FIT2086 Assignment 3

Due Date: 11:55PM, Monday, 17/10/2022

Introduction

There are total of three questions worth $8 + 18 + 14 = 40$ marks in this assignment.

This assignment is worth a total of 20% of your final mark, subject to hurdles and any other matters (e.g., late penalties, special consideration, etc.) as specified in the FIT2086 Unit Guide or elsewhere in the FIT2086 Moodle site (including Faculty of I.T. and Monash University policies).

Students are reminded of the Academic Integrity Awareness Training Tutorial Activity and, in particular, of Monash University's policies on academic integrity. In submitting this assignment, you acknowledge your awareness of Monash University's policies on academic integrity and that work is done and submitted in accordance with these policies.

Submission: No files are to be submitted via e-mail. Correct files are to be submitted to Moodle, as given above. You must submit the following three files:

1. One PDF file containing non-code answers to all the questions that require written answers. This file should also include all your plots.
2. An R script file containing R code answers. Please make sure this is clearly commented so it is obvious which R statements are answering which questions, and the questions are answered in the order they appear in the assignment.

Please read these submission instructions carefully and take care to submit the correct files in the correct places.

Question 1 (8 marks)

This question will require you to analyse a regression dataset. In particular, you will be looking at predicting the fuel efficiency of a car (in kilometers per litre) based on characteristics of the car and its engine. This is clearly an important and useful problem. The dataset `fuel.ass3.2022.csv` contains $n = 500$ observations on $p = 9$ predictors obtained from actual fuel efficiency tables for car models available for sale during the years 2017 through to 2020. The target is the fuel efficiency of the car measured in kilometers per litre. The higher this score, the better the fuel efficiency of the car. The data dictionary for this dataset is given in Table 1. Provide working/R code/justifications for each of these questions as required.

1. Fit a multiple linear model to the fuel efficiency data using R. Using the results of fitting the linear model, which predictors do you think are possibly associated with fuel efficiency, and why? Which three variables appear to be the strongest predictors of fuel efficiency, and why? **[2 marks]**

2. How would your assessment of which predictors are associated change if you used the Bonferroni procedure with $\alpha = 0.05$? **[1 marks]**

3. Describe what effect engine displacement (`Eng.Displacement`) appears to have on the mean fuel efficiency of a car. Describe the effect that the `Drive.SysF` variable has on the mean fuel efficiency of a car. **[2 marks]**

4. Use the stepwise selection procedure with the BIC penalty (using `direction="both"`) to prune out potentially unimportant variables. Write down the final regression equation obtained after pruning. **[1 mark]**

5. Imagine that you are looking for a new car to buy to replace your existing car. The characteristics of the new car that you are looking at are given by the thirty-third row of the dataset.

   (a) Use your BIC model to predict the mean fuel efficiency for this new car. Provide a 95% confidence interval for this prediction. **[1 mark]**

   (b) The current car that you own has a mean fuel efficiency of $11 km/l$ (measured over the life time of your ownership). Does your model suggest that the new car will have better fuel efficiency than your current car? **[1 mark]**

Variable name	Description	Values
`Model.Year`	Year of sale	$2017 - 2020$
`Eng.Displacement`	Engine Displacement (litres, l)	$0.9 - 8.4$
`No.Cylinders`	Number of Cylinders	$3 - 16$
`Aspiration`	Engine Aspiration (Oxygen intake)	`N`: Naturally*
		`OT`: Other
		`SC`: Supercharged
		`TC`: Turbocharged
		`TS`: Turbo+supercharged
`No.Gears`	Number of Gears	$1 - 10$
`Lockup.Torque.Converter`	Lockup torque converter present?	`N`* and `Y`
`Drive.Sys`	Drive System	`4`*: 4-wheel drive
		`A`:All-wheel
		`F`:Front-wheel
		`P`:Part-time 4-wheel
		`R`:Rear-wheel
`Max.Ethanol`	Maximum % of Ethanol allowed	$10 - 85$
`Fuel.Type`	Type of Fuel	`G`*: Regular Unleaded
		`GM`: Mid-grade Unleaded Recommended
		`GP`: Premium Unleaded Recommended
		`GPR`: Premium Unleaded Required
`Comb.FE`	Fuel Efficiency (km/l)	$4.974 - 26.224$

Table 1: Fuel efficiency data dictionary. The * denotes the reference category for each categorical variable.

