Concept Drift Estimation with Graphical Models

Drift Estimation with Graphical Models

Luigi Riso[1,*] and Marco Guerzoni[2]

[1]University of Turin
[2]DEMS, University of Milan-Bicocca

January 2020

Abstract

This paper deals with the issue of concept drift in supervised machine learning. We make use of graphical models to elicit the visible structure of the data and we infer from there changes in the hidden context. Differently from previous concept-drift-detection methods, this application does not depend on the supervised machine learning model in use for a specific target variable, but it tries to assess the concept drift as independent characteristic of the evolution of a data set. Specifically we investigate how a graphical model evolves by looking at the creation of new links and the disappearing of existing ones in different time periods. The paper suggests a method that highlights the changes and eventually produce a metric to evaluate the stability over time. The paper evaluate the method with real world data on the Australian Electric market.

1 Introduction

In the last decades, both the increasing availability of digitised information and the improvement in the algorithms made the use of machine learning widespread across different industries. Specifically, supervised machine learning became a standard tool for predicting key information in various organization processes such as for instance to mention a few risk default of firms and individual, fraudulent claims, customers churn, and machine failures. The assessment of model uncertainty within a supervised machine learning exercise is based on testing the goodness on a test-set, whose observations have not been employed in the model training. This practice allows for flexibility in the choice of the model and prevents from the risk of over-fitting. However, this analysis relies on the assumption the data generating structure is similar between the test-set and the future observations. While this assumption is rarely debatable in physical process, social process change overtime and a model trained on past data might

*To whom correspondence should be addressed. Department of Economics and Statistics, Lungo Dora Siena 100A, 10122, Turin, Italy. Email: luigi.riso@unito.it

see a deterioration of its predictive power Gama et al. [2014]. This phenomenon is known as concept- or model- drift and describes the situation in which there exists an hidden context of data generative structure, that is any effect of the outcome variable not captured by the model features, which changes over time abruptly, incrementally, or periodically Widmer and Kubat [1996], Webb et al. [2016]. Scholars addressed this issue and developed a battery of techniques for concept drift detection and early detection. As reviewed in Klinkenberg and Joachims [2000] and Elwell and Polikar [2011], traditional techniques in concept drift detection typically relies by adopting different time windows or size of the training data [Klinkenberg and Renz, 1998] or in explaining how the weights of different features change overtime in the outcome prediction [Klinkenberg and Renz, 1998, Taylor et al., 1997, Klinkenberg, 2004]. A recent review [Althabiti and Abdullah, 2020] surveys also methods which can also deal with model update with stream data [Bose et al., 2011]. However, all of these techniques rely on some sort of computation or statistical comparison of the changes on classification error overtime and from this evidence they deduct the presence of concept drift [Widmer and Kubat, 1996]. In this paper, we approach the problem from a different angle. We make use of graphical models [Lauritzen, 1996] to elicit the visible structure of the data and we infer from there changes in the hidden context with use of statistical measure. Thus, differently from previous concept-drift-detection methods, this application does not depend on the supervised machine learning model in use, but it tries to assess the concept drift as an independent characteristic of the evolution of a data set.

2 Graphical Models, background

Consider a dataset, composed by p random variables $\mathbf{X}_p$, where p can be divided in d discrete and q continuous random variables. Graphical Models are a method to display the conditional independence relationships between random variables in a dataset. The conditional independence relationships can be showed as a networks of variables with an undirected graph, that is mathematical object $G = (V, E)$, where V is a finite set of nodes, one-to-one correspondence with the p random variables present in the dataset, and $E \subset V \times V$, is a subset of ordered couples of V. Links represent interactions between the nodes. If a link between two nodes is absent, the two variables represented by the node are conditional independent given the dependence of the remaining variables.

Pairwise, local and global Markov properties are the connections between graph theory and statistical modeling [Lauritzen, 1996]. As said before, there exist a one-to-one correspondence between the variables and the nodes in the graph and, for this reason, the sets of nodes is Δ and Γ, where $V = \{\Delta \cup \Gamma\}$. Let the corresponding random variables be $(\mathbf{Z}, \mathbf{Y})$ where $\mathbf{Z} = (Z_1, ..., Z_d)$ and $\mathbf{Y} = (Y_1, ..., Y_q)$ and a i-observation be $(\mathbf{z}_i, \mathbf{y}_i)$. This means that $\mathbf{z}$ is a d-tuple containing the values of discrete variables, and $\mathbf{y}$ is a real vector of length q. Our interest is to estimate the joint probability distribution $P(\mathbf{x})$ for the random variables $(\mathbf{Z}, \mathbf{Y})$ to build a conditional (undirected) graph from the data. A product approximation of $P(\mathbf{x})$ is defined to be a product of several of its component distribution of lower order $P_a(\mathbf{x})$. As suggest Chow and Liu

