An innovating Statistical Learning Tool based on Partial Differential Equations, intending livestock Data Assimilation

Hélène Flourent, Emmanuel Frénod, Vincent Sincholle

An innovating Statistical Learning Tool based on Partial Differential Equations, intending livestock Data Assimilation.

Hélène Flourent[a,b,*], Emmanuel Frénod[b,c,**], Vincent Sincholle[a]

[a] *NutriX*[☆], *France*
[b] *Université Bretagne Sud, Laboratoire de Mathématiques de Bretagne Atlantique,UMR CNRS* 6205, *Campus de Tohannic, Vannes, France*
[c] *See-d,* 6, *rue Henri Becquerel - CP* 101, 56038 *Vannes Cedex, France*

Abstract

The realistic modeling intended to quantify precisely some biological mechanisms is a task requiering a lot of a priori knowledge and generally leading to heavy mathematical models. On the other hand, the structure of the classical Machine Learning algorithms, such as Neural Networks, limits their flexibility and the possibility to take into account the existence of complex underlying phenomena, such as delay, saturation and accumulation.

The aim of this paper is to reach a compromise between precision, parsimony and flexibility to design an efficient biomimetic predictive tool extracting knowledge from livestock data. To achieve this, we build a *Mathematical Model* based on Partial Differential Equations (PDE) embarking the mathematical expression of biological determinants.

We made the hypothesis that all the physico-chemical phenomena occurring in animal body can be summarized by the evolution of a global information. Therefore the developed PDE system describes the evolution and the action of an information circulating in an *Avatar* of the *Real Animal*. This *Avatar* outlines the dynamics of the biological reactions of animal body in the framework of a specific problem. Each PDE contains parameters corresponding to biological-like factors which can be learnt from data by the developed *Statistical Learning Tool*.

Keywords: Statistical Learning, PDE, Forecasting, Data Assimilation, Model-Data Coupling, Biological Mathematical Modeling

1. Introduction

According to Vázquez-Cruz *et al.* (2014), among the existing methods for analyzing biological data, two approaches can be distinguished. The first one corresponding to a

☆Society wishing to remain anonymous.
*helene.flourent@univ-ubs.fr
**emmanuel.frenod@univ-ubs.fr

realistic modeling, aims at the exact description and quantification of all the biological processes observed from the injection or the ingestion of a set of molecules until its action somewhere in a living organism. In the image of the works achieved by Bastianelli and Sauvant (1997) and by Martin and Sauvant (2010a) the construction of realistic models is a task requiring time and a lot of biological knowledge and generally leading to models containing a large number of equations and parameters. The in-silico experiments allowed by this type of models are valuable to describe and explain specific biological processes through the use of particular *Inputs*. However, the complex implementation of these models limits their adaptability and flexibility, in particular when it comes to processing field data presenting high variability, missing and aberrant values. The second approach corresponds to «Black Box» models, such as Neural Networks. As it is explains in Domingos (2012), for a decade the use of Machine Learning (ML) algorithms and especially Neural Networks (NN) has been on the rise. According to Gorczyca *et al.* (2018), Valletta *et al.* (2017), Ma *et al.* (2014) and Ip *et al.* (2018), the popularity of these tools can be explained by the ease of their implementation and the diversity of issues which can be dealt with by those algorithms. Nevertheless, these algorithms are based on relatively simple mathematical models unsuitable to take easily into account complex phenomena such as delay and saturation. Hence, the tools based on those types of ML algorithms contain little a priori biological knowledge. Thus, Tan and Gilbert (2003), Shavlik *et al.* (1995), Hubbard and Reinhardt (1998) and Dumpala *et al.* (2017) explain that it is necessary to learn the parameters of these models from a lot of data to compensate the absence of biological expertise.

The goal of this paper is to introduce a new paradigm combining those two kinds of approaches. Our first aim is to build a tool able to link *Inputs* and *Outputs* concerning an animal or a group of farm animals to thereafter perform simulation and forecasting. The second aim is to be able to interpret and synthesize a more or less continuous data stream collected in farms, in order to perform Data Assimilation. Renzullo *et al.* (2008), Ingalls (2019), Zúñiga *et al.* (2014), Vázquez-Cruz *et al.* (2014) and McPhee (2009), explain that to achieve those objectives, it is necessary to have on hand a mathematical model which is able to take into account some aspects of the dynamics of the system under study, corresponding in this study to animal body. In the light of the limits of the already existing methods for predicting biological responses, we decided to explore an approach aiming the construction of a tool combining accuracy, parsimony and flexibility. We designed a biomimetic predictive tool able to deal with the existence of complex underlying phenomena. To achieve this, we built an advanced *Mathematical Model* based on a system of Partial Differential Equations (PDE) embarking the mathematical expression of various biological phenomena (diffusion, convection, accumulation, saturation, *etc.*) and depending on parameters carrying the leaning capability of the tool.

After this Introduction, putting this research work in its proper context, we will detail in the second Section the encountered problems in the field of biological modeling. In this particular context the existing tools are limited and not totally suitable to achieve the previously presented objectives. Therefore, in the third Section we will present the *Mathematical Model* and the built *Statistical Learning Tool*. In Section 4, we will study the functioning of the *Mathematical Model* and the ranges of values of the different involved parameters. To verify the tool capacities, several tests are

performed. In Section 5 we will present the tests by simulation performed in order to verify the ability of the model to learn parameters from noisy data. Then, in Section 6, an application of our approach on field data concerning the growth of animals during a given period will be presented. This application demonstrates the prediction capability of the tool in real conditions. In order to have an idea about the real potential of this new *Statistical Learning Tool* we compared our biomimetic model with some Logistic Models, Mechanistic Models and some Machine Learning algorithms such as Neural Networks. Those comparisons will be detailed in Section 7.

2. Problem description

2.1. Biological modeling: difficulties and challenges

In their review, Dumas *et al.* (2008) explain that the construction of mathematical models to tackle livestock production issues, began between 1910 and 1925 with the aim of predicting and simulating processes by integrating knowledge. Nowadays, according to Dumas *et al.* (2008), McPhee (2009) and Vázquez-Cruz *et al.* (2014), mathematical modeling remains decisive to simplify, describe and simulate the mechanisms and the links existing between factors especially in biological field survey. As it can be identified in McPhee (2009), Puillet *et al.* (2011), Martin and Sauvant (2010b), Nkrumah *et al.* (2007), Nesetrilova (2005) and Basarab *et al.* (2003), in the agri-food sector, simulate and predict the effects of nutrition on animal performances are two decisive and strategic points for breeders and companies to understand how optimize animal efficiency.

Yet, as it is illustrated in Locke *et al.* (2005) and Qi *et al.* (2006), databases collected on living organisms generally contain a large amount of variability. A part of this variability is related to individual differences. There is also noise, generated by the measuring instruments and heterogeneity resulting to the lack of continuity over the various experiments. This variability is in addition to more or less missing and aberrant values. Moreover, farm data collection comes within an evolving framework. Indeed Jemila and Priyadharsini (2018), Miekley *et al.* (2012), Tol and Kamp (2010), Büchel and Sundrum (2014) and Holman *et al.* (2011), present some new technologies enable the monitoring of animals (connected collar, troughs recording the feeding behavior, connected scales, boluses...). But these are still expensive and their democratization takes time.

All those elements constrained and guided our modeling approach. As a mater of fact, the choices we made permit to carry-out simulations and Data Assimilation via a light and parsimonious tool. This parsimony allows also to quickly adapt our model to the different farm species studied by the agri-food companies. Our choices also lead to a tool having a high information extraction potential. This extraction potential aims to make our tool compatible with the complexity of the studied phenomena coupled with the current lack of exploitable data, as well as with the big volumes of data which will result from the evolution of farm data collection.

2.2. Exploration of an intermediate approach: The Model-Data Coupling

The intermediate approach we implemented combines the integration of knowledge and the usage of data in order to extract complex information from available data. Therefore our work falls within the Model-Data Coupling theory. Model-Data Coupling

is essentially used in the fields of meteorology (See Simmons and Hollingsworth (2002)), hydrology (See Kim and Barros (2002),Crosson *et al.* (2002) and Mackay *et al.* (2003)), biogeochemistry (See Barrett (2002), Barrett *et al.* (2005), Rayner *et al.* (2005) and Sacks *et al.* (2006)) and oceanography (See Ailliot *et al.* (2006)). Like biology, these fields are domains in which it is necessary to take into account certain aspects of the dynamics of the studied system to perform forecasting. But the system under study is often complex and its exact modeling would take time and result in a heavy mathematical model. Therefore, as it can seen in Frénod (2017), Rousseau and Nodet (2013), Sacks *et al.* (2007) and Wang *et al.* (2010), the approach consists in building a parsimonious mathematical model, corresponding to a synthetically mathematical translation of the studied system. Then the parameters contained in this model are optimized and fitted from data. As in those studies, the construction of our tool is based on an optimal combination between knowledge - to design a $Mathematical\ Model$ presenting the «optimal» degree of complexity - and data - to optimize the model parameters - in order to obtain a predictive tool which is both accurate, parsimonious and flexible.

To be more specific, we built a $Mathematical\ Model$ based on a system of PDE embarking the mathematical expression of various biological phenomena such as diffusion, convection, accumulation, saturation, *etc.* This $Mathematical\ Model$ aims at integrating biological knowledge but it is not intended to describe with precision and exactitude what is occurring in an animal body. Indeed, the objective is to build a Biomimetic *Statistical Learning Tool* able to predict accurately biological responses. To achieve this target, we want this tool to take into account the global biological dynamics occurring in animal body but without necessarily describing all the processes inducing those responses. Indeed, our exploration is based on the hypothesis that the synthetic consideration of the biological processes may enable to gain in precision, in comparison with a classical Machine Learning tool which integrate no a priori knowledge, while keeping a parsimonious and light tool, in comparison with a realistic tool.

Our exploration relies on the articulation of several and diverse elements (Figure 1).

The *Real Animal* is a complex living organism in which a high number of physical flows and chemical reactions interact and act. Therefore, our support of reflection is not directly the *Real Animal*. The support of reflection used to construct the $Mathematical\ Model$ is an *Avatar* of the *Real Animal*. This *Avatar* outlines in the framework of a specific problem, the dynamics of the biological reactions occurring in animal body. In the present study we decided to outline all those physico-chemical phenomena by the circulation, the evolution and the action of a global information. Therefore, this information synthesis all the phenomena of convection, diffusion, accumulation, saturation and delay, that a set of molecules may undergo in the body of an animal.
The $Mathematical\ Model$ mathematically traduces the evolution and the action of a global information circulating in the *Avatar*.

Therefore we distinguished different dimensions. There is the *Reality* in which there are $Intakes$ and $Injections$ inducing complex biological processes in animal body. Some $Sensors$ permit to extract from this *Reality* databases made of $Inputs$ and $Outputs$. The $Inputs$ are traduced by a mathematical function in $Entries$, that are pieces of

