# Arbitrated Dynamic Ensemble with Abstaining for Time Series Forecasting on Data Streams

Dihia Boulegane[1,2], Albert Bifet[1], Sara El-Bouch[2], and Giyyarpuram Madhusudan[2]

[1] Télécom ParisTech 46 Rue Barrault, 75013 Paris, France
[2] Orange Labs Meylan 28 Chemin du Vieux Chêne, 38240 Meylan France

**Abstract.** A well-known challenge in mining temporal data streams is their dynamic nature where changes and recurrent concepts are likely to happen. Ensemble methods are powerful techniques to improve overall accuracy and tackle the aforementioned challenges by combining several classifiers. Dynamic Ensemble Selection (DES) allows selecting on the fly, the most accurate models only to contribute to the final output. This is motivated by the assumption that ensemble components have different degrees of expertise on different sub-spaces of the data. Existing DES methods are shaped either to batch learning or classification tasks but less suited to stream mining and forecasting tasks. In this paper, we propose a new Arbitrated Dynamic Ensemble Selection technique STREAMING-ADE for time series forecasting on data streams that takes advantage of meta-learning to monitor the predictive power of ensemble components and accordingly select and weight experts. Our selection is based on an abstaining policy where poorly performing classifiers are excluded from experts committee. Besides, since diversity is a critical need in ensemble methods, we propose to explicitly handle the interdependence of classifiers when selecting experts and not only their predictive power. Our contribution is threefold: (i) we introduce two different approaches of abstaining: threshold-based and random-based selection and (ii) we adapt a set of ensemble diversity measures for forecasting tasks to meet the requirements of streaming data (iii) we conduct an extensive experimental study to compare different methods on both real-world and synthetic time series.

**Keywords:** Dynamic Ensemble Selection, Meta-Learning, Data Stream Mining, Time Series Forecasting, Diversity Measures

## 1 Introduction

Mining time-evolving streams is a challenging task due to the dynamic nature of data. Streams are often non-stationary and include different regimes, drifts, and recurrent concepts. Therefore, learning algorithms should be able to detect and adapt to these changes as soon as they occur while avoiding a high rate of false alarms. Combining the outputs of several models, noted as ensemble methods or Multiple Classifier Systems (MCS ) [34], is a common practice in the data mining field to improve the correctness of predictions. Ensemble methods are particularly well suited to tackle the challenges of stream data mining as different classifiers have varying degrees of expertise on different sub-spaces of the data. Besides, ensembles adopt a highly attractive approach

allowing to add classifiers trained on recent data whereas classifiers representing outdated data can be pruned [19]. Ensemble methods are based on the individual accuracy of base-learners but also on how different they behave from one another, also known as *diversity*. In classifier combination, *diversity* is vital for the success of the ensemble, otherwise, there will be no gain from combining several classifiers [20].

One of the most promising approaches of MCS is Dynamic Ensemble Selection (DES) where only a subset of base classifiers, referred to as committee or Ensemble of Classifiers (EoC), is selected on the fly according to each test instance. DES methods aim to only select the most accurate components according to the instance under study based on their estimated performance and accordingly weight and combine their outputs to predict the future values of the target. The rationale for these techniques is to select expert base-classifiers to contribute to the final output and exclude the less qualified ones from the committee. The problem of Dynamic Ensemble Selection can be considered as a meta-problem as described in [25]. The meta-problem uses different features describing the behavior of each base classifier $\mathrm{M^i}$ in the ensemble $\mathrm{M}$ to predict whether it is fairly competent to contribute to the final output on a given test instance and therefore be selected in the committee.

Ensemble methods have been extensively studied for the classification tasks [14] but less for forecasting and regression [19]. There are very few works addressing time series forecasting using dynamic ensemble selection. The Arbitrated Dynamic Ensemble (ADE ) proposed in [5] uses meta-learning based on Arbitrating architecture [23] to achieve Dynamic Ensemble Selection for time series forecasting. ADE uses a pool of base forecasters $\mathrm{M}$ trained off-line and a set of meta-learners $\mathrm{Z}$ where each meta-learner $\mathrm{Z^i} \in Z$ is trained to predict the Mean Squared Error (MSE) $\hat{e}^i_{t+1}$ of its base counterpart $\mathrm{M^i}$ when trying to predict $y_{t+1}$. The $\alpha\%$ best base-models with the lowest predicted errors are then selected in a committee ${}^{\alpha}M$ and their outputs weighted using the $\mathrm{softmax}$ function and combined. The ADE workflow is described in Figure 1.

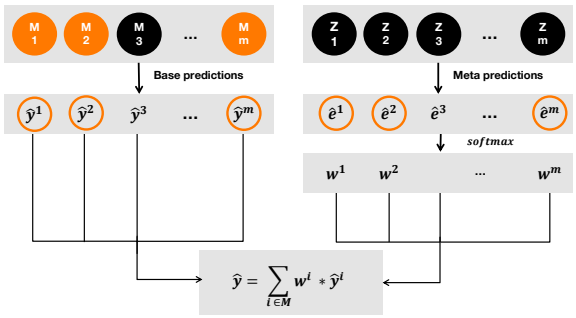


Fig. 1: Example of ADE workflow for each test instance. Each base-learner $\mathrm{M^i}$ predicts $\hat{y}^i, i \in \{1, ..., m\}$ to be the next value of the time series while the meta-learner $Z^i$ predicts the error $\hat{e}^i$ that its base counterpart will induce. Base-models are then selected and weighted using a $\mathrm{softmax}$ function. The final output $\hat{y}$ is a weighted average of selected base-models predictions.

ADE has led to very promising results but still limited when it comes to data streams. That is because base-models are trained once offline but never updated which makes ADE unsuitable for novel concepts. Moreover, the $\alpha\%$ best base-models are selected based on their relative performance regardless of the absolute value of the predicted error, probably leading the ensemble to perform very poorly in case of higher errors that may coincide with a concept drift. Another crucial aspect to be considered in ensemble methods is inter-dependence of base-learners. Selected models should be diverse in order to reduce redundancy otherwise, there will be no gain in combining them. A more sophisticated version of ADE was proposed in [6] that explicitly models redundancy among selected experts when aggregating their predictions. However, while diversity has been extensively studied in the context of static data, it has not still received equivalent interest for evolving data streams. Authors in [4] proposed an adaptation of the calculation of some diversity measures to meet data stream requirements. Besides, in contrast to the classification task [21], diversity measures have received very limited attention when it comes to continuous values and notably, forecasting tasks.