[1968], we can consider the class of second-order approximation, i.e:

$$P_a(\mathbf{x}) = \prod_{i=1}^{p} P(x_i, x_{j(i)}), \quad 0 \leqslant j(i) \leqslant p \tag{1}$$

where $(j_1, ..., j_p)$ is an unknown permutation of integers $(1, 2, ..., p)$, where p=d+q. Chow and Liu in [6] proved that for discrete random $\mathbf{Z}$, the problem of finding the goodness of approximation between $P(\mathbf{z})$ and $P_a(\mathbf{z})$ with the minimization of the closeness measure:

$$I(P, P_a) = \sum_{z} P(\mathbf{z}) \log \frac{P(\mathbf{z})}{P_a(\mathbf{z})} \tag{2}$$

where $\sum_z P(\mathbf{z})$ is nothing more than the sum over all levels of discrete variables. The equation (2), is equivalent to maximizing the total branch (link) weight $\sum_{i=1}^{p} I(z_i, z_{j(i)})$, where:

$$I(z_i, z_{j(i)}) = \sum_{z_i, z_{j(i)}} P(z_i, z_{j(i)}) \log \left(\frac{P(z_i, z_{j(i)})}{P(z_i) P(z_{j(i)})} \right) \tag{3}$$

The task is to build a tree or forest (different trees) of maximum weight. We make use of the Kruskal's algorithm [Kruskal, 1956] to compute trees with the minimum of total length. To choose a tree of maximum total branch weight, we first index the $d(d-1)/2$ according to decreasing weight. This algorithm starts from a square weighted matrix $d \times d$, where a weight for a couple of variables (Z_i, Z_j) is given by the mutual information $I(z_i, z_j)$. In the real world the probability distributions are no given explicitly, for this reason we have to estimate the mutual information. Let $\mathbf{z}^1, \mathbf{z}^2, ..., \mathbf{z}^N$ be independent samples of finite discrete variables $\mathbf{z}$. Then the mutual information is given by:

$$\hat{I}(z_i, z_j) = \sum_{u,v} f_{u,v}(i,j) \log \frac{f_{u,v}(i,j)}{f_u(i) f_v(j)}, \tag{4}$$

where $f_{u,v}(i,j) = \frac{n_{uv}(i,j)}{\sum_{uv} n_{uv}(i,j)}$ and $n_{uv}(i,j)$ is the number of samples such that their ith and jth components assume the values of u and v, respectively. It was showed that with this estimator we also maximize the likelihood for a dependence tree Chow and Liu [1968]. This procedure works only with the discrete random variables, but it can be extended to data with both discrete and continuous random variables Edwards et al. [2010]. To present this extension, we have to consider the distributional assumption of our random variables $\mathbf{X}$ i.e. the distribution of $\mathbf{Y}$ given $\mathbf{Z} = \mathbf{z}$ is a multivariate normal $\mathcal{N}(\mu_i, \Sigma_i)$ so that both the conditional mean and covariance may depend on ith component.

We distinguish between homogenous and heterogeneous case, if Σ depend on i we are in the homogenous case, otherwise we are in the heterogeneous case. More details this conditional Gaussian distribution can be found in Sudderth et al. [2004]. Before to apply the Kruskal's algorithm, we need to find an estimator of the mutual information $I(z_u, y_v)$ between each couple of variables in the mixed case. For a couple of variables

(Z_u, Y_v) we can write the sample cell count, mean, and finally the variance, respectively, $\{n_i, \bar{y}_v, s_i^{(v)}\}_{i=1,\dots,|Z_u|}$. An estimator of mutual information, in the homogenous case is give by:

$$\hat{I}(z_u, y_v) = \frac{N}{2} \log\left(\frac{s_0}{s}\right), \tag{5}$$

where $s_0 = \sum_{k=1}^{N}(y_v^{(k)} - \hat{y}_v)/N$ and $s = \sum_{i=1}^{|Z_u|} n_i s_i / N$. $k_{z_u,y_v} = |Z_u|-1$ are the degree of freedom associated to the mutual information in the homogenous case.
While,i n the heterogeneous case an estimator of the mutual information is equal to

$$\hat{I}(z_u, y_v) = \frac{N}{2} \log(s_0) - \frac{1}{2} \sum_{i=1,\dots,|Z_s|} n_i \log(s_i) \tag{6}$$

