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

Abstract

Mining temporal data streams is an important and challenging task due to the changing nature of data. Models have to adapt to concept drift and learn novel concepts when needed. Combining different algorithms is an effective solution to meet these needs and improve predictive accuracy. Dynamic ensemble selection is based on the assumption that different algorithms have different areas of expertise, and selecting, for each instance, the most accurate ones improves global performance. Existing methods are either tailored for batch learning or for classification tasks. We propose a new Arbitrated Dynamic Ensemble Selection method for time series forecasting where the predictive power of all models is monitored using meta-learning that is able to predict the competence of base-learners. Selection is based on abstaining policy, where poorly performing models are excluded from making a prediction. Our contribution is twofold: (i) We introduce two different approaches of abstaining: threshold-based and random-based selection and (ii) We conduct an experimental study to compare different methods on both real-world and synthetic data for time-series forecasting. To the best of our knowledge, this is the first time-series forecasting method based on dynamic ensemble selection for data streams.

1 Introduction

Mining temporal data streams is becoming an active research area due to its importance in real-time applications such as network behavior analysis and anomaly detection. Time-evolving data comprises different regimes, concept drift changes, and recurring concepts. Learning algorithms should be able to detect and adapt to these changes as soon as they occur to produce more reliable predictions, while avoiding a high rate of false alarms.

Ensemble methods or Multiple Classifier Systems (MCS) [Woźniak *et al.*, 2014], adopt a very attractive approach to tackle the dynamic nature of data streams [Krawczyk *et al.*, 2017]. Classifiers trained on more recent data can be added to the ensemble, whereas classifiers representing outdated data can be pruned. One of the most promising approaches of MCS is Dynamic Ensemble Selection (DES) where a subset of base classifiers, referred to as committee or Ensemble of Classifiers (EoC), is selected on the fly according to each test instance. Selected models are then weighted based on their estimated performance and combined to predict the new target value. The rationale for these techniques is that each base-model in the pool is expert in a different region of the data. This is motivated by the *No free Lunch* theorem [Wolpert, 2002] stating that no algorithm is better than any other one amongst all possible classes of problem. Ensemble methods have been widely studied for the classification problem [Gomes *et al.*, 2017] but less for forecasting and regression [Krawczyk *et al.*, 2017]. The use of meta-learning has been extensively investigated in the dynamic selection problem, as it was first described in [Rice, 1975]. The meta-problem uses different features describing the behavior of each base classifier M^j in the pool M in order to predict whether it is competent enough to contribute to the final output.

There are few works addressing time-series forecasting using dynamic ensemble selection [Krawczyk *et al.*, 2017]. The Arbitrated Dynamic Ensemble (ADE) proposed in [Cerqueira *et al.*, 2017] uses a set of base forecasters M trained off-line and a set of meta-learners Z where each meta-learner $Z^j \in Z$ is trained to predict the Mean Squared Error (MSE) $\hat{e}^j_{t+1}$ of its base counter-part M^j when trying to predict y_{t+1}. The $\alpha\%$ best base-models are then selected in a committee ${}^{\alpha}M$ and weighted according to their estimated errors using the $\mathrm{softmax}$ function.

The approach has led to very promising results yet limited when it comes to data streams. That is because base-models are never updated once trained which makes them unsuitable for predicting data instances belonging to novel concepts. Moreover, the $\alpha\%$ best base-models are selected based on their relative performance and regardless of the absolute value of the predicted error, probably leading the ensemble to perform very poorly in case of higher errors.

We propose an Arbitrated Dynamic Ensemble Selection framework for data streams that tackles the limitations of ADE proposed in [Cerqueira *et al.*, 2017]. We introduce an abstaining policy where only expert base-models can contribute to the final label, instead of selecting the $\alpha\%$ best ones. We present different selection approaches based on the predicted error of each base-model and the confidence level of

the meta-models.

The main contributions of this paper are the following ones:

1. We propose STREAMING-ADE, a new ensemble method that extends the ADE approach [Cerqueira *et al.*, 2017] which supports the traditional batch learning, to the streaming setting.
2. We propose different selection approaches based on abstaining where the competence level of base-models is considered in order to exclude uncertain ones for the committee.
3. We validate our approach on real and synthetic time-series data.

