



R  
M  
T

Modélisation  
& Agriculture

[www.modelia.org](http://www.modelia.org)



Exploration et analyse de modèles  
pour l'élevage et l'environnement :  
Création de ressources  
pédagogiques sous le logiciel R.

M2 - Modélisation des Systèmes Écologiques

Stage de fin d'étude

Université Paul Sabatier Toulouse 3

Sous la direction de François Brun (ACTA)

Juliette ADRIAN

20 juin 2013

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Modèles retenus</b>	<b>5</b>
2.1	Modèle de lactation de Neal . . . . .	6
2.2	Modèle de croissance de carcasse de Hoch . . . . .	7
2.3	Ecriture des simulateurs sous R . . . . .	8
<b>3</b>	<b>Evaluation des performances des modèles</b>	<b>8</b>
3.1	Critères d'évaluation . . . . .	9
3.2	Exemple de résultat obtenu . . . . .	10
<b>4</b>	<b>Estimation bayésienne des paramètres</b>	<b>11</b>
4.1	Importance Sampling . . . . .	12
4.1.1	Algorithme . . . . .	12
4.1.2	Résultats obtenus . . . . .	13
4.2	Metropolis Hastings . . . . .	15
4.2.1	Algorithme . . . . .	15
4.2.2	Résultats obtenus . . . . .	17
<b>5</b>	<b>Contribution au package « ZeBook »</b>	<b>19</b>
<b>6</b>	<b>Conclusions et perspectives</b>	<b>20</b>
<b>7</b>	<b>Bibliographie</b>	<b>21</b>
<b>8</b>	<b>Annexes</b>	<b>23</b>

# 1 Introduction

Dans le domaine agricole, l'utilisation de la modélisation s'est fortement développée depuis une vingtaine d'années comme outil d'investigation pour la recherche ou l'accompagnement des acteurs pour le développement. Par modélisation, on entend ici modélisation dynamique des systèmes d'intérêt agricole, c'est-à-dire une représentation mathématique et simplifiée du système « réel » décrivant de manière formelle les éléments de ce système et explicitant les relations entre ces éléments (les processus physique, biologique, voire socio-économique). Le plus souvent, les modèles utilisés sont présentés sous la forme d'un système d'équations différentielles ou à différence qui représentent la dynamique des différentes composantes que l'on peut retrouver dans le système. Cela permet notamment de simuler l'évolution d'un système au cours du temps en fonction de variables d'entrée.

Contrairement aux modèles statistiques classiques, en décrivant le fonctionnement du système de manière plus mécaniste, cette voie de modélisation permettrait de réaliser des simulations pour de nouvelles situations, pour lesquelles nous n'avons pas d'information sur le comportement global du système. Ainsi, de nombreux ingénieurs et chercheurs du domaine agricole cherchent à utiliser ce type de modélisation afin d'analyser l'impact de nouvelles conditions et pratiques sur des systèmes d'intérêt.

Par exemple, pour les questions concernant l'élevage et l'environnement, ces modèles peuvent être utilisés pour explorer les effets de changements dans l'alimentation des animaux (ration et apport d'énergie) et des pratiques d'élevage sur leur développement (croissance) et leur production (lait) ou l'impact sur l'environnement (émission de gaz à effet de serre). En tant qu'outil pour la recherche et le développement, ils permettent d'évaluer et d'aider à la conception de nouvelles pratiques agricoles. Ils peuvent servir pour réaliser des expérimentations virtuelles afin de rationaliser l'effort expérimental (extrêmement coûteux en élevage). Pour les acteurs du monde agricole (éleveurs ou conseillers agricoles), sous la forme d'outils d'aide à la décision, ces modèles peuvent dans certains cas servir à aider au pilotage effectif des systèmes de production (choix des rations avec le modèle INRAtion par exemple).

Le développement de ces modèles demande à l'utilisateur une connaissance poussée du fonctionnement du système étudié (zootechnie, agronomie). Cependant, l'élaboration et l'utilisation de ces modèles requièrent également des méthodes relevant du domaine des mathématiques (formalisation sous forme d'équations), des statistiques (exploration et analyse) ou de l'informatique (programmation du simulateur). Pour autant, maîtriser

ces différentes disciplines n'est pas facile et si ces méthodes ont atteint une certaine maturité et sont documentées, elles peuvent rester délicates à utiliser pour les ingénieurs et chercheurs agronomes. L'idée est donc de renforcer les compétences de ces acteurs en modélisation afin de les rendre plus autonomes pour concevoir et utiliser ces modèles. Il s'agit notamment qu'ils s'approprient les méthodes mathématiques et statistiques leur permettant d'utiliser ces modèles (évaluation, estimation de paramètres, analyse d'incertitude ou de sensibilité). De plus, certaines de ces méthodes sont relativement accessibles du point de vue des concepts sous-jacents et avec la puissance de calcul des ordinateurs et les facilités offertes par des langages comme R, elles peuvent être mises en application par tous les modélisateurs.

Le Réseau Mixte Technique (RMT) Modélisation et Agriculture ([www.modelia.org](http://www.modelia.org)) a pour vocation d'organiser des échanges autour de la modélisation pour l'agronomie et l'élevage. Ainsi, le réseau organise régulièrement des échanges, des formations et des projets méthodologiques ayant pour but d'améliorer le développement et l'utilisation de modèles par les acteurs de la recherche et développement, essentiellement de l'INRA et des Instituts Techniques Agricoles (dont l'Institut de l'élevage, IFIP – institut du porc et ITAVI pour l'aviculture). Cette animation nationale est assurée par François Brun (ACTA, tête de réseau des Instituts techniques agricoles), David Makowski (INRA) et François Piraux (Arvalis - Institut du végétal). Le public visé se compose de chercheurs, ingénieurs et doctorants en agronomie qui sont amenés à aborder leurs sujets d'étude par le biais de la modélisation ou à élaborer des outils d'aide à la décision basés sur des modèles.

Les formations organisées par le réseau peuvent concerner la conception et l'élaboration du modèle (modélisation conceptuelle et aspects informatiques), mais également peuvent se focaliser sur l'exploration et l'analyse des modèles (mobilisant différentes méthodes mathématiques et statistiques). Depuis 2010, le RMT a mis en place plusieurs sessions d'une formation intitulée « Introduction aux méthodes mathématiques et statistiques pour les modèles dynamiques en agronomie et pour l'élevage » (à Montpellier, Helsinki, Paris ou Toulouse de un à quatre jours). Cette formation se basait sur des illustrations concernant les productions végétales. La prochaine formation (à Rennes, en septembre 2013 sur quatre jours) vise le public des ingénieurs et chercheurs travaillant sur les productions animales et comportera essentiellement des illustrations sur l'élevage. Cela demande un investissement conséquent pour constituer ces nouvelles ressources pédagogiques et mon stage constitue une contribution significative à ce travail.

Le sujet de mon stage s'inscrit dans le cadre de la préparation et de l'organisation de

cette formation. L'objectif est donc d'élaborer des ressources pédagogiques centrées sur des modèles de production animale et d'élevage ainsi que les méthodes d'analyses statistiques s'appliquant sur ceux-ci. Pour commencer nous avons cherché des modèles pouvant servir d'exemple, puis je les ai implémentés sous le langage R. Ensuite j'ai construit des exemples d'application de méthodes statistiques sur ces modèles. Pour mon stage je me suis concentrée sur les méthodes d'évaluation et d'estimation bayésienne de paramètres. Enfin, nous avons rassemblé les fonctions des modèles, les données d'entrée et d'observation, les fonctions pour l'application des méthodes et les exemples d'application de méthodes dans un package R documenté pour en faciliter la diffusion et l'utilisation lors de la session de formation.

Par souci de clarté et de pédagogie, le rapport ne sera pas structuré selon le plan classique (Introduction, Matériel et méthodes, Résultats, Discussion), mais plutôt selon les points méthodologiques abordés lors de la formation. Dans ce rapport je vais donc présenter dans une première partie les deux modèles choisis (lactation et carcasse) ainsi que leur implémentation sous R. La deuxième partie contiendra un exemple d'évaluation sur le modèle de lactation. Dans la troisième partie on retrouvera l'application de méthodes bayésiennes pour estimer les paramètres du modèle de croissance de carcasse. Je présenterais ensuite rapidement ma contribution au package R créé. Enfin, dans une dernière partie de conclusion, je présenterai le travail restant à accomplir durant le dernier mois de stage, notamment concernant la finalisation des ressources pédagogiques créées. Je conclurai en présentant les enjeux que représente la création de telles ressources pédagogiques et l'utilisation qui en sera faite dans un avenir proche.

## 2 Modèles retenus

La première étape est de choisir des modèles qui pourraient servir d'illustration lors des formations. Pour ce faire, l'équipe a collaboré avec d'autres chercheurs travaillant ou ayant travaillé sur des modèles animaux et pouvant proposer certains de leurs modèles (Florence Garcia-Launay, Jacques Agabriel, Sophie Lemosquet). Ainsi, quatre modèles ont été proposés pour évaluer leur possible utilisation lors de la future formation. Un premier modèle, décrit par Neal et. al en 1983, traite de la lactation de bovins, il décrit le rendement de lait produit par une vache en fonction du lait trait. Le deuxième, étudié par Blanc en 1997, est une complexification du premier dans lequel est intégré l'interaction mère-jeune. Le troisième modèle simule la composition et la croissance du corps de bovins en fonction des conditions nutritionnelles, il a été décrit par Hoch et. al en 2004. Et enfin

le dernier modèle, étudié par Moset-Hernandez en 2006, représente les flux d'ammoniac dans une salle d'engraissement de porcs.

Le choix des modèles à conserver est guidé par d'une part la clarté et la simplicité du modèle, en effet, les formations étant de courtes durées (3-4 jours), le modèle doit être compris rapidement par les personnes en formation qui vont devoir le manipuler par la suite. D'autre part, le deuxième critère pour le choix du modèle est l'information disponible concernant le modèle, en effet les analyses qui devront être faites par la suite sur ce modèle nécessitent l'existence d'un nombre de données suffisant pour qu'elles aient un sens. De plus, ces données doivent être diffusables pour permettre aux utilisateurs de les réutiliser une fois la formation terminée. En prenant en compte ces deux conditions, deux modèles ont été retenus.

## 2.1 Modèle de lactation de Neal

Ce modèle est un modèle de lactation, i.e. de production de lait par les glandes mammaires de bovidés, décrit par Neal et al. (1983). Ce modèle a ensuite inspiré d'autres modèles plus complexes basés sur ces principes de base et il est bien connu par les zootechniciens.

Ce modèle simule la dynamique de la production de lait de vache (cf Figure 1). Il est représenté par 5 variables : la variation du taux d'hormones (H), la production et la perte de cellules sécrétrices de lait (CS), la sécrétion et le retrait de lait (M), la quantité moyenne de lait contenu dans l'animal (Mmoy) et la quantité de lait traité (RM). Ces variables dépendent du temps, le modèle a un pas de temps d'intégration de  $dt=0.1$  pour une consommation régulière de lait par un veau et de  $dt=0.001$  pour un tirage par pulse des machines de traite.