with $k_{z_u,y_v} = 2(|Z_u|-1)$ degrees of freedom. According Edwards et al. [2010] it is useful to use either $\hat{I}^{AIC} = \hat{I}(x_i, x_j) - 2k_{x_i,x_j}$ or $\hat{I}^{BIC} = \hat{I}(x_i, x_j) - \log(n)k_{x_i,x_j}$, where k_{x_i,x_j} are the degree of freedom, to avoid inclusion of links not supported by the data. This aspect is suggested by the algorithm to find the best spanning tree, because it stop when it has added the maximum number of edges. Furthermore the algorithm avoid inside the tree a forbidden path. The definition of forbidden path is a path between tow not adjacent discrete nodes which passes through continuous nodes [de Abreu et al., 2009]. However, we can start from the best spanning tree and determine the best strongly decomposable graphical model. A strongly decomposable graphical model whose graph neither contains cycles of length more than three nor forbidden path. Strongly decomposable model is an important class of model that can be used to analyze mixed data. This class restrict the class of possible interaction model which would be to huge to be explored [Abbruzzo and Mineo, 2015]. The graph build to find the best spanning tree, can be see with a symmetric adjacency matrix AM, with dimension $V \times V$, in which each element takes value of 1 if an edge exists between two of the V variables, and zero otherwise. Elements in the main diagonal are zeros, since self-loops are not allowed.

3 A measure of dynamic stability as proxy for the model drift

Considering the additional dimension of time t to the dataset of N observations and p variables as a tensor X with dimension $(N \times p \times T)$, we are interested in modeling the evolution of the joint probability $P(X_1, ..., X_p)$ over T time periods. In other words, considering the graph G, with $V = p$ vertices of the maximum spanning tree with mutual information as express in Eq. 6 for each period $t = 1, ..., T$ and the corresponding T adjacency matrices AM_t, the aim of the paper is to describe how the graphs, as represented by their adjacency matrix AM_t with $t = 1, 2, ...T$, change over time.

3.1 Transition Matrix Processes

In order to accomplish this task, we analyse the transition process which connects the original adjacency matrix AM_1 to any adjacency matrices in a subsequent period AM_T. We first introduce a function which maps any possible state of AM_t into a transition matrix $TM = f(AM_t)$ with $t = 1, 2, 3, ..T$, noted TM_T, of dimension $V \times V$. Its generic element $w_{i,j}$ registers all possible states of dependence of any couple of variable V_i and V_j in T periods. Specifically, the function takes the following form:

$$TM_t = \sum_{t=1}^{T} 2^{(T-t)} AM_t \tag{7}$$

For the sake of clarity, the following paragraph describes the process up to $T = 3$ and, thereafter, generalizes for T periods.

	AM_1	AM_2	AM_3	TM_3
	0	0	0	0
	1	0	0	4
	1	1	0	6
	1	1	1	7
	0	1	0	2
	0	0	1	1
	1	0	1	5
	0	1	1	3

Table 1: All possible AM_t values for two nodes i and j and the resulting $w_{i,j}$ in TM_T function for $T = 3$

As a starting point, in $t = 1$ the transition matrix TM_1 is equal to the adjacency matrix AM_t, where $w_{i,j;1} = 0$ means that the i-node and j-node are not connected, while when $w_{i,j;1} = 1$ means that the i-node and j-node are connected. At $t = 2$ existing links can persist or not, while non-existing links can appears or not. From Eq. 7,

$$TM_2 = 2 \times AM_1 + AM_2 \tag{8}$$

Thus, TM_2 maps any possible evolution of connections $w_{i,j;2}$ with values $\{0, 1, 2, 3\}$. When V_i and V_j are never connected,that is $AM_{i,j;t=1} = AM_{i,j;t=2} = 0$, then $TM_{i,j;2} = 0$. If V_i and V_j stay connected, that is $AM_{i,j;t=1} = AM_{i,j;t=2} = 1$, then $w_{i,j;2} = 3$. For $AM_{i,j}$ changing from 0 in $t = 1$ to 1 in $t = 0$ and viceversa, we have $w_{i,j;2} = 2$ and $w_{i,j;2} = 1$, respectively. At time $t = 3$ the possible evolution of AM can be described has 8 levels, since it can be either 0 or one three times, given by:

$$TM_3 = 2^2 \times AM_1 + 2^1 \times AM_2 + 2^0 \times AM_3 \tag{9}$$

Table 1 summarizes all possible combinations between two nodes of binary values of the AM_t in the three periods, mapped on TM_3. Generally, for time T we can derive Eq. 7:

$$
\begin{aligned}
TM_2 &= 2 \times AM_1 + AM_2 \\
TM_3 &= 2 \times TM_{1,2} + AM_3 \\
TM_3 &= 2 \times (2 \times AM_1 + AM_2) + AM_3 \\
TM_3 &= 2^2 \times AM_1 + 2^1 \times AM_2 + 2^0 \times AM_3 \\
TM_3 &= \sum_{t=1}^{T} 2^{(3-t)} AM_t \\
&\dots \\
TM_T &= \sum_{t=1}^{T} 2^{(T-t)} AM_t
\end{aligned}
\tag{10}
$$

In general, the value of the generic element $w_{i,j;t} \in \mathcal{W} \subset \mathbb{N}$ of TM_t can be considered as a discrete random variable with density $f(w_{i,j;t})$

$$
f(w_{i,j;t}) = P(\mathcal{W}_{i,j;t} = w_{i,j;t}), \quad t = 2, ..., T \tag{11}
$$

Thus, $w_{i,j;t}$ represents the evolution of the connection between i-node with j-node at time T, for each node V. The numerosity of the set $\mathcal{W}_{i,j;T} = \{0, 1, 2.., 2^T - 1\}$ is 2^T.

