Replace with title page

CITS4009 Computational Data Analysis
Semester Two 2021

This Paper Contains: **7** pages **(including title page)**

Time allowed: 2 hours

INSTRUCTIONS:

This paper contains 6 questions.

TOTAL: 60 MARKS

Students should attempt ALL questions.

Answers are to be written in the answer booklet provided.
Question paper is to be collected with the answer booklet.

- Answers should be concise rather than lengthy.
- If you think that a question is ambiguous, state clearly any assumptions that you make in constructing your answer.
- For questions that require you to write R code, minor syntactic errors will not be penalised; however, syntactic errors that obscure the meaning of your answer might cost you marks. Pseudo code may be given partial marks.

Students can bring in one sheet of A4-size paper with hand-written or typed notes on both sides.

This page has been left intentionally blank

Note that 2021 was a closed book exam. Students were allowed to bring to the exam an A4 sheet of paper containing written or typed note on both sides (so 2 pages). So, some answers in the questions below could be found in the lecture note. The exam paper for 2022 will focus more on applying what you have learned from the lecture note to specific problems described in the questions. **Note: The sample answers given below are incomplete.** You should refer to the lecture note to construct the full answers.

1. a) (2 marks) Briefly state the differences between *hypothesis generation* and *hypothesis confirmation*.

   Ans: (1 mark) Difference #1: in Hypothesis Confirmation, a precise mathematical model is needed in order to generate falsifiable predictions; in Hypothesis Generation, subject knowledge is used to generate interesting hypotheses to help explain why the data behaves the way it does.
   (1 mark) Difference #2: in HC, an observation can be used only once to confirm a hypothesis; in HG, hypotheses are evaluated informally and you can use your skepticism to challenge the data in multiple ways.

   b) (2 marks) Use an example to illustrate that *Exploratory Data Analysis* is an iterative process.

   Ans: (see Chapters 1 or 2 of the textbook)

   c) (2 marks) What are the three essential components of the layered grammar of graphics that *gglot* implements? Give an example for each component.

   Ans: (see Week04.pdf, page 3) 1. a dataset (e.g., `ggplot(df)`); 2. a geom (e.g., ...); 3. a set of mappings (e.g., ...). There are other components also, e.g., a stat, a position adjustment, a coordinate system, and a faceting scheme. However, the above three are the most important components.

   d) (2 marks) What visualisation (or plot) is most suitable to illustrate the covariation between two continuous variables? Give an example and write R code using the *ggplot2* library to illustrate your answer.

   Ans: You should be able to work this out. Breakdown: 1 mark for stating the plot that is suitable and a brief explanation about it. 1 mark for R code (e.g., you could use the *income* and *age* variables from the *customer* dataset).

   e) (2 marks) Briefly explain the differences between a *histogram* plot and a *density* plot. When would it be more suitable to use a density plot than a histogram plot?

   Ans: (1.5 marks) For a histogram plot, values are divided into histogram bins and the frequencies (counts) of the histogram bins are shown in the plot. A proper bin-width value would need to be specified if the default bin-width value is not suitable. This needs to be decided ahead of time. A density plot can be considered as a continuous histogram except that the area under the density plot is rescaled to equal one.

   (0.5 mark) A density plot is more suitable when we are more interested in the overall shape of the curve than the actual values on the vertical axis.

2. a) (4 marks) Given below is a data frame `df` showing the *for sale* prices of some properties in a good suburb in Western Australia. The property type and the number of bedrooms are in the first two columns. The last column contains the prices in 10^3 dollars.

Type	Num.bedrooms	Price
Villa	2	525
House	3	1200
Apartment	1	460
Apartment	2	950
House	3	1000
Villa	1	395
House	4	1300
Villa	2	600

i. (2 marks) Describe a plot that is suitable for visualising variable `Type` versus variable `Num.bedrooms`. Write R code using the *ggplot2* library to illustrate your answer.
Ans: (1 mark) There are several plots suitable. You only need to state one of them. To get the full 1 mark, you need to state why (e.g., what is the type of each variable?). (1 mark) Write the R code for your chosen geom above.

ii. (2 marks) Describe a plot that is suitable for visualising variable `Type` versus variable `Price`. Write R code using the *ggplot2* library to illustrate your answer.
Ans: (1 mark for description; 1 mark for R code) Same as above.

b) (1 mark) Describe when it would be suitable to convert a continuous variable into a categorical one.