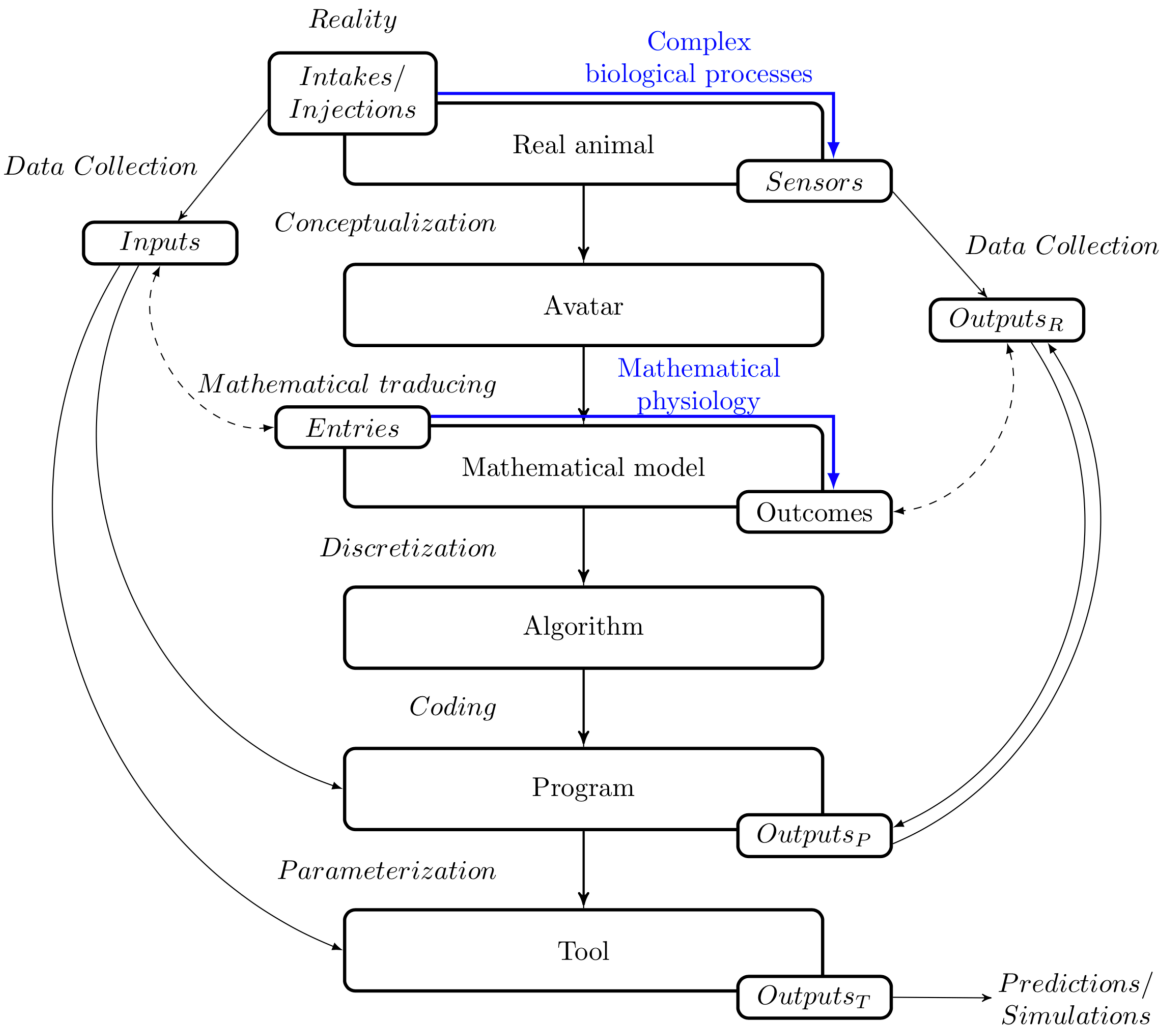

Figure 1: Articulation of the elements of the exploration

information integrated into the *Mathematical Model* and inducing the generation of *Outcomes*, also linked to the *Outputs* extracted from the *Reality* by a mathematical function.

The *Mathematical Model* has no biological state. Indeed, The *Real Animal* has a biological condition induced by the introduction of molecules in its body. Whereas, the model has no biological condition but a physiological-like condition induced by the integration of *Entries* in an involved geometrical space. This mathematical physiological-like status links the *Entries* and the *Outcomes*.

The *Algorithm* comes out of the discretization of the *Mathematical Model*, that is the PDE system mathematically translating what takes place in the *Avatar*. This system of PDE contains parameters corresponding to biological-like factors: convection and diffusion speeds, some saturation levels, the fixation speed, *etc.* These parameters can be learnt from a database and by using optimization algorithms. Therefore, the presence in the *Mathematical Model* of parameters which can be learnt from data, confers to the tool based on this model a learning ability. Hence, the constructed tool is a *Statistical Learning Tool*.

The *Program* corresponds to the code permitting to manage the learning of the pa-

rameter values via an iterative process during which an optimization algorithm permits to find the values of the parameters minimizing the difference between the measured and the predicted *Outputs*.
The *Tool* finally corresponds to the *Mathematical Model* parametrized with the values of the parameters obtained at the end of the learning step.

3. Structure and discretization of the *Mathematical Model*

3.1. Description of the Mathematical Model

We worked under the hypothesis that, when an active or a molecule enters the body of a living organism, it circulates in the body through a network of vessels containing a fluid. This element integrates this fluid and uses it as a vector to evolve via convection and diffusion phenomena. In the network of vessels the element may be in competition with others which may delay its progression. Then the circulating element may be caught and accumulated into an organ or a specific tissue. During its storage the element can be used and induce change in some biological variables.
We can mathematically traduce all those processes through a PDE system which is illustrated by Figure 2.

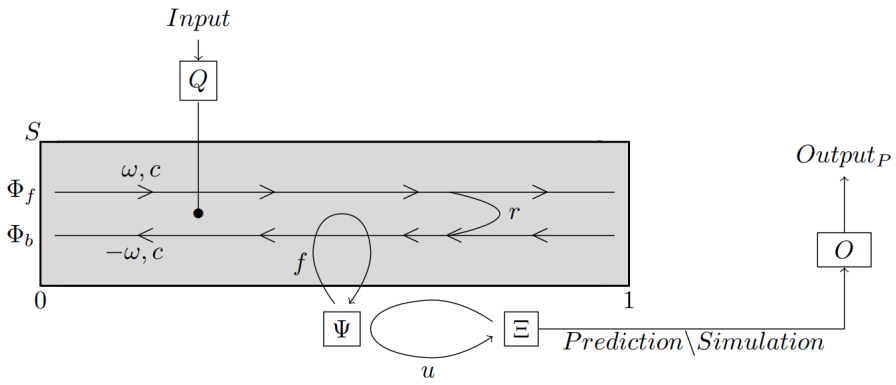

Figure 2: Schematization of the *Mathematical Model*

Concretely, we decided to model our *Avatar* using variables, densities and fields that are all unit-less and dimensionless. We also decided to reduce the geometrical space S relative to the *Avatar* to interval $[0\,;1]$. We considered a Forward Flow Φ_f, and a Backward Flow Φ_b streaming in this one-dimensional geometrical space. These flows could be seen as a very synthetic summary of a blood, a nervous or a digestive circulation. The involved *Inputs* essentially correspond to collected data concerning intakes, water intakes and medicine injections. Those *Inputs* can be included within the *Mathematical Model* via a function Q transforming those *Inputs* into information inflows, named *Entries* and injected in the involved geometrical space. A part of the injected information circulates forward, via Φ_f and the rest circulates backward, via Φ_b. This information evolves via convection and diffusion phenomena. During its circulation the information can be delayed, caught, stored and used to finally induce a modification of the *Outcome* O. This *Outcome* corresponds to the model prediction or simulation of a biological variable .
Therefore, to an information $\mathtt{d}$, are associated different elements. $\{\Phi_f(\mathtt{d})\}(t,\mathbf{x})$ and

$\{\Phi_b(\mathtt{d})\}(t,\mathbf{x})$ are at each instant t two spatial densities respectively associated to a forward flux with a velocity $\omega_{\mathtt{d}}$ and a backward flux with a velocity $-\omega_{\mathtt{d}}$.
The spacial density $\{\Phi_f(\mathtt{d})\}(t,\mathbf{x})$ is supposed to be solution to:

$$\frac{\partial\{\Phi_f(\mathtt{d})\}}{\partial t}(t,\mathbf{x})+\omega_{\mathtt{d}}\frac{\partial\{\Phi_f(\mathtt{d})\}}{\partial \mathbf{x}}(t,\mathbf{x})-\frac{\partial\left[c_{\mathtt{d}}\chi\dfrac{\partial\left[\{\Phi_f(\mathtt{d})\}+\{\Phi_b(\mathtt{d})\}\right]}{\partial\mathbf{x}}\right]}{\partial\mathbf{x}}(t,\mathbf{x})$$
$$=\frac{1}{2}\{Q(\mathtt{d})\}(t,\mathbf{x})-f_{\mathtt{d}}\{F(\mathtt{d})\}(\mathbf{x})\{\Phi_f(\mathtt{d})\}(t,\mathbf{x})-r_{\mathtt{d}}\{\Phi_f(\mathtt{d})\}(t,\mathbf{x}),\quad(1)$$

Similarly, $\{\Phi_b(\mathtt{d})\}(t,\mathbf{x})$ is supposed to be solution to:

$$\frac{\partial\{\Phi_b(\mathtt{d})\}}{\partial t}(t,\mathbf{x})-\omega_{\mathtt{d}}\frac{\partial\{\Phi_b(\mathtt{d})\}}{\partial \mathbf{x}}(t,\mathbf{x})-\frac{\partial\left[c_{\mathtt{d}}\chi\dfrac{\partial\left[\{\Phi_f(\mathtt{d})\}+\{\Phi_b(\mathtt{d})\}\right]}{\partial\mathbf{x}}\right]}{\partial\mathbf{x}}(t,\mathbf{x})$$
$$=\frac{1}{2}\{Q(\mathtt{d})\}(t,\mathbf{x})-f_{\mathtt{d}}\{F(\mathtt{d})\}(\mathbf{x})\{\Phi_b(\mathtt{d})\}(t,\mathbf{x})+r_{\mathtt{d}}\{\Phi_f(\mathtt{d})\}(t,\mathbf{x}),\quad(2)$$

In those equations, the parameter $c_{\mathtt{d}}$ is the diffusion velocity of the information. The space time density $\{Q(\mathtt{d})\}$, corresponds to an external source of information. The function $\{F(\mathtt{d})\}$ is worth 0 in certain area of the involved geometrical space and 1 in others. The area where this function is worth 1 corresponds to the location of the entity catching the information. The parameter $f_{\mathtt{d}}$ determines the rate of fixed information. The parameter $r_{\mathtt{d}}$ determines the part of the circulating information transferred from the Forward Flow to the Backward Flow which induces a delay in the progression of the information. At each instant t, the spatial density $\{\Psi(\mathtt{d})\}(t,\mathbf{x})$, associated to the fixed information, is solution to:

$$\frac{\partial\{\Psi(\mathtt{d})\}}{\partial t}(t,\mathbf{x})=f_{\mathtt{d}}\{F(\mathtt{d})\}(\mathbf{x})\left[\{\Phi_b(\mathtt{d})\}(t,\mathbf{x})+\{\Phi_f(\mathtt{d})\}(t,\mathbf{x})\right]-u_{\mathtt{d}}\{\Psi(\mathtt{d})\}(t,\mathbf{x}).\quad(3)$$

The parameter $u_{\mathtt{d}}$ is the coefficient determining the usage rate of the fixed information. At each instant t, the spatial density $\{\Xi(\mathtt{d})\}(t,\mathbf{x})$, associated to the used information, is solution to:

$$\frac{\partial\{\Xi(\mathtt{d})\}}{\partial t}(t,\mathbf{x})=u_{\mathtt{d}}\{\Psi(\mathtt{d})\}(t,\mathbf{x}).\quad(4)$$

The parameter $\Omega(\mathtt{d})$ corresponds to the action area of the circulating information on the *Outcome*. $\{O(\mathtt{d})\}(t)$ is the *Outcome* of the model, given by:

$$\{O(\mathtt{d})\}(t)=\int_{\Omega(\mathtt{d})}\{\Xi(\mathtt{d})\}(t,\mathbf{x})\,d\mathbf{x}.\quad(5)$$

3.2. The «usage» equation

The fourth equation of the model is the «usage» equation. This equation determines the action of the injected information on the variable to predict. Therefore, this equation has to adapt the different ways in which an intake or an injection may impact a

biological variable. Equation (4) models an accumulative phenomenon. Hence it can be used to tackle data concerning the evolution of a total production over a given period.

To model a limited growth, a limiter is added in this equation. In this case, the «usage» equation becomes:

$$\frac{\partial\{\Xi(\mathtt{d})\}}{\partial t}(t,\mathbf{x}) = u_{\mathtt{d}}\{\Psi(\mathtt{d})\}(t,\mathbf{x})\left(\frac{L_{\mathtt{d}} - \{O(\mathtt{d})\}(t)}{L_{\mathtt{d}}}\right) \tag{4b}$$

With this version of the equation, data related to the weight evolution of an animal can be treated. This equation can be assimilated to the differential equation of Verhulst (1838):

$$\frac{\partial y}{\partial t}(t) = r\ y(t)\left(\frac{K - y(t)}{K}\right) \tag{6}$$

whose structure is equivalent. We indeed may notice that in the case when nothing depends on x, the value of $u_{\mathtt{d}}$ is very high and $\Omega(\mathtt{d})$ is the whole interval $[0\ ;\ 1]$, $\{\Xi(\mathtt{d})\}$, $\{\Psi(\mathtt{d})\}$ and $\{O(\mathtt{d})\}$ are very close to each other. Hence, Equation (6) and (4b) are essentially the same.