3.2 From the transition process to stability

The main idea of the paper is to consider as a proxy for the model drift the appearance or disappearance of connections between nodes, that is changes of the conditional independence structure of a dataset over time. For this reason, we are specifically interested in two specific levels. The one describing the state of the word in which a connection between two nodes never exists, that is $AM_{i,j;t} = 0 \ \forall \ t$ and the one describing a stable connection over time, that is $AM_{i,j;t} = 1 \ \forall \ t$. For the case $T = 3$, the two cases map into $w_{i,j;3} = 0$ and $w_{i,j;3} = 7$, as showed in Table 1. In general for a generic T, we have a stability of connections when connections are always absent, with $w_{i,j;T} = 0$, or always existing, with $w_{i,j;T} = 2^T - 1$. This transition process is a partition process (Fig 1) of the set of $\mathcal{V}$ possible connections between the $\mathbf{V}$ nodes in the undirected graph: $\mathcal{V} = \frac{V(V-1)}{2}$. Each transition in time t generates a subsequent partition of $\mathcal{V}$, one of whose will always contain elements for which $w_{i,j;t} = 0$ or always $w_{i,j;t} = 2^t - 1$. This *transition processes* is a special case of the Tail-free processes [Jara and Hanson, 2011]. Consider a sequence $\mathcal{T}_0 = \{\mathcal{V}\}$, $\mathcal{T}_1 = \{A_0, A_1\}$, $\mathcal{T}_2 = \{A_{00}, A_{01}, A_1\}$, and so on, of measurable partitions of the $\mathcal{V}$ elements, obtained by slitting every set in the preceding partition into two new sets for the node on left and maintain the same node for the others.

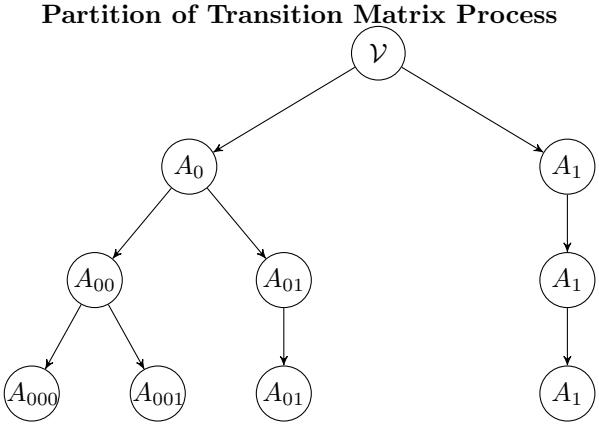

Figure 1: Representation of *Transition Matrix process* with Tail-free processes

Specifically, at each time t we can partition the elements between stable and unstables ones. Fig. 1 shows a tree diagram that represents the distribution of mass over time $\mathcal{V} = A_0 \cup A_1 = (A_{00} \cup A_{01}) \cup A_{10}$ of the elements at each time. A_0 contains elements for $w_{i,j,2} = 0, 3$, that is stable connections while A_1, the remaining ones. At the subsequent period, A_0 is partitioned between A_{00}, in which connection remain stable with $w_{i,j,3} = 0, 7$,while $A_{01} = 1, 6$ and A_1 the remaining ones.

Clearly, every partition is composed by the union of all possible evolution of the connection given by the levels of $\mathcal{W}$, and, by construction, there is always a partition with elements $w_{i,j;t} = 0$ and $w_{i,j;t} = 2^t - 1$, that containing stable links between the i-node and the j-node until time t. We describe this process as a variable $Y_{i,j;t}$ with values :

$$Y_{i,j;t} = \begin{cases} y_{i,j;t} = 1 & \text{if } w_{i,j;t} = 0 \vee w_{i,j;t} = 2^t - 1 \\ y_{i,j;t} = 0 & \text{otherwise} \end{cases}, \quad t = 2, ..., T \tag{12}$$

Thus, $Y_{i,j;t}$ is indicate persistent status of dependence over time $Y_{i,j;k} = 1$ or not $Y_{i,j;k} = 0$. Be Y_t the vectorization of $Y_{i,j;t}$, $vec(Y_{i,j;t}) = Y_t$ with length $\mathcal{V} = \frac{V \times (V-1)}{2}$, that is at each time we observe the stability of the $\mathcal{V}$ connection between each possible pair of nodes. The structure of the *transition matrix process* depend by the spanning forest at time $t = 1$, and for each period we have a partition of $\mathcal{V}$ given by $\mu_t = \sum_{i=1}^{N} Y_{i,t}$ with $t = 1, ..., T - 1$.

