Machine learning for agri-food processes: learning from data, human knowledge and interactions

Nathalie Mejean Perrot, Alberto Tonda, Nadia Boukhelifa, Ilaria Brunetti, Anastasia Bezerianos, Evelyne Lutton

Chapter Title:
Machine learning for agri-food processes: learning from data, human knowledge and interactions[1]*

Nathalie Mejean Perrot[1], Alberto Tonda[1], Nadia Boukhelifa[1], Ilaria Brunetti[2], Anastasia Bezerianos[3], Evelyne Lutton[1]

[1] Unité MIA-Paris, AgroParisTech, INRAE, Université Paris-Saclay, France. email: nom.prenom@inrae.fr

[2] UR 767 ECODEVELOPPEMENT, INRAE; UMR 8568 CIRED, CNRS, CIRAD, AgroParisTech, France. email: ilariabrun@gmail.com

[3]Université Paris-Saclay, CNRS, INRIA, France. email: anastasia.bezerianos@lri.fr

Abstract:

This chapter presents three examples of data-based machine learning on time series. The common denominator of these case studies is the sparseness of data, making machine learning results fragile and inaccurate. We show how human expertise can be effectively mobilized for building useful systems, for instance useful decision support systems, able to better meet the needs of the agri-food chain. The design and analysis of different features of machine learning coupled with human knowledge enables us to sketch future human-centered machine learning systems.

This approach is very relevant for the modeling of agri-food systems, because human expertise, skills and know-how are rich and numerous, but often implicit, data are heterogeneous -- big and sparse -- and processes are complex and deeply conditioned by human needs and interactions.

[1]*corresponding author: nathalie.mejean@inrae.fr

Keywords:

Machine learning, interactive machine learning, knowledge integration, visualisation, interactive systems, stochastic methods, agri-food processes, expertise formalisation.

Introduction

Artificial Intelligence (AI) techniques are now commonly used in the agri-food domain (Kavalenko (2020)) and are expected to be more and more used. From a recent study of Accenture Research, AI has the potential to noticeably increase profitability of industries in the next decades (Hunkefer, 2017). While improving productivity, AI is also and above all described as a means to meet the urgent requirements of the agroecological transition: reduce environmental impact, reduce wastage, increase traceability across the supply chain, provide markets with safe, high quality products to meet consumer demands (Vilani, 2017).

Current machine learning (ML) techniques, in particular Deep Learning, have taken off thanks to the availability of a huge amount of data (big data). This has been made possible in the food chain thanks to IoT (Internet of things) (Misra, 2020) for instance. Challenges associated with IoT have been highlighted in a recent review of the GODAN (Global Open Data for Agriculture and Nutrition) to build actors' confidence in a sustainable food system (Serazetdinova et al., 2018).

But when data is sparse, incomplete or inaccurate, solutions can still be found (Perrot et al., 2016)(Perrot & Baudrit, 2012). Algorithms have to be adapted, regarding uncertainty management in particular. Solutions based on the use of complex stochastic optimisation

heuristics (Lutton et al., 2016) have been proposed. Another way to deal with this issue is to rely on human knowledge and expertise (implicit as well as explicit) and build more and more rich and adapted human-ML interactions (Boukhelifa et al., 2017).

This chapter is our attempt to deal with the Integration of human knowledge and user-interactions into AI methods in the case of time dependent data sets. Through three examples, we show how human-centered approaches can be built to deal with sparse data:

- A complex model, based on a Long-Short-Term Memory Network, built from time series data thanks to ML: it is shown that human-driven choices have a drastic influence on the quality of the results.
- A tool to predict grape berries quality, thanks to a Dynamic Bayesian Network coupled with expert rules. In this system, expert knowledge has been integrated into the model thanks to an elicitation process, making it explicit into the structure of the model through the determination of the variables dependencies and discretization.
- Finally the last example deals with implicit knowledge, such as the priorities experts give to model objectives as they explore trade-offs. It has been shown that such implicit knowledge is able to provide extremely useful information if mobilised thanks to an appropriate data visualization. It describes an exploratory analysis that explicits how experts used an interactive visualization system to explore time series datasets in the food domain.

1 A Long-Short-Term Memory Network model for biscuit baking

In this case study, we describe how machine learning can be used to model a dynamic process, such as biscuit baking. Despite the efficiency of the technique, we show how an

appropriate, human-curated choice of the training set can dramatically improve the final results

Long-Short-Term Memory (LSTM) networks are specific to the field of Artificial Neural Networks (ANNs). LSTM networks are specifically tailored for machine learning of time series, where the outputs of a system are not just a function of their inputs, but also of an internal state. The state itself can be seen as dependent on the historical series of all inputs seen by the system up to that point in time. We present an application of LSTM networks to the modeling of biscuit baking. Starting from 16 real-world time series of biscuit baking, gathered by the United Biscuits company under different conditions, we show how the proposed LSTM network can correctly predict unseen values. Remarkably, the network is also able to reproduce a dynamic behavior up to variations that might be overlooked as noise.

The process of baking biscuits in industrial ovens involves a considerable number of different biochemical and physical phenomena, such as denaturation of proteins, Maillard reactions, and gelatinization of starch. Due to the complex interactions between these phenomena, creating a physically accurate mathematical model of the biscuit baking process is a challenging task. An alternative to a mechanistic model is to use a data-driven approach, a machine learning technique, to derive a black-box model of the whole process from experimental data. In order to assess generality, the model should then be tested on unseen data. Such an approach could also potentially be effective at modeling outputs that are traditionally harder to describe mathematically, such as the color of the biscuits. While most machine learning techniques are ill-equipped to deal with time series, there is a sub-category of algorithms specifically designed to tackle dynamic problems. LSTM networks are currently among the state of the art in the field.

In this case study, we propose the use of a LSTM network to model the biscuit baking process. Starting from a training dataset of real-world time series of biscuit baking, collected

by the company United Biscuits, the proposed approach learns the dynamics of two output variables of interest, color and weight loss, and it is then tested on an unseen test dataset.

In this section we provide minimal information on biscuit baking and LSTM networks, in order to introduce the scope of this work.

1.1 Biscuit baking

During the process of biscuit baking, raw biological materials are transformed into a final product that must satisfy multiple criteria. For example, the color of the product must be pleasant enough to entice customers, or its thickness must be within given thresholds to not create issues for packaging. The industrial transformation process from dough to biscuit is usually performed resorting to tunnel ovens: interestingly, even if the process has been thoroughly studied and can now be precisely controlled, there are complex coupled biochemical and physical phenomena not completely understood and controlled (Savoye et al. (1992)).

Phenomena involved in biscuit baking include gelatinization of starch, denaturation of proteins, and Maillard reactions, that give browned food its distinctive flavor. Such biochemical reactions are linked to water activity inside the biscuits, temperature, and humidity (Wade (1988)). Depending on the structure of the industrial baking oven, convection, radiation and conduction also contribute to baking, to different degrees. Describing mathematically the global heat-mass transfer is not simple, because very little information on the thermal properties of commercial dough is accessible, and the characteristics of the product dynamically change during the process. Furthermore, even if the control variables are known and it would be useful to represent the process, it is extremely difficult to mathematically describe the evolution of sensory properties of biscuits, such as formation of color, loss of moisture, and change in mass.

Given this complexity, it is not surprising that several approaches have been proposed to model and control the industrial baking process, ranging from fuzzy logic (Perrot et al., 2000; Perrot et al., 2006), to heat-transport models (Sablani et al., 1998; Trystram et al., 1993), to models tackling air properties in tunnel ovens (Mirade et al., 2004).

The company United Biscuits, Inc. collected 16 time series of biscuit cooking under different conditions, in the scope of the DREAM FP7 European Project (2009-2013). The oven used during the experiments features four different zones, with different temperatures. During the cooking process, biscuits are slowly moved from one zone to the next on metal trays, while the heat flux in the oven is manually regulated by an employee. The considered input variables are: the heat flux measured in the top part of the oven (tf in W/m^2), the heat flux measured in the bottom part of the oven (bf in W/m^2), the nominal heat flux in the current zone of the oven (zc in, W/m^2), and the nominal heat fluxes in the previous zones of the oven that the biscuit tray has already passed (zp1...zp4 in W/m^2). The considered output variables are: the color of the biscuits based on the reflected light measured, L of the CIELAB system (c), and the weight loss of the biscuits, measured (wl in g). Each variable is measured every 5 s, with each baking process lasting 350 s, with a total of 70 points per time-series. Color is always measured on the same individual reference biscuit during the whole time series, weight loss is taken as an average on the same 3 reference biscuits during the experiment. Additionally, the initial conditions of variables c, and wl are used as inputs during the experiments.

Out of the 16 time-series, several are repetitions of an experiment under the same conditions (in groups of 3, 3, 2, 3, 2, 3 time series, respectively). Table 1 summarizes the features of the dataset. Figure 1 shows an example of time series, highlighting the non-negligible differences even between repetitions under the same conditions. Another notable feature is that output variable wl presents a behavior that, at a first glance, seems extremely noisy.

Table 1 about here.

Figure 1 about here.

1.2 Long-Short-Memory networks

LSTM networks (Hochreiter and Schmidhuber, 1997; Gers et al., 1999) are a category of ANNs, belonging to the class of Recurrent Neural Networks (RNNs) (Hopfield,1987). Classical ANNs (Rosenblatt, 1958) are machine learning approaches loosely inspired by neural networks in the brain that can work as general function approximators. ANNs are composed by a series of units called artificial neurons connected to each other, able to receive and send signals. Usually, the signal at a connection between artificial neurons is a real number, and the output of each artificial neuron is calculated by a non-linear function of the weighted sum of its inputs. Like most machine-learning approaches, ANNs can approximate an unknown function by learning the appropriate weights in the artificial neurons from a training dataset featuring several combinations of inputs and outputs for a target phenomenon. In order to evaluate the generalization ability of trained models, ANNs are then usually tested on a dataset of unseen values, called test dataset or test set.

ANNs have success stories in applications ranging from games (Silver et al.,2016) to image classification (Sermanet et al., 2013), they are designed to model processes for which the outputs depend exclusively on the current inputs. In dynamic processes, however, the outputs are also a function of an internal state that is itself dependent on the history of inputs until that point. RNNs try to address this issue, by adding connections between units to form directed cycles. Thanks to this feedback mechanism, RNNs exhibit dynamic temporal behavior and are used in issues where the sequence of inputs is relevant for the outputs, such as speech recognition or natural language processing.

