KL-ADWIN: enhanced concept drift detection over multiple time windows.

Jakob Jelenčič, Jože M. Rožanec
Jožef Stefan International Postgraduate School
Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia
{jakob.jelencic, joze.rozanec}@ijs.si

Dunja Mladenić
Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia
dunja.mladenic@ijs.si

Abstract. *The main contribution of the paper is method that combines adaptive windowing and the drift detection method for detecting concept drift and therefore determine when a machine learning model should be retrained. We evaluated the method on four real-world datasets with concept drift. Our results show that the proposed method improves the models' performance (in three of four datasets) on unseen data when compared to the baseline.*

Keywords. Concept Drift, Kullback–Leibler Divergence, Drift Detection Method

1 Introduction

Unforseeable changes that occur in the underlying distribution of data over time is known as concept drift. Concept drift affects machine learning models, degrading their performance. While this phenomena can be observed in many use cases, it frequently affects time-series related models. Learning and running the best model in a given point in time turns out to be a non-optimal decision when the target distributions shift. Constantly re-learning a new model to address the concept shift can be an unfeasible strategy when deploying models into production, given the re-learning cost and time needed to do so.

When drastic changes take place in the data distribution, some of the historic data may no longer provide adequate learning ground for the algorithm, most probably increasing the cost of learning and hurting performance. In this paper, we propose combining two concept drift methods (Drift Detection Method (DDM) and Kullback–Leibler (KL) divergence) to determine the optimal time to relearn a machine learning model based on monitoring their performance across different time windows. The size of the re-learning window is considered a parameter.

We evaluate the method on four real datasets that present concept drift. As a baseline, we built a model that was able to see all historical data and was retrained for each step. In our research, we aimed to develop a concept drift method that would allow us to outperform the baseline or reduce the amount of retrains required to match the baselines' performance. Our results show that the proposed method improves the machine learning models' performance on unseen data.

The remainder of the paper is organised as follows. Section 2 is a short overview of related work and current state of the art methods. Section 3 describes the data we use, Section 4 reviews existing concept drift methods and we describe the proposed methodology. In Section 5, we present the results we obtained. Finally, in Section 6, we discuss the results and provide directions for future work.

2 Related Work

2.1 Concept drift

Gama et al. (2014) defines concept drift as a change in relation between in supervised learning scenario of input data and the target variable over time. Souza et al. (2020) define concept drift as a data stream where changes in distribution happened, governed by the dynamics of real-world problems and application domains that evolve. In the context of machine learning, these changes in data distribution are named concept drifts.

In practice dealing with concept drift is a wide researched problem Gama et al. (2014). Usually once you detect that a real-world data stream on which you relay to perform a activity has changed, major costs could occur. Imagine a fine tuned trading algorithm that predict change in underlying price, equity for example. Systems like that are often very sensitive to small changes in relations between input data and target variable, meaning that a small change can trick the algorithm to make a series of bad decisions, resulting in potentially enormous costs.

There exist a couple of methods how to detect and handle the concept drift problem. One way is to measure the difference in the distribution of target variable in two separate point of time Tsymbal (2004). In some case one can detect a change in distribution before there is a significant effect in the target metric, such as accuracy. A good example would be a Kullback-Leibler divergence. Another possible approach is to measure an error rate in target metric, and once there is a sufficient change there is a good chance that a concept drift occurred. Example of such method would be drift detection method Baena-Garcıa et al. (2006).

2.2 Kullback-Leibler divergence

The Kullback-Leibler (Van Erven and Harremos (2014)) divergence is a method of measuring a statistical distance between two different distributions. It is a distance, not a metric, and is often used to detect concept drift (Goldenberg and Webb (2019)). Let Q and P be two different probability distributions defined on the same probability space X, then the Kullback-Leibler divergence is defined as:

$$D_{KL}(P||Q) = \sum_{x\in X} P(x)\log\frac{P(x)}{Q(x)}. \tag{1}$$

2.3 Drift detection method

The drift detection method (Baena-Garcıa et al. (2006)) is a data stream error monitoring method where one knows the realized truth in a relatively short feedback loop. DDM compares the current error and its variance with the minimum error and its variance in a given time window (Gama et al. (2004)). DDM is divided into two zones: Alarm zone (2) and Detection zone (3).

$$p_{min} + 2 \cdot s_{min} \leq p_i + s_i \tag{2}$$

$$p_{min} + 3 \cdot s_{min} \leq p_i + s_i \tag{3}$$

where p_i is the error rate at time i, s_i is the standard deviation at time i, and p_{min}, s_{min} are the minimum values in the selected window. One of the disadvantages of DDM is that it provides only three types of signals, either no displacement, alarm or displacement detection.