Therefore, we pool together the $T - 1$ periods and define *Stability*, the resulting variable Y with length $n = \mathcal{V} \times (T - 1)$. *Stability* is the cornerstone of our strategy to estimate an empirical measure of model drift.

3.3 The stability index

In this section we introduce the *Stability* as a latent variable, which capture stability of connection of a graph overtime.
Consider the following variable with same length $i = 1, ..., n$:

- Y, *Stability* as defined above
- $W = vec(TM_{i,j,t})$ that is the vectorization of the value $w_{i,j;t}$ of TM.
- T the corresponding time for each Y_i.

We build a dataset with this variables and call it $\mathbf{D}$. Note that by construction the observations of $\mathbf{D}$ is exchangeable since we have built $\mathbf{D}$ respecting the temporal period of the *adjacent matrices*, thus:

$$P(\mathbf{D}_1, ..., \mathbf{D}_n) = P(\mathbf{D}_{\sigma(1)}, ..., \mathbf{D}_{\sigma(n)})$$

for all $n \geq 1$ and all permutations σ of $(1, ..., n)$. In other words, the order of appearance of the observation does not matter in terms of their joint distribution. Let θ_i the probability of a realization of $Y_i = 1$ of *Stability* with odds of stability $\frac{\theta_i}{1-\theta_i}$. Thus the dichotomous variable Y can be described by a Bernoulli distribution with probability of success θ_i:

$$Y_i|\theta_i \overset{ind}{\sim} Bern(\theta_i), \quad i = 1, .., n$$

Consider a logistic regression model[1], which writes that the logit of the probability θ_i, or the log of the its odd is a linear function of some predictor variables $\mathbf{x}_i$:

$$\text{Logit}(\theta_i) = \log\left(\frac{\theta_i}{1-\theta_i}\right) = \beta_0 + \sum_j^{2^t} \beta_j \mathbf{x}_{j,i} \tag{13}$$

where the j predictors are T, that is the time of the realization of Y and W, that is the corresponding value. Since W has 2^t levels, we regress $2^t - 1$ dummy variable and keep $W = 0$ as the reference category:

$$\log\left(\frac{\theta_i}{1-\theta_i}\right) = \beta_0 + \beta_1 \times T + \sum_j^{2^t-1} \beta_j \mathbf{w}_{j,i} \tag{14}$$

By construction, the intercept of this model β_0 can be interpreted as the baseline risk for *Stability*. A high β_0 suggests that the underlying graphical model is not changing much over time. β_t captures the effect of the drift over time. It can be shown that *Stability* is weakly decreasing over time and, thus β_1 define the speed of convergence towards the absence of stability. Finally, since the variable Y takes value 1 for $W_{i,j} = (0, T^2-1)$, the coefficient β_{T^2-2}, that is the coefficient for $W_{i,j} = T^2-1$ with reference $W_{i,j} = 0$ captures which component of *Stability* originates in the persistence of existing connections, rather than on the persistence of absence of connections.

The computation is straightforward: by rearranging the logistic regression Equation 13, it is possible to express the regression as a nonlinear equation for the probability

[1]The logistic regression seem the most natural way to describe this phenomenon. However, according to the type of expected drift, we could employ other function, without loss of generalization.

of success θ_i :

$$\begin{aligned}
\log\left(\frac{\theta_i}{1-\theta_i}\right) &= \beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i} \\
\frac{\theta_i}{1-\theta_i} &= \exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\} \\
\theta_i &= \frac{\exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}}{1 + \exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}}
\end{aligned} \tag{15}$$

From the Equation 15 we can define the likelihood for the sequence of Y_i over data set of n subjects is then

$$\begin{aligned}
p(\mathbf{D}|\beta_0, \boldsymbol{\beta}_p) = \prod_{i=1}^{n} \Bigg[&\left(\frac{\exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}}{1 + \exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}}\right)^{y_i} \\
&\left(1 - \frac{\exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}}{1 + \exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}}\right)^{(1-y_i)} \Bigg]
\end{aligned} \tag{16}$$

where $\mathbf{D}$ is the dataset composed by T_i and the corresponding dummy variables generated by the level of W_i. The set of unknown parameters consists of $\beta_0, \beta_T, ..., \beta_{T^2-2}$. In general, any prior distribution can be used, depending on the available prior information. The literature suggests the use of informative prior distributions if something is known about the likely values of the unknown parameters, otherwise, the use of non-informative prior if either little is known about the coefficient values or if one wishes to see what the data themselves provide as inferences. In this case, we will use the most common priors for logistic regression parameters:

$$\beta_j \sim N(\mu_j, \sigma_j^2) \tag{17}$$

The most common choice for μ is zero with σ large enough to considered as non-informative in the range from $\sigma = 10$ to $\sigma = 100$. The posterior distribution of $\boldsymbol{\beta}_j$ is extrapolated by combining likelihood Eq. 16, with the prior in Eq. 17:

