Bayesian approach to abrupt concept drift detection

Cano, A., Gomez-Olmedo, M., Moral, S.
{acu, mgomez, smc}@decsai.ugr.es

July 5, 2019

Abstract

This vignette describes the software developed for the experimental work presented in the paper *A Bayesian Approach to Abrupt Concept Drift* in order to allow the reproduction of the experiments.

1 Introduction

This vignette is included to descrive the *R* package developed for performing the experimental work of the paper *A Bayesian Approach to Abrupt Concept Drift*. The package is named *acdr (Abrupt Concept Drift in R)* and allows the user to perform all the experiments included in the paper as well as define new estimation methods and perform new comparisons.

2 Package structure

The package is organised in a set of folders commented in the following subsections.

2.1 R folder

R folder contains **R** files with the code of the package. The files included in it are:

- **utils.R**: utility functions for the methods of estimation.
- **execution.R**: functions related to the execution of experiments.
- **estimateXX.R**: code for the the different algorithms of estimation considered for the paper where XX is the number identifying the particular method and in some cases the acronym used to identify the method in the paper. These files contains two different functions: **estimateXX** and **sexpXX**. The first one implements the real method of estimation having as arguments the stream to analyze and the concrete values for its parameters. The second one allows to execute a batch of tests varying some of the parameters in a certain interval and using several cores in parallel in some cases.
- **email.R**: sentences required for the analysis of email data.

2.2 data folder

This folder contains the **csv** file with emails data.

2.3 extdata folder

It contains the description of the experiments to perform with each estimation method. The files in this folder are named using the format: $X - experiment$ and $X - experiment - test$. X number denotes the method of estimation. The set of files with names $X - experiment$ contain the complete description of the experiments while the second set ($X - experiment - test$) contain simplified versions just for testing the correct execution of the algorithms.

The structure of these files is:

- the first row contains a number with the length of the data sequence to generate and how many changes in the sequence will be considered. Therefore a random value of p will be used and a sequence with the given length will be generated using this value. This procedure will be repeated as many times as indicated by the second number. As an example a content $1000, 4$ for this first line will produce a complete sequence of 4000 data where each subsequence of 1000 corresponds to a given value of p.
- the second line contains the number of repetitions to perform for each estimation, in order to obtain accurate results being independent of the particular sequence generated
- from the third line on follows a description of particular tests. As an example let us consider the file $1 - experiment$ for $estimate1$ method. This method required two parameters: a window size and a value for α parameter. Therefore, the line $1, 5, 40, 0.01$ specifies: estimation method (1), a first value of window size to consider (5), a last value of window size to analyze (40) and α value (0.01).

The particular configuration for each estimation method will be described below.

2.4 results folder

This folder will contain the results obtained from each estimation. This allows to store the results and to perform the experiments through different sessions as well as analyzing the results in a posterior step. Result files follow the format:

- identifier of estimation mehod (a number)
- identifier of the iteration (a number)
- a list of parameter names and values (pairs of strings and values)
- **res** extension

For example, the file named $1 - 45 - n - 11 - alpha - 0.05.res$ will contain the result of the 45th iteration with **estimate1** with a value of 11 for n parameter and 0.05 value for α parameter.

3 Estimation methods and parameters

3.1 ADWIN

This the algorithm considered as the state of the art for concept drift detection. This method uses a single parameter named δ. The experiment considers the best value for this parameter, 0.2, as indicated in the corresponding paper. This is showed in the file **11-experiment**:

```
1000,4
100
11, 0.2
```

The command for making the experiments is:

```
> experiment("../extdata/11-experiment")
```