LSTM networks are one of the most successful paradigms of RNNs: in a LSTM network, each unit is considerably more complex than a simple artificial neuron in an ANN (see Figure 2). A LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell is responsible for storing values over an arbitrary time interval, while each gate regulates the flow of values that goes through the connections of the LSTM: the input gate controls the extent to which a new value flows into the cell, the forget gate controls the extent to which a value remains in the cell and the output gate controls the extent to which the value in the cell is used to compute the output activation of the LSTM unit. Thanks to the ability of storing information over variable intervals of time, LSTM networks currently represent the state of the art in several domains, such as speech recognition (Xiong et al., 2017).

Figure 2 about here.

1.3 Experiments

The 16 time-series are split into a training set (12 time series) and a test set (4 time series), the latter of which will be unseen by the LSTM network during the training phase. The test set has been selected among the repetitions of experiments in conditions already present in the training set. All variables have been normalized by subtracting the mean and scaling to unit variance, on the basis of the values contained in the training set. After a few tentative runs, the parameters of the network are configured as follows: 8 inputs (all previously described input variables plus the initial conditions for the 2 output variables), 50 units in a single hidden layer, 2 outputs (all output variables); tanh activation function (hyperbolic tangent), 3000 training epochs (iterations of the optimization process for the weights), RMSprop gradient descent optimizer (Hinton et al., 2014). All the code of the machine learning algorithm is

implemented in the Keras (Chollet et al., 2015) and scikit-learn (Pedregosa et al., 2011) Python libraries.

The final model has excellent fitting on the test set, with mean squared error MSE = 0.015 and R2 = 0.9863. An interesting result is that, visually, the model is able to reproduce trends in unseen data that at a first glance might be mistaken for noise: for example, in Figure 3, the model is able to closely predict the behavior of the weight loss showing that the signal-to-noise ratio is better than what a human expert could have considered from a superficial analysis of the data.

While the results are quality-wise satisfying, it is important to remark that the correct choice of the training set can make a considerable difference in the final generalization capabilities of the model. In a second set of experiments, the same model previously described is tested on a leave-one-out cross-validation, being iteratively trained on all available time series except one, and tested on the one being left out. From the results reported in Table 2, it is noticeable how the performance on some of the time series is subpar, probably because the type of information they contain cannot be extrapolated from the rest of the data, and thus represent a unique contribution that the machine learning model needs in order to properly characterize the phenomenon.

Table 2 about here.

Figure 3 about here.

2 Prediction of grape berries quality for decision making

The grape berries maturation is a complex process relying on physicochemical and biochemical reactions. These reactions depend on multiple factors of which the climate is the most influential, especially in the last weeks preceding the harvest. Since berries maturity plays a major role in determining wine potentialities, to anticipate the maturation process and determine the right harvesting date is a significant challenge for the wine industry. Different indicators to evaluate the maturation state can be considered, which might be chemical, and thus exactly measurable as the content of sugar, or sensory characteristics, as the seeds color, which requires an expert evaluation on a symbolic ordinated scale. Sensors have been developed in last decade measuring some grape characteristics as color, sugar content or aromatic potentialities, (Ben Ghozlen et al., 2010; Geraudie et al., 2010). Nevertheless, those analysis are most of time realized in laboratory, time consuming and generally expensive for a close monitoring of the grape berries maturity and never predictive.

For the grape maturity prediction, a model has been developed by (Baudrit et al., 2015; Perrot et al., 2015) linking chemical indicators to weather conditions on Cabernet Franc grape berries. The approach presented here is an extension of this work.

2.1 Grape berries maturation

Experimental data are generated from vineyard plots located in the Loire Valley region followed by the IFV institute over several years with weekly sample collection. It covers years from 1989 to 2017, with plots distributed between two geographical places: “Anjou” and “Touraine” for a total of 30 parcels including between 2 to 5 points by kinetics for each parcel according to the year of the experiment.

As inputs, meteorological conditions over 7 days were supplied by Meteo France meteorological stations located near and/or on the parcels. :

- Temperature (°C) labeled “t”, $\{\sum_{i=1\ to\ 7days}(t_{min,i}+t_{max,i})\div 2)\}$,
- Rainfall (mm) labeled “pl”, $\{\sum_{i=1\ to\ 7days}(pl_i)\}$,
- Relative humidity (%) labeled “hr” $\{\sum_{i=1\ to\ 7days}(hr_{min,i}+hr_{max,i})\div 2)\}$.

The solar radiation (in hours) over 7 days, labeled “Ins”, $\{\sum_{i=1\ to\ 7days}(Ins_i)\}$ was only given by one meteorological station located at Montreuil-Bellay, in the center of the area of study.

As outputs, physicochemical and sensory measurements were achieved:

- Physicochemical measurements selected for this study are those defined by the experts as essential: sugar(s) en g/l, total acidity (ac) in equivalent H_2SO_4 g/L and malic acid (ac_m) in g/L, (Barbeau, 2003; Riou, 1994). Their variations during a week (between two points) are also considered: Var_s; Var_ac; Var_ac_m. Each week, a lot of two-hundred berries of Chenin, with pedicels, were randomly picked up from each parcel at each ripening stage according to the method of Vine and Wine French Institute (ITV-France) (Cayla et al., 2002) in order to limit the effects of the grape heterogeneity. With the lot of two-hundred berries of each sampling, a crushing was realized with a blender, then the must was filtered through a Whatman paper filter. Reducing sugars concentration (g/l) was measured with a refractometer; total acidity (g/l eq. H_2SO_4) by the titration method and malic acid (g/l).

- Sensory measurements were also achieved on the berries managed by the IFV institute and the ESA institute. Several measurements were described on the grape berries, the juice and the global flavor using an ordinated scale range from 0 to 5. For the DBN model we have chosen to select an integrative flavor indicator: the global aromatic intensity (IntGloAro) to complete the physicochemical predictions.

2.2 Expert knowledge handling

3 scientists and 2 winegrowers experts working on the areas considered in this study were asked. Each of them was interviewed during one or two sessions (2–3 h). Each of the elicitation sessions was attended by one expert and one or two interviewers. To build the interview, adapted methods proposed by (Sicard et al., 2011) were applied. The elicitation process was based on a set of predetermined structured open-ended questions used to direct the interviews. Questions were designed according to techniques based on survey methods with the aim of optimizing the expression of expert knowledge. We paid particular attention to context reinstatement. This involves having the expert think about and describe his feelings during the episodes being recalled.

2.3 The Dynamic Bayesian Model (DBN)

The model used in our approach is a Dynamic Bayesian Network (DBN), a probabilistic graphical model able to describe phenomena developing over time (Jensen & Nilsen, 2010; Pearl, 1988). The structure of a DBN is an oriented graph, representing correlations between variables that in our case was created by interacting with human experts. Once the structure of a DBN is fixed, it is then possible to compute its parameters starting from a dataset: the parameters are conditional probability tables, assessing the probability for variables taking a

specific value, knowing the values of the variables they depend on. For our specific application, the values of the variables need to be discretized. Differently from a classical Bayesian Network, a DBN makes it possible to estimate variable values over several subsequent time steps. In our case, each time step is equivalent to two weeks in the grape ripening processes. DBNs have been successfully adopted for several agri-food applications (Baudrit et al., 2015; Perrot et al., 2015).

More formally, a DBN is a graph-based model of a joint multivariate probability distribution, capturing properties of conditional independence between variables. Like a BN, a DBN is a directed acyclic graph (DAG) where the nodes represent variables, and the missing arcs represent conditional independence between variables. In DBNs in particular, nodes $X(t) = (X_1(t), \ldots, X_n(t))$, represent n discrete random variables, indexed by time t, providing a compact representation of joint probability distribution P for a finite time interval $[1, \tau]$. In other words, the joint probability P can be written as the product of the local probability distribution of each node and its parents, as follows:

$$P(X(1), \ldots, X(\tau)) = \prod_{i=1}^{n} \prod_{t=1}^{\tau} P(X_i(t)|U_i(t))$$

Where $U_i(.)$ denotes the set of all parents of node $X_i(.)$, and $P(X_i(.)|U_i(.))$ describes the conditional probability function associated with random variable $X_i(.)$ given the values of $U_i(.)$. $X_i(t)$ is termed "slices", and it represents the set of all variables at time t. This factorization of the joint probability distribution, based on information from the graph, makes it possible to straightforwardly represent large models, and use them for practical applications. In other words, DBNs represent the beliefs of possible trajectories of the variables involved in a dynamic process.

In order to make the problem treatable, DBNs assume the first-order Markov property: the parents of a variable in time slice t must appear in either slice $t - 1$ or t. As a consequence, for the first-order homogeneous Markov property, the conditional probabilities are time-invariant, meaning that $P(U(t)) = P(U(2)) \; \forall \, t \in (1, \, \tau)$. In order to fully specify a DBN, we will then need to define the intra-slice topology (within a time slice), the inter-slice topology (between two time slices), as well as the parameters (i.e. conditional probability functions) just for the first two time slices. The structure of a model can be explicitly built on the basis of knowledge available in the literature and parameters can be automatically learned without a priori knowledge on the basis of a dataset, a processed termed parameter learning. The techniques for learning DBNs are generally extensions of the techniques for learning BNs. Specialized literature reports several methods to learn the structure or the parameters of a DBN from substantial and/or incomplete data (Geiger & Heckerman, 1997; Heckerman, 1999). In this work, the topology of the graph is obtained from expert knowledge; for parameter learning, we consider the simplest and most commonly adopted methodology, simply evaluating the co-occurrence rate of values of variables in the training data.

One a DBN is fully specified, it can be used to estimate marginal probabilities for target variables, through a process also known as Bayesian inference:

$$P\left(O\left(t^{'}\right)\right) = o\left(t^{'}\right), \; \forall \, t^{'} \in [1, \tau]$$

Where X is a set of variables whose values we are interested in predicting, and O is a set of variables whose values are known (for example, in food processing X might be the variables representing the physicochemical properties of a product and O might be the variables representing the observed environmental conditions). In general, given a way of calculating $P(X(t)|O\left(t^{'}\right))$ from the knowledge of $P(X(t^{'})|O(t))$, inference in a DBN is performed using recursive operators and Bayes' theorem, updating the belief state of the DBN as new observations become (Murphy, 2002).

The DBN previously introduced is evaluated with a leave-one-out cross-validation (LOOCV), where the model is repeatedly trained on the whole dataset, minus one sample, and the remaining sample is used for testing. The procedure is repeated until each sample has been used for testing. Considering the mean and standard deviation on the results of a LOOCV provides a better estimate of the model's capabilities than just considering a random split of the available data between a training set and a test set (Geisser, 1993).

For the choices made in this study, before training the model, it is necessary to discretize the real-valued variables in the dataset (see subsection 3.1.1). However, in order to evaluate the performance of the model's predictions against the ground truth, the results of the model will have to be converted back into real values. Recalling that the predictions of a DBN model for variable x will consist in a series of probabilities P_i for each possible discrete class $i = 0, 1, ..., n_x$ associated with variable x, the predicted outcome can be converted to a real value using the following equation:

$$x^{predicted} = \sum_{i=1}^{n_x} \overline{x}_i P_i$$

Where $\overline{x}_i$ is the average value of all samples of variable x that fall under class i.