3 Datasets

Our research focuses on real-world use cases that are prone to concept drift. We address four datasets (Amazon stock prices [1], credit card fraud detection [2], National Oceanic and Atmospheric Administration (NOAA) (NOAA) weather measurements Souza et al. (2020), and Electricity Market prices Souza et al. (2020)). All of them provide time series describing different phenomena, and a Boolean target value. We therefore address forecasting the target value as a binary classification problem. The last two datasets (NOAA and Electricity) were considered given they are part of a benchmark for streaming algorithms (Souza et al. (2020)). When selecting a subset of datasets from the benchmark, we opted for the two above-mentioned ones, given they were not strongly imbalanced.

3.1 Equity Dataset: Amazon Stock Price

Stocks are assumed to follow some form of stochastic process, either the Black-Scholes process (Merton (1976)) or more complex processes with an unknown formulation. Such processes evolve over time, and the distribution can change dramatically with extreme events (known as Black Swan events (Taylor and Williams (2009)) that alter the macroeconomic environment. We collected freely available daily stock prices from 2007 and derived an outcome variable as the change in the simple moving average over the last ten days.

3.2 Credit Card Fraud Detection Dataset

The credit card fraud dataset contains transactions made by European cardholders with credit cards in September 2013. The dataset is highly imbalanced and can be found on Kaggle. For the purpose of this research, we under-sampled the original dataset, to decrease the imbalance with respect to the minority class and thus enhance the learning of our machine learning model for the experiments we performed. We randomly under-sampled the majority class until the percentage of minority class was around ten percent.

3.3 NOAA Weather Measurements Dataset

Souza et al. (2020) published the NOAA dataset in 2020. The dataset consists of weather measurements collected over 50 years at Bellevue, Nebraska by the National Oceanic and Atmospheric Administration. This dataset contains eight features: temperature, dew point, sea-level pressure, visibility, average wind speed, max sustained wind-speed, minimum temperature, and maximum temperature. The learning task is to determine whether it will rain or not. The dataset contains 18,159 daily readings of which 5,698 are rain and the remaining 12,461 are no rain.

3.4 Electricity Market Prices Dataset

Electricity market prices dataset probably is one of the most used for the tasks of stream classification and drift detection. The data are from the Australian New South Wales Electricity Market. Prices are affected by the demand and supply. The learning task is to predict a rise or a fall in electricity prices, given recent consumption and prices in the same and neighboring regions. The dataset contains 45,312 instances, eight attributes, and

[1] The data was retrieved from the following URL (last access May 17th 2022): `https://finance.yahoo.com/`

[2] The dataset can be found at the following URL (last access May 17th 2022): `https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud`

two class labels. The dataset was published in Souza et al. (2020).

4 Methodology

4.1 Data preprocessing

All three datasets-credit fraud, NOAA, and electricity-were partitioned to match the size of the Amazon stock dataset to speed the calculation. With respect to features, the two datasets from real-world streaming dataset collection and credit card dataset were taken as they are. For the Amazon stock price dataset we have taken the most common indicators (various crossovers from moving averages, Bollinger bands, volume analysis etc.) from technical analysis Edwards et al. (2018) and transformed the one publicly available on Yahoo Finance. All four dataset were normalized using standard Z normalization. Since nothing especially complex was done and preprocessing adds nothing to the actual contribution of this work we will not go into further details.

4.2 KL-ADWIN: enhacing concept drift detection

Concept deviations can manifest themselves with varying speed, severity, and patterns (Souza et al. (2020)). There is no universal method for detection that would work perfectly in every scenario. A practical way to detect shifts is to monitor the distribution with sliding windows of different sizes or to monitor errors over time and their variance (Tsymbal (2004)). Failure to detect a shift can have a high cost, depending on the target area. For example, if a forecasting model used for automated trading in the stock markets fails to detect a shift in the distribution and starts making incorrect trading decisions. On the other hand, constant relearning may not be feasible given the frequency of new data or the hardware capacity required for complex models.