3.2 BAF

This algorithm requires two arguments: n (window size) and α (value to used for statistical tests). The experiments performed for this method consider values of n from 90 to 110 and α values going from 0.0001 to 0.01. The file defining the configuration for the execution of this algorithm is included below (it is stored under **extdata** folder and named **34-experiment**).

```
1000,4
100
34, 90, 110, 0.0001
34, 90, 110, 0.0002
34, 90, 110, 0.0003
34, 90, 110, 0.0004
34, 90, 110, 0.0005
34, 90, 110, 0.0006
34, 90, 110, 0.0007
34, 90, 110, 0.0008
34, 90, 110, 0.0009
34, 90, 110, 0.001
34, 90, 110, 0.003
34, 90, 110, 0.005
34, 90, 110, 0.008
34, 90, 110, 0.01
```

The command to execute for generating the result files for this algorithm is:

```
> experiment("../extdata/34-experiment")
```

The experiments included in the paper consider the following parameterization:

- **BFA01** for $n = 100$ and $\alpha = 0.01$
- **BFA001** with $n = 100$ and $\alpha = 0.001$
- **BFA0001** with $n = 100$ and $\alpha = 0.0001$

Note: the path for the experiment file should change depending on the base folder used for running it. This comment can be applied to the rest of algorithms described in this vignette.

3.3 BFV1

The algorithm requires the following parameters:

- n for window size.
- α_1 and α_2 defines an interval for performing statistical tests.
- k as the number of samples to discard as a result of the forgetting process.

The file called **35-experiment** contains the parameters used for testing this algorithm.

```
1000,4
100
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
```

```
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
35, 90, 110, 0.0001, 0.01, 10
```

The command to execute for getting the corresponding result files is:

```
> experiment("../extdata/35-experiment")
```

The experiments of the paper consider the following parameterization:

- values for α between 0.0001 and 0.01, window size of $n = 100$ and a number of samples to forget $k = 10$ and an uniform prior distribution for α.

3.4 BFV2

This algorithm uses the same parameters as the previous one:

- n for window size.
- α_1 and α_2 defines an interval for performing statistical tests.
- k as the number of samples to discard as a result of the forgetting process.

The experiments for this algorithm are specified in the file **36-experiment**:

```
1000,4
100
36, 90, 110, 0.0001, 0.01, 10
36, 90, 110, 0.0002, 0.01, 10
36, 90, 110, 0.0003, 0.01, 10
36, 90, 110, 0.0004, 0.01, 10
36, 90, 110, 0.0005, 0.01, 10
36, 90, 110, 0.0006, 0.01, 10
36, 90, 110, 0.0007, 0.01, 10
36, 90, 110, 0.0008, 0.01, 10
36, 90, 110, 0.0009, 0.01, 10
36, 90, 110, 0.001, 0.01, 10
36, 90, 110, 0.003, 0.01, 10
36, 90, 110, 0.005, 0.01, 10
36, 90, 110, 0.008, 0.01, 10
36, 90, 110, 0.01, 0.01, 10
```

The comnand for executing the set of experiments is:

```
> experiment("../extdata/36-experiment")
```

3.5 FW

This algorithm presents a single argument defining the size of the active window. The selection of the best value for this parameter was achieved observing the results of the executions included in the file **estimate4-FW.R**:

```
1000,4
100
4, 5, 80
```

The best value is 68 as showed in the paper. The execution of the experiments is launched with the following command:

```
> experiment("../extdata/4-experiment")
```

3.6 FF

This method considers a parameter named ρ defining the forgetting factor of the algorithm. The file used for testing the behavior of the algorithm is **3-experiment**:

```
1000,4
100
3, 0.8, 0.81, 0.82, 0.83, 0.84, 0.85, 0.86, 0.87, 0.88, 0.89, 0.9, 0.91,
0.92, 0.93, 0.94, 0.95, 0.96, 0.97, 0.98, 0.99, 0.999, 0.9999, 1
```

These experiments can be carried out with this command:

```
> experiment("../extdata/3-experiment")
```

The best value estimated by the experinments is 0.97.

3.7 SWB

This method uses two parameters:

- n: size of active window.
- α: value used for the statistical tests.

The file with the definition of the experiments performed for this method is **7-experiment** (stored if **extdata** folder):

```
1000,4
100
7, 5, 40, 0.01
```

The experiments for the algorithm are executed with the command:

```
> experiment("../extdata/7-experiment")
```

Our experiments show that the best value for the significance level of the tests (second parameter) is 0.04.