The first metric used to evaluate the quality of the predictions against the ground truth is the root mean squared error (RMSE):

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i^{predicted} - x_i^{observed})^2}$$

Where N is the number of predictions considered for target variable, and $x^{observed}$ indicates its observed value. In this study, we will also use the relative RMSE (RRMSE) that expresses the RMSE as a percentage of the range of observed values for the target variable, and it is thus more informative as an error metric:

$$RRMSE = \frac{RMSE}{max(x^{observed}) - max(x^{observed})} \times 100$$

2.4 Experiments

Inspired by previous work on Cabernet-Franc and Gamay wines (Baudrit et al., 2015), the network structure predicts physicochemical indicators starting from weather conditions (Figure 4). Only the climatic variables having an influence on each physicochemical maturity indicator are selected from expert knowledge and literature. In particular, relative humidity only affects the two acidities, sunshine influences sugar content, while temperature and rainfall have an impact on the four variables considered: sugar (s), total acidity (ac), acid malic (ac_m) and the global aromatic intensity (IntGloAro).

Figure 4 about here

As the DBN needs to be able to capture dynamical variations of the values over time, to better predict the four variables it is necessary to define new intermediate state variables. We consider that a month before the harvest, only alterations in the weather can cause a significant deviation from an established trajectory in time (see Figure 5).

More formally, considering each physicochemical variable $x \in \{ac, ac_m, s\}$ at time t and $t + 1$:

$$x(t + 1) = var\_x(t + 1) + x(t)$$

And consequently

$$var\_x(t) = x(t) - x(t - 1)$$

As already anticipated, the (absolute) value of a variable can be used as an indicator of the current stage of ripening, while the variation, as a function of the climatic variables, will dictate the ripening trajectory.

Figure 5 about here

At time = 0, the value of each variable is observed and known; for the next two-times steps, only the climatic variables are observed and known, while the physicochemical quantities and their variations are predicted by the model.

As previously described, to create the CPTs of our DBN model, it is necessary to define the discretization of the continuous variables in the problem. In this context, discretizing variable x amounts to finding several intervals $\{[x_1, x_2), [x_2, x_3), \ldots, [x_{n-1,} x_n)\}$ of continuous values such that $x_1 < x_2 < \ldots < x_n$, with each interval corresponding to a discrete class.

For the climatic variables, the following intervals were defined:

- $Ins = [[15, 30], [30, 40], [40, 55], [55, 60], [60, 75]]$
- $pl = [[0, 10], [10, 20], [20, 30], [30, 45], [45, 70], [70, 100]]$
- $t = [[0, 11], [11, 15], [15, 17], [17, 19.5], [19.5, 22]]$
- $hr = [[60, 70], [70, 75], [75, 80], [80, 90], [90, 100]]$

For the grape sensory variable IntGloAro, the discretization is fixed to 1 inside the sensory scale [1,5]. It is linked to the limit of sensitivity evaluated to be 0.5 by the experts.

For the physicochemical variables, an interactive semi-automated discretization approach was developed, based on the notion of co-occurrence between variable values and their variations. The methodology is based on a visualization software, EvoGraphDice, coupled with an evolutionary optimization approach (Boukhelifa et al., 2017). For example, the variation of the variable sugar var_S, is fixed on the basis of the expert description, more focused on the variations of the value of the variable than on the variable itself. The optimal discretization of the sugar is then calculated to ensure the most as possible a homogeneous repartition of the var_S classes of interval for each sugar interval in the data basis.

The results of the optimization and thus the discretization proposed for the physicochemical variables are presented table 3.

Table 3 about here

After performing a LOOCV on the dataset, where at each iteration the network is trained on the whole dataset minus one sample, and then tested on that sample, we obtained a mean RRMSE for each predicted variable. For the sensory variable, the RMSE is also calculated but also the percentage of points well classified in the five classes considered with a threshold at 0.25 or 0.5, 0.5 being considered as the classical sensory threshold of sensibility for those measurements.

Table 4 about here

The results (table 4) show that it is possible to predict with good results the total acidity and the sugar in a range that is satisfying for the experts (10% for the sugar, 6% for the total acidity) and so anticipate the maturation two weeks before. For the malic acid it seems to be more complex to have a good prediction with the only variables considered as inputs of the DBN. Probably for this variable, for a better prediction, we would have to define the state by including other parameters or variables.

For the sensory variable, results are also relevant with a variable that seems to be relatively well predicted two weeks before with less good results for a shorter time step. As regards to the integrative and more uncertain measurement represented by this sensory variable, it is possible that the slope for two weeks indicates more the tendency of evolution than the one for one time step, which could explain this result. Nevertheless results of sensory prediction are quite good with 72.5% of good prediction at two time steps.

3 Machine learning user interactions to understand how agronomy experts explore model simulations

Machine Learning ML algorithms build models that are trained to recognize certain types of patterns (Bishop, 2006). Domain experts and decision-makers often rely on these models to reason over new data and to make informed decisions. Whereas it is generally possible to determine what predictions the machine learning model is likely to make based on new input data, how domain experts will use those model predictions for reasoning and to make inferences is uncertain. Because domain experts may not fully understand ML models and their domain knowledge may not be fully encoded in the model, conflicts may arise and they themselves may not be consistent in interpreting and responding to ML results (Valdez et al., 2017; Fernandes et al., 2018; Dimara et al., 2018).

In previous work Boukhelifa et al. (2019) have conducted an observational study to understand how domain experts use ML models to explore agri-food processes. Multiple interactive sessions were organized where experts from agronomy explored model simulation datasets using an existing exploratory visualization tool (Elmqvist et al., 2008; Cancino et al., 2012) (Figure 6). These exploration sessions were video-recorded and experts' interactions with each other and with the tool were logged. The main exploration task was open-ended, but experts' primary goal was to explore alternative *trade-offs*, such as between the amount of fertilizer supplied and the quantity of crop yield.

Figure 6 about here

These exploration sessions were helpful to the domain experts who, guided by the views proposed by the ML model, found useful insights in the form of interesting correlations, temporal trajectories and trade-offs that they have not considered before. As a group, they

were able to guide the ML component to interesting views and to reason about their data. However, the qualitative analysis of the video recordings shows that experts often lose track of their analysis steps and therefore the many trade-off scenarios they were trying to compare, as noted by one participant from the study: "*What was the basis of the reflection here? In fact, we seem to go faster than we have time to note down*". Domain experts also appear to structure their investigation into mini-analysis scenarios, during which they explore different hypotheses and research questions. But, when asked, they were not able to give a clear overview of past exploration, or an accurate evaluation of whether their exploration strategy was a robust or exhaustive one.

Exploratory Data Analysis EDA tools (Grinstein, 1996), such as the visualization system used in this study, provide different types of visual and statistical methods to analyze the data and to examine them from different viewpoints. However, they offer limited support for viewing the exploration history, for example, by visualizing past analysis steps or data queries (Heer et al., 2008). Little support is typically provided to show high-level information to entice users to reflect upon and make sense of their past exploration. This type of information, called *provenance* (North et al., 2011; Bors et al., 2019; Madanagopal et al., 2019), could provide opportunities to review and share insights, but importantly, it can potentially improve user exploration practices and strategies (Carrasco et al., 2017).

In what follows, ML is not only considered as a means to guide visual exploration, but also to structure and help users revisit and reflect on past exploration sessions. This work describes (a) the modelling of the user exploration history of the aforementioned exploration sessions, and (b) the visualization of provenance information to the analysts as high-level views of their past exploration (Barczewski et al., 2000). The main goal is to establish a methodology to automatically detect key analysis stages of the exploration, which correspond to the change of focus in the trade-off analysis space. To detect such changes, unsupervised learning and time

series modelling (Hidden Markov Models HMM) are applied to two use cases from agronomy.

3.1 Characterizing exploratory data analysis

There is an established body of work from cognitive psychology that looks at how people make sense of data during exploratory analysis. Prominent sensemaking theories such as by Klein et al. (Klein et al., 2007) and Pirolli et al. (Pirolli et al., 2005) focused on the cognitive processes involved. Their results show that analysts continuously re-frame their research questions (Klein et al., 2007), and interleave new and refined hypotheses in a non-linear fashion (Pirolli et al., 2004). This work builds on these theories and findings, and focuses on different aspects of sensemaking such as uncertainty (Boukhelifa et al., 2017b), alternatives (Liu et al., 2019) and structures within the so-called exploration scenarios (Boukhelifa et al., 2019). These analysis scenarios correspond to shifts of user focus in the search space at different stages of the exploration. Six types of scenarios are identified including instances where analysts examine new and refined research questions and hypotheses, and others where they attempt to recap and establish common ground (Boukhelifa et al., 2019; Goyal et al., 2016). The approach thus far in studying sensemaking activities has been based on qualitative research methods such as observational studies, walkthroughs and interviews (Creswell, 2002). Although this approach can yield deep insights, it is often time and resource intensive, and findings may be hard to generalize. In a follow-up work (Barczewski et al., 2000), an automatic procedure is implemented to detect scenarios from logs of user interactions, and new visual designs to incite analysts to reflect on their progress and exploration strategies are investigated.

Logging user interaction is common in interactive systems. User interaction logs are often analyzed not only to evaluate how tools are operated by end-users, but also to help the end-users themselves reflect and track their progress. For example, in the context of web

browsing, Carrasco et al. (Carrasco et al., 2017) showed that when high level semantic information is shown, users tend to reflect on their browsing habits and are able to infer areas of improvement. Guo et al. (Guo, 2015) found that visualization of interaction logs improves analysts' performance in finding insights. Like in this study, they found that exploration is composed of multiple chunks which have generic analysis patterns that lead to insight.

Analyzing user interaction logs is also used in analytics provenance (North et al., 2011; Bors et al., 2019; Madanagopal et al., 2019) to understand user's reasoning processes, and to support collaborative communication and replication (Ragan et al., 2015). Mining user interactions also serves other purposes than making sense of user exploration, such as to predict users personality traits (Brown et al., 2014), or to detect cognitive biases (Wall et al., 2017). In the context of exploratory data analysis, this work is similar to Aboufoul et al. (2018) and Dung et al. (2016) who used HMMs to model user's search behavior. HMMs are powerful techniques to generate sequences of observations and to learn about the hidden states that produce those observations. In the present study, it is shown that analysis scenarios can be retrieved when considered as hidden states of a Markov chain. Results from the HMM are provided in pseudo real time, which can continuously give high-level semantic information to the analyst.