In our research, we propose KL-ADWIN, a method similar to the ADaptive WINdowing (ADWIN) algorithm (Bifet and Gavalda (2009)), where we monitor different time windows to detect possible distribution shifts in the underlying data. In conjunction, we use a slightly modified version of DDM, defined as:

$$mDDM(t) = p_i + s_i - (p_{min} + 2 \cdot s_{min}). \qquad (4)$$

In this way, we obtain a real number for each moment in time. The difference between classic DDM is that by doing so we can assign a weight to it and create a continuous indicator. Ideally mDDM is a function that would output something close to 0 and when the output is higher than 0 that would indicate alarm zone or possible change.

Similarly, we calculate KL divergence for two time windows that follow each other without overlap. For better understanding, imagine test folds $[1, 2, 3, 4, 5]$. For windows of size two and one, we would first compare the distributions from folds one and two with the distribution in three. Then the distributions from folds two and three with the one in fold four, and so on. To make the signal comparable to that of $mDDM$, we divide by the mean and subtract one. Let m and k be the indices of the windows, then we define Kl criteria as:

$$mKL(t) = \frac{KL(X_m, X_k)}{\frac{1}{t}\sum_{i=1}^{t} mKL(i)}. \qquad (5)$$

To illustrate the idea of KL divergence we have plotted distribution of target variable of Amazon stock prices dataset after concept drift. Both distributions can be seen on Figure 1. With the help of KL divergence one can measure how much is the difference between two. The bigger the difference the higher the integral will be. Now since we cannot understand the meaning of a single number, we have divided the current one with the historic average. To elaborate on example, if the difference plotted on Figure 1 would happen all the time, then the (5) will not sound the alarm since the similar change happened in historic data and the model should be able to handle it. If not, then the (2) will be higher than one, and possibly signaled that relearn is needed.

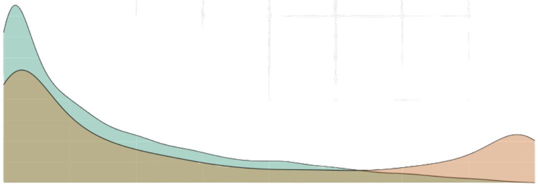

Figure 1: Example of distribution shift from Amazon stock prices dataset.

Finally, we can define the proposed method for determining the optimal time for relearning the model. Let w_1 and w_2 be the weights that sum to 1; $w_1 + w_2 = 1$. Let be the th scalar threshold to be optimized. Then we can define the criteria as

$$\textit{KL-ADWIN} = w_1 \cdot mDDM(t) + w_2 \cdot (mKL(t) - 1) > th. \qquad (6)$$

4.3 Relearning window

In machine learning, it is widely assumed that more data usually means a better model. Nevertheless, more data does not always mean more information (a phenomena exploited by e.g., active learning (Kumar and Gupta (2020)), and more recently, the datacentric artificial intelligence approaches (Marcu and Prügel-Bennett (2021); Paiva et al. (2021))). Furthermore, if the distribution of the data changes over time (concept drift), past data may no longer provide cues for forecasting current target values. For example, imagine a world where stock prices only go up and the market suddenly reverses. Adding additional data from the historical distribution that is no longer accurate could only hurt performance. While determining the optimal relearning window remains a subject of future work, for this research work we have fixed the time window on 25% of the existing dataset.

5 Results

In this section we present the results we obtained with the proposed method. We evaluated and analyzed both $mDDM$ and mKL over time for each test dataset to determine the optimal relearning point. We used the random forest model with 200 trees due to its robustness in terms of fine-tuning.

We used the following parameters: $w_1 = 0.5$, $w_2 = 0.5$ and $th = 0$. Relearning window was fixed at $p = 0.25$ (25% of total dataset). For further analysis, we also track the two combinations where one of w_i is set to zero. In Figures 2 through 5, the green dots represent mKL and the red dots represent $mDDM$. Relearning occurred only where both dots appeared together. Where single dots appear, either red or green, this indicates that one of the methods detected a shift (w_i of the other was set to zero), but the other was not strong enough to warrant relearning.