3.8 SWF

As the previous algorithm, this method uses two parameters:

- n: size of active window.
- α: value used for the statistical tests.

The file named **8-experiment** defines the experiments performed for testing the optimal values for the parameters: 28 and 0.006.

```
1000,4
100
8, 5, 40, 0.001
8, 5, 40, 0.0015
8, 5, 40, 0.002
8, 5, 40, 0.0025
8, 5, 40, 0.003
8, 5, 40, 0.0035
8, 5, 40, 0.004
8, 5, 40, 0.005
8, 5, 40, 0.006
8, 5, 40, 0.007
8, 5, 40, 0.008
8, 5, 40, 0.009
8, 5, 40, 0.01
8, 5, 40, 0.02
8, 5, 40, 0.03
8, 5, 40, 0.04
8, 5, 40, 0.05
8, 5, 40, 0.06
8, 5, 40, 0.08
8, 5, 40, 0.1
```

The experiments for the algorithm are executed with the command:

```
> experiment("../extdata/8-experiment")
```

3.9 SWMTF

This method uses a single parameter for the parameter α defining the significance level of the tests. The file named **11-experiment** contains the configurations tests looking for the best value of this parameter:

```
1000,4
100
15, 0.001, 0.0015, 0.002, 0.0025, 0.003, 0.0035, 0.004, 0.0045, 0.005, 0.0055,
0.006, 0.0065, 0.007, 0.0075, 0.008, 0.0085, 0.009, 0.0095, 0.01, 0.02, 0.03,
0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1
```

The experiments for the algorithm are executed with the command:

```
> experiment("../extdata/15-experiment")
```

The results show that the best value for α is 0.01.

3.10 SWMTFIn

This algorithm requires the following parameters:

- n: size of the active window
- α: value to consider for statistical tests

The content of the file **21-experiment** shows the configurations tested during the experimental work:

```
1000,4
100
21, 5, 40, 0.001
21, 5, 40, 0.0015
21, 5, 40, 0.002
21, 5, 40, 0.0025
21, 5, 40, 0.003
21, 5, 40, 0.0035
21, 5, 40, 0.004
21, 5, 40, 0.005
21, 5, 40, 0.006
21, 5, 40, 0.008
21, 5, 40, 0.01
21, 5, 40, 0.02
21, 5, 40, 0.03
21, 5, 40, 0.04
21, 5, 40, 0.05
21, 5, 40, 0.06
21, 5, 40, 0.08
21, 5, 40, 0.1
```

3.11 SWMTB

This algorithm receives as input a single parameter used for the statistical tests: α. The experiments executed in order to find the best value for it are defined in **23-experiment.R**:

```
1000,4
100
23, 0.001, 0.0015, 0.002, 0.0025, 0.003, 0.0035, 0.004, 0.0045, 0.005, 0.0055,
0.006, 0.0065, 0.007, 0.0075, 0.008, 0.0085, 0.009, 0.0095, 0.01, 0.02, 0.03,
0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1
```

The optimal value is $\alpha = 0.06$.

3.12 SWMTBIn

This method uses two parameters:

- n: size of active window
- α: value to use for statistical tests

The file named **24-experiment** contains all the tested parameterizations:

```
1000,4
100
24, 5, 40, 0.001
24, 5, 40, 0.0015
24, 5, 40, 0.002
24, 5, 40, 0.0025
24, 5, 40, 0.003
24, 5, 40, 0.0035
24, 5, 40, 0.004
24, 5, 40, 0.005
24, 5, 40, 0.006
24, 5, 40, 0.008
24, 5, 40, 0.01
24, 5, 40, 0.02
24, 5, 40, 0.03
24, 5, 40, 0.04
24, 5, 40, 0.05
24, 5, 40, 0.06
24, 5, 40, 0.08
24, 5, 40, 0.1
```