Le modèle est décrit par 11 équations (Annexe 1) et 19 paramètres (Annexe 2).

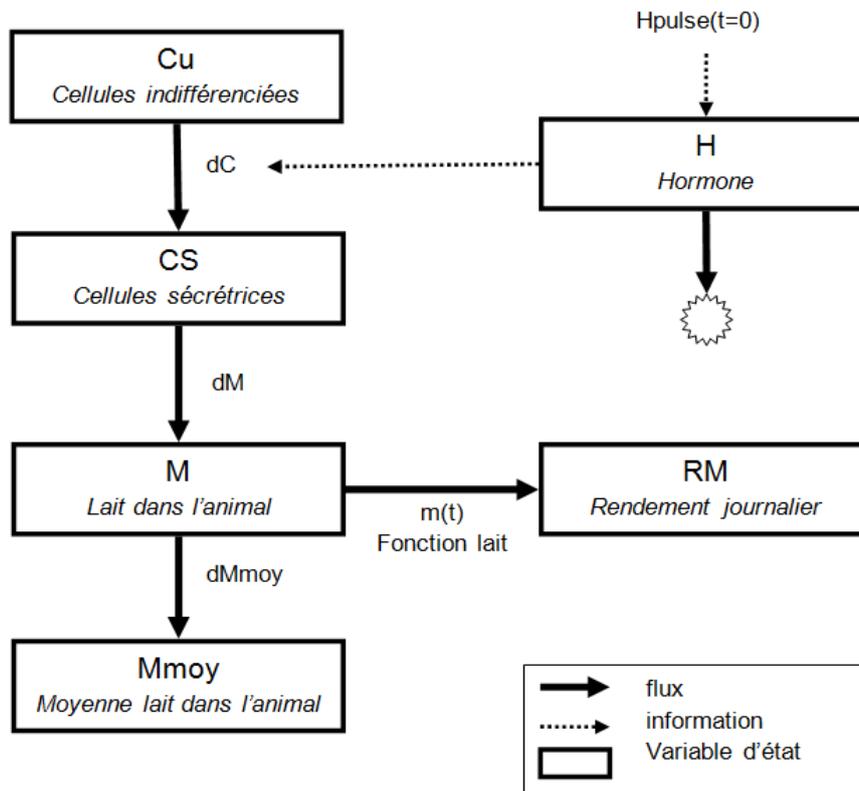


FIGURE 1 – Modèle conceptuel du modèle de lactation selon le formalisme de Forrester

## 2.2 Modèle de croissance de carcasse de Hoch

Ce modèle est décrit par Hoch et. al (2004), il représente la croissance de bovins et la composition relative du corps de différents types d'animaux selon les conditions nutritionnelles. Ce modèle simule la dynamique de l'évolution de la composition du corps en lipides et protéines en fonction de l'apport nutritif (cf Figure 2). Il prend en compte 4 variables : les protéines et les lipides dans la carcasse (resp. ProtC et LipC) et dans les autres tissus (resp. ProtNC et LipNC) regroupés sous le nom de 5ème quartier (foie, tube digestif, peau...). Un calcul sur ces 4 variables permet d'en obtenir une 5ème le poids vif (PV). Ces variables dépendent du temps, le pas de temps d'intégration utilisé est  $dt=1$  jour.

Ce modèle est décrit par 20 équations (Annexe 3) et 18 paramètres (Annexe 4).

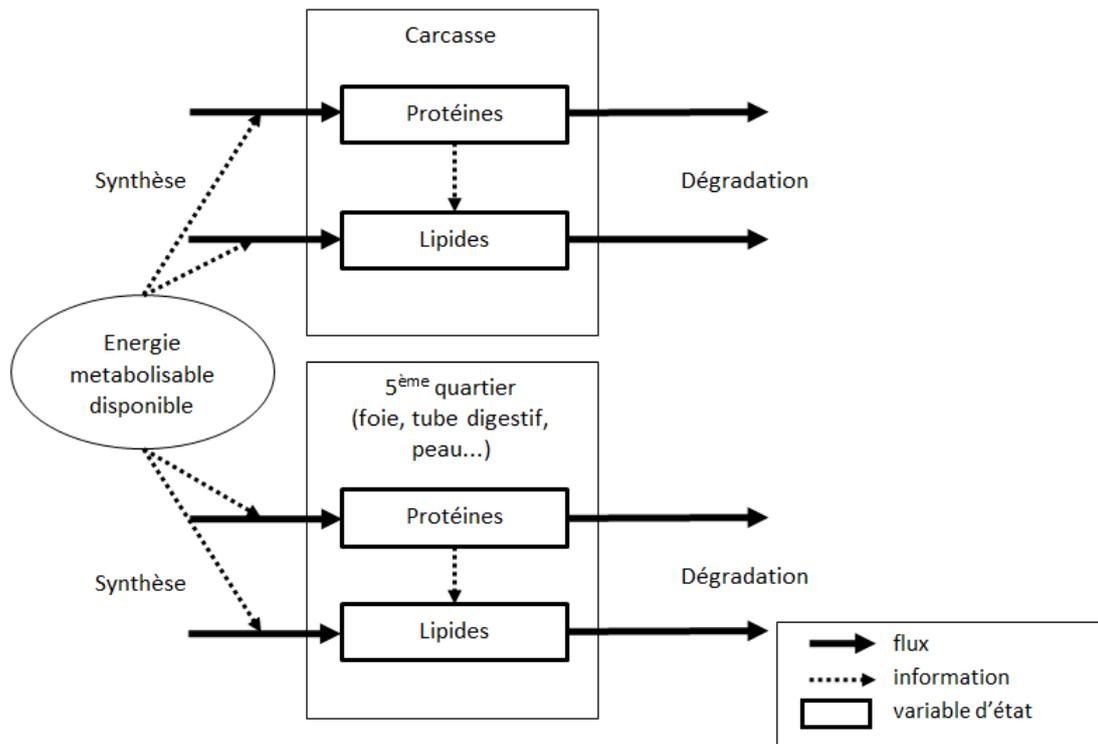


FIGURE 2 – Modèle de croissance de carcasse selon le formalisme de Forrester

## 2.3 Ecriture des simulateurs sous R

Une fois les modèles bien définis, les équations décrites et le fonctionnement compris, l'étape suivante est l'écriture du simulateur correspondant sous le langage R (Annexe 5). Typiquement, pour ces modèles l'algorithme de simulation comprend trois grandes étapes :

- Initialisation des variables d'état** : création des vecteurs qui contiendront les valeurs des variables d'état au cours du temps, puis indication de la valeur initiale pour chacune des variables.
- Boucle d'intégration** : calcul des taux de variation puis mise à jour des variables d'état.
- Renvoi des résultats** : renvoi des résultats sous forme d'un tableau.

## 3 Evaluation des performances des modèles

Le premier type d'analyse réalisé pour la formation est l'évaluation du modèle créé. Contrairement à l'analyse de sensibilité qui concerne uniquement le modèle et aide à mieux comprendre son fonctionnement, l'évaluation se concentre sur la capacité du modèle à représenter de manière fiable le système considéré. Ainsi elle consiste à comparer les résultats des simulations par le modèle à des données provenant d'observation réalisée lors d'expérimentations. Cette analyse va donc donner une idée de la similarité existante entre

des mesures réelles et celles que l'on peut obtenir grâce à une simulation.

L'évaluation du modèle est importante pour plusieurs raisons. D'abord, elle oblige l'utilisateur ou le concepteur du modèle à formaliser l'objectif du modèle, les conditions dans lesquelles le modèle sera utilisé et le niveau de qualité de prédiction acceptable. De plus, l'amélioration d'un modèle est impossible sans l'avoir évalué précédemment. En effet, l'évaluation permet d'avoir une idée de la qualité du modèle mais également de mieux comprendre les sources des erreurs. Enfin, l'évaluation du modèle est importante pour un utilisateur potentiel qui aura besoin d'informations sur la qualité du modèle pour savoir quel crédit accorder aux prédictions.

### 3.1 Critères d'évaluation

L'évaluation consiste en une comparaison des valeurs observées aux valeurs calculées grâce au modèle. Il existe plusieurs méthodes permettant cette comparaison, par des graphiques représentant les valeurs calculées en fonction des valeurs observées ou bien par des mesures numériques.

L'erreur quadratique moyenne MSE est probablement le critère quantitatif le plus utilisé pour comparer valeurs calculées et valeurs observées. Il mesure l'erreur comme la différence entre les valeurs observées et simulées au carré puis en fait une moyenne.

$$MSE = \left(\frac{1}{N}\right) \sum_{i=1}^N (Y_i - \hat{Y}_i)^2$$

avec  $Y_i$ =observation ,  $\hat{Y}_i$ =prédiction et  $N$ =nombre d'observations

Le RMSE représente la racine carré du MSE.

$$RMSE = \sqrt{MSE}$$

L'efficacité est un deuxième critère très utilisé pour mesurer l'accord entre valeurs mesurées et valeurs calculées et présente l'avantage d'être de même unité que la variable considérée.

$$EF = 1 - \frac{\sum_{i=1}^N (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^N (Y_i - \bar{Y})^2}$$

avec  $\bar{Y}$ =moyenne des observations

Si le modèle est parfait, les valeurs calculées égalent exactement les valeurs mesurées et  $EF=1$ . C'est la valeur maximale. Si les valeurs calculées égalent la moyenne des valeurs

mesurées, le numérateur sera égal au dénominateur et  $EF=0$ . C'est-à-dire, si un modèle n'explique pas plus de la variabilité des mesures que la moyenne des mesures, son efficacité est 0. L'efficacité peut être encore pire et prendre des valeurs négatives.

J'ai donc créé une fonction (Annexe 6) qui permet de calculer ces trois critères d'évaluation à partir des données observées ( $Y_{obs}$ ) et simulées ( $Y_{pred}$ ).

### 3.2 Exemple de résultat obtenu

L'évaluation a été appliquée sur plusieurs jeux de données correspondant au modèle de lactation. Ces données ont été récupérées dans des articles fournis par Sophie Lemosquet (personne ressource pour le modèle de lactation). J'ai pu ainsi récupérer quatre jeux de données différents (Pomiès 2002, 2007, 2010, Remond 2009). Je vais ici présenter les résultats obtenus pour un des jeux de données (Pomiès 2002).

Ce jeu de données représente les courbes de lactation de vaches de la race Holstein en fonction du temps en semaines. Il est composé de deux expérimentations :

-1T : Le lait est tiré une fois par jour à 6h30 par la machine de traite

-2T : Le lait est tiré deux fois par jour à 6h30 et 17h par la machine de traite

La première étape est de simuler ces deux expérimentations avec les conditions initiales fournies dans l'article. On obtient ainsi un tableau contenant les valeurs simulées ainsi que les valeurs observées (tableau simobs). On peut ensuite représenter ces valeurs sur des graphiques en fonction du temps pour les deux expérimentations (cf Figure 3).