Ans: When we are more interested in the range (or interval) of values of the variable rather than its absolute values.

c) (2 marks) Referring to the data frame `df` in part a) above, suppose that we want to convert the `Price` column to the following levels to form a new categorical column called `Price.Range`:

- `Low` if Price $\leq$ 500;
- `Medium` if 500 $<$ Price $\leq$ 1,000;
- `High` if Price $>$ 1,000.

Write R code to add `Price.Range` to the data frame.

Ans:
```
df <- within(df, Price.Range <-
ifelse(Price <= 500000, "Low",
ifelse(Price > 500000 & Price < 1000000, "Medium", "High"))
```
Other variations are also accepted, e.g., using `cut`.

d) (1 mark) Explain what is meant by *listwise deletion* in data cleaning.

Ans: If only a small proportion of values are missing and if they are for the same data points, then we can consider dropping those rows from our analysis. This is called *listwise deletion.*

e) (2 marks) Describe two different ways for imputing missing values in a numerical column.

Ans: (1 mark for each way)
Way 1: impute the missing values by the mean or median value of other values in the column.
Way 2: use a regression model to predict the missing value (if other values for that data point are available).

3. a) (2 marks) Explain what *z-normalisation* is. Is it suitable for detecting outliers? Explain your answer.

   Ans: The key here is z-normalisation involves using the mean which is sensitive to outliers. Your answer should cover more detail about z-normalisationto get the full 2 marks.

   b) (4 marks) For a given vector `v`, five numbers are output by `boxplot.stats(v)$stats`. Explain what each of these numbers represents. By inspecting these numbers alone, can we determine whether `v` is free of outliers? Explain your answer.

   Ans: (2 marks for description of each of the five numbers. 2 marks for the outlier part.) Details can be found in the lecture notes.

   c) (4 marks) Given two data frames `authors` and `books`, which have a common `surname` column, as shown below:

surname	nationality	deceased
Tukey	US	yes
Venables	Australia	yes
Ripley	NZ	no
Tierney	US	no
Winton	UK	no

surname	title	other.author
Tukey	Exploratory Data Analysis	NA
Venables	Modern Applied Statistics	Ripley
Tierney	LISP-STAT	NA
Ripley	Spatial Statistics	NA
Ripley	Stochastic Simulation	NA
McNeil	Interactive Data Analysis	NA
R Core	An Introduction to R	Venables & Smith

   i. (3 marks) Explain the difference between the *inner join* and *left outer join* operations on these data frames.
   Ans: (1.5 marks for *inner join*; 1.5 marks for *left outer join*)
   Inner join and *left outer join* are both **mutating joins**. See the lecture note for more detail. You can include in your description how many observations and how many features (columns) are in each output.

   ii. (1 marks) Write R code to show how the output tables can be produced from the *inner join* and *left outer join* operations on `authors` and `books`.
   Ans: Suggestion: There are a couple of ways to do *inner join* and *left outer join*. For inner join (0.5 mark): you can use `merge` or `inner_join`.
   For Left outer join (0.5 mark): you can use `merge` or `left_join`. Note: you need to supply the full R code.

4. Each row of the data frame `df` below shows the measurement of a type of blood test and whether the patient currently smokes (`smoke`), has never smoked (`never`), or has smoked before but has now quit (`quit`). The last column of the data frame is a binary variable indicating whether the patients have been diagnosed with a type of cancer.

