MACHINE LEARNING FOR DATA STREAMS

ENSEMBLE

- Ensemble methods can also be considered as blind approaches. In fact, the general technique applied by these methods is that the data stream is divided into sequential blocks of fized size, and each of these blocks is used to train a classifier. The ensemble is continously refined by adding a new classifier, removing the oldest or the weakest classifier, increasing or decreasing the classifier weights using some criteria usually based on current data block performance. (Paper “Classifying evolving data streams with partially labeled data (2011)”)
- Ensemble methods have the advantage of robustness in the context of data streams. It should be pointed out that many of these methods use sampling in order to improve the classification accuracy. (Document “A Survey of Stream Classification Algorithms (2014)”)
- According to Krawczyk et al. (2017, Ensemble survey), data stream researchers are shifting their focus to ensemble-based solutions. The performance of these solutions depend on the strength of their base learners and the statistical correlation between them. Hence, ensembles can use only weak learners as long as their correlation is low (Breiman, 2001). Thus, learners with very similar predictive performance could be used as base learners for an ensemble and have virtually the same performance. However, the use of several base-learners increase memory costs, limiting the use of ensembles. (Paper “Strict Very Fast Decision Tree: a memory conservative algorithm for data stream mining (2018)”)
- Paper “Making Data Stream Classification Tree-based Ensembles Lighter (2018)”: Recently, several classification algorithms capable of dealing with potentially infinite data streams have been proposed. One of the main challenges of this task is to continuously update predictive models to address concept drifts without compromise their predictive performance. Moreover, the classification algorithm used must be able to efficiently deal with processing time and memory limitations. In the data stream mining literature, ensemble-based classification algorithms are a good alternative to satisfy the previous requirements.
- Ensemble batch learning algorithms such as Boosting and Bagging have proven to be highly effective from disk–resident data sets. These techniques perform repeated resampling of the training set, making them a priori inappropriate in a data streams environment. Despite what might be expected, novel ensemble methods are increasingly gaining attention because of they have proved to offer an improvement in prediction accuracy. In general, every incremental ensemble approach uses some criteria to dynamically delete, reactivate, or create new

- ensemble learners in response to the base models' consistency with the current
- data.

1. **Online bagging and boosting (2001)_NOT READ YET**

   RESUMEN DEL PAPER "Making Data Stream Classification Tree-based Ensembles Lighter (2018)": Oza [7] adapted the idea of bagging and boosting to an online scenario, creating the OzaBag and OzaBoost algorithms.

2. **New Options for Hoeffding Trees(2007)_NOT READ YET**

   RESUMEN DEL PAPER "Strict Very Fast Decision Tree: a memory conservative algorithm for data stream mining (2018)": Another algorithm based on VFDT, the Hoeffding option tree (HOT) (Pfahringer et al., 2007), includes option nodes, which makes an instance go down into multiple leaves. An option node is essentially a split node with multiple conditions. Thus, a new instance travels along all children nodes whose conditions are true. HOT performs a prediction by averaging the weight of the predictions of all leaves reached. This algorithm presented predictive performance higher than VFDT, at the cost of significant memory increase.

3. **A Practical Approach to Classify Evolving Data Streams_Training with Limited Amount of Labeled Data (2008): No aparece en los surveys (2), (3), (5) y (6)_NOT READ YET**
   - Reference in the paper "Classifying evolving data streams with partially labeled data (2011)": To the best of our knowledge, only two relevant previous works have addressed the problem of scarceness of labeled instances in concept drifting data streams. … The second work was recently proposed by Masud et al. [22]. It is based on an ensemble approach where each model in the ensemble is built as micro-clusters using a semi-supervised clustering technique. In fact, the learning step of each model starts by choosing k_c points from the labeled data of class C to initialize k_c centroids. Then the EM algorithm is applied by iterating the following two steps until convergence: The E-step assigns each unlabeled data point x to a cluster such that its contribution to a cluster-impurity function is minimized, and the M-step recomputes each cluster centroid by averaging all the points in that cluster. Finally, a summary of the statistics of the instances belonging to each built cluster is saved as a micro-cluster. These micro-clusters serve as a classification model.
   - Reference in the paper "Classifying evolving data streams with partially labeled data (2011)": To cope with stream evolution, Masud et al. [22] keep an ensemble of L models. Whenever a new model is built from a new data chunk, they update the ensemble by choosing the best L models from L+1 models (previous L models and the new model), based on their individual accuracies on the labeled instances of the new data chunk.

Besides, they refine the existing models in the ensemble whenever a new class of data evolves in the stream.

- Reference in the paper “Classifying evolving data streams with partially labeled data (2011)”: Note finally that this approach is blind since it does not incorporate any drift detection method.
- Reference in the document “A Survey on Ensemble Learning for Data Stream Classification (2017)”: In [Masud et al. 2008] instances are grouped into microclusters, which are then used as input to a k-Nearest Neighbor ensemble to predict new instances class labels.
- Reference in the document “A Survey on Ensemble Learning for Data Stream Classification (2017)”: It uses clustering methods based on a radius measure, similar to the classic k-means algorithm, therefore they are unable to capture non-spherical clusters and often degrade to a single large cluster.

4. **Leveraging bagging for evolving data streams (2010)_NOT READ YEY:**

   RESUMEN DEL PAPER “Making Data Stream Classification Tree-based Ensembles Lighter (2018)”: By further extending and exploring this idea (Online bagging and boosting (2001)), Bifet et al. [8] proposed an ensemble algorithm called Leveraging Bagging (LevBag), which combines the ADWIN [9] with the bagging strategy in [7].

5. **Mining recurring concepts in a dynamic feature space_MReC-DFS (2014):**

RESUMEN MReC-DFS (RGBNC): They utilized the Naive Bayes (NB) algorithm with ensemble weighting mechanism to handle the recurring concept drift for data stream classification. In their method, the ensemble weight mechanism considered the accuracy and error values.

Due to the dynamic nature of data, classes and data samples are not constant over the period of time. So, considering accuracy and error may affect the performance of the classification if one class attribute has bigger data samples. So, the multiple objective criteria like, sensitivity, specificity should be included to ensemble weighting.

6. **Data Stream Mining using Decision Tree Learning Algorithms (2014)_NOT READ YET: No aparece en los surveys (1), (2), (3), (4), (5), (6) , (7), (8) y (9)**

7. **Adaptive random forests for evolving data stream classification (2017)_NOT READ YET**

8. **Making Data Stream Classification Tree-based Ensembles Lighter (2018): No aparece en los surveys (1), (2), (3), (4), (5), (6) , (7), (8) y (9)**
   - To manage the trade-off between accuracy, memory space, and processing time, this paper proposes to use the Strict VFDT (SVFDT) algorithm as an alternative weak learner for ensemble solutions which is capable of

reducing memory consumption without harming the predictive performance.