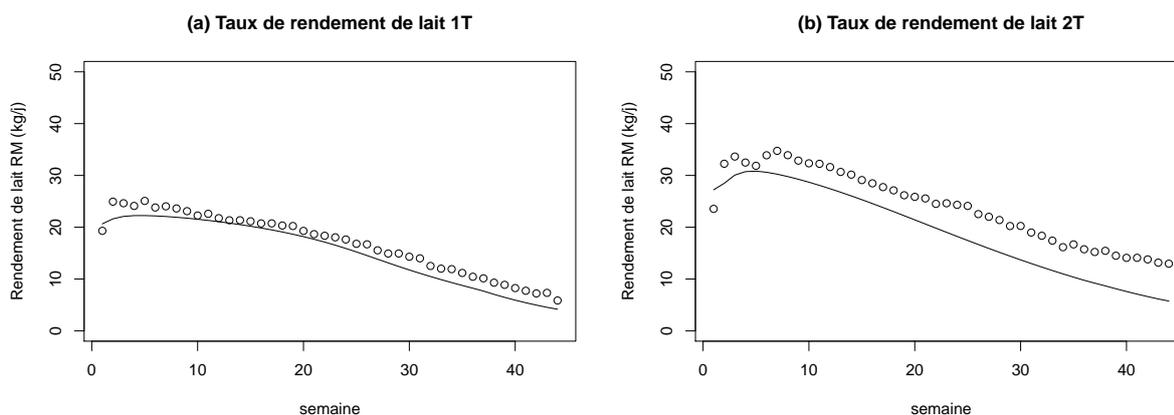


FIGURE 3 – Taux de lait trait en fonction du temps (a) 1T 1 tirage par jour (b) 2T 2 tirages par jour

On applique ensuite la fonction d'évaluation sur le tableau simobs et on obtient les valeurs des critères d'évaluation pour les deux expérimentations et pour l'ensemble des deux (cf Table 1).

	N	Mobs	Msim	MSE	RMSE	EF
T1	44	16.98061	15.31083	3.447226	1.856671	0.8946586
T2	44	24.0212	19.48789	24.25299	4.924733	0.5034926
tous	88	20.5009	17.39936	13.85011	3.721573	0.7395532

TABLE 1 – Valeurs des critères d’évaluation pour le modèle lactation

L’observation de la Figure 3 et de la Table 1 montre que le modèle fait une légère sous-estimation de la réalité, puisque les moyennes simulées sont légèrement inférieures à celles observées. Le RMSE est assez fort, il correspond à 10 à 20% des valeurs de moyennes. Enfin, l’efficacité est assez bonne sachant que la simulation n’a pas une variation strictement croissante ou décroissante mais commence par croître pour décroître par la suite.

Pour terminer, on trace le graphique des valeurs observées en fonction des valeurs simulées (cf Figure 4).

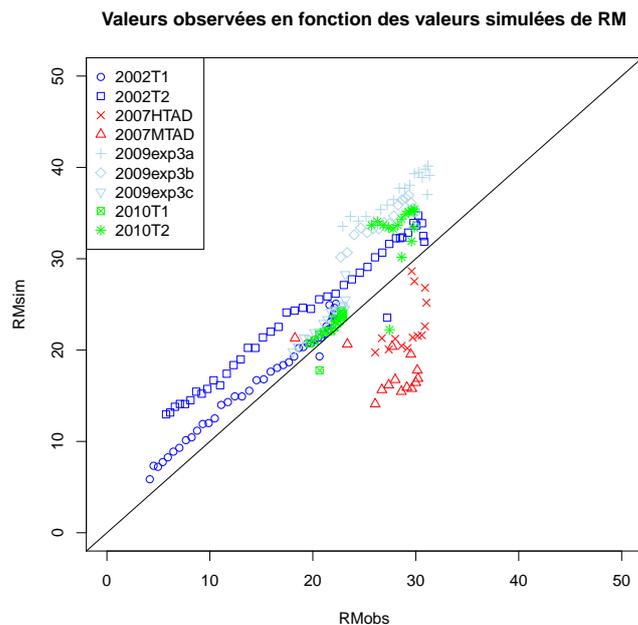


FIGURE 4 – Valeurs observées par le modèle de lactation en fonction des valeurs simulées de RM

## 4 Estimation bayésienne des paramètres

Pour améliorer un modèle on peut modifier sa structure (ajouter une équation) ou bien choisir des valeurs de paramètre en calibrant le modèle, i.e. en estimant le modèle. Il existe plusieurs techniques d’approche pour calculer la valeur de ces paramètres, notamment les approches fréquentiste et bayésienne.

L'approche bayésienne combine les informations de plusieurs sources pour estimer des valeurs de paramètres inconnues. Elle prend en compte les données mais également d'autres informations. Elle permet d'obtenir une probabilité de distribution des paramètres. Cette méthode se fait en deux étapes. La première étape est de définir une probabilité de distribution des paramètres basée sur la littérature. Cette distribution est appelée distribution *a priori* et reflète l'état des connaissances à propos des valeurs des paramètres. La seconde étape consiste à calculer une nouvelle probabilité de distribution grâce à la distribution antérieure et les données disponibles en utilisant le théorème de Bayes :

**Fonction de vraisemblance :** fonction reliant les données aux paramètres

**Distribution *a priori* des paramètres :** probabilité de distribution décrivant nos connaissances initiales des paramètres

$$P(\Theta|y) = \frac{P(y|\Theta) P(\Theta)}{\int_{\Theta} P(y|\Theta) P(\Theta)}$$

**Distribution *a posteriori* des paramètres :** probabilité de distribution des paramètres combinant les informations *a priori* et les données

L'utilisation de ce théorème est possible pour des exemples simples, en revanche lorsque les modèles se complexifient il est difficile voire impossible de résoudre ce théorème. Dans ce cas là on utilise des algorithmes permettant d'approximer la probabilité *a posteriori*. Il existe plusieurs méthodes pour approximer les paramètres par une approche bayésienne : les méthodes non adaptatives et les méthodes adaptatives comme celle de Monte Carlo par chaînes de Markov (MCMC).

## 4.1 Importance Sampling

### 4.1.1 Algorithme

Pour les méthodes non adaptatives, tous les vecteurs de paramètres peuvent être générés au début de la procédure. Le choix des paramètres à tester ne dépend pas des résultats pour les paramètres précédents. Un exemple d'algorithme pour cette méthode est *Importance Sampling* :

Etape 1 : Choisir un type de distribution de proposition pour le paramètre à estimer  $g(\theta)$ .

Pour simplifier, on considère que la distribution de proposition est égale à la distribution *a priori*  $P(\Theta)$  Etape 2 : Générer un échantillon de taille N à partir de  $g(\theta) = P(\Theta)$ .

Etape 3 : Calculer le poids  $w_i = \frac{P(y|\Theta_i) P(\Theta)}{g(\Theta_i)} = P(y|\Theta_i)$ .

Etape 4 : Calculer le poids normalisé  $w_i^* = \frac{w_i}{\sum_{i=1}^N (w_i)}$ .

Etape 5 : L'échantillon pesé  $(\Theta_1, w_1^*), (\Theta_2, w_2^*), \dots, (\Theta_N, w_N^*)$  fournit une approximation de la distribution postérieure. On peut ensuite faire un nouvel échantillonnage à partir de ces poids.

Dans cet algorithme le calcul du poids est en fait un calcul de vraisemblance entre le résultat du modèle et les valeurs observées. Plus le poids sera grand, plus les valeurs observées et simulées seront proches les unes des autres, i.e. plus la valeur du paramètre sélectionnée représentera le mieux la réalité à travers le modèle. L'ensemble de la fonction de vraisemblance et de la distribution *a priori* va nous fournir une approximation de la distribution *a posteriori* comme indiqué dans le théorème de Bayes présenté précédemment.

L'écriture de l'algorithme de Importance Sampling dans le langage R s'est inspiré du livre *Working with dynamic crop models*. Il se décompose ainsi (cf Annexe 9) :

- Échantillonnage des paramètres à estimer de taille N à partir de la distribution *a priori* (étapes 1 et 2)
- Fixation des autres paramètres
- Simulation du modèle avec les N paramètres (N itérations)
- Calcul de la vraisemblance/poids de PV pour chaque N paramètres (étape 3)
- Calcul du poids normalisé (étape 4)
- Nouvel échantillonnage en utilisant le poids normalisé (étape 5)

#### 4.1.2 Résultats obtenus

Cette méthode a été réalisée sur un jeu de données associé au modèle de croissance de carcasse. Ces données ont été fournies par Jacques Agabriel le chercheur avec qui je collabore pour ce modèle. Il est composé entre autres des données finales de protéines et lipides dans la carcasse (ProtC, LipC) et dans les autres tissus (ProtNC, LipNC) et de poids vif (PV) mais également du suivi au cours du temps du poids vif et des données d'entrée d'énergie (EMI). Jacques Agabriel nous a également fourni des informations

concernant tous les paramètres du modèle qui vont constituer les informations *a priori* (cf Annexe 8). Pour l'illustration de cette méthode, j'ai choisi de réaliser l'estimation sur 3 paramètres *protcmax*, *protncmax* et *amW* qui ont une distribution *a priori* normale (cf Table 2).

Paramètre	Distribution	BorneInf	BorneSup	Moyenne	EcartType	Remarque
<i>protcmax</i>	normale	104	126	115	11.5	Robelin 1986
<i>protncmax</i>	normale	50	60	55	5	Robelin 1986
<i>amW</i>	normale	1.2	1.8	1.5	0.3	Hoch 2004

TABLE 2 – Informations *a priori* concernant les paramètres estimés

Pour  $N=10000$  on obtient les résultats décrits dans la Table 3.

Paramètre	Moyenne	EcartType	Q1%	Q5%	Q25%	Q75%	Q95%	Q99%
<i>protcmax</i>	110.5	10.4	86.2	94.5	102.7	117.8	127.2	134.3
<i>protncmax</i>	53.6	4.9	41.9	45.6	50.5	56.9	62.1	64.3
<i>amW</i>	1.416	0.114	1.11	1.24	1.34	1.49	1.59	1.67

TABLE 3 – Moyennes, écarts types et quantiles des paramètres estimés pour  $N=10000$  par Importance Sampling

La Figure 5 compare la distribution *a priori* à celle obtenue *a posteriori*. Pour *protcmax* et *protncmax*, la distribution a peu varié, les données d'observation apportent donc peu d'information sur ces paramètres. En revanche, pour *amW*, la distribution s'est largement resserrée, les données ont donc une grande influence sur l'estimation de ce paramètre.