It could be also necessary to model variations to use our tool to treat data about drug impacts on a biological variable for example. To do that, we have to be able to model an increase or a decrease in the *Outcome* which could variate between an upper and a lower bound. The equation

$$\frac{\partial\{\Xi(\mathtt{d})\}}{\partial t}(t,\mathbf{x}) = -\Big(\{\Xi(\mathtt{d})\}(t,\mathbf{x}) - Upp_{\mathtt{d}}\Big) - u_{\mathtt{d}}\{\Psi(\mathtt{d})\}(t,\mathbf{x})\Big(\{\Xi(\mathtt{d})\}(t,\mathbf{x}) - Low_{\mathtt{d}}\Big) \tag{4c}$$

models that the fixed information $\{\Psi(\mathtt{d})\}$ attracts the *Outcome* $\{O(\mathtt{d})\}$ toward a state which is lower than the steady one, and the equation

$$\frac{\partial\{\Xi(\mathtt{d})\}}{\partial t}(t,\mathbf{x}) = -u_{\mathtt{d}}\{\Psi(\mathtt{d})\}(t,\mathbf{x})\Big(\{\Xi(\mathtt{d})\}(t,\mathbf{x}) - Upp_{\mathtt{d}}\Big) - \Big(\{\Xi(\mathtt{d})\}(t,\mathbf{x}) - Low_{\mathtt{d}}\Big) \tag{4d}$$

models that the fixed information $\{\Psi(\mathtt{d})\}$ attracts the *Outcome* $\{O(\mathtt{d})\}$ toward a state which is upper than the steady one. In these two previous cases the *Outcome* varies between a lower bound $Low_{\mathtt{d}}$ and an upper bound $Upp_{\mathtt{d}}$.

The «usage» equation has to be defined according to the problematic and the acquired knowledge about the link existing between the *Entries* and the *Outcomes*.

3.3. Initial and boundary conditions

The function χ is compactly supported in $(0,1)$, mainly constant and worthing 1. This function integrated in the diffusion term permits to make the diffusion vanish at the edges of the domain.

We also imposed :

$$\forall t \in (0,\infty),\ \big\{\Phi_f(\mathtt{d})\big\}(t,0) = \big\{\Phi_b(\mathtt{d})\big\}(t,0) \text{ and } \big\{\Phi_b(\mathtt{d})\big\}(t,1) = \big\{\Phi_f(\mathtt{d})\big\}(t,1) \qquad (7)$$

Those conditions allows the circulating information to move back and forth between the two edges of the domain.

The initial conditions $\big\{\Phi_f(\mathtt{d})\big\}(0,\mathbf{x})$, $\big\{\Phi_b(\mathtt{d})\big\}(0,\mathbf{x})$, $\big\{\Psi(\mathtt{d})\big\}(0,\mathbf{x})$, $\big\{\Xi(\mathtt{d})\big\}(0,\mathbf{x})$ and $\big\{O(\mathtt{d})\big\}(0)$ are given for all x in $(0,1)$.

3.4. The model parameters

The system of Partial Differential Equations contains several parameters that have to be learnt. There are $\omega_{\mathtt{d}}$, $c_{\mathtt{d}}$, $r_{\mathtt{d}}$, $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$. To simplify, in the first studies we fixed $c_{\mathtt{d}}$ at 0.001.

All the other parameters are learnt from a database by using an optimization algorithm permitting to find the parameter values minimizing the error associated to the model on a training database. To do that we used the function *directL* developped by Johnson (2008), which is embedded in R (R Core Team (2014)) and applying the DIRECT algorithm developped by Finkel (2003).

Among the parameters $\omega_{\mathtt{d}}$, $r_{\mathtt{d}}$, $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$ some parameters offset each other.

The speed impact $\omega_{\mathtt{d}}$, may be offset by the delay $r_{\mathtt{d}}$, undergone by the information. Indeed, a low convection speed associated to a low delay may induce equivalent kinetics to the one induced by a high convection speed associated to an important delay.

The fixation $f_{\mathtt{d}}$, and the use of the information $u_{\mathtt{d}}$, are also two counterbalanced processes. Indeed an important fixation followed by a low usage of the information may induce the same impact on the *Outcome* as a low fixation followed by an important use of the fixed information.

The compensation effects existing between the parameters call into question the identifiability of the model. Indeed, if the parameters counterbalanced each others there may exist a series of couples $(\omega_{d_{Opt}}, r_{d_{Opt}})$ and $(f_{d_{Opt}}, u_{d_{Opt}})$ minimizing the error associated to the model on the *Training Database*. Therefore some studies of those compensation effects are introduced in Section 5.3 and the unicity of the set of optimal parameter values is verified in Section 5.4.

4. Study of the *Mathematical Model* functioning

To discretize the *Mathematical Model* we first used the classical Finite Difference method with a given space step, to obtain semi-discrete in space equations. And, since the *Mathematical Model* is coded under the software R, we used the R-function *Ode*.1*D* developped by Soetaert *et al.* (2010) to manage the discretization in time of the semi-discrete equations. This R-function calls upon the fourth order Runge Kutta method with a given time step (See Enright (1989)).

4.1. Mathematical study of the model and its discretization

A detailed mathematical analysis of the model and its discretization will be performed in an upcoming paper. Nevertheless, we already know that since we fixed

the discretization steps, the convection and diffusion speeds have to respect the CFL conditions and be not larger than given limits (See Courant *et al.* (1928) and Weisstein (2014)). In this first exploration, in order to find a compromise between precision and calculation time we decided to parametrize the mesh with a time step of 0.001 and a space step of 0.025. Therefore ω must be smaller than 25 and c must be smaller than 0.625.

We also already observed some properties of the model that we briefly describe in the forthcoming paragraphs.

4.2. Study of the ranges of values of the parameters

Before starting the learning of the parameters, we have to specify for each parameter a lower and an upper values between which the optimization algorithm will search the value minimizing the error associated to the model.

We already know that all the parameters are positive, hence the lower bound of the different ranges of values is zero.

We also know that ω and c have to respect the CFL conditions. Therefore the upper bound of these parameters are worth respectively 25 and 0.625 (See Section 4.1).

Then, a saturation effect of the impact of the parameters $r_{\mathtt{d}}$, $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$ on the model is observed. A comprehensive study of this phenomenon and its components was performed in the working paper Flourent (2019). We refer to it for the details of this study of which we give only a few elements in the present paper. Figure 3 illustrates what can be observed when several *Output Curves*, $O(t)$ are generated by setting the value of all the parameters but one. The modulated parameter is $f_{\mathtt{d}}$, $u_{\mathtt{d}}$ or $r_{\mathtt{d}}$. The color gradient applied to the curves is associated to the value of the studied parameter: The higher the value of the modulated parameter, the darker the *Output Curve* generated from the parameterized model. Therefore, Figure 3 shows that when the value of one of these three parameters increases, the evolution of the *Output Curve* profile slows down, settles down and then does not evolve anymore. Therefore, for each parameter there exists a saturation level beyond which the parameter does not influence the model anymore. Indeed the range of values of each of these parameters corresponds to an interval from 0 to the saturation level of the impact of this parameter on the model.

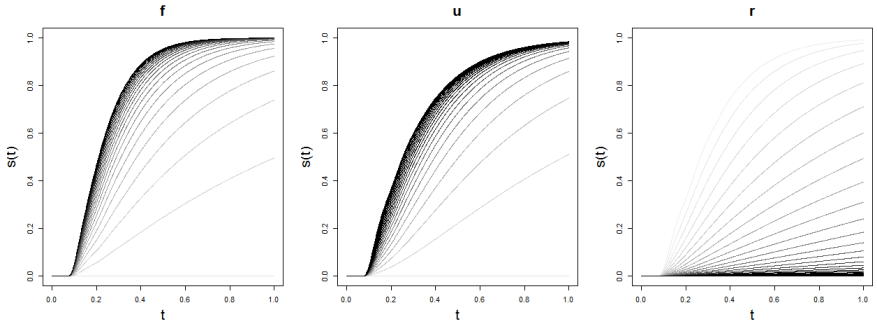

Figure 3: Saturation of the impact of the parameters on the model

To know for each parameter the value of this saturation level we calculated an indicator of the evolution speed of the *Output Curve* profile according to the value of the studied parameter. When this indicator becomes very low that means the profile of

the *Output Curve* hardly evolves anymore and that the saturation level of the studied parameter is achieved. See Flourent (2019) for further details.

Therefore, from the built saturation indicator and the CFL conditions, we established the ranges of values of all the parameters. They are given in Table 1.

Table 1: The ranges of values of the different parameters.

Parameter	Range of values
ω	$[0; 25]$
c	$[0; 0.625]$
u	$[0; 200]$
r	$[0; 284]$
f	$[0; 1035]$

5. Simulation tests of the learning capability of the model

The objective of this Section is to present the tests by simulation, performed to verify the ability of the tool to learn parameters from noisy biological data. To do that we started by generating a fictitious database from our parametrized *Mathematical Model*. Then we used this database to study the compensation effects existing between the parameters. Finally we simulated the learning of the parameters from the fictitious data and verified if the fitting of the model was done correctly.

5.1. Generation of a Learning Database

In order to test the learning capability of the model we generated a Learning Database containing 50 individuals, that is 50 *Output Curves*. The objective is to obtain a database having the same characteristics as a real field database. To do that we integrated in this fictitious database noise and individual variability.

5.2. Integration of individual variability

The model parameters are constants to determine. Nevertheless, in order to introduce individual variability in the generated data, we considered -only in this Section- the parameters as biological-like factors following a Normal distribution. Indeed, to simulate individual differences we assigned to each parameter a Normal distribution centered in an arbitrarily chosen value and with a relative variance of 0.005 (See Table 2). From those Normal probability laws we generated 50 values of the parameters $\omega_{\mathtt{d}}$, $r_{\mathtt{d}}$, $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$. Their respective statistical and probabilistic distributions are drawn in Figure 4.

5.2.1. Generation of fictitious Inputs

The *Inputs* integrated in the model correspond to the injected volume ($VolQ$) and the moment of the injection (c_t). These parameters can take any values between 0 and 1, therefore we applied to these two types of *Inputs* an Uniform distribution over the interval $[0; 1]$ (Table 2).

From the values of the parameters and the fictitious *Inputs*, we generated 50 *Output Curves*.

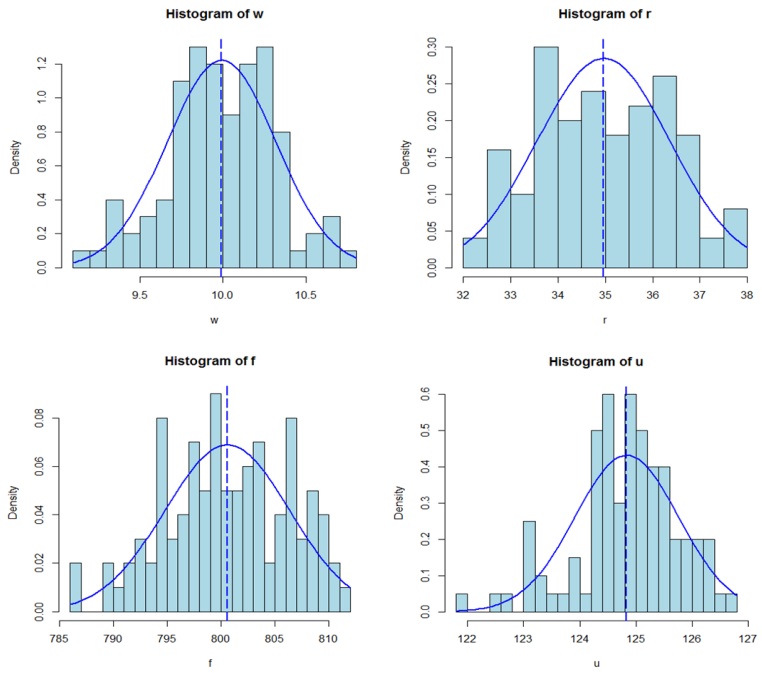

Figure 4: Distributions of the parameters ω, r, f and u

5.2.2. Addition of a random noise

Still with the objective of obtaining an experimental-like database, we add noise to the *Output Curves*. To do that we add to the generated curves a random component following a Gaussian distribution centered in 0 and with a variance of 0.05 (Table 2).

Figure 5 shows some examples of generated curves without and with noise. We divided the obtained database into two datasets: A *Training Database* made of 30 curves and a *Test Database* made of 20 curves.

In the rest of this Section, we supposed that we have an experimental-like database and a model containing four parameter values to determine.

Table 2: The distributions followed by the parameters and the *Inputs*.

Parameter	Probability law
ω	$\mathcal{N}(10, 0.3125)$
r	$\mathcal{N}(35, 1.42)$
f	$\mathcal{N}(800, 5.175)$
u	$\mathcal{N}(125, 1)$
$VolQ$	$\mathcal{U}(0, 1)$
c_t	$\mathcal{U}(0, 1)$
$Noise$	$\mathcal{N}(0, 0.05)$

5.3. Study of the compensation effects

The couples $(\omega_{\mathtt{d}}, r_{\mathtt{d}})$ and $(f_{\mathtt{d}}, u_{\mathtt{d}})$ are two couples of counterbalanced parameters. Therefore, relations exist between the parameters of those two couples. The objective of this part is to use the fictitious *Training Database* to study those relations.