$$p(\beta_0, \boldsymbol{\beta}_p|\mathbf{D}, \sigma_j, \mu_j) = \prod_{i=1}^{n} \left[\left(\frac{\exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}}{1 + \exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}} \right)^{y_i} \right.$$
$$\left. \left(1 - \frac{\exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}}{1 + \exp\left\{\beta_0 + \sum_j^p \beta_j \mathbf{x}_{j,i}\right\}} \right)^{(1-y_i)} \right] \tag{18}$$
$$\times \prod_{j=0}^{p} \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left\{ -\frac{1}{2} \left(\frac{\beta_j - \mu_j}{\sigma_j} \right)^2 \right\}$$

Now, we are not that much interested in the regression parameters $\boldsymbol{\beta}_j$, we want to find the posterior probability distribution of the *stability*. Furthermore, this model gives us the opportunity to compute the prediction of the *stability* over a specific time t. If $\widetilde{y}_i$ represents the number of similarity connection between n nodes at time t, then one would be interested in the posterior predictive distribution of the fraction $\widetilde{y}_i/n$ One represents this predictive density of $\widetilde{y}_i$ as:

$$f(\widetilde{Y}_i|y) = \int p(\beta_0, \boldsymbol{\beta}_p|\mathbf{D}, \sigma_j, \mu_j) p(\widetilde{y}_i, \mathbf{X}|\beta_0, \boldsymbol{\beta}_p) d\boldsymbol{\beta} \tag{19}$$

where $p(\beta_0, \boldsymbol{\beta}_p|\mathbf{D}, \sigma_j, \mu_j)$ is the posterior density of $\boldsymbol{\beta}$ and $p(\widetilde{y}_i, \mathbf{X}|\beta_0, \boldsymbol{\beta}_p)$ is the Binomial sampling density of $\widetilde{y}_i$ conditional of regression vector $\boldsymbol{\beta} = (\beta_0, \boldsymbol{\beta}_p)$. Figure 2 represents the Bayesian graphical model of the stability, in particular, we can see all process that describes from the adjacent matrix to the coefficients of the logistic, that say us how changes the relationship between the variables over the time. Where we have an adjacent matrix (AM) for each time t, for each pair sequential of the AM we have a transition matrix TM. From the TM we can build the dataset to compute the stability with n observation, where $n = \mathcal{V} \times (T-1)$, and three variables: **W,T,Y**.

4 Empirical experiment

As a test bed for this theoretical approach, we apply the stability index to the *ELEC2* dataset [Harries, 1999], a benchmark for drift evaluation [Baena-Garcıa et al., 2006, Kuncheva and Plumpton, 2008, among the many]. It holds information on the Australian New South Wales (NSW) Electricity Market, containing 27552 records dated from May 1996 to December 1998, each referring to a period of 30 minutes. These records have 5 fields: a binary class label Y and four covariates X_1, X_2, X_3 and X_4 capturing different aspects of electricity demand and supply. In order to compute the empirical evolution of the drift over time, we group observations in one week period. Thus, for each week we have a panel dataset of 5 variables and 336 observation. Thus, we have a tensor X with dimension $(N \times p \times T)$ with $N = 336$ records for a week, $p = 5$ the variables as described above and $T = 82$ temporal periods.

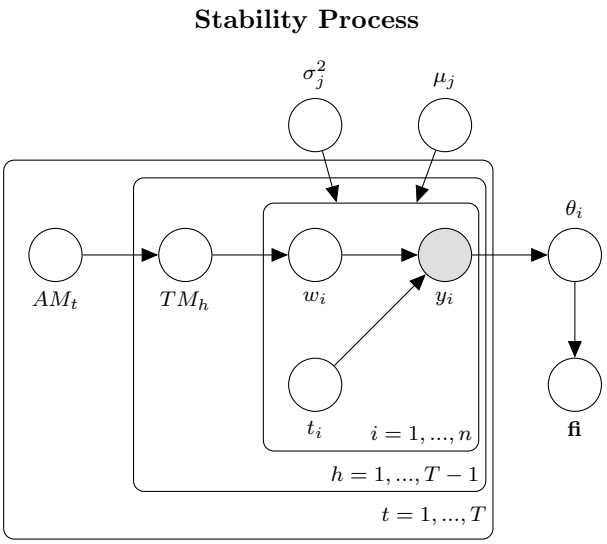

Figure 2: Bayesian Graphical Model of the *stability*

First, we realize a Graphical Models for each period t as the start point of our strategy to compute the drift. Figure 3 portraits the graphs for some selected periods and shows that the structure of the graph changes overtime. We thus expect a presence of the drift.

