Practical 6

**Classification Practice - Example**

# What are we doing?

This practical demonstrates a run-through of a Classification practice example. By doing this practice, you will experience a whole process of a very simple classification project – applying a number of different classification methods to one data set and compare their results and try to evaluate different classifiers.

**Submission:**

You are required to submit one document containing screen-captured images via the weekly-practical submission box (available on CP1407 LearnJCU). You will need to capture your computer screen after completing each of main stages of this practical – to prove your work for this practical.

# Task

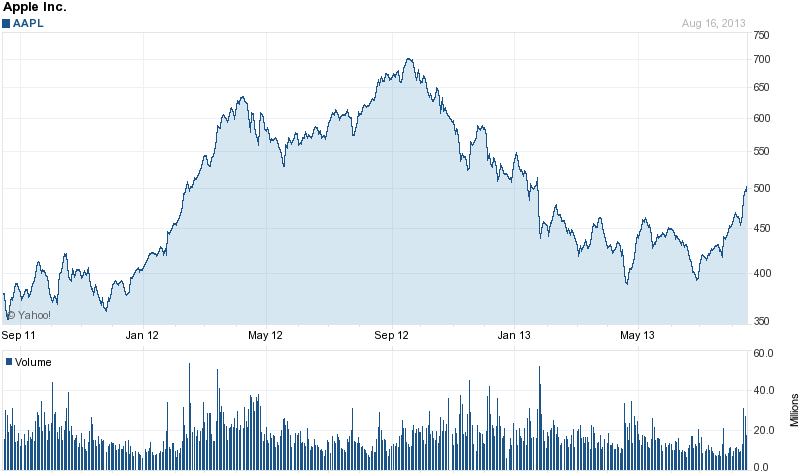
Use two data files provided, both of which are about the share price of Apple Corportation (symbol AAPL) for this practice:

1. **AAPL\_discrete.arff** has the future change in share price in 3 different labels, buy, sell, or hold.
2. **AAPL\_continuous.arff** has the future change in share price still as a continuous number. The changes are sorted and ranked, so 0 represents the biggest drop in share price, and 1 represents the biggest rise in share price, and 0.5 is the median change in share price.

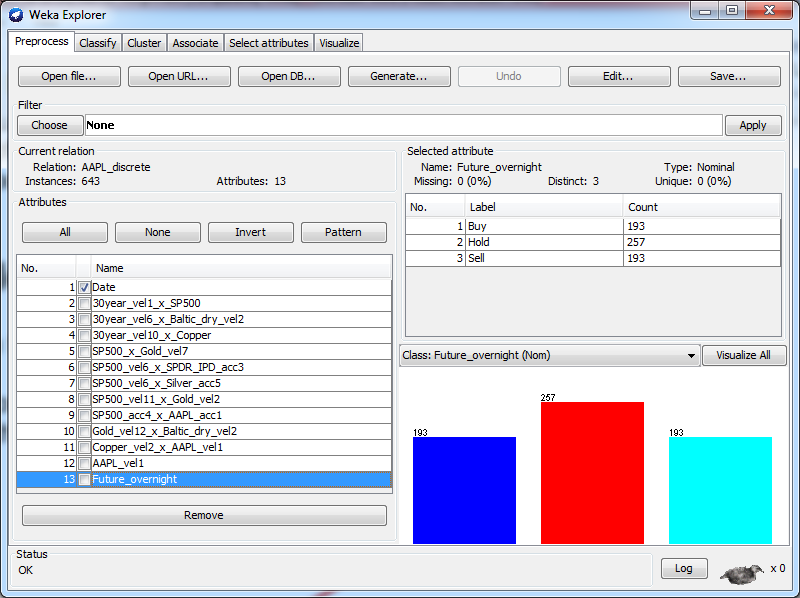
The output is the future change in share prices is the right-hand column, labelled Future\_overnight.

The inputs are a range of data sources, such as the price of gold and oil, interest rates on U.S. Treasury bonds, consumer confidence, and such like.