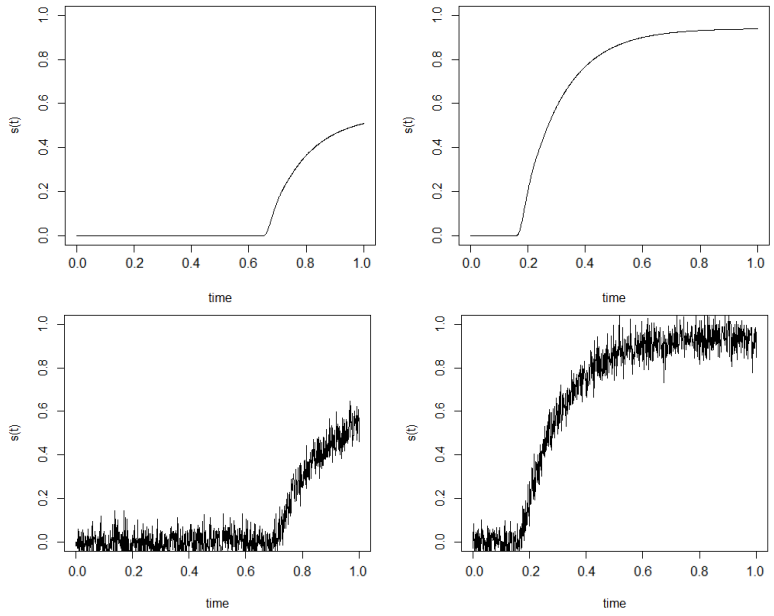

Figure 5: Example of simulated curves without and with noise

5.3.1. Study of the relation existing between ω and r

As a first step, we decided to put in evidence the relation existing between $\omega_{\mathtt{d}}$ and $r_{\mathtt{d}}$ by calculating the error made on the *Training Database* by the model parametrized with different couples $(\omega_{\mathtt{d}}, r_{\mathtt{d}})$. To do that we ranged the domain $\omega_{\mathtt{d}} \times r_{\mathtt{d}}$ and we calculated the Relative Residual Sum of Squares ($RRSS$) (8) associated to the models parametrized with different tested couples $(\omega_{\mathtt{d}}, r_{\mathtt{d}})$:

$$RRSS(\omega_{\mathtt{d}}, r_{\mathtt{d}}) = \sum_{i=1}^{n} \left(\sum_{j=1}^{m} \left(\frac{(y_{ij_{obs}} - y_{ij_{pred}}(\omega_{\mathtt{d}}, r_{\mathtt{d}}))}{y_{ij_{obs}}} \right)^2 \right), \tag{8}$$

where n corresponds to the number of individuals contained in the *Training Database* and m the number of points on the curves. $y_{ij_{obs}}$ and $y_{ij_{pred}}$ correspond respectively to the observed and the predicted value of the j^{th} point of the i^{th} individual. Therefore $RRSS$ corresponds to the sum of the squared relative differences between the predicted curves and the initially generated curves.

Figures 6 and 7 represent the values of the RRSS according to the values of $\omega_{\mathtt{d}}$ and $r_{\mathtt{d}}$. The existence of a series of equivalent couples - that is a series of couples inducing the same value of $RRSS$ - can be seen in Figure 6(a). There is an area where the $RRSS$ are lower (Figure 7) and corresponding to the curve $EC1$ of Figure 6(b). We took for granted that the optimal couple $(\omega_{d_{Opt}}, r_{d_{Opt}})$ inducing the lowest RRSS, belongs to this curve. Therefore we sought the equation of the curve $EC1$.

5.3.2. Search of the couples $(\omega_{d_{Opt}}, r_{d_{Opt}})$ inducing the lowest RRSS

To find the equation of the curve $EC1$ we sought for different values of $\omega_{\mathtt{d}}$, the value of $r_{\mathtt{d}}$ minimizing the $RRSS$ value. To do that, for each tested value of $\omega_{\mathtt{d}}$ we used

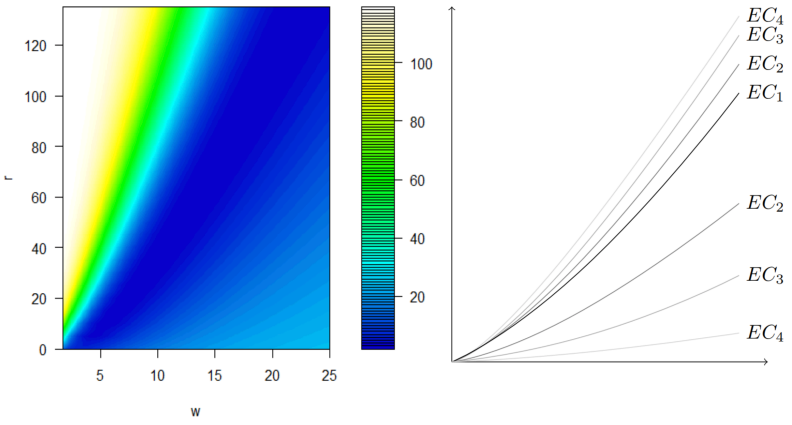

Figure 6: The value of the $RRSS$ according to ω and r (a), and the schema of the different Equivalent Couples (EC) (b)

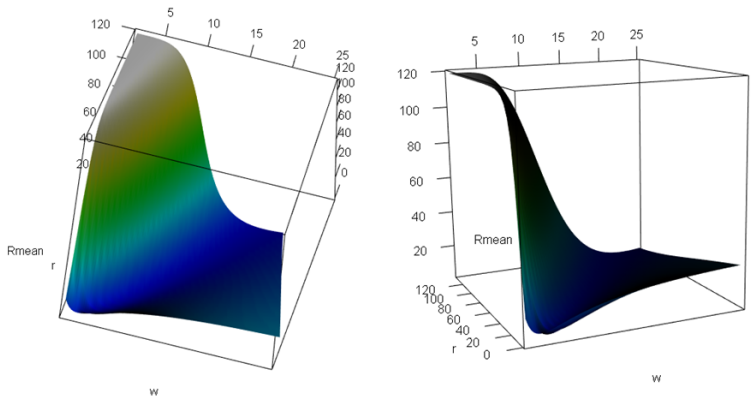

Figure 7: The $3D$ representation of the value of the $RRSS$ according to ω and r

the optimization algorithm DIRECT to find the value of $r_{\mathtt{d}}$ minimizing the objective function,

$$f_{obj}(r) = \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m}\left(\frac{(y_{ij_{obs}} - y_{ij_{pred}}(\omega, r))}{y_{ij_{obs}}}\right)^2\right) \tag{9}$$

corresponding to the average RRSS.

In order to have several fitted values of $r_{\mathtt{d}}$ for each tested value of $\omega_{\mathtt{d}}$ we performed a sampling of the *Training Database*. Indeed, for each fitting we sampled 20 curves from 30 and we fitted $r_{\mathtt{d}}$ on those 20 selected curves. At the end of the fitting we obtained three values of $r_{\mathtt{d}}$ for each tested value of $\omega_{\mathtt{d}}$ (Figure 8). Thanks to a Nadaraya-Watson kernel regression (See Nadaraya (1964) and Watson (1964)), we obtained a non-parametric relation linking $\omega_{d_{Opt}}$ and $r_{d_{Opt}}$ in the form of:

$$r_{opt} = \hat{m}(\omega_{opt}) + \epsilon, \tag{10}$$

where $\hat{m}$ corresponds to the Nadaraya-Watson estimator.

Knowing the relation existing between $\omega_{d_{Opt}}$ and $r_{d_{Opt}}$, it is possible to deduce one of these two parameters according to the value of the other one. Hence, this relation permits to reduce the number of parameters to simultaneously learn.

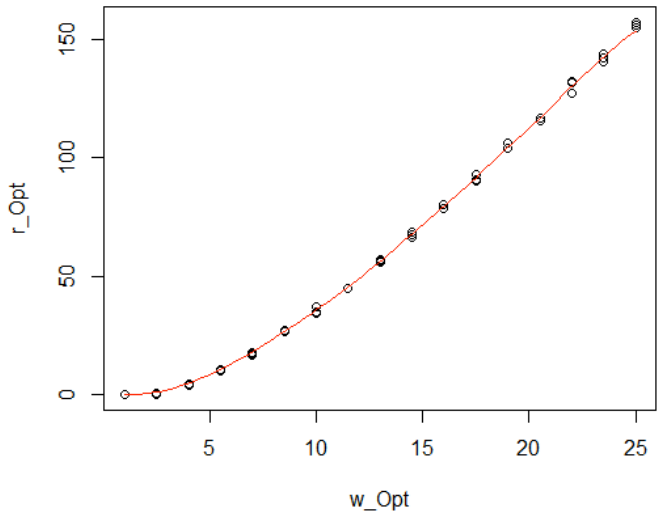

Figure 8: The Nadaraya-Watson kernel regression linking the couples $(\omega_{d_{Opt}}, r_{d_{Opt}})$.

5.3.3. Study of the relation between the parameters $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$

There also exists a compensation effect between $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$: a high value of $f_{\mathtt{d}}$ can be compensated by a low value of $u_{\mathtt{d}}$, and the contrary.

As previously for $\omega_{\mathtt{d}}$ and $r_{\mathtt{d}}$, we sought the relation existing between $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$ in order to be able to deduce one of these two parameters according to the other one and further reduce the number of parameters to simultaneously learn.

As previously we range the domain $f_{\mathtt{d}} \times u_{\mathtt{d}}$ and we calculate the $RRSS$ of the models parameterized with different couples $(f_{\mathtt{d}}, u_{\mathtt{d}})$ (Figures 9 and 10). This study puts in evidence a series of equivalent couples. There is an area where the $RRSS$ are lower (Figure 10) and corresponding to the curve $EC1$ of the Figure 9(a). We took for granted that the optimal couple $(f_{d_{Opt}}, u_{d_{Opt}})$ inducing the lowest $RRSS$, belongs to this curve. Therefore we sought the equation of this curve.

5.3.4. Search of the couples $(f_{d_{Opt}}, u_{d_{Opt}})$ inducing the lowest RRSS

To find the equation of the curve $EC1$ associated to the lowest $RRSS$, we sought for different values of $f_{\mathtt{d}}$ the value of $u_{\mathtt{d}}$ minimizing the $RRSS$ value. For each value of $f_{\mathtt{d}}$ we used the optimization algorithm DIRECT to find the value of $u_{\mathtt{d}}$ minimizing the objective function (11) corresponding to the average $RRSS$.

$$f_{obj}(u) = \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m}\left(\frac{(y_{ij_{obs}} - y_{ij_{pred}}(f,u))}{y_{ij_{obs}}}\right)^2\right) \tag{11}$$

As previously, in order to have several fitted values of $u_{\mathtt{d}}$ for each tested values of $f_{\mathtt{d}}$ we performed a sampling of the *Training Database*. At the end of the fitting we obtained three values of $u_{\mathtt{d}}$ for each tested values of $f_{\mathtt{d}}$ (Figure 11). Thanks to a Nadaraya-Watson kernel regression, we obtained a non-parametric relation linking $f_{d_{Opt}}$

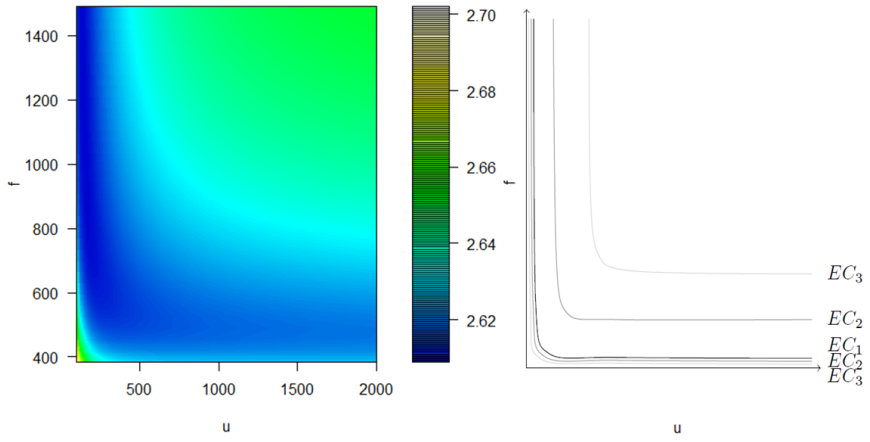

Figure 9: The value of the RRSS according to f and u (a) and the schema of the different Equivalent Couples (EC) (b)

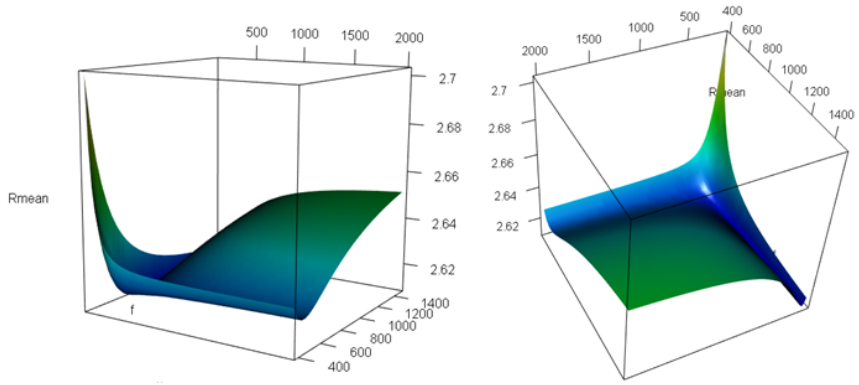

Figure 10: The $3D$ representation of the value of the RRSS according to f and u

and $u_{d_{Opt}}$ in the form of:

$$u_{d_{Opt}} = \hat{m}(f_{d_{Opt}}) + \epsilon, \tag{12}$$

where $\hat{m}$ corresponds to the Nadaraya-Watson estimator.