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.

The rest of the paper is organized as follows. Section 2 outlines the related work of dynamic ensemble selection and data stream mining. The STREAMING-ADEapproach is addressed in Section 3 where we formalize the meta-learning problem and different selection approaches. Experimental set-up along with the results are discussed in Section 4. Finally, Section 5 concludes this paper and tackles future improvements.

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 [Bifet *et al.*, 2010]. 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 real-time responsiveness. Finally, algorithms should be incremental and be able to detect and adapt to concept drift, and changes in the characteristics of the data [Gama *et al.*, 2014].

An ensemble learning method is a set of complementary and diverse individual models (components or base classifiers) 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 [Cruz *et al.*, 2018]. Ensemble methods are widely studied for data streams due to their good performance compared to single learners as reported in [Gomes *et al.*, 2017]. Authors enumerated a plethora of algorithms dedicated to classification tasks. Remarkably, very few works address regression and forecasting [Oza and Russell, 2001; Kolter and Maloof, 2005; Soares and Araújo, 2015a; Soares and Araújo, 2015b]. Ensembles are useful for mining data streams as they allow adaptation to changes in data, by adding new components trained on recent data, and removing components representing outdated data [Krawczyk and Cano, 2018]. However, changes are often recurrent, hence removing outdated classifiers will lead to forgetting useful historical knowledge that might be reused in the future [Gama and Kosina, 2009]. One of the most encouraging techniques of ensemble methods is dynamic selection, where the set of base classifiers is selected on the fly according to each new test instance [Cruz *et al.*, 2018]. 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 in order to determine whether it is competent enough to predict on a given test instance.

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

Very promising results can be reached by simply averaging predictions of the available base forecasters [Clemen and Winkler, 1986; Timmermann, 2006; Oliveira and Torgo, 2014]. The AEC (Adaptive Ensemble Combination) [Sánchez, 2008] is based on a windowing strategy that dynamically combines forecasters according to their past performance, including a forgetting factor to emphasize more recent data. [Timmermann, 2008] studied an adaptive combination of forecasters based on their recent coefficient of determination and simply averaging their outputs. ADE uses a more sophisticated and proactive mechanism [Cerqueira *et al.*, 2017] where only the most accurate base-learners are selected to contribute to the final output based on the predicted error of meta-learners.

Meta-learning allows to model the behavior of learning algorithms [Brazdil *et al.*, 2008]. A single classifier or an ensemble selection could be linked to a meta-problem [Rice, 1975] where the goal is to determine whether a base classifier M^j from the pool of base-models 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 [Gama and Kosina, 2009]. The base-classifier M^j learns to predict future values of the target $\hat{y}_{t+1}$ of the stream, whereas the meta-classifier Z^j learns the behavior of its base counter-part M^j and predicts its future errors $\hat{e}^j_{t+1}$.

2.2 Algorithm Selection on Data Streams

Several works have investigated the use of meta-learning for ensemble/classifier selection on data streams in order to address concept drift. The MetaStream framework proposes to periodically select the most adequate regression algorithm or set of regressors using a prequential evaluation method on a sliding window [Rossi *et al.*, 2014]. 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 regressors perform best to be used in the next window. The Online Performance Estimation proposed in [van Rijn *et al.*, 2015] estimates the predictive power of base-models on data streams. It is based on the assumption that recent examples are more relevant than older ones. Authors actually measure how ensemble components have performed on recent data and accordingly adjust their weights in the voting. The BLAST (**B**est **LAST**) framework is based on Online Performance Estimation and selects one of its base classifiers to be the only active model for every w test examples. The

BLAST Framework could be assimilated to a Dynamic Classifier Selection if $w = 1$.

Both the MetaStream [Rossi *et al.*, 2014] and BLAST [van Rijn *et al.*, 2015] methods are based on some user predefined window size w which is not trivial to determine. Conversely to previous methods, the ADE[Cerqueira *et al.*, 2017] proposes a more pro-active selection method. It is based on an arbitrated architecture discussed in [Ortega *et al.*, 2001] where base classifiers are trained off-line 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 classifiers are then selected and weighted accordingly to their estimated performance in order to combine their outputs and predict the future value y_{t+1}. The use of meta-learning for model selection on data streams classification and recurrent concepts was addressed in [Gama and Kosina, 2014]. 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 selection was introduced in [Krawczyk and Cano, 2018] based on an abstaining policy of base-models, where the less confident classifiers are allowed to abstain from contributing to final decision according to a dynamic threshold that is updated at every instance. The confidence level is estimated for each incoming instance. Forcing uncertain classifiers to abstain from making predictions is useful for noisy data streams.

