Early Classification of Time Series: Cost-based multiclass Algorithms

1st Paul-Emile Zafar
Orange Labs
Châtillon, France
zafarpe@gmail.com

2nd Youssef Achenchabe
Orange Labs
Université Paris-Saclay
Paris, France
youssef.achenchabe@universite-paris-saclay.fr

3rd Alexis Bondu
Orange Labs
Châtillon, France
alexis.bondu@orange.com

4th Antoine Cornuéjols
Université Paris-Saclay
Paris, France
antoine.cornuejols@agroparistech.fr

5th Vincent Lemaire
Orange Labs
Lannion, France
vincent.lemaire@orange.com

***Abstract*—Early classification of time series assigns each time series to one of a set of pre-defined classes using as few measurements as possible while preserving a high accuracy. This implies solving online the trade-off between the earliness and the prediction accuracy. This has been formalized in previous work where a cost-based framework taking into account both the cost of misclassification and the cost of delaying the decision has been proposed. The best resulting method, called ECONOMY-γ, is unfortunately so far limited to binary classification problems. This paper presents a set of six new methods that extend the ECONOMY-γ method in order to solve multiclass classification problems. Extensive experiments on 33 datasets allowed us to compare the performance of the six proposed approaches to the state-of-the-art one. The results show that: (i) all proposed methods perform significantly better than the state of the art one; (ii) the best way to extend ECONOMY-γ to multiclass problems is to use a confidence score, either the Gini index or the maximum probability.**

***Index Terms*—time series, online decision making**

I. INTRODUCTION

The problem of early classification of time series is important in many application areas where the data about events is available over time and one must decide its class as early as possible but still with high accuracy if possible. This could for instance apply in a hospital's emergency room where a patient is monitored and it must be decided what must be done. Each passing minute without taking decision might be dangerous for the patient, but it also brings new information that can help the diagnosis, hence a difficult trade-off that must be solved in real-time.

While early classification of time series has been introduced to the machine learning field in 2004 [1], it is only recently that new approaches have been proposed that explicitly take into account both the misclassification cost and the delay cost in a single criterion in order to decide when is the apparent best time to make a prediction about the incoming time series [2]. These approaches, furthermore, are able to look ahead of time to estimate at one point in the future should be the optimal time to make a decision. This might be very useful in order to make preparations for the decisions that may have to be taken.

However, within the techniques presented to implement this approach, the most successful ones are limited to binary classification tasks while the ones that could tackle multiclassification have lower performances. The purpose of this paper is therefore to propose new techniques based on the well-grounded optimization criterion for early classification of time series presented in [2], [3], but which allow for multiclassification and have better performances than the existing ones.

Formally, we suppose that we are given a set $\mathcal{S}$ of "complete" time series together with their labels $\mathcal{S} = \{(\mathbf{x}_T^i, y_i)\}_{1 \leq i \leq m}$ where T is the length of the time series, $\mathbf{x}_T^i = \langle x_1^i, \ldots, x_t^i, \ldots, x_T^i \rangle$ is the ith time series with x_t^i the multi-valued measurement made at time t and $y_i \in \mathcal{Y}$ the label of $\mathbf{x}_T^i$. The task is to make a prediction about the class of an incoming time series as early as possible because a cost is incurred at the time of the decision, where the cost function increases with time.

The paper is organized as follows. Section II outlines some important contributions in the field of early classification of time series that pave the way for our

work. In section III, we present the approach proposed in [2], [3] that we extend to the multi classification setting by proposing two families of algorithms in section IV. In Section V, their characteristics and performances are tested using a wide set of 33 datasets. Finally, Section VI presents lessons from the study and perspectives.

II. Related work

Supervised classification of time series is a very active field of research, as it is a useful and challenging learning task. Recent advances in this field have shown: i) that ensemble methods are the most efficient [4]; ii) that the choice of time series representation has an important impact on the quality of the classifiers [5]; iii) that the extraction of informative features is a key point to obtain good performance [6].

In the classification of time series, the successive *measurements* are not supposed to be i.i.d. In the absence of an assumption about the generative process of the times series, it is supposed that there exists a labeled training set $\mathcal{S}$, as underlined in Section I, which allows the discovery of the underlying regularities. In the test phase, the scenario goes as follows. At each time step $t < T$, a new measurement x_t is collected and a decision has to be made as whether to make a prediction now or to defer the decision to some future time step. When $t = T$, a decision is forced. To the best of our knowledge, [1] was the earliest paper explicitly mentioning "classification when only part of the series are presented to the classifier".

For many researchers, the question to solve is *can we classify an incomplete times series while ensuring some minimum probability threshold that the same decision would be made on the complete input?*

One approach is to assume that the *time series* are generated i.i.d. according to some probability distribution, and to estimate the parameters of the class distributions from the training set. Once $p(\mathbf{x}_T|\mathbf{x}_t)$ the conditional probability of the entire time series $\mathbf{x}_T$ given an incomplete realization $\mathbf{x}_t$ is estimated, it becomes possible to derive guarantees of the form:

$$p\big(h_T(\mathbf{X}_T) = y|\mathbf{x}_t\big) \;=\; \int_{\mathbf{x}_T \text{ s.t. } h_T(\mathbf{x}_T)=y} p(\mathbf{x}_T|\mathbf{x}_t)\, d\mathbf{x}_T \;\geq\; \epsilon$$

where $\mathbf{X}_T$ is a random variable associated with the complete times series, ϵ is a confidence threshold, and $h_T(\cdot)$ is a classifier learnt over the training set $\mathcal{S}$ of complete times series. At each time step t, $p(h_T(\mathbf{X}_T) = y|\mathbf{x}_t)$ is evaluated and the prediction is triggered if this term becomes greater than some predefined threshold. [7], [8] present this method and propose ways to make the required estimations, in particular the mean and the covariance of the complete training data, when the time series are generated by Gaussian processes. It so far applies only with linear and quadratic classifiers.

In [9], a system, called TEASER, is presented that combines three components: (i) a set of slave classifiers that estimate the class probabilities of the incoming series, (ii) a master classifier which assesses the confidence that one can have in the class that has the higher probability according to the slave classifier at the current time step t, and finally (iii), the TEASER system which outputs a class if the master classifier has vetted this class for at least v time steps consecutively. The cost of delaying decision is not explicitly taken into account in this work. The authors heuristically propose to optimize the harmonic mean between accuracy and earliness which indirectly corresponds to a particular tradeoff and a particular cost of delaying the decision.