Figure 4 depicts the evaluation of the drift overtime. The red dots are the percentage of stable relations among variables, that is the the sum of variable $Y_{i,t}$ in Equation 12, while the blue line is the estimation of the Equation 18 with its related confidence interval as the gray contour.The figure highlights 6 periods of drift. The different *Stability* values are reported in the table 4. In the table 3 are reported the magnitude of the coefficients for the baseline β_0 or intercept, β_{2^T-1} for the $W = 2^T - 1$ with reference level $W = 0$ and for the time β_{time}.

Percent of Stability	**Evolution of the Drift**					
	$ty = 2$	$ty = 8$	$ty = 12$	$ty = 14$	$ty = 19$	$ty = 41$
$\frac{\sum_{i=1}^{N} Y_{i,t}}{N}$	1.0	0.8	0.6	0.5	0.2	0.1

Table 2: Approximation of the drift for selected period

5 Conclusion

This paper presented an algorithm to estimate the magnitude of a model drift in a context of machine learning. While past solutions relies on how the classification errors of a specific target variable changes over time, the present method tries to describe the

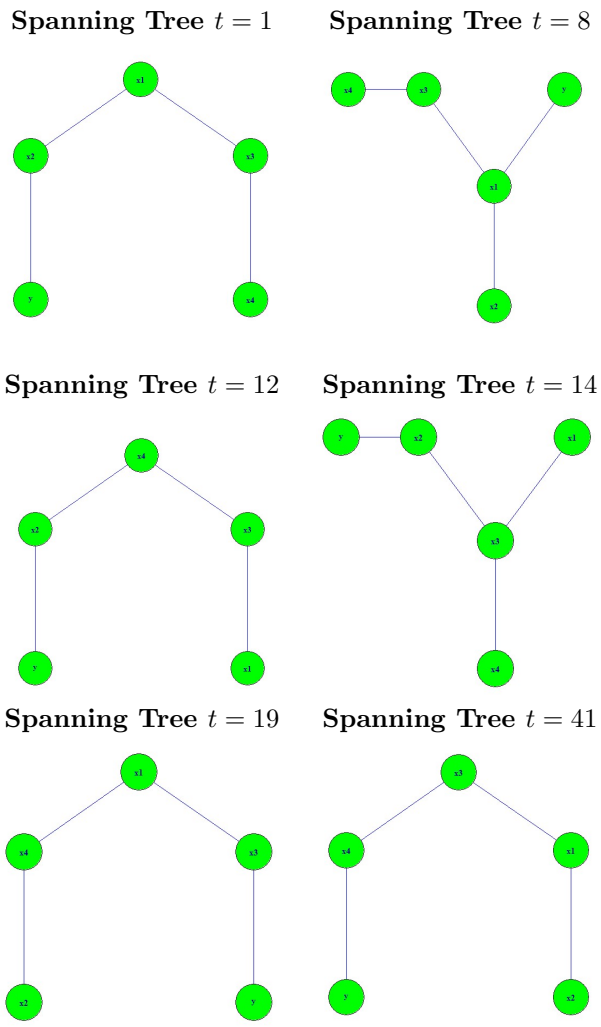

Figure 3: Graph over the time

Regression Summary

Coefficients	**Estimation**
β_0	**7.66**
β_{2^T-1}	**19.75**
β_{Time}	**-0.30**

Table 3: Coefficients of logistic regression

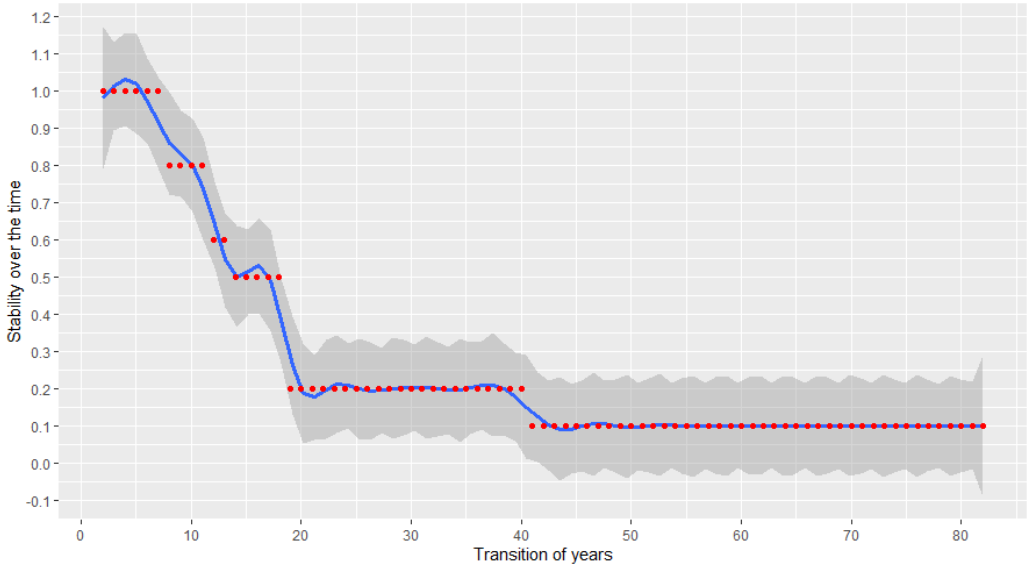

Figure 4: Evolution of Stability

underlying hidden context with the use of graphical models and to estimate how the observable context changes over time. Specifically, we provide not only an assessment of the drift, which is independent from the model in use, but also an estimation of the confidence interval of this prediction. These two characteristics combined together allow to signal when a data driven process shows an excessive risk due to the drift and needs to be retrained or re-calibrated. Possible applications are countless such as predicting defaults, online recommendations systems, or spam filtering. More specific, any prediction which involves human behaviour is prone to constant changes in the data generating process, while biological and physical phenomena tend to be more stable over time. Further lines of research in this area include a fine tuning for estimating different type of drift, allowing for temporary drift, and testing the index on a wider array of applications.

References

João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, 46(4):1–37, 2014.

Gerhard Widmer and Miroslav Kubat. Learning in the presence of concept drift and hidden contexts. *Machine learning*, 23(1):69–101, 1996.

Geoffrey I Webb, Roy Hyde, Hong Cao, Hai Long Nguyen, and Francois Petitjean. Characterizing concept drift. *Data Mining and Knowledge Discovery*, 30(4):964–994, 2016.

Ralf Klinkenberg and Thorsten Joachims. Detecting concept drift with support vector machines. In *ICML*, pages 487–494, 2000.

Ryan Elwell and Robi Polikar. Incremental learning of concept drift in nonstationary environments. *IEEE Transactions on Neural Networks*, 22(10):1517–1531, 2011.

