Data Mining
COMP5009/COMP3009

ASSIGNMENT

Due Date: Week 12 - Monday 19-October-2020, 12:00pm Perth time (mid day).
Weight: 40% of the unit mark.

Note: *This document is subject to minor corrections and updates. Announcements will be made promptly on Blackboard and during lectures. Always check for the latest version of the assignment. Failure to do so may result in you not completing the tasks according to the specifications.*

1 Overview

In this assignment, you will solve a real-world data mining problem. This assignment requires you to understand the theory discussed in the workshops, conduct some research into the data mining problem to solve, and use the skills that you should have developed through completing practical exercises to perform various data mining tasks.

Please note that this is an individual assignment. Whilst you may discuss general data mining topics related to this assignment with other students, you must make sure that your work is not accessible by anyone else. There are a large number of choices to make and therefore it is very unlikely to have identical submissions by chance. Submissions that are very similar will be investigated for academic misconduct.

2 Problem Description

In this assignment, you will perform predictive analytics. You are given a CSV data file (`data2020.student.csv`) which contains a total of 1100 samples. The first 1000 samples have already been categorised into two classes. You are asked to predict the class labels of the last 100 samples associated with IDs from 1001 to 1100. You are given the following information

- The attribute Class indicates the class label. For each of the first 1000 samples, the class label is either 0 or 1. For each of the last 100 samples, the class label is missing. You are asked to predict these missing class labels.
- There are exactly 50 samples from each class in the last 100 samples to be predicted.

- Attributes are either categorical or numeric. Note that some attributes may appear numeric. You will need to decide whether to treat them as numeric or categorical and justify your action.
- The data is known to contain imperfections:
  - There are missing/corrupted entries in the data set.
  - There are duplicates, both instances and attributes.
  - There are irrelevant attributes that do not contain any useful information useful for the classification task.
  - The labelled data is imbalanced: there is a considerable difference between the number of samples from each class.

Note that the attribute names and their values have been obfuscated. Any pre-processing and analytical steps to the data need to be based entirely on the values of the attributes. No domain-specific knowledge is available.

Attempt the following:

- **Data Preparation**: In this phase, you will need to study the data and address the issues present in the data. At the end of this phase, you will need to obtain a processed version of the original data ready for classification, and suitably divide the data into two subsets: a training set and a test set.
- **Data Classification**: In this phase, you will perform analytical processing of the training data, build suitable predictive models, test and validate the models, select the models that you believe the most suitable for the given data, and then predict the missing labels.
- **Report**: You will need to write a complete report documenting the steps taken, from data preparation to classification. In addition, you should also give comments or explain your choice/decision at every step. For example, if an attribute has missing entries, you have to describe what strategy taken to address them, and why you employ that particular strategy based on the observation of the data. Importantly, the report must also include your prediction of the missing labels.

You may choose either of the following approaches to complete the assignment:

- **Programming Approach:** If you choose the programming approach, it is required that you submit your Python/R program to produce the prediction. If you plan to use any extra tools/packages, you must obtain a written approval from the Unit Coordinator. This is to ensure fairness among students.
- **Non-Programming Approach:** If you choose the non-programming approach, it is required that you submit an additional document `myweka.pdf` detailing how you use Weka to accomplish the tasks. See Subsection 3.4 for further detail.

3 The Tasks

3.1 Data Preparation

In this first task, you will examine the data attributes and identify issues present in the data. For each of the issues that you have identified, decide and perform necessary actions to address it. Finally, you will need to suitably split the data into two sets: one for training and one for testing, the latter contains 100 samples with missing class labels. The two sets must also be submitted electronically with your report. They must be presented in Weka ARFF format. Your marks for this task will depend on how well you identify the issues and address them. Below is a list of data preparation

- **Irrelevant attributes**: this data set is known to have irrelevant attributes.
  - Describe what you think irrelevant attributes are.
  - For each attribute, carefully examine it and decide whether it is irrelevant. If so, give a brief explanation and remove the attribute.
- **Missing entries**
  - Which attributes/instances have missing entries?
  - For those attributes/instances, how many missing entries are present?
  - For each attribute/instance with missing entries, make a suitable decision, justify it, and proceed.
- **Duplicates**
  - Detect if there are any duplicates (instances/attributes) in the original data?
  - For each attribute/instance with duplicates, make a suitable decision, justify it, and proceed.
- **Data type**:
  - For each attribute, carefully examine the default data type (e.g. Numeric, Nominal, Binary, String, etc.) that has been decided when Weka loads the original CSV file.
  - If the data type of an attribute is not suitable, give a brief explanation and convert the attribute to a more suitable data type. Provide detailed information of the conversion.
- **Scaling and standardisation**:
  - For each numeric attribute, decide if any pre-processing (e.g. scaling, standardisation) is required. Explain why it is needed (this should be discussed in relation to the subsequent classification task).
- **Feature/Attribute selection**: if applicable, clearly indicate which attributes you decide to remove in addition to those (obviously) irrelevant attributes that you have identified above and give a brief explanation why.
- **Data instances**: if you decide to make changes to the data instances with class labels (this may include selecting only a subset of the data, removing instances, randomizing/reordering instances, or synthetically injecting new data instances to the training data, etc.), provide an explanation.
- **Data imbalance**: examine if class imbalance is present and explain why it could be an issue for the prediction. Perform any necessary actions to address class imbalance and justify them.
- **Feature engineering**: you may also come up with attributes derived from existing attributes. If this is the case, give an explanation of the new attributes that you have created.
- **Others**: describe other data-preparation steps not mentioned above.
- **Training, Validation, and Test Sets**: suitably divide the prepared data into training, validation and test sets. These sets must be in ARFF format and submitted together with the electronic version of your report. See the Submission section for further information.