Knowing the relation existing between $f_{d_{Opt}}$ and $u_{d_{Opt}}$, it is possible to deduce one of these two parameters according to the value of the other one. Hence, this relation permits to further reduce the number of parameters to simultaneously learn.

5.4. Fitting of the parameters and calculation of the model accuracy

We fitted the parameters from the *Training Database* and then we tested the accuracy of the obtained model by calculating the error made on the *Test Database*.

5.4.1. Fitting of $\omega_{\mathtt{d}}$ and $f_{\mathtt{d}}$:

In order to perform several fittings from different datasets, we performed a sampling of the *Training Database*: From the 30 curves of the *Training Database* we sampled 20 curves and we fitted the parameters from the 20 sampled curves. By proceeding in this manner, we performed 30 fittings. In order to determine the values of $\omega_{\mathtt{d}}$, $r_{\mathtt{d}}$, $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$ we fitted $\omega_{\mathtt{d}}$ and $f_{\mathtt{d}}$ on the selected curves of the *Training Database* and then we deduced the values of $r_{\mathtt{d}}$ and $u_{\mathtt{d}}$.

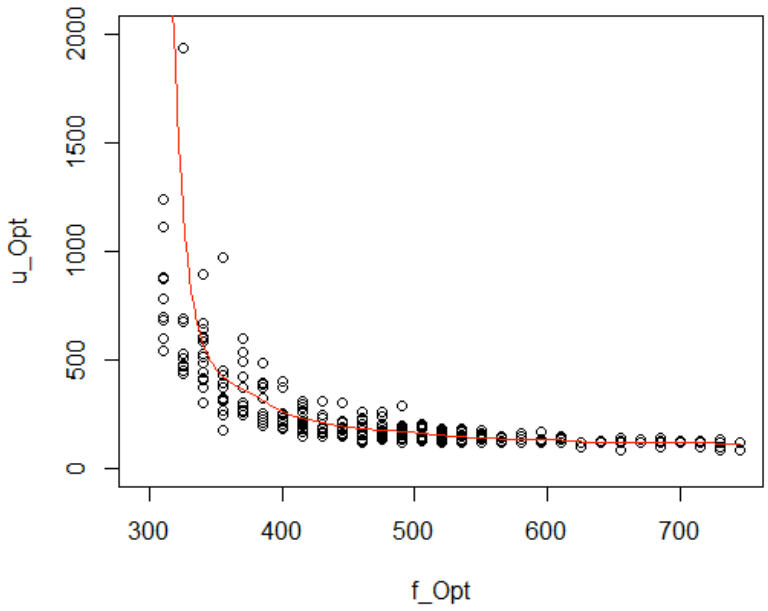

Figure 11: The Nadaraya-Watson kernel regression linking $f_{d_{Opt}}$ and $u_{d_{Opt}}$.

To optimize the parameters we used the algorithm DIRECT permitting us to find the couple $(\omega_{\mathtt{d}}, f_{\mathtt{d}})$ minimizing the objective function (13).

$$f_{obj}(\omega, f) = \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{m}\left(\frac{(y_{ij_{obs}} - y_{ij_{pred}}(\omega, f))}{y_{ij_{obs}}}\right)^2\right) \tag{13}$$

After 200 iterations we obtained the values of $\omega_{\mathtt{d}}$ and $f_{\mathtt{d}}$ and deduced the values of $r_{\mathtt{d}}$ and $u_{\mathtt{d}}$.

After the 30 fittings we obtained 30 values of the parameters $\omega_{\mathtt{d}}$, $r_{\mathtt{d}}$, $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$ (Figure 13). We calculated the mean and the Relative Standard Deviation (RSD) of each parameter (Table 3). We also looked at the fit of the model (Figure 13) and we calculated from the *Training Database* the value of the Determination Coefficient (R^2) of the obtained model (Table 3). We noticed that the Determination Coefficient is high, that mean that the model fits well the curves of the *Training Database*.

Table 3: Average and Relative Standard Deviation of the parameters and the Determination Coefficient calculated on the *Training Database*.

Parameter	Average	Relative standard deviation
ω	9.9	0.009
f	920.3	0.001
r	35.6	0.016
u	139.5	0.001
R^2	0.97	0.011

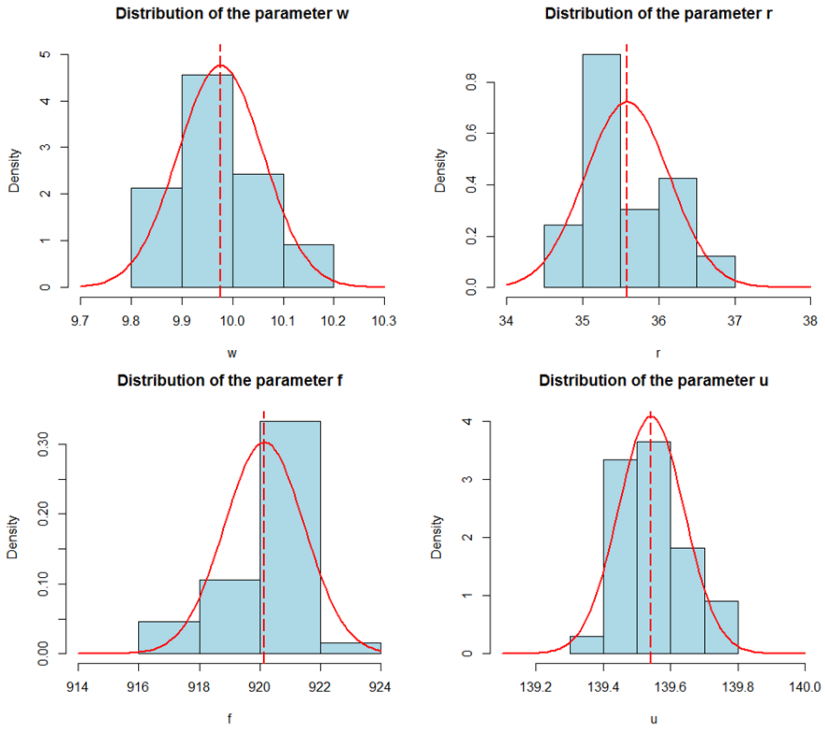

Figure 12: Distributions of the parameters obtain after the learning step

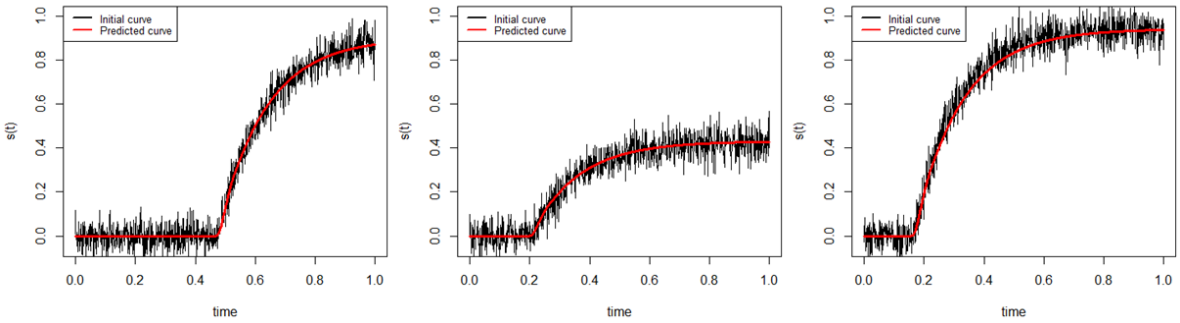

Figure 13: Examples of results

5.4.2. The model accuracy:

To validate the capability of the tool to learn parameters from noisy data, we calculated the accuracy of the model on the *Test Database*. To do that we calculated the RRSS and the Determination Coefficient associated to each curve contained in the *Test Database* and we obtained the distributions of those indicators (Figure 14). The *RRSS* is low and the Determination Coefficient is high. Hence, the model fits the curves of the *Test Database* well.

We compared the R^2 (R^2_{Gener}) and the *RRSS* ($RRSS_{Gener}$) associated to the *Generator model* - that is the model used to generate the Learning Database - and the R^2 (R^2_{Fit}) and the *RRSS* ($RRSS_{Fit}$) associated to the *Fitted Modele* (Figure 14 and Table 4). $RRSS_{Fit}$ is low and this value is very close to the value of $RRSS_{Gener}$. The R^2_{Fit} is high and this value is also very close to the value of R^2_{Gener}. Therefore, those indicators show that the fitting of the model is well done and the error associated to the adjusted

model is limited to the amount of noise and individual differences innitially integrated into the generated database.

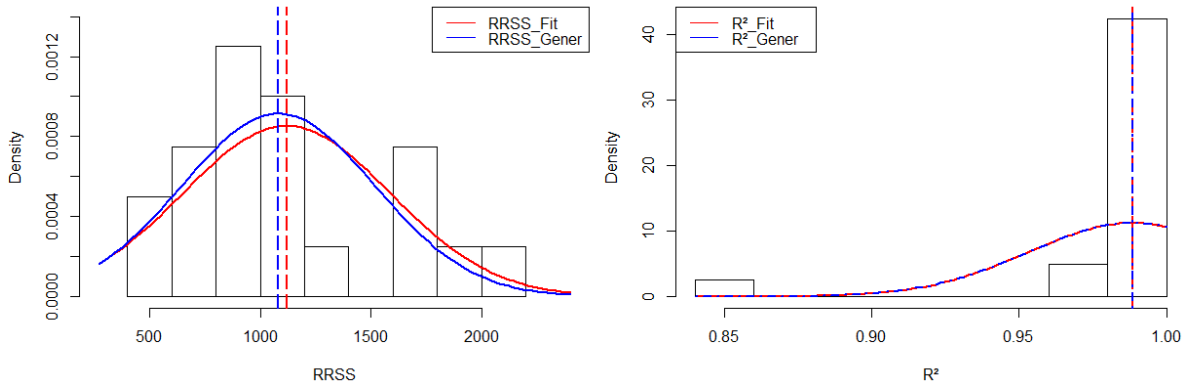

Figure 14: Distributions of the $RRSS$ and of the R^2 coefficient associated to the *Generator Model* and the *Fitted Model*.

Table 4: Comparison between the indicators associated to the *Generator Model* and the *Fitted Model*.

	$RRSS$	R^2
Generator Model	1082	0.9887
Fitted Model	1119	0.9886

6. Application of the *Statistical Learning Tool* on field data

In this Section we will present an application of our approach on field data. The used database is confidential therefore only the dimensionless *Inputs* and *Outputs* are presented in this Section.

6.1. Objectives of this application on field data

The objective of this application is to build a tool able to predict the evolution of a logistical growth process according to an initial state and intakes.

Adaptations of the base model were done to make it adapted to the evolution of the variable to predict.

6.2. Adaptation of the model

In order to mimic a logistic behavior, we chose to use as «usage» equation, Equation (4b) containing a limiter. In this equation Ld corresponds to the maximum value of the variable to predict. Experts have an idea of the maximum standard value reachable by this variable. Therefore, during the fitting, the value of Ld minimizing the error of the model is sought in a restricted range of values.

In total there are five parameters to fit: $\omega_{\mathtt{d}}$, $r_{\mathtt{d}}$, $f_{\mathtt{d}}$, $u_{\mathtt{d}}$ and $L_{\mathtt{d}}$.