In [10], by contrast, the authors do not make assumptions about the form of the underlying distributions on the time series. They propose to use a 1NN classifier that chooses the nearest training time series $\mathbf{x}_t^i \in \mathcal{S}$ to the incoming one $\mathbf{x}_t = \langle x_1, \ldots, x_t \rangle$ to make its prediction. To determine for which time step t it is appropriate to make the prediction, the method is based on the idea of the *minimum prediction length* (MPL) of a time series. For a time series $\mathbf{x}_t^i$, one finds the set of every training time series $\mathbf{x}_t^j$ that have $\mathbf{x}_t^i$ as their one nearest neighbor (1-NN). The MPL of $\mathbf{x}_t^i$ is then defined as the smallest time index for which this set does not change when the rest of the time series $\mathbf{x}_t^i$ is revealed. In the test phase, at time step t, it is deemed that $\mathbf{x}_t$ can be safely labeled if its 1-NN $= \mathbf{x}_t^i$ for which the MPL is t. The idea is that from this point on, the prediction about $\mathbf{x}_t$ should not change. The authors found experimentally that this procedure, called ECTS (*Early Classification of Times Series*), leads to too conservative estimations of the earliest safe time step for prediction. They therefore proposed heuristic means to lower the estimated values. The stability criterion acts in a way as a proxy for a measure of confidence in the prediction. Similarly, [11] proposes a method where the evolution of the accuracy of a set of probabilistic classifiers is monitored over time, which allows the identification of timestamps from whence it seems safe to make predictions.

Another line of research is concerned with finding good descriptors of the time series, especially on their starting subsequences, so that early predictions can be reliable because they would be based on relevant similarities on the time series. For instance, in the works of [12]–

[14], the principle is to look for shapelets, subsequences of time series which can be used to distinguish time series of one class from another, so that it is possible to perform classification of time series as soon as possible.

The methods described above make use of the complete knowledge available in the training set. There are methods however that do not use this information during training. For instance, in [8], [15], [16], a model $h_t(\cdot)$ is learnt for each early timestamp and various stopping rules are defined in order to decide whether, at time t, a prediction should be made or not. The price to pay for not taking into account information about the likely future of the time series is that decisions are made in a myopic fashion which may prevent one from seeing that a better trade-off between earliness and accuracy is achievable in the future.

This is also the case for the work presented in [17]. In the paper, the authors recognize the conflict between earliness and accuracy. But instead of setting a tradeoff in a single objective optimization criterion [18], they propose to keep it as a multi-objective criterion and to explore the Pareto front of the multiple dominating tradeoffs. They then suggest a family of triggering functions involving hyper parameters to be optimized for each tradeoff. This contrasts with approaches whereby the decision is made solely on the basis of a given confidence threshold which should be attained. However, the optimization criterion put forward is heuristic, supposes that the cost of delaying a decision is linear in time, and involves a complex setup. Most importantly, again, it is a myopic procedure which does not consider the foreseeable future. For all these apparent shortcomings, this method has been found to be quite effective, beating most competing methods in extensive experiments.

In [2], for the first time, the problem of early classification of time series is cast as the optimization of a loss function which combines the expected cost of misclassification at the time of decision plus the cost of having delayed the decision thus far. Besides the fact that this optimization criterion is well-founded, it permits also to apply the LUPI framework[1] because the expected costs for an incoming subsequence $\mathbf{x}_t$ can be estimated for future time steps and thus a non-myopic decision procedure can be used. These expectations can indeed be learned from the training set of m complete time series $\mathcal{S} = \{(\mathbf{x}_T^i, y_i)\}_{1 \leq i \leq m}$.

[1]Early classification of time series can be seen as an instance of the LUPI (Learning Under Privileged Information) framework [19]: during the learning phase, the learner has access to the full knowledge about the training time series $\mathcal{S} = \{(\mathbf{x}_T^i, y_i)\}_{1 \leq i \leq m}$, while at testing time, only a subsequence $\mathbf{x}_t$ ($t < T$) is known.

The idea presented in [2] has been extended in [3] both formally and in the presentation of new algorithms and of a wider set of experiments where it was shown that the performances obtained topped the performances obtained so far with previous methods. We therefore base our investigation on this work.

In the following section, we present the approach proposed in [2], [3] that we extend to the multi classification setting in section IV.

III. A COST-BASED NON-MYOPIC FRAMEWORK

This section provides an overview of the cost-based non-myopic framework presented in [3] to tackle the Early Classification problem.

In [3] the learning phase is carried out offline (I), and the deployment phase, is performed online (II):

I- For each time step, $t \in \{1, \ldots, T\}$, a classifier[2] h_t can be learned from $\mathcal{S}$, such that $h_t : \mathcal{X}^t \rightarrow \mathcal{Y}$. In addition, some knowledge is extracted from $\mathcal{S}$ to estimate the probable future of an incoming time series $\mathbf{x}_t$, namely, the probabilistic terms in equation 1.

II- Using these classifiers and the extracted knowledge, it is possible to estimate the optimal instant for making a decision, i.e. triggering a prediction about its class. More precisely, given the *misclassification cost* function $\mathrm{C}_m(\hat{y}|y) : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ that expresses the cost of predicting $\hat{y}$ when the true class is y and the *delay cost* function $C_d(t) : \mathbb{R} \rightarrow \mathbb{R}$ which is assumed to be an increasing function of time, the expectancy of the cost of taking a decision at time t given the incoming time series $\mathbf{x}_t$ is:

$$\begin{aligned} f(\mathbf{x}_t) &= \mathbb{E}^{\,\mathrm{t}}_{(\hat{\mathrm{y}},\mathrm{y})\in\mathcal{Y}^2}\left[\mathrm{C}_m(\hat{y}|y)|\mathbf{x}_t\right] + \mathrm{C}_d(t) \\ &= \sum_{y\in\mathcal{Y}} P_t(y|\mathbf{x}_t) \sum_{\hat{y}\in\mathcal{Y}} P_t(\hat{y}|y,\mathbf{x}_t)\,\mathrm{C}_m(\hat{y}|y) \;+\; \mathrm{C}_d(t) \end{aligned} \tag{1}$$

where $\mathbb{E}^{\,\mathrm{t}}_{(\hat{\mathrm{y}},\mathrm{y})\in\mathcal{Y}^2}[.]$ is the expectancy at time t, over the variables y and $\hat{y}$. $P_t(y|\mathbf{x}_t)$ is the probability of the class y given a time series that starts as $\mathbf{x}_t$, and $P_t(\hat{y}|y,\mathbf{x}_t)$ is the probability that the classifier h_t makes the prediction $\hat{y}$ given $\mathbf{x}_t$ while y would be its true label. In this non-myopic setting, the idea is that the decision of making a prediction is made at the current time t only insofar as it is not expected that a lower cost could be achieved at a later time. This could happen if the expected misclassification cost would drop sufficiently to offset the increase of $\mathrm{C}_d(t)$.