We evaluated the method on all four datasets and compared it to the model that would be relearned for each new dataset (following a streaming fashion, despite being a batch machine learning model). It is important to emphasize that the proposed method is much more economical, since the baseline requires constant retraining of the model. When the proposed method signals relearning, we only take the last 25% of historical data, making the retrainig cheaper and faster than the baseline, which requires all of the historical data. The ROC AUCs of the streaming models are shown as a dashed line, while the proposed method has a solid line.

5.1 Equity dataset: Amazon Stock Price

In the stock dataset, we forecast the change in the moving average of the Amazon stock price (trend change). The results are shown in Figure 2. We can see that the proposed approach is more stable than the streaming approach with only three learnings. Also, we can see a green dot indicating a possible shift just before the single peak in the ROC AUC of the streaming model, which can lead to another relearn with a better parameter choice and possibly outperform the ROC AUC of the streaming model in this part of the test set as well.

5.2 Credit Card Fraud Detection dataset

Figure 3 shows the results of the credit card fraud detection dataset. At first glance, it is clear that the results are not as good as the previous dataset. Here, only two new learnings were required, and twice the signal indicated possible shifts but did not exceed the new learning threshold.

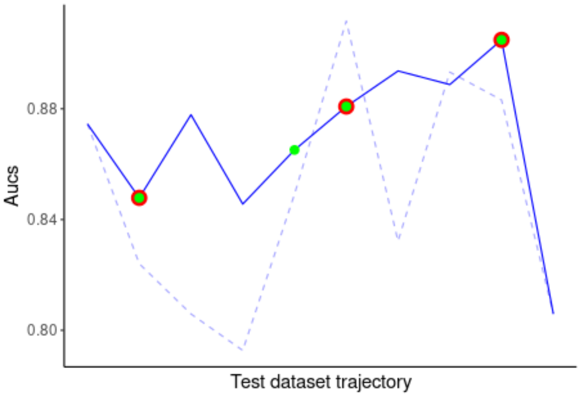

Figure 2: Results of proposed method on equity dataset.

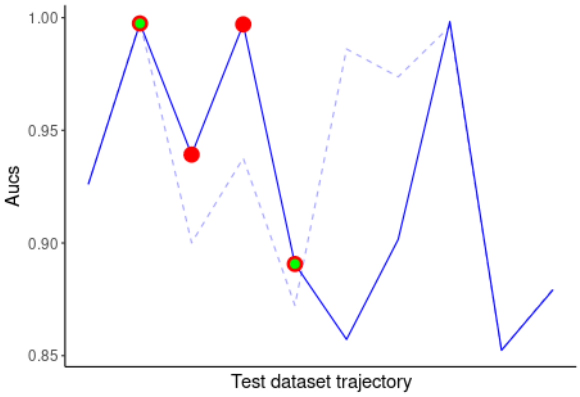

Figure 3: Results of proposed method on credit card fraud detection dataset.

5.3 NOAA Weather Measurements Dataset

Figure 4 shows the results for the NOAA dataset. The NOAA and Equity datasets best fit the type of problems that this method is designed to solve. We can see that the proposed method performs significantly better with only 2 learning operations.

5.4 Electricity market prices dataset

Figure 5 shows the results for the current dataset. This is the only dataset where we can say that the streaming model is better. The method detected only a single shift point, briefly outperformed the ROC AUC of the streaming model, and then performed worse all the time. We believe the reason for this is the nature of the dataset. Unlike the stock and NOAA data, where the distributions change and are somewhat dependent on the near past, the streaming dataset is seasonal and

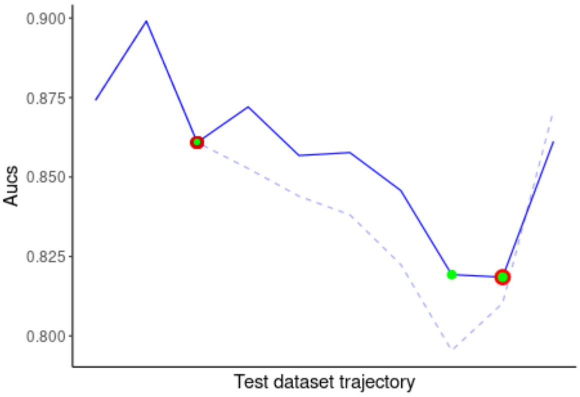

Figure 4: Results of the proposed method on NOAA weather measurements dataset.

cyclical. To illustrate the example, fitting the model to data from the spring does not help predict demand or prices in the summer. Then it makes sense that the model that saw the previous summer would perform better.

