# CSCI 3151: Assignment 2

## NOTE: FINAL EXAM IS ON FRIDAY, APRIL 8, 1700-1930, IN TEACHING LAB 4 IN THE COMPUTER SCIENCE BUILDING.

## Q1(40). Association Rule Mining in R

In this exercise, you will learn to work with association rule mining in R, using the `arules` package. Resources on association rule mining with R are available online from RDataMining.com and examples. In order to get some experience with association rules, we work with Apriori. As you will discover, it can be challenging to extract useful information using this algorithm.

To get a feel for how to apply Apriori, start by mining rules from the Titanic dataset (search for "titanic" in the examples script provided). The data is in the data directory. Note that this algorithm expects data that is purely nominal: If present, numeric attributes must be discretized first.

Q1.1
Experiment with different values of support, confidence and maximum rule length. Describe the insights you obtained, and comment on the process of deriving them. What is the total number of possible rules for the Titanic data for sensible combinations of values?

Q1.2
Consider now a real-world dataset, `vote.arff.txt`, which gives the votes of 435 U.S. congressmen on 16 key issues gathered in the mid-1980s, and also includes their party affiliation as a binary attribute. This is a purely nominal dataset with some missing values (corresponding to abstentions). It is normally treated as a classification problem, the task being to predict party affiliation based on voting patterns. However, association-rule mining can also be applied to this data to seek interesting associations. More information on the data appears in the comments in the ARFF file.

a. Convert the arff file into a csv file using search and replace in your text editor. Read the csv file into R, and set prepare it for the application of association rule mining.

b. Run Apriori on this data with default settings. Comment on the rules that are generated. Several of them are quite similar. How are their support and confidence values related? Experiment with different settings, and comment on the resulting rules.

c. How many rules in the default output involve `Class = republican` versus `Class = democrat`. Can you explain?

## Q2(40). Decision Trees in R

Decision trees are one of the simpler machine-learning methods. They are a completely transparent method of classifying observations, which, after training, look like a series of if-then statements arranged into a tree. Once you have a decision tree, it's quite easy to see how it makes all of its decisions. Just follow the path down the tree that answers each question correctly and you'll eventually arrive at an answer. Tracing back from the node where you ended up gives a rationale for the final classification.

In this lab exercise you will experiment with classification using decision trees using R. The idea here is that you should get a feel for how this classification algorithm works and how it performs on real data sets.

Use the risky loans dataset (`credit.csv`) that we discussed in the tutorial. Split the whole dataset into two subsets: 1) the training subset (70%) and 2) test subset (30%).

a) Experiment with three different random splits of the dataset, and repeat the learning of a decision tree on the training set and its testing (prediction) on the test set. Show the resulting contingency matrix from each split. Comment on how different the contingency matrix is from each split.

b) For each split you computed in part a) apply the learned decision tree to predict the training set. Show the resulting contingency matrix from each split. Do you observe that performance on the training set is significantly better than on the test set? What would be a possible explanation of your observation?

c) For each split you computed in part a) learn decision trees that take into account different costs of false-positive and false-negative errors. Experiment with two cost matrices in which the errors have relative costs $(1, 4)$ and $(4, 1)$ respectively. Show the resulting contingency matrix from each split and cost matrix. What do you observe in the performance of the classifier and in the type of errors that occur, in comparison with the performance observed in part a)?

## Q3(40). kNN classification in R

In this question, you will experiment with the kNN classifier we discussed in the tutorial, using the breast cancer data set (`wisc_bc_data.csv`).

a) Repeat the experiment using three different splits of the data set, where the training subset is 70% of the data and test subset is 30% of the data set. Show the resulting contingency tables for $k = 11, 21,$ and 27. Comment on how different is the performance of the classifier for the different $k$ values.

b) Repeat part a) but set the optional argument prob of the knn function to `prob=TRUE`. For each split compute the statistics of the returned probability vector. What do you conclude about the consistency of the results?

c) Repeat part a) but this time the training subset should be 20% of the data set. What was the impact of a significantly smaller training set?

## Q4(20). Probability Review

A box contains 10 white and 5 red balls; a second box contains 20 white and 20 red balls.
a. A ball is drawn from each box. What is the probability that the first will be white and the second red?
b. One of the two boxes is chosen at random and a ball is drawn from the selected box. What is the probability that the **ball** is white?
c. One of the two boxes is chosen at random and a ball is drawn from the selected box. The ball is white. What is the probability that the selected box is the first box?

## Q5(40). Naive Bayes classification

In this question you will experiment with Naive Bayes classification for spam detection in SMS.

a) experiment with different frequency thresholds in constructing the dictionary (sms_dict). For each threshold you choose, show the corresponding dictionary size. Report on your findings.

b) experiment with different values of the Laplace estimator argument to the `naiveBayes` function of R. Report on your findings.

c) report on the computation time (training and testing) for each of your experiments in part (a).

**Additional exercises will be posted up to one week before the due date. It is strongly recommended to start working on the assignment as soon as possible, and to upload solved questions as you complete them (as successive versions of your submission), as per instructions below.**

**Instructions for submitting the assignment.**

Your assignment should have a cover page with the following information:

CSCI 3151 (summer 2014)
Assignment X (where X = 1, 2, ...)
Last name
First name
Banner ID
CS username

The file that you submit for assignment X should be named as:
`CSusername-aX.pdf` If you submit a supplementary file on a specific question Y, name it as:
`CSusername-aX-qY.pdf`

**Grading**: Each question will be graded with a letter grade, based on content (0.7), and quality of writing / neatness (0.3). If the assignment is so poorly written that content becomes difficult to understand with normal effort, then content mark will be further reduced.

Typesetting assignments is labour-intensive due to the mathematics. Therefore, typeset assignments are not required. If your handwriting is not perfectly legible, you may want to type the text, and fill in the math by hand. If you write your assignment by hand, please scan it for submission (use black marker pen for the handwritten parts, scan as black/white into a pdf file to minimize the file size).

The overall grade for the assignment will be a weighted average of the individual grades. Letter grades are being averaged using their GPA equivalent. No rounding takes place. The meaning of the grades is as per the University Calendar, section 17.1. This style of grading will be used in all evaluation components of this course.