[2]Note that these classifiers are learned off-line and that the concepts to be learned are considered stationary.

For any time $t+\tau$ in the future ($1 \leq \tau \leq T-t$), the expected cost of making a prediction can be estimated as:

$$\begin{aligned} f_\tau(\mathbf{x}_t) &= \mathbb{E}^{\,t+\tau}_{(\hat{y},y)\in\mathcal{Y}^2}\left[\mathrm{C}_m(\hat{y}|y)|\mathbf{x}_t\right] + \mathrm{C}_d(t+\tau) \\ &= \sum_{y\in\mathcal{Y}} P_{t+\tau}(y|\mathbf{x}_t) \sum_{\hat{y}\in\mathcal{Y}} P_{t+\tau}(\hat{y}|y,\mathbf{x}_t)\,\mathrm{C}_m(\hat{y}|y) \\ &+ \mathrm{C}_d(t+\tau) \end{aligned} \tag{2}$$

where $P_{t+\tau}(\hat{y}|y,\mathbf{x}_t)$ is one term of the confusion matrix expected at time $t+\tau$ given that the time series starts as $\mathbf{x}_t$. In [3], the authors propose different ways for estimating this term as will be seen below. When $\tau = 0$, we have $f_0(\mathbf{x}_t) = \mathbb{E}^{\,t}_{y\in\mathcal{Y}}\left[\mathrm{C}_m(\hat{y}_t|y)|\mathbf{x}_t\right] + \mathrm{C}_d(t)$ since we have access to predictions at current time.

Then, the optimal decision time, at time t, is expected to be:

$$\tau^* = \underset{\tau\in\{0,\ldots,T-t\}}{\mathrm{ArgMin}}\; f_\tau(\mathbf{x}_t) \tag{3}$$

The idea is to estimate the cost of a decision at all future time steps, up until $t = T$, based on the current knowledge about the incoming time series, and to postpone the decision to the time step that appears to be the best.

If $\tau^* = 0$, then it seems that there is no better time for making a prediction than now. Therefore, the prediction $h_t(\mathbf{x}_t)$ is returned and the classification process is terminated. Otherwise the decision is postponed to the next time step, and Eq. 3 is computed again, this time with $\mathbf{x}_{t+1}$. The process goes on until a decision is made or $t = T$ at which point a prediction is forced.

Equation 2 has been proposed and has given way to several different algorithmic versions generically called ECONOMY as described in [3]. They differ in the way they make groups $\mathfrak{g}_k \in \mathcal{G}$ of time series, in order to estimate the future expected cost $f_\tau(\mathbf{x}_t)$:

$$\begin{aligned} f_\tau(\mathbf{x}_t) &= \mathbb{E}^{\,t+\tau}_{(\hat{y},y)\in\mathcal{Y}^2}\left[\mathrm{C}_m(\hat{y}|y)\right] \\ &= \sum_{\mathfrak{g}_k\in\mathcal{G}} P_t(\mathfrak{g}_k|\mathbf{x}_t) \sum_{y\in\mathcal{Y}} P_t(y|\mathfrak{g}_k) \sum_{\hat{y}\in\mathcal{Y}} P_{t+\tau}(\hat{y}|y,\mathfrak{g}_k)\,\mathrm{C}_m(\hat{y}|y) \\ &+ \mathrm{C}_d(t+\tau) \end{aligned} \tag{4}$$

Of all these methods, ECONOMY-γ is the one that stands out, both thanks to its refined way of predicting the likely future of an incoming time series and because of its significantly better performances demonstrated in extensive experiments over the other ECONOMY versions as well as with the method of [18]. However, the ECONOMY-γ approach is limited to binary classification problems, therefore this paper aims to extend this approach to multiclass classification problems.

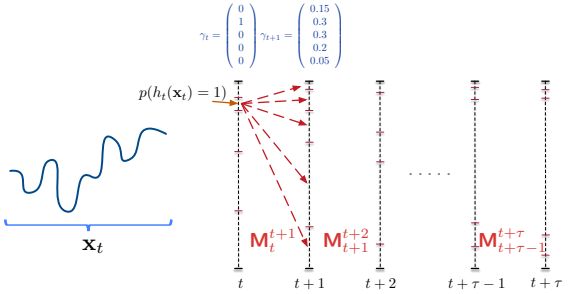


Fig. 1: Figure from [3]. ECONOMY-γ, computing the probability distribution $p(\gamma_{t+\tau}|\gamma_t)$. Here $h_t(\mathbf{x}_t)$ falls in the second confidence level interval. Given the supposed learned transition matrix M_t^{t+1}, the next vector of confidence levels will be $(0.15, 0.3, 0.3, 0.2, 0.05)^\top$.

More precisely, in ECONOMY-γ, the groups $\mathcal{G}$ are obtained by pooling the time series by confidence levels relative to the class 1 (opposite to class 0) $p(y=1|\mathbf{x}_t)$[3] of h_t. For each time series $\mathbf{x}_t$ observed up to time t, the confidence level $p(h_t(\mathbf{x}_t) = 1)$ can take a value in $[0,1]$. Examining the confidence levels thus obtained for all time series in the validation set $\mathcal{S}'^t$ truncated to the first t observations, we can discretize the interval $[0,1]$ into K equal frequency intervals, denoted $\{I_t^1, \ldots, I_t^K\}$. For instance, if $K = 5$, and $|\mathcal{S}'^t| = 1000$, the intervals $I_t^1 = [0, 0.30[$, $I_t^2 = [0.30, 0.45[$, $I_t^3 = [0.45, 0.58[$, $I_t^4 = [0.58, 0.83[$, $I_t^5 = [0.83, 1]$ could each correspond to 200 training time series. The discretization of confidence levels into equal frequency intervals corrects any bias in the calibration of h_t, in a similar way to isotonic calibration [20].

