor[[40]][2] <- fct_cor(i112ng, autre_P1)
4744. DF_cor[[41]][2] <- fct_cor(i112ng, arbustive_P1)
4745. DF_cor[[42]][2] <- fct_cor(i112ng, chromo_P1)
4746. DF_cor[[43]][2] <- fct_cor(i112ng, abris_P1)
4747. # i121
4748. DF_cor[[4]][3] <- fct_cor(i121j, gram_1_P1)
4749. DF_cor[[5]][3] <- fct_cor(i121j, gram_2_P1)

```

```

4750. DF_cor[[6]][3] <- fct_cor(i121j, gram_3_P1)
4751. DF_cor[[7]][3] <- fct_cor(i121j, dicot_1_P1)
4752. DF_cor[[8]][3] <- fct_cor(i121j, dicot_2_P1)
4753. DF_cor[[9]][3] <- fct_cor(i121j, dicot_3_P1)
4754. DF_cor[[10]][3] <- fct_cor(i121j, autre_P1)
4755. DF_cor[[11]][3] <- fct_cor(i121j, arbustive_P1)
4756. DF_cor[[12]][3] <- fct_cor(i121j, chromo_P1)
4757. DF_cor[[13]][3] <- fct_cor(i121j, abris_P1)
4758. DF_cor[[14]][3] <- fct_cor(i121n, gram_1_P1)
4759. DF_cor[[15]][3] <- fct_cor(i121n, gram_2_P1)
4760. DF_cor[[16]][3] <- fct_cor(i121n, gram_3_P1)
4761. DF_cor[[17]][3] <- fct_cor(i121n, dicot_1_P1)
4762. DF_cor[[18]][3] <- fct_cor(i121n, dicot_2_P1)
4763. DF_cor[[19]][3] <- fct_cor(i121n, dicot_3_P1)
4764. DF_cor[[20]][3] <- fct_cor(i121n, autre_P1)
4765. DF_cor[[21]][3] <- fct_cor(i121n, arbustive_P1)
4766. DF_cor[[22]][3] <- fct_cor(i121n, chromo_P1)
4767. DF_cor[[23]][3] <- fct_cor(i121n, abris_P1)
4768. DF_cor[[24]][3] <- fct_cor(i121g, gram_1_P1)
4769. DF_cor[[25]][3] <- fct_cor(i121g, gram_2_P1)
4770. DF_cor[[26]][3] <- fct_cor(i121g, gram_3_P1)
4771. DF_cor[[27]][3] <- fct_cor(i121g, dicot_1_P1)
4772. DF_cor[[28]][3] <- fct_cor(i121g, dicot_2_P1)
4773. DF_cor[[29]][3] <- fct_cor(i121g, dicot_3_P1)
4774. DF_cor[[30]][3] <- fct_cor(i121g, autre_P1)
4775. DF_cor[[31]][3] <- fct_cor(i121g, arbustive_P1)
4776. DF_cor[[32]][3] <- fct_cor(i121g, chromo_P1)
4777. DF_cor[[33]][3] <- fct_cor(i121g, abris_P1)
4778. DF_cor[[34]][3] <- fct_cor(i121ng, gram_1_P1)
4779. DF_cor[[35]][3] <- fct_cor(i121ng, gram_2_P1)
4780. DF_cor[[36]][3] <- fct_cor(i121ng, gram_3_P1)
4781. DF_cor[[37]][3] <- fct_cor(i121ng, dicot_1_P1)
4782. DF_cor[[38]][3] <- fct_cor(i121ng, dicot_2_P1)
4783. DF_cor[[39]][3] <- fct_cor(i121ng, dicot_3_P1)
4784. DF_cor[[40]][3] <- fct_cor(i121ng, autre_P1)
4785. DF_cor[[41]][3] <- fct_cor(i121ng, arbustive_P1)
4786. DF_cor[[42]][3] <- fct_cor(i121ng, chromo_P1)
4787. DF_cor[[43]][3] <- fct_cor(i121ng, abris_P1)
4788. # i122
4789. DF_cor[[4]][4] <- fct_cor(i122j, gram_1_P1)
4790. DF_cor[[5]][4] <- fct_cor(i122j, gram_2_P1)
4791. DF_cor[[6]][4] <- fct_cor(i122j, gram_3_P1)
4792. DF_cor[[7]][4] <- fct_cor(i122j, dicot_1_P1)
4793. DF_cor[[8]][4] <- fct_cor(i122j, dicot_2_P1)
4794. DF_cor[[9]][4] <- fct_cor(i122j, dicot_3_P1)
4795. DF_cor[[10]][4] <- fct_cor(i122j, autre_P1)
4796. DF_cor[[11]][4] <- fct_cor(i122j, arbustive_P1)
4797. DF_cor[[12]][4] <- fct_cor(i122j, chromo_P1)
4798. DF_cor[[13]][4] <- fct_cor(i122j, abris_P1)
4799. DF_cor[[14]][4] <- fct_cor(i122n, gram_1_P1)
4800. DF_cor[[15]][4] <- fct_cor(i122n, gram_2_P1)
4801. DF_cor[[16]][4] <- fct_cor(i122n, gram_3_P1)

```

```

4802. DF_cor[[17]][4] <- fct_cor(i122n, dicot_1_P1)
4803. DF_cor[[18]][4] <- fct_cor(i122n, dicot_2_P1)
4804. DF_cor[[19]][4] <- fct_cor(i122n, dicot_3_P1)
4805. DF_cor[[20]][4] <- fct_cor(i122n, autre_P1)
4806. DF_cor[[21]][4] <- fct_cor(i122n, arbustive_P1)
4807. DF_cor[[22]][4] <- fct_cor(i122n, chromo_P1)
4808. DF_cor[[23]][4] <- fct_cor(i122n, abris_P1)
4809. DF_cor[[24]][4] <- fct_cor(i122g, gram_1_P1)
4810. DF_cor[[25]][4] <- fct_cor(i122g, gram_2_P1)
4811. DF_cor[[26]][4] <- fct_cor(i122g, gram_3_P1)
4812. DF_cor[[27]][4] <- fct_cor(i122g, dicot_1_P1)
4813. DF_cor[[28]][4] <- fct_cor(i122g, dicot_2_P1)
4814. DF_cor[[29]][4] <- fct_cor(i122g, dicot_3_P1)
4815. DF_cor[[30]][4] <- fct_cor(i122g, autre_P1)
4816. DF_cor[[31]][4] <- fct_cor(i122g, arbustive_P1)
4817. DF_cor[[32]][4] <- fct_cor(i122g, chromo_P1)
4818. DF_cor[[33]][4] <- fct_cor(i122g, abris_P1)
4819. DF_cor[[34]][4] <- fct_cor(i122ng, gram_1_P1)
4820. DF_cor[[35]][4] <- fct_cor(i122ng, gram_2_P1)
4821. DF_cor[[36]][4] <- fct_cor(i122ng, gram_3_P1)
4822. DF_cor[[37]][4] <- fct_cor(i122ng, dicot_1_P1)
4823. DF_cor[[38]][4] <- fct_cor(i122ng, dicot_2_P1)
4824. DF_cor[[39]][4] <- fct_cor(i122ng, dicot_3_P1)
4825. DF_cor[[40]][4] <- fct_cor(i122ng, autre_P1)
4826. DF_cor[[41]][4] <- fct_cor(i122ng, arbustive_P1)
4827. DF_cor[[42]][4] <- fct_cor(i122ng, chromo_P1)
4828. DF_cor[[43]][4] <- fct_cor(i122ng, abris_P1)
4829. # i131
4830. DF_cor[[4]][5] <- fct_cor(i131j, gram_1_P1)
4831. DF_cor[[5]][5] <- fct_cor(i131j, gram_2_P1)
4832. DF_cor[[6]][5] <- fct_cor(i131j, gram_3_P1)
4833. DF_cor[[7]][5] <- fct_cor(i131j, dicot_1_P1)
4834. DF_cor[[8]][5] <- fct_cor(i131j, dicot_2_P1)
4835. DF_cor[[9]][5] <- fct_cor(i131j, dicot_3_P1)
4836. DF_cor[[10]][5] <- fct_cor(i131j, autre_P1)
4837. DF_cor[[11]][5] <- fct_cor(i131j, arbustive_P1)
4838. DF_cor[[12]][5] <- fct_cor(i131j, chromo_P1)
4839. DF_cor[[13]][5] <- fct_cor(i131j, abris_P1)
4840. DF_cor[[14]][5] <- fct_cor(i131n, gram_1_P1)
4841. DF_cor[[15]][5] <- fct_cor(i131n, gram_2_P1)
4842. DF_cor[[16]][5] <- fct_cor(i131n, gram_3_P1)
4843. DF_cor[[17]][5] <- fct_cor(i131n, dicot_1_P1)
4844. DF_cor[[18]][5] <- fct_cor(i131n, dicot_2_P1)
4845. DF_cor[[19]][5] <- fct_cor(i131n, dicot_3_P1)
4846. DF_cor[[20]][5] <- fct_cor(i131n, autre_P1)
4847. DF_cor[[21]][5] <- fct_cor(i131n, arbustive_P1)
4848. DF_cor[[22]][5] <- fct_cor(i131n, chromo_P1)
4849. DF_cor[[23]][5] <- fct_cor(i131n, abris_P1)
4850. DF_cor[[24]][5] <- fct_cor(i131g, gram_1_P1)
4851. DF_cor[[25]][5] <- fct_cor(i131g, gram_2_P1)
4852. DF_cor[[26]][5] <- fct_cor(i131g, gram_3_P1)
4853. DF_cor[[27]][5] <- fct_cor(i131g, dicot_1_P1)

```

```

4854. DF_cor[[28]][5] <- fct_cor(i131g, dicot_2_P1)
4855. DF_cor[[29]][5] <- fct_cor(i131g, dicot_3_P1)
4856. DF_cor[[30]][5] <- fct_cor(i131g, autre_P1)
4857. DF_cor[[31]][5] <- fct_cor(i131g, arbustive_P1)
4858. DF_cor[[32]][5] <- fct_cor(i131g, chromo_P1)
4859. DF_cor[[33]][5] <- fct_cor(i131g, abris_P1)
4860. DF_cor[[34]][5] <- fct_cor(i131ng, gram_1_P1)
4861. DF_cor[[35]][5] <- fct_cor(i131ng, gram_2_P1)
4862. DF_cor[[36]][5] <- fct_cor(i131ng, gram_3_P1)
4863. DF_cor[[37]][5] <- fct_cor(i131ng, dicot_1_P1)
4864. DF_cor[[38]][5] <- fct_cor(i131ng, dicot_2_P1)
4865. DF_cor[[39]][5] <- fct_cor(i131ng, dicot_3_P1)
4866. DF_cor[[40]][5] <- fct_cor(i131ng, autre_P1)
4867. DF_cor[[41]][5] <- fct_cor(i131ng, arbustive_P1)
4868. DF_cor[[42]][5] <- fct_cor(i131ng, chromo_P1)
4869. DF_cor[[43]][5] <- fct_cor(i131ng, abris_P1)
4870. # i132
4871. DF_cor[[4]][6] <- fct_cor(i132j, gram_1_P1)
4872. DF_cor[[5]][6] <- fct_cor(i132j, gram_2_P1)
