A Robust Drift Detection Algorithm with High Accuracy and Low False Positives Rate SafeAI 2023

Maxime Fuccellaro, Laurent Simon, Akka Zemmari

February 13, 2023

Table of Contents

Concept Drift

Concept drift occurs when the underlying distribution of a data source changes over time.

- Drift renders predictive models obsolete
- Drift can appear at various speeds or in a cyclical manner

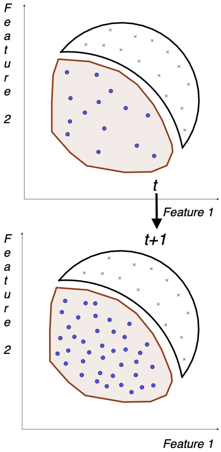
There are three types of drift

$$\mathbb{P}(y \mid X)_{(c)} = \frac{\mathbb{P}(X \mid y)_{(b)} \, \mathbb{P}(y)_{(a)}}{\mathbb{P}(X)}$$

Drift Types: Class Priors

$$\mathbb{P}(y \mid X) = \frac{\mathbb{P}(X \mid y)\,\mathbb{P}(y)}{\mathbb{P}(X)}$$

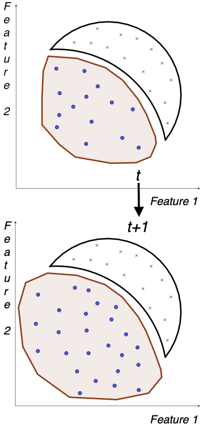
- This drift is characterized by a change in class prior probabilities
- Example: A credit card fraud spike
- Class priors change won't impact trained models but can lead to class imbalance



Drift Types: Posterior Probability

$$\mathbb{P}(y \mid X) = \frac{\mathbb{P}(X \mid y)\mathbb{P}(y)}{\mathbb{P}(X)}$$

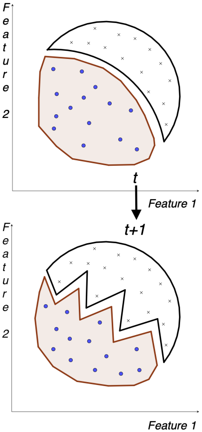
- Class domain definition will change, class boundaries remain unchanged
- Example: Unseen values in the feature space are fed to a model



Drift Type: Class Distribution

$$\mathbb{P}(y \mid X) = \frac{\mathbb{P}(X \mid y)\,\mathbb{P}(y)}{\mathbb{P}(X)}$$

- The decision boundaries change with time
- Example: New regulations kick in loan applications



Handling Drift

Drift can be handled by design or be detected.

- Models built to handle drift
    - Online learners
    - Ensemble methods
- Drift Detectors, can be
    - Prediction based
    - Windowing techniques

Motivations

Today, a very large number of drift detectors exist. However:

- Most drift detectors need real labels
- Window based drift detectors work in one dimension

We aim to tackle drift detection without those constraints.

Real-Drift Detector

- A Decision Tree is trained over some labelled data, a weight is assigned to each leaf based on it's separation power
- For each leaf, we grab the training and inference samples that belongs to it
- On each dimension, we check for variance equality in between the training and inference sets
- If the ratio of features that fail the homoscedasticity test exceeds a user defined threshold, the leaf is drifting
- A drift is detected when the weighted sum of drifting leafs exceeds a user defined threshold

Experimental Setup

- We shuffle the data to remove existing drifts
- We induce virtual and real drift by adding perturbations (Noise and Shuffle) on the least and most informative features

Virtual vs Real Drift

	Train	Val.	LN	LS	MN	MS
Adult	1.	.85	.85	.85	**.28**	**.59**
Bank	1.	.94	.92	.94	**.52**	**.52**
Cov	.99	.85	.84	.84	**.49**	**.46**
D08	1.	1.	.99	.99	**.77**	**.69**
D17	1.	.99	1.	1.	**.54**	**.80**
Elec	1.	.89	.87	.87	**.57**	**.62**
Musk	1.	.98	**.94**	.97	**.51**	**.56**
Phis.	.99	.97	.96	.96	**.69**	**.47**
Spam	1.	.98	.98	.98	**.58**	**.59**
Wine	1.	1.	1.	1.	**.63**	**.44**
Hyp.	1.	.87	.87	.85	**.71**	.87
LED	1.	1.	1.	1.	**.58**	**.31**
Wav.	1.	.85	.85	.85	.72	**.41**

Results

Table reports the metric $2 * \frac{\widehat{DA} * TN}{\widehat{DA} + TN}$.
Where $\widehat{DA}$ is the Drift Accuracy and TN the True Negative rate.

	Adult	Bank	Cov	D08	D17	Elec	Musk	Phish.
RDD	**0.99**	**0.96**	0.67	0.80	**0.92**	**0.93**	**0.86**	0.93
ADWIN	0.00	0.00	0.00	0.00	0.00	0.50	0.00	0.00
D3	0.60	0.36	0.50	0.09	0.00	0.50	0.00	0.00
KS	0.09	0.00	0.60	0.00	0.00	0.60	0.00	0.00
MMD	0.00	0.00	0.46	0.00	0.00	0.00	0.00	0.00
ST	0.81	0.39	**0.92**	0.71	0.75	0.87	0.50	**1.00**
TSDD	0.86	0.86	0.67	**0.81**	0.75	0.67	0.75	0.67

Results

	Spam	Wine	Hyp.	LED	Wav.
RDD	**0.89**	**0.89**	**0.86**	0.77	**0.86**
ADWIN	0.00	0.00	0.71	0.48	0.29
D3	0.00	0.00	0.71	0.50	0.29
KS	0.00	0.00	0.65	0.00	0.36
MMD	0.00	0.00	0.59	0.51	0.19
ST	0.73	0.67	**0.86**	0.91	0.71
TSDD	0.65	0.52	**0.86**	**0.92**	**0.86**

End words

We introduced RDD, a drift detection method that will not need labels during inference making it a good alternative for real-world applications.
We empirically showed that RDD ignores virtual drift due to the structure of the decision tree algorithm.
It shows a good compromise between False Positives, False Negatives. We also demonstrated it's ability to work in any dimension. We showed that there is still room for improvement when it comes to detecting drift in an unsupervised environment.