3.2 Data Classification

For this task, you will demonstrate **convincingly** how you select a suitable classification scheme to learn the predictive model from training data and use that model to predict the missing labels. You will also need to **estimate** the prediction accuracy on the actual test data. Finally, you will need to provide your prediction as a table in the report and a CSV file to be submitted electronically. You will need to demonstrate the following:

- **Classifier selection**: you will need to select at least three (3) classifiers that have been discussed in the workshops: k-NN, Naive Bayes, and Decision Trees (J48). Other classifiers, including meta classifiers, are also encouraged. Every classifier typically has parameters to tune. If you change the default parameters to achieve higher cross-validation performance, clearly indicate what the parameters mean, and what values you have selected.
- **Cross validation**: you will need to address the following
  - How to evaluate the effectiveness of a classifier on the given data?
  - How to address the issue of class imbalance in the training data?
  - What is your choice of validation/cross-validation?
  - For each classifier that you've selected, what is the validation/cross-validation performance? Give an interpretation of the confusion matrix.
  - For each classifier that you've selected, what is the estimated classification accuracy on the actual test data?
- **Classifier comparison**:
  - Compare the classification performance between difference classifiers. You need to select at least two (2) evaluation metrics, for example F-measure and classification accuracy, when comparing them. Your comparison must take into account the variation between different runs due to cross-validation.
  - Based on the comparison, select the best two (2) classification schemes for final prediction. Note that the two classification schemes can be one type of classifier, but with two different parameters. Clearly indicate the final choice of parameters if they are not the default values.
- **Prediction**:
  - Use the best two classification schemes that you have identified in the previous step to predict the missing class labels of the last 100 samples in the original data set.
  - Provide your prediction in the report by creating a table, the first column is the sample ID, the second and third columns are the predicted class labels respectively.
  - Produce a CSV file with the name `predict.csv` that contain your prediction in a similar format: the first column is the sample ID, the second and third columns are the predicted class labels. This file must be submitted electronically with the electronic copy of the report via Blackboard. An example of such a file is given below

```
ID,Predict1,Predict2
1001,1,1
1002,1,0
1003,0,0
...
1100,0,1
```

  - IMPORTANT: Please ensure that your prediction is correctly formatted as required. Your marks will be deduced if your prediction file does not meet the above requirements. If your submitted file has more than 2 predictions, only the first two will be marked. No correction to the prediction is allowed after your assignment is submitted.
  - You must also indicate clearly in the report your estimated **prediction accuracy**. This should be based on the validation study.

3.3 Report

You will also need to submit a written report. It should serve the following objectives:

- It demonstrates your understanding of the problem and the necessary steps you have attempted to solve the tasks.
- It contains information necessary for marking your work.

Note of the following restriction on the report

Page limit: your report must not exceed 20 pages. Pages beyond 20 will be ignored when marking!

What you should include in the report:

- Structure of the report
  - Cover page: this must show your identity.
  - Summary: briefly list the major findings (data preparation and classification) and the lessons you've learned.
  - Methodology: address the requirements described above for
    - Data preparation
    - Data classification
  - Prediction: produce a table that describes the best two prediction results.
  - References: list any relevant work that you refer to.
  - Appendices: important things not mentioned above.
- Visual illustration to support your analysis which may include: tables, figures, plots, diagrams, and screenshots.

Note: The report should be concise: the marking of your report is based on the arguments presented and not the length. You may use bullet points to present your analysis. For each issue you address, you must

- Demonstrate that the issue exists in the data with suitable illustration/evidence;
- Provide sufficient detail about the issues, for example which attributes, which instances, which values;
- Explain why addressing the issue is important for subsequent analysis;
- Clearly state your choice of actions to address the issue and provide justification;
- Demonstrate that the actions have addressed the issue satisfactorily.