Our main contributions in this paper are: (i) We propose STREAMING-ADE , a new ensemble method that extends ADE [5], which supports the traditional batch learning, to the streaming setting (ii) We propose different selection approaches based on abstaining where the competence level of base-models is considered to exclude uncertain ones from experts committee (iii) We present different diversity measures that can be used to quantify base-models inter-dependence and their adaptation to stream processing requirements (iv) We validate our approach with an extensive empirical study on several real and synthetic time series to emphasize on the contribution of Dynamic Ensemble Selection and diversity on global accuracy. To the best of our knowledge, this is the first work that addresses time series forecasting using meta-learning for Dynamic Ensemble Selection on data streams using abstaining policy and explicitly modeling ensemble's diversity. The proposed STREAMING-ADE approach is generic and can be used with different classifiers on different types of data.

The rest of the paper is organized as follows. Section 2 outlines the related work of Dynamic Ensemble Selection using meta-learning and addresses the problem of diversity within ensembles of forecasters and streaming setting. The proposed STREAMING-ADE approach is described in Section 3 where we formalize the meta-learning problem and introduce different selection approaches. Furthermore, we detail different diversity measures for the forecasting task and their adaptation to streaming requirements. Experimental set-up along with the results are discussed in Section 4. Finally, Section 5 concludes the paper and tackles future directions of work.

## 2 Related Work

A data stream is a potentially infinite sequence where instances arrive rapidly over time. The streaming setting imposes a set of constraints to be considered in learning algorithms [2]. Due to the infinite size of the stream, one cannot store the entire data in memory. Moreover, each instance should be processed once and only once as quickly as possible to allow for real-time responsiveness. Finally, algorithms should be incremen-

tal and be able to detect and adapt to concept drift, and changes in the characteristics of the data [13].

An ensemble learning method is a set of complementary and diverse individual models (components) whose predictions are combined resulting in a better global prediction accuracy. The rationale is that not every classifier in the pool is an expert in all unknown samples. Rather, each base classifier is an expert in a different local region of the feature space [8]. Ensemble methods are widely studied for data streams due to their good performance compared to single learners as reported in [14]. Authors enumerated a plethora of algorithms dedicated to classification tasks, however, very little work addresses regression and forecasting tasks [17,24,29,30]. Ensembles are useful for mining data streams as they allow adaptation to changes, by adding new components trained on recent data, and removing components representing outdated data [18]. However, changes are often recurrent, hence removing outdated base-models will lead to forgetting useful historical knowledge that might be reused in the future [11]. One of the most encouraging techniques of ensemble methods is dynamic selection, where a set of components is selected on the fly according to each new test instance [8]. Nonetheless, identifying the best algorithm for a given test instance is not trivial. Practically, it can be seen as a meta-problem that uses different features describing the behavior of a base-classifier so as to determine whether it is competent enough to predict on a given test instance. Aside from components expertise, diversity is an essential aspect of ensemble methods. However, it is not trivial to quantify how different base-models behave when it comes to continuous values.

### 2.1 Dynamic Ensemble Selection Using Meta-Learning for Time Series Forecasting

Very encouraging results can be reached by simply averaging predictions of the available base forecasters [7, 22, 32]. The AEC (Adaptive Ensemble Combination) [28] is based on a windowing strategy that dynamically combines forecasters according to their past performance, including a forgetting factor to emphasize on more recent data. An adaptive combination of forecasters based on their recent coefficient of determination and simply averaging their outputs was studied in [33]. ADE uses a more proactive mechanism [5] where only the most accurate base-learners are selected to contribute to the final output. ADE is based on meta-learning and arbitrated architecture [23] where meta-learning allows to model the behavior of algorithms [3]. Classifier or Ensemble selection could be seen as a meta-problem [25] where the goal is to determine whether a base-model $\mathrm{M^i}$ from the pool $\mathrm{M}$ is competent enough to predict on a given test instance. The system uses a two-layered learning schema where each layer trains its own classifiers and receives its own data [11]. The base-learner $\mathrm{M^i}$ learns to predict future values $\hat{y}_{t+1}$ of the stream, whereas the meta-learner $\mathrm{Z^i}$ learns the behavior of its base counterpart $\mathrm{M^i}$ and predicts its future errors $\hat{e}^i_{t+1}$.

The use of meta-learning for dynamic classifier/ensemble selection has been widely investigated in the case of data streams for classification tasks but barely for forecasting. The MetaStream framework proposes to periodically select the most accurate regression algorithm or set of regressors using a prequential evaluation method on a sliding window [27]. A meta-example is generated for every window of size $w$ and a meta-classifier

is trained on the set of pre-computed meta-examples. Once a new meta-example is calculated, the meta-classifier predicts which regressor performs best to be used in the next window. The Online Performance Estimation proposed in [26] estimates the predictive power of base-models on data streams by measuring how ensemble components have performed on recent data and accordingly adjust their weights in the voting. The BLAST (**B**est **LAST**) framework for classification tasks is based on Online Performance Estimation and selects one of its base classifiers to be the only active model for every $w$ test examples.

Both MetaStream [27] and BLAST [26] are based on a user predefined window size $w$ which is not trivial to determine. ADE [5] proposes a more pro-active selection method based on an arbitrated architecture [23] where base-classifiers are trained offline to predict future values of the time series, whereas each meta-learner is responsible for predicting the loss that its base counterpart will incur at each test instance. The $\alpha\%$ best base-learners are then selected and weighted accordingly to their estimated performance to combine their outputs and predict the future value $y_{t+1}$ as described in Figure 1. The use of meta-learning for model selection on data streams classification and recurrent concepts was addressed in [12]. When a change is detected, a meta-learning algorithm decides whether a previously trained model on the same stream could be reused, otherwise a new model is trained on the deviating data. Another approach for ensemble selection was introduced in [18] based on an abstaining policy of base-models, where the less confident classifiers are forced to abstain from contributing to the final decision according to a dynamic threshold.