Question 2 (18 marks)

In this question we will analyse the data in `heart.train.ass3.2022.csv`. In this dataset, each observation represents a patient at a hospital that reported showing signs of possible heart disease. The outcome is presence of heart disease (`HD`), or not, so this is a classification problem. The predictors are summarised in Table 2. We are interested in learning a model that can predict heart disease from these measurements. To answer this question you must:

When answering this question, you must use the `rpart` package that we used in Studio 9. The wrapper function for learning a tree using cross-validation that we used in Studio 9 is contained in the file `wrappers.R`. Don't forget to source this file to get access to the function.

1. Using the techniques you learned in Studio 9, fit a decision tree to the data using the `tree` package. Use cross-validation with 10 folds and $5,000$ repetitions to select an appropriate size tree. What variables have been used in the best tree? How many leaves (terminal nodes) does the best tree have? **[2 marks]**

2. Plot the tree found by CV. Clearly describe in plain English what conditions are required for the tree to predict that someone has heart disease. *(hint: use the* `text(cv$best.tree,pretty=12)` *function to add appropriate labels to the tree).* **[3 marks]**

3. For classification problems, the `rpart` package only labels the leaves with the most likely class. However, if you examine the tree structure in its textural representation on the console, you can determine the probabilities of having heart disease (see Question 2.3 from Studio 9 as a guide) in each leaf (terminal node). Take a screen-capture of the plot of the tree (don't forget to use the "zoom" button to get a larger image) or save it as an image using the "Export" button in R Studio.

   Then, use the information from the textual representation of the tree available at the console and annotate the tree in your favourite image editing software; next to all the leaves in the tree, add text giving the probability of contracting heart disease. Include this annotated image in your report file. **[1 mark]**

4. According to your tree, which predictor combination results in the lowest probability of having heart-disease? **[1 mark]**

5. We will also fit a logistic regression model to the data. Use the `glm()` function to fit a logistic regression model to the heart data, and use stepwise selection with the KIC score (using `direction="both"`) to prune the model. What variables does the final model include, and how do they compare with the variables used by the tree estimated by CV? Which predictor is the most important in the logistic regression? **[3 marks]**

6. Write down the regression equation for the logistic regression model you found using step-wise selection. **[1 mark]**

7. Please describe the effect the variable `CA` has on heart-disease according to this logistic regression model? **[1 mark]**

8. The file `heart.test.ass3.2022.csv` contains the data on a further $n' = 92$ individuals. Using the `my.pred.stats()` function contained in the file `my.prediction.stats.R`, compute the prediction statistics for both the tree and the step-wise logistic regression model on this test data. Contrast and compare the two models in terms of the various prediction statistics? Does one seem better than the other? Justify your answer. **[2 marks]**

9. Calculate the *odds* of having heart disease for the 10th patient in the test dataset. The odds should be calculated for both:

   (a) the tree model found using cross-validation; and

   (b) the step-wise logistic regression model.

   How do the predicted odds for the two models compare? **[2 marks]**

10. For the logistic regression model using only those predictors selected by KIC in Question 2.5, use the bootstrap procedure (use at least $5,000$ bootstrap replications) to find a confidence interval for the odds of having heart disease for the 65th and 66th patients in the test data. Use the `bca` option when computing this confidence interval.

    Using these intervals, do you think there is any evidence to suggest that there is a real difference in the population odds of having heart disease between these two individuals? **[2 marks]**

Variable name	Description	Values
AGE	Age of patient in years	$29 - 77$
SEX	Sex of patient	M = Male F = Female
CP	Chest pain type	Typical = Typical angina Atypical = Atypical angina NonAnginal = Non