We present in Section 3 our STREAMING-ADE framework that performs Arbitrated Dynamic Ensemble Selection based on abstaining using meta-learning. We detail two different approaches and highlight our contributions to the data stream context. Our selection methods are tailored to changing data and are able to detect and react to concept drift.

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. Every instance y_t is described in a vector $\vec{v_t} = \langle y_{t-(k-1)}, y_{t-(k-2)}, \ldots, y_t \rangle$ with K past lags known using time embedding. Our proposed method STREAMING-ADE uses meta-learning to perform dynamic ensemble selection. 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 for each incoming instance where we explore two different approaches: threshold-based and random-based (iii) selection according to predicted errors of each meta-model for its base counter-part (iv) prediction of y_{t+1} using predicted values $\hat{y}_{t+1}$ of each base-model $M^{j \in {}' M}$ in the selected committee and weighting them using the predicted errors $\hat{e}^j_{t+1}$ of each meta-model $Z^j \in Z$.

3.1 Meta-Learning Ensemble and Error Prediction

The meta-level is composed of an ensemble Z of meta-models $Z = \{Z^1, Z^2, \ldots, Z^m\}$ where each component Z^j is in charge of predicting future errors $\hat{e}^j_{t+1}$ of its base counterpart M^j. [Cerqueira *et al.*, 2017] have used MSE to quantify base-models errors. However, it is not trivial, when using threshold-based methods, to set the threshold value θ while using MSE as a measure of error. This latter is not relative and thus closely linked to data, which requires prior expert knowledge about it. We use SMAPE for Symmetric Mean Absolute Percentage Error [Hyndman and Koehler, 2006] as a relative measure to overcome this limitation. According its initial definition [Hyndman and Koehler, 2006], the SMAPE can be formalized as shown in Equation 1:

$$\text{SMAPE} = \begin{cases} 0, & \text{if } y_t, \hat{y}_t = 0. \\ \frac{100}{N} * \sum_{i=t}^{N} \frac{2*|\hat{y}_t - y_t|}{|y_i| + |\hat{y}_t|} & \text{otherwise.} \end{cases} \quad (1)$$

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

3.2 Meta-Model Confidence

Base-models selection and weighting is based on meta-models predicted errors. Performance of meta-model affects the relevance of the selection, thus we introduce a confidence measure to quantify to what extent the meta-model Z^j was accurate in predicting the errors of its base counter-part M^j on past instances. The confidence $c^j_t \in]0, 1]$ of a meta-model Z^j at time t is defined as

$$c^j_t = \exp(-\text{MSE}^j_t) \quad (2)$$

$$\text{MSE}^j_t = \frac{1}{N} * ((\hat{e}^j_t - e^j_t)^2 + \lambda * \text{SSE}^j_{t-1}) \quad (3)$$

with N the number of instances seen, $\lambda \in]0, 1]$ a fading factor to emphasize on more recent instances ad SSE the Sum of Squared Errors.

3.3 Base-Model Selection

ADE is based on a fixed selection mechanism where the $\alpha\%$ base-models with the lowest predicted error $\hat{e}^j_{t+1}$ are part of the committee. This selection does not take into account error values and may be limited when all meta-models predict high values of error. This can be translated to a concept-drift where all base-models fail to predict $\hat{y}_{t+1}$. We propose to tackle this shortcoming by introducing an abstaining policy that takes into consideration predicted error values, 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 M^j is considered expert and thus selected in the committee to predict at time $t + 1$ if and only if $\hat{e}^j_t < \theta$. Algorithm 1 shows how the committee ${}^{\theta}M$ is selected for each incoming test instance.