3.2 Preliminary analysis of two use cases from agronomy

We collected interaction log data from an observational study based on two real-world use cases in agronomy, one for wheat fertilization and the other for wine fermentation. In each use case, experts from different domains (such as oenology and microbiological engineering for the wine use case) explored model simulation data using a scatterplot matrix (SPLOM)-based tool (Cancino et al., 2012) projected on a large shared tactile display (84" screen, Figure 6). We recorded videos of two exploration sessions per use case, and followed the thematic

analysis method (Braun, 2012) to analyze them. Findings from this analysis are reported in a previous study (Boukhelifa et al., 2019). Figure 7 shows the different types of scenarios that were identified.

Figure 7 about here

The visualization tool domain experts used to explore their model simulations data had four key functionalities: **(a)** visual query selection to help experts narrow down their search to important dimensions and value ranges; **(b)** a history bookmark to keep track of previous views (scatterplots) they visited; **(c)** Favorite views album to store interesting scatter plots and findings; **(d)** and a dimension editor to manually specify new dimensions using a mathematical formulae. Alternatively, combined dimensions can be generated automatically, using an interactive evolutionary algorithm, which learns from user interactions and feedback. Log data events were collected pertaining to user visits of scatterplots in the SPLOM, whether this originates directly in the SPLOM through cell selection, or indirectly by retrieving views from the favorites store or the bookmark history.

A preliminary analysis of the *scatterplot visits data* shows that the manually identified analysis scenarios often correspond to localized areas of the search space (Figure 8). For instance, for the wine use case, scenario 1 focuses on changes in the amount of initial nitrogen (N0) at five different stages of the wine fermentation process (shown as a vertical line of colored blue dots for each stage: T0, T25, T50, T75 and T100). In scenario 2, experts examined the relationship between N0 and a target aromatic combination that they entered manually, also with regards to the different stages of fermentation (horizontal line of colored green dots). These initial findings inspired the next analysis step where the *scatterplot visits data* was used to cluster user interactions into different analysis scenarios.

Figure 8 about here

In the following section, clustering and machine learning methods are used to detect those scenarios automatically from a sequence of user events (*scatterplot visits data*). Two unsupervised methods were implemented: a clustering method based on a spatiotemporal similarity measure, and a Hidden Markov Model HMM to detect transitions between scenarios. The ground truth in both cases is the manually labelled video dataset. To evaluate the proposed methods, the existing notions of Type I and Type II classification errors were used. However, since the scenarios are chronologically structured, changes of scenarios are more important than knowing the exact identity of a class, say whether it is scenario '1' instead of '3'.

3.3 Spatiotemporal distances to cluster user interaction events

To cluster the scatterplot visits events into different scenarios, there are three steps to follow:

(i) Data preparation: since the datasets the experts explored are trade-off datasets and describe biological processes that are dynamic in nature (fermentation and fertilization processes), the data dimensions are grouped into three generic types : objective dimensions (quantities experts would like to optimize, such as through minimization or maximization), parameters (model parameters experts can control or modify) and trajectories (a subset of parameters whose values change over time).

(ii) Distance calculation: the distance is then calculated between area clicks on the SPLOM (cells) using the Jaccard distance. The result of this step is a distance matrix;

(iii) Clustering: finally, the DBSCAN algorithm is applied to the distance matrix from step ii. The rationale behind this method is to group user interactions with the SPLOM that are close both spatially (based on the location of cells in the SPLOM) and temporally (based on the time elapsed between two selection events).

Figure 9 (bottom timeline) shows the results of the clustering method for the wine use case. The top timeline shows the ground truth data. Each dot represents a user event (a scatterplot selection or visit) and color corresponds to scenarios S1-14 of this use case. The clustering method detects more scenarios than there are in the labelled dataset. When considering scenario transitions only, the method correctly detects 61% of scenario transitions for the wine use case, and only 55% of transitions for the wheat use case. An example of a transition that is correctly detected, is between scenarios 5 and 6 in Figure 9. Cases where the clustering method does not perform well are typically the shorter scenarios where domain experts quickly explore different areas of the search space, more likely to confirm previous knowledge. Another limitation of the clustering method concerns step (i) for the data preparation. In this step, data dimensions are grouped into three types that are identified as pertinent for the different use cases, and more generally when exploring trade-off datasets for dynamic systems. The clustering method is thus highly dependent on the order of dimensions in the SPLOM, yet this order is arbitrary. Moreover, the adopted timescale is also arbitrary and may have a big impact on the clustering performance.

Figure 9 about here

An alternative non-supervised clustering method that addresses these limitations is proposed in the next section.

3.4 A Hidden Markov model to detect scenario transitions

Hidden Markov Models HMMs are used in many real-world applications to model sequences of events where the probability of each event depends solely on the state of the previous event (Baum et al., 1966). The basic assumption which underpins HMMs is that observations are created by hidden states whose successions depend on transition probabilities. In

unsupervised use cases, studying the observations helps to find the hidden states or patterns in the data. For modelling purposes, it is assumed that the hidden states correspond to the analysis scenarios, and the observations (i.e. scatterplot visits events) fall into a Markov system. The number of hidden states in this study is the hyper parameter, which is set to two as it corresponds, conceptually, to whether or not there is a "change" or "shift" in the exploration strategy or search direction.

To build our HMM, two types of information are needed, which are extracted from the user interaction logs: the time-delta between observations, and the row and column combination for each scatterplot visited during the exploration. To summarize the performance of the HMM, Figure 10 presents the confusion matrix. Since the goal is to detect transitions between scenarios, rather than the scenario labels themselves, the shape of the path in Figure 10 is more important than the inferred labels (the closer to the diagonal the better). Figure 10 shows that change of scenarios are well detected for ground truth labels between 5 and 9. Overall, the obtained HMM model is able to detect scenario transitions in 91% of cases for the wine use case, and 75% for the wheat use case.

Figure 10 about here

3.5 Visualizing the machine-learned storyline

Besides modelling user exploration into sequences of scenarios, a second goal of this work is to visualize the results of the HMM method to analysts during or after the exploration. To design this type of visualization, user-centered design methodologies are used (Norman and Draper, 1986) to explore the design space and to gather user requirements. We organized three brainstorming sessions with nine participants in total. Each session lasted roughly two hours. The first two sessions had five participants with design, HCI, or visualization background,

and the third session had four participants from an agronomy research centre in France. Participants were either researchers or PhD students.

These brainstorming sessions were organized into two parts. First, participants were trained to use the SPLOM-tool, similar to what was proposed in previous work (Boukhelifa et al, 2017a). An ideation part followed in which participants brainstormed about new features they would like to see implemented to support sensemaking of their exploration history. The design ideas were collected and organized using affinity diagramming and thematic analysis. The results are the following high-level user requirements which are ordered by how frequently they were mentioned by the participants: (1) story-tell and author; (2) highlight interesting views; (3) show trends; (4) preview and replay; (5) filter views; (6) compare views; (7) group views; (8) show overview and summary; (9) annotate; (10) save and reuse; (11) steer; (12) initialize; and (13) learn and update.

The most frequent user requirements mentioned during the brainstorming sessions corresponds to the storytelling and authoring category. Here participants were interested in tools to automatically create a storyboard of their past exploration, and to annotate it such as by adding tags to places where the exploration branched out, or where they found an important insight. Other participants suggested a git-like visualization that gives both an overview of visited cells and possible branching paths. Inspired by those requirements and findings from previous work (Boukhelifa et al., 2019), a timeline of past exploration, called a "storyline", has been implemented, where nodes are events linked through time. Figure 11 shows a preliminary result of this provenance visualization integrated into an existing SPLOM-tool as a widget, which can be enabled on demand. Here, analysis scenarios are automatically identified using the HMM method and are then visualized using color.

Figure 11 about here

For the implementation of the storyline visualization, a client-server architecture provided by Flask (a Python web framework) was used. The *server side* handles the logging and modelling components described earlier. User interactions with the SPLOM were collected, as well as the history bookmark and the favorites views album, which were stored in a text file. Using those events (or observations), a HMM detects the hidden states (i.e. the scenarios). Since an unsupervised approach is adopted, the model is applied each time a new set of events are recorded to our log file. The update rate for the timeline widget was arbitrarily set to five minutes, but the user can request an update at any time by pressing the update button.

The visualization and rendering are handled on the *client side*, which implements a storyline widget. To facilitate the sharing of the tool, this widget is integrated into a web-version of the SPLOM-Tool. The storyline widget is composed of linked nodes, each node corresponds to a scatterplot selection event, and its color corresponds to a detected scenario. Clicking on the node renders the corresponding scatterplot in the zoomed in area of the user interface, and highlights that cell in the SPLOM through brushing and linking. Since the storyline visualization is not the primary task for domain experts, it is placed at the bottom of the user interface, to avoid interfering with the main exploration tasks.

The storyline visualization provides an overview of how domain experts structure their exploratory analysis, and can be helpful for self-reflection and tracking progress. However, more work is needed to confirm whether indeed such history visualizations encourage reflection and result in a change in exploration strategies. There are currently three main limitations to this work. First, the log data comes from two case studies where domain experts explored trade-offs between multiple dimensions (or objectives). The ML approach needs to be tested with more use cases and different types of datasets. Second, the visualization tool used in the previous case study relies on a SPLOM representation of the data. The way domain experts structure their exploration may depend on how the tool's user interface is

organized. However, the machine learning method can be generalized and applied to other visualization types, since it only requires information about the data dimensions consulted during the exploration and the time of viewing. Third, although cases of branching were observed during exploration, where experts explored an alternative trade-off subset, the automatic method does not currently detect multiple, parallel or branching storylines. The machine-learned storyline visualization can be improved in future work by detecting branching and different types of scenarios, and by allowing experts to augment these storylines with their own annotations, thus integrating their expertise and the insights gained during the exploration.

4 Conclusions

This chapter offered several viewpoints and approaches for machine learning techniques, learning from data, expert knowledge and interactions. It has been shown in the first chapter that generalization capabilities of some ML models highly depend on the quality of the dataset, the larger the better. However, even a simple expert's choice for building a good quality training set (representativeness, extrapolation ability) makes a considerable difference.

In the second chapter, it has been shown that expertise can be made explicit (in the form of a graph model and variable discretization) and embedded into a complex model to build an efficient decision-making system.

Going further, the implicit expertise, i.e. non conscious skills or knowledge difficult to explain verbally, can be fed into a ML process thanks to interactive visualization. The analysis of the system proposed in the third chapter, a SPLOM-based visualization tool, has led to the design of a new tool, to assist exploratory data analysis. The proposed storyline visualization helps

domain experts self-reflect and track their progress when analyzing complex model simulations.