3.4 Source Code

In addition to the main report which details your analysis of the assignment tasks, you will also need to submit fully commented source code that can be used to reproduce your prediction results.

- **Programming Approach:** If you use this approach, you are required to include all source code (Python or R scripts) in your submission. You must provide a `README.txt` file that explains your program and any known problems. Note that your programs must be able to run from the command line as I will not be using any IDE to test your programs. Please make sure you have the following master script:
  - Python: `run.py`. Your program must run without error with `python run.py`. You may assume that the original data file is in the same directory as the Python script.
  - R: `run.R`. Your program must run without error with `R CMD BATCH run.R`.

  Make sure that all your scripts and dependency are placed under the top level of your submission, do not place them under any subfolder. Before submitting your files, properly test your program by unzipping all contents to a directory and execute the above command in a terminal.
- **Non-Programming Approach:** If you use this approach, you are required to submit a document in **PDF format** named `myweka.pdf` detailing how you use Weka GUI to complete the tasks. The document should also be structured similarly to the report for cross referencing. For each task, you should briefly list the chosen method (e.g. filter, classifier, attribute evaluator, etc.), screenshots and the results that you obtain. Note that a screenshot must be accompanied by some texts describing what the screenshot is for. Submitting only a set of screenshots without any information will result in you loosing the marks for this part. You must ensure that a reader of your document would be able to reproduce what you did. You do not have to explain any reason - it should be in the main report instead. Also note that only an electronic copy is required. Treat this document as source code.

4 Mark Allocation

NOTE As per the unit outline, you need to demonstrate a reasonable attempt of this assignment. **Reasonable attempt** has been defined as scoring at least 40 marks out of 100 marks for this assignment. If you do not achieve this basic pass mark you will fail the unit regardless of how well you perform in the final assessment and the average score.

The total mark of this assignment is 100, and it is distributed as follows

- Satisfactory submission: 18 marks. This is based on
  - All requires files are submitted correctly.
  - Declaration correctly executed and submitted.
  - Either Python/R code or your documentation `myweka.pdf`. For programming students, your code must run without errors and produce the same prediction that you submitted. For non-programming students, your documentation `myweka.pdf` must be clear enough for a reader to reproduce the prediction by following the steps described. It is also expected that the structure of `myweka.pdf` must allow easy cross-referencing with the main report.
  - Abstract/introduction/conclusion and references in the report.
  - The overall presentation of the report.

- Data Preparation: 31 marks. This is based on how well you identify and address issues in the data preparation steps in the report. This includes: irrelevant attributes, duplicates, missing entries, data types, scaling/standardisation, class imbalance, and other inventive steps you took.
- Data Classification: 23 Marks. This is based on how well you present the training, tuning, and validation of different models, and how you arrive at the prediction as described in the report.
- Prediction: 30 Marks. This is based on two factors: actual prediction accuracy (maximum 24 marks) and your estimate of the prediction accuracy (maximum 6 marks). For the actual prediction accuracy, the allocation is as follows:

Accuracy	Marks
$\leq$ 60%	0
61%	1
62%	2
63%	3
64%	4
65%	5
66%	7
67%	9
68%	11
69%	13
70%	15
71%-74%	20
$\geq$ 75%	24

For the estimate of the prediction accuracy, the allocation is as follow:

Estimate of Accuracy	Marks
Within $\pm$ 2%	6
Within $\pm$ 3%	5
Within $\pm$ 4%	4
Within $\pm$ 5%	3
Within $\pm$ 6%	2
Within $\pm$ 7%	1
Outside$\pm$ 7%	0

5 Submission

The assignment is submitted in two parts:

- The main report in PDF format (`report.pdf`) must be submitted through Turnitin. A submission link will be provided on Blackboard.
- Other files must be submitted through another assignment submission link. You must put the following files in a single zip file using your surname and student ID as the name of the zip file (for example `trump_12345678.zip`):
  - ☐ PDF copy of the signed declaration form.
  - ☐ Correctly formatted and named prediction file `predict.csv`.
  - ☐ Training, validation, and test files in ARFF format.

- ☐ Source code (Python/R) or `myweka.pdf`.
- ☐ Any other files that are relevant, such as model files, plots, screenshots that you cannot include in the report and may help explain your approach if needed.

6 Academic Misconduct Plagiarism and Collusion

Please note the following:

Copying material (from other students, websites or other sources) and presenting it as your own work is plagiarism. Even with your own (possibly extensive) modifications,it is still plagiarism.

Exchanging assignment solutions, or parts thereof, with other students is collusion. Engaging in such activities may lead to a grade of ANN (Result Annulled Due to Academic Misconduct) being awarded for the unit, or other penalties. Serious or repeated offences may result in termination or expulsion.

You are expected to understand this at all times, across all your university studies, with or without warnings like this.

END OF ASSIGNMENT