GE461: **Introduction to Data Science**
Assignment for Data Stream Mining
April 25, 2024
Due date: May 17, 2024; 11:59 pm
Notes on late submission policy: A penalty of 10 points will be deducted from the assignment score for every two days of delay (May 19, May 21, and May 23). No submissions will be accepted after May 23, 11:59 pm.

In this assignment, you will build evolving data stream classification models. You will use scikit-multiflow, an open-source Python library that provides tools for data stream mining and classification. You can find more information about this library on the official webpage (https://scikit-multiflow.github.io/).

Your task is to train data stream classification methods on the given data streams. The assignment focuses on two main challenges in this domain, namely concept drift, and adversarial attacks (you can find more information about these terms in Section (D.1) of the assignment). You will use test-then-train –i.e., prequential evaluation method to compare performance of the classification models, and report your findings using plots and tables.

Teaching Assistant: Sepehr Bakhshi, sepehr.bakhshi@bilkent.edu.tr

TA Office Hour Administration: Please send your question to Sepehr. He can arrange a meeting upon request.

What to Submit and Work to be Done:

Your submission has two components:

A) **Code**. It must contain proper comments. You must also include your name at the top as a signature that confirms that you are the programmer. Please remember that MOSS is in our plans for plagiarism check.

B) **Report**. Your report must be in pdf form and must cover all "Work to be Done" sections of the assignment (part D.1 (Concepts) and D.2 (Requirements)). Explicitly write the steps that you have taken in each section.
- Please also write a brief introduction that includes the overall structure of the report (You can use introduction part of the scientific papers as an example to write a similar one for this homework).
- For each section of your work explain the purpose and what has been done and achieved in that section.
- Provide a comparison of results that contains tables and plots as appropriate.
- Make sure that you follow the principles of scientific writing.
- Use simple past or simple present tense in your report. If you plan to propose future work then in that case you may use future tense.

Your report must have proper title like a scientific paper, reflecting its true content. It must have your name and address etc. If you like, for experience and fun, you may use the ACM conference paper format[1]. You must use latex or Microsoft Word or their equivalent.

Optional: As an optional part, at the beginning of your report, you may have a related works section that covers data stream mining briefly with proper references.

Optional: Another optional part is comparison of the effectiveness of the methods using statistical tests. The design and administration of these tests should be decided by you by looking at the available papers in literature.

See Justin Zobel's book *Writing for Computer Science* for further hints on the style of CS related scientific paper writing.

C) Submitting Your Work. You will submit your work by uploading it to Moodle in a zipped file. Its name must be streamMiningYourFirstNameYourLastName. For a student with the name "Ali Can Ok" it is streamMiningAliOk.

[1] https://www.acm.org/binaries/content/assets/publications/taps/acm_layout_submission_template.pdf

D) Work to be Done. In this section, we start with a brief explanation of the concepts you will learn by doing this assignment. Following that, you will find the requirements for the implementation, including the packages you need to install. The third part will guide you on how to use and generate the datasets, as well as how to implement the data stream classifiers. Finally, you are asked to report your results and findings in the fourth part.

1. Concepts

- *Data Stream*: refers to an environment where data arrives continuously over time. This means we cannot assume that we have access to all the data at the beginning. Instead, we need to update our model incrementally as new data arrives. This is in contrast to traditional batch learning, where we can access all the data simultaneously and train the model on the entire dataset.
- *Concept drift*: refers to a change in the underlying distribution of the data. For example, in the case of a spam email detection model, the characteristics of spam emails may change over time, which means that the system needs to adapt and update itself to identify the new characteristics of spam emails correctly. Data stream classification models need to adopt a concept drift detection and handling method to address concept drift. We refer to a data stream with concept drift as evolving data stream.
- *Adversarial attack:* Adversarial setting assumes a poisoning attack that may be conducted in order to damage the underlying classification system by forcing an adaptation to false data.
- *Prequential Evaluation:* is a commonly used evaluation approach for data stream classification tasks. In the prequential evaluation method, the model is tested on each incoming instance before it is used to update the model. It is also known as interleaved test-then-train evaluation method.

