# Machine Learning Topic 4

## Classifier evaluation

In the lecture I performed a comparison of Tree classifiers. In this lab you will do the same for Bayesian classifiers. We do not cover Naïve Bayes or BayesNets as they are covered in both Information Retrieval and Artificial Intelligence. However, it is good to have access to these classifiers. If you are unfamiliar with Naïve Bayes, have a read of the lectures on blackboard (under useful resources).

By the end of lab on topic 3 you had:

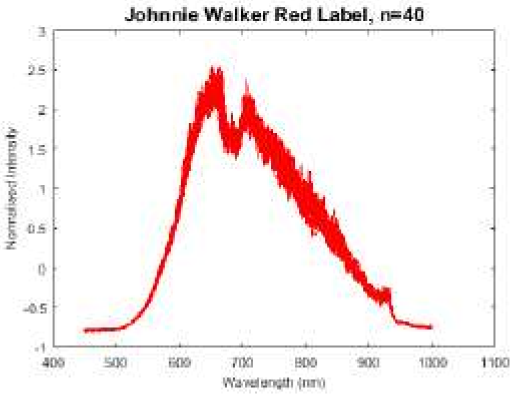
1. Explored ensembles in Weka
2. Performed some exploratory comparisons of different ensembles

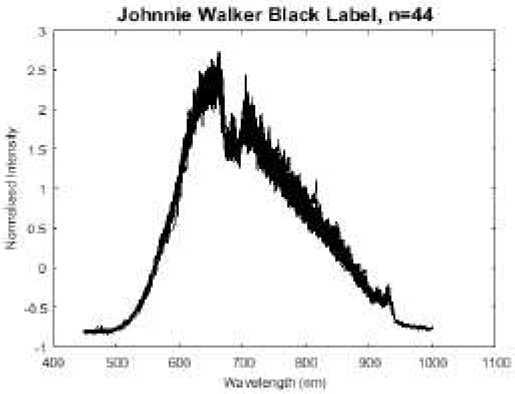
By the end of the lab on topic 4 you will be able to:

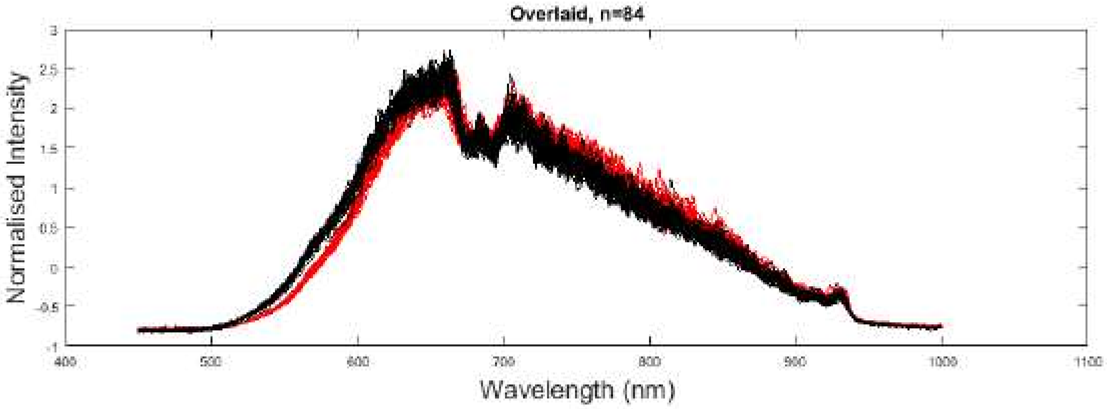
1. load data, perform splits, train a classifier and output the results in the correct format;
2. Perform a cross validation experiment on a single problem and format the output
3. Perform resample experiments over multiple data sets
4. Analyse the results to assess the merits of different classifiers.

# 1. Train/Test single dataset experiments (whisky classification)

Download the whisky data from blackboard. This data forms part of James Large’s PhD project. This is a dataset of spectra of real Johnnie Walker whisky bottles, Red label and Black label. Light is shined through the bottle, non-invasively, and the intensities of wavelengths from the visible light range up to near infrared are stored. The classification problem is to determine whether the bottle contains red-label whisky, or black-label. Black label is more expensive, and cheaper whisky (such as Red label) may be put into black label bottles to be sold for profit, illegally, of course.