4873. DF_cor[[6]][6] <- fct_cor(i132j, gram_3_P1)
4874. DF_cor[[7]][6] <- fct_cor(i132j, dicot_1_P1)
4875. DF_cor[[8]][6] <- fct_cor(i132j, dicot_2_P1)
4876. DF_cor[[9]][6] <- fct_cor(i132j, dicot_3_P1)
4877. DF_cor[[10]][6] <- fct_cor(i132j, autre_P1)
4878. DF_cor[[11]][6] <- fct_cor(i132j, arbustive_P1)
4879. DF_cor[[12]][6] <- fct_cor(i132j, chromo_P1)
4880. DF_cor[[13]][6] <- fct_cor(i132j, abris_P1)
4881. DF_cor[[14]][6] <- fct_cor(i132n, gram_1_P1)
4882. DF_cor[[15]][6] <- fct_cor(i132n, gram_2_P1)
4883. DF_cor[[16]][6] <- fct_cor(i132n, gram_3_P1)
4884. DF_cor[[17]][6] <- fct_cor(i132n, dicot_1_P1)
4885. DF_cor[[18]][6] <- fct_cor(i132n, dicot_2_P1)
4886. DF_cor[[19]][6] <- fct_cor(i132n, dicot_3_P1)
4887. DF_cor[[20]][6] <- fct_cor(i132n, autre_P1)
4888. DF_cor[[21]][6] <- fct_cor(i132n, arbustive_P1)
4889. DF_cor[[22]][6] <- fct_cor(i132n, chromo_P1)
4890. DF_cor[[23]][6] <- fct_cor(i132n, abris_P1)
4891. DF_cor[[24]][6] <- fct_cor(i132g, gram_1_P1)
4892. DF_cor[[25]][6] <- fct_cor(i132g, gram_2_P1)
4893. DF_cor[[26]][6] <- fct_cor(i132g, gram_3_P1)
4894. DF_cor[[27]][6] <- fct_cor(i132g, dicot_1_P1)
4895. DF_cor[[28]][6] <- fct_cor(i132g, dicot_2_P1)
4896. DF_cor[[29]][6] <- fct_cor(i132g, dicot_3_P1)
4897. DF_cor[[30]][6] <- fct_cor(i132g, autre_P1)
4898. DF_cor[[31]][6] <- fct_cor(i132g, arbustive_P1)
4899. DF_cor[[32]][6] <- fct_cor(i132g, chromo_P1)
4900. DF_cor[[33]][6] <- fct_cor(i132g, abris_P1)
4901. DF_cor[[34]][6] <- fct_cor(i132ng, gram_1_P1)
4902. DF_cor[[35]][6] <- fct_cor(i132ng, gram_2_P1)
4903. DF_cor[[36]][6] <- fct_cor(i132ng, gram_3_P1)
4904. DF_cor[[37]][6] <- fct_cor(i132ng, dicot_1_P1)
4905. DF_cor[[38]][6] <- fct_cor(i132ng, dicot_2_P1)

```

4906. DF_cor[[39]][6] <- fct_cor(i132ng, dicot_3_P1)
 4907. DF_cor[[40]][6] <- fct_cor(i132ng, autre_P1)
 4908. DF_cor[[41]][6] <- fct_cor(i132ng, arbustive_P1)
 4909. DF_cor[[42]][6] <- fct_cor(i132ng, chromo_P1)
 4910. DF_cor[[43]][6] <- fct_cor(i132ng, abris_P1)
4911. # i141
 4912. DF_cor[[4]][7] <- fct_cor(i141j, gram_1_P1)
 4913. DF_cor[[5]][7] <- fct_cor(i141j, gram_2_P1)
 4914. DF_cor[[6]][7] <- fct_cor(i141j, gram_3_P1)
 4915. DF_cor[[7]][7] <- fct_cor(i141j, dicot_1_P1)
 4916. DF_cor[[8]][7] <- fct_cor(i141j, dicot_2_P1)
 4917. DF_cor[[9]][7] <- fct_cor(i141j, dicot_3_P1)
 4918. DF_cor[[10]][7] <- fct_cor(i141j, autre_P1)
 4919. DF_cor[[11]][7] <- fct_cor(i141j, arbustive_P1)
 4920. DF_cor[[12]][7] <- fct_cor(i141j, chromo_P1)
 4921. DF_cor[[13]][7] <- fct_cor(i141j, abris_P1)
 4922. DF_cor[[14]][7] <- fct_cor(i141n, gram_1_P1)
 4923. DF_cor[[15]][7] <- fct_cor(i141n, gram_2_P1)
 4924. DF_cor[[16]][7] <- fct_cor(i141n, gram_3_P1)
 4925. DF_cor[[17]][7] <- fct_cor(i141n, dicot_1_P1)
 4926. DF_cor[[18]][7] <- fct_cor(i141n, dicot_2_P1)
 4927. DF_cor[[19]][7] <- fct_cor(i141n, dicot_3_P1)
 4928. DF_cor[[20]][7] <- fct_cor(i141n, autre_P1)
 4929. DF_cor[[21]][7] <- fct_cor(i141n, arbustive_P1)
 4930. DF_cor[[22]][7] <- fct_cor(i141n, chromo_P1)
 4931. DF_cor[[23]][7] <- fct_cor(i141n, abris_P1)
 4932. DF_cor[[24]][7] <- fct_cor(i141g, gram_1_P1)
 4933. DF_cor[[25]][7] <- fct_cor(i141g, gram_2_P1)
 4934. DF_cor[[26]][7] <- fct_cor(i141g, gram_3_P1)
 4935. DF_cor[[27]][7] <- fct_cor(i141g, dicot_1_P1)
 4936. DF_cor[[28]][7] <- fct_cor(i141g, dicot_2_P1)
 4937. DF_cor[[29]][7] <- fct_cor(i141g, dicot_3_P1)
 4938. DF_cor[[30]][7] <- fct_cor(i141g, autre_P1)
 4939. DF_cor[[31]][7] <- fct_cor(i141g, arbustive_P1)
 4940. DF_cor[[32]][7] <- fct_cor(i141g, chromo_P1)
 4941. DF_cor[[33]][7] <- fct_cor(i141g, abris_P1)
 4942. DF_cor[[34]][7] <- fct_cor(i141ng, gram_1_P1)
 4943. DF_cor[[35]][7] <- fct_cor(i141ng, gram_2_P1)
 4944. DF_cor[[36]][7] <- fct_cor(i141ng, gram_3_P1)
 4945. DF_cor[[37]][7] <- fct_cor(i141ng, dicot_1_P1)
 4946. DF_cor[[38]][7] <- fct_cor(i141ng, dicot_2_P1)
 4947. DF_cor[[39]][7] <- fct_cor(i141ng, dicot_3_P1)
 4948. DF_cor[[40]][7] <- fct_cor(i141ng, autre_P1)
 4949. DF_cor[[41]][7] <- fct_cor(i141ng, arbustive_P1)
 4950. DF_cor[[42]][7] <- fct_cor(i141ng, chromo_P1)
 4951. DF_cor[[43]][7] <- fct_cor(i141ng, abris_P1)
4952. # i142
 4953. DF_cor[[4]][8] <- fct_cor(i142j, gram_1_P1)
 4954. DF_cor[[5]][8] <- fct_cor(i142j, gram_2_P1)
 4955. DF_cor[[6]][8] <- fct_cor(i142j, gram_3_P1)
 4956. DF_cor[[7]][8] <- fct_cor(i142j, dicot_1_P1)
 4957. DF_cor[[8]][8] <- fct_cor(i142j, dicot_2_P1)

4958. DF_cor[[9]][8] <- fct_cor(i142j, dicot_3_P1)
 4959. DF_cor[[10]][8] <- fct_cor(i142j, autre_P1)
 4960. DF_cor[[11]][8] <- fct_cor(i142j, arbustive_P1)
 4961. DF_cor[[12]][8] <- fct_cor(i142j, chromo_P1)
 4962. DF_cor[[13]][8] <- fct_cor(i142j, abris_P1)
 4963. DF_cor[[14]][8] <- fct_cor(i142n, gram_1_P1)
 4964. DF_cor[[15]][8] <- fct_cor(i142n, gram_2_P1)
 4965. DF_cor[[16]][8] <- fct_cor(i142n, gram_3_P1)
 4966. DF_cor[[17]][8] <- fct_cor(i142n, dicot_1_P1)
 4967. DF_cor[[18]][8] <- fct_cor(i142n, dicot_2_P1)
 4968. DF_cor[[19]][8] <- fct_cor(i142n, dicot_3_P1)
 4969. DF_cor[[20]][8] <- fct_cor(i142n, autre_P1)
 4970. DF_cor[[21]][8] <- fct_cor(i142n, arbustive_P1)
 4971. DF_cor[[22]][8] <- fct_cor(i142n, chromo_P1)
 4972. DF_cor[[23]][8] <- fct_cor(i142n, abris_P1)
 4973. DF_cor[[24]][8] <- fct_cor(i142g, gram_1_P1)
 4974. DF_cor[[25]][8] <- fct_cor(i142g, gram_2_P1)
 4975. DF_cor[[26]][8] <- fct_cor(i142g, gram_3_P1)
 4976. DF_cor[[27]][8] <- fct_cor(i142g, dicot_1_P1)
 4977. DF_cor[[28]][8] <- fct_cor(i142g, dicot_2_P1)
 4978. DF_cor[[29]][8] <- fct_cor(i142g, dicot_3_P1)
 4979. DF_cor[[30]][8] <- fct_cor(i142g, autre_P1)
 4980. DF_cor[[31]][8] <- fct_cor(i142g, arbustive_P1)
 4981. DF_cor[[32]][8] <- fct_cor(i142g, chromo_P1)
 4982. DF_cor[[33]][8] <- fct_cor(i142g, abris_P1)
 4983. DF_cor[[34]][8] <- fct_cor(i142ng, gram_1_P1)
 4984. DF_cor[[35]][8] <- fct_cor(i142ng, gram_2_P1)
 4985. DF_cor[[36]][8] <- fct_cor(i142ng, gram_3_P1)
 4986. DF_cor[[37]][8] <- fct_cor(i142ng, dicot_1_P1)
 4987. DF_cor[[38]][8] <- fct_cor(i142ng, dicot_2_P1)
 4988. DF_cor[[39]][8] <- fct_cor(i142ng, dicot_3_P1)
 4989. DF_cor[[40]][8] <- fct_cor(i142ng, autre_P1)
 4990. DF_cor[[41]][8] <- fct_cor(i142ng, arbustive_P1)
 4991. DF_cor[[42]][8] <- fct_cor(i142ng, chromo_P1)
 4992. DF_cor[[43]][8] <- fct_cor(i142ng, abris_P1)
4993. # i211
 4994. DF_cor[[4]][9] <- fct_cor(i211j, gram_1_P2)
 4995. DF_cor[[5]][9] <- fct_cor(i211j, gram_2_P2)
 4996. DF_cor[[6]][9] <- fct_cor(i211j, gram_3_P2)
 4997. DF_cor[[7]][9] <- fct_cor(i211j, dicot_1_P2)
 4998. DF_cor[[8]][9] <- fct_cor(i211j, dicot_2_P2)
 4999. DF_cor[[9]][9] <- fct_cor(i211j, dicot_3_P2)
 5000. DF_cor[[10]][9] <- fct_cor(i211j, autre_P2)