### 2.2 Diversity Measures in Ensemble of Forecasters

Ensembles methods are not only based on the individual accuracy of base-learners but also on how different they behave mutually. Diversity is the degree in which component classifiers make different predictions for the same test instance [20]. Diversity measures have been extensively studied for classification tasks but most of the measures deriving from the literature are not directly applicable to forecasting since they are dealing with continuous values. A myriad of diversity measures for ensembles of classifiers has been studied in [21] to asses the difference within an ensemble of classifiers. Similar attempts have been made to address diversity for continuous values in [10]. Despite the interest and extensive study of diversity in ensembles of classifiers, the problem of measuring diversity remains an open research-question.

In what follows, we present a set of measures proposed in [10] used to asses the diversity between two forecasters noted $\mathrm{M^i}$ and $\mathrm{M^j}$ with $\hat{Y}^i$ and $\hat{Y}^j$ their series of continuous outputs respectively. We assume that there are $N$ instances in each series and thus $\hat{Y}^i = [\hat{y}^i_1, \hat{y}^i_2, \ldots, \hat{y}^i_N]$ (resp. $\hat{Y}^j = [\hat{y}^j_1, \hat{y}^j_2, \ldots, \hat{y}^j_N]$) where $\hat{y}^i_t$ (resp. $\hat{y}^j_t$ ) stands for the prediction of $\mathrm{M^i}$ (resp. $\mathrm{M^j}$) to be the value of the time series $y_t$ at time $t$.

**Correlation Coefficient ($\rho$)** The Pearson's correlation coefficient $\rho$ between $Y^i$ and $Y^j$ is defined as

$$\rho_{i,j} = \frac{\sum_{t=1}^{N}(\hat{y}^i_t - \mu_{Y^i}) \times (\hat{y}^j_t - \mu_{Y^j})}{\sqrt{\sum_{t=1}^{N}(\hat{y}^i_t - \mu_{Y^i}) \times \sum_{t=1}^{N}(\hat{y}^j_t - \mu_{Y^j})}} \tag{1}$$

Where $\mu_{Y^i}$ (resp. $\mu_{Y^j}$), is the mean of $Y^i$ (resp. $Y^j$). Pearson's correlation is inversely proportional to the diversity of the two base-models. Models with low correlation are preferred over the ones with high correlation coefficient.

**Disagreement Measure (D)** the disagreement is a measure defined for classification as the probability that two classifiers will disagree on their decision. It is the ratio of instances on which one classifier is correct and the other one incorrect to the total number of instances. Table 1 provides ensemble components relationship where 1 (resp. 0) denotes correct (resp. incorrect) prediction and $N^{00} + N^{10} + N^{01} + N^{11} = N$. Disagreement is then defined in Equation 2.

Table 1: The 2x2 ensemble components relationship table with counts

| | $M^i$ **correct** | $M^i$ **wrong** |
|---|---|---|
| $M^j$ **correct** | $N^{11}$ | $N^{01}$ |
| $M^j$ **wrong** | $N^{10}$ | $N^{00}$ |

$$D_{i,j} = \frac{N^{01} + N^{10}}{N^{00} + N^{10} + N^{01} + N^{11}} \tag{2}$$

The Disagreement measure is intuitively applicable to classification tasks and can be extended to continuous-valued targets [10]. A prediction $\hat{y}_t^i$ is considered correct if and only if $y_t - \sigma < \hat{y}_t^i < y_t + \sigma$, otherwise it is considered incorrect. We define $\sigma$ as the standard deviation of the models' predictions. Disagreement can be measured by monitoring entries of Table 1 for each pair of base-learners and counting the number of their correct and incorrect predictions. Unlike for Pearson's correlation coefficient, true values of the target are required to compute $D_{i,j}$. Diversity decreases with the value of the disagreement.

**Double Fault (DF)** The double Fault measure counts the number of times both classifiers make mistakes. Therefore, Double Fault is defined as $DF_{i,j} = \frac{N^{00}}{N}$ expressed as a precentage using Table 1. Diversity decreases when the value of the double-fault increases.

### 2.3 Diversity Measures for Streaming Data

Many researchers claim that ensemble accuracy and diversity are closely related, thus it is of crucial importance to consider ensemble diversity when selecting experts committee. Nevertheless, such interest did not get the attention it deserves when it comes to data streams. Researchers assume that diversity results from the fact that ensemble components are learning from different parts of the stream representing different concepts but no measures adapted to stream processing requirements were proposed [4].

Diversity measures presented in Section 2.2 were designed to the batch setting and not to stream processing setting. We discuss in this section the ability of each measure to be adapted to stream requirements based on [4]. We discuss three mechanisms as follows:

- **Incremental**: In this setting, we assume that the diversity measure under study can be computed based only on a summary of all previous instances and a single new one. This is not trivial for all diversity measures but can be achieved for *Disagreement* and *Double Fault* if we monitor the number of processed instances and update all entries of Table 1 at each new instance.
- **Prequential**: This is based on incremental computing with forgetting in order to emphasize on more recent data. We investigate two approaches, *sliding window* and *fading factor*.
  - Sliding Window: keeps in memory the $d$ most recent instances of the stream. The window is updated with the latest instance and the oldest one is removed. Diversity measures are recomputed on the sliding window at each time point as for batch setting. All of *Correlation*, *Disagreement* and *Double Fault* can be intuitively adapted to the streaming setting.
  - Fading Factor: A fading factor $\lambda$ is used to blur old information. Diversity measures based on counts such as *Disagreement* and *Double Fault* can be easily adapted to this setting. Let $x$ be one of the entries of Table 1 and $N$ be the number of processed instances. Computation with $\lambda$ fading factor can be achieved as described in Equation 3.

$$\begin{aligned} S_{x,\lambda}(t) &= x_t + \lambda \times S_{x,\lambda}(t-1) \\ N_{\lambda}(t) &= 1 + \lambda \times N_{\lambda}(t-1) \end{aligned} \tag{3}$$

We present in Section 3 our proposed STREAMING-ADE framework that performs Arbitrated Dynamic Ensemble Selection based on abstaining using meta-learning and explicitly includes diversity when selecting the committee of experts.