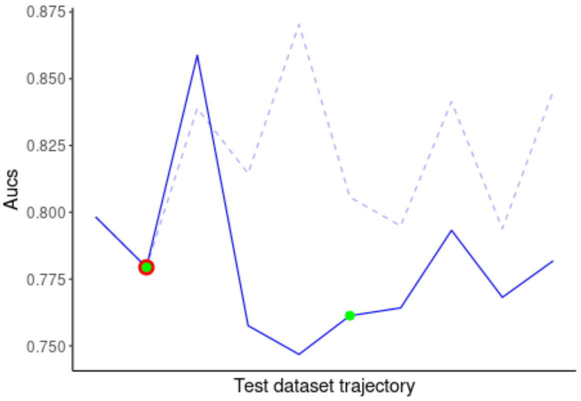

Figure 5: Results of proposed method on the electricity market prices dataset.

6 Conclusions, and Future Work

In this research, we have described how we combined two concept drift methods into a new one, which shows promising results. We evaluated the method on four real-world datasets and achieved comparable or better results than the baseline model, except for the Electricity Market Prices dataset. However, we consider that the performance on that dataset was degraded given that the concept drift detector could not distinguish between concept drift and seasonality changes, therefore discarding valuable data. On the other hand, we were pleasantly surprised with the results obtained from the NOAA dataset: our method significantly reduced the relearning cost and notably increased the models' performance.

Future work will focus on three directions of research. First, we will explore better ways to determine optimal parameters given a concept drift detection (e.g., size of time window considered to retrain and test the model). Second, the concept drift on feature values can be empirically monitored; therefore, such information is used to negate the shift, leading to better performance Turnbull (1976). Finally, we will explore new mechanisms to distinguish between concept drift and seasonality changes to enhance the proposed method further and make it more robust in such scenarios.

Acknowledgments

This work was supported by the Slovenian Research Agency.

References

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

Bifet, A. and Gavalda, R. (2009). Adaptive learning from evolving data streams. In *International Symposium on Intelligent Data Analysis*, pages 249–260. Springer.

Edwards, R. D., Magee, J., and Bassetti, W. C. (2018). *Technical analysis of stock trends*. CRC press.

Gama, J., Medas, P., Castillo, G., and Rodrigues, P. (2004). Learning with drift detection. In *Brazilian symposium on artificial intelligence*, pages 286–295. Springer.

Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., and Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, 46(4):1–37.

Goldenberg, I. and Webb, G. I. (2019). Survey of distance measures for quantifying concept drift and shift in numeric data. *Knowledge and Information Systems*, 60(2):591–615.

Kumar, P. and Gupta, A. (2020). Active learning query strategies for classification, regression, and clustering: a survey. *Journal of Computer Science and Technology*, 35(4):913–945.

Marcu, A. and Prügel-Bennett, A. (2021). On data-centric myths. *arXiv preprint arXiv:2111.11514*.

Merton, R. C. (1976). Option pricing when underlying stock returns are discontinuous. *Journal of financial economics*, 3(1-2):125–144.

Paiva, P. Y. A., Smith-Miles, K., Valeriano, M. G., and Lorena, A. C. (2021). Pyhard: a novel tool for generating hardness embeddings to support data-centric analysis. *arXiv preprint arXiv:2109.14430*.

Souza, V. M. A., Reis, D. M., Maletzke, A. G., and Batista, G. E. A. P. A. (2020). Challenges in benchmarking stream learning algorithms with real-world data. *Data Mining and Knowledge Discovery*, 34:1805–1858.

Taylor, J. B. and Williams, J. C. (2009). A black swan in the money market. *American Economic Journal: Macroeconomics*, 1(1):58–83.

Tsymbal, A. (2004). The problem of concept drift: definitions and related work. *Computer Science Department, Trinity College Dublin*, 106(2):58.

Turnbull, B. W. (1976). The empirical distribution function with arbitrarily grouped, censored and truncated data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 38(3):290–295.

Van Erven, T. and Harremos, P. (2014). Rényi divergence and kullback-leibler divergence. *IEEE Transactions on Information Theory*, 60(7):3797–3820.