 5001. DF_cor[[11]][9] <- fct_cor(i211j, arbustive_P2)
 5002. DF_cor[[12]][9] <- fct_cor(i211j, chromo_P2)
 5003. DF_cor[[13]][9] <- fct_cor(i211j, abris_P2)
 5004. DF_cor[[14]][9] <- fct_cor(i211n, gram_1_P2)
 5005. DF_cor[[15]][9] <- fct_cor(i211n, gram_2_P2)
 5006. DF_cor[[16]][9] <- fct_cor(i211n, gram_3_P2)
 5007. DF_cor[[17]][9] <- fct_cor(i211n, dicot_1_P2)
 5008. DF_cor[[18]][9] <- fct_cor(i211n, dicot_2_P2)
 5009. DF_cor[[19]][9] <- fct_cor(i211n, dicot_3_P2)

```

5010. DF_cor[[20]][9] <- fct_cor(i211n, autre_P2)
5011. DF_cor[[21]][9] <- fct_cor(i211n, arbustive_P2)
5012. DF_cor[[22]][9] <- fct_cor(i211n, chromo_P2)
5013. DF_cor[[23]][9] <- fct_cor(i211n, abris_P2)
5014. DF_cor[[24]][9] <- fct_cor(i211g, gram_1_P2)
5015. DF_cor[[25]][9] <- fct_cor(i211g, gram_2_P2)
5016. DF_cor[[26]][9] <- fct_cor(i211g, gram_3_P2)
5017. DF_cor[[27]][9] <- fct_cor(i211g, dicot_1_P2)
5018. DF_cor[[28]][9] <- fct_cor(i211g, dicot_2_P2)
5019. DF_cor[[29]][9] <- fct_cor(i211g, dicot_3_P2)
5020. DF_cor[[30]][9] <- fct_cor(i211g, autre_P2)
5021. DF_cor[[31]][9] <- fct_cor(i211g, arbustive_P2)
5022. DF_cor[[32]][9] <- fct_cor(i211g, chromo_P2)
5023. DF_cor[[33]][9] <- fct_cor(i211g, abris_P2)
5024. DF_cor[[34]][9] <- fct_cor(i211ng, gram_1_P2)
5025. DF_cor[[35]][9] <- fct_cor(i211ng, gram_2_P2)
5026. DF_cor[[36]][9] <- fct_cor(i211ng, gram_3_P2)
5027. DF_cor[[37]][9] <- fct_cor(i211ng, dicot_1_P2)
5028. DF_cor[[38]][9] <- fct_cor(i211ng, dicot_2_P2)
5029. DF_cor[[39]][9] <- fct_cor(i211ng, dicot_3_P2)
5030. DF_cor[[40]][9] <- fct_cor(i211ng, autre_P2)
5031. DF_cor[[41]][9] <- fct_cor(i211ng, arbustive_P2)
5032. DF_cor[[42]][9] <- fct_cor(i211ng, chromo_P2)
5033. DF_cor[[43]][9] <- fct_cor(i211ng, abris_P2)
5034. # i212
5035. DF_cor[[4]][10] <- fct_cor(i212j, gram_1_P2)
5036. DF_cor[[5]][10] <- fct_cor(i212j, gram_2_P2)
5037. DF_cor[[6]][10] <- fct_cor(i212j, gram_3_P2)
5038. DF_cor[[7]][10] <- fct_cor(i212j, dicot_1_P2)
5039. DF_cor[[8]][10] <- fct_cor(i212j, dicot_2_P2)
5040. DF_cor[[9]][10] <- fct_cor(i212j, dicot_3_P2)
5041. DF_cor[[10]][10] <- fct_cor(i212j, autre_P2)
5042. DF_cor[[11]][10] <- fct_cor(i212j, arbustive_P2)
5043. DF_cor[[12]][10] <- fct_cor(i212j, chromo_P2)
5044. DF_cor[[13]][10] <- fct_cor(i212j, abris_P2)
5045. DF_cor[[14]][10] <- fct_cor(i212n, gram_1_P2)
5046. DF_cor[[15]][10] <- fct_cor(i212n, gram_2_P2)
5047. DF_cor[[16]][10] <- fct_cor(i212n, gram_3_P2)
5048. DF_cor[[17]][10] <- fct_cor(i212n, dicot_1_P2)
5049. DF_cor[[18]][10] <- fct_cor(i212n, dicot_2_P2)
5050. DF_cor[[19]][10] <- fct_cor(i212n, dicot_3_P2)
5051. DF_cor[[20]][10] <- fct_cor(i212n, autre_P2)
5052. DF_cor[[21]][10] <- fct_cor(i212n, arbustive_P2)
5053. DF_cor[[22]][10] <- fct_cor(i212n, chromo_P2)
5054. DF_cor[[23]][10] <- fct_cor(i212n, abris_P2)
5055. DF_cor[[24]][10] <- fct_cor(i212g, gram_1_P2)
5056. DF_cor[[25]][10] <- fct_cor(i212g, gram_2_P2)
5057. DF_cor[[26]][10] <- fct_cor(i212g, gram_3_P2)
5058. DF_cor[[27]][10] <- fct_cor(i212g, dicot_1_P2)
5059. DF_cor[[28]][10] <- fct_cor(i212g, dicot_2_P2)
5060. DF_cor[[29]][10] <- fct_cor(i212g, dicot_3_P2)
5061. DF_cor[[30]][10] <- fct_cor(i212g, autre_P2)
5062. DF_cor[[31]][10] <- fct_cor(i212g, arbustive_P2)
5063. DF_cor[[32]][10] <- fct_cor(i212g, chromo_P2)
5064. DF_cor[[33]][10] <- fct_cor(i212g, abris_P2)
5065. DF_cor[[34]][10] <- fct_cor(i212ng, gram_1_P2)
5066. DF_cor[[35]][10] <- fct_cor(i212ng, gram_2_P2)
5067. DF_cor[[36]][10] <- fct_cor(i212ng, gram_3_P2)
5068. DF_cor[[37]][10] <- fct_cor(i212ng, dicot_1_P2)
5069. DF_cor[[38]][10] <- fct_cor(i212ng, dicot_2_P2)
5070. DF_cor[[39]][10] <- fct_cor(i212ng, dicot_3_P2)
5071. DF_cor[[40]][10] <- fct_cor(i212ng, autre_P2)
5072. DF_cor[[41]][10] <- fct_cor(i212ng, arbustive_P2)
5073. DF_cor[[42]][10] <- fct_cor(i212ng, chromo_P2)
5074. DF_cor[[43]][10] <- fct_cor(i212ng, abris_P2)
5075. # i221
5076. DF_cor[[4]][11] <- fct_cor(i221j, gram_1_P2)
5077. DF_cor[[5]][11] <- fct_cor(i221j, gram_2_P2)
5078. DF_cor[[6]][11] <- fct_cor(i221j, gram_3_P2)
5079. DF_cor[[7]][11] <- fct_cor(i221j, dicot_1_P2)
5080. DF_cor[[8]][11] <- fct_cor(i221j, dicot_2_P2)
5081. DF_cor[[9]][11] <- fct_cor(i221j, dicot_3_P2)
5082. DF_cor[[10]][11] <- fct_cor(i221j, autre_P2)
5083. DF_cor[[11]][11] <- fct_cor(i221j, arbustive_P2)
5084. DF_cor[[12]][11] <- fct_cor(i221j, chromo_P2)
5085. DF_cor[[13]][11] <- fct_cor(i221j, abris_P2)
5086. DF_cor[[14]][11] <- fct_cor(i221n, gram_1_P2)
5087. DF_cor[[15]][11] <- fct_cor(i221n, gram_2_P2)
5088. DF_cor[[16]][11] <- fct_cor(i221n, gram_3_P2)
5089. DF_cor[[17]][11] <- fct_cor(i221n, dicot_1_P2)
5090. DF_cor[[18]][11] <- fct_cor(i221n, dicot_2_P2)
5091. DF_cor[[19]][11] <- fct_cor(i221n, dicot_3_P2)
5092. DF_cor[[20]][11] <- fct_cor(i221n, autre_P2)
5093. DF_cor[[21]][11] <- fct_cor(i221n, arbustive_P2)
5094. DF_cor[[22]][11] <- fct_cor(i221n, chromo_P2)
5095. DF_cor[[23]][11] <- fct_cor(i221n, abris_P2)
5096. DF_cor[[24]][11] <- fct_cor(i221g, gram_1_P2)
5097. DF_cor[[25]][11] <- fct_cor(i221g, gram_2_P2)
5098. DF_cor[[26]][11] <- fct_cor(i221g, gram_3_P2)
5099. DF_cor[[27]][11] <- fct_cor(i221g, dicot_1_P2)
5100. DF_cor[[28]][11] <- fct_cor(i221g, dicot_2_P2)
5101. DF_cor[[29]][11] <- fct_cor(i221g, dicot_3_P2)
5102. DF_cor[[30]][11] <- fct_cor(i221g, autre_P2)
5103. DF_cor[[31]][11] <- fct_cor(i221g, arbustive_P2)
5104. DF_cor[[32]][11] <- fct_cor(i221g, chromo_P2)
5105. DF_cor[[33]][11] <- fct_cor(i221g, abris_P2)
5106. DF_cor[[34]][11] <- fct_cor(i221ng, gram_1_P2)
5107. DF_cor[[35]][11] <- fct_cor(i221ng, gram_2_P2)
5108. DF_cor[[36]][11] <- fct_cor(i221ng, gram_3_P2)
5109. DF_cor[[37]][11] <- fct_cor(i221ng, dicot_1_P2)
5110. DF_cor[[38]][11] <- fct_cor(i221ng, dicot_2_P2)
5111. DF_cor[[39]][11] <- fct_cor(i221ng, dicot_3_P2)
5112. DF_cor[[40]][11] <- fct_cor(i221ng, autre_P2)
5113. DF_cor[[41]][11] <- fct_cor(i221ng, arbustive_P2)

```

5114. DF_cor[[42]][11] <- fct_cor(i221ng, chromo_P2)
 5115. DF_cor[[43]][11] <- fct_cor(i221ng, abris_P2)
5116. # i222
 5117. DF_cor[[4]][12] <- fct_cor(i222j, gram_1_P2)
 5118. DF_cor[[5]][12] <- fct_cor(i222j, gram_2_P2)
 5119. DF_cor[[6]][12] <- fct_cor(i222j, gram_3_P2)
 5120. DF_cor[[7]][12] <- fct_cor(i222j, dicot_1_P2)
 5121. DF_cor[[8]][12] <- fct_cor(i222j, dicot_2_P2)
 5122. DF_cor[[9]][12] <- fct_cor(i222j, dicot_3_P2)
 5123. DF_cor[[10]][12] <- fct_cor(i222j, autre_P2)
 5124. DF_cor[[11]][12] <- fct_cor(i222j, arbustive_P2)
 5125. DF_cor[[12]][12] <- fct_cor(i222j, chromo_P2)
 5126. DF_cor[[13]][12] <- fct_cor(i222j, abris_P2)
 5127. DF_cor[[14]][12] <- fct_cor(i222n, gram_1_P2)
