AUCKLAND UNIVERSITY OF TECHNOLOGY
TE WĀNANGA ARONUI O TAMAKI MAKAU RAU

SCHOOL OF Engineering, COMPUTER AND MATHEMATICAL SCIENCES

FACULTY OF DESIGN AND CREATIVE TECHNOLOGIES

BACHELOR OF COMPUTER AND INFORMATION SCIENCES

Data Mining and Knowledge Engineering - COMP 723

Semester 2, 2019

Assessment 1

Due date: 12 midnight, August 30 2019, through Turnitin

Submission:
- The assignment must be submitted via AUTonline via the submission tab. This assignment must be answered on an individual basis.

Marking:
- This assignment will be marked out of 100 marks and is worth 40% of the overall mark for the paper.
- To pass this module you must pass each assessment separately, and gain at least 50% in total. The minimum pass mark for this assignment is 35%.

The purpose of this assignment is two-fold. Firstly, it gives you an in-depth exposure to real world data mining applications, involving retail sales, banking, insurance and other sectors of Business, Commerce and Health sectors. Secondly, it provides you with an opportunity to solve real-world data mining problems using the R and Weka machine learning workbenches.

Part A – Literature Review of Data Mining Applications

In this part of the assessment you are required to source and review one case study involving the application of Data Mining in Industry. You will need to select ***ONE*** paper from a list of 3 papers provided at the end of this section. Your review must be provided in a report form and include the following:

- Background information on the organisation that initiated the Data Mining application.
- A brief description of the target application (e.g. detecting credit card fraud, diagnosing heart disease, etc.) and the objectives of the data mining exercise undertaken.
- A description of the data used in the mining exercise (the level of detail published here will differ due to commercial sensitivity, hence flexibility will be used in the marking of this section).
- A description of the mining tools (data mining software) used, together with an identification (no details required) of the mining algorithms *and* how the mining algorithms were applied on the data.
- Discussion of the outcomes and benefits (be as specific as possible, talk about accuracy of results, potential or actual savings in dollar terms or time savings; do not talk in vague, general terms) to the organisation that resulted from the mining exercise. This discussion should contain, in addition to the published material, your own reflection on the level of success achieved by the organisation in meeting their stated aims and objectives.

Your review must be approximately 2 pages in length (excluding the Introduction).

Criterion	**Marks**
Overall Quality of Presentation	5
Application background description, description of data, identification of mining objectives	7
Tools and Mining algorithms	8
Outcomes and Benefits	10

References

Gerritsen, R. (1999). Assessing loan risks: a data mining case study. *IT Professional, 1*(6), 16-21.

Kraft, M. R., Desouza, K. C., & Androwich, I. (2003). Data mining in healthcare information systems: case study of a veterans' administration spinal cord injury population.

Carneiro, N., Figueira, G., Costa M. (2017). A data mining based system for credit-card fraud detection in e-tail. Decision Support Systems, 95 , pp. 91-101

Part B – Data Mining in R and Weka

The objective is to mine a real world dataset and obtain the best possible classification outcome. The dataset that will be used is LSVT which contains data on people who have Parkinson’s disease. Parkinson’s disease causes loss of control over muscles and one of the symptoms is a decrease in the quality of speech. Speech therapy helps such patients but not all of them react well to such therapy. Those whose speech quality improves are categorized as class 1 and those who do not are labelled as class 2.

The overall objective of mining data is to be able to identify both categories with the best possible accuracy so that the effects of therapy can be maximized. The accuracy measure that you need to use is the *weighted F score* taken over both classes of patients.

The dataset is challenging due to two reasons. Firstly, there are 310 features (apart from the class feature) and only a small subset of them are relevant to the task of classifying these patients. Thus, the first challenge to be overcome is to identify which subset of features gives the best possible F score. The second challenge is the imbalanced nature of the dataset – there are 42 patients in class 1 while there are 84 patients in class 2. Hence data balancing methods needs to be applied to improve performance.

F_weighted = (F_1*nc1+F_2*nc2) where F_1 and F_2 are the F ratio values across classes 1 and 2 respectively; nc1 and nc2 are the number of instances of class 1 and class 2 respectively in the test dataset (LSVT_test.arff). Refer to week 3 Lab sheet for the formula to calculate the F value for any given class.

You are required to experiment with four data mining algorithms namely; OneR, J48, Naïve Bayes and 1NN (nearest neighbour, called IBk in Weka). You are required to perform the following tasks:

<u>Task 1: Feature Selection</u>

Write code in R to identify the best set of features by using the Gain Ratio feature selection filter in Weka. Your R code will need to call the Gain Ratio filter with a given number of features (N) to keep. Your first call will identify the best 305 features to keep, the second call will identify the best 300 features, and so on until the effects of keeping the best 5 features are examined. Essentially this means that you will experiment with values of N in the range [5,305] in steps of 5.

For each value of N, you will keep the best (top N) features in the train dataset and then use this subset of features to build a model by applying a mining algorithm on your feature reduced train dataset. You should make use of the code given in week 3 Lab sheet for this task.

Once the model is built on the training dataset you will need to apply it on the test dataset and determine the F-weighted score. When you iterate over the entire range of N [5,305] you will be able to identify the feature set that produced the highest F_weighted score. Note that the value of N that produces the highest F_weighted score can differ from algorithm to algorithm – do not assume that it is the same.