The threshold θ is dynamic and self-adaptive as discussed by [Krawczyk and Cano, 2018]. If ensemble prediction $\hat{y}_t$ is correct, this means that we have selected competent classifiers and we may increase the threshold θ in order to seek for similarly good base classifiers. If the prediction is wrong,

Algorithm 1 Threshold-based Dynamic Ensemble Selection

Input: New test instance with embedding from S, θ
Output: ${}^{\theta}M \in M$ of non-abstaining base-models

1: Let ${}^{\theta}M = \emptyset$
2: **for** $M^j \in M$ **do**
3: $\quad \hat{e}^j_{t+1} \leftarrow$ get prediction from Z^j
4: $\quad$ **if** $\hat{e}^j_{t+1} \leq \theta$ **then** add M^j to ${}^{\theta}M$
5: **end for**
6: **return** : ${}^{\theta}M$

we need to decrease θ in order to exclude poorly performing base-learners form the committee. We explore two different update strategies, **iterative** $\theta \leftarrow \pm s$ and **multiplicative** $\theta \leftarrow \theta(1 \pm s)$, where $s \in [0, 1]$ is previously decided by the user.

Random-Based Selection

We study a selection approach based on a random Bernoulli distribution law, whose parameters are related to the predicted error of base-models and confidence level of their meta counter-parts. We model the selection of M^j to predict y_{t+1} as a Bernoulli trial of parameter $p_e = 1 - p$ where $p = \hat{e}^j_{t+1}$. This means that a base-model is selected with probability $q_e = 1 - p_e$. The smaller the error is the greater is the probability of M^j to be selected. However, concept drift may happen anywhere in the stream [Žliobaitė *et al.*, 2014]. Hence, it might be interesting, from time to time, to select other base-models with high predicted errors and inversely prune the ones with low errors. This way, we select the most confident base-learners but occasionally select less confident ones to have a greater chance not to miss concept drift. Algorithm 2 describes the random-based selection process using predicted errors only.

We introduce a more restrictive randomised selection using the confidence level of meta-learners. We model the relevance of a meta-model Z^j at time t as a Bernoulli trial of parameter $p_c = c^j_t$. The greater the confidence is, the more relevant is the meta-model. Moreover, this approach allows to detect and react to concept drift. When a change occurs in the stream, a meta-model Z^j will most likely fail in predicting the error e^j_{t+1}. Consequently, the error MSE^j_t incurred by Z^j increases and, as a result, the confidence c^j_t drops. A lower confidence can be translated to M^j being eliminated from the committee. A base-model M^j is selected if and only if itself and its meta-model counter-part are selected through the two random distributions. Algorithm 2 is changed starting from line 4 in order to consider another random variables r_c related to the confidence level of meta-models as shown in Algorithm 3.

3.4 STREAMING-ADE

Based on the assumption that each base-learner is expert in some subspace of the data, we will only select the most competent base-learners at each test instance according to the predicted errors from meta-models. In contrast to the method proposed in [Cerqueira *et al.*, 2017], our selection model uses an abstaining policy where only the most confident learners are allowed to contribute to the final output. High predicted error value $\hat{e}^j_{t+1}$ may reflect a concept drift in the data which renders the base-model M^j more likely to fail in predicting y_{t+1}. Moreover, selection on predicted errors highly depends on the competence level of meta-models. Therefore, we consider a factor c^j_t to quantify the extent to which a meta-model Z^j was accurate in predicting errors of its base-model counter-part in the past.

We select the set of base experts ${}'M$ at line 3 of Algorithm 4 according to one of the methods explained in section 3.3. We compute a weight w^j_{t+1} for each $M^j \in {}' M$ using meta-predictions obtained in line 5 according to the $\mathrm{softmax}$ function. Base-experts outputs $\hat{y}^j_{t+1}$ are then combined to get final output $\hat{y}_{t+1}$ corresponding to line 8. 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 θ 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 which allows.