## 3 Streaming Arbitrated Dynamic Ensemble

A time series $Y$ is a sequence $Y = \{y_1, y_2, \ldots, y_t\}$, where $y_t$ is the value of the series at time $t$. We use time-delay embedding [31] to represent $Y$ in a $K$ dimensional space where each instance $y_t$ is represented in a vector $v_t$ using the $K$ past lags (values) of the time series, that is, $\boldsymbol{v_t} = \langle y_{t-(k-1)}, y_{t-(k-2)}, \ldots, y_t \rangle$ . Our proposed method STREAMING-ADE uses meta-learning to perform dynamic ensemble selection and explicitly considers diversity to reduce redundancy within the committee of experts. It is based on, (i) incremental learning of models in both meta-ensemble $Z$ and base-ensemble $M$ (ii) a dynamic selection of base-models to build a committee of experts $'\mathrm{M}$ based on their predicted performance for each incoming instance. We explore two different approaches: threshold-based and random-based. (iii) an explicit mechanism to measure and include diversity with the aim to reduce redundancy within selected experts. (iv) a fusion approach that combines all base-experts' $\mathrm{M}^{\mathrm{i}} \in {}'\mathrm{M}$ respective predictions $\hat{y}^i_{t+1}$ weighted according to their predicted errors $\hat{e}^i_{t+1}$.

### 3.1 Meta-Learning Layer

The meta-level is composed of an ensemble of meta-models $Z = \{Z^1, Z^2, \ldots, Z^m\}$ where each component $\mathrm{Z}^{\mathrm{i}}$ is in charge of predicting at time $t$ future error $\hat{e}^i_{t+1}$ of its

base counterpart $\mathrm{M^i}$. MSE was used in [5] to evaluate base-learners competence, however, this latter would not be in line with our approach since the value of the error is considered during the selection, unlike ADE . This limitation raises the need for a relative measure such as Symmetric Mean Percentage Error ($\mathrm{SMAPE}$ [15]) The $\mathrm{SMAPE}$ is defined in Equation (4)

$$\mathrm{SMAPE} = \begin{cases} 0, & \text{if } y_t, \hat{y}_t = 0. \\ \frac{100}{N} * \sum_{t=1}^{N} 2 \times \frac{|\hat{y}_t - y_t|}{|\hat{y}_t| + |y_t|} & \text{otherwise.} \end{cases} \tag{4}$$

such that $N$ is the number of instances seen. For a more comprehensive error measure, we remove the factor 2 in the numerator to make predicted errors $\hat{e}^i \in [0, 1]$ expressed as a percentage.

### 3.2 Meta-Model's Confidence

STREAMING-ADE selection relevance relies heavily on the accuracy of meta-models predictions. Therefore we introduce a confidence measure to quantify to what extent the meta-model $\mathrm{Z^i}$ was accurate in predicting the errors of its base counterpart $\mathrm{M^i}$ on past instances. The confidence $c_t^i \in ]0, 1]$ of a meta-model $\mathrm{Z^i}$ at time $t$ is defined in Equation 5 as:

$$\begin{aligned} c_t^i &= \exp(-\mathrm{MSE}_t^i) \\ \mathrm{MSE}_t^i &= \frac{1}{N} * ((\hat{e}_t^i - e_t^i)^2 + \lambda * \mathrm{SSE}_{t-1}^i) \\ \mathrm{SSE}_{t-1}^{\mathrm{i}} &= \sum_{k=1}^{t-1} (\hat{e}_k^i - e_k^i)^2 \end{aligned} \tag{5}$$

with $N$ the number of instances processed in the stream up to time $t$, $\lambda \in [0, 1]$ a fading factor to emphasize on more recent instances and $SSE$ the Sum of Squared Errors. The measure of confidence $c_t^i$ is used to tackle concept drift and meta-models relevance over time. If a concept drift happens at time $t = t_c$, a meta-model $\mathrm{Z^i}$ is likely to fail in predicting $e_{t_c}^i$ which increases $\mathrm{MSE}_{t_c}^i$ and thus drops the confidence level of the meta-model $c_{t_c}^i$. If the confidence of a meta-model is low, its predicted errors is very unlikely to be taken into account while selecting experts.

### 3.3 Base-Model Selection

Selection mechanism adopted in ADE is static with the $\alpha\%$ base-models having the lowest predicted errors being selected regardless of their values. This approach may be limited when all base-models are predicted to fail drastically in the presence of a concept drift. Moreover, the value of $\alpha$ is not trivial to determine, setting it too small may exclude experts from the committee whereas setting it too large may include poorly performing ones. We suggest addressing this shortcoming by introducing an abstaining policy, where the less confident base-models are excluded from the committee. We introduce two different selection approaches : threshold-based and random-based.

**Threshold-Based Selection** A base-model $\mathrm{M}^{\mathrm{i}}$ is considered expert if and only if $\hat{e}^i_{t+1} < \theta$ where, $\theta$ is a user-defined threshold to set the maximum tolerated value of SMAPE. The threshold $\theta$ is dynamic and self-adaptive as discussed by [18]. If ensemble prediction $\hat{y}_{t+1}$ is correct, this means that we have selected competent classifiers and we may increase $\theta$ to seek for similarly good base-models. On the other hand, if the prediction is wrong, we need to decrease $\theta$ to exclude poorly performing base-models form the committee. We explore two different update strategies: **iterative** where $\theta \leftarrow \pm s$ and **multiplicative** where $\theta \leftarrow \theta(1 \pm s)$, with $s \in [0, 1]$ is a user defined parameter.

**Random-Based Selection** Even though the threshold-based selection offers a relative and adaptive threshold $\theta$, it is not trivial to set its value and requires prior human knowledge. Hence, we study a selection approach based on a random Bernoulli distribution. We model the selection of $\mathrm{M}^{\mathrm{i}}$ to predict at time $t+1$ as a Bernoulli trial with a variable parameter $p^e_{t+1} = 1 - \hat{e}^i_{t+1}$ which means that a base-model is selected with probability $p^e_{t+1}$. The smaller the error is, the greater is the probability of $\mathrm{M}^{\mathrm{i}}$ to be selected. However, concept drift may happen anywhere in the stream [35]. Thus, it might be interesting, from time to time, to select other base-models with high predicted errors and inversely prune the ones with low predicted errors. Algorithm 1 describes the random-based selection process.