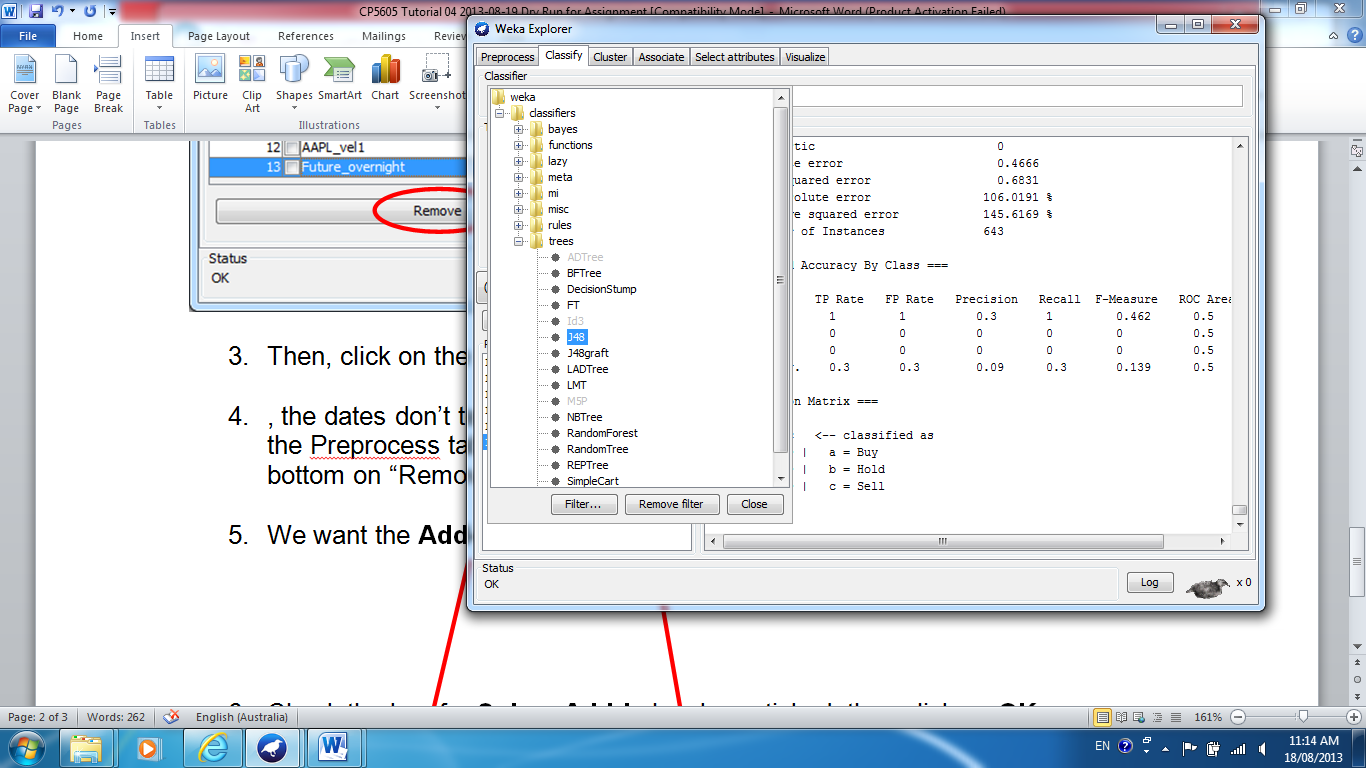
You can do graphs of Apple’s share price at Yahoo Finance http://finance.yahoo.com



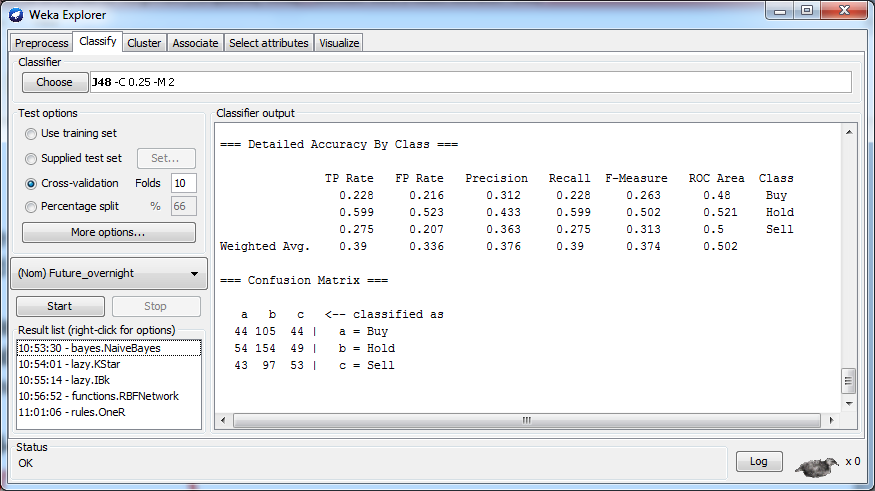
1. Start up Weka on the discrete file, **AAPL\_discrete.arff**
2. First, the dates don’t tell us anything, so we can delete that whole column. Using the Preprocess tab, click on the box that says “Date”, and then click at the bottom on “Remove”.