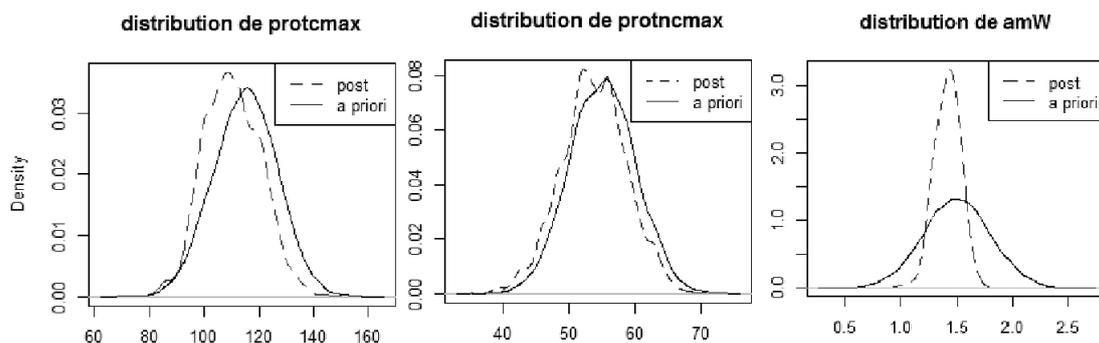


FIGURE 5 – Distribution des différents paramètres estimés selon l'algorithme Importance Sampling pour  $N=10000$

Une des questions que l'on peut se poser avec cette méthode est à partir de quel nombre de simulations la distribution *a posteriori* va se stabiliser. La Figure 6 représente les résultats obtenus en fonction du nombre d'itérations  $N$ . Les paramètres *protcmax* et *protncmax* ont un comportement similaire, ils se stabilisent assez rapidement (avant  $N=1000$ ) alors que le paramètre *amW* met un peu plus de temps à se stabiliser (après  $N=1000$ ).

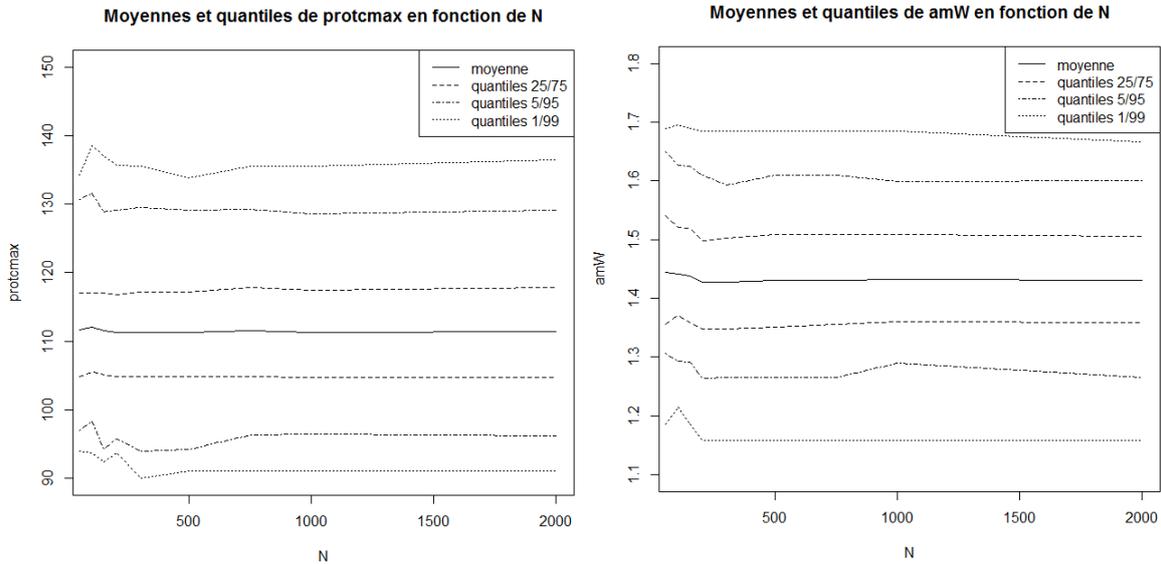


FIGURE 6 – Représentation des moyennes et quantiles des différents paramètres estimés par Importance Sampling en fonction de N

## 4.2 Metropolis Hastings

### 4.2.1 Algorithme

Pour les méthodes adaptatives telles que la méthode MCMC, les valeurs des paramètres sont générées par une chaîne de Markov. La valeur du paramètre à tester pour l'étape  $i+1$  peut dépendre de la valeur du paramètre pour l'étape  $i$ . L'exemple d'algorithme le plus connu pour cette méthode et l'algorithme de *Metropolis-Hastings* :

Etape 1 : Choisir une valeur initiale  $\Theta_1$ . Définir une distribution de proposition  $P_P(\Theta_c|\Theta_i)$ . On choisit de manière à simplifier les calculs une distribution normale centrée sur la valeur initiale avec comme variance celle donnée dans la distribution *a priori* à laquelle on ajoute un coefficient. Celui-ci appelé *tune* permet d'ajuster la taille et la direction de la distribution proposée afin d'obtenir un taux de rejet proche de 0.76 (Robert and Casella, 2004).

Pour N itérations :

Etape 2 : Générer une valeur du paramètre candidat  $\Theta_c$  à partir de  $P_P(\Theta_c|\Theta_i)$ .

Etape 3 : Calculer  $T = \frac{P(Y|\Theta_c) P(\Theta_c) P_P(\Theta_i|\Theta_c)}{P(Y|\Theta_i) P(\Theta_i) P_P(\Theta_c|\Theta_i)}$

Etape 4 : Si  $T \geq 1$ , alors  $\Theta_{i+1} = \Theta_c$ . Si  $T < 1$ , alors tracer  $u$  à partir d'une distribution uniforme dans l'intervalle  $(0,1)$ . Si  $T < u$ , alors  $\Theta_{i+1} = \Theta_c$  sinon  $\Theta_{i+1} = \Theta_i$ .

La valeur candidate va servir de point de départ à cet algorithme. Puis à chaque itération, on va comparer la nouvelle valeur candidate à la précédente. Si celle-ci permet d'obtenir une meilleure vraisemblance pour le modèle elle sera conservée, sinon la valeur précédente sera gardée. On réitère cette opération jusqu'à ce que le résultat devienne le plus stable possible. Le résultat de cet algorithme est donc une liste de N valeurs de paramètres. La même valeur peut être répétée plusieurs fois.

Il a été prouvé que pour un nombre suffisant d'itérations le résultat de cet algorithme convergeait vers la distribution *a posteriori*. Pour vérifier cette convergence on peut utiliser deux chaînes avec deux points de départ différents.

L'écriture de l'algorithme de Metropolis Hastings dans le langage R s'est inspiré du livre *Working with dynamic crop models*. Il se décompose ainsi (cf Annexe 11) :

- Création d'une fonction permettant de faire tourner le modèle pour chaque individu pour éviter d'alourdir le script
  - Création d'un tableau contenant tous les paramètres
  - Calcul de tune
  - Choix d'une valeur initiale et d'une distribution pour le paramètre (étape 1)
- Pour N itérations :
- Génération de la valeur candidate (étape 2)
  - Simulation avec la valeur candidate
  - Calcul des vraisemblances
  - Calcul du test (étape 3)
  - Sélection ou rejet du candidat (étape 4)

Pour un paramètre le taux de rejet va peu varier, il est donc facile d'estimer une valeur de tune et de la conserver pour toutes les itérations. En revanche, lorsque que l'on cherche à estimer plusieurs paramètres il va falloir changer régulièrement la valeur de tune pour que le taux de rejet reste acceptable. Pour ce faire il faudra créer un algorithme adaptatif qui changera la valeur de tune lorsque le taux de rejet ne sera pas compris entre 70 et 80%. Ceci fait partie des choses qu'il me reste à faire dans la suite de mon stage.

## 4.2.2 Résultats obtenus

Pour cette méthode je ne vais donc estimer qu'un paramètre pour le moment. J'ai choisi de travailler sur  $amW$  qui semble avoir une influence plus importante sur le modèle. Les informations *a priori* sur ce paramètre ainsi que les données d'entrée et de sortie sont les mêmes que précédemment.

Pour  $N=10000$  on obtient les résultats décrits dans la Table 4.

Paramètre	Moyenne	EcartType	Q1%	Q5%	Q25%	Q75%	Q95%	Q99%
$amW$	1.36394	0.023225	1.31484	1.32604	1.34771	1.38029	1.40438	1.41629

TABLE 4 – Moyennes, écarts types et quantiles de  $amW$  estimés pour  $N=10000$  par Metropolis Hastings

La Figure 7 représente les distributions *a priori* et *a posteriori* de  $amW$ . Ici encore la distribution s'est fortement resserrée, les données d'observation ont donc une grande influence sur l'estimation du paramètre.

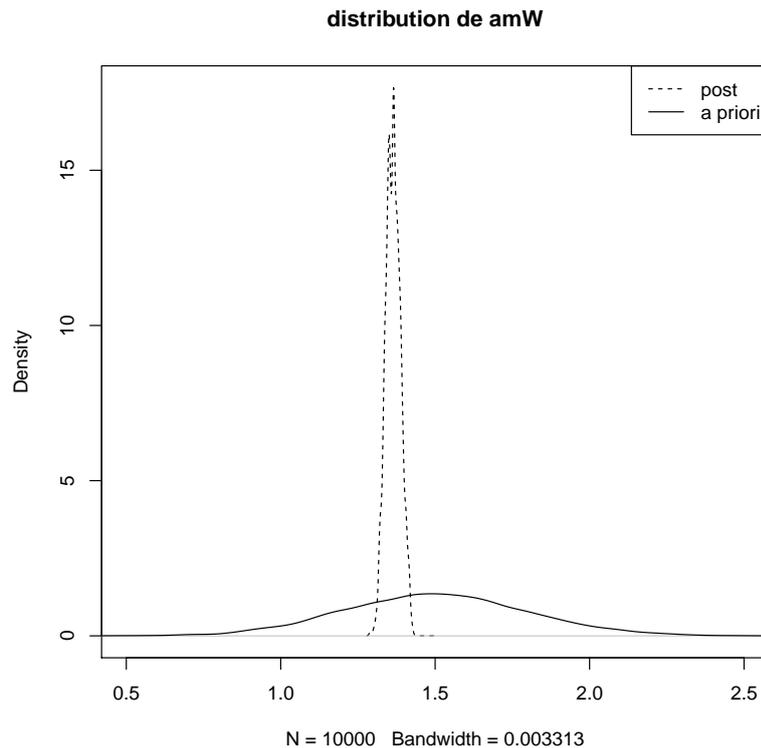


FIGURE 7 – Distribution de  $amW$  selon l'algorithme Metropolis Hastings pour  $N=10000$

Ensuite, comme précédemment on peut voir l'évolution des moyennes et quantiles en fonction de  $N$  sur la Figure 8.