**Algorithm 1** Error Random-Based Selection

**Input**: New test instance from $S$
**Output:** ${}^rM \in M$ of selected experts

1: Let ${}^rM = \emptyset$
2: **for** $M^i \in M$ **do**
3: $\quad \hat{e}^i_{t+1} \leftarrow$ get prediction from $\mathrm{Z}^{\mathrm{i}}$
4: $\quad$ generate $r_e \sim Bernoulli(1 - \hat{e}^i_{t+1})$
5: $\quad$ **if** $r_e = 1$ **then** add $\mathrm{M}^{\mathrm{i}}$ to ${}^rM$
6: **end for**
7: **return** : ${}^rM$

**Algorithm 2** Confidence-Error Random-Based Selection

3: $\hat{e}^i_{t+1} \leftarrow$ get prediction from $\mathrm{Z}^{\mathrm{j}}$
4: $\hat{c}^i_t \leftarrow$ get confidence of $\mathrm{Z}^{\mathrm{i}}$
5: generate $r_e \sim Bernoulli(1 - \hat{e}^i_{t+1})$
6: generate $r_c \sim Bernoulli(\hat{c}^i_t)$
7: **if** $r_e = 1$ **and** $r_c = 1$ **then**
8: $\quad$ add $\mathrm{M}^{\mathrm{j}}$ to ${}^rM$
9: **end if**

We introduce a more restrictive randomized selection using the confidence level of meta-learners discussed in Section 3.2. We model the relevance of a meta-model $\mathrm{Z}^{\mathrm{i}}$ to predict on time $t+1$ as a Bernoulli trial of parameter $p^c_{t+1} = c^i_t$. The greater the confidence is, the more relevant is the meta-model. Moreover, this approach allows to detect and react to concept drifts. If a change occurs in the stream at time $t = t_c$, a meta-model $\mathrm{Z}^{\mathrm{i}}$ has most likely failed in predicting the error $e^i_{t_{c+1}}$. Consequently, the error $\mathrm{MSE}^i_{t_{c+1}}$ increased considerably leading the confidence $c^j_{t_{c+1}}$ to drop. A low meta-confidence $c^i_t$ can be translated to $\mathrm{M}^{\mathrm{i}}$ being pruned from the committee. Therefore, a base-model $\mathrm{M}^{\mathrm{i}}$ is selected if and only if the latter and its meta-model counterpart are selected through the two random processes. Algorithm 1 is changed starting from line 4 and considers another random variable $r_c$ related to the confidence level of meta-models as shown in Algorithm 2.

### 3.4 Handling Experts Diversity and Redundancy

Ensemble methods require : (a) training procedures that result in relatively independent experts (b) and aggregation methods that explicitly or implicitly model the dependence among experts [16]. Requirement (a) is addressed by focusing on heterogeneous ensembles only as we use a combination of different learning algorithms having different inductive biases. On the other hand, requirement (b) is addressed by explicitly measuring diversity among base-models and accordingly selecting poorly dependent experts. We introduce a twofold selection as follows : (i) we build a committee $'\mathrm{M}$ comprising the best performing models according to one of the methods discussed in 3.3 (ii) we reduce $'\mathrm{M}$ to ${}^{\mathrm{d}}\mathrm{M} \subseteq {}'\mathrm{M}$ to exclude redundant models. We first select the best model from the committee to be a seed in ${}^{d}M$. A model $\mathrm{M}^{\mathrm{i}}$ from the remaining in $'M$ is added if and only if it is independent from all previously selected experts in ${}^{\mathrm{d}}\mathrm{M}$. Two models $\mathrm{M}^{\mathrm{i}}$ and $\mathrm{M}^{\mathrm{j}}$ are said independent if their pairwise diversity $div_{i,j}$ is above a predefined threshold $\theta_{div}$. Out of the two steps, we obtain a committee ${}^{\mathrm{d}}\mathrm{M}$ of diverse experts.

### 3.5 STREAMING-ADE

We dwell in this section on the proposed approach STREAMING-ADE . In contrast to the ADE method proposed in [5], our approach uses an abstaining policy where only the most competent base-learners are allowed to contribute to the final output. We address the issue of redundancy within selected committee by explicitly measuring pairwise diversity and filtering experts based on their inter-dependence. Poorly dependent experts are preferred over highly dependent ones.

The proposed diversity-based filtering lines-up with the approach proposed in [6] that re-weights experts based on their pairwise correlation. Algorithm 3 provides detailed steps of STREAMING-ADE where performance-based selection is achieved in line 3 using one of the methods presented in Section 3.3. The diversity-based filtering is performed at line 4 aiming to reduce redundancy using one of the diversity measures and stream computation mechanism detailed in Section 2.3. Once the twofold selection is finished, we compute a weight $w^i_{t+1}$ for each $\mathrm{M}^{\mathrm{i}} \in {}^{\mathrm{d}}\mathrm{M}$ using meta-predictions obtained in line 6 according to the $\mathrm{softmax}$ function. Base-experts outputs $\hat{y}^i_{t+1}$ are then combined to get the final output $\hat{y}_{t+1}$ (line 9). The rest of the algorithm is used to update base and meta-models when getting the true value of $y_{t+1}$ and possibly the threshold $\theta$ if needed according to one of the strategies discussed in 3.3. All meta and base models keep learning on all instances of the stream.