2. Requirements

You will need the following libraries in Python to complete this challenge:

• numpy: a Python library for numerical computing
• scikit-learn: a library for machine learning in Python
• scikit-multiflow: a library for data stream mining and classification

3. Dataset Preparation and Classifier Implementation Details

a) Dataset preparation

You will use two synthetic and two real datasets as data streams to compare performance of the classification models. As synthetic data streams, use AGRAWALGenerator and SEAGenerator classes from scikit-multiflow to generate 100,000 data instances for each. For future access, write the generated data instances into files named AGRAWALDatasetr and SEADataset.

To apply concept drift on these datasets, please read the documentation of SEAGenerator documentation. Apply three abrupt drift points at 25,000, 50,000 and 75,000 points.

As real datasets, you will experiment with the Spam and Electricity datasets. You can obtain these datasets from https://github.com/ogozuacik/concept-drift-datasets-scikit-multiflow.

b) Implementations for Handling Concept Drift

b.1) Implement an instance of the following classification models available on scikit-multiflow.

- Adaptive Random Forest (ARF)
- Streaming Agnostic Model with k-Nearest Neighbors (SAM-kNN)
- Dynamic Weighted Majority (DWM)

b.2) Build your own ensemble from scratch. Use HoeffdingTreeClassifier classifier as your base learner. Find a solution for handling drift in your ensemble and implement it. The ensemble approach should be

implemented from scratch. You are only allowed to import the HoeffdingTreeClassifier, your preferred drift detector, and other basic libraries (numpy, pandas, etc.)

Hint: You can use drift detection mechanisms included in the scikit-multiflow to detect drifts and react to them.

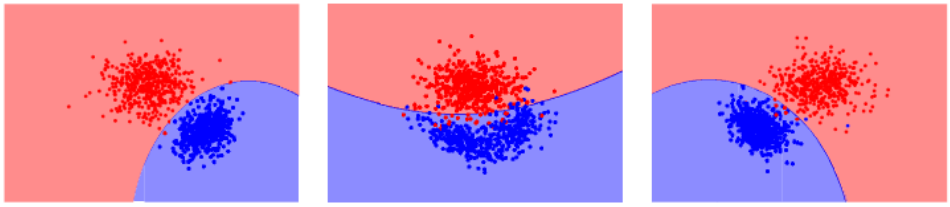
c) Implementations for Handling Adversarial Attack

Data streams are prone to malicious poisoning of the data, and the continuous learning paradigm makes the model vulnerable to these attacks. The attacks may resemble concept drift, and the model should be able to differentiate between a "valid" concept drift and an adversarial attack that seems like a drift but is actually the result of a malicious attack. Figure 1 and Figure 2 compare a valid concept drift with an instance-based adversarial attack. In the valid concept drift, the decision boundaries are correctly adapted to the changes in the data distribution (as shown in Figure 1.b and 1.c). In Figure 2, the adversarial attack causes a false detection of drift, leading to an inappropriate adaptation of the decision boundary based on the malicious attack.

Adversarial attack can happen in two ways in data streams:

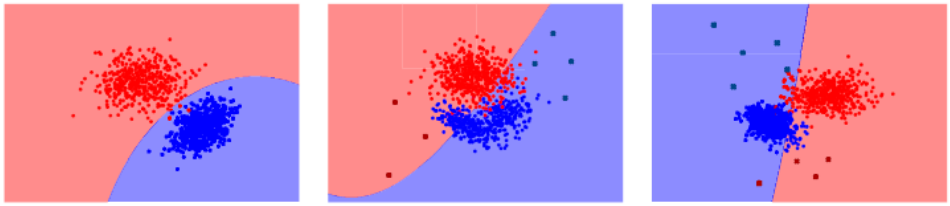
- **Instance-based attacks:** The first type assumes that the malicious party injects singular corrupted instances into the stream. They may be corrupted original instances with flipped labels or with modified feature values.
- **Concept-based attacks:** With the concept-based poisoning attack, we may assume that the malicious party poses a significant knowledge about the real data distributions and is able to craft such concepts that are going to directly cause false alarms and conflicts with valid concept drift.