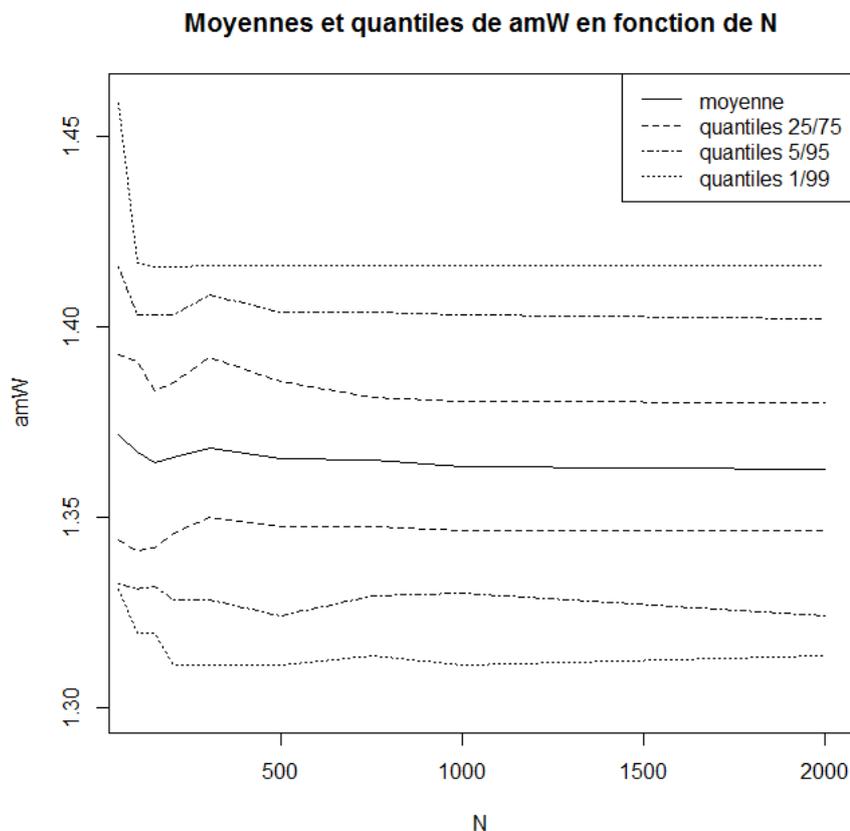


FIGURE 8 – Représentation des moyennes et quantiles de amW estimé par Metropolis Hastings en fonction de  $N$

Enfin, pour vérifier que l'estimation converge bien vers un point bien défini, on change le point de départ  $\Theta_1$ . La Figure 9 représente l'évolution du paramètre amW au cours du temps selon deux points de départ différents, on voit bien que la convergence se fait assez rapidement.

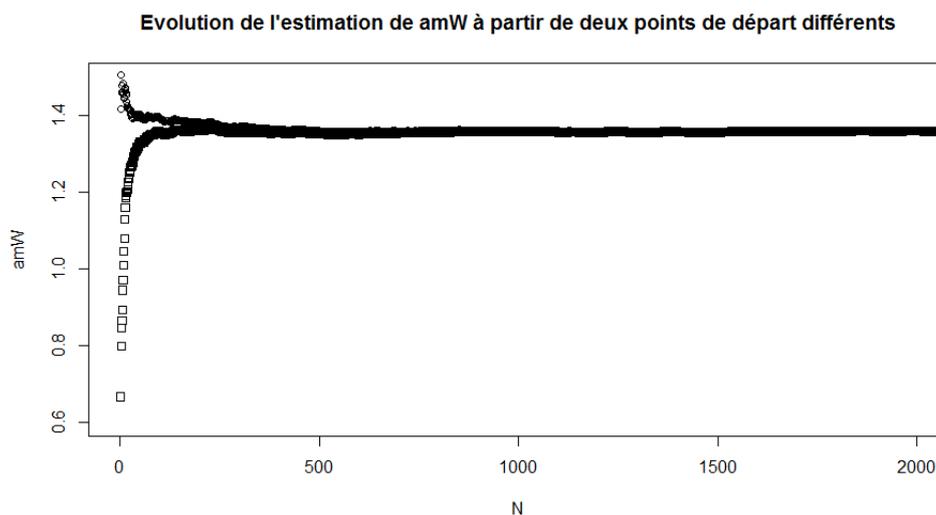


FIGURE 9 – Evolution de amW selon deux points de départ différents

## 5 Contribution au package « ZeBook »

Le package ZeBook (<http://cran.r-project.org/web/packages/ZeBook/>) plus large et déjà existant comprend différentes fonctions correspondant aux différents modèles sélectionnés comme exemples. Elles sont codées sous la forme de scripts R documentés au format R-Oxygen. Ces fonctions permettent de réaliser des simulations en utilisant différents jeux de données qui sont inclus également dans le package ou encore en modifiant facilement les valeurs de certains paramètres.

Le package contient également de nombreuses fonctions permettant de mettre en oeuvre différentes techniques statistiques appliquées aux modèles. On y retrouve plusieurs techniques ainsi que des exemples de leur fonctionnement sous R et des résultats pouvant être obtenus. Ces ressources sont en anglais.

Le tableau suivant présente une partie non exhaustive de ma contribution au package ZeBook.

Type	Nom	Description
Fonction	<code>lactation.calf.model</code>	fonction du modèle de lactation
	<code>lactation.calf.model2</code>	fonction du modèle de lactation adaptée aux simulations
	<code>lactation.calf.simule</code>	simulation du modèle lactation
	<code>lactation.define.param</code>	définit les valeurs des paramètres du modèle de lactation
	<code>lactation.machine.model</code>	modèle de lactation avec la fonction de machine de traite
	<code>lactation.machine.model2</code>	fonction du modèle de lactation adaptée aux simulations
Fonction	<code>carcass.model</code>	fonction du modèle carcasse
	<code>carcass.EMI.model2</code>	fonction du modèle carcasse adapté aux simulations
	<code>carcass.EMI.simule</code>	simulation du modèle carcasse
	<code>carcass.EMI.multi</code>	simulation du modèle carcasse pour plusieurs animaux
	<code>carcass.define.param</code>	définit les valeurs du paramètre du modèle carcasse
Fonction	Evaluation	calcul des critères d'évaluation
Données	<code>carcass-data</code>	données pour le modèle carcasse

## 6 Conclusions et perspectives

Le travail réalisé durant ce stage a permis d'atteindre les objectifs fixés en début de stage, c'est à dire la création de ressources pédagogiques incluses dans un package R qui servira de support lors de formations. Ces différentes ressources sont notamment basées sur des fonctions permettant de réaliser facilement et en grand nombre des simulations de différents modèles agronomiques, de comparer aisément les résultats à des données observées. Ils permettent également par le biais de fonctions simples d'accès d'utiliser différentes techniques d'analyse de sensibilité, d'estimation de paramètres et d'évaluation de modèle. Il est à noter que pour d'autres techniques concernant l'estimation de paramètres par des méthodes Fréquentistes, l'analyse d'incertitude ou encore l'analyse de sensibilité ont été réalisés par mes co-encadrants.

Le travail que j'ai réalisé m'a permis de me perfectionner dans l'utilisation de R comme langage de programmation en m'introduisant le principe de la documentation R-Oxygen. Il m'a également permis de maîtriser des outils de suivi de version (SVN) ainsi que la récupération de données sous R. Le travail n'est toutefois pas terminé, il reste de nombreuses choses à accomplir avant la mise à disposition du package R, certaines méthodes restent à y être incluses et certaines documentations sont encore incomplètes.

## 7 Bibliographie

BLANC F. (1997-1998). Modélisation de la dynamique d'allaitement en système d'élevage bovin allaitant. Etude de la relation entre production laitière de la mère et croissance du jeune. D.E.A. Modèles et Instruments en Médecine et Biologie.

DANFAER A. (1991). Mathematical modeling of metabolic regulation and growth. *Livest. Prod. Sci.* 27,1-18.

FRANCE J. et THORNLEY J. H. M. (1984). *Mathematical Models in Agriculture : A quantitative approach to problems in Agriculture and related sciences.* Butterworth et Co (Publishers) Ltd 1984.

GARCIA F., SAINZ R. D., AGABRIEL J., BARIONI L. G. et OLTJEN J. W. (2008). Comparative analysis of two dynamic mechanistic models of beef cattle growth. *Animal Feed Science and Technology* 143, 220-241.

GOUACHE D., BENSADOUN A., BRUN F., PAGE C., MAKOWSKI D., WALLAC D. (2012). Modeling climate change impact on *Septoria tritici* blotch (STB) in France : Accounting for climate model and disease model uncertainty. *Agricultural and Forest Meteorology*, Volume 170, Pages 242-252.

HOCH T. et AGABRIEL J. (2004a). A mechanistic dynamic model to estimate beef cattle growth and body composition : 1. Model description. *Agricultural Systems* 81 1-15.

HOCH T. et AGABRIEL J. (2004b). A mechanistic dynamic model to estimate beef cattle growth and body composition : 2. Model evaluation. *Agricultural Systems* 81 17-35.

HOCH T., PRADEL P. et AGABRIEL J. (2004). Modélisation de la croissance de bovins : évolution des modèles et applications. *INRA Prod. Anim.*, 17 (4), 303-314.

HOCH T., PRADEL P. et CHAMPCIAUX P. (2006). *Simulating Chemical and Tissue Composition of Growing Beef Cattle : From the Model to the Tool.* CABI, *Nutrient Digestion and Utilization in Farm Animals : Modeling approaches.*

KORVER S. et VAN ARENDONK J. A. M. (1988). *Modeling of Livestock Production Systems.* Kluwer Academic Publishers.

MARTIN O. (2009). Modélisation systémique des performances des femelles de ruminants : Application au cas de la vache laitière. Thèse à l'institut des Sciences et Industries du vivant et de l'Environnement (Agro Paris Tech).

MICOL D., DUBROEUCQ H., MARTIN C., GARCIA F., MIALON M.M. et AGABRIEL J. (2007). Utilization of contrasting diets in Aquitaine Blond young bull production. *Renc. Rech. Ruminants*, 2007, 14.

MOSET-HERNANDEZ V. (2006). Modélisation dynamique du flux d'ammoniac dans une salle d'engraissement de porcs et prédiction de la teneur dans l'ambiance. Mémoire de fin d'études diplôme d'agronomie approfondie (DAA) Spécialité Ingénierie Zootechnique.