## 4 Experimental Study

### 4.1 Ensemble Set-up

We compare several ensemble methods with different selection and redundancy reduction strategies as described in Table 2. Selection methods discussed in Section 3.3 are detailed in Line 1 whereas Line 2 describes the diversity handling approaches described in Section 3.4. The proposed STREAMING-ADE is compared against two simple baseline ensemble methods: **NAIVE** where all base-learners have the same weight and all

**Algorithm 3** Streaming Arbitrated Dynamic Ensemble

**Input**: Infinite stream $S = \{y_1, y_2, \ldots, y_t, \ldots\}$,
$\mathrm{M} = \{\mathrm{M}^1, \mathrm{M}^2, \ldots, \mathrm{M}^{\mathrm{m}}\}$,
$\mathrm{Z} = \{\mathrm{Z}^1, \mathrm{Z}^2, \ldots, \mathrm{Z}^{\mathrm{m}}\}$,
**Parameter**: Selection strategy
Threshold $\theta$, update step $s$ // If threshold-based selection
**Output:** A prediction $y_{t+1}$ for each time $t$

1: **while** *end of stream* = False **do**
2: Obtain new instance using time-embedding on $k$ lags
3: Let $'\mathrm{M}$ be the set of selected base-models using one of the three methods in Section 3.3
4: Let ${}^{\mathrm{d}}\mathrm{M}$ be the set of diverse experts stemming from $'\mathrm{M}$ according to Section 3.4
5: Let ${}^{\mathrm{d}}\mathrm{Z}$ be the set of ${}^{\mathrm{d}}\mathrm{M}$ meta-models counterparts
6: Get meta-predictions $\hat{e}^i_{t+1}$ from $\mathrm{Z}^{\mathrm{i}} \in {}^{\mathrm{d}}\mathrm{Z}$
7: Compute weights $w^i_{t+1} = \frac{\exp(-\hat{e}^i_{t+1})}{\sum_{\mathrm{Z}^{\mathrm{j}} \in {}^{\mathrm{d}}\mathrm{Z}} exp(-\hat{e}^j_{t+1})}$
8: Get predictions $\hat{y}^i_{t+1}$ from every $\mathrm{M}^{\mathrm{i}} \in {}^{\mathrm{d}}\mathrm{M}$
9: Compute final prediction $\hat{y}_{t+1} = \sum_{\mathrm{M}^{\mathrm{i}} \in {}^{\mathrm{d}}\mathrm{M}} \hat{y}^i_{t+1} * w^i_{t+1}$
10: Output $\hat{y}_{t+1}$
11: Obtain true value of $y_{t+1}$
12: Update all base-models $\mathrm{M}^{\mathrm{i}} \in M$
13: Update all meta-models $\mathrm{Z}^{\mathrm{i}} \in Z$
14: Update threshold $\theta$ // if threshold-based selection
15: **end while**

contribute to the final output whereas the **WEIGHTED** allows all base-learners to contribute and weigh their contributions based on meta-predictions as in Algorithm 3 Line 6 and 7. We also compare our method to the proposed selection proposed in ADE [5] noted **PERCENT**. Table 2 details all selection methods and diversity measures along with their streaming computation mechanism as shown in Section 2.3. The abbreviations presented will be re-used in Section 4.3 to facilitate naming.

For the sake of diversity, STREAMING-ADE is based on *heterogeneous* ensembles by combining base-learners inducing different biases. We have used four online classifiers as base-learners : Adaptive Hoeffding Tree [1], Hoeffding Tree [9], K-Nearest Neighbors (KNN) and a weighted version of the KNN, where each neighbor is inversely weighted to its distance to the test instance. We have used different parameter settings for each base-learner to ensure diversity within the base-ensemble. All meta-models were implemented using an Adaptive Hoeffding Tree with default parameters setting. Our implementation is based on scikit-multiflow[3], an open-source package written in python.

We have set general parameters such as ensemble size to $m = 30$ and time embedding lags to $k = 7$. Parameters related to threshold-based selection approach involve competence threshold, set to $\theta = 20\%$ and update step $s = 0.01$. Finally, diversity measures computation parameters selection parameters are related measures computa-

[3] Available at https://scikit-multiflow.github.io/

Table 2: Methods naming and their respective symbols

| | Symbol | Description | Variants and Parameters |
|---|---|---|---|
| Selection methods | **TH** | Threshold-based selection | – $\theta$: Competence threshold<br>– Update method (Section 3.3):<br>– **ST**: static (no update)<br>– **PR** multiplicative<br>– **SUM**: additive |
| | **PROB** | Random Based Selection | **CONF**: when confidence and error selection (Algorithm 2) |
| | **PERCENT** | Select the $\alpha\%$ best models [5] | $\alpha = 50\%$ |
| Diversity methods | **SEQ-W** | Sequential re-weighting [6] | redistribute weights of selected experts according to their redundancy |
| | **DV** | Use of diversity measures for two-fold selection (Section 3.4) | – Diversity measures:<br>– **CORR**: Correlation<br>– **DIS**: Disagreement<br>– **DF**: Double Fault<br>– Computation:<br>– **IC**: Incremental<br>– **SW**: Sliding window<br>– **FF**: Fading factor $\lambda$ |

tion such as fading factor $\lambda = 0.995$ and sliding windows whereas diversity threshold $\theta_{div} = 0.75$ stands for the minimum pairwise diversity to assess the independence of two base-models.

### 4.2 Data sets

Table 3 describes some properties of several real and synthetic times series used to validate the proposed approach STREAMING-ADE in our experiments.

Table 3: Data sets description

| Dataset | nb | Min length | Max length | Avg length |
|---|---|---|---|---|
| Real [4] | 48 | 946 | 3000 | 2369 |
| Blockchain[5] | 2 | 1264 | 1267 | 1265 |
| BTC-USD[6] | 3 | 3235 | 3235 | 3235 |
| Synthetic | 2 | 8000 | 8000 | 8000 |

We have used $48$ real data sets covered in [6] (Real) in addition to bitcoin data (Blockchain, BTC-USD) describing the daily crypto-currency prices, number of transactions, cost per transaction and many others metrics from 2009-01-31 until 2019-05-18. On the other hand, we have generated synthetic data to model concept drift and

[4] Available at https://github.com/vcerqueira/forecasting_experiments/tree/master/data

[5] Available at https://www.blockchain.com/en/stats

[6] https://github.com/melvfnz/data_science_portfolio/blob/master/Cryptocurrency Market Analysis.ipynb

recurrent patterns in data streams. We have created 2 synthetic time series by concatenating sequences of data generated from different Auto-Regressive models, where each sequence stands for a different concept. We have injected white Gaussian noise along the time series. All synthetic data can be reproduced using TimeSynth[7].