The key message of the work presented in this chapter is that human expertise can be efficiently -- and finely -- nested into data-driven machine learning schemes, which is particularly beneficial in the case of sparse or uncertain data. The three examples presented above outline future components of such interactive systems. This approach is relevant for food-related systems, where at the same time (i) some data still remain unusable, expensive and time consuming to acquire, (ii) human expertise, skills and know-how are rich and numerous, but often implicit, and (iii) there is a strong need for efficient predictive models, decision support systems and knowledge preservation.

Acknowledgements

First chapter: this work was achieved under the frame of data coming from partners of the FP7 European DREAM Project: “Design and development of realistic food models with well characterized micro- and macro-structure and composition”. https://cordis.europa.eu/project/id/222654/it

Second chapter: this project was founded by the CASDAR 2015 project coordinate by the “Institut de la Vigne et Vin” (IFV): "Développement et transfert d’un outil d’aide à la décision appliqué à la maturation des baies de raisins rouge et blanc". We acknowledge the IFV and particularly Laurence Guerin and Etienne Goulet, InterLoire, the ESA Angers USC1422 GRAPPE and all the partners for the data production and opening. We also acknowledge Hervé Guillemin, Bruno Perret and Daniel Picque from INRAE for their help in this project.

Third chapter: is based on a late-breaking extended abstract (Barczewski et al., 2020)

co-authored by Antoine Barczewski.

References

Aboufoul, M., Wesslen, R., Cho, I., Dou, W. and Shaikh, S., 2018. Using hidden markov models to determine cognitive states of visual analytic users. In Proceedings of the Machine Learning from User Interaction for Visualization and Analytics Workshop at IEEE VIS.

Barbeau, G. B.-H. (2003). Comportement de quatre cépages rouges du Val de Loire en fonction des variables climatiques. J Int Sci Vigne Vin, 38, 35-40.Boukhelifa, N., Tonda, A., Trelea, I.C., Perrot, N., Lutton, E. (2017). Interactive knowledge integration in modelling for food sustainability: challenges and prospects. ACM CHI Workshop on Designing Sustainable Food Systems, 2017, NA, France. hal-01604947

Barczewski, A., Bezerianos, A. and Boukhelifa, N., 2020, April. How Domain Experts Structure Their Exploratory Data Analysis: Towards a Machine-Learned Storyline. In Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (pp. 1-8).

Baudrit, C., Perrot, N., Brousset, J.-M., Guillemin, H., Perret, B., Picque, D., . . . Barbeau, G. (2015). A probabilistic graphical model for describing the grape berry maturity. Computers and Electronics in Agriculture. Computers and Electronics in Agriculture, 118, 124-135.

Baudrit, C., Wuillemin, P.-H., & Perrot, N. (2013). Parameter elicitation in probabilistic graphical models for modelling multi-scale food complex systems. Journal of Food Engineering, 115(1), 1-10.Chollet F. et al., (2015). Keras. https://github.com/fchollet/keras.

Baum, L.E. and Petrie, T., 1966. Statistical inference for probabilistic functions of finite state Markov chains. The annals of mathematical statistics, 37(6), pp.1554-1563.

Ben Ghozlen, N., Moise, N., Latouche, G., Martinon, V., Mercier, L., Besancon, E., & Cerovic, Z. (2010). Assessment of grapevine maturity using a new portable sensor: Non-destructive quantification of anthocyanins. Journal International des Sciences de la Vigne et du Vin, Vol. 44, pp. 1-8.

Bishop, C.M., 2006. Pattern recognition and machine learning. springer.

Braun, V. and Clarke, V., 2012. Thematic analysis.

Bors, C., Wenskovitch, J., Dowling, M., Attfield, S., Battle, L., Endert, A., Kulyk, O. and Laramee, R.S., 2019. A provenance task abstraction framework. IEEE computer graphics and applications, 39(6), pp.46-60.

Boukhelifa, N., Bezerianos, A., Trelea, I.C., Perrot, N.M. and Lutton, E., 2019, May. An exploratory study on visual exploration of model simulations by multiple types of

experts. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (pp. 1-14).

Boukhelifa, N., Bezerianos, A., Cancino, W. and Lutton, E., 2017. Evolutionary visual exploration: evaluation of an IEC framework for guided visual search. Evolutionary computation, 25(1), pp.55-86.

Boukhelifa, N., Perrin, M.E., Huron, S. and Eagan, J., 2017, May. How data workers cope with uncertainty: A task characterisation study. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (pp. 3645-3656).

Boukhelifa, N., Tonda, A., Trelea I.-C., Perrot, N., & Lutton, E. (2017). Interactive Knowledge Integration in Modelling for Food Sustainability: Challenges and Prospects. ACM CHI Workshop on Designing Sustainable Food Systems.

Brousset, J. (2009). Caractérisation multifactorielle et modélisation de la maturité de baies de Cabernet Franc en moyenne vallée de Loire. Rapport InterLoire.

Brown, E.T., Ottley, A., Zhao, H., Lin, Q., Souvenir, R., Endert, A. and Chang, R., 2014. Finding waldo: Learning about users from their interactions. IEEE Transactions on visualization and computer graphics, 20(12), pp.1663-1672.Claverie, M., Prud'Homme, P., Mongendre, J., Zabollone, E., Raynal, M., Coulon, T., . . . Forget, D. (2008). Modélisation statistique de la qualité en viticulture par la méthode PLS Spline. VIIe Congrès International des terroirs viticoles.

Cancino, W., Boukhelifa, N. and Lutton, E., 2012, June. Evographdice: Interactive evolution for visual analytics. In 2012 IEEE Congress on Evolutionary Computation (pp. 1-8). IEEE.

Carrasco, M., Koh, E. and Malik, S., 2017, May. popHistory: animated visualization of personal web browsing history. In Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (pp. 2429-2436).

Creswell, J.W., 2002. Educational research: Planning, conducting, and evaluating quantitative (pp. 146-166). Upper Saddle River, NJ: Prentice Hall.

Dimara, E., Franconeri, S., Plaisant, C., Bezerianos, A. and Dragicevic, P., 2018. A task-based taxonomy of cognitive biases for information visualization. IEEE transactions on visualization and computer graphics, 26(2), pp.1413-1432.

Dungs, S., 2016. Describing user's search behaviour with Hidden Markov Models. Bulletin of IEEE Technical Committee on Digital Libraries, 12, p.2.

Elmqvist, N., Dragicevic, P. and Fekete, J.D., 2008. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. IEEE transactions on Visualization and Computer Graphics, 14(6), pp.1539-1148.

Fernandes, M., Walls, L., Munson, S., Hullman, J. and Kay, M., 2018, April. Uncertainty displays using quantile dotplots or cdfs improve transit decision-making. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (pp. 1-12).

Geiger D., Heckerman D. (1997) A characterization of the Dirichlet distribution through global and local parameter independence. The Annals of Statistics 25: 1344–1369.

Geisser, S. (1993). Predictive Inference. New York: Chapman and Hall.

Geraudie, V., Roger, & J. M., O. H. (2010). Développement d'un appareil permettant de prédire la maturité du raisin par spectroscopie proche infra-rouge(PIR). Revue Française d'Oenologie, 240, 2-8.

Gers F.A.; Schmidhuber J.; and Cummins F., (1999). Learning to forget: Continual prediction with LSTM.

Goyal, N. and Fussell, S.R., 2016, February. Effects of sensemaking translucence on distributed collaborative analysis. In Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing (pp. 288-302).

Grinstein, G.G., 1996, August. Harnessing the Human in Knowledge Discovery. In KDD (pp. 384-385).

Guo, H., Gomez, S.R., Ziemkiewicz, C. and Laidlaw, D.H., 2015. A case study using visualization interaction logs and insight metrics to understand how analysts arrive at insights. IEEE transactions on visualization and computer graphics, 22(1), pp.51-60.

Goelzer, A., Charnomordic, B., Colombié, S., Fromion, V., & Sablayrolles, J. (2009). Simulation and optimization software for alcoholic fermentation in winemaking conditions. Food Control, 20(7), 635-642.

Heckerman D. (1999) A Tutorial on Learning with Bayesian Networks. MIT Press, Cambridge, MA, USA, 301–354.

Heer, J., Mackinlay, J., Stolte, C. and Agrawala, M., 2008. Graphical histories for visualization: Supporting analysis, communication, and evaluation. IEEE transactions on visualization and computer graphics, 14(6), pp.1189-1196.

Hinton G. et al., (2014). Neural Networks for Machine Learning. http://www.cs.toronto.edu/~tijmen/ csc321/slides/lecture_slides_lec6.pdf.

Hochreiter S. and Schmidhuber J., (1997). Long short-term memory. Neural computation, 9, no. 8, 1735–1780.

Hopfield J.J., (1987). Neural networks and physical systems with emergent collective computational abilities. In Spin Glass Theory and Beyond: An Introduction to the Replica Method and Its Applications, World Sci- entific. 411–415.

Hunkefer H. (2017) Accenture Report: Artificial Intelligence Has Potential to Increase Corporate Profitability in 16 Industries by an Average of 38 Percent by 2035. https://newsroom.accenture.com/news/accenture-report-artificial-intelligence-has-potential-to-increase-corporate-profitability-in-16-industries-by-an-average-of-38-percent-by-2035.htm

Jensen Finn V. and Nielsen Thomas D.. (2010) Bayesian Networks and Decision Graphs, Springer-Verlag. 464p.

Kavalenko O. (2020). Machine Learning and AI in Food Industry: Solutions and Potential. internalreport:https://spd.group/machine-learning/machine-learning-and-ai-in-food-industry/

Klein, G., Phillips, J.K., Rall, E.L. and Peluso, D.A., 2007, January. A data-frame theory of sensemaking. In Expertise out of context: Proceedings of the sixth international

conference on naturalistic decision making (pp. 113-155). New York, NY: Lawrence Erlbaum Assoc Inc.

Liu, J., Boukhelifa, N. and Eagan, J.R., 2019. Understanding the role of alternatives in data analysis practices. IEEE transactions on visualization and computer graphics, 26(1), pp.66-76.

Lutton, E, Perrot, N., Tonda, A. (2016) Evolutionary Algorithms for Food Science and Technology, John Wiley & Sons.

Madanagopal, K., Ragan, E.D. and Benjamin, P., 2019. Analytic provenance in practice: The role of provenance in real-world visualization and data analysis environments. IEEE computer graphics and applications, 39(6), pp.30-45.

Mirade P.; Daudin J.; Ducept F.; Trystram G.; and Clement J., 2004. Characterization and CFD mod- elling of air temperature and velocity profiles in an industrial biscuit baking tunnel oven. Food research international, 37, no. 10, 1031–1039.