Algorithm 2 Error Random-Based Dynamic Selection

Input: New test instance with embedding from S
Output: ${}^{r}M \in M$ of selected base-models

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

Algorithm 3 Confidence Random-Based Dynamic Selection

3: $\hat{e}^j_{t+1} \leftarrow$ get prediction from Z^j
4: $\hat{c}^j_t \leftarrow$ get confidence of Z^j
5: generate $r_e \sim Bernoulli(1 - \hat{e}^j_{t+1})$
6: generate $r_c \sim Bernoulli(\hat{c}^j_t)$
7: **if** $r_e = 1$ **and** $r_c = 1$ **then** add M^j to ${}^{r}M$

4 Experimental Study

In this section, we present an experimental study to evaluate our proposed dynamic ensemble selection approaches. We empirically analyze their performance against baselines that combine all base-models of the pool. We discuss experimental set-up of the ensemble and discuss different data sets used in the evaluation. Experiments were designed to evaluate the efficiency of selection approaches based on abstaining.

4.1 Ensemble Set-up

We compare seven different ensemble methods where the **NAIVE** approach that simply averages all base-models outputs stands for the baseline. The **WEIGHTED** ensemble uses a $\mathrm{softmax}$ to weight all base-models outputs according to meta-models' predicted errors. Different selection methods discussed in Section 3.3 were tested: Threshold and

Algorithm 4 Streaming Arbitrated Dynamic Ensemble

Input: Infinite stream $S = \{y_1, y_2, \ldots, y_t, \ldots\}$,
$M = \{M^1, M^2, \ldots, M^m\}$,
$Z = \{Z^1, Z^2, \ldots, Z^m\}$,
Parameter: Selection strategy
Threshold θ, 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 $'M$ be the set of selected base-models using one of the three methods in Algorithm 1, 2 or 3
4: Let $'Z$ be the set of $'M$ meta-models counter-parts
5: Get meta-predictions $\hat{e}^j_{t+1}$ from $Z^j \in {}'Z$
6: Compute weights $w^j_{t+1} = \frac{\exp(-\hat{e}^j_{t+1})}{\sum_{Z^j \in {}'Z} exp(-\hat{e}^j_{t+1})}$
7: Get predictions $\hat{y}^j_{t+1}$ from every $M^j \in {}'M$
8: Compute final prediction $\hat{y}_{t+1} = \sum_{M^j \in {}'M} \hat{y}^j_{t+1} * w^j_{t+1}$
9: Output $\hat{y}_{t+1}$
10: Obtain true value of y_{t+1}
11: Update all base-models $M^j \in M$
12: Update all meta-models $Z^j \in Z$
13: Update threshold θ // if threshold-based selection
14: **end while**

Random based. We study three threshold-based approaches that differ in the update strategy of the threshold θ where: (i) **AB-TH-SUM** uses an additive strategy, (ii) **AB-TH-PR** implements the multiplicative strategy, (iii) **AB-TH-ST** never updates θ. We finally compare two random-based selection approaches as described in Section 3.3 where **AB-PROB** implements a random selection using base-models predicted errors only, whereas **AB-PROB-CONF** considers both base-models errors and meta-models confidence level in the selection process.

For the experiments, we have used four online classifiers as base-learners : Adaptive Hoeffding Tree [Bifet and Gavaldà, 2009], Hoeffding Tree [Domingos and Hulten, 2000], K-Nearest Neighbors (KNN) and a weighted version of the KNN, where each neighbor is weighted inversely to its distance to the test instance. Our implementation is based on scikit-multiflow[1], an open-source package written in python. We have used different parameter settings for each base-learner in order to ensure diversity within the base-ensemble. We have used an Adaptive Hoeffding Tree as meta-model with default parameters setting. The ensemble size is set to $m = 30$, the threshold $\theta = 20\%$, the update step $s = 0.01$ and the fading factor $\lambda = 0.995$. We set the value k of time lags embedding to $k = 7$. All methods were evaluated using both prequential MSE and MAE metrics using the test-then-train scenario where each instance is first used to evaluate and then to update all models.