Note 1: The definitions are from the paper named "Adversarial Concept Drift Detection Under Poisoning Attacks for Robust Data Stream Mining" by Korycki and Krawczyk [1]. Please refer to the paper for more information about these attacks.



(a) Before drift (b) Valid drift (c) Second valid drift

Fig. 1 Accurate adaptation to valid concept drift



(a) Before drift (b) Hindered adaptation (c) Overfitting

Fig. 2 Adversarial drift via instance-based poisoning attacks hinders (**b**) or exaggerates (**c**) the adaptation process

c.1) In this assignment, our focus is on **instance-based attacks**. For two synthetic datasets generated in part (a), synthesize a malicious attack in two points:

- Between the 40,000 and 40,500 data instances, by flipping 10% of the labels randomly.
- Between the 60,000 and 60,500 data instances, by flipping 20% of the labels randomly.

Note 1: The poisoned data instances should be labeled correctly when used for testing, with the changes applied only in the training phase.

Note 2: If the negative effect of the adversarial attack on overall accuracy is very minimal (less than 0.5 percent), try increasing the flipping percentage to more than 20%.

c.2) Propose and implement a solution for handling this attack. You can modify your ensemble model in b.2 for this purpose. How can your proposed ensemble approach handle such an attack and differentiate it from concept drift? In this part, the logic behind your proposed approach and its implementation is important. If the results are not aligned with your expectations, please explain the possible reasons for the failure and ways we can improve it.

Hint 1: "How can we differentiate between the changes that we want to follow and changes that may be of an adversarial nature? Here, we should assume that concept drift will become increasingly present as the stream progresses, while adversarial attacks may have a periodic nature and usually constitute only a small portion of the incoming instances. It is highly unlikely that we will have equal proportions of valid and adversarial instances in the stream" [1].

Hint 2: Unlike concept drift, adversarial attacks appear as isolated outliers.

Hint 3: To effectively manage an adversarial attack, the first step is to develop a method to detect it. Once detected, you can simply exclude the compromised data items during the training phase to prevent the model from being influenced by the attack.

4. Results and Discussion

Construct an instance of the classification models. For each dataset, use Interleaved Test-Then-Train approach to train and evaluate performance of these classifiers. Use prediction accuracy as evaluation metric. In your comparisons use plots and tables when appropriate. Number all plots and tables and provide proper subtitles for them. Make sure that you refer to each of them in the text of your report. Report the following results for the classification models on each dataset:

- Overall accuracy: Overall prediction accuracy of the models.
- Prequential accuracy plot: Prequential accuracy is defined as the prediction accuracy of a model over the most recent w data instances. Use sliding windows of size 1,000 data instances to calculate prequential accuracy values. Plot the obtained accuracy values over time for each dataset.

4.1. How does your ensemble model perform compared to the state-of-the-art approaches in b.2? What could be possible improvements for a more robust ensemble.

4.2. Discuss your findings on the accuracy plots. What is inferred from the drops in the prequential accuracy?

4.3. Discuss your findings and results in section c. Use plots and tables to report and analyze your results. Compare two versions of your ensemble (the one in section b.2, and after applying your method on handling adversarial attack in section c.2).

4.4. Please also include a paragraph that summarizes your findings in this assignment. What did you learn from this assignment?

5. Grading Policy:

Clarity and Organization of Report: **10 Points**
Dataset Preparation: **10 Points**
Implementations for Handling Concept Drift, b.1: **20 Points**
Implementations for Handling Concept Drift, b.2: **35 Points**
Implementations for Handling Adversarial Attack, c.1 and c.2: **25 Points**

Total: **100 Points**

Note 1: To earn full points for each part, you should provide the respective implementation and include the results and discussion as outlined in Section 4.

Note 2: Multiple submissions are allowed and only the last submission will be graded according to the last submission date.

References:

[1] *Korycki, Ł., & Krawczyk, B. (2023). Adversarial concept drift detection under poisoning attacks for robust data stream mining. Machine Learning, 112(10), 4013-4048.*