For a new incoming time series known up to time t: $\mathbf{x}_t$, and knowing the corresponding sequence of confidence intervals $\langle I_1, \ldots, I_u, \ldots, I_t\rangle$ where we have $I_u = k \in K | p(h_u(\mathbf{x}_u) = 1) \in I_u^k$, it is possible to evaluate $\gamma_{t+\tau}^i = p(h_{t+\tau}(\mathbf{x}_{t+\tau}) = 1) \in I_{t+\tau}^i$, and then:

$$\begin{aligned} f_\tau(\mathbf{x}_\tau) = \Bigg[&\sum_{j=1}^{K} \gamma_{t+\tau}^j \sum_{y\in\mathcal{Y}} P(y|I_{t+\tau}^j) \\ &\sum_{\hat{y}\in\mathcal{Y}} P_{t+\tau}(\hat{y}|y, I_{t+\tau}^j)\,\mathrm{C}_m(\hat{y}|y) \Bigg] + \mathrm{C}_d(t+\tau) \end{aligned} \tag{5}$$

In ECONOMY-γ, a Markov-chain model is used for estimating the terms $\gamma_{t+\tau}^j$ (see Figure 1). A fully detailed description of the ECONOMY-γ is provided in [3].

[3] This restricts these methods to binary classification problems.

Algorithm 1 summarizes the main steps of the learning phase.

Algorithm 1: ECONOMY-γ - learning stage

1 **forall** $t \in \{1, ..., T\}$ **do**
2 learn a classifier $h_t()$ from a set of truncated labelled time series $\mathcal{S}_t$;
3 discretize the confidence $p(h_t(\mathbf{x}_t) = 1)$ of the learned classifier into K equal frequency intervals $\{I_t^1, \ldots, I_t^K\}$. ;
4 learn a part of the Markov-chain model by estimating the transition matrix formed by the terms:
5 $p(h_t(\mathbf{x}_t) = 1) \in I_t^j \,|\, p(h_{t-1}(\mathbf{x}_{t-1}) = 1) \in I_{t-1}^i$
6 **end**

IV. The proposed approaches

While the original ECONOMY$-\gamma$ algorithm summarized an incoming time series $\mathbf{x}_t$ as a sequence of scalars $\langle p(h_1(\mathbf{x}_1) = 1, \ldots, p(h_t(\mathbf{x}_t) = 1\rangle$ in order to estimate the likely future of $\mathbf{x}_t$ as used in Equation 5, the extension to the multi class problem requires using a more complex summary. Now, instead of having one scalar by time stamp, we must do with a vector of $|\mathcal{Y}|$ real values: $< p(h_t(\mathbf{x}_t) = 1), ..., p(h_t(\mathbf{x}_t) = |\mathcal{Y}|) >$ *for each time stamp*, where $\mathcal{Y}$ is the set of classes.

In this section, we propose two leads to adapt ECONOMY-γ to multiclass problems: i) by using a confidence score that aggregates the $|\mathcal{Y}|$ probabilities estimated by the classifier into a scalar value; ii) by using a clustering algorithm in the vector space formed by the $|\mathcal{Y}|$ probabilities as in [21]–[23]. The complexity of the proposed approaches is studied in the supplementary material available in our Git repository.

A. Confidence scores aggregating probabilities

A first way to adapt ECONOMY-γ is to use a confidence score which aggregates the output vector of probabilities of the classifiers into a single scalar value: *Confidence*() : $\mathbb{R}^K \rightarrow \mathbb{R}$. The algorithm 1 is then slightly modified, replacing only line 3 by a new step, i.e. discretizing the output range of the *Confidence*() function into K equal frequency intervals. To do this, this function is applied to the time series of the validation set $\mathcal{S}'^t$. The obtained intervals are used as the states of the Markov Chain, and the rest of Algorithm 1 remains unchanged. This section presents the four approaches we propose, which use different confidence scores.

i) The **ECO-γ-entropy** approach uses the Shannon's entropy function to compute a confidence score with *Confidence*$(p_1, ..., p_{|\mathcal{Y}|}) = -\sum_{i=1}^{|\mathcal{Y}|} p_i \log(p_i)$, where $p_i = p(h_t(\mathbf{x}_t) = i)$. For each validation example, the entropy value is estimated based on the conditional probabilities of the classes. A high entropy indicates scattered probability values, which corresponds to an uncertain prediction. By contrast, low probabilities on all classes except one which is dominant lead to a low entropy value which corresponds to a highly confident prediction.

ii) The **ECO-γ-gini** approach exploits the gini impurity index in a very similar way as the entropy approach. Gini index is defined as *Confidence*$(p_1, ..., p_{|\mathcal{Y}|}) = 1 - \sum_{i=0}^{K-1} p_i^2$. This score behaves the same way as the entropy function for the different case scenarios, and it is computationally less expensive than the entropy score.

iii) The **ECO-γ-margins** approach uses the function *Confidence*$(p_1, ..., p_{|\mathcal{Y}|}) = p_i - p_j$ where p_i is the maximum conditional probability and p_j the second largest, with $p_i \geq p_j$. This margin score is commonly used as a confidence score in active learning strategies [24], [25]. A large margin corresponds to a high confidence level in the prediction. Conversely, if the two highest probabilities are close, the margin is low and this corresponds to an uncertain prediction.

iv) The **ECO-γ-max** approach focuses only on the maximum probability estimated by the classifier by taking *Confidence*$(p_1, ..., p_{|\mathcal{Y}|}) = \max_{1 \leq i \leq |\mathcal{Y}|} p_i$. Since $\sum_{i=1}^{|\mathcal{Y}|} p_i = 1$, a high maximum probability value implies low values for the other probabilities. Conversely, an important value of the maximum probability correspond to a confident prediction. This confidence score is less sophisticated than the margin one, since it cannot differentiate the cases where the two largest probabilities are close or not. It is thus interesting to see how it behaves nonetheless.

These four approaches differ only in the confidence scores they use. One objective of the experiments (Section V) is to compare them and to identify the confidence score that leads to the best performances.

B. Clustering

Another way to extend the ECONOMY-γ approach to multiclass problems is to use clustering methods on the outputs of the classifiers such as to form a set $\mathcal{G}$ of groups. At time t, for the classifier h_t, the output vector $< p(h_t(\mathbf{x}_t) = 1), \ldots, p(h_t(\mathbf{x}_t) = |\mathcal{Y}|) >$ belongs to the vector space $\mathbb{R}^{|\mathcal{Y}|}$, and each validation example of $\mathcal{S}'$

can be represented by a point in this vector space whose coordinates correspond to the conditional probabilities estimated by the classifier at time t. Moreover, these points belong to a sub-variety of $\mathbb{R}^{|\mathcal{Y}|}$ which is a hyperplane defined by the equation $\sum_{i=1}^{|\mathcal{Y}|} p(h_t(\mathbf{x}_t) = i) = 1$, i.e. the estimated probabilities over all class values must sum to 1. The use of a clustering algorithm can identify dense groups belonging to this hyperplane, and which differ by their confidence level on (all or part of) the classes. This section presents two approaches using a clustering algorithm.

i) The **ECO-γ-Kmeans** approach directly applies the K-means algorithm[4] on the validation examples which are defined as probability vectors in $\mathbb{R}^{|\mathcal{Y}|}$. Line 3 of Algorithm 1 is replaced by: cluster the time series of $\mathcal{S}'$ into K clusters (which correspond to the groups $\mathcal{G}$).

ii) The **ECO-γ-Kmeans-cal** approach is a variant that includes a pre-processing step consisting in calibrating the probabilities provided by the classifiers before using the K-means algorithm. In this approach, the probabilities $p(h_t(\mathbf{x}_t) = i)$ estimated by the classifier are replaced by their normalized rank, in a similar way to isotonic calibration [20]. Intuitively, this calibration seems to be required to prevent the K-means algorithm from misidentifying dense groups due to biases in the calibration of the classifiers h_t $(1 \leq t \leq T)$.