1. Write a method that trains a NaiveBayes classifier on the train data, then saves the test output to file
2. Check your results format by loading the results from the above into a `ClassifierResults` object and calculate the stats. This is best done as

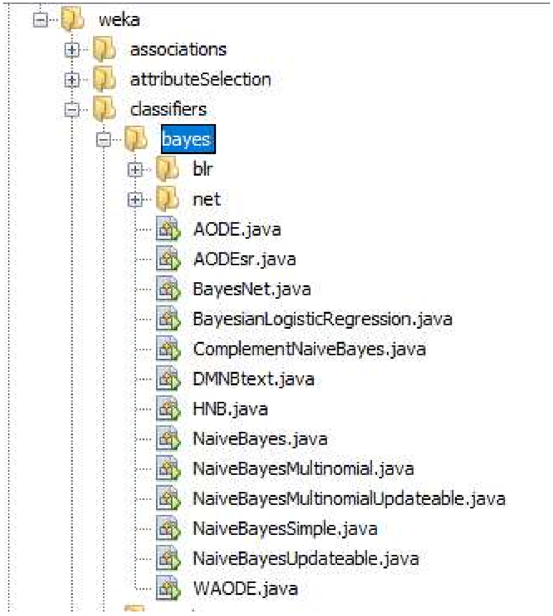
```
ClassifierResults res=new ClassifierResults();

res.loadResultsFromFile(“FULLPATHTORESULTS");

res.findAllStats();
```

Explore the class `ClassifierResults` to see what it can do. Find out the accuracy, balanced accuracy, NLL and AUC for this classifier.

3. Have a look at the Bayesian classifiers

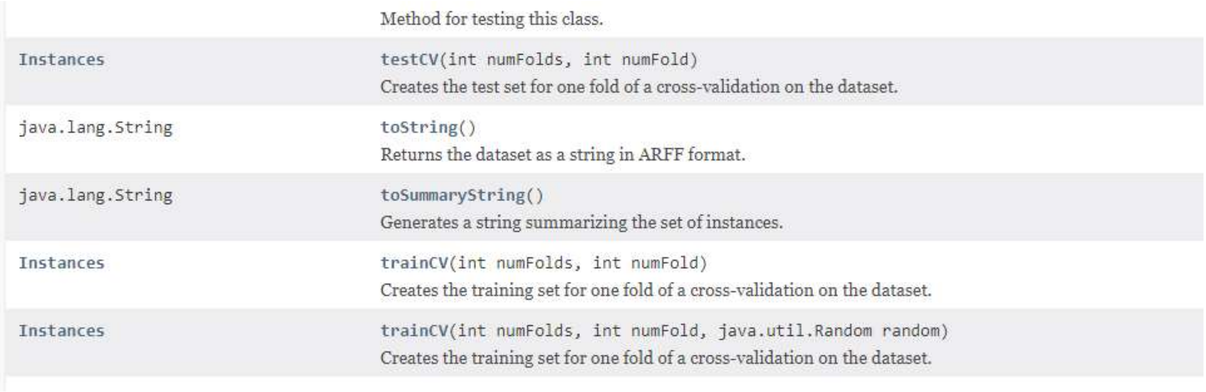


weka
- associations
- attributeSelection
- classifiers
  - bayes
    - blr
    - net
    - AODE.java
    - AODEsr.java
    - BayesNet.java
    - BayesianLogisticRegression.java
    - ComplementNaiveBayes.java
    - DMNBtext.java
    - HNB.java
    - NaiveBayes.java
    - NaiveBayesMultinomial.java
    - NaiveBayesMultinomialUpdateable.java
    - NaiveBayesSimple.java
    - NaiveBayesUpdateable.java
    - WAODE.java

Write a method that can generate train/test results for some of these classifiers on the whisky problem. If a classifier throws an Exception, it probably isn't your fault, some of them have odd restrictions, just pick another. Check the Capabilities of any classifier that doesn't seem to work. For those that do, save the results, generate the stats as above, and compare.