 5128. DF_cor[[15]][12] <- fct_cor(i222n, gram_2_P2)
 5129. DF_cor[[16]][12] <- fct_cor(i222n, gram_3_P2)
 5130. DF_cor[[17]][12] <- fct_cor(i222n, dicot_1_P2)
 5131. DF_cor[[18]][12] <- fct_cor(i222n, dicot_2_P2)
 5132. DF_cor[[19]][12] <- fct_cor(i222n, dicot_3_P2)
 5133. DF_cor[[20]][12] <- fct_cor(i222n, autre_P2)
 5134. DF_cor[[21]][12] <- fct_cor(i222n, arbustive_P2)
 5135. DF_cor[[22]][12] <- fct_cor(i222n, chromo_P2)
 5136. DF_cor[[23]][12] <- fct_cor(i222n, abris_P2)
 5137. DF_cor[[24]][12] <- fct_cor(i222g, gram_1_P2)
 5138. DF_cor[[25]][12] <- fct_cor(i222g, gram_2_P2)
 5139. DF_cor[[26]][12] <- fct_cor(i222g, gram_3_P2)
 5140. DF_cor[[27]][12] <- fct_cor(i222g, dicot_1_P2)
 5141. DF_cor[[28]][12] <- fct_cor(i222g, dicot_2_P2)
 5142. DF_cor[[29]][12] <- fct_cor(i222g, dicot_3_P2)
 5143. DF_cor[[30]][12] <- fct_cor(i222g, autre_P2)
 5144. DF_cor[[31]][12] <- fct_cor(i222g, arbustive_P2)
 5145. DF_cor[[32]][12] <- fct_cor(i222g, chromo_P2)
 5146. DF_cor[[33]][12] <- fct_cor(i222g, abris_P2)
 5147. DF_cor[[34]][12] <- fct_cor(i222ng, gram_1_P2)
 5148. DF_cor[[35]][12] <- fct_cor(i222ng, gram_2_P2)
 5149. DF_cor[[36]][12] <- fct_cor(i222ng, gram_3_P2)
 5150. DF_cor[[37]][12] <- fct_cor(i222ng, dicot_1_P2)
 5151. DF_cor[[38]][12] <- fct_cor(i222ng, dicot_2_P2)
 5152. DF_cor[[39]][12] <- fct_cor(i222ng, dicot_3_P2)
 5153. DF_cor[[40]][12] <- fct_cor(i222ng, autre_P2)
 5154. DF_cor[[41]][12] <- fct_cor(i222ng, arbustive_P2)
 5155. DF_cor[[42]][12] <- fct_cor(i222ng, chromo_P2)
 5156. DF_cor[[43]][12] <- fct_cor(i222ng, abris_P2)
5157. # i231
 5158. DF_cor[[4]][13] <- fct_cor(i231j, gram_1_P2)
 5159. DF_cor[[5]][13] <- fct_cor(i231j, gram_2_P2)
 5160. DF_cor[[6]][13] <- fct_cor(i231j, gram_3_P2)
 5161. DF_cor[[7]][13] <- fct_cor(i231j, dicot_1_P2)
 5162. DF_cor[[8]][13] <- fct_cor(i231j, dicot_2_P2)
 5163. DF_cor[[9]][13] <- fct_cor(i231j, dicot_3_P2)
 5164. DF_cor[[10]][13] <- fct_cor(i231j, autre_P2)
 5165. DF_cor[[11]][13] <- fct_cor(i231j, arbustive_P2)

5166. DF_cor[[12]][13] <- fct_cor(i231j, chromo_P2)
 5167. DF_cor[[13]][13] <- fct_cor(i231j, abris_P2)
 5168. DF_cor[[14]][13] <- fct_cor(i231n, gram_1_P2)
 5169. DF_cor[[15]][13] <- fct_cor(i231n, gram_2_P2)
 5170. DF_cor[[16]][13] <- fct_cor(i231n, gram_3_P2)
 5171. DF_cor[[17]][13] <- fct_cor(i231n, dicot_1_P2)
 5172. DF_cor[[18]][13] <- fct_cor(i231n, dicot_2_P2)
 5173. DF_cor[[19]][13] <- fct_cor(i231n, dicot_3_P2)
 5174. DF_cor[[20]][13] <- fct_cor(i231n, autre_P2)
 5175. DF_cor[[21]][13] <- fct_cor(i231n, arbustive_P2)
 5176. DF_cor[[22]][13] <- fct_cor(i231n, chromo_P2)
 5177. DF_cor[[23]][13] <- fct_cor(i231n, abris_P2)
 5178. DF_cor[[24]][13] <- fct_cor(i231g, gram_1_P2)
 5179. DF_cor[[25]][13] <- fct_cor(i231g, gram_2_P2)
 5180. DF_cor[[26]][13] <- fct_cor(i231g, gram_3_P2)
 5181. DF_cor[[27]][13] <- fct_cor(i231g, dicot_1_P2)
 5182. DF_cor[[28]][13] <- fct_cor(i231g, dicot_2_P2)
 5183. DF_cor[[29]][13] <- fct_cor(i231g, dicot_3_P2)
 5184. DF_cor[[30]][13] <- fct_cor(i231g, autre_P2)
 5185. DF_cor[[31]][13] <- fct_cor(i231g, arbustive_P2)
 5186. DF_cor[[32]][13] <- fct_cor(i231g, chromo_P2)
 5187. DF_cor[[33]][13] <- fct_cor(i231g, abris_P2)
 5188. DF_cor[[34]][13] <- fct_cor(i231ng, gram_1_P2)
 5189. DF_cor[[35]][13] <- fct_cor(i231ng, gram_2_P2)
 5190. DF_cor[[36]][13] <- fct_cor(i231ng, gram_3_P2)
 5191. DF_cor[[37]][13] <- fct_cor(i231ng, dicot_1_P2)
 5192. DF_cor[[38]][13] <- fct_cor(i231ng, dicot_2_P2)
 5193. DF_cor[[39]][13] <- fct_cor(i231ng, dicot_3_P2)
 5194. DF_cor[[40]][13] <- fct_cor(i231ng, autre_P2)
 5195. DF_cor[[41]][13] <- fct_cor(i231ng, arbustive_P2)
 5196. DF_cor[[42]][13] <- fct_cor(i231ng, chromo_P2)
 5197. DF_cor[[43]][13] <- fct_cor(i231ng, abris_P2)
5198. # i232
 5199. DF_cor[[4]][14] <- fct_cor(i232j, gram_1_P2)
 5200. DF_cor[[5]][14] <- fct_cor(i232j, gram_2_P2)
 5201. DF_cor[[6]][14] <- fct_cor(i232j, gram_3_P2)
 5202. DF_cor[[7]][14] <- fct_cor(i232j, dicot_1_P2)
 5203. DF_cor[[8]][14] <- fct_cor(i232j, dicot_2_P2)
 5204. DF_cor[[9]][14] <- fct_cor(i232j, dicot_3_P2)
 5205. DF_cor[[10]][14] <- fct_cor(i232j, autre_P2)
 5206. DF_cor[[11]][14] <- fct_cor(i232j, arbustive_P2)
 5207. DF_cor[[12]][14] <- fct_cor(i232j, chromo_P2)
 5208. DF_cor[[13]][14] <- fct_cor(i232j, abris_P2)
 5209. DF_cor[[14]][14] <- fct_cor(i232n, gram_1_P2)
 5210. DF_cor[[15]][14] <- fct_cor(i232n, gram_2_P2)
 5211. DF_cor[[16]][14] <- fct_cor(i232n, gram_3_P2)
 5212. DF_cor[[17]][14] <- fct_cor(i232n, dicot_1_P2)
 5213. DF_cor[[18]][14] <- fct_cor(i232n, dicot_2_P2)
 5214. DF_cor[[19]][14] <- fct_cor(i232n, dicot_3_P2)
 5215. DF_cor[[20]][14] <- fct_cor(i232n, autre_P2)
 5216. DF_cor[[21]][14] <- fct_cor(i232n, arbustive_P2)
 5217. DF_cor[[22]][14] <- fct_cor(i232n, chromo_P2)

```

5218. DF_cor[[23]][14] <- fct_cor(i232n, abris_P2)
5219. DF_cor[[24]][14] <- fct_cor(i232g, gram_1_P2)
5220. DF_cor[[25]][14] <- fct_cor(i232g, gram_2_P2)
5221. DF_cor[[26]][14] <- fct_cor(i232g, gram_3_P2)
5222. DF_cor[[27]][14] <- fct_cor(i232g, dicot_1_P2)
5223. DF_cor[[28]][14] <- fct_cor(i232g, dicot_2_P2)
5224. DF_cor[[29]][14] <- fct_cor(i232g, dicot_3_P2)
5225. DF_cor[[30]][14] <- fct_cor(i232g, autre_P2)
5226. DF_cor[[31]][14] <- fct_cor(i232g, arbustive_P2)
5227. DF_cor[[32]][14] <- fct_cor(i232g, chromo_P2)
5228. DF_cor[[33]][14] <- fct_cor(i232g, abris_P2)
5229. DF_cor[[34]][14] <- fct_cor(i232ng, gram_1_P2)
5230. DF_cor[[35]][14] <- fct_cor(i232ng, gram_2_P2)
5231. DF_cor[[36]][14] <- fct_cor(i232ng, gram_3_P2)
5232. DF_cor[[37]][14] <- fct_cor(i232ng, dicot_1_P2)
5233. DF_cor[[38]][14] <- fct_cor(i232ng, dicot_2_P2)
5234. DF_cor[[39]][14] <- fct_cor(i232ng, dicot_3_P2)
5235. DF_cor[[40]][14] <- fct_cor(i232ng, autre_P2)
5236. DF_cor[[41]][14] <- fct_cor(i232ng, arbustive_P2)
5237. DF_cor[[42]][14] <- fct_cor(i232ng, chromo_P2)
5238. DF_cor[[43]][14] <- fct_cor(i232ng, abris_P2)
5239. # i241
5240. DF_cor[[4]][15] <- fct_cor(i241j, gram_1_P2)
5241. DF_cor[[5]][15] <- fct_cor(i241j, gram_2_P2)
5242. DF_cor[[6]][15] <- fct_cor(i241j, gram_3_P2)
5243. DF_cor[[7]][15] <- fct_cor(i241j, dicot_1_P2)
5244. DF_cor[[8]][15] <- fct_cor(i241j, dicot_2_P2)
5245. DF_cor[[9]][15] <- fct_cor(i241j, dicot_3_P2)
5246. DF_cor[[10]][15] <- fct_cor(i241j, autre_P2)
5247. DF_cor[[11]][15] <- fct_cor(i241j, arbustive_P2)
5248. DF_cor[[12]][15] <- fct_cor(i241j, chromo_P2)
5249. DF_cor[[13]][15] <- fct_cor(i241j, abris_P2)
5250. DF_cor[[14]][15] <- fct_cor(i241n, gram_1_P2)
5251. DF_cor[[15]][15] <- fct_cor(i241n, gram_2_P2)
5252. DF_cor[[16]][15] <- fct_cor(i241n, gram_3_P2)
5253. DF_cor[[17]][15] <- fct_cor(i241n, dicot_1_P2)