C. ECONOMY-K

The **ECO-K** approach was first introduced in [2] and it was shown to be significantly outperformed by ECONOMY-γ for binary classification problems [3]. In this article, ECO-K is considered as a competing approach since it is able to deal with multiclass problems.

Basically, ECO-K relies on clustering the full length time series $\mathbf{x}_T$ to form a single partition $\mathcal{G}$ using the K-means algorithm provided with the $\mathcal{L}_2$ norm. Then, given an incoming time series $\mathbf{x}_t$, the memberships $P(\mathfrak{g}_k|\mathbf{x}_t)$ are estimated using a logistic function of a distance between $\mathbf{x}_t$ and the centers of the clusters $\mathfrak{g}_k$. The continuation of the time series that belong to the groups is exploited to estimate the future expected costs. Namely, for each time step $t = 1, \ldots, T$, the confusion matrix of the classifier h_t is used to estimate $P_t(\hat{y}|y, \mathfrak{g}_k)$ within previously formed groups.

ECO-K is natively able to tackle multiclass problems. Indeed, the groups of $\mathcal{G}$ only rely on the time series themselves, and not on the classifiers output. In addition, the confusion matrices computed at each time step are not restricted to binary classification problems.

[4] We used the K-means algorithm provided with the $\mathcal{L}_2$ norm, and with 10 random initializations using Kmeans++ [26].

V. EXPERIMENTS

All the methods presented extend the ECONOMY-γ technique to multiclass problems, except ECO-K which is natively adapted to multiclass problems.

The *first question* that our experiments aim at answering is whether the proposed multiclass approaches bring significant performance gains compared to the state of the art approaches especially in case of multi classes multiclass classification problems[5].

The *second question* concerns the different ways to extend ECONOMY-γ to multiclass problems:

1) Is it a good idea to aggregate the probabilities estimated by the classifier into a scalar value? Or is it better to form the groups without aggregating these probabilities and using a clustering algorithm over the outputs of the classifier?
2) For the approaches which aggregate the estimated probabilities, which univariate confidence score leads to the best performances ?

Section V-D presents the obtained results.

A. Evaluation criterion

In order to compare the methods, we use the evaluation criterion introduced in [3] which expresses the average cost that is incurred using a particular early classification method. For a given dataset $\mathcal{S}$, it is defined as follows:

$$AvgCost(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{(\mathbf{x}_T, y) \in \mathcal{S}} \left(\mathrm{C}_m \left(h_{\widehat{t^*}}(\mathbf{x}_{\widehat{t^*}}) | y \right) + \mathrm{C}_d(\widehat{t^*}) \right) \quad (6)$$

where $\widehat{t^*}$ is the decision time chosen by the method to optimize the trade-off between earliness and accuracy. The criterion $AvgCost$ is used in our experiments both to optimize K on a validation set and to evaluate each early classification approach on each dataset (i.e. on test sets). Significant differences in performances are detected using statistical tests.

B. Datasets

In order to be able to make direct comparisons with [18] we use the same datasets as they did. This benchmark consists of 45 datasets of variable sizes that come from a variety of application areas. This collection of

[5] Note: This was already demonstrated in the case of binary classification [3] by comparing the ECONOMY family of algorithms with [18] which is currently the best performing myopic approach, as confirmed by a recent paper [27]

datasets has also been used in [9] and [17], making our experiences easily comparable to previous works. We keep the 33 datasets for which the number of classes is greater than two, which is appropriate for multiclass problems. Additional results using all the 45 datasets are provided in the supplementary material, leading substantially to the same conclusions. In order to reduce the computation time of the experiments and to compare datasets with time series of different lengths, we trained a classifier every 5% of the total length of the time series, instead of one classifier per time step, as done in [18]. Furthermore, for each dataset and for each possible length (i.e *5%, 10%, ...* of the total length), we extracted 60 features[6] from the corresponding truncated time series in order to train the associated classifiers. To do this, we used the Time Series Feature Extraction Library [28], which automatically extracts features on the statistical, temporal and spectral domains.

C. Experimental protocol

The same experimental[7] protocol as in [3] is used. The datasets were divided by uniformly selecting 70% of the examples for the training set and using the remaining 30% for the test set. Then, the training sets were divided into three disjoint subsets as follows:

- 40% for training the collection of classifiers $\{h_t\}_{t\in\{1,...,T\}}$ using the Python XGboost library[8] with the default values of the hyper-parameters;
- 40% for learning the meta-parameters of the proposed approaches, which consists of: (i) the discretization of the confidence score into K intervals for each classifier, and (ii) the transition matrices between a time step to the next one (i.e. every 5% of the time series length);
- 20% to optimize the number of groups K: all the approaches were trained by varying the number of groups between 1 to 10, and evaluated by $AvgCost(.)$ (see Equation 6). In order to manage datasets with a large number of classes, the values $K \in \{|\mathcal{Y}|, 2|\mathcal{Y}|\}$ are also evaluated. The value which minimizes the $AvgCost(.)$ criterion has been kept.

Costs setting: the misclassification cost was set in the same way for all datasets: $\mathrm{C}_m(\hat{y}|y) = 1$ if $\hat{y} \neq y$, and $= 0$ otherwise. The delay cost $\mathrm{C}_d(t)$ is provided by the domain experts in actual use cases. In the absence of this knowledge, we define it as a linear function of time, with coefficient, or slope, α:

$$\mathrm{C}_d(t) = \alpha \times \frac{t}{T} \tag{7}$$

The range of values used for α is $\{0.001, 0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$.

D. Results

First, our experiments compare the performance of the proposed approaches with the state of the art.