Now repeat the entire process for the rest of the algorithms.

(a) Produce the R code to perform Task 1. Note that your entire code snippet MUST be given for a SINGLE algorithm (say J48). **(7 marks)**

(b) For the other 3 algorithms, there will be no need to supply entire code snippets – only one line that calls the classifier algorithm needs to change, so simply supply that single line of code for the other 3 algorithms. **(3 marks)**

Task 2: Performance Analysis

In this task, you will analyse the performance of each algorithm.

(a) First, run each of the 4 algorithms with the full set of features (N=310) and note the F_weighted score for each of them. **(4 marks)**

(b) Now prepare a 2 by 4 table with algorithms as columns. The first row of the table must contain the F-weighted score for each algorithm with the full feature set (i.e. all 310 features). The second row must contain a pair of values for each algorithm. The first value in the pair should be the highest F_weighted score, while the second value in the pair must be the value of N that produced that highest F value. **(4 marks)**

(c) Explain, for EACH classifier algorithm the effect of applying feature selection. Use your **knowledge** of how that algorithm works to **explain** why feature selection had a positive or negative effect on the F_weighted score. **(9 marks)**

(d) Using this 2 by 4 table identify the mining algorithm that produces the highest F_weighted score after feature selection was performed. **(2 marks)**

Task 3: Data Distribution

In this task, you will use the Resample filter to balance the dataset and attempt to further improve the F_weighted score by balancing the data.

For each of the four algorithms **take the version of the training dataset that produced the best feature set** (the one that produced the highest F-weighted score) from your experimentation in Task1. Extend the R code developed in Lab 4 to determine the combination of “BiasToUniformClass” (B) and “sampleSizePercent” (Z) parameters that produce the highest F_weighted score. You need to experiment with B values in the range [0.3,1.0] in steps of 0.1 and Z values in the range [100,1000] in steps of 100. In order to find the best combination, you need to keep one parameter fixed (say B) at a particular value and then step through the entire range of values for Z. In total this will involve running 80 trials.

(a) Produce the R code for the above data balancing operation. **(9 marks)**

(b) Run the code for **each** of the four algorithms and produce an 8 by 10 table for each algorithm with rows as Z values and columns as B values. Each cell should contain the F_weighted value for that row and column. There should be 4 such tables, one for each algorithm. From each of the 4 tables, identify the combination of B and Z that produces the highest F_weighted score for the given algorithm. **(8 marks)**

(c) For this part you need to use Weka. From the table produced in part (b) above you should be able to identify the best performing algorithm (i.e. the one with the highest F_weighted score).

1) Use this algorithm in the Weka GUI and the version of the training dataset that produced the highest F score. Generate a model using "Use training set" option in Weka. Once the model is created, deploy the model using the "Supplied test set" option and supply LSVT_test.arff as your test set. Once the result is generated, produce a Precision Recall Curve (PRC). This can be done by right clicking in the result pane and selecting the "Visualize threshold curve" option. Select the "1' option to plot the curve for class 1. Choose Precision as the Y axis and Recall as the X axis. Paste this curve into your report. **(3 marks)**

2) Produce a PRC for the same algorithm using the original training dataset (i.e. with all 310 features and no data balancing). Paste this curve into your report as well. **(3marks)**

(d) By comparing the two PRC curves produced in part (c) above, **explain** the effects of feature selection and data balancing on improving accuracy for class 1. **(7 marks)**

Task 4: Building Meta-learner

In this task you need to build a meta-learner using the top 3 algorithms (the algorithms that produced the 3 highest F_weighted scores) in Task 3 (b) above. Use Weka to build the meta-learner. Take each of the top 3 algorithms and use the original training dataset (LSTVT_train.arff) to generate models. For each algorithm, generate a model using the 'Use training set" option, just as you did in Task 3.

Now deploy the model using the "Supplied test set" option and supplying LSVT_test.arff as your test dataset. Before deploying the model, select "More Options" and supply CSV as the 'Output Predictions Option". Once the model is deployed, Weka will output the predicted class value for each instance, just as shown below:

```
inst#, actual, predicted, error, prediction
1,1:1,1:1,,1
2,1:1,1:1,,1
3,1:1,1:1,,1
4,1:1,1:1,,1
5,1:1,2:2,+,1
```

Copy this output into the clipboard and extract the 4th number in each line. The 4th number is the predicted class value for that instance. For example, for instance 1 the predicted class value is 1 and for instance 5 it is 2.

Store the predicted class column only in a .CSV file. Now repeat the process for the other two algorithms. You should now have 3 files, each containing 42 rows and 1 column (predicted class value for that instance).

Create a merged file containing the predicted class values from each of the 3 files. You should now have a single file containing 42 rows and 3 columns (predicted class for alg1, predicted class for alg2 and predicted class for alg3). Save this as a .csv file and import into Weka.

Now use the **Multilayer Perceptron** to build the meta-learner. Use the “Use training set” option to generate the meta-learner. Repeat this with choosing the **Random Forest** to build the meta-learner.

(a) Assess the impact of meta-learning by comparing the F-weighted value obtained through meta learning with the value obtained by running each of the 4 algorithms on the original training dataset. Has it improved accuracy in terms of the F score? **(8 marks)**

(b) How important was the choice of meta-learner algorithm in the mining process? **(3 marks)**