5254. DF_cor[[18]][15] <- fct_cor(i241n, dicot_2_P2)
5255. DF_cor[[19]][15] <- fct_cor(i241n, dicot_3_P2)
5256. DF_cor[[20]][15] <- fct_cor(i241n, autre_P2)
5257. DF_cor[[21]][15] <- fct_cor(i241n, arbustive_P2)
5258. DF_cor[[22]][15] <- fct_cor(i241n, chromo_P2)
5259. DF_cor[[23]][15] <- fct_cor(i241n, abris_P2)
5260. DF_cor[[24]][15] <- fct_cor(i241g, gram_1_P2)
5261. DF_cor[[25]][15] <- fct_cor(i241g, gram_2_P2)
5262. DF_cor[[26]][15] <- fct_cor(i241g, gram_3_P2)
5263. DF_cor[[27]][15] <- fct_cor(i241g, dicot_1_P2)
5264. DF_cor[[28]][15] <- fct_cor(i241g, dicot_2_P2)
5265. DF_cor[[29]][15] <- fct_cor(i241g, dicot_3_P2)
5266. DF_cor[[30]][15] <- fct_cor(i241g, autre_P2)
5267. DF_cor[[31]][15] <- fct_cor(i241g, arbustive_P2)
5268. DF_cor[[32]][15] <- fct_cor(i241g, chromo_P2)
5269. DF_cor[[33]][15] <- fct_cor(i241g, abris_P2)

```

```

5270. DF_cor[[34]][15] <- fct_cor(i241ng, gram_1_P2)
5271. DF_cor[[35]][15] <- fct_cor(i241ng, gram_2_P2)
5272. DF_cor[[36]][15] <- fct_cor(i241ng, gram_3_P2)
5273. DF_cor[[37]][15] <- fct_cor(i241ng, dicot_1_P2)
5274. DF_cor[[38]][15] <- fct_cor(i241ng, dicot_2_P2)
5275. DF_cor[[39]][15] <- fct_cor(i241ng, dicot_3_P2)
5276. DF_cor[[40]][15] <- fct_cor(i241ng, autre_P2)
5277. DF_cor[[41]][15] <- fct_cor(i241ng, arbustive_P2)
5278. DF_cor[[42]][15] <- fct_cor(i241ng, chromo_P2)
5279. DF_cor[[43]][15] <- fct_cor(i241ng, abris_P2)
5280. # i242
5281. DF_cor[[4]][16] <- fct_cor(i242j, gram_1_P2)
5282. DF_cor[[5]][16] <- fct_cor(i242j, gram_2_P2)
5283. DF_cor[[6]][16] <- fct_cor(i242j, gram_3_P2)
5284. DF_cor[[7]][16] <- fct_cor(i242j, dicot_1_P2)
5285. DF_cor[[8]][16] <- fct_cor(i242j, dicot_2_P2)
5286. DF_cor[[9]][16] <- fct_cor(i242j, dicot_3_P2)
5287. DF_cor[[10]][16] <- fct_cor(i242j, autre_P2)
5288. DF_cor[[11]][16] <- fct_cor(i242j, arbustive_P2)
5289. DF_cor[[12]][16] <- fct_cor(i242j, chromo_P2)
5290. DF_cor[[13]][16] <- fct_cor(i242j, abris_P2)
5291. DF_cor[[14]][16] <- fct_cor(i242n, gram_1_P2)
5292. DF_cor[[15]][16] <- fct_cor(i242n, gram_2_P2)
5293. DF_cor[[16]][16] <- fct_cor(i242n, gram_3_P2)
5294. DF_cor[[17]][16] <- fct_cor(i242n, dicot_1_P2)
5295. DF_cor[[18]][16] <- fct_cor(i242n, dicot_2_P2)
5296. DF_cor[[19]][16] <- fct_cor(i242n, dicot_3_P2)
5297. DF_cor[[20]][16] <- fct_cor(i242n, autre_P2)
5298. DF_cor[[21]][16] <- fct_cor(i242n, arbustive_P2)
5299. DF_cor[[22]][16] <- fct_cor(i242n, chromo_P2)
5300. DF_cor[[23]][16] <- fct_cor(i242n, abris_P2)
5301. DF_cor[[24]][16] <- fct_cor(i242g, gram_1_P2)
5302. DF_cor[[25]][16] <- fct_cor(i242g, gram_2_P2)
5303. DF_cor[[26]][16] <- fct_cor(i242g, gram_3_P2)
5304. DF_cor[[27]][16] <- fct_cor(i242g, dicot_1_P2)
5305. DF_cor[[28]][16] <- fct_cor(i242g, dicot_2_P2)
5306. DF_cor[[29]][16] <- fct_cor(i242g, dicot_3_P2)
5307. DF_cor[[30]][16] <- fct_cor(i242g, autre_P2)
5308. DF_cor[[31]][16] <- fct_cor(i242g, arbustive_P2)
5309. DF_cor[[32]][16] <- fct_cor(i242g, chromo_P2)
5310. DF_cor[[33]][16] <- fct_cor(i242g, abris_P2)
5311. DF_cor[[34]][16] <- fct_cor(i242ng, gram_1_P2)
5312. DF_cor[[35]][16] <- fct_cor(i242ng, gram_2_P2)
5313. DF_cor[[36]][16] <- fct_cor(i242ng, gram_3_P2)
5314. DF_cor[[37]][16] <- fct_cor(i242ng, dicot_1_P2)
5315. DF_cor[[38]][16] <- fct_cor(i242ng, dicot_2_P2)
5316. DF_cor[[39]][16] <- fct_cor(i242ng, dicot_3_P2)
5317. DF_cor[[40]][16] <- fct_cor(i242ng, autre_P2)
5318. DF_cor[[41]][16] <- fct_cor(i242ng, arbustive_P2)
5319. DF_cor[[42]][16] <- fct_cor(i242ng, chromo_P2)
5320. DF_cor[[43]][16] <- fct_cor(i242ng, abris_P2)
5321. # sortir correlation

```

```

5322. write.xlsx(DF_cor, "correlation.xlsx")
5323. ## ANOVA
5324. # NB : 40 AV2 croisees fixes (+ moyennes + ecarts types)
5325. # creer DF_AV
5326. DF_AV <- data.frame(rep(NA,40), rep(NA,40), rep(NA,40), rep(NA,40),
5327.   rep(NA,40), rep(NA,40), rep(NA,40), rep(NA,40),
5328.   rep(NA,40), rep(NA,40), rep(NA,40), rep(NA,40),
5329.   rep(NA,40), rep(NA,40), rep(NA,40), rep(NA,40),
5330.   rep(NA,40))
5331. colnames(DF_AV) <- c("moy_global", "ET_global", "p_interaction",
5332.   "moy_pature1", "ET_pature1", "moy_pature2", "ET_pature2",
5333.   "p_pature", "moy_ndama", "ET_ndama", "moy_zebu", "ET_zebu",
5334.   "moy_ndapol", "ET_ndapol", "moy_zepol", "ET_zepol", "p_race")
5335. rownames(DF_AV) <- c("cor_gram1_j", "cor_gram2_j", "cor_gram3_j",
5336.   "cor_dicot1_j", "cor_dicot2_j", "cor_dicot3_j",
5337.   "cor_autre_j", "cor_arbustive_j", "cor_chromo_j", "cor_abris_j",
5338.   "cor_gram1_n", "cor_gram2_n", "cor_gram3_n",
5339.   "cor_dicot1_n", "cor_dicot2_n", "cor_dicot3_n",
5340.   "cor_autre_n", "cor_arbustive_n", "cor_chromo_n", "cor_abris_n",
5341.   "cor_gram1_g", "cor_gram2_g", "cor_gram3_g",
5342.   "cor_dicot1_g", "cor_dicot2_g", "cor_dicot3_g",
5343.   "cor_autre_g", "cor_arbustive_g", "cor_chromo_g", "cor_abris_g",
5344.   "cor_gram1_ng", "cor_gram2_ng", "cor_gram3_ng",
5345.   "cor_dicot1_ng", "cor_dicot2_ng", "cor_dicot3_ng",
5346.   "cor_autre_ng", "cor_arbustive_ng",
5347.   "cor_chromo_ng", "cor_abris_ng")
5348. # fonction ecart type
5349. fct_ET <- function(DF) {
5350.   ET <- sqrt(sum((DF- mean(DF))^2)/length(DF))
5351.   return(ET)
5352. }
5353. # boucle pour stat par variable a tester
5354. for (i in 1:40) {
5355.   DF_AV[[1]][i] <- mean(DF_cor[,3+i])
5356.   DF_AV[[2]][i] <- fct_ET(DF_cor[,3+i])
5357.   DF_AV[[3]][i] <- anova(aov(
5358.     as.formula(paste(names(DF_cor[3+i]),
5359.       "pature * race", sep = "~")), DF_cor))[,3,5]
5360.   DF_AV[[4]][i] <- mean(DF_cor[DF_cor$pature==1,3+i])
5361.   DF_AV[[5]][i] <- fct_ET(DF_cor[DF_cor$pature==1,3+i])
5362.   DF_AV[[6]][i] <- mean(DF_cor[DF_cor$pature==2,3+i])
5363.   DF_AV[[7]][i] <- fct_ET(DF_cor[DF_cor$pature==2,3+i])
5364.   DF_AV[[8]][i] <- anova(aov(
5365.     as.formula(paste(names(DF_cor[3+i]),
5366.       "pature * race", sep = "~")), DF_cor))[,1,5]
5367.   DF_AV[[9]][i] <- mean(DF_cor[DF_cor$race==1,3+i])
5368.   DF_AV[[10]][i] <- fct_ET(DF_cor[DF_cor$race==1,3+i])
5369.   DF_AV[[11]][i] <- mean(DF_cor[DF_cor$race==2,3+i])
5370.   DF_AV[[12]][i] <- fct_ET(DF_cor[DF_cor$race==2,3+i])
5371.   DF_AV[[13]][i] <- mean(DF_cor[DF_cor$race==3,3+i])
5372.   DF_AV[[14]][i] <- fct_ET(DF_cor[DF_cor$race==3,3+i])
5373.   DF_AV[[15]][i] <- mean(DF_cor[DF_cor$race==4,3+i])
5374.   DF_AV[[16]][i] <- fct_ET(DF_cor[DF_cor$race==4,3+i])
5375.   DF_AV[[17]][i] <- anova(aov(
5376.     as.formula(paste(names(DF_cor[3+i]),
5377.       "pature * race", sep = "~")), DF_cor))[,2,5]
5378.   }
5379. # sortie resultats
5380. write.xlsx(DF_AV, "geostatistiques.xlsx")
5381. ## ! p_interaction pas > 0.05 pour toutes les variables !