The **SR approach** is a very strong competitor. It was demonstrated in [18] to dominate all other algorithms in the literature over a benchmark with numerous datasets. In this algorithm, a trigger function is used to decide if the current prediction is reliable (output 1) or if it is better to wait for other measures (output 0):

$$\mathit{Trigger}\,(h_t(\mathbf{x}_t)) = \begin{cases} 0 & \text{if } \gamma_1 p_1 + \gamma_2 p_2 + \gamma_3 \frac{t}{T} \leq 0 \\ 1 & \text{otherwise} \end{cases} \tag{8}$$

where p_1 is the largest conditional probability estimated by the classifier h_t, p_2 is the difference between the two largest probabilities and $\frac{t}{T}$ represents the proportion of the incoming time series that is visible at time t.

The parameters $\gamma_1, \gamma_2, \gamma_3$ are real values in $[-1, 1]$ to be optimized. In our experiments, these parameters were tuned for each value of $\alpha \in [10^{-3}, 1]$ by minimizing the value of $AvgCost$ thanks to a grid-search on the values $[-1, -0.90, ..., 0, 0.1, ..., 0.90, 1]$.

After training, the $AvgCost$ criterion was evaluated on the 33 test sets for all values of α, both for the SR algorithm and for the proposed approaches. Then, Wilcoxon signed-rank tests were carried out to compare the SR approach with the six proposed variants of the ECONOMY approach, for each value of $\alpha \in [10^{-3}, 1]$. The results are presented in Figure 2, which shows that all the ECONOMY approaches perform significantly better than the SR approach, whatever the value of α.

A similar result was obtained in [3] in the case of binary classification problems. Figure 1 shows that the dominance of the Economy approaches is still verified for multiclass problems and confirms that the design choices of the proposed approaches are reasonable.

At this point, it remains to identify the best approach among those proposed, i.e., identify the best way to extend ECONOMY-γ to multiclass problems.

[6] More details are available in: https://cutt.ly/jvaKejI

[7] For full reproducibility of these experiments, a code is available at https://github.com/YoussefAch/Eco-gamma-multiclass.

[8] XGBoost is available in: https://xgboost.readthedocs.io

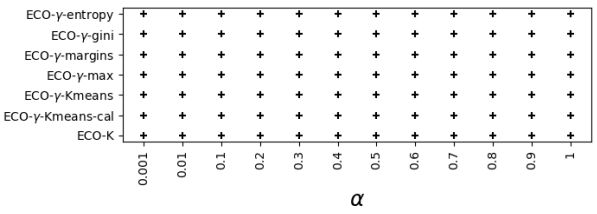


Fig. 2: SR *vs.* ECONOMY approaches: the evaluation is based on $AvgCost$ using the Wilcoxon signed-rank test, for different values of α. The symbol "+" indicates that all ECONOMY approaches win over the SR method. It is remarkable that the table only contains "'+".

For this purpose, we compare each ECONOMY approach to all others and for all values of α, using the Wilcoxon signed-rank test (as in Figure 2). This comparison is reported in Table I, where the second column counts the number of significant wins of each approach against all others; the third column counts the number of significant defeats;the fourth column reports the number of non-significant differences in performance; and, finally, the fifth column corresponds to the difference between the number of wins and the number of defeats.

TABLE I: ECONOMY approaches comparison using Wilcoxon signed-rank test: significant wins / defeats of each approach (against all the other) counted for all α, based onthe $AvgCost$ criterion.

Algorithm	wins	defeats	ties	balance
ECO-γ-max	16	0	56	+16
ECO-γ-gini	16	0	56	+16
ECO-γ-entropy	9	6	57	+3
ECO-K	8	4	60	4
ECO-γ-margins	1	9	62	-8
ECO-γ-Kmeans-cal	0	15	57	-15
ECO-γ-Kmeans	0	16	56	-16

Table I shows that the best performances are achieved by the ECO-γ-max and ECO-γ-gini approaches, when considering all the values of α. Actually, these two approaches have no significant defeats and have a large number of wins.

Surprisingly, the performance gap between the ECO-γ-gini and ECO-γ-entropy approaches is important. Even if these two confidence scores are mathematically very close, they do not produce exactly the same ranking of the examples and therefore the groups resulting from the discretization of these confidence scores are different.

At the other end of the spectrum, the ECO-γ-Kmeans and ECO-γ-Kmeans-cal approaches are the worst performing ones, which shows that the most promising lead to adapt ECONOMY-γ to the multiclass problems is to aggregate the classifier outputs into a confidence score.

Table I provides the ranking of the different approaches by their performance level, but this result is aggregated for all α values. The rest of the results presented in this section study the impact of the delay cost $\mathrm{C}_d(t)$ on the ranking of these approaches.

In our experiments, the proposed ECONOMY approaches are not distinguishable for the large majority of the cases where $\alpha > 0.4$ (see the supplementary material). Thus, we choose here to show detailed results for three representative cases, which correspond to $\alpha \in \{0.01, 0.1, 0.3\}$. The same results are available in the supplementary material for the other α values.

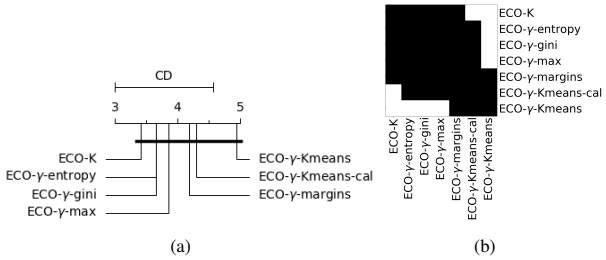
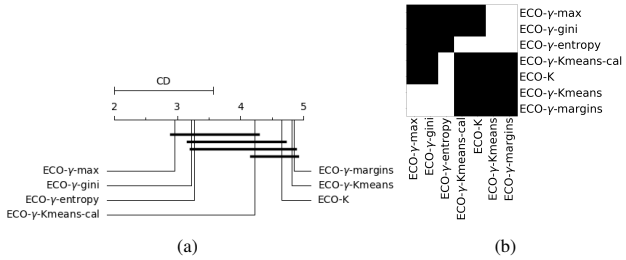


Fig. 3: Comparison of ECONOMY approaches for α = 0.01 using (a) Nemenyi and (b) Wilcoxon signed-rank tests.

Figure 3 (a) shows the Nemenyi test [29] applied for $\alpha = 0.01$. This test consists of two successive steps. First, the Friedman test is applied to the $AvgCost$ obtained by the competing approaches to determine whether their overall performance is similar. If not, the post-hoc test is applied to determine groups of approaches whose overall performance is significantly different from that of the other groups. In this case, the Nemenyi test is not able to show a significant difference, since all approaches belong to the same group.