6.3. The used data

The used database is made of two parts corresponding to two different individual groups monitored during two different periods (Table 5). The first group contained 8 individuals, monitored over a unit-period from $t = 0$ until $t = 1$. For this group the variable to predict was measured at $t = 0$ and at $t = 1$. The second group contained 7 individuals, monitored from $t = 0$ until $t = 2.5$. For this group the variable to predict was measured at $t = 0$, $t = 0.6$, $t = 1.52$ and at $t = 2.5$. For the two groups, intakes of each individual are recorded over each time-step of 0.16 time-unit. Therefore, for each individual, an information relative to those intakes is periodically injected in the model with a time step of 0.16.

The dataset concerning the first group constitutes our *Training Database* and the dataset concerning the second group constitutes our *Test Database*. The objective is to fit the parameters on the *Training Database* and test the accuracy of the *Fitted Modele* on the *Test Database*.

Table 5: Descritpion of the used data.

	First group	Second group
Number of individuals	8	7
Output measured at	$t = 0$ $t = 1$	$t = 0$ $t = 0.60$ $t = 1.52$ $t = 2.50$
Time step of the *Entries* injections	$\Delta t_{In} = 0.16$	$\Delta t_{In} = 0.16$

6.4. Study of the relations existing between the model parameters

As in Section 5.3 we analyzed the relations existing between the model parameters by applying the same methodology on the field *Training Database*.

6.4.1. Study of the relations existing between $\omega_{\mathtt{d}}$ and $r_{\mathtt{d}}$

As in Section 5.3.1 we sought the relation existing between $\omega_{\mathtt{d}}$ and $r_{\mathtt{d}}$. We sought for several values of $\omega_{\mathtt{d}}$, the value of $r_{\mathtt{d}}$ minimizing the error of the model on the *Training Database*. To do that we used the algorithm DIRECT.

Thanks to a Nadaraya-Watson kernel regression, we obtained a non-parametric regression linking $\omega_{d_{Opt}}$ and $r_{d_{Opt}}$ (Figure 15).

Knowing the relation existing between those two parameters, it is possible to deduce one according to the value of the other one.

6.4.2. Study of the relations existing between $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$

As in Section 5.3.3 we sought the relation existing between $f_{\mathtt{d}}$ and $u_{\mathtt{d}}$. We sought for several values of $f_{\mathtt{d}}$, the value of $u_{\mathtt{d}}$ minimizing the error of the model on the *Training Database*. To do that we used the algorithm DIRECT.

As prevously, thanks to a Nadaraya-Watson kernel regression, we obtained a non-parametric regression linking $f_{d_{Opt}}$ and $u_{d_{Opt}}$ (Figure 15).

Knowing the relation existing between those two parameters, it is possible to deduce one according to the value of the other one.

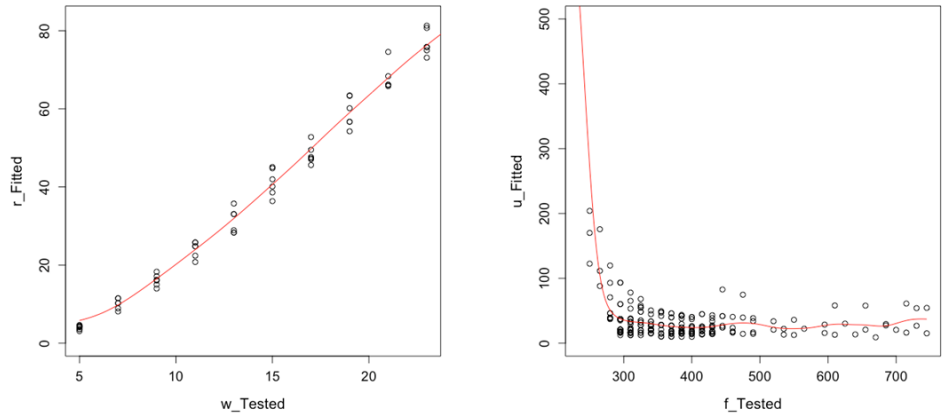

Figure 15: The Nadaraya-Watson kernel regression linking the couples ($\omega_{d_{Opt}}$, $r_{d_{Opt}}$) (left) and the one linking the couples ($f_{d_{Opt}}$,$u_{d_{Opt}}$).

6.5. Fitting of the parameters

We only fitted $\omega_{\mathtt{d}}$, $f_{\mathtt{d}}$ and $L_{\mathtt{d}}$ and deduced the values of $r_{\mathtt{d}}$ and $u_{\mathtt{d}}$.

The parameters are fitted on the *Training Database* by minimizing the difference between the simulated and the real *Outputs* at the instant $t = 1$. To fit the parameters we used the algorithm DIRECT minimizing the following objective function:

$$f_{obj}(\omega_{\mathtt{d}}, r_{\mathtt{d}}, f_{\mathtt{d}}, u_{\mathtt{d}}, L_{\mathtt{d}}) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{(s_{i_{obs}}(1) - s_{j_{pred}}(1))}{s_{i_{obs}}(1)}\right)^2, \tag{14}$$

where n is the number of individuals and $O_{i_{obs}}(1)$ and $O_{i_{pred}}(1)$ correspond respectively to the value of the observed and the predicted *Output* value for the i^{th} individual at $t = 1$.

To test the stability of the set of values of the parameters minimizing the error of the model on the *Training Database*, we performed several fittings. To do that we sampled the *Training Database*: before each fitting we randomly selected 7 individuals from 8 and we fitted the parameters on the data associated to the selected individuals. Therefore we performed 8 fittings and we obtained 8 sets of values of $(\omega_{\mathtt{d}}, r_{\mathtt{d}}, f_{\mathtt{d}}, u_{\mathtt{d}}, L_{\mathtt{d}})$.

6.6. Results

We calculated the average and the Relative Standard Deviation (*RSD*) of each parameter (Table 6). The *RSD* of each parameter is low. This means that our fitting method permits to identify one set containing the parameter values minimizing the error associated to the *Fitted Modele*. The existence of a single optimal set of values of $(\omega_{\mathtt{d}}, r_{\mathtt{d}}, f_{\mathtt{d}}, u_{\mathtt{d}}, L_{\mathtt{d}})$ attests of the identifiability of the model.

We parametrized the model with the average values of the parameters.

We calculated the error associated to the model on the *Training Database*. To do that we calculated the Average Relative Error (*ARE*) between the measured and the predicted value of the *Output* at the instant $t = 1$ (15).

$$ARE(t) = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\left(\frac{(s_{i_{obs}}(t) - s_{j_{pred}}(1))}{s_{i_{obs}}(t)}\right)^2} \tag{15}$$

Table 6: Average values and Relative Standard Deviation (RSD) of the adjusted parameters. ARE calculated at the instant $t = 1$ on the *Training Database*.

Parameter	*Mean*	*RSD*
ω	9.24	0.079
r	17.91	0.14
f	707.01	0.36
u	21.49	0.17
$L_{\mathtt{d}}$	1.70	0.009
$ARE(1)$ (%)	1.83	0.013

The ARE value calculated at the instant $t = 1$, on the *Training Database* is worth 1.83%. It is a satisfying result but the accuracy of the model has to be calculated on a *Test Database* to assert that the model does not overfit the training data.

To do that we calculated the ARE on the *Test Database* at the instants $t = 0.6$, $t = 1.52$ and $t = 2.5$ (Table 7 and Figure 16).

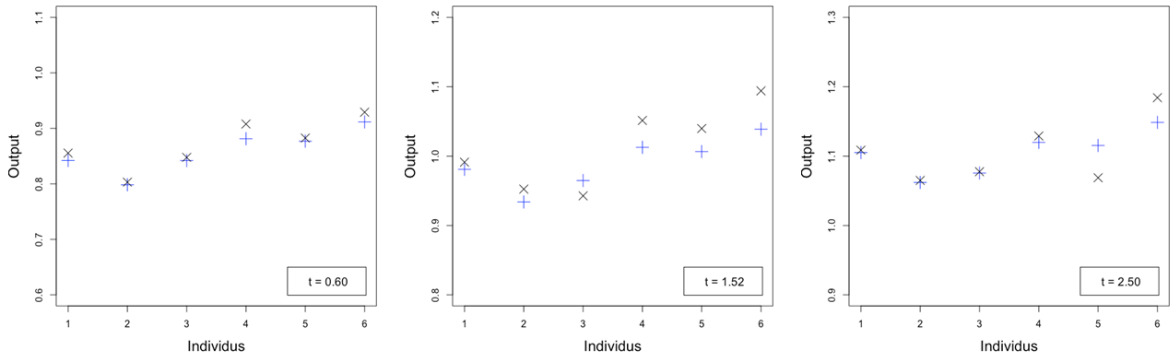

Figure 16: Difference obtained between measured (+) and the predicted (×) values of the *Output* variable at different instant t for the individuals of the *Test Database*.

Table 7: Average Relative Error (ARE) calculated on the *Test Database* at different instants.

t	0.6	1.52	2.5
ARE(t) (%)	1.3	2.9	1.5

6.7. Discussion of the results

The error associated to the model is low on the *Test Database*. We also noticed that the error made before and beyond the $t = 1$ remains low. This result shows that the model is able to learn dynamics on a certain period and to remain pertinent on a period 2.5 times longer than the training period. Therefore the model has an interpolation and an extrapolation capability.

7. Comparison with existing growth models

According to Vázquez-Cruz *et al.* (2014) and Guzmán-Cruz *et al.* (2011), among the existing methods to simulate and predict logistical growth processes, two types of models are distinguished: The Phenomenological Models corresponding to the «Black Box» models, and the Mechanistic Models corresponding to the «White Box» models. In this Section we will compare some models belonging to these two types of models with the Biomimetic Model introduced in this paper.

7.1. The Phenomenological Models

As defined in Vázquez-Cruz *et al.* (2014), the Phenomenological Models are «Black Box» models corresponding to direct descriptions of the data. This type of models comprise Linear, Multiple Linear and Nonlinear Regressions, but also Logistic Models and Neuronal Networks. We chose to compare our Biomimetic Model with Logistic Models and Neuronal Networks.

7.1.1. Comparison between the Biomimetic Growth Model with Classical Logistic Growth Models

The models of Gompertz (1825),

$$\frac{dN(t)}{dt} = a_G.N(t).\ln\left(\frac{K_G}{N(t)}\right), \tag{16}$$

and Verhulst (1838),

$$\frac{dN(t)}{dt} = a_V.N(t).\left(1 - \frac{N(t)}{K_V}\right), \tag{17}$$

are two models usually used to model growth processes (See for example: Winsor (1932), Sakomura *et al.* (2005), Buyse *et al.* (2004), Robertson (1916), Robertson (1923) and Román-Román and Torres-Ruiz (2012)). The models built by Gompertz and Verhulst are based on the hypothesis that growth processes are bounded respectively by KG and KV.

We fitted the parameters of the Gompertz's and the Verhulst's models on our *Training Database* by using the same optimization algorithm that we used to fit the Biomimetic Model. As the Biomimetic one, those two classical models are fitted by minimizing $ARE(1)$.

On the *Training Database*, the Biomimetic Model is associated to the highest accuracy (Table 8), but the accuracy of the different models is globally similar on this dataset.

To test and compare the accuracy of the different models we calculated on the *Test Database* the Average Relative Accuracy, ARA (18) at different s t.
To do that we used the three parametrized model to generate the growth curve of each individual contained in the *Test Database* and we compared the measured and the predicted values at $t = 0.6$, $t = 1.52$, $t = 2.5$.
The results contained in Table 8 and the curves of Figures 17 and 18 show that the

curves generated from the Gompertz's model featured a too quick slow-down. However, the Verhulst's model is associated to a good accuracy over the whole studied period.

The similarity between the results coming from the model of Verhulst and the Biomimetic Growth Model was expected. Indeed, an equation assimilable to the Verhulst's equation is integrated in our model (See Section 3.2). The real advantage of the biomimetic growth model is its Data Assimilation capability. Indeed the Verhulst's equation only takes into account the initial conditions of the system under study. Our model takes into account the initial conditions but it also integrates *Inputs* all along the studied period. The integration of additional information appears to allow the refining of the results and the increase in the accuracy of the model.

$$ARA = 1 - ARE \tag{18}$$

Table 8: Parameters values and $ARA(1)$ calculated on the *Training Database*.

Model	a	K	ARA(1)
Gompertz	$a_G = 0.412$	$K_G = 0.563$	0.978
Verhulst	$a_V = 0.411$	$K_V = 1.563$	0.979
Biomimetic			0.981

Table 9: The ARA calculated on the *Test Database* at different instants associated to different models.

t	Verhulst	Gompertz	Biomimetic
0.6	0.985	0.980	0.986
1.52	0.968	0.937	0.971
2.5	0.979	0.923	0.985

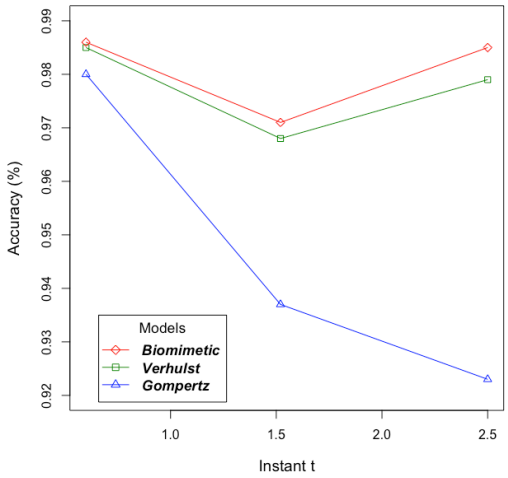

Figure 17: Predicted growth curves of each individual contained in the *Test Database* from different models.