5382. # NB : 4 variables avec interaction significative
5383. # NB : 4 variables * 4 races = 16 AV1 a 1 facteur fixe (pature)
5384. # creer nouvelle DF_corr et nouvelle DF_AV
5385. DF_cor_sup <- data.frame(DF_cor$pature, DF_cor$race, DF_cor$bete,
5386.   DF_cor$gram1_n, DF_cor$chromo_n,
5387.   DF_cor$gram1_ng, DF_cor$chromo_ng)
5388. colnames(DF_cor_sup) <- c("pature", "race", "bete",
5389.   "gram1_n", "chromo_n", "gram1_ng", "chromo_ng")
5390. rownames(DF_cor_sup) <- c("1NA", "1NB", "1ZA", "1ZB",
5391.   "1NpA", "1NpB", "1ZpA", "1ZpB", "2NA", "2NB", "2ZA", "2ZB",
5392.   "2NpA", "2NpB", "2ZpA", "2ZpB")
5393. DF_AV_sup <- data.frame(rep(NA,4), rep(NA,4), rep(NA,4), rep(NA,4),
5394.   rep(NA,4), rep(NA,4), rep(NA,4), rep(NA,4),
5395.   rep(NA,4), rep(NA,4), rep(NA,4), rep(NA,4),
5396.   rep(NA,4), rep(NA,4), rep(NA,4), rep(NA,4),
5397.   rep(NA,4), rep(NA,4), rep(NA,4), rep(NA,4))
5398. colnames(DF_AV_sup) <- c("moy_ndama_pat1", "ET_ndama_pat1",
5399.   "moy_ndama_pat2", "ET_ndama_pat2", "p_ndama",
5400.   "moy_zebu_pat1", "ET_zebu_pat1",
5401.   "moy_zebu_pat2", "ET_zebu_pat2", "p_zebu",
5402.   "moy_ndapol_pat1", "ET_ndapol_pat1",
5403.   "moy_ndapol_pat2", "ET_ndapol_pat2", "p_ndapol",
5404.   "moy_zepol_pat1", "ET_zepol_pat1",
5405.   "moy_zepol_pat2", "ET_zepol_pat2", "p_zepol")
5406. rownames(DF_AV_sup) <- c("cor_gram1_n", "cor_chromo_n",
5407.   "cor_gram1_ng", "cor_chromo_ng")
5408. # double boucle
5409. for (r in 1:4) {
5410.   for (i in 1:4) {
5411.     DF_AV_sup[(((r-1)*5)+1)][i] <- mean(
5412.       DF_cor_sup[(DF_cor_sup$race==r & DF_cor_sup$pature==1),i+3])
5413.     DF_AV_sup[(((r-1)*5)+2)][i] <- fct_ET(
5414.       DF_cor_sup[(DF_cor_sup$race==r & DF_cor_sup$pature==1),i+3])
5415.     DF_AV_sup[(((r-1)*5)+3)][i] <- mean(
5416.       DF_cor_sup[(DF_cor_sup$race==r & DF_cor_sup$pature==2),i+3])
5417.     DF_AV_sup[(((r-1)*5)+4)][i] <- fct_ET(
5418.       DF_cor_sup[(DF_cor_sup$race==r & DF_cor_sup$pature==2),i+3])
5419.     DF_AV_sup[(((r-1)*5)+5)][i] <- anova(aov(
5420.       as.formula(paste(names(DF_cor_sup[i+3]),
5421.         "pature", sep = "~")), DF_cor_sup[DF_cor_sup$race==r,]))[,1,5]
5422.   }
5423. }
5424. # sortie resultats
5425. write.xlsx(DF_AV_sup, "geostatistiques_sup.xlsx")

```

Annexe 30 : tableau récapitulatif des moyennes et des écarts-types de toutes les variables pour tous les groupes d'animaux

Sources de biais	Global	Pâturage 1	Pâturage 2	Ndama	Zébu Goudali	Ndapol	Zépol
Pluviométrie de la journée d'observation (mm)	7,7 ± 7,7	3,0 ± 5,2	12,3 ± 17,6	8,1 ± 8,7	8,0 ± 8,5	14,5 ± 22,6	0,0 ± 0,0
Gain quotidien moyen (kg/j)	0,071 ± 0,071	0,069 ± 0,326	0,073 ± 0,436	-0,045 ± 0,573	0,018 ± 0,292	0,083 ± 0,202	0,228 ± 0,311
Observations directes continues	Global	Pâturage 1	Pâturage 2	Ndama	Zébu Goudali	Ndapol	Zépol
Attaques perpétrées sur d'autres animaux :							
• Nombre total	2,1 ± 2,2	3,0 ± 2,4	1,1 ± 1,6	2,0 ± 2,4	1,3 ± 1,6	1,8 ± 1,9	3,3 ± 2,4
• Nombre de Ndams	0,6 ± 0,7	0,8 ± 0,7	0,4 ± 0,7	0,5 ± 0,5	0,0 ± 0,0	0,5 ± 0,5	1,3 ± 0,8
• Nombre de Zébus Goudalis	0,5 ± 1,1	0,4 ± 0,7	0,6 ± 1,3	0,0 ± 0,0	0,0 ± 0,0	1,3 ± 1,6	0,8 ± 0,8
• Nombre de Ndapols	0,4 ± 0,9	0,6 ± 1,1	0,1 ± 0,3	0,0 ± 0,0	1,0 ± 1,2	0,0 ± 0,0	0,5 ± 0,9
• Nombre de Zépols	0,4 ± 1,0	0,9 ± 1,3	0,0 ± 0,0	1,0 ± 1,7	0,3 ± 0,4	0,0 ± 0,0	0,5 ± 0,5
• Nombre de bêtes hors protocole	0,2 ± 0,4	0,4 ± 0,5	0,0 ± 0,0	0,5 ± 0,5	0,0 ± 0,0	0,0 ± 0,0	0,3 ± 0,4
• Intensité des attaques (0-3)	1,4 ± 0,4	1,3 ± 0,3	1,3 ± 0,5	1,7 ± 0,2	1,5 ± 0,5	1,0 ± 0,0	1,3 ± 0,3
Broutage de ligneux :							
• Nombre total	2,7 ± 2,3	3,8 ± 2,4	1,6 ± 1,7	1,0 ± 1,0	2,5 ± 1,1	3,5 ± 3,5	3,8 ± 1,5
• Nombre de <i>Gardenia ternifolia</i>	1,4 ± 1,5	1,6 ± 1,4	1,3 ± 1,5	0,3 ± 0,4	1,5 ± 1,1	1,3 ± 1,6	2,8 ± 1,1
• Nombre de <i>Sarcocephalus latifolius</i>	0,3 ± 0,4	0,5 ± 0,5	0,0 ± 0,0	0,0 ± 0,0	0,3 ± 0,4	0,5 ± 0,5	0,3 ± 0,4
• Nombre de <i>Alchornea cordifolia</i>	0,1 ± 0,3	0,3 ± 0,4	0,0 ± 0,0	0,0 ± 0,0	0,0 ± 0,0	0,3 ± 0,4	0,3 ± 0,4
• Nombre de <i>Bridelia ferruginea</i>	0,9 ± 1,1	1,4 ± 1,2	0,4 ± 0,7	0,8 ± 0,8	0,8 ± 0,8	1,5 ± 1,5	0,5 ± 0,9
• Nombre de ligneux d'une autre espèce	0,0 ± 0,0	0,0 ± 0,0	0,0 ± 0,0	0,0 ± 0,0	0,0 ± 0,0	0,0 ± 0,0	0,0 ± 0,0
• Nombre total de bouchées	9,8 ± 11,2	15,0 ± 13,2	4,6 ± 4,8	3,8 ± 4,9	9,0 ± 4,6	15,0 ± 19,5	11,5 ± 3,0
• Hauteur des bouchées (cm)	77,2 ± 27,1	83,9 ± 22,4	56,1 ± 37,7	82,5 ± 17,5	69,8 ± 9,0	45,2 ± 35,7	95,6 ± 31,0
Nombre de flehments	1,1 ± 2,7	2,0 ± 3,6	0,3 ± 0,4	3,8 ± 4,5	0,0 ± 0,0	0,5 ± 0,5	0,3 ± 0,4
Léchage d'autres animaux :							
• Nombre d'animaux léchés	1,8 ± 2,3	2,1 ± 2,7	1,4 ± 1,7	0,0 ± 0,0	4,8 ± 1,9	0,5 ± 0,9	1,8 ± 1,5
• Nombre total de coups de langue donnés	76,8 ± 153,4	123,8 ± 200,5	29,8 ± 49,4	0,0 ± 0,0	227,8 ± 220,3	8,5 ± 14,7	70,8 ± 109,5
Périodes de meuglements :							
• Nombre de périodes	2,0 ± 4,6	1,0 ± 1,7	3,0 ± 6,2	6,5 ± 7,4	0,0 ± 0,0	1,5 ± 1,5	0,0 ± 0,0

• Nombre total de cris	19,8 ± 54,4	7,5 ± 16,6	32,0 ± 73,1	70,8 ± 90,6	0,0 ± 0,0	8,3 ± 10,5	0,0 ± 0,0
Périodes passées allongé :							
• Nombre de périodes	2,3 ± 1,9	2,4 ± 2,2	2,3 ± 1,5	0,8 ± 0,8	3,8 ± 0,8	1,5 ± 1,7	3,3 ± 1,9
• Temps total (minutes)	35,3 ± 34,2	40,5 ± 36,5	30,0 ± 30,9	17,5 ± 19,2	52,0 ± 23,8	8,3 ± 8,3	63,3 ± 39,8
Observations directes discontinues	Global	Pâturage 1	Pâturage 2	Ndama	Zébu Goudali	Ndapol	Zépol
Nombre total de bêtes présentes dans le groupe	5,0 ± 1,0	5,3 ± 0,5	4,8 ± 1,2	4,6 ± 1,5	5,7 ± 0,4	4,6 ± 0,5	5,3 ± 0,3
Distance à laquelle se trouve la bête la plus proche (m)	4,4 ± 1,3	4,2 ± 1,6	4,7 ± 0,8	5,9 ± 0,4	3,3 ± 1,2	5,2 ± 0,3	3,4 ± 0,4