- NEAL H. D. S. C. et THORNLEY J. H. M. (1983). The lactation curve in cattle : a mathematical model of the mammary gland. *J. Agric. Sci., Camb.*, 101, 389-400.
- POMIÈS D. et RÉMOND B. (2002). Once a day milking of dairy cows during the whole lactation : consequences and performances and milk quality. *Renc. Rech. Ruminants*, 2002, 9.
- POMIÈS D., MARTIN B., CHILLIARD Y., PRADEL P. et RÉMOND B. (2007). Once a day milking of Holstein and Montbéliarde cows for 7 weeks in mid-lactation. *Animal* (2007), 1 :10, pp 1497–1505.
- POMIÈS D., CARÉ S. et VEISSIER I. (2010). Once daily milking combined with suckling in Holstein cows. *Renc. Rech. Ruminants*, 2010, 17.
- RÉMOND B., POMIÈS D., JULIEN C. et GUINARD-FLAMENT J. (2009). Performance of dairy cows milked twice daily at contrasting intervals. *Animal* (2009), 3 :10, pp 1463–1471.
- ROBELIN. (1986). Bases physiologiques de la production de viande : croissance et développement des bovins. In : Micol D. (Ed.), *Production de viande bovine*. INRA Publication, Paris.
- ROBERT C.P. et CASELLA G. (2004). *Monte Carlo Statistical Methods*, second ed. Springer, New York.
- THORNLEY J. H. M. (1976). *Mathematical models in plant physiology : a quantitative approach to problems in plant and crop physiology*. Academic press, a subsidiary of Harcourt Brace Jovanovich, Publishers.
- WALLACH D., MAKOWSKI D. et JONES J. W. (2006). *Working with dynamic crop models*. Elsevier Science Editions.

## 8 Annexes

### Annexe 1 : Equations du modèle de lactation

$$(1) H(t + dt) = H(t) + dH$$

$$(2) CS(t + dt) = CS(t) + dCS$$

$$(3) M(t + dt) = M(t) + dM$$

$$(4) Mmoy(t + dt) = Mmoy(t) + dMmoy$$

$$(5) RM(t) = \frac{M(t)}{kdl + M(t)} \times m(t)$$

avec

$$(6) dH = -kdh \times H(t) dt$$

$$(7) dCS = \left( mum \times \frac{H(t)}{kdiv + H(t)} \times cu - \left( ks + ksm \times \frac{(Mmoy(t)/mh)^p}{(1 + Mmoy(t)/mh)^p} \right) \times CS(t) \right) dt$$

$$(8) dM = \left( km \times CS(t) \times \frac{mm - M(t)}{mm - M(t) + ksl} - \frac{M(t)}{kdl + M(t)} \times m(t) \right) dt$$

$$(9) dMmoy = kr \times (M(t) - Mmoy(t)) dt$$

$$(10) m(t) = rc \text{ pour une consommation régulière par le veau}$$

$$(11) m(t) = rma \times (PI1 + PI2) \text{ pour un tirage par pulse des machines}$$

avec

$$PI1 = 1 \text{ si } t1 < t \leq t2 \text{ premier tirage}$$

$$PI1 = 0 \text{ sinon}$$

et

$$PI2 = 0 \text{ si } t3 < t \leq t4 \text{ deuxième tirage}$$

$$PI2 = 1 \text{ sinon}$$

## Annexe 2 : Paramètres du modèle de lactation

Nom	Valeur	Unité	Description	Equation
cu	1000		Nombre de cellules indifférenciées	7
kdiv	0.2	kg hormone / $m^3$	Taux de division des cellules	7
kdl	5.0	kg	Constante de dégradation du lait	5,7
kdh	0.01	/jour	Taux de décomposition de l'hormone	6
km	0.00506	kg/cel/j	Constante de sécrétion de lait	8
ksl	3.0	kg	Taux de sécrétion de lait	8
kr	0.048	/jour	Constante moyenne du lait	9
ks	0.1	/jour	Taux de dégradation des cellules basales	7
ksm	0.2	/jour	Taux de dégradation des cellules sécrétrices	7
mh	27.0	kg	Paramètre de l'équation	7
mm	30.0	kg	Capacité de stockage de lait	8
p	10	kg	Paramètre de l'équation	7
mum	1.0	div/cel/j	Paramètre du taux max de division des cellules	7
rc	40	kg/j	Paramètre de la fonction lait	10
rma	4000			11
t1	0.250	jour	Paramètre de la fonction lait	11
t2	0.255			
t3	0.750			
t4	0.755			

### Annexe 3 : Equations du modèle de croissance de carcasse

#### Carcasse : Protéines

$$(1) ProtC(t+1) = ProtC(t) + (ProtCSyn(t) - ProtCDeg(t))$$

Synthèse :

$$(2) ProtCSyn(t) = alphac \times ProtC(t) \times \ln\left(\frac{protcmax}{ProtC(t)}\right) \times \left(\frac{EMI(t)}{CPM(t)+EMI(t)}\right)$$

$$(3) EMI(t) = cem \times (0.0157 \times PV(t)^{0.9} + 3.3161)$$

$$(4) CPM(t) = k \times PV(t)^{0.75}$$

Dégradation :

$$(5) ProtCDeg(t) = gammac \times ProtC(t) \times \ln\left(\frac{protcmax}{ProtC(t)}\right)$$

#### Carcasse : Lipides

$$(6) LipC(t+1) = LipC(t) + (LipCSyn(t) - LipCDeg(t))$$

Synthèse :

$$(7) LipCSyn(t) = betac \times LipC(t) \times \ln\left(\frac{LipCmax(t)}{LipC(t)}\right) \times \left(\frac{EMI(t)}{CPM(t)+EMI(t)}\right)$$

Dégradation :

$$(8) LipCDeg(t) = delta \times LipC(t) \times \ln\left(\frac{LipCmax(t)}{LipC(t)}\right)$$

$$(9) LipCmax(t) = lip0 + lipc1 \times \left(\frac{ProtC(t)}{protcmax}\right)$$

#### 5ème quartier : Protéines

$$(10) ProtNC(t+1) = ProtNC(t) + ProtNCSyn(t) - ProtNCDeg(t)$$

Synthèse :

$$(11) ProtNCSyn(t) = alphanc \times ProtNC(t) \times \ln\left(\frac{protncmax}{ProtNC(t)}\right) \times \frac{EMI(t)}{CPM(t)+EMI(t)}$$

Dégradation :

$$(12) ProtNCDeg(t) = gammanc \times ProtNC(t) \times \ln\left(\frac{protncmax}{ProtNC(t)}\right)$$

#### 5ème quartier : Lipides

$$(13) LipNC(t+1) = LipNC(t) + LipNCSyn(t) - LipNCDeg(t)$$

Synthèse :

$$(14) LipNCSyn(t) = beta \times LipNC(t) \times \ln\left(\frac{LipNCmax(t)}{LipNC(t)}\right)$$

Dégradation :

$$(15) LipNCDeg(t) = delta \times LipNC(t) \times \ln\left(\frac{LipNCmax(t)}{LipNC(t)}\right)$$

$$(16) LipNCmax(t) = lip0 + lipnc1 \times \left(\frac{ProtNC(t)}{protncmax}\right) \times PoidsNC(t)$$

$$\text{Poids vif : } (17) PV(t) = c0 \times PVV(t)^{c1}$$

$$\text{Poids vif vide : } (18) PVV(t) = PoidsC(t) + PoidsNC(t)$$

$$(19) PoidsC(t) = LipC(t) + b0c \times ProtC(t)^{b1c}$$

$$(20) PoidsNC(t) = LipNC(t) + b0nc \times ProtNC(t)^{b1nc}$$

#### Annexe 4 : Paramètres du modèle de croissance de carcasse

Nom	Valeur	Unité	Description	Equation
protcmax	131	kg	Quantité de protéines dans la carcasse adulte	2,5
protnemax	57	kg	Quantité de protéines dans le 5ème quartier de l'adulte	11,12
alphac	0.016	$j^{-1}$	Taux de synthèse protéique max dans la carcasse	2
alphanc	0.032	$j^{-1}$	Taux de synthèse protéique max dans le 5ème quartier	11
gammac	0.0025	$j^{-1}$	Taux de dégradation protéique max dans la carcasse	5
gammanc	0.009	$j^{-1}$	Taux de dégradation protéique max dans le 5ème quartier	12
lip0	0.12	%	Concentration lipidique max	9,16
lipc1	0.13	%	Coefficient d'augmentation de la concentration lipidique max dans la carcasse	9
lipnc1	0.13	%	Coefficient d'augmentation de la concentration lipidique max dans le 5ème quartier	16
beta	0.1	$j^{-1}$	Taux de synthèse lipidique	7,14
delta	0.025	$j^{-1}$	Taux de dégradation lipidique	8,15
k	1.5	$MJ.kg^{0.75}$	Paramètre de l'équation	4
b0c	5.665		Coefficient reliant masse délipidée et protéines de la carcasse	19
b1c	0.949		Exposant reliant masse délipidée et protéines de la carcasse	19
b0nc	7.224		Coefficient reliant masse délipidée et protéines du 5ème quartier	20
b1nc	0.85		Exposant reliant masse délipidée et protéines du 5ème quartier	20
c0	1.3022		Coefficient reliant poids vif et poids vif vide	
c1	0.9766		Exposant reliant poids vif et poids vif vide	

## Annexe 5 : Modèle de lactation sous R

```

lactation.calf.model <- fonction(cu, kdiv, kdl, kdh, km, ksl, kr, ks, ksm, mh, mm, p, mum, rc, duration
, dt) {
# Initialisation des variables
H=rep(NA,(duration-1)/dt)
CS=rep(NA,(duration-1)/dt)
M=rep(NA,(duration-1)/dt)
Mmoy=rep(NA,(duration-1)/dt)
RM=rep(NA,(duration-1)/dt)
# Initialisation des variables d'etat
H[1]=1.0
CS[1]=520
M[1]=0.0
Mmoy[1]=0.0
i=1
# Boucle d'integration
for (t in seq(0, duration, by = dt)) {
# Calcul des taux de variation
dH = - kdh * H[i] * dt
dCS = (mum * (H[i]/(kdiv+H[i]))*cu - (ks + ksm*((Mmoy[i]/mh)^p/(1+(Mmoy[i]/mh)^p)))*
CS[i]) * dt
dM = (km * CS[i] * ((mm-M[i])/(mm-M[i]+ksl))-(M[i]/(kdl+M[i]))*rc) * dt
dMmoy = kr*(M[i]-Mmoy[i]) * dt
# Mise a jour des variables d'etat
H[i+1]= H[i] +dH
CS[i+1]= CS[i] + dCS
M[i+1]= M[i] + dM
Mmoy[i+1]= Mmoy[i] + dMmoy
RM[i]=(M[i]/(kdl+M[i]))*rc
i=i+1 }
day=seq(dt, duration, by=dt)
week=day%/%7
results1=data.frame(M=M[1:(duration/dt)],Mmoy=Mmoy[1:(duration/dt)],CS=CS[1:(duration/dt)]
, RM=RM[1:(duration/dt)], day=day, week=week)
# Renvoi des resultats
result = by(results1[,c("week", "M", "Mmoy", "CS", "RM")], results1$week, function(x) apply(x
, 2, mean))
results2 = matrix(unlist(result), ncol=5, byrow=TRUE, dimnames=list(NULL, c("week", "M", "
Mmoy", "CS", "RM")))
return(results2)

```

## Annexe 6 : Fonction évaluation sous R

```
evaluation=function(Ypred, Yobs)
{
  select=! is.na(Ypred)&! is.na(Yobs)
  Ypred=Ypred[select]
  Yobs=Yobs[select]
  N<-length(Yobs)
  mean.Yobs<-mean(Yobs)
  mean.Ypred<-mean(Ypred)
  SSE<- sum((Yobs-Ypred)^2)
  MSE<- SSE/N
  RMSE=MSE^0.5
  EF<- 1 - sum((Yobs-Ypred)^2)/sum((Yobs-mean.Yobs)^2)
  return(data.frame(N=N, mean.Yobs=mean.Yobs, mean.Ypred=mean.Ypred, MSE=MSE, RMSE=RMSE, EF=EF
  ))
}
```

## Annexe 7 : Présentation des données du modèle de croissance de carcasse

### Article 2002, Pomiès

Dans cet article sont observées les courbes de lactation par semaine de vaches de la race Holstein pour des tirages une fois par jour (à 6h30) ou deux fois par jour (à 6h30 et à 17h).