### 4.3 Results

We report in this section the results of our experimental study described in 4.1. We analyzed the performance of all methods using the Mean Squared Error (MSE). We First discuss the contribution of explicitly including diversity criteria in the two selection strategies discussed in Section 3.3 against sequential re-weighting [6]. Finally, we compare the overall accuracy of the proposed STREAMING-ADE against both ADE and baseline ensemble methods **WEIGHTED** and **NAIVE** described in Section 4.1.

**Diversity measures Vs. sequential re-weighting** : Figure 2 reports methods rankings (by increasing $MSE$) using prequential evaluation over all data sets described in Table 3. Sub-figure 2a summarizes the average and median of Random-based approach rankings with all its variants that include diversity measures (**DV**) or sequential re-weighting (**SQW**). Results show that explicit pruning based on diversity measures yields very bad results whereas the use of sequential re-weighting slightly improves performance over simple selection. Results also suggest that the dual random selection using meta-models confidence (Section 3.2) outperforms selection based on predicted errors only.

In the same way, threshold-based methods using diversity-filtering techniques perform worse than their simple counterparts in most cases. Sub-figure 2b depicts all variants using diversity measures presented in Section 2.2 performance. However, the improvement of sequential re-weighting is less significant as opposed to random-based selection. Our intuition is that the correlation is necessarily very high between models' of which predictions lay within a small interval defined by $\theta$. This makes the global prediction very close to individual base-predictions and thus lowers the improvement of sequential re-weighting. Conversely, random-based selection leaves a little more room for diversity within experts especially when dealing with border values of $\hat{e}_t^i$ (Section 3.3).

Out of the two figures, the use of Disagreement (**DIS**) to measure diversity scored the lowest compared to any other diversity measures whereas sequential re-weighting provides a lightweight improvement at the detriment of time. We may suggest that formalizing diversity within ensemble of forecasters is a delicate task that requires more specialized measures designed for continuous values and stream processing requirements. Besides, the twofold selection may be too narrow and thus go against the rationale of ensemble methods, unlike the **SQW** approach that only re-weighs experts and does not prune any of them.

**Global comparison** : We compare below all selection methods (**PERCENT**, **TH**, **PROB**) with and without sequential re-weighting to assess the advantage of reducing the redundancy among selected experts to increase overall accuracy. Remarkably,

[7] Available at https://github.com/TimeSynth/TimeSynth

(a) Random-based selection (b) Threshold-based selection

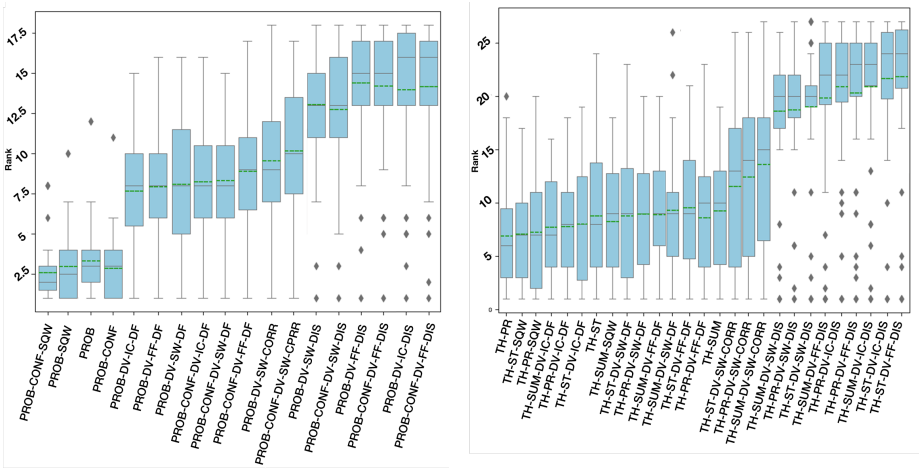


Fig. 2: Random and Threshold selection and their variants performance by increasing $MSE$ where dashed (resp. continuous) line stands for the mean (resp. median)

all random and threshold based-methods perform better then the approach noted **PERCENT** used in ADE [5]. These results motivate our abstaining policy that excludes poorly performing base-learners based on their predicted errors in contrast to ADE that selects the $\alpha\%$ best base-learners regardless of the values of the errors. Furthermore, sequential re-weighting considerably improves the overall performance of the dynamic ensemble selection as shown in Figure 3. However, random-based selection using meta-models confidence, noted **AB-PROB-SQW-CONF**, surpasses all the other methods. The results achieved assess the ability of the proposed STREAMING-ADE to cope with concept drift on data streams. Randomness allows looking from time to time to other base-models that were predicted to fail that may be efficient in detecting a concept drift and most likely help in adaptation.

## 5 Conclusions

We have presented STREAMING-ADE , an Arbitrated Dynamic Ensemble Selection framework for data streams that uses meta-learning to select a committee of experts on the fly according to each test instance. We have proposed different selection approaches where the less confident base-models are forced to abstain. We address the problem of redundancy by pruning redundant base-learners from the committee of experts. Experimental results show that STREAMING-ADE abstaining policy based on random selection with regards to meta-models confidence are particularly effective in predicting future values of times-series when dealing with concept drifts. Future works will involve : (i) Ensemble size monitoring where new models can be added and outdated ones can be pruned with regards to recurrent patterns. (ii) A trade-off selection where both predictive power and diversity are considered at the same time unlike the twofold,

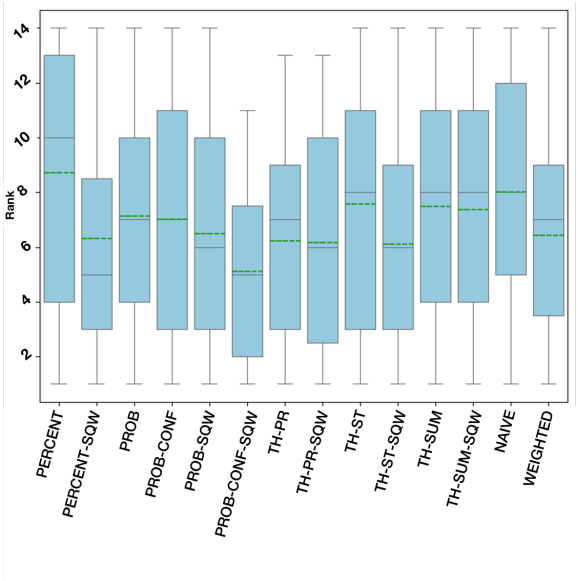


Fig. 3: Boxplots of PERCENT, TH and PROB based selection methods rankings

that we believe is very constraining. (iii) Diversity within data where several ensembles of base-learners can be trained on a disjoint subspaces of the data stream.