## 2. Cross validate single dataset (whisky classification)

A single train test split gives less information that the test results of a cross validation. The easiest way to cross validate is  to use Weka Instances methods to build the folds then repeat as before



| | |
|---|---|
| | Method for testing this class. |
| `Instances` | `testCV(int numFolds, int numFold)`<br>Creates the test set for one fold of a cross-validation on the dataset. |
| `java.lang.String` | `toString()`<br>Returns the dataset as a string in ARFF format. |
| `java.lang.String` | `toSummaryString()`<br>Generates a string summarizing the set of instances. |
| `Instances` | `trainCV(int numFolds, int numFold)`<br>Creates the training set for one fold of a cross-validation on the dataset. |
| `Instances` | `trainCV(int numFolds, int numFold, java.util.Random random)`<br>Creates the training set for one fold of a cross-validation on the dataset. |

http://weka.sourceforge.net/doc.dev/weka/core/Instances.html

Repeat the above experiments, but now generate a single output file that includes the test results of a ten fold cross validation. To do this, for each fold call trainCV and testCV, build on train, then write the test results to file as before, but write the test to the same file. Repeat the diagnostics as before.

Load the results into a spreadsheet. Can you think of a way of generating the ROC curve?

## 3. Comparing two classifiers over multiple datasets

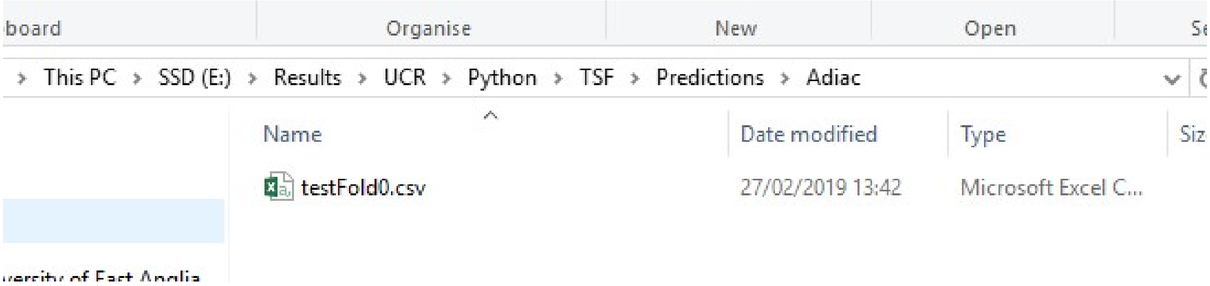
Suppose we wish to test whether Bayes Net is, on average, more accurate than Naïve Bayes (BayesNet can be flakey; If it crashes, just use another classifier). To do this we could compare performance over multiple data sets.   Using the UCI data set from the ensembles lab,

1. **Perform a single resample** of each dataset using 70% for train and 30% for test.
2. **Train** both Naïve Bayes and BayesNet on the same train resample.
3. **Save** the test results for both to a separate file

Use the following file structure if you want to use our code to collate results

E:\\ResultsDirectoryCHANGE\\<ClassifierName>\\<Predictions>\\<ProblemName>\\testFold0.csv

e.g. for a classifier called TSF, we have this



› This PC › SSD (E:) › Results › UCR › Python › TSF › Predictions › Adiac

| Name | Date modified | Type | Siz |
|---|---|---|---|
| testFold0.csv | 27/02/2019 13:42 | Microsoft Excel C... | |

4. **Collate** the results.

Easiest way to do this is to use my code experiments.CollateResults.collate(String[] args)