Misra, N.N., Dixit, Y., Al-Mallahi, A., Bhullar, M. S. , Upadhyay R. and Martynenko A. , (2020) "IoT, big data and artificial intelligence in agriculture and food industry," in IEEE Internet of Things Journal, doi: 10.1109/JIOT.2020.2998584.

Murphy, K. P. (2002). Dynamic bayesian networks: representation, inference and learning. PhD Dissertation. Berkeley: University of California.

Norman, D. A, and Draper, S.W.. User-Centered System Design: New Perspectives on HumanComputer Interaction. Erlbaum, Hillsdale, NJ, 1986.

North, C., Chang, R., Endert, A., Dou, W., May, R., Pike, B. and Fink, G., 2011. Analytic provenance: process+ interaction+ insight. In CHI'11 Extended Abstracts on Human Factors in Computing Systems (pp. 33-36).

Pearl J. (1988). Probabilistic Reasoning in Intelligent systems: Networks of Plausible Inference. Morgan Kaufmann, San Diego. 552p.

Pedregosa F.; Varoquaux G.; Gramfort A.; Michel V.; Thirion B.; Grisel O.; Blondel M.; Prettenhofer P.; Weiss R.; Dubourg V.; Vanderplas J.; Passos A.; Cournapeau D.; Brucher M.; Perrot M.; and Duch- esnay E., 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

Perrot, N., De Vries, H., Lutton, E., Van Mil, H.G.J., Donner, M., Tonda, A., Martin, S., alvarez, A., Bourgine, P., van der Linden, E. Axelos, M. (2016). Some remarks on computational approaches towards sustainable complex agri-food systems. Trends in Food Science and Technology, 48, 88-101.

Perrot, N., Baudrit, C. (2012) Intelligent Quality control systems in food processing based on fuzzy logic in Robotics and automation in the food industry: Current and future technologies. Edited by D Caldwell, Italian Institute of Technology, Italy, December 2012 ISBN 1 84569 801 0, Woodhead Publishing Series in Food Science, Technology and Nutrition No. 236

Perrot, N., Baudrit, C., Brousset, J. M., Abbal, P., Guillemin, H., Perret, B., & Picque, D. (2015). A Decision support system coupling fuzzy logic and probabilistic graphical approaches for the agri-food industry: prediction of grape berry maturity. PloS one, 10(7), e0134373.

Perrot N.; Ioannou I.; Allais I.; Curt C.; Hossenlopp J.; and Trystram G., 2006. Fuzzy concepts applied to food product quality control: A review. Fuzzy sets and systems, 157, no. 9, 1145–1154.

Perrot, N., Trelea, I., Baudrit, C., Trystram, G., & Bourgine, P. (2011). Modelling and analysis of complex food systems: state of the art and new trends. Trends in Food Science & Technology, 22(6), 304-314.

Perrot N.; Trystram G.; Guely F.; Chevrie F.; Schoeset- ters N.; and Dugre E., 2000. Feed-back Quality Control in the Baking Industry Using Fuzzy Sets. Journal of food process engineering, 23, no. 4, 249–279.

Pirolli, P. and Card, S., 2005, May. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In Proceedings of international conference on intelligence analysis (Vol. 5, pp. 2-4).

Ragan, E.D., Endert, A., Sanyal, J. and Chen, J., 2015. Characterizing provenance in visualization and data analysis: an organizational framework of provenance types and purposes. IEEE transactions on visualization and computer graphics, 22(1), pp.31-40.

Raynal, M., Debord, C., Guittard, S., & Vergnes, M. (2010). Epicure, a geographic information decision support system risk assessment of downy and powdery mildew epidemics in Bordeaux vineyards. Sixth international workshop on the grapevine downy and powdery mildew (p. 144-146). Bordeaux: INRA-ISV.

Riou, C. (1994). Le déterminisme climatique de la maturation du raisin: application au zonage de la teneur en sucre dans la Communauté Européenne. Office des Publications Officielles des Communautés Européennes.

Rosenblatt F., 1958. The perceptron: A probabilistic model for information storage and organization in the brain. Psychological review, 65, no. 6, 386.

Sablani S.; Marcotte M.; Baik O.; and Castaigne F., 1998. Modeling of simultaneous heat and water transport in the baking process. LWT-Food Science and Technology, 31, no. 3, 201–209.

Savoye I.; Trystram G.; Duquenoy A.; Brunet P.; and Marchin F., 1992. Heat and mass transfer dynamic modelling of an indirect biscuit baking tunnel-oven. Part I: Modelling principles. Journal of food engi- neering, 16, no. 3, 173–196.

Serazetdinova L, Garratt J, Baylis A, Stergiadis S, Collison M, Davis S. (2019)How should we turn data into decisions in AgriFood? J Sci Food Agric; 99(7):3213-9.

Sermanet P.; Eigen D.; Zhang X.; Mathieu M.; Fergus R.; and LeCun Y., 2013. Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:13126229.

Silver D.; Huang A.; Maddison C.J.; Guez A.; Sifre L.; Van Den Driessche G.; Schrittwieser J.; Antonoglou I.; Panneershelvam V.; Lanctot M.; et al., 2016. Mas- tering the game of Go with deep neural networks and tree search. nature, 529, no. 7587, 484–489.

Trystram G.; Fahloul D.; Duquenoy A.; and Al- lache M., 1993. Dynamic modelling and simulation of the biscuit baking oven process. Comput- ers & Chemical Engineering, 17, S203–S208. doi: 10.1016/0098-1354(93)80230-k. URL https://doi.org/10.1016/0098-1354(93)80230-k.

Valdez, A.C., Ziefle, M. and Sedlmair, M., 2017. Priming and anchoring effects in visualization. IEEE transactions on visualization and computer graphics, 24(1), pp.584-594.

Vermersh, P. (2006). L'entretien d'explicitation. Paris: ESF.

Vilani c. (2017). FOR A MEANINGFUL ARTIFICIAL INTELLIGENCE. report: https://www.aiforhumanity.fr/pdfs/MissionVillani_Report_ENG-VF.pdf

Wade P., 1988. Biscuit, cookies and crackers: The principles of the craft. Vol. I.

Wall, E., Blaha, L.M., Franklin, L. and Endert, A., 2017, October. Warning, bias may occur: A proposed approach to detecting cognitive bias in interactive visual analytics. In 2017 IEEE Conference on Visual Analytics Science and Technology (VAST) (pp. 104-115). IEEE.

Xiong W.; Wu L.; Alleva F.; Droppo J.; Huang X.; and Stolcke A., 2017. The Microsoft 2017 Conversational Speech Recognition System. Tech. rep.

Figures

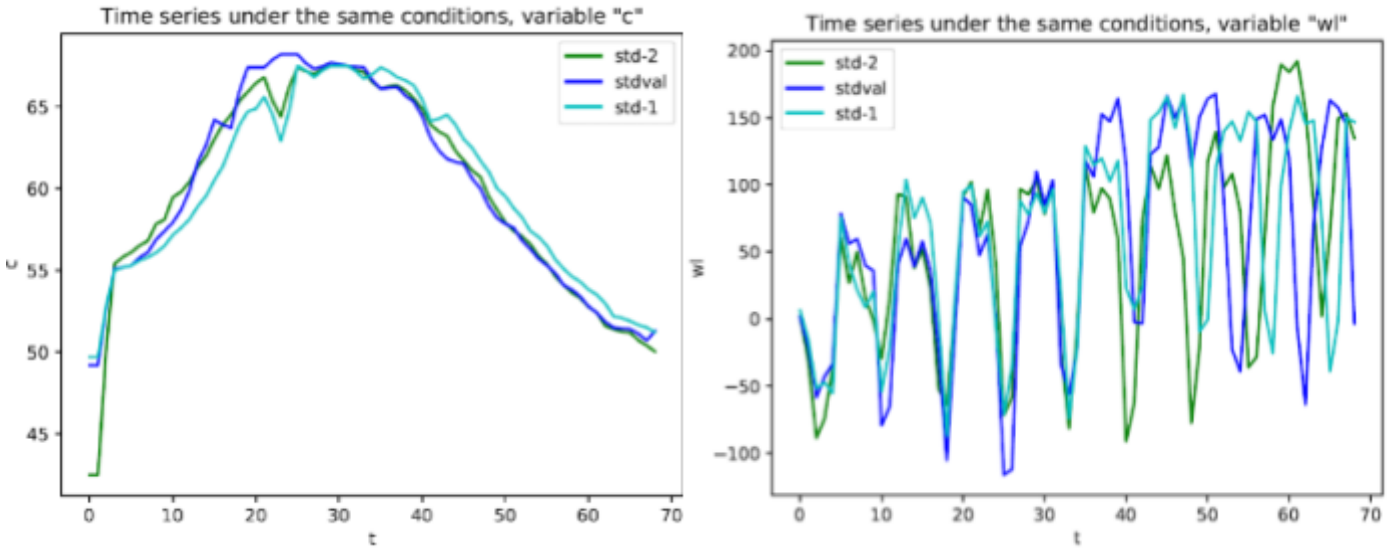

Figure 1 : Three time series describing the change of color (c) and weight loss (wl) during biscuit baking. It is noticeable how, even though the three datasets have been collected under the same conditions, there are relevant differences in the values.

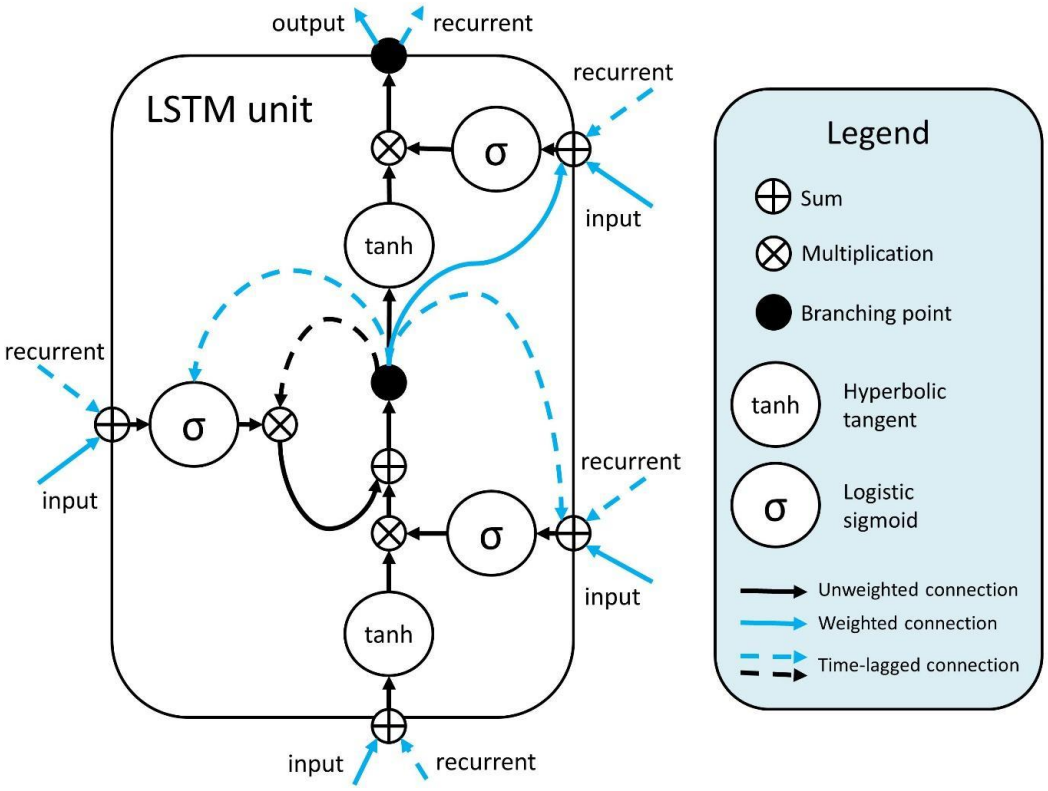

Figure 2 : Log visualization widget prototype (integrated into the visualization tool) showing a single machine-learned storyline. Nodes indicate scatterplot selections and colour indicates scenarios.