Temps passé seul (heures)	1,5 ± 2,1	1,1 ± 1,2	1,9 ± 2,7	3,1 ± 3,2	0,1 ± 0,1	2,5 ± 1,0	0,3 ± 0,2
Temps passé avec chaque catégorie de bêtes (heures) :							
• La bête le plus souvent la plus proche	8,4 ± 2,5	9,0 ± 2,2	7,9 ± 2,7	5,9 ± 2,6	10,9 ± 0,8	7,0 ± 0,8	9,9 ± 0,8
• Au moins un Ndama	5,3 ± 1,5	5,7 ± 1,6	4,8 ± 1,2	5,5 ± 1,5	4,8 ± 1,4	5,3 ± 1,5	5,3 ± 1,4
• Tous les Ndamas	3,9 ± 1,6	3,8 ± 2,0	4,1 ± 1,2	5,5 ± 1,5	3,0 ± 1,7	3,5 ± 1,0	3,7 ± 0,9
• Au moins un zébu Goudali	8,4 ± 2,7	8,4 ± 2,7	8,3 ± 2,6	5,8 ± 2,4	10,5 ± 1,2	6,6 ± 1,3	10,6 ± 0,3
• Tous les zébus Goudalis	7,6 ± 2,7	8,0 ± 2,9	7,3 ± 2,5	4,9 ± 2,3	10,5 ± 1,2	6,0 ± 1,2	9,1 ± 0,8
• Au moins un Ndapol	7,3 ± 1,9	7,1 ± 1,3	7,6 ± 2,4	6,6 ± 2,8	8,4 ± 0,8	6,3 ± 1,2	8,0 ± 1,4
• Tous les Ndapols	5,2 ± 1,5	4,8 ± 1,0	5,6 ± 1,8	3,7 ± 1,2	5,6 ± 1,5	6,3 ± 1,2	5,1 ± 0,9
• Au moins un Zépol	8,2 ± 2,7	8,2 ± 2,6	8,1 ± 2,8	5,7 ± 2,8	11,0 ± 0,9	6,8 ± 1,2	9,1 ± 1,6
• Tous les Zépols	6,9 ± 2,5	6,9 ± 2,4	6,9 ± 2,6	4,4 ± 2,1	8,5 ± 1,0	5,4 ± 1,0	9,1 ± 1,6
• Au moins une bête hors protocole	3,0 ± 2,6	2,8 ± 2,2	3,3 ± 2,9	2,4 ± 0,8	5,3 ± 2,9	3,2 ± 2,6	1,3 ± 1,7
• Toutes les bêtes hors protocole	1,6 ± 2,2	0,6 ± 0,7	2,5 ± 2,6	0,8 ± 0,6	3,0 ± 3,3	2,0 ± 1,8	0,5 ± 0,4
Temps pendant lequel une bête de chaque catégorie est la bête la plus proche (heures) :							
• La bête le plus souvent la plus proche	3,5 ± 1,5	3,9 ± 1,6	3,1 ± 1,3	2,2 ± 0,8	5,5 ± 0,7	2,2 ± 0,2	3,9 ± 0,6
• Un Ndama	1,3 ± 1,0	1,5 ± 1,2	1,1 ± 0,7	1,5 ± 1,1	0,6 ± 0,5	2,1 ± 1,0	0,9 ± 0,6
• Un zébu Goudali	3,3 ± 2,0	3,5 ± 2,3	3,2 ± 1,6	1,7 ± 1,3	4,7 ± 1,8	2,0 ± 1,0	4,9 ± 1,2
• Un Ndapol	1,9 ± 1,1	2,0 ± 1,4	1,9 ± 0,7	3,0 ± 1,5	1,2 ± 0,4	1,5 ± 0,4	2,0 ± 0,8
• Un Zépol	3,2 ± 1,3	3,3 ± 1,2	3,1 ± 1,4	1,8 ± 1,3	4,3 ± 0,9	3,1 ± 0,7	3,8 ± 0,7
• Une bête hors protocole	0,8 ± 0,8	0,7 ± 0,8	0,8 ± 0,8	1,0 ± 0,7	1,2 ± 0,9	0,8 ± 0,7	0,1 ± 0,1

Vitesse des bêtes et accès à l'eau et au sel : jour	Global	Pâturage 1	Pâturage 2	Ndama	Zébu Goudali	Ndapol	Zépol
Vitesse moyenne (m/h)	427 ± 85	419 ± 77	435 ± 91	430 ± 79	429 ± 108	433 ± 88	416 ± 54
Temps passé entre 0 et 0,5 m/s (%)	91,32 ± 1,59	91,63 ± 0,86	91,00 ± 2,03	91,75 ± 1,66	91,20 ± 1,41	90,93 ± 2,02	91,39 ± 0,98
Temps passé entre 0,5 et 2 m/s (%)	8,64 ± 1,57	8,33 ± 0,84	8,95 ± 2,01	8,20 ± 1,65	8,78 ± 1,41	9,00 ± 1,97	8,59 ± 0,97
Temps passé à plus de 2 m/s (%)	0,04 ± 0,04	0,04 ± 0,04	0,04 ± 0,04	0,06 ± 0,05	0,03 ± 0,02	0,06 ± 0,05	0,02 ± 0,01
Temps passé à moins de 10 m d'un bac à sel (%)	1,04 ± 1,19	0,26 ± 0,31	1,81 ± 1,24	0,82 ± 0,74	1,00 ± 0,77	1,44 ± 1,93	0,89 ± 0,76
Temps passé à moins de 10 m d'un point d'eau (%)	2,77 ± 1,73	3,00 ± 2,02	2,54 ± 1,35	2,87 ± 0,87	3,29 ± 1,81	2,70 ± 2,28	2,22 ± 1,48
Vitesse des bêtes et accès à l'eau et au sel : nuit	Global	Pâturage 1	Pâturage 2	Ndama	Zébu Goudali	Ndapol	Zépol
Vitesse moyenne (m/h)	96 ± 40	88 ± 38	104 ± 39	91 ± 40	120 ± 55	76 ± 25	97 ± 6
Temps passé entre 0 et 0,5 m/s (%)	98,32 ± 0,65	98,52 ± 0,57	98,12 ± 0,66	98,48 ± 0,53	98,11 ± 0,90	98,57 ± 0,53	98,11 ± 0,37
Temps passé entre 0,5 et 2 m/s (%)	1,67 ± 0,64	1,47 ± 0,57	1,87 ± 0,65	1,51 ± 0,52	1,88 ± 0,90	1,42 ± 0,52	1,87 ± 0,37
Temps passé à plus de 2 m/s (%)	0,01 ± 0,01	0,01 ± 0,01	0,01 ± 0,01	0,01 ± 0,01	0,01 ± 0,01	0,01 ± 0,01	0,01 ± 0,01
Temps passé à moins de 10 m d'un bac à sel (%)	1,40 ± 3,86	2,75 ± 5,11	0,04 ± 0,12	4,27 ± 6,80	0,04 ± 0,06	0,93 ± 1,33	0,35 ± 0,35
Temps passé à moins de 10 m d'un point d'eau (%)	4,47 ± 9,18	8,92 ± 11,35	0,03 ± 0,03	5,23 ± 8,95	9,60 ± 13,85	0,37 ± 0,63	2,68 ± 4,19
Vitesse des bêtes et accès à l'eau et au sel : grazing	Global	Pâturage 1	Pâturage 2	Ndama	Zébu Goudali	Ndapol	Zépol
Vitesse moyenne (m/h)	307 ± 95	294 ± 79	321 ± 107	280 ± 44	349 ± 135	314 ± 92	286 ± 67
Temps passé entre 0 et 0,5 m/s (%)	95,25 ± 1,71	95,61 ± 1,14	94,89 ± 2,07	95,82 ± 0,94	94,55 ± 2,23	94,96 ± 1,79	95,67 ± 1,23
Temps passé entre 0,5 et 2 m/s (%)	4,73 ± 1,70	4,37 ± 1,13	5,09 ± 2,06	4,16 ± 0,94	5,43 ± 2,22	5,02 ± 1,78	4,32 ± 1,22
Temps passé à plus de 2 m/s (%)	0,02 ± 0,01	0,01 ± 0,01	0,02 ± 0,01	0,02 ± 0,01	0,01 ± 0,01	0,02 ± 0,01	0,01 ± 0,01
Temps passé à moins de 10 m d'un bac à sel (%)	1,10 ± 1,54	0,52 ± 0,75	1,68 ± 1,87	1,14 ± 0,82	0,80 ± 0,81	1,80 ± 2,66	0,65 ± 0,52
Temps passé à moins de 10 m d'un point d'eau (%)	1,37 ± 0,86	1,52 ± 1,05	1,23 ± 0,59	1,24 ± 0,66	1,62 ± 0,60	1,12 ± 0,70	1,51 ± 1,24
Vitesse des bêtes et accès à l'eau et au sel : non grazing	Global	Pâturage 1	Pâturage 2	Ndama	Zébu Goudali	Ndapol	Zépol
Vitesse moyenne (m/h)	256 ± 58	241 ± 63	271 ± 48	272 ± 72	249 ± 62	234 ± 59	269 ± 12
Temps passé entre 0 et 0,5 m/s (%)	94,14 ± 1,17	94,55 ± 1,04	93,74 ± 1,15	94,16 ± 1,38	94,47 ± 0,84	94,33 ± 1,37	93,60 ± 0,74
Temps passé entre 0,5 et 2 m/s (%)	5,82 ± 1,16	5,42 ± 1,03	6,23 ± 1,15	5,80 ± 1,37	5,51 ± 0,83	5,62 ± 1,35	6,38 ± 0,75
Temps passé à plus de 2 m/s (%)	0,03 ± 0,03	0,03 ± 0,03	0,03 ± 0,03	0,04 ± 0,04	0,02 ± 0,02	0,05 ± 0,04	0,02 ± 0,02

Temps passé à moins de 10 m d'un bac à sel (%)	1,32 ± 2,46	1,96 ± 3,35	0,69 ± 0,31	3,22 ± 4,29	0,38 ± 0,28	0,96 ± 0,84	0,73 ± 0,23
Temps passé à moins de 10 m d'un point d'eau (%)	4,84 ± 7,14	8,14 ± 8,91	1,53 ± 0,92	6,03 ± 7,38	8,54 ± 10,36	1,70 ± 1,59	3,09 ± 3,40
Position des bêtes et liens avec la végétation : jour	Global	Pâtûre 1	Pâtûre 2	Ndama	Zébu Goudali	Ndapol	Zépol
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation :							
• Graminées rampantes ou gazonnantes	0,17 ± 0,09	0,19 ± 0,07	0,14 ± 0,10	0,19 ± 0,06	0,17 ± 0,05	0,13 ± 0,09	0,18 ± 0,13