### Article 2007, Pomiès

Dans cet article sont observées les courbes de lactation par semaine de vaches des races Holstein et Montbeliarde pour des tirages deux fois par jour (à 6h30 et 16h).

### Article 2009, Rémond

Dans cet article sont observées les courbes de lactation par semaine de vaches de la race Holstein pour des tirages une fois par jour (à 6h) ou deux fois par jour (à 6h et 16h ou à 6h et 11h).

### Article 2010, Pomiès

Dans cet article sont observées les courbes de lactation par semaine de vaches de la race Holstein pour des tirages une fois par jour (à 6h) ou deux fois par jour (à 6h et 18h).

Ces données seront disponibles dans le package ZeBook.

## Annexe 8 : Informations sur les paramètres du modèle de croissance de carcasse

Paramètre	Distribution	BorneInf	BorneSup	Moyenne	EcartType	Remarque
protcmax	normale	104	126	115	11.5	Robelin 1986
protnemax	normale	50	60	55	5	Robelin 1986
amW	normale	1.2	1.8	1.5	0.3	Hoch 2004
alphac	uniforme	0.015	0.016	0.0155	0.0005	Agabriel
alphanc	uniforme	0.031	0.032	0.0315	0.0005	Agabriel
lip0	uniforme	0.11	0.13	0.12	0.01	Hoch 2004
lipc1	uniforme	0.12	0.14	0.13	0.01	Hoch 2004
lipnc1	uniforme	0.12	0.14	0.13	0.01	Hoch 2004
beta	uniforme	0.09	0.11	0.1	0.01	Danfaer 1991
delta	uniforme	0.0225	0.0275	0.025	0.0025	Danfaer 1991
b0c	constante	NA	NA	5.665	NA	Robelin 1986
b1c	constante	NA	NA	0.949	NA	Robelin 1986
b0nc	constante	NA	NA	7.224	NA	Robelin 1986
b1nc	constante	NA	NA	0.85	NA	Robelin 1986
c0	constante	NA	NA	1.3022	NA	Garcia 2008
c1	constante	NA	NA	0.9766	NA	Garcia 2008
gammac	constante	NA	NA	0.0025	NA	Agabriel
gammanc	constante	NA	NA	0.009	NA	Agabriel

## Annexe 9 : Script de l'algorithme Importance Sampling

```
# Chargement des librairies
library(ZeBook)
library(sensitivity)
source("models/model.carcass.EMI.r")
# Lecture des donnees a partir de la library ZeBook
list_individuals=carcass_data$list_individuals[c(1,12,24),] #[1:5,]
energy=carcass_data$energy
init_condition=carcass_data$init_condition
observation_dynamic=carcass_data$observation_dynamic
observation_slaughter=carcass_data$observation_slaughter
observation_dynamic$time=observation_dynamic$time*7-6
distribution=read.csv("description/parametre_carcasse.csv",dec="," ,h=T,sep=";")
#Echantillonnage des parametres a estimer de taille N a partir de la distribution a priori
#Lecture des moyennes et ecarts types a partir du fichier
m.protcmax=subset(distribution$moyenne,distribution$parametre=="protcmax")
s.protcmax=subset(distribution$ecarttype,distribution$parametre=="protcmax")
m.protnemax=subset(distribution$moyenne,distribution$parametre=="protnemax")
s.protnemax=subset(distribution$ecarttype,distribution$parametre=="protnemax")
m.amW=subset(distribution$moyenne,distribution$parametre=="amW")
s.amW=subset(distribution$ecarttype,distribution$parametre=="amW")
#Choix de la taille N
N=10000
#Etapes 1 et 2 : Echantillonnage
v.protcmax<-rnorm(N,mean=m.protcmax,sd=s.protcmax)
v.protnemax<-rnorm(N,mean=m.protnemax,sd=s.protnemax)
```

```

v.amW<-rnorm(N,mean=m.amW,sd=s.amW)
# Fixation des autres parametres
v.alphac<-rep(subset(distribution$valeur,distribution$parametre=="alphac"),N)
v.alphanc<-rep(subset(distribution$valeur,distribution$parametre=="alphanc"),N)
v.lip0<-rep(subset(distribution$valeur,distribution$parametre=="lip0"),N)
v.lipc1<-rep(subset(distribution$valeur,distribution$parametre=="lipc1"),N)
v.lipnc1<-rep(subset(distribution$valeur,distribution$parametre=="lipnc1"),N)
v.beta<-rep(subset(distribution$valeur,distribution$parametre=="beta"),N)
v.delta<-rep(subset(distribution$valeur,distribution$parametre=="delta"),N)
v.b0c<-rep(subset(distribution$valeur,distribution$parametre=="b0c"),N)
v.b1c<-rep(subset(distribution$valeur,distribution$parametre=="b1c"),N)
v.b0nc<-rep(subset(distribution$valeur,distribution$parametre=="b0nc"),N)
v.b1nc<-rep(subset(distribution$valeur,distribution$parametre=="b1nc"),N)
v.c0<-rep(subset(distribution$valeur,distribution$parametre=="c0"),N)
v.c1<-rep(subset(distribution$valeur,distribution$parametre=="c1"),N)
v.gammac<-rep(subset(distribution$valeur,distribution$parametre=="gammac"),N)
v.gammanc<-rep(subset(distribution$valeur,distribution$parametre=="gammanc"),N)
# Creation d'une liste avec les parametres
v.param<-as.matrix(data.frame("protcmax"=v.protcmax,"protncmax"=v.protncmax,"alphac"=v.
  alphac,"alphanc"=v.alphanc,"gammac"=v.gammac,"gammanc"=v.gammanc,"lip0"=v.lip0,"lipc1
  "=v.lipc1,"lipnc1"=v.lipnc1,"beta"=v.beta,"delta"=v.delta,"amW"=v.amW,"b0c"=v.b0c,"
  b1c"=v.b1c,"b0nc"=v.b0nc,"b1nc"=v.b1nc,"c0"=v.c0,"c1"=v.c1))
# Simulation du modele avec les N parametres (N iterations)
simobs=data.frame()
for(j in 1:N)
{ sim=data.frame()
  for(id in list_individuals)
  { energyI=subset(energy,Individu==id)
    energyInd=energyI$energie/7*10
    day=energyI$time*7-6
    energy_interpol<-approx(day,energyInd,xout=1:max(day),method="constant")
    duration=length(energy_interpol$y)
    model<-carcass.model.EMI2(v.param[j,],energy_interpol,init_condition[init_condition$
      Individu==id,"Pvi"],duration)
    model=data.frame(cbind(individu=rep(id,max(day)),Ration=rep(energyI$Ration[1],max(day)
      )),model)
    sim=rbind(sim,model)
  }
  res=merge(sim[,c("individu","Ration","time","PV")],observation_dynamic,by=c("individu","
    time","Ration"),all=F)
  res=data.frame(cbind("id"=j,res))
  simobs=rbind(simobs,res)
}
# Etape 3 : Calcul de la vraisemblance/poids de PV pour chaque N parametres
LogLike_vec<-rep(NA,N)
Ssample<-rep(70,N)
for(i in 1:N)
{ LogLikelihood.i<-sum(log(dnorm(simobs[simobs$id==i,"PVobs"],simobs[simobs$id==i,"
  PV"],Ssample[i])))
  LogLike_vec[i]<-LogLikelihood.i
  #ajout d'une constante numerique pour eviter les problemes numeriques
  if(any(is.infinite(LogLike_vec))) {print("pbl_numerique,_changer_la_constante_!")}
  LogLike_vec = LogLike_vec + 3000
}
# Etape 4 : Calcul du poids normalise

```

```

w.norm<-exp(LogLike_vec)/sum(exp(LogLike_vec))
# Etape 5 : Nouvel echantillon
post.protcmax<-sample(v.protcmax,prob=w.norm,replace=T)
post.protnmax<-sample(v.protnmax,prob=w.norm,replace=T)
post.amW<-sample(v.amW,prob=w.norm,replace=T)

```

### Annexe 10 : Résultats de l'algorithme Importance Sampling pour les trois paramètres estimés pour différents N

Paramètre	N	Moyenne	EcartType	Q1%	Q5%	Q25%	Q75%	Q95%	Q99%
protcmax	100	110.9	10.9	85.3	94.3	102.9	117.6	128.9	133.4
	1000	110.5	10.2	86.9	95.1	102.8	117.7	127.1	133.4
	10000	110.2	10.3	86.2	94.5	102.7	117.8	127.2	134.3
protnmax	100	53.4	4.7	42.6	46.7	50.6	56.3	61.6	64.6
	1000	53.5	5.0	40.9	45.6	50.2	56.9	61.6	64.3
	10000	53.6	4.9	41.9	45.6	50.4	56.9	62.0	64.5
amW	100	1.43	0.10	1.25	1.26	1.36	1.50	1.59	1.64
	1000	1.42	0.11	1.11	1.25	1.34	1.50	1.59	1.66
	10000	1.41	0.11	1.11	1.24	1.34	1.49	1.59	1.67

TABLE 5 – Moyennes, écarts types et quantiles des paramètres estimés par Importance Sampling