Ralf Klinkenberg and Ingrid Renz. Adaptive information filtering: Learning in the presence of concept drifts. *Learning for text categorization*, pages 33–40, 1998.

Charles Taylor, Gholamreza Nakhaeizadeh, and Carsten Lanquillon. Structural change and classification. In *Workshop Notes on Dynamically Changing Domains: Theory Revision and Context Dependence Issues, 9th European Conf. on Machine Learning (ECML'97), Prague, Czech Republic*, pages 67–78. April, 1997.

Ralf Klinkenberg. Learning drifting concepts: Example selection vs. example weighting. *Intelligent data analysis*, 8(3):281–300, 2004.

Mashail Althabiti and Manal Abdullah. Classification of concept drift in evolving data stream. *Emerging Extended Reality Technologies for Industry 4.0: Early Experiences with Conception, Design, Implementation, Evaluation and Deployment*, page 189, 2020.

RP Jagadeesh Chandra Bose, Wil MP van der Aalst, Indrė Žliobaitė, and Mykola Pechenizkiy. Handling concept drift in process mining. In *International Conference on Advanced Information Systems Engineering*, pages 391–405. Springer, 2011.

SL Lauritzen. Graphical models, ser. *Oxford Statistical Science Series. Oxford University Press*, 1996.

C Chow and Cong Liu. Approximating discrete probability distributions with dependence trees. *IEEE transactions on Information Theory*, 14(3):462–467, 1968.

Joseph B Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical society*, 7(1):48–50, 1956.

David Edwards, Gabriel CG De Abreu, and Rodrigo Labouriau. Selecting high-dimensional mixed graphical models using minimal aic or bic forests. *BMC bioinformatics*, 11(1):18, 2010.

Erik B Sudderth, Martin J Wainwright, and Alan S Willsky. Embedded trees: Estimation of gaussian processes on graphs with cycles. *IEEE Transactions on Signal Processing*, 52(11):3136–3150, 2004.

Gabriel CG de Abreu, Rodrigo Labouriau, and David Edwards. High-dimensional graphical model search with graphd r package. *arXiv preprint arXiv:0909.1234*, 2009.

Antonino Abbruzzo and Angelo M Mineo. Inferring networks from high-dimensional data with mixed variables. In *Advances in Complex Data Modeling and Computational Methods in Statistics*, pages 1–15. Springer, 2015.

Alejandro Jara and Timothy Hanson. A class of mixtures of dependent tail-free processes. *Biometrika*, 98(3):553–566, 2011.

Michael Harries. *Splice-2 Comparative Evaluation: Electricity Pricing*. PANDORA electronic collection. University of New South Wales, School of Computer Science and Engineering, 1999. URL https://books.google.it/books?id=1Zr1vQAACAAJ.

Manuel Baena-Garcıa, José del Campo-Ávila, Raúl Fidalgo, Albert Bifet, R Gavalda, and R Morales-Bueno. Early drift detection method. In *Fourth international workshop on knowledge discovery from data streams*, volume 6, pages 77–86, 2006.

Ludmila I Kuncheva and Catrin O Plumpton. Adaptive learning rate for online linear discriminant classifiers. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 510–519. Springer, 2008.