1. Then, click on the “Classify” tab, so we can choose classifier algorithms. The first one to try would be **J48**, a decision tree algorithm.



1. Scroll down to the bottom of the output, where it talks about the True Positive rate (TP rate) and all those other measures of accuracy from lecture in Week 4.



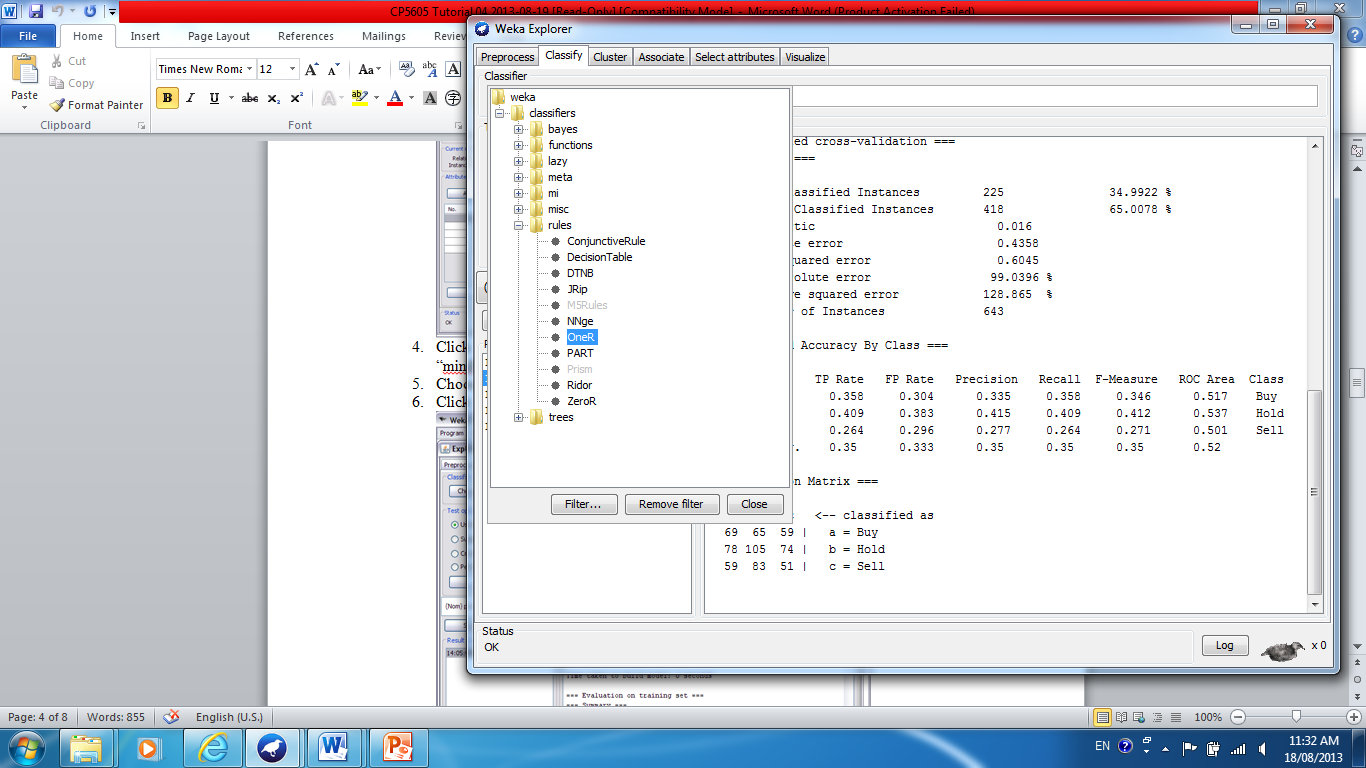
1. Pick a measure that you think you need to maximize or minimize:

* If you want to predict a share price *rise*, maximize TP rate for Buy.
* If you want to predict a share price *fall*, maximize TP rate for Sell.
* If you want to avoid errors in predicting a *rise*, minimize FP rate for Buy.
* If you want general accuracy, maximize ROC Area for Weighted Avg.
* If it’s okay to miss some buying opportunities, but the ones you choose are more likely to be correct, then maximize Precision for Buy.