Patient	Smoke	Test	Cancer
1	never	0.56	negative
2	quit	1.10	positive
3	smoke	1.50	positive
4	never	1.20	negative
5	smoke	1.60	positive
6	quit	0.98	negative

a) (4 marks) Explain how a *decision tree* classifier partitions the data frame and assigns a piece-wise constant to each partition. You can use any input feature as the first variable. The output variable that the classifier should predict is the `Cancer` column.

Ans: DT can handle both categorical and numerical variables. See lecture note for detail. Your description should be **specific for the smoke-cancer dataset given here**. Do not copy and paste the lecture note to form your answer as the example given in the lecture note is different.

b) (3 marks) Explain how the *k-nearest neighbours* classifier works for this data frame for predicting the `Cancer` variable. List two aspects that need to be considered during data preparation for this classifier.

Ans: (1 mark) Again, describe kNN for this given dataset.
(2 marks: 1 mark for each aspect) Key ideas to help you construct your answers: a) kNN needs to use distance values to find neighbours, so you need to normalise your features; b) you need to apply proper treatment to missing data, NAs, etc; c) use Hamming distance if you have categorical variables or convert them to numerical (not always possible).

c) (3 marks) Explain how the *receiver operating characteristic curve* and the double density plots can be used to compare the performance of the two binary classifiers above.

Ans: See lecture notes.

5. a) (3 marks) Define what a typical Null model would be like for the data frame in Question 4 above, where the response variable that we want to predict is the `Cancer` column. Write R code to show the predicted probability produced by your Null model.

Ans: (2 marks for definition; 1 mark for R code)
Again, make sure your answer is for the dataset given in Question 4.

b) (2 marks) Explain how the `dist()` function in R can be used to find the distances between data points.

Ans: More details are in the lecture note. The output from the function is a lower triangular matrix (as the matrix is symmetric so only half of the matrix is needed. If there are n data points then there are n-choose-2 distance values.

c) (3 marks) What is the *k-means* algorithm designed for? Outline the steps involved in this algorithm.

Ans: See Figure 9.16 (page 333) in the textbook or the lecture note for the steps.

d) (2 marks) Explain what the *Calinski-Harabasz index* measures.

Ans: See lecture note or the textbook.

6. Given below are the first 8 observations of data frame `df` for a simple *dry bean* dataset. It has 3 classes in the last column and 4 features (or variables): `Perimeter`, `roundedness`, `ShapeFactor1`, and `ShapeFactor2`.

Perimeter	roundness	ShapeFactor1	ShapeFactor2	Class
954.496	0.864	0.00598	0.00119	Cali
716.507	0.954	0.00641	0.00250	Seker
1040.323	0.853	0.00561	0.00105	Cali
776.180	0.877	0.00632	0.00224	Seker
898.660	0.698	0.00605	0.00224	Seker
941.694	0.855	0.00593	0.00132	Barbunya
1105.912	0.851	0.00514	0.00108	Cali
750.314	0.945	0.00609	0.00247	Seker

a) (1 mark) Write R code to relabel the `Class` column to the following values:
   - 1, to replace `Seker`, and
   - 0, to replace `Cali` and `Barbunya`

   for binary classification.

   Ans: Too easy. You should be able to do it yourself.

b) (2 marks) Write an R function called `calDeviance`, which should take in two arguments, `ytrue` (for the ground truth vector) and `ypred` (for the predicted vector). The function should compute the *deviance* and return it as the output value. You may assume that the saturated model has zero deviance.

   Ans: Too easy. You should be able to do it yourself.

c) (2 marks) Write R code to split the dataset into a training set and a calibration set. Use an 80/20 ratio for the splitting.

   Ans: Too easy. You should be able to do it yourself.

d) (5 marks) For each of the 4 features, write R code to
   i. (2 marks) train a *logistic regression* classifier model using the training set (Hint: you can use the `glm` function),
      Ans: Too easy. You should be able to do it yourself.
   ii. (2 marks) apply the trained model on the calibration set, and
      Ans: Too easy. You should be able to do it yourself.
   iii. (1 mark) call the `calDeviance` function above and print the output value.
      Ans: Too easy. You should be able to do it yourself.