## References

1. Bifet, A., Gavaldà, R.: Adaptive learning from evolving data streams. In: International Symposium on Intelligent Data Analysis. pp. 249–260. Springer (2009)
2. Bifet, A., Holmes, G., Kirkby, R., Pfahringer, B.: Moa: Massive online analysis. Journal of Machine Learning Research **11**(May), 1601–1604 (2010)
3. Brazdil, P., Carrier, C.G., Soares, C., Vilalta, R.: Metalearning: Applications to data mining. Springer Science & Business Media (2008)
4. Brzezinski, D., Stefanowski, J.: Ensemble diversity in evolving data streams. In: International Conference on Discovery Science. pp. 229–244. Springer (2016)
5. Cerqueira, V., Torgo, L., Pinto, F., Soares, C.: Arbitrated ensemble for time series forecasting. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 478–494. Springer (2017)
6. Cerqueira, V., Torgo, L., Pinto, F., Soares, C.: Arbitrage of forecasting experts. Machine Learning pp. 1–32 (2018)
7. Clemen, R.T., Winkler, R.L.: Combining economic forecasts. Journal of Business & Economic Statistics **4**(1), 39–46 (1986)
8. Cruz, R.M., Sabourin, R., Cavalcanti, G.D.: Dynamic classifier selection: Recent advances and perspectives. Information Fusion **41**, 195–216 (2018)
9. Domingos, P., Hulten, G.: Mining high-speed data streams. In: Kdd. vol. 2, p. 4 (2000)
10. Dutta, H.: Measuring diversity in regression ensembles. In: IICAI. vol. 9, p. 17p. Citeseer (2009)
11. Gama, J., Kosina, P.: Tracking recurring concepts with meta-learners. In: Portuguese Conference on Artificial Intelligence. pp. 423–434. Springer (2009)

12. Gama, J., Kosina, P.: Recurrent concepts in data streams classification. Knowledge and Information Systems **40**(3), 489–507 (2014)
13. Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., Bouchachia, A.: A survey on concept drift adaptation. ACM computing surveys (CSUR) **46**(4), 44 (2014)
14. Gomes, H.M., Barddal, J.P., Enembreck, F., Bifet, A.: A survey on ensemble learning for data stream classification. ACM Computing Surveys (CSUR) **50**(2), 23 (2017)
15. Hyndman, R.J., Koehler, A.B.: Another look at measures of forecast accuracy. International journal of forecasting **22**(4), 679–688 (2006)
16. Jacobs, R.A.: Methods for combining experts' probability assessments. Neural computation **7**(5), 867–888 (1995)
17. Kolter, J.Z., Maloof, M.A.: Using additive expert ensembles to cope with concept drift. In: Proceedings of the 22nd international conference on Machine learning. pp. 449–456. ACM (2005)
18. Krawczyk, B., Cano, A.: Online ensemble learning with abstaining classifiers for drifting and noisy data streams. Applied Soft Computing **68**, 677–692 (2018)
19. Krawczyk, B., Minku, L.L., Gama, J., Stefanowski, J., Woźniak, M.: Ensemble learning for data stream analysis: A survey. Information Fusion **37**, 132–156 (2017)
20. Kuncheva, L.I.: Combining pattern classifiers: methods and algorithms. John Wiley & Sons (2004)
21. Kuncheva, L.I., Whitaker, C.J.: Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine learning **51**(2), 181–207 (2003)
22. Oliveira, M.R., Torgo, L.: Ensembles for time series forecasting (2014)
23. Ortega, J., Koppel, M., Argamon, S.: Arbitrating among competing classifiers using learned referees. Knowledge and Information Systems **3**(4), 470–490 (2001)
24. Oza, N.C., Russell, S.: Experimental comparisons of online and batch versions of bagging and boosting. In: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 359–364. ACM (2001)
25. Rice, J.R.: The algorithm selection problem (1975)
26. van Rijn, J.N., Holmes, G., Pfahringer, B., Vanschoren, J.: Having a blast: Meta-learning and heterogeneous ensembles for data streams. In: Data Mining (ICDM), 2015 IEEE International Conference on. pp. 1003–1008. IEEE (2015)
27. Rossi, A.L.D., de Leon Ferreira, A.C.P., Soares, C., De Souza, B.F., et al.: Metastream: A meta-learning based method for periodic algorithm selection in time-changing data. Neurocomputing **127**, 52–64 (2014)
28. Sánchez, I.: Adaptive combination of forecasts with application to wind energy. International Journal of Forecasting **24**(4), 679–693 (2008)
29. Soares, S.G., Araújo, R.: A dynamic and on-line ensemble regression for changing environments. Expert Systems with Applications **42**(6), 2935–2948 (2015)
30. Soares, S.G., Araújo, R.: An on-line weighted ensemble of regressor models to handle concept drifts. Engineering Applications of Artificial Intelligence **37**, 392–406 (2015)
31. Takens, F.: Detecting strange attractors in turbulence. In: Dynamical systems and turbulence, Warwick 1980, pp. 366–381. Springer (1981)
32. Timmermann, A.: Forecast combinations. Handbook of economic forecasting **1**, 135–196 (2006)
33. Timmermann, A.: Elusive return predictability. International Journal of Forecasting **24**(1), 1–18 (2008)
34. Woźniak, M., Graña, M., Corchado, E.: A survey of multiple classifier systems as hybrid systems. Information Fusion **16**, 3–17 (2014)
35. Žliobaitė, I., Bifet, A., Pfahringer, B., Holmes, G.: Active learning with drifting streaming data. IEEE transactions on neural networks and learning systems **25**(1), 27–39 (2014)