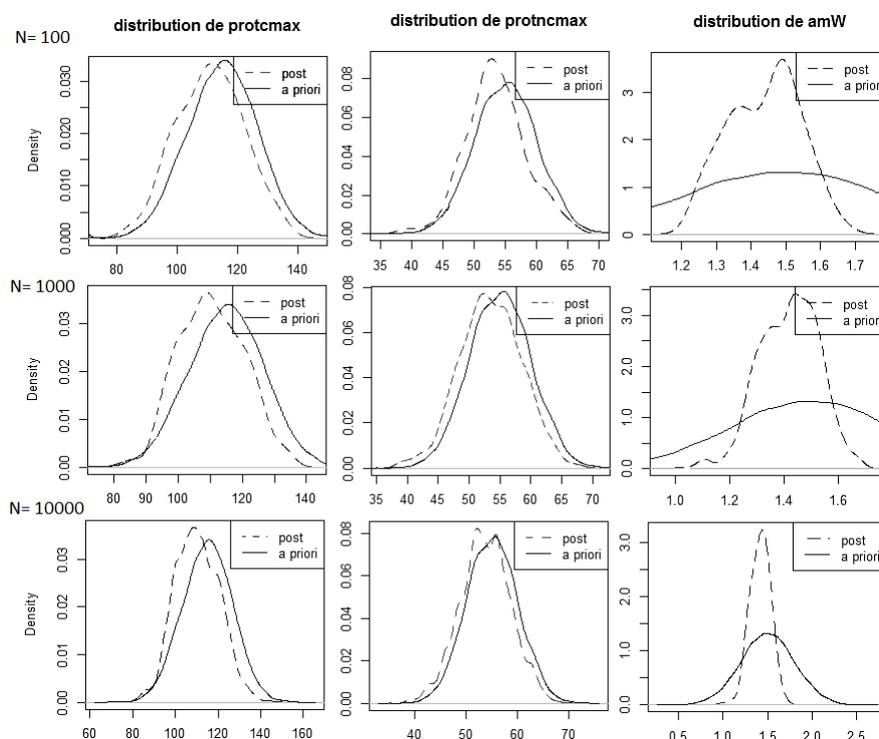


FIGURE 10 – Distribution des paramètres estimés par Importance Sampling pour différents N

## Annexe 11 : Script de l'algorithme Metropolis Hastings

```

# Fonction faisant tourner le modele pour chaque individu
carcass.multi <- function(param, list_individuals, energy, init_condition) {
sim=data.frame()
  for (id in list_individuals) {
    energyI=subset(energy, Individu==id)
    energyInd=energyI$energie/7*10
    day=energyI$time*7-6
    energy_interpol<-approx(day, energyInd, xout=1:max(day), method = "constant")
    duration=length(energy_interpol$y)
    modele<-carcass.model.EMI2(param, energy_interpol, init_condition[init_condition$Individu
      ==id, "Pvi"], duration)
    modele=data.frame(cbind(individu=rep(id, max(day)), Ration=rep(energyI$Ration[1], max(day))
      , modele))
    sim=rbind(sim, modele)
  }
  return(sim)
}

# Chargement des librairies
library(ZeBook)
library(sensitivity)
source("models/model.carcass.EMI.r")

# Lecture des donnees a partir de la library ZeBook
list_individuals=carcass_data$list_individuals[c(1,12,24),] #[1:5,]
energy=carcass_data$energy
init_condition=carcass_data$init_condition
observation_dynamic=carcass_data$observation_dynamic
observation_slaughter=carcass_data$observation_slaughter
observation_dynamic$time=observation_dynamic$time*7-6
distribution=read.csv("description/parametre_carcasse.csv", dec=".", h=T, sep=";")

# Tableau de parametres
param.default=as.matrix(rbind(nominal=distribution$ valeur, binf=distribution$ borneinf, bsup
  =distribution$ bornesup,
  mean=distribution$moyenne, sd=distribution$ecarttype))
colnames(param.default)=distribution$parametre
param.default=param.default[,c("protcmax", "protnmax", "alphac", "alphanc", "gammac", "
  gammanc", "lip0", "lipc1", "lipnc1", "beta", "delta", "amW", "b0c", "b1c", "b0nc", "b1nc", "c0",
  "c1")]
m.amW=param.default["mean", "amW"]
s.amW=param.default["sd", "amW"]
m.sigma=17

# MCMC
N=10

# Tune : parametre permettant d'ajuster la taille et la direction de la distribution
  proposee
# Calcule pour obtenir un taux de rejet compris entre 70 et 80%
tune = 0.5
sP.amW=s.amW * tune
set.seed(123)

# Vecteurs des parametres pour conserver les valeurs
v.sigma=rep(NA, N)
v.amW=rep(NA, N)
v.amWP=rep(NA, N)

```

```

# Etape 1 : Initialisation de la chaine
depart = 1
v.amW[1]=m.amW*depart
v.sigma[1:N]=m.sigma
v.param=param.default["nominal",]
v.param["amW"]=v.amW[1]
# Simulation pour le point de depart
modelTB=carcass.multi(v.param, as.vector(t(list_individuals)),energy,init_condition)
simobs=merge(modelTB[,c("individu", "Ration", "time", "PV")],observation_dynamic,by=c("
    individu", "time", "Ration"),all=F)
# Calcul de la vraisemblance pour le point de depart
v.LogLikelihood<-rep(NA,N)
v.LogLikelihood[1]<-sum(log(dnorm(simobs$PVobs, simobs$PV,m.sigma)))
# Creation des vecteurs pour le MSE et le test
v.MSE<-rep(NA,N)
v.MSE[1]=sum((simobs$PVobs - simobs$PV)^2)/length(simobs$PVobs)
v.test <-rep(NA,N)
v.test[1]=TRUE
# Boucle d'integration
Count<-1
for (j in 2:N) {
# Etape 2 : Echantillon de la valeur candidate du parametre
    amW.P<-rnorm(1, v.amW[j-1], sP.amW)
    v.amWP[j]=amW.P
    sigma<-v.sigma[j-1]
    v.param.P=param.default["nominal",]
    v.param.P["amW"]=amW.P
# Simulation et vraisemblance avec les candidats
    modelTB=carcass.multi(v.param.P, as.vector(t(list_individuals)),energy,init_condition)
    simobs=merge(modelTB[,c("individu", "Ration", "time", "PV")],observation_dynamic,by=c("
        individu", "time", "Ration"),all=F)
    LogLikelihood.P<-sum(log(dnorm(simobs$PVobs, simobs$PV, sigma)))
    SSE = sum((simobs$PVobs - simobs$PV)^2)
    nob=length(simobs$PVobs)
    Logprior.P=log(dnorm(amW.P,m.amW,s.amW))
    Logprior.0=-log(dnorm(v.amW[j-1],m.amW,s.amW))
# Etape 3 : Test
    Test.P<-(LogLikelihood.P+Logprior.P) - (v.LogLikelihood[j-1]+Logprior.0)
# Etape 4 : Selection ou rejet des candidats
    if (Test.P>0 | Test.P>log(runif(1,0,1))) { # on accepte
        v.test[j]=TRUE
        v.amW[j]<-amW.P
        v.LogLikelihood[j]<-LogLikelihood.P
        v.MSE[j]=SSE/nobs
        Count<-Count+1
    }
    else { # on rejette
        v.test[j]=FALSE
        v.amW[j]<-v.amW[j-1]
        v.LogLikelihood[j]<-v.LogLikelihood[j-1]
        v.MSE[j]=v.MSE[j-1]
    }
}

```

## Annexe 12 : Résultats de l'algorithme Metropolis Hastings pour amW pour différents N

N	Moyenne	EcartType	Q1%	Q5%	Q25%	Q75%	Q95%	Q99%
100	1.36723	0.0290182	1.31969	1.331031	1.34139	1.39109	1.403132	1.41677
1000	1.36351	0.0232134	1.31125	1.33002	1.34657	1.38059	1.40311	1.41612
10000	1.36394	0.023225	1.31484	1.32604	1.34771	1.38029	1.40438	1.41629

TABLE 6 – Moyennes, écarts types et quantiles de amW estimés par Metropolis Hastings

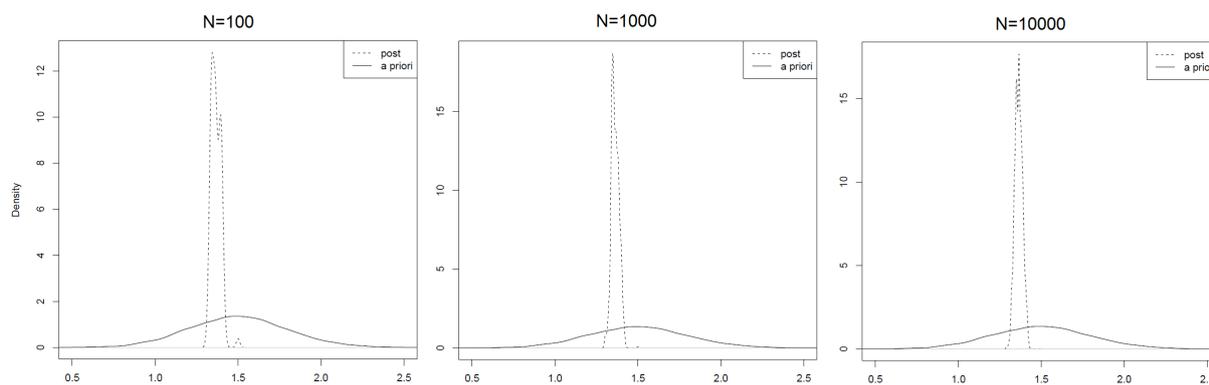


FIGURE 11 – Distribution de amW en fonction de différents N selon l'algorithme Metropolis Hastings

## **Résumé :**

La modélisation a une grande importance dans la recherche en agronomie pour permettre de suivre l'évolution d'un système au cours du temps. Cependant, la plupart des chercheurs en agronomie ne possède pas les connaissances mathématiques et statistiques nécessaires à la création et la manipulation de modèles dynamiques. Le travail présenté ici a permis la création d'un package R qui va aider les utilisateurs à obtenir ces connaissances. Il contient divers modèles centrés sur la production animale et l'élevage, ainsi que plusieurs types d'applications d'analyses statistiques sur ces modèles. Dans ce rapport sont présentées les méthodes d'évaluation et d'estimation bayésienne des paramètres du modèle. Ce package sera utilisé par le RMT Modelia ([www.modelia.org](http://www.modelia.org)) lors de formations qu'il organise pour apprendre aux chercheurs à utiliser ces méthodes. Il servira d'outil à tous les modélisateurs en agronomie grâce à la documentation et les nombreux exemples associés à chaque fonction.

## **Abstract :**

Modeling is very important for research in agronomy because it permits to follow system's evolution during the time. However, most of researchers in agronomy doesn't have mathematic's knowledge to create and manipulate dynamic models. The work presented here led to the creation of a R package that will help users to obtain this knowledge. This package contains several models focused on animal production and farming, as well as several types of statistical analysis' applications on this models. In this report we can find evaluation and parameters' bayesian estimation methods. This package will be used by RMT Modelia ([www.modelia.org](http://www.modelia.org)) during courses that it organises to learn use this methods to researchers. It will use as a tool for all agronomy modellers thanks to the documentation and the examples associated with each function.