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

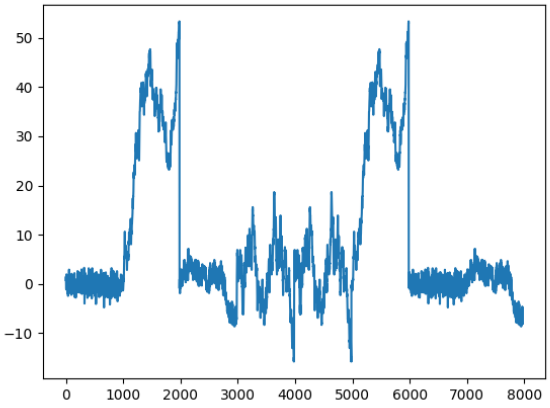

Figure 1: Synthetic data with concept drift and recurrent patterns

4.2 Data sets

We used different time-series represented in data streams to evaluate the performance of the set of online ensembles and dynamic selection approaches presented in Section 4.1. We have selected a diverse set of data streams with varying characteristics,including real datasets and synthetic generators. We use 46 real-world time-series [2] proposed in [Cerqueira *et al.*, 2017]. We validate our work on synthetic data that model concept drift and recurrent patterns in data streams. We created 4 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 some white Gaussian noise along the time series. Figure 1 shows a time-series generated with 4 different concepts each of length 2000 instances . Theses concepts are then repeated in order to model recurring patterns. All synthetic data can be reproduced using a python package TimeSynth[3] to implement this approach.

4.3 Results

We report in this section results of our experimental study described in 4.1. We analyzed the performance of all methods using the Mean Absolute Error (MAE) that is the absolute average difference between the predicted value $\hat{y}_t$ and the real value y_t. Figure 2 presents the average rank and respective deviation of each ensemble method across the 50 datasets. Comparing the different strategies of combining and selecting base-models, our STREAMING-ADE method based on random selection shows improvements that prove the contribution of base-models selection in the performance of the ensemble against the **NAIVE** and **WEIGHTED** approach. These results suggest that our proposed **AB-PROB** and **AB-PROB-CONF** are tailored to changing temporal data and concept drift where both predicted error and the suggested

[2]Available at https://github.com/vcerqueira/forecasting_experiments

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

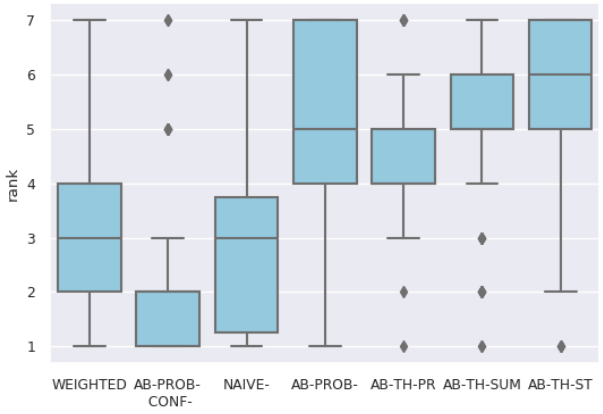

Figure 2: Average ranking in terms of MAE over all 50 data-sets

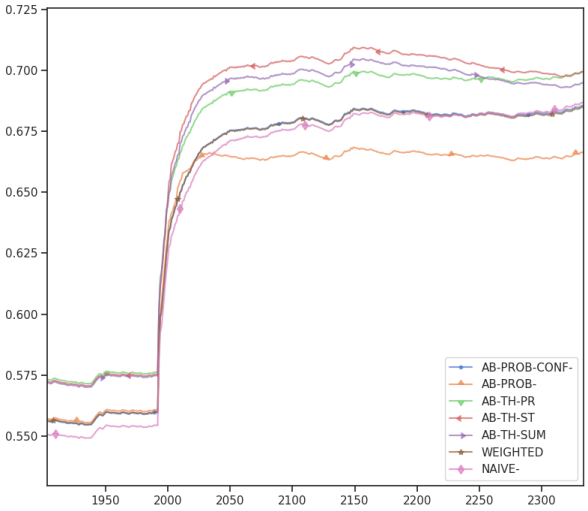

Figure 3: Concept drift adaptation of all methods when concept drift occurs in Auto-Regressive synthetic data

confidence level introduced in 2 contribute greatly in the improvement of the ensemble performance.

Figures 3 and 4 emphasize the behavior of each method when facing a concept drift in the stream where Figure 3 reflects the behavior when a change first occurs whereas Figure 4 highlights methods behavior when concept drift reappears . Evolution of MAE asses the ability of random-based approaches **AB-PROB** and **AB-PROB-CONF** to cope with changes. Randomness allows to look from time to time to other base-models that were predicted to fail. This may be efficient in detecting a concept drift and mostly likely help in adaptation. Experimental results demonstrate that selecting the most confident base-models is especially effective compared to combing all base-models of the pool. The threshold-based selection methods: **AB-TH-ST**, **AB-TH-PR** and **AB-TH-SUM** are relatively good overall data-sets and comparable to each other but significantly worse than all other ensemble methods.