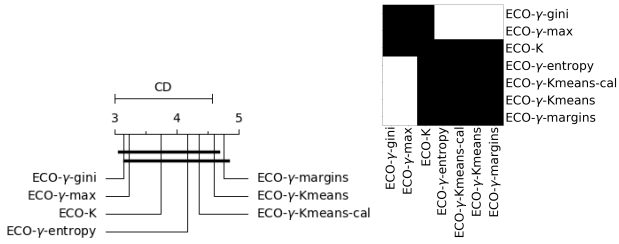
Figure 3 (b) shows pairwise comparison using the Wilcoxon signed-rank test between the approaches. The small black squares identify pairs of approaches that do not differ significantly in performance. It appears that: (i) ECO-γ-Kmeans is dominated by ECO-K, ECO-γ-entropy, ECO-γ-gini and ECO-γ-max; (ii) the ECO-γ-Kmeans-cal is dominated only by ECO-K. These results confirm the bad ranking of clustering based approaches observed in Table I.

Figure 4 (a) plots the Nemenyi test for $\alpha = 0.1$, and shows that: (i) ECO-γ-max is significantly better than ECO-K, ECO-γ-Kmeans and ECO-γ-margins; (ii) ECO-γ-gini and ECO-γ-entropy are significantly better than ECO-γ-Kmeans and ECO-γ-margins. The Wilcoxon tests in Figure 4 (b) confirm these results, except for ECO-γ-max that is not significantly better than ECO-K.



(a) (b)

Fig. 4: Comparison of ECONOMY approaches for $\alpha = 0.1$ using (a) Nemenyi and (b) Wilcoxon signed-rank tests



(a) (b)

Fig. 5: Comparison of ECONOMY approaches for $\alpha = 0.3$ using (a) Nemenyi and (b) Wilcoxon signed-rank tests

Figure 5 shows the same plots for a higher delay cost set by $\alpha = 0.3$. In this case, the approaches ECO-γ-gini and ECO-γ-max remain at the top of the ranking, and these two methods are significantly better than all the other except ECO-K, considering the Wilcoxon signed-rank tests.

Finally, these results based on statistical tests are in line with the results of Table I, and show that the two approaches ECO-$\boldsymbol{\gamma}$-max and ECO-γ-gini are consistently in the top group. Henceforth, the following results compare the competing approaches by varying α in a more fine-grained way, and by evaluating both their: *(i) earliness*; *and (ii) predictive performance*.

For a given dataset and a given value of $\alpha \in [10^{-3}, 1]$, the *earliness* is evaluated using the median of the trigger times $\hat{t^*}$ normalized by the length of the series, defined as: $\textit{Earliness} = med\{\hat{t^*}\}/T$. On the other hand, the *predictive performance* is evaluated using the Cohen's Kappa score [30] computed at the time of decision $\hat{t^*}$, since this criterion properly manages unbalanced datasets.

In Figure 6, the coordinates of each point are given by the average *Earliness* and the average Kappa score obtained over the 33 used datasets when the delay cost α is chosen in the range $[10^{-3}, 1]$. The Pareto curve is then drawn for each of the competing approaches. Two distinct groups of approaches can be identified in this figure: (i) the *top group* consists of the ECO-$\boldsymbol{\gamma}$-max, ECO-$\boldsymbol{\gamma}$-entropy, ECO-γ-gini approaches; (ii) the *second group* includes the other approaches, ECO-$\boldsymbol{\gamma}$-Kmeans, ECO-γ-Kmeans-cal, ECO-$\boldsymbol{\gamma}$-margins and ECO-K. The *top group* dominates the *second group* on both *Earliness* and Kappa criteria, i.e. two curves belonging to the different groups do not intersect. In contrast, the approaches within each group can not be clearly distinguished, since the curves in the same group cross each other. Furthermore, it can be noticed that for $\alpha \geq 0.4$ the curves of the two groups are very close to each other (see the lower left part of Figure 6), which is consistent with previous Wilcoxon signed-rank tests that failed to significantly distinguish the performance of the competing approaches based on the $AvgCost$ criterion.

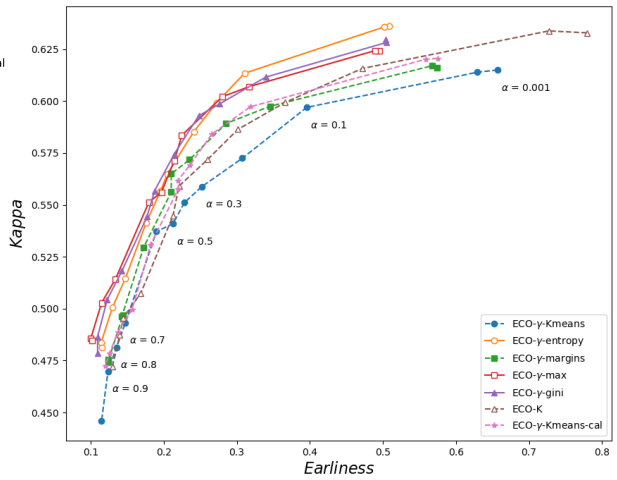


Fig. 6: Average Earliness vs. Average Kappa score obtain over the 33 datasets by varying the slope of the time cost, such as $\alpha \in [10^{-3}, 1]$.

VI. Conclusion

An increasing number of applications require the ability to recognize the class of an incoming time series as early as possible without unduly compromising the accuracy of the prediction. In response to this problem, the best performing *early classification* approaches [3] takes into account both the cost of misclassification and the cost of delaying the decision, and are able to anticipate the expected future gain in information in balance with the cost of waiting. Especially, ECONOMY-γ is the state of the art algorithm but it is limited to binary classification problems.

In this paper, we proposed two leads to extend ECONOMY-γ to multiclass problems: (i) by using a *confidence score* that aggregates the probabilities estimated by the classifier into a scalar value; (ii) by using a *clustering* algorithm in the vector space formed by the estimated probabilities. The first lead has resulted in

several competing approaches that used entropy, Gini index, margins, and maximum probability as confidence scores. In addition, we proposed two approaches derived from the second lead which used the K-means algorithm on the probabilities estimated by the classifier, with an optional calibration step.

Extensive experiments on 33 datasets of multiclass classification problems allowed us to compare the performance of the six proposed approaches to the state-of-the-art method [18]. Our experiments show that: (i) all proposed methods perform significantly better than the state of the art method; and (ii) the best way to extend ECONOMY-γ to multiclass problems is to use a confidence score, either the Gini index or the maximum probability.