• Graminées cespiteuses de moins de 1 m	-0,02 ± 0,13	-0,14 ± 0,03	0,09 ± 0,08	-0,02 ± 0,13	-0,03 ± 0,15	-0,01 ± 0,11	-0,03 ± 0,12
• Graminées cespiteuses de plus de 1 m	-0,03 ± 0,12	-0,13 ± 0,04	0,07 ± 0,08	-0,04 ± 0,13	-0,03 ± 0,13	-0,01 ± 0,08	-0,06 ± 0,12
• <i>Calopogonium mucunoides</i>	-0,12 ± 0,08	-0,12 ± 0,04	-0,12 ± 0,10	-0,19 ± 0,06	-0,07 ± 0,09	-0,09 ± 0,06	-0,14 ± 0,01
• <i>Sida cf. acuta</i>	-0,05 ± 0,04	-0,06 ± 0,02	-0,05 ± 0,05	-0,06 ± 0,05	-0,06 ± 0,02	-0,03 ± 0,04	-0,07 ± 0,02
• <i>Mitracarpus hirtus</i>	-0,03 ± 0,09	0,03 ± 0,06	-0,10 ± 0,07	-0,07 ± 0,05	-0,02 ± 0,11	-0,01 ± 0,10	-0,04 ± 0,09
• Végétation herbacée hors 6 critères précédents	0,00 ± 0,10	-0,05 ± 0,08	0,05 ± 0,10	-0,05 ± 0,08	0,01 ± 0,09	-0,04 ± 0,08	0,07 ± 0,10
• Végétation arbustive à l'exception de <i>C. odorata</i>	0,06 ± 0,06	0,07 ± 0,07	0,05 ± 0,05	0,12 ± 0,05	0,03 ± 0,04	0,02 ± 0,05	0,07 ± 0,06
• <i>C. odorata</i>	-0,03 ± 0,12	-0,01 ± 0,12	-0,05 ± 0,12	-0,10 ± 0,13	-0,03 ± 0,09	0,01 ± 0,14	0,00 ± 0,09
• Abris contre le soleil et la pluie	0,06 ± 0,09	0,09 ± 0,08	0,03 ± 0,09	0,00 ± 0,07	0,08 ± 0,07	0,07 ± 0,10	0,09 ± 0,10
Position des bêtes et liens avec la végétation : nuit	Global	Pâtûre 1	Pâtûre 2	Ndama	Zébu Goudali	Ndapol	Zépol
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation :							
• Graminées rampantes ou gazonnantes	0,04 ± 0,06	0,09 ± 0,03	-0,01 ± 0,04	0,05 ± 0,03	0,04 ± 0,06	0,03 ± 0,07	0,05 ± 0,08
• Graminées cespiteuses de moins de 1 m	-0,01 ± 0,05	0,02 ± 0,02	-0,04 ± 0,04	-0,02 ± 0,03	0,00 ± 0,06	0,01 ± 0,04	-0,03 ± 0,05
• Graminées cespiteuses de plus de 1 m	0,01 ± 0,06	0,05 ± 0,05	-0,02 ± 0,04	-0,01 ± 0,05	0,01 ± 0,07	0,05 ± 0,05	0,00 ± 0,05
• <i>Calopogonium mucunoides</i>	0,00 ± 0,05	0,02 ± 0,01	-0,02 ± 0,06	-0,03 ± 0,06	-0,01 ± 0,02	0,04 ± 0,02	0,01 ± 0,04
• <i>Sida cf. acuta</i>	-0,05 ± 0,03	-0,07 ± 0,02	-0,03 ± 0,01	-0,04 ± 0,02	-0,06 ± 0,03	-0,04 ± 0,02	-0,05 ± 0,03
• <i>Mitracarpus hirtus</i>	-0,06 ± 0,10	-0,15 ± 0,03	0,04 ± 0,03	-0,06 ± 0,10	-0,04 ± 0,12	-0,05 ± 0,08	-0,07 ± 0,08
• Végétation herbacée hors 6 critères précédents	0,00 ± 0,05	0,03 ± 0,03	-0,03 ± 0,04	0,02 ± 0,03	-0,04 ± 0,04	0,00 ± 0,04	0,00 ± 0,05
• Végétation arbustive à l'exception de <i>C. odorata</i>	0,02 ± 0,04	0,03 ± 0,03	0,01 ± 0,05	0,04 ± 0,06	0,00 ± 0,03	0,01 ± 0,01	0,03 ± 0,04
• <i>C. odorata</i>	-0,05 ± 0,04	-0,08 ± 0,03	-0,02 ± 0,03	-0,04 ± 0,02	-0,06 ± 0,03	-0,04 ± 0,05	-0,04 ± 0,06
• Abris contre le soleil et la pluie	-0,03 ± 0,03	-0,04 ± 0,02	-0,01 ± 0,02	-0,03 ± 0,02	-0,04 ± 0,03	-0,02 ± 0,03	-0,02 ± 0,02

Position des bêtes et liens avec la végétation : grazing	Global	Pâturage 1	Pâturage 2	Ndama	Zébu Goudali	Ndapol	Zépol
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation :							
• Graminées rampantes ou gazonnantes	0,16 ± 0,13	0,22 ± 0,06	0,10 ± 0,15	0,14 ± 0,13	0,19 ± 0,04	0,14 ± 0,09	0,17 ± 0,19
• Graminées cespiteuses de moins de 1 m	-0,01 ± 0,13	-0,09 ± 0,07	0,08 ± 0,11	-0,05 ± 0,01	0,03 ± 0,18	0,04 ± 0,12	-0,04 ± 0,10
• Graminées cespiteuses de plus de 1 m	0,00 ± 0,12	-0,07 ± 0,08	0,06 ± 0,11	-0,04 ± 0,07	0,01 ± 0,12	0,07 ± 0,06	-0,04 ± 0,16
• <i>Calopogonium mucunoides</i>	-0,07 ± 0,09	-0,08 ± 0,05	-0,06 ± 0,12	-0,14 ± 0,06	-0,05 ± 0,09	-0,01 ± 0,11	-0,10 ± 0,03
• <i>Sida cf. acuta</i>	-0,09 ± 0,06	-0,10 ± 0,04	-0,08 ± 0,08	-0,09 ± 0,05	-0,11 ± 0,05	-0,04 ± 0,07	-0,14 ± 0,03
• <i>Mitracarpus hirtus</i>	-0,05 ± 0,11	-0,02 ± 0,09	-0,08 ± 0,11	-0,06 ± 0,05	-0,04 ± 0,13	-0,06 ± 0,10	-0,03 ± 0,12
• Végétation herbacée hors 6 critères précédents	0,00 ± 0,11	-0,04 ± 0,05	0,04 ± 0,13	-0,05 ± 0,05	0,00 ± 0,11	0,02 ± 0,10	0,02 ± 0,13
• Végétation arbustive à l'exception de <i>C. odorata</i>	0,07 ± 0,07	0,07 ± 0,07	0,07 ± 0,07	0,10 ± 0,07	0,04 ± 0,07	0,05 ± 0,05	0,10 ± 0,07
• <i>C. odorata</i>	-0,03 ± 0,10	-0,02 ± 0,09	-0,04 ± 0,11	-0,04 ± 0,07	-0,05 ± 0,02	-0,03 ± 0,14	-0,01 ± 0,12
• Abris contre le soleil et la pluie	0,00 ± 0,07	0,04 ± 0,06	-0,03 ± 0,07	0,00 ± 0,07	-0,01 ± 0,03	0,02 ± 0,07	0,00 ± 0,10
Position des bêtes et liens avec la végétation : non grazing	Global	Pâturage 1	Pâturage 2	Ndama	Zébu Goudali	Ndapol	Zépol
Corrélation entre la présence des bêtes et celle de chaque critère de la végétation :							
• Graminées rampantes ou gazonnantes	0,06 ± 0,06	0,11 ± 0,04	0,01 ± 0,04	0,07 ± 0,03	0,06 ± 0,07	0,04 ± 0,06	0,07 ± 0,08
• Graminées cespiteuses de moins de 1 m	-0,01 ± 0,04	0,01 ± 0,03	-0,03 ± 0,04	-0,02 ± 0,01	0,00 ± 0,06	0,01 ± 0,03	-0,02 ± 0,04
• Graminées cespiteuses de plus de 1 m	0,02 ± 0,05	0,05 ± 0,05	-0,01 ± 0,03	-0,01 ± 0,05	0,01 ± 0,05	0,05 ± 0,05	0,01 ± 0,04
• <i>Calopogonium mucunoides</i>	-0,01 ± 0,05	0,01 ± 0,01	-0,03 ± 0,05	-0,04 ± 0,06	-0,02 ± 0,03	0,02 ± 0,02	-0,01 ± 0,04
• <i>Sida cf. acuta</i>	-0,05 ± 0,03	-0,07 ± 0,03	-0,03 ± 0,01	-0,04 ± 0,02	-0,07 ± 0,04	-0,04 ± 0,02	-0,06 ± 0,04
• <i>Mitracarpus hirtus</i>	-0,07 ± 0,10	-0,16 ± 0,03	0,03 ± 0,03	-0,07 ± 0,09	-0,06 ± 0,12	-0,05 ± 0,08	-0,09 ± 0,08
• Végétation herbacée hors 6 critères précédents	0,00 ± 0,05	0,03 ± 0,04	-0,02 ± 0,03	0,03 ± 0,03	-0,03 ± 0,04	0,00 ± 0,03	0,02 ± 0,06
• Végétation arbustive à l'exception de <i>C. odorata</i>	0,02 ± 0,05	0,04 ± 0,03	0,01 ± 0,06	0,05 ± 0,06	0,00 ± 0,05	0,00 ± 0,01	0,03 ± 0,05
• <i>C. odorata</i>	-0,05 ± 0,04	-0,09 ± 0,03	-0,02 ± 0,03	-0,05 ± 0,02	-0,07 ± 0,03	-0,04 ± 0,05	-0,05 ± 0,06
• Abris contre le soleil et la pluie	-0,02 ± 0,03	-0,04 ± 0,03	0,00 ± 0,03	-0,03 ± 0,02	-0,02 ± 0,04	-0,02 ± 0,03	-0,01 ± 0,03