PHT (Page Hinkley Test):

Definition:

The PH(Page-Hinkley) test is a sequential analysis technique typically used for moinotoring change detection in the average of a Gaussian signal.

Formula/approach:

This test considers a cumulative variable mT, defined as the cumulative difference between the observed values and their mean till current moment.





The minimum value of the variable mT is also computed (MT).

Finally, the test monitors the difference between MT and mT given by PHT and if the difference is greater than a given threshold(λ) drift is said to be detected.

Threshold

The threshold is provided as input(λ) and is based on the user-preference and the input parameter. Giving a small threshold, raises many false alarms while increasing it might miss/delay a drift.

When to use/ assumptions

The PHT is used for detecting model drift. This is achieved by passing in any parameter that defines the model performance. We have taken f1-score as the input parameter stream. Any model with a valid f1-score can be test for drift using the PHT.

Data required

As mentioned, we need f1-scores of the model over a period of time.

Ground truth is required for PHT.

Links

1.Main paper: <https://www.jstor.org/stable/2333009>

2. <https://repositorio.inesctec.pt/bitstream/123456789/5355/1/P-002-0BR.pdf>

3. <http://www.liaad.up.pt/area/jgama/DataStreamsCRC.pdf>

4. <https://ria.ua.pt/bitstream/10773/18961/1/109_ISMIS2017.pdf>

LFR (Linear Four Rates):

Definition

LFR is a framework for detecting the concept drifts and subsequently identifying the data points that belong to the new concept.

It can be applied to both batch and stream data and is not limited by distribution properties of the response variable. It is independent of the underlying statistical model.

Formula/approach:

LFR identifies the change in P(ˆyt,y) where ˆyt is the predicted value of a binary classifier.

The true positive rate, true negative rate, positive predicted value and negative predicted value are found from the confusion matrix.

LFR framework uses estimators of the rates in P\*, where (\* € {tpr, tnr, ppv, npv}).

LFR uses modified rates R\*(t) as the test statistics for P\*(t). R\*(t) is a modified version of the empirical rate Pˆ\*(t). At each t, R\*(t) is updated as : R (t) ? ← η\*R\*(t−1) + (1 − η\*)1{yt=ˆyt} for those empirical rates \* “influenced by (yt, yˆt)”. R\*(t) is essentially a linear combination of classifier’s previous performance R\*(t−1) and current performance 1{yt=ˆyt}, where η\* is a time decay factor for weighting the classifier’s performance at current instance.

Threshold

The threshold is defined by the user. LFR has three user defined parameters as follows:

* Time decaying factor
* Warning significance level
* Detection significance level

When to use/ assumptions

LFR is used for binary classification models only as the framework is based on metrices calculated from a confusion matrix.

Data required

The ground truth is required for the model being considered.

Links

1. <https://arxiv.org/pdf/1504.01044.pdf>

DDM(Drift Detection Method):

Definition

DDM (Drift Detection Method) [1] is a concept change detection method based on the PAC learning model premise, that the learner’s error rate will decrease as the number of analysed samples increase, as long as the data distribution is stationary.

If the algorithm detects an increase in the error rate, that surpasses a calculated threshold, either change is detected or the algorithm will warn the user that change may occur in the near future, which is called the warning zone.

Formula/approach

The detection threshold is calculated in function of two statistics, obtained when (pi + si) is minimum:

* *p*min: The minimum recorded error rate.
* *s*min: The minimum recorded standard deviation.

At instant *i*, the detection algorithm uses:

* *p*i: The error rate at instant i.
* *s*i: The standard deviation at instant i.

The conditions for entering the warning zone and detecting change are as follows:

* if *p*i + *s*i ≥ *p*min +2∗ *s*min -> Warning zone
* if *p*i + *s*i ≥ *p*min +3∗ *s*min -> Change detected

Threshold:

There are no user defined thresholds for DDM. As could be seen from the formula above, the threshold is calculated from the data fed into the algorithm.

When to use/ assumptions:

The DDM assumes that the data stream is composed by a set of contexts i.e., we assume that the function generating examples is stationary.

DDM is used for binary classifiers only as we need to feed in 0/ 1 based on value being predicted correctly or not.

Data required

The input for DDM is a stream of 1 and 0. 0 is fed when the underlying models predicts the output correctly and is equal to ground truth. 1 is fed when underlying model wrongly predicts from ground truth.

Ground truth is required.

DDM is applicable only for streaming data and the input needs to be fed into the function one at a time.

Links

* 1. <https://www.researchgate.net/publication/220974771_Learning_with_Drift_Detection>
  2. <https://scikit-multiflow.readthedocs.io/en/stable/api/generated/skmultiflow.drift_detection.DDM.html?highlight=ddm#r09c26f62a89a-1>

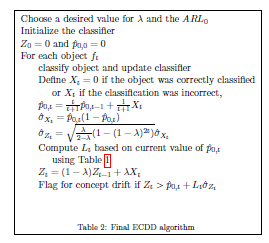
ECDD:

Definition

ECDD is a method for detecting concept drift in streaming classification problems based on the Exponentially Weighted Moving Average chart.

Formula/approach

Given a streaming classification problem, first choose both the classifier to use, a desired value for the ARL0, and a value for λ. Objects in the stream are sequentially presented to the classifier, and at each time point define Xt = 0 if the predicted class label was correct, and Xt = 1 if it was incorrect. The estimates are updated using Xt. The EWMA estimator Zt is updated and then it is flagged if concept drift has occurred.



Threshold

The threshold(λ) is user-defined and is generally taken as 0.2.

When to use/ assumptions

It can be used for any streaming classification error stream and is assumed to have ground truth.

Data required

Ground truth is required.

Input needs to be fed in a stream format.

Links

1. <https://arxiv.org/pdf/1212.6018.pdf>