REFERENCES

[1] C. J. Alonso González and J. J. R. Diez, "Boosting interval-based literals: Variable length and early classification," in *Data mining in time series databases*. World Scientific, 2002, pp. 149–171.

[2] A. Dachraoui, A. Bondu, and A. Cornuéjols, "Early classification of time series as a non myopic sequential decision making problem," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2015, pp. 433–447.

[3] Y. Achenchabe, A. Bondu, A. Cornuéjols, and A. Dachraoui, "Early classification of time series. cost-based optimization criterion and algorithms," *to appear in Machine Learning Journal (MACH) - arXiv preprint arXiv:2005.09945*, 2020.

[4] J. Lines, S. Taylor, and A. Bagnall, "Time series classification with hive-cote: The hierarchical vote collective of transformation-based ensembles," *ACM Transactions on Knowledge Discovery from Data*, vol. 12, no. 5, 2018.

[5] A. Bagnall, L. Davis, J. Hills, and J. Lines, "Transformation based ensembles for time series classification," in *Proceedings of the 2012 SIAM international conference on data mining*. SIAM, 2012, pp. 307–318.

[6] A. Bondu, D. Gay, V. Lemaire, M. Boullé, and E. Cervenka, "Fears: a feature and representation selection approach for time series classification," in *Asian Conference on Machine Learning*. PMLR, 2019, pp. 379–394.

[7] H. S. Anderson, N. Parrish, K. Tsukida, and M. Gupta, "Early time-series classification with reliability guarantee," *Sandria Report*, 2012.

[8] N. Parrish, H. S. Anderson, M. R. Gupta, and D. Y. Hsiao, "Classifying with confidence from incomplete information," *J. of Mach. Learning Research*, vol. 14, no. 1, pp. 3561–3589, 2013.

[9] P. Schäfer and U. Leser, "Teaser: early and accurate time series classification," *Data Mining and Knowledge Discovery*, vol. 34, no. 5, pp. 1336–1362, 2020.

[10] Z. Xing, J. Pei, and P. S. Yu, "Early prediction on time series: A nearest neighbor approach," in *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, ser. IJCAI'09. Morgan Kaufmann Publishers Inc., 2009, p. 1297–1302.

[11] U. Mori, A. Mendiburu, S. Dasgupta, and J. Lozano, "Early classification of time series from a cost minimization point of view," in *Proceedings of the NIPS Time Series Workshop*, 2015.

[12] Z. Xing, J. Pei, S. Y. Philip, and K. Wang, "Extracting interpretable features for early classification on time series." in *SDM*, vol. 11. SIAM, 2011, pp. 247–258.

[13] M. F. Ghalwash, V. Radosavljevic, and Z. Obradovic, "Utilizing temporal patterns for estimating uncertainty in interpretable early decision making," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 402–411.

[14] G. He, Y. Duan, R. Peng, X. Jing, T. Qian, and L. Wang, "Early classification on multivariate time series," *Neurocomputing*, vol. 149, pp. 777–787, 2015.

[15] N. Hatami and C. Chira, "Classifiers with a reject option for early time-series classification," in *Computational Intelligence and Ensemble Learning (CIEL), 2013 IEEE Symposium on*. IEEE, 2013, pp. 9–16.

[16] M. F. Ghalwash, D. Ramljak, and Z. Obradović, "Early classification of multivariate time series using a hybrid hmm/svm model," in *2012 IEEE International Conference on Bioinformatics and Biomedicine*. IEEE, 2012, pp. 1–6.

[17] U. Mori, A. Mendiburu, I. M. Miranda, and J. A. Lozano, "Early classification of time series using multi-objective optimization techniques," *Information Sciences*, vol. 492, pp. 204–218, 2019.

[18] U. Mori, A. Mendiburu, S. Dasgupta, and J. A. Lozano, "Early classification of time series by simultaneously optimizing the accuracy and earliness," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 10, pp. 4569–4578, 2017.

[19] V. Vapnik and A. Vashist, "A new learning paradigm: Learning using privileged information," *Neural networks*, vol. 22, no. 5-6, pp. 544–557, 2009.

[20] P. A. Flach, "Classifier calibration," *Encyclopedia of Machine Learning and Data Mining*, pp. 1–8, 2016.

[21] F. Zhdanov, "Diverse mini-batch active learning," *arXiv:1901.05954 [cs.LG]*, 2019.

[22] V. Lemaire, O. A. Ismaili, A. Cornuéjols, and D. Gay, "Predictive k-means with local models," in *Trends and Applications in Knowledge Discovery and Data Mining*, W. Lu and K. Q. Zhu, Eds. Springer International Publishing, 2020, pp. 91–103.

[23] V. Lemaire, F. Clérot, and N. Creff, "K-means clustering on a classifier-induced representation space : application to customer contact personalization," *Annals of Information Systems, Special Issue on Real-World Data Mining Application*, no. Special Issue on Real-World Data Mining Application, pp. 139–153, 2015.

[24] M.-F. Balcan, A. Broder, and T. Zhang, "Margin based active learning," in *International Conference on Computational Learning Theory*. Springer, 2007, pp. 35–50.

[25] L. Desreumaux and V. Lemaire, "Learning active learning at the crossroads? evaluation and discussion," in *Proceedings of the Workshop on Interactive Adaptive Learning, co-located with (ECML PKDD 2020)*, ser. CEUR Workshop Proceedings, D. Kottke, G. Krempl, V. Lemaire, A. Holzinger, and A. Calma, Eds., vol. 2660. CEUR-WS.org, 2020, pp. 38–54. [Online]. Available: http://ceur-ws.org/Vol-2660/ialatecml_paper3.pdf

[26] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '07. USA: Society for Industrial and Applied Mathematics, 2007, p. 1027–1035.

[27] M. Rußwurm, S. Lefevre, N. Courty, R. Emonet, M. Körner, and R. Tavenard, "End-to-end learning for early classification of time series," *arXiv preprint arXiv:1901.10681*, 2019.

[28] M. Barandas, D. Folgado, L. Fernandes, S. Santos, M. Abreu, P. Bota, H. Liu, T. Schultz, and H. Gamboa, "Tsfel: Time series feature extraction library," *SoftwareX*, vol. 11, p. 100456, 2020, https://github.com/fraunhoferportugal/tsfel.

[29] P. Nemenyi, "Distribution-free multiple comparisons," *Biometrics*, vol. 18, no. 2, p. 263, 1962.

[30] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and psychological measurement*, vol. 20, no. 1, pp. 37–46, 1960.