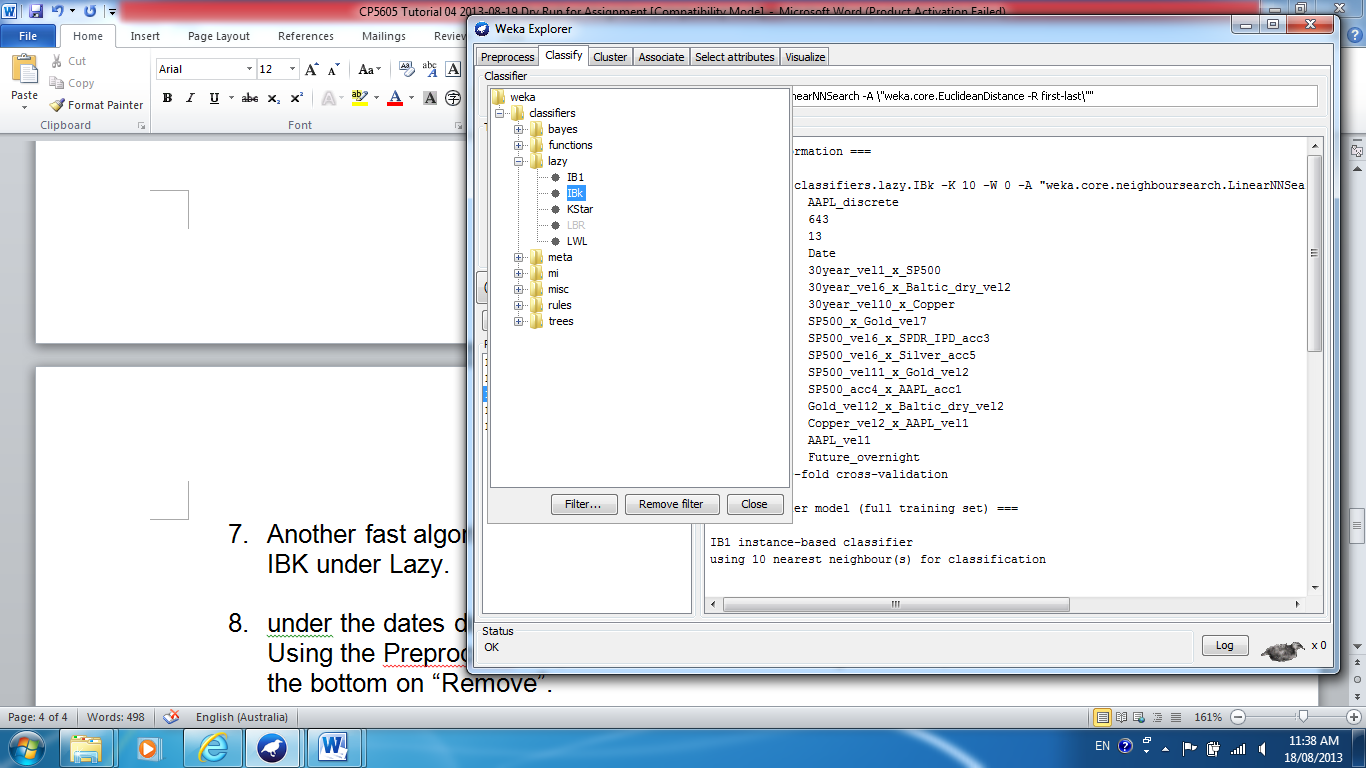
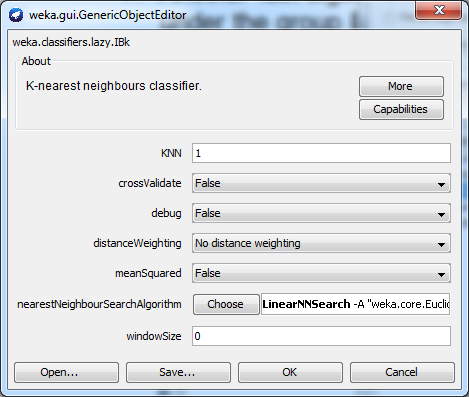
Note that Recall = TP rate, they’re the same thing.

On the assignment, you can choose any measure of accuracy you think is appropriate, but describe why. For this problem, maximizing Precision for weighted average (“Weighted Avg.”) over all 3 outputs might be suitable, as it’s the percentage of cases classified into each class that are indeed in that class. For J48, that number is only 0.376 which is not very high.