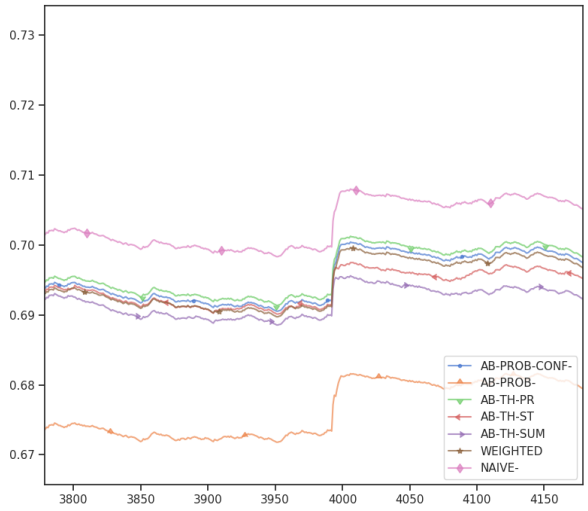

Figure 4: Concept drift adaptation of all methods when concept drift reappears in Auto-Regressive synthetic data

5 Conclusions

We have presented in this paper STREAMING-ADE, an Arbitrated Dynamic Ensemble Selection framework for data streams that uses meta-learning to select a committee of models on the fly according to each test instance. We have proposed different selection approaches : threshold-based and random-based where the less confident base-models are forced to abstain. The threshold-based strategy compares the predicted error of each base-model to a user defined threshold to determine whether it is competent enough to contribute to the final output. The random-based selection approach models the selection process as a Bernoulli trial where the probability varies directly with the meta-models confidence and inversely with the predicted errors of base-models.

Experimental results utilizing both real-world and synthetic time-series shows that STREAMING-ADE abstaining policy based on random selection with regards to meta-models confidence and base-models predicted errors are specially effective in predicting future values of times-series when dealing with concept drifts.

Future works on this topic will involve :

- Consideration of diversity measures in the selection process. In this work, diversity is assumed with the use of different bias (different base-models) and parameters. Nonetheless, STREAMING-ADE does not directly quantify the diversity amongst the pool of base-learners.
- Ensemble size monitoring where new models can be added and outdated ones pruned.
- Consideration of computation performance measures related to the data stream setting such as compute time and memory usage.

References

[Bifet and Gavaldà, 2009] Albert Bifet and Ricard Gavaldà. Adaptive learning from evolving data streams. In *International Symposium on Intelligent Data Analysis*, pages 249–260. Springer, 2009.

[Bifet *et al.*, 2010] Albert Bifet, Geoff Holmes, Richard Kirkby, and Bernhard Pfahringer. Moa: Massive online analysis. *Journal of Machine Learning Research*, 11(May):1601–1604, 2010.

[Brazdil *et al.*, 2008] Pavel Brazdil, Christophe Giraud Carrier, Carlos Soares, and Ricardo Vilalta. *Metalearning: Applications to data mining*. Springer Science & Business Media, 2008.

[Cerqueira *et al.*, 2017] Vítor Cerqueira, Luís Torgo, Fábio Pinto, and Carlos Soares. Arbitrated ensemble for time series forecasting. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 478–494. Springer, 2017.

[Clemen and Winkler, 1986] Robert T Clemen and Robert L Winkler. Combining economic forecasts. *Journal of Business & Economic Statistics*, 4(1):39–46, 1986.

[Cruz *et al.*, 2018] Rafael MO Cruz, Robert Sabourin, and George DC Cavalcanti. Dynamic classifier selection: Recent advances and perspectives. *Information Fusion*, 41:195–216, 2018.

[Domingos and Hulten, 2000] Pedro Domingos and Geoff Hulten. Mining high-speed data streams. In *Kdd*, volume 2, page 4, 2000.

[Gama and Kosina, 2009] João Gama and Petr Kosina. Tracking recurring concepts with meta-learners. In *Portuguese Conference on Artificial Intelligence*, pages 423–434. Springer, 2009.