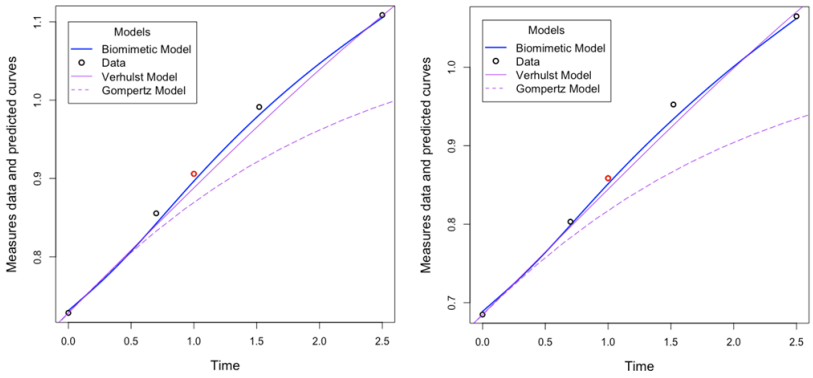

Figure 18: Plot of the predicted growth curves of two individuals contained in the *Test Database* with the different models.

7.1.2. Comparison between the Biomimetic Growth Model and Neural Networks

We applied different Neural Networks on our *Training Database* in order to compare the capacities of this kind of ML tools and the ones of our Biomimetic Growth Model. We tested six Neural Networks having different numbers of nodes and hidden layers (Table 10) and taken as *Inputs* the initial state of each individual and their periodically recorded intakes.

Table 10: The *ARA* calculated on the *Train Database* (ARA_{Train}), and on the *Test Database* (ARA_{Test}), at $t = 1$, with different Neural Networks. The Neural Network (k_1,...,k_i,...,k_n) corresponds to a NN containing n hidden layers and the i^{th} hidden layer contains k_i nodes.

Structure	$ARA_{Train}(1)$ (%)	$ARA_{Test}(1)$ (%)
(4)	99.9	78.8
$(4, 3)$	99.8	90.5
$(6, 5)$	99.7	93.4
$(4, 6, 6, 3)$	99.9	94.8
$(5, 7, 7, 7, 4)$	99.8	95.3
$(5, 9, 9, 9, 5)$	99.9	93

We fitted each tested Neural Network on our *Training Database* by using the R-function *neuralnet* developped by Fritsch *et al.* (2012), and we calculated the accuracy of those Neural Networks on the *Training* and on the *Test Database*.

The results contained into Table 10 show that all the tested NN fit very well the curves of the *Training Database*, but less the curves of the *Test Database*. So those NN overfit the training curves and even more when the structure of the studied NN is made of too much or too little nodes and hidden layers. Indeed, we noticed that the accuracy of the NN on the *Test Database* increase until a certain number of nodes and hidden layers and then decrease when the complexity of the structure still increases. The higher accuracy value is reached by using a NN containing 5 hidden layers but it is less high than the one obtained by using the Biomimetic Model (Table 10).

Therefore in the framework of the study of a globally well known process with few available data, the NN overfit the training curves and remain less accurate than the built Biomimetic Model.

Nevertheless the accuracy of those ML tools is satisfying and the real advantage of the Biomimetic Model over the NN does not correspond to its prediction capability. Indeed, as the Biomimetic Model, the studied NN are fitted only from the value of the *Output* at $t = 1$. In this case, the fitted classical NN can only be used to predict the *Output* at $t = 1$. Hence, the NN do not have interpolation or extrapolation capacities, contrary to the Biomimetic Model.

7.2. The Mechanistic Growth Models

Some Mechanistic specific and complex Growth Models have been developed by Bastianelli *et al.* (1996), Mach and Kristkova (2010), Brun-Lafleur *et al.* (2013) and Zúñiga *et al.* (2014). Those models integrate numerous *Inputs* and some of them are not available in our study. Hence, those models can not be applied on our database. Therefore we just compared the structure, the functioning and the objectives of the Mechanistic Explanatory Models and the Biomimetic Model.

As for the Biomimetic Model the goal of these kind of models is to integrate existing knowledge in a mathematical model, but more with the purpose to build realistic and explanatory model than to perform Data Assimilation. Indeed, as it is said in Vázquez-Cruz *et al.* (2014), Tedeschi *et al.* (2005), Bastianelli and Sauvant (1997) and Beever *et al.* (1991), those Mechanistic Growth Models remain abstractions of the reality, but those models are used to perform quantitative analysis, in the framework of very specific process studies. So the construction of those models is generally focused on the biological meaning of the global model. This objective explains the need to take into account the dynamics of the system under study with higher precision.

Therefore the construction of the explanatory mechanistic models takes time, needs a lot of zootechnical knowledge and results in complex models. As it is explained in Wallach *et al.* (2001), Bastianelli and Sauvant (1997) and Emmans (1995), those models contain a large number of unknown parameters and take into account a lot of factors, forcing the user to enter a large number of *Input* values sometimes difficult or costly to obtain. Hence, the complex structure of those models makes the Mechanistic Realistic Model not really suitable to perform Data Fitting and Data Assimilation.

Therefore, the structure of those two types of models are very different but pertinent in the light of the respective objectives of those modeling methods.

8. Conclusion

To conclude, we can say that we built a *Biomimetic Statistical Learning Tool* based on a PDE system, embarking the mathematical expression of biological determinants. The performed tests and the application on field data showed that this tool is associated to a satisfying accuracy.

The comparison of our *Biomimetic Model* with existing models showed that the structure and the functionning of the tested models are very different but appropriate and suitable for their respective objectives and fields of application. In the context of developing tools to simulate and predict biological phenomena from very few data, the built *Biomimetic Statistical Learning Tool* is the most accurate. But this tool really stands out from the existing tools by an interpolation and an extrapolation capacities

and also by its flexibility and its Data Assimilation capability.

Nevertheless the results coming from the Biomimetic Model was obtained from a certain number of hypothesis. Some Model Selection methods could be applied in order to select the *Mathematical Model* structure permitting to obtain a more satisfying model in terms of *ARE* and number of parameters to learn.

Acknowledgement

The authors are very grateful to D. Causeur, G. Durrieu and E. Fokoué for fruitful discussions related to this article.

References

Ailliot, P., E. Frénod, and V. Monbet (2006). "Long term object drift in the ocean with tide and wind." In: *SIAM Journal on Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal* 5 (2), pp. 514 –531. URL: https://hal.archives-ouvertes.fr/hal-00129093.

Barrett, D., M. Hill, L. Hutley, J. Beringer, J. H. Xu, G. Cook, J. Carter, and R. J. Williams (2005). "Prospects for improving savanna biophysical models by using multiple-constraints model-data assimilation methods". In: *Australian Journal of Botany* 53(7). DOI: 10.1071/BT04139.

Barrett, D. J. (2002). "Steady state turnover time of carbon in the Australian terrestrial biosphere". In: *Global Biogeochemical Cycles* 16.4, pp. 55–1–55–21. DOI: 10.1029/2002GB001860. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2002GB001860. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2002GB001860.

Basarab, J., M. Price, J. L. Aalhus, E. Okine, W. M. Snelling, and K. L. Lyle (2003). "Residual Feed intake and body composition in young growing cattle". In: *Canadian Journal of Animal Science* 83, pp. 189–204. DOI: 10.4141/A02-065.

Bastianelli, D. and D. Sauvant (1997). "Modelling the mechanisms of pig growth." In: *Livestock Production Science*.

Bastianelli, D., D. Sauvant, and A. Rerat (1996). "Mathematical modeling of digestion and nutrient absorption in pigs." In: *Journal of animal science*.

Beever, D. E., A. J. Rook, J. France, M. S. Dhanoa, and M. Gill (1991). "A review of empirical and mechanistic models of lactational performance by the dairy cow". In: *Livestock Production Science* 29.2, pp. 115 –130. ISSN: 0301-6226. DOI: https://doi.org/10.1016/0301-6226(91)90061-T. URL: http://www.sciencedirect.com/science/article/pii/030162269190061T.

Brun-Lafleur, L., E. Cutullic, P. Faverdin, L. Delaby, and C. Disenhaus (2013). "An individual reproduction model sensitive to milk yield and body condition in Holstein dairy cows". In: *Animal : an international journal of animal bioscience* 7, pp. 1–12. DOI: 10.1017/S1751731113000335.

Buyse, J., B. Geypens, R. D. Malheiros, V. M. Moraes, Q. Swennen, and E. Decuypere (2004). "Assessment of age-related glucose oxidation rates of broiler chickens by using stable isotopes". In: *Life sciences* 75.18, pp. 2245–2255.

Büchel, S. and A. Sundrum (2014). "Short communication: Decrease in rumination time as an indicator of the onset of calving". In: *Journal of Dairy Science* 97.5, pp. 3120 –3127. ISSN: 0022-0302. DOI: https://doi.org/10.3168/jds.2013-7613. URL: http://www.sciencedirect.com/science/article/pii/S0022030214001684.

Courant, R., K. Friedrichs, and H. Lewy (1928). "Über die partiellen Differenzengleichungen der mathematischen Physik". In: *Mathematische annalen* 100.1, pp. 32–74.

Crosson, W. L., C. A. Laymon, R. Inguva, and M. P. Schamschula (2002). "Assimilating remote sensing data in a surface flux–soil moisture model". In: *Hydrological Processes* 16, pp. 1645–1662. DOI: 10.1002/hyp.1051.

Domingos, P. (2012). "A Few Useful Things to Know About Machine Learning". In: *Commun. ACM* 55, 78–87. DOI: 10.1145/2347736.2347755.

Dumas, A., J. Dijkstra, and J. France (2008). "Mathematical modelling in animal nutrition: A centenary review". In: *Journal of Agricultural Science 146 (2008) 2* 146. DOI: 10.1017/S0021859608007703.

Dumpala, S. H., R. Chakraborty, and S. K. Kopparapu (2017). *k-FFNN: A priori knowledge infused Feed-forward Neural Networks*. arXiv: 1704.07055 [cs.LG].

Emmans, G. C. (1995). "Problems in applying models in practice". In: *Publication-European Association For Animal Production* 78, pp. 223–223.

Enright, W. (1989). "The Numerical Analysis of Ordinary Differential Equations: Runge Kutta and General Linear Methods". In: *SIAM Review* 31.4, pp. 693–693. DOI: 10.1137/1031147. eprint: https://doi.org/10.1137/1031147. URL: https://doi.org/10.1137/1031147.

Finkel, D. E. (2003). *DIRECT Optimization Algorithm*. North Carolina State University.

Flourent, Hélène (2019). "Study of the ranges of values of a Biomimetic Statistical Learning Tool parameters". Working paper. URL: https://hal.archives-ouvertes.fr/hal-02067374.

Fritsch, Stefan, Frauke Guenther, and following earlier work by Marc Suling (2012). *neuralnet: Training of neural networks*. R package version 1.32. URL: http://CRAN.R-project.org/package=neuralnet.

Frénod, E. (2017). "A PDE-like Toy-Model of Territory Working". In: *Understanding Interactions in Complex Systems - Toward a Science of Interaction*. Understanding Interactions in Complex Systems - Toward a Science of Interaction. Cambridge Scholar Publishing, pp. 37–47. URL: https://hal.archives-ouvertes.fr/hal-00817522.

Gompertz, B. (1825). "XXIV. On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life contingencies. In a letter to Francis Baily, Esq. FRS &c". In: *Philosophical transactions of the Royal Society of London* 115, pp. 513–583.

Gorczyca, M. T., H. F. M. Milan, A. S. C. Maia, and K. G. Gebremedhin (2018). "Machine learning algorithms to predict core, skin, and hair-coat temperatures of piglets". In: *Computers and Electronics in Agriculture* 151, pp. 286 –294. ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2018.06.028. URL: http://www.sciencedirect.com/science/article/pii/S0168169918303612.

Guzmán-Cruz, R., R. Castaneda-Miranda, J. Garcia-Escalante, L. Solis-Sánchez, D. Alaniz-Lumbreras, J. Mendoza-Jasso, A. Lara-Herrera, G. Ornelas-Vargas, E. Gonzalez-

Ramirez, and R. Montoya-Zamora (2011). "Evolutionary Algorithms in Modelling of Biosystems". In: ISBN: 978-953-307-171-8. DOI: 10.5772/16231.