1. Now try lots of classification algorithms. **1R** is fast, but how accurate is it?



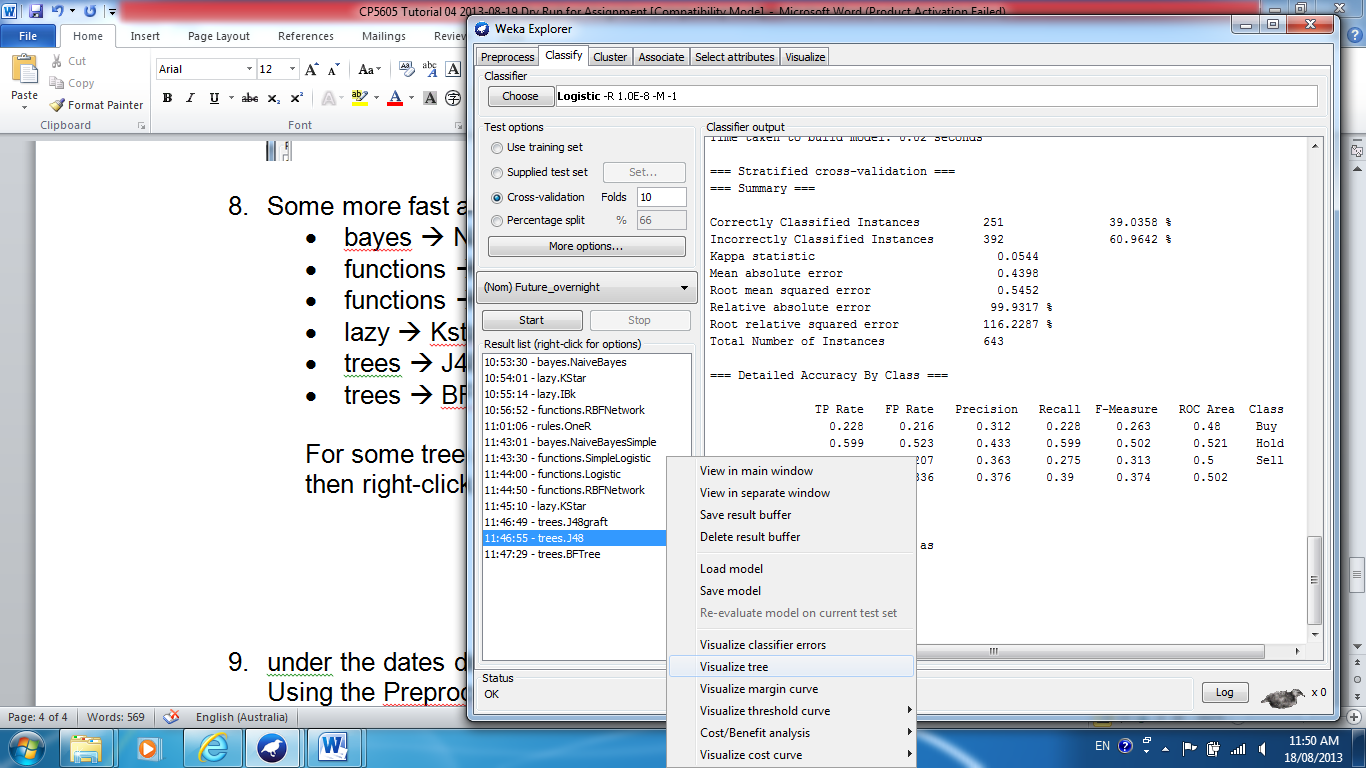
1. Another fast algorithm you should try is K Nearest Neighbour, one version is under the group **lazy** called **IBk** and you should change the value for K to 10.

1. Some more fast algorithms you should try include:

* bayes 🡪 Naïve Bayes Simple
* functions 🡪 Simple Logistic
* functions 🡪 Logistic
* lazy 🡪 Kstar (a more sophisticated K Nearest Neighbour algorithm)
* trees 🡪 J48graft (any better to J48?)
* trees 🡪 BFtree

For some tree algorithms, right-click on the output, select “**Visualize Tree**”, then right-click to “**Fit to screen**”.



1. A slow algorithm is a neural network, try functions 🡪 Multi Layer Perceptron.
2. Now load the file with continuous numbers as outputs, **AAPL\_continuous.arff** and delete the dates column as in step 2 above.
3. As there are no discrete outputs, there is no TP rate or any of those measures. Instead, there are some statistical measures such as:

* **correlation** (bigger is better)
* **relative absolute error** (smaller is better).

Some algorithms only work with continuous output. You could try:

* functions 🡪 Linear Regression
* functions 🡪 Pace Regression
* functions 🡪 Gaussian Processes
* functions 🡪 Multi Layer Perceptron (again, but output is continuous)
* lazy 🡪 IBk (k nearest neighbour again, but output is continuous)

Again, try lots of algorithms that are now available for continuous outputs.

1. So what algorithm(s) give the best results, on your chosen measure of accuracy? Try running them again, with different parameters.

For example, does K Nearest Neighbour work best with knn = 1 or 5 or 10 or 20 nearest neighbours?