[Gama and Kosina, 2014] João Gama and Petr Kosina. Recurrent concepts in data streams classification. *Knowledge and Information Systems*, 40(3):489–507, 2014.

[Gama *et al.*, 2014] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, 46(4):44, 2014.

[Gomes *et al.*, 2017] Heitor Murilo Gomes, Jean Paul Barddal, Fabrício Enembreck, and Albert Bifet. A survey on ensemble learning for data stream classification. *ACM Computing Surveys (CSUR)*, 50(2):23, 2017.

[Hyndman and Koehler, 2006] Rob J Hyndman and Anne B Koehler. Another look at measures of forecast accuracy. *International journal of forecasting*, 22(4):679–688, 2006.

[Kolter and Maloof, 2005] Jeremy Z Kolter and Marcus A Maloof. Using additive expert ensembles to cope with concept drift. In *Proceedings of the 22nd international conference on Machine learning*, pages 449–456. ACM, 2005.

[Krawczyk and Cano, 2018] Bartosz Krawczyk and Alberto Cano. Online ensemble learning with abstaining classifiers for drifting and noisy data streams. *Applied Soft Computing*, 68:677–692, 2018.

[Krawczyk *et al.*, 2017] Bartosz Krawczyk, Leandro L Minku, João Gama, Jerzy Stefanowski, and Michał Woźniak. Ensemble learning for data stream analysis: A survey. *Information Fusion*, 37:132–156, 2017.

[Oliveira and Torgo, 2014] Mariana Rafaela Oliveira and Luis Torgo. Ensembles for time series forecasting. 2014.

[Ortega *et al.*, 2001] Julio Ortega, Moshe Koppel, and Shlomo Argamon. Arbitrating among competing classifiers using learned referees. *Knowledge and Information Systems*, 3(4):470–490, 2001.

[Oza and Russell, 2001] Nikunj C Oza and Stuart Russell. 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*, pages 359–364. ACM, 2001.

[Rice, 1975] John R Rice. The algorithm selection problem. 1975.

[Rossi *et al.*, 2014] André Luis Debiaso Rossi, André Carlos Ponce de Leon Ferreira, Carlos Soares, Bruno Feres De Souza, et al. Metastream: A meta-learning based method for periodic algorithm selection in time-changing data. *Neurocomputing*, 127:52–64, 2014.

[Sánchez, 2008] Ismael Sánchez. Adaptive combination of forecasts with application to wind energy. *International Journal of Forecasting*, 24(4):679–693, 2008.

[Soares and Araújo, 2015a] Symone G Soares and Rui Araújo. A dynamic and on-line ensemble regression for changing environments. *Expert Systems with Applications*, 42(6):2935–2948, 2015.

[Soares and Araújo, 2015b] Symone Gomes Soares and Rui Araújo. An on-line weighted ensemble of regressor models to handle concept drifts. *Engineering Applications of Artificial Intelligence*, 37:392–406, 2015.

[Timmermann, 2006] Allan Timmermann. Forecast combinations. *Handbook of economic forecasting*, 1:135–196, 2006.

[Timmermann, 2008] Allan Timmermann. Elusive return predictability. *International Journal of Forecasting*, 24(1):1–18, 2008.

[van Rijn *et al.*, 2015] Jan N van Rijn, Geoffrey Holmes, Bernhard Pfahringer, and Joaquin Vanschoren. Having a blast: Meta-learning and heterogeneous ensembles for data streams. In *Data Mining (ICDM), 2015 IEEE International Conference on*, pages 1003–1008. IEEE, 2015.

[Wolpert, 2002] David H Wolpert. The supervised learning no-free-lunch theorems. In *Soft computing and industry*, pages 25–42. Springer, 2002.

[Woźniak *et al.*, 2014] Michał Woźniak, Manuel Graña, and Emilio Corchado. A survey of multiple classifier systems as hybrid systems. *Information Fusion*, 16:3–17, 2014.

[Žliobaitė *et al.*, 2014] Indrė Žliobaitė, Albert Bifet, Bernhard Pfahringer, and Geoffrey Holmes. Active learning with drifting streaming data. *IEEE transactions on neural networks and learning systems*, 25(1):27–39, 2014.