Holman, A., J. Thompson, J. E. Routly, J. Cameron, D. N. Jones, D. Grove-White, R. F. Smith, and H. Dobson (2011). "Comparison of oestrus detection methods in dairy cattle". In: *Veterinary Record* 169.2, pp. 47–47. ISSN: 0042-4900. DOI: 10.1136/vr.d2344. eprint: https://veterinaryrecord.bmj.com/content/169/2/47.full.pdf. URL: https://veterinaryrecord.bmj.com/content/169/2/47.

Hubbard, T. and A. Reinhardt (1998). "Using neural networks for prediction of the subcellular location of proteins". In: *Nucleic Acids Research* 26.9, pp. 2230–2236. ISSN: 0305-1048. DOI: 10.1093/nar/26.9.2230. eprint: http://oup.prod.sis.lan/nar/article-pdf/26/9/2230/9471729/26-9-2230.pdf. URL: https://dx.doi.org/10.1093/nar/26.9.2230.

Ingalls, B. (2019). "Mathematical Modelling in Systems Biology: An Introduction". In:

Ip, R. H. L., L. M. Ang, K. P. Seng, J. C. B., and J. E. Pratley (2018). "Big data and machine learning for crop protection". In: *Computers and Electronics in Agriculture* 151, pp. 376 –383. ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2018.06.008. URL: http://www.sciencedirect.com/science/article/pii/S0168169917314588.

Jemila, J. S. and S. S. Priyadharsini (2018). "A Sensor-Based Forage Monitoring of Grazing Cattle in Dairy Farming". In: *International Journal on Smart Sensing and Intelligent Systems* 11, pp. 1–9. DOI: 10.21307/ijssis-2018-014.

Johnson, S. G. (2008). "The NLopt nonlinear-optimization package".

Kim, G. and A. P. Barros (2002). "Space–time characterization of soil moisture from passive microwave remotely sensed imagery and ancillary data". In: *Remote Sensing of Environment* 81.2, pp. 393 –403. ISSN: 0034-4257. DOI: https://doi.org/10.1016/S0034-4257(02)00014-7. URL: http://www.sciencedirect.com/science/article/pii/S0034425702000147.

Locke, J. C. W., A. J. Millar, and M. S. Turner (2005). "Modelling genetic networks with noisy and varied experimental data: the circadian clock in Arabidopsis thaliana." In: *Journal of theoretical biology* 234 3, pp. 383–93.

Ma, C., H. H. Zhang, and X. Wang (2014). "Machine learning for Big Data analytics in plants". In: *Trends in Plant Science* 19.12, pp. 798 –808. ISSN: 1360-1385. DOI: https://doi.org/10.1016/j.tplants.2014.08.004. URL: http://www.sciencedirect.com/science/article/pii/S1360138514002192.

Mach, J. and Z. Kristkova (2010). "Modelling The Cattle Breeding Production in the Czech Republic". In: *AGRIS on-line Papers in Economics and Informatics* 2.

Mackay, D. S., S. Samanta, R. R. Nemani, and L. E. Band (2003). "Multi-objective parameter estimation for simulating canopy transpiration in forested watersheds". In: *Journal of Hydrology* 277.3, pp. 230 –247. ISSN: 0022-1694. DOI: https://doi.org/10.1016/S0022-1694(03)00130-6. URL: http://www.sciencedirect.com/science/article/pii/S0022169403001306.

Martin, O. and D. Sauvant (2010a). "A teleonomic model describing performance (body, milk and intake) during growth and over repeated reproductive cycles throughout the lifespan of dairy cattle. 1. Trajectories of life function priorities and genetic scaling." In: *Animal*.

— (2010b). "A teleonomic model describing performance (body, milk and intake) during growth and over repeated reproductive cycles throughout the lifespan of dairy cattle.

2. Voluntary intake and energy partitioning". In: *Animal* 4.12, 2048ñ2056. DOI: 10.1017/S1751731110001369.

McPhee, M. (2009). "Mathematical modelling in agricultural systems : A case study of modelling fat deposition in beef cattle for research and industry". In:

Miekley, Bettina, Imke Traulsen, and Joachim Krieter (2012). "Detection of mastitis and lameness in dairy cows using wavelet analysis". In: *Livestock Science* 148.3, pp. 227 –236. ISSN: 1871-1413. DOI: https://doi.org/10.1016/j.livsci.2012.06.010. URL: http://www.sciencedirect.com/science/article/pii/S187114131200220X.

Nadaraya, E. A. (1964). "On estimating regression". In: *Theory of Probability & Its Applications* 9.1, pp. 141–142.

Nesetrilova, H. (2005). "Multiphasic growth models for cattle". In: *Czech Journal of Animal Science* 50, pp. 347–354. DOI: 10.17221/4176-CJAS.

Nkrumah, J. D., J. Basarab, Z. Wang, C. Li, M. Price, E. Okine, D. H. Crews, and S. S. Moore (2007). "Genetic and phenotypic relationships of feed intake and measures of efficiency with growth and carcass merit of beef cattle". In: *Journal of animal science* 85, pp. 2711–20. DOI: 10.2527/jas.2006-767.

Puillet, L., O. Martin, D. Sauvant, and M. Tichit (2011). "Introducing efficiency into the analysis of individual lifetime performance variability: a key to assess herd management". In: *animal* 5.1, pp. 123–133. DOI: 10.1017/S175173111000162X. URL: https://hal.archives-ouvertes.fr/hal-01137029.

Qi, Y., Z. Bar-Joseph, and J. Klein-Seetharaman (2006). "Evaluation of different biological data and computational classification methods for use in protein interaction prediction". In: *Proteins: Structure, Function, and Bioinformatics* 63.3, pp. 490–500. DOI: 10.1002/prot.20865. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/prot.20865. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.20865.

R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria. URL: http://www.R-project.org/.

Rayner, P. J., M. Scholze, W. Knorr, T. Kaminski, R. Giering, and H. Widmann (2005). "Two decades of terrestrial carbon fluxes from a carbon cycle data assimilation system (CCDAS)". In: *Global Biogeochemical Cycles* 19.2. DOI: 10.1029/2004GB002254. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2004GB002254. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2004GB002254.

Renzullo, L. J., D. J. Barrett, A. S. Marks, M. J. Hill, J. P. Guerschman, Q. Mu, and S. W. Running (2008). "Multi-sensor model-data fusion for estimation of hydrologic and energy flux parameters". In: *Remote Sensing of Environment* 112.4. Remote Sensing Data Assimilation Special Issue, pp. 1306 –1319. ISSN: 0034-4257. DOI: https://doi.org/10.1016/j.rse.2007.06.022. URL: http://www.sciencedirect.com/science/article/pii/S0034425707003227.

Robertson, T. B. (1916). "Experimental studies on growth II. The normal growth of the white mouse". In: *Journal of Biological Chemistry* 24.3, pp. 363–383.

— (1923). *The chemical basis of growth and senescence*. JB Lippincott Company.

Román-Román, P. and F. Torres-Ruiz (2012). "Modelling logistic growth by a new diffusion process: Application to biological systems". In: *Biosystems* 110.1, pp. 9–21.

Rousseau, A. and M. Nodet (2013). "Modélisation mathématique et assimilation de données pour les sciences de l'environnement". In: *Bulletin de l'APMED* 505, pp. 467–472.

Sacks, W. J., D. S. Schimel, R. K. Monson, and B. H. Braswell (2006). "Model-data synthesis of diurnal and seasonal CO2 fluxes at Niwot Ridge, Colorado". In: *Global Change Biology* 12.2, pp. 240–259. DOI: 10.1111/j.1365-2486.2005.01059.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1365-2486.2005.01059.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1365-2486.2005.01059.x.

Sacks, W. J., D. S. Schimel, and R. K. Monson (2007). "Coupling between carbon cycling and climate in a high-elevation, subalpine forest: a model-data fusion analysis". en. In: *Oecologia* 151.1, pp. 54–68. ISSN: 0029-8549, 1432-1939. DOI: 10.1007/s00442-006-0565-2. URL: http://link.springer.com/10.1007/s00442-006-0565-2 (visited on 11/22/2018).

Sakomura, N. K., F. A. Longo, E. O. Oviedo-Rondon, C. Boa-Viagem, and A. Ferraudo (2005). "Modeling energy utilization and growth parameter description for broiler chickens". In: *Poultry Science* 84.9, pp. 1363–1369.

Shavlik, J., L. Hunter, and D. Searls (1995). "Introduction". In: *Machine Learning* 21.1, pp. 5–9. ISSN: 1573-0565. DOI: 10.1007/BF00993376. URL: https://doi.org/10.1007/BF00993376.

Simmons, A. J. and A. Hollingsworth (2002). "Some aspects of the improvement in skill of numerical weather prediction". In: *Quarterly Journal of the Royal Meteorological Society* 128.580, pp. 647–677. DOI: 10.1256/003590002321042135. URL: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1256/003590002321042135.

Soetaert, K., T. Petzoldt, and R. Woodrow Setzer (2010). "Solving Differential Equations in R: Package deSolve". In: *Journal of Statistical Software* 33.9, pp. 1–25. ISSN: 1548-7660. DOI: 10.18637/jss.v033.i09. URL: http://www.jstatsoft.org/v33/i09.

Tan, A. C. and D. Gilbert (2003). "An empirical comparison of supervised machine learning techniques in bioinformatics". In: *Proceedings of the First Asia-Pacific bioinformatics conference on Bioinformatics 2003-Volume 19*. Australian Computer Society, Inc., pp. 219–222.

Tedeschi, L. O., D. G. Fox, R. D. Sainz, L. G. Barioni, S. R. de Medeiros, and C. Boin (2005). "Mathematical models in ruminant nutrition". en. In: *Scientia Agricola* 62, pp. 76 –91. ISSN: 0103-9016. URL: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-90162005000100015&nrm=iso.

Tol, R. van der and A. van der Kamp (2010). "Time series analysis of live weight as health indicator". In: *Proceedings of the first North American conference precision dairy management*, pp. 230–231.

Valletta, J. J., C. Torney, M. Kings, A. Thornton, and J. Madden (2017). "Applications of machine learning in animal behaviour studies". In: *Animal Behaviour* 124, pp. 203 –220. ISSN: 0003-3472. DOI: https://doi.org/10.1016/j.anbehav.2016.12.005. URL: http://www.sciencedirect.com/science/article/pii/S0003347216303360.

Vázquez-Cruz, M. A., A. Espinosa-Calderón, A. R. Jiménez-Sánchez, and R. Guzmán-Cruz (2014). "Mathematical Modeling of Biosystems". In: *Biosystems Engineering: Biofactories for Food Production in the Century XXI*. Cham: Springer International Publishing, pp. 51–76. DOI: 10.1007/978-3-319-03880-3_2. URL: https://doi.org/10.1007/978-3-319-03880-3_2.

Verhulst, P. F. (1838). "Notice sur la loi que la population suit dans son accroissement". In: *Corresp. Math. Phys.* 10, pp. 113–126. URL: https://ci.nii.ac.jp/naid/10015246307/en/.

Wallach, D., B. Goffinet, J. E. Bergez, P. Debaeke, D. Leenhardt, and J. N. Aubertot (2001). "Parameter estimation for crop models". In: *Agronomy journal* 93.4, pp. 757–766.

Wang, L., H. Zhang, K. C. L. Wong, H. Liu, and P. Shi (2010). "Physiological-model-constrained noninvasive reconstruction of volumetric myocardial transmembrane potentials". In: *IEEE Transactions on Biomedical Engineering* 57.2, pp. 296–315.

Watson, G. S. (1964). "Smooth regression analysis". In: *Sankhyā: The Indian Journal of Statistics, Series A*, pp. 359–372.

Weisstein, E. W. (2014). "Courant-friedrichs-lewy condition". In: *Wolfram MathWorld–A Wolfram Web Resource.*

Winsor, C. P. (1932). "The Gompertz Curve as a Growth Curve". In: *Proceedings of the National Academy of Sciences* 18.1, pp. 1–8. ISSN: 0027-8424. DOI: 10.1073/pnas.18.1.1. eprint: https://www.pnas.org/content/18/1/1.full.pdf. URL: https://www.pnas.org/content/18/1/1.

Zúñiga, E. C. T., I. L. L. Cruz, and A. R. García (2014). "Parameter estimation for crop growth model using evolutionary and bio-inspired algorithms". In: *Applied Soft Computing* 23, pp. 474 –482. ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2014.06.023. URL: http://www.sciencedirect.com/science/article/pii/S156849461400297X.