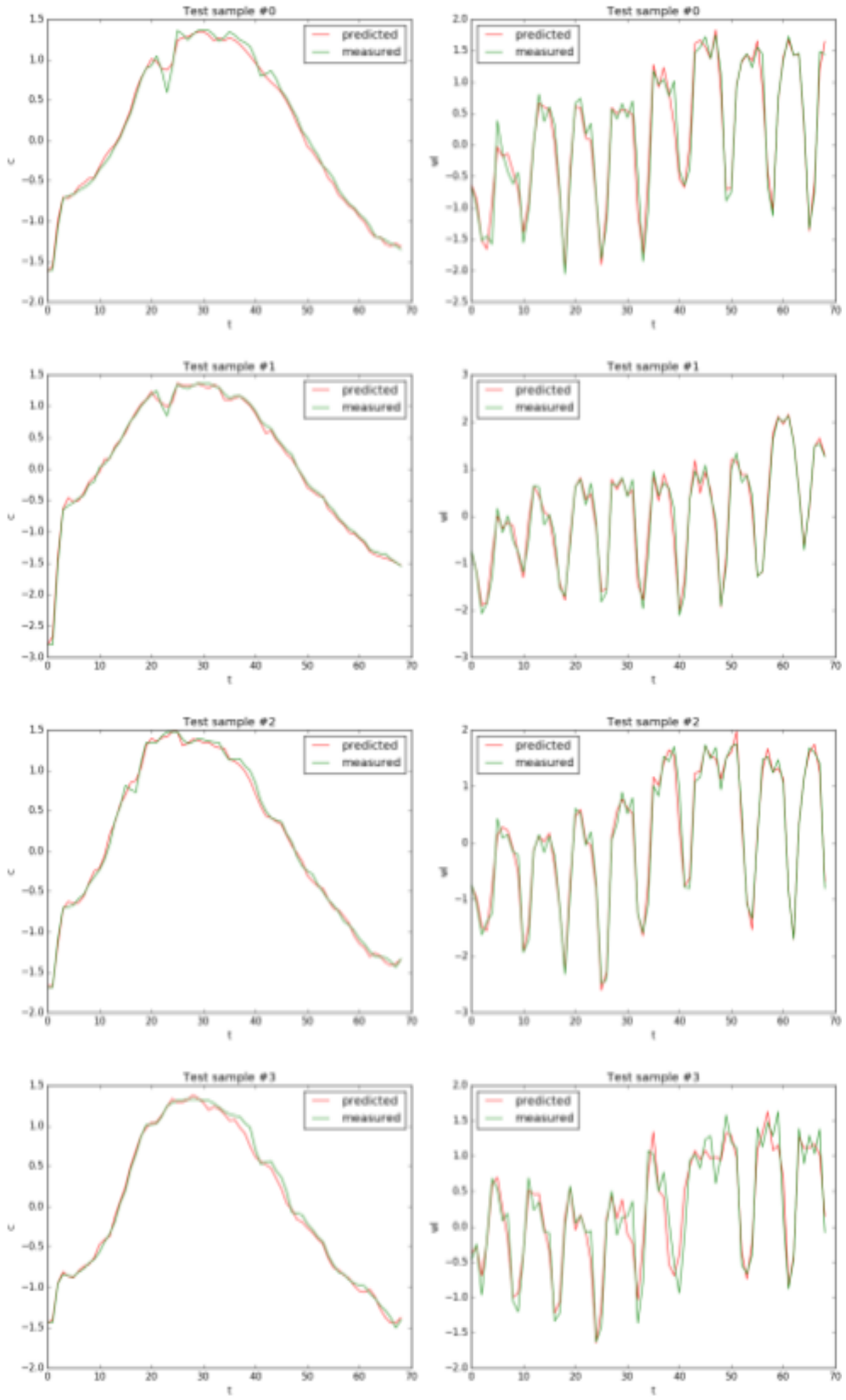

Figure 3: Models' predictions versus measured values for test samples, unseen during the training process. The model is able to remarkably fit the data, even for parts that might be naively believed to be noisy. The scale is different from the previous plots, as all variables have been normalized.

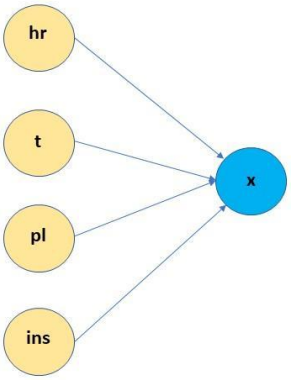

Figure 4 : Network structure describing the interaction between the climatic variables and the output variables: physicochemical and sensory variables quoted x in this figure.

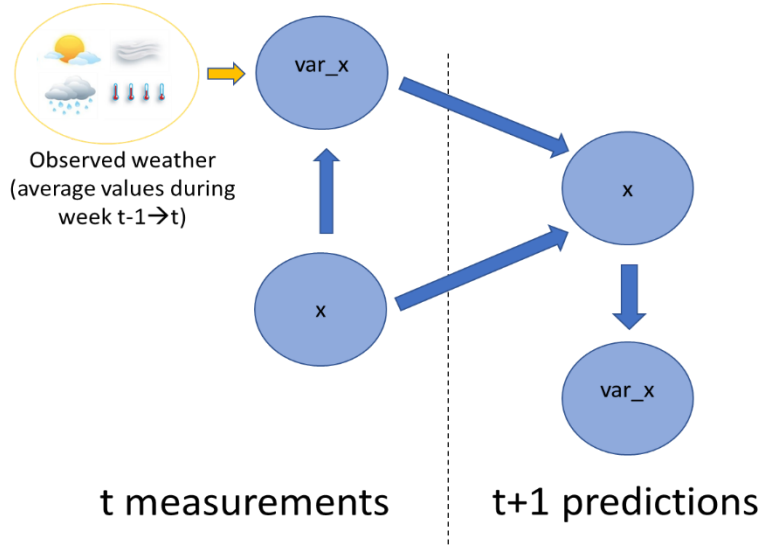

Figure 5: the dynamical representation of the DBN. It is represented in the form of two generic slices that can be developed on several slices representing the different step times. DBNs assume the first-order Markov property which means that the parents of a variable in time slice t must occur in other slices and the conditional probabilities are time-invariant. The slice representing the time t (t measurements) is concerned at the beginning of the iterations by variables that are measured at time t0. The consecutive slice: time t+1 is dedicated to predictions. If several slices are added, for example t, t+1 and t+2, it starts at t0 with an initialization where variables are measured, followed by two slices predicted t+1 and t+2, with t+2 predicted on the basis of the prediction of t+1.

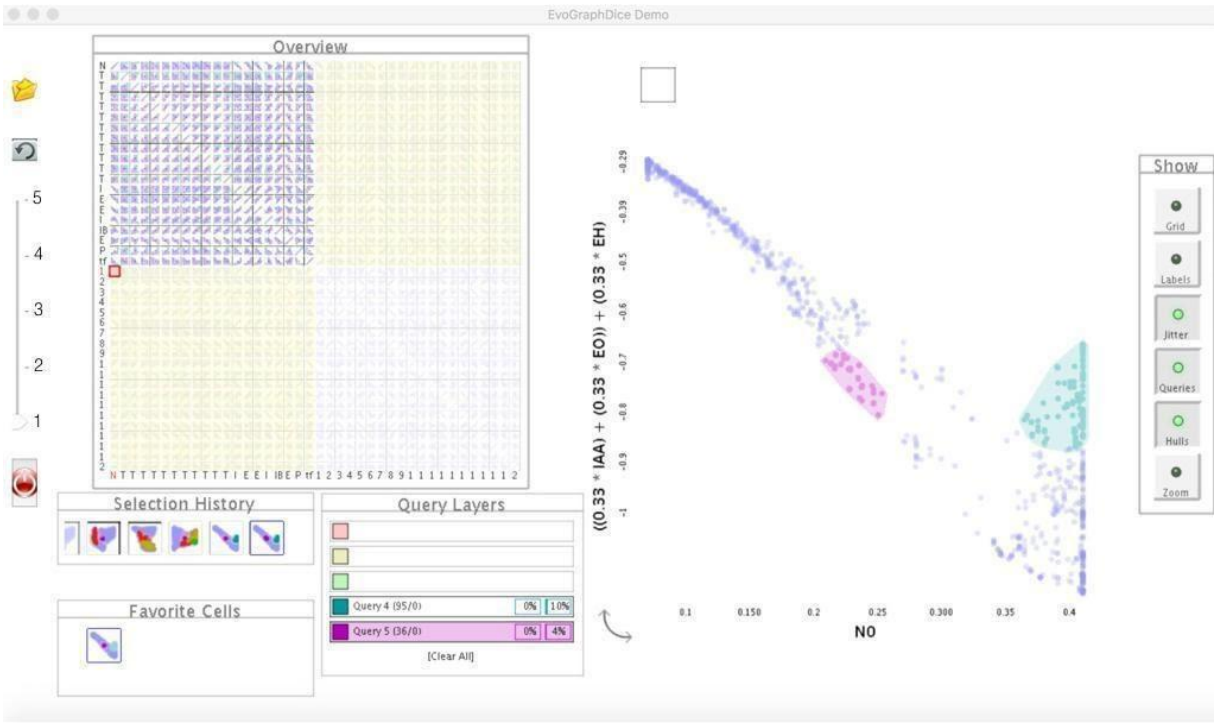

Figure 6 : The scatterplot matrix visualization tool used in our study (Boukhelifa et al, 2019).