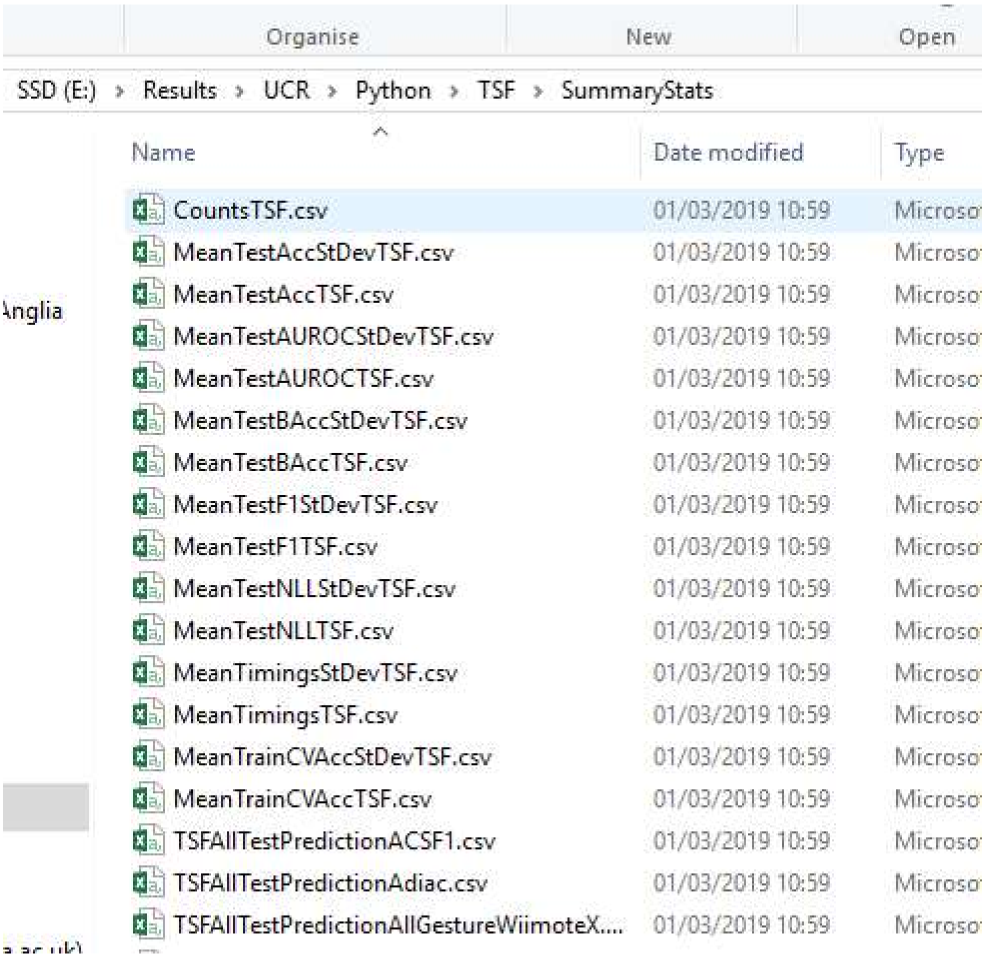
```
String[] str={"E:\\ResultsDirectory\\",
   "Z:\\DataDirectory\\",
  "1",//Number of resamples for each dataset
  "false",//Don’t worry at the moment
  “ClassifierName”,
  ”0”}; //Don’t worry at the moment
  collate(str);
```

so for the above data I would use

```
String[] str={"E:\\Results\\UCR\\Python\\","E:\\Data\\","1","false",“TSF”,”0”};
collate(str);
```

5. **Interpret** the results.

There should be a directory called SummaryStats in the directory of each classifier.



SSD (E:) › Results › UCR › Python › TSF › SummaryStats

| Name | Date modified | Type |
|---|---|---|
| CountsTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestAccStDevTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestAccTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestAUROCStDevTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestAUROCTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestBAccStDevTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestBAccTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestF1StDevTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestF1TSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestNLLStDevTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTestNLLTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTimingsStDevTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTimingsTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTrainCVAccStDevTSF.csv | 01/03/2019 10:59 | Microso |
| MeanTrainCVAccTSF.csv | 01/03/2019 10:59 | Microso |
| TSFAllTestPredictionACSF1.csv | 01/03/2019 10:59 | Microso |
| TSFAllTestPredictionAdiac.csv | 01/03/2019 10:59 | Microso |
| TSFAllTestPredictionAllGestureWiimoteX.... | 01/03/2019 10:59 | Microso |

Compare the test accuracy, balanced test acc, NLL and AUC of Naïve Bayes and Bayes Net. Which is best? Perform a paired t-Test in excel.

**NOTE: Normally we run 30 resamples and average, giving them names testFold0.csv to testFold29.csv.** Set up your code to perform multiple resamples. Run it there is time and compare results to a single fold.

## 4. Multiple Classifiers on Multiple Data Sets

Repeat the previous exercise for two or three other classifiers. Use any we have explored in previous lectures. Manually collate the accuracy results into a single spreadsheet. Download the matlab file critdiff.m (attached to the coursework). Remote into the lab machines if required, open matlab, then do something like this

```
DATA_PATH='C:\Users\ajb\Dropbox\AccuracyResults.csv';
names={'NaiveBayes','BayesNet','C45','IBk'}
data=csvread(DATA_PATH)
critdiff(1-data,names,0.05)
```

you will probably need to sort paths out to use critdiff. What can you conclude from the graph?