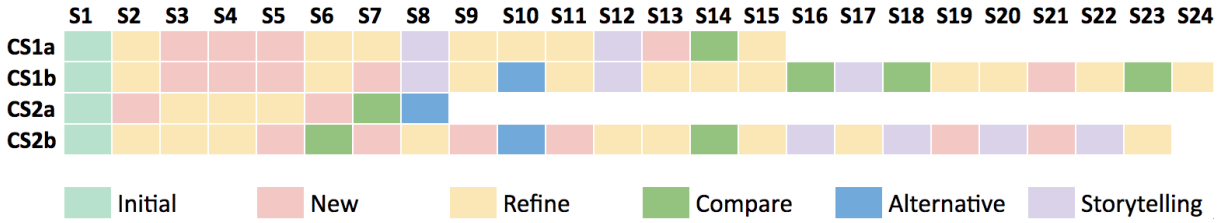

Figure 7 : User exploration scenario sequences and types for four sessions with domain experts, as identified from manual video coding and qualitative analysis.

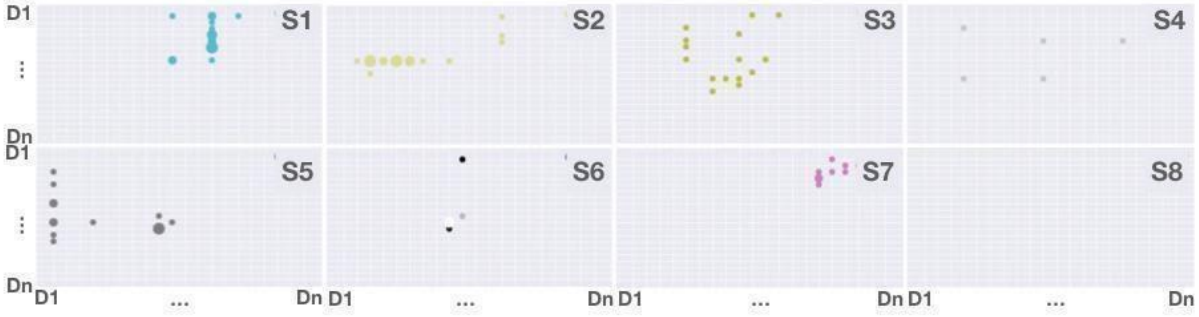

Figure 8 : The first eight analysis scenarios of the wine use case S1-8 as identified from manual video coding. Each grid corresponds to one scenario, rows and columns are data dimensions D1-n including any combined dimensions (created manually or automatically). Circles indicate scatterplot visits, and their size the frequency of visits. Analysis scenarios are usually focused on one area of the search space.

Figure 9 : Results of clustering (bottom timeline), top timeline is the ground truth. Dots are scatterplot selections, and color corresponds to scenarios S1-14.

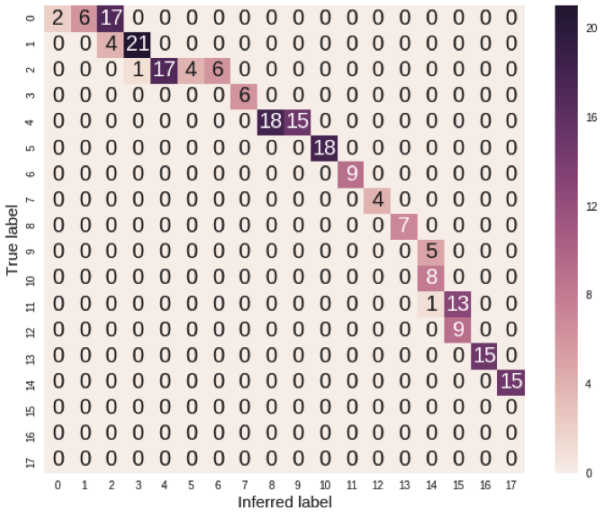

Figure 10 : Confusion matrix for the Hidden Markov Model, for the wine use case.

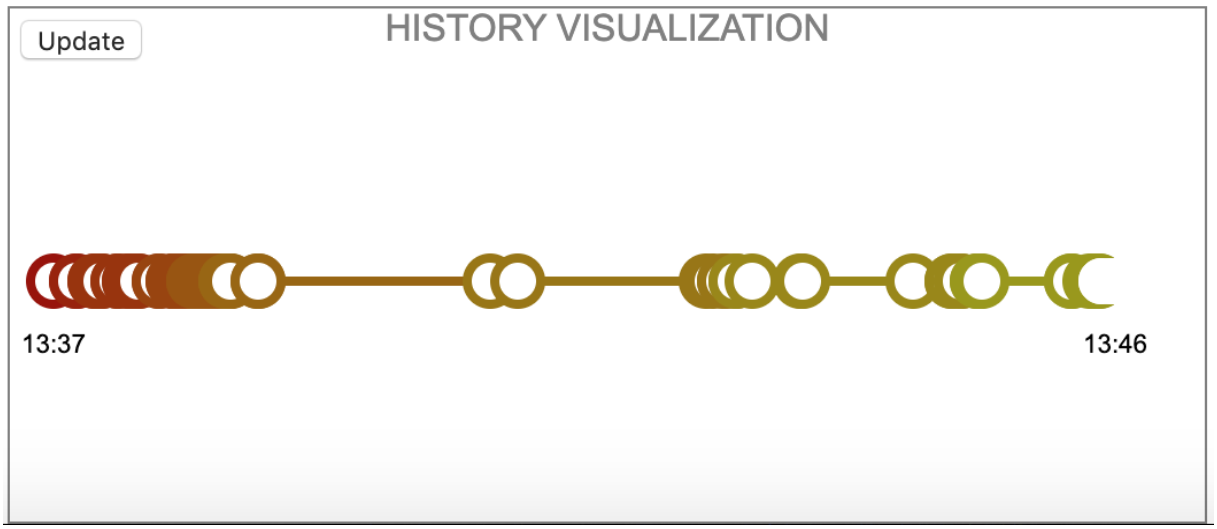

Figure 11 : Log visualization widget prototype (integrated into the visualization tool) showing a single machine-learned storyline. Nodes indicate scatterplot selections and colour indicates scenarios.

Tables

Table 1: Summary of the 16 time series on biscuit cooking gathered by United Biscuits. During the experiments, the temperature in different zones of the oven is changed, in order to explore several possible behaviors.

ID	Training?	Heat flux (W/m^2)				
		z1	z2	z3	z4	z5
std-1	yes	2500	3500	4000	4000	2000
std-2	yes	2500	3500	4000	4000	2000
stdval	no	2500	3500	4000	4000	2000
T1-1	yes	4000	3500	4000	4000	2000
T1-2	yes	4000	3500	4000	4000	2000
T1val	no	4000	3500	4000	4000	2000
T2-1	yes	2500	3500	4000	4000	3000
T2-2	yes	2500	3500	4000	4000	3000

ID	Training?	Heat flux (W/m^2)				
		z1	z2	z3	z4	z5
T3-1	yes	2500	3500	6000	4000	2000
T3-2	yes	2500	3500	6000	4000	2000
T3val	no	2500	3500	6000	4000	2000
T4-1	yes	2500	3500	4000	6000	2000
T4-2	yes	2500	3500	4000	6000	2000
T5-1	yes	2500	5000	1000	5000	2000
T5-2	yes	2500	5000	1000	5000	2000
T5val	no	2500	5000	1000	5000	2000

Table 2: Results of the leave-one-out cross-validation.

ID	R2	MSE	ID	R2	MSE
std-1	0.9557	0.0491	*T3-1*	0.6644	0.2535
std-2	0.9789	0.0251	*T3-2*	0.4579	0.3678
stdval	0.9785	0.0280	*T3val*	0.0810	1.1728
T1-1	0.9592	0.0279	*T4-1*	0.4740	0.5685
T1-2	0.5561	0.4572	*T4-2*	0.9718	0.0311
T1val	0.4674	0.5844	*T5-1*	0.9584	0.0497
T2-1	0.6942	0.2956	*T5-2*	0.9859	0.0186
T2-2	0.1237	0.8653	*T5val*	0.9645	0.0241

Table 3: Discretization of the physico-chemical variables, fixed by experts for the variation var_X and calculated by optimization for the variable X.

	Discretization variable X	Discretization Var_X
s (sugar)	• Class 0 = [[∞,156.9], • Class 1 = [156.9,182.86] • Class 2 = [182.86,201.8] • Class 3 = [201.8,210] • Class 5 = [210,220] • Class 6 = [220,230] • Class 7 = [230,240] • Class 8 = [240,+∞]],	• Classe 0 = $[0,\ 12]$, • Classe 1 = $[12,\ 20]$ • Classe 2 = $[20,\ 35]$ • Classe 3 = $[35, +\ \infty]$
ac (total acidity)	• Class 0 = $[-\ \infty,\ 5.47]$ • Class 1 = $[5.47,\ 6.33]$ • Class 2 = $[\ 33,\ 7.94]$ • Class 3 = $[7.94,\ +\ \infty]$	• Class 0 = $[-\ \infty,\ -\ 1.5]$ • Class 1 = $[-\ 1.5,\ -\ 1]$ • Class 2 = $[-\ 1,\ -\ 0.6]$ • Class 3 = $[-\ 0.6,\ 0]$
ac_m (malic acid)	• Classe 0 = $[-\ \infty,\ 3.66]$ • Classe 1 = $[3.66,\ 4.6]$ • Classe 2 = $[4.6,\ 5.68]$ • Classe 3 = $[5.68,\ 6.88]$ • Classe 4 = $[6.88,\ +\ \infty]$	• Class 0 = $[-\ \infty,\ -\ 2.5]$ • Class 1 = $[-\ 2.5,\ -\ 1.5]$ • Class 2 = $[-\ 1.5,\ -\ 0.75]$ • Class 3 = $[-\ 0.75,\ -\ 0.5]$ • Class 4 = $[-\ 0.5,\ 0]$

Table 4: Results of prediction for the four variables for two time steps of prediction (anticipation of two weeks): 1 and 2, labeled X_1 and X_2. RMSE tolerance: for s: 12g/l; for ac: 0.5 g/l; for ac_m: 0.5 g/l

Variable	RMSE	RRMSE % ac = [3.4,12.5]; ac_m = [1.7,10] ; s = [144,271.8]
ac_1 (g/l)	0.536	6
ac_2 (g/l)	0.648	7
ac_m_1 (g/l)	0.825	9
ac_m_2 (g/l)	0.867	10
s_1 (g /l)	11.37	8
s_2 (g/l)	12.87	10

Variable	RMSE	Abs(Pred-Obs)<0.5 %	Abs(Pred-Obs)<0.25 %
IntGloAro_1	0.73	47.5	30
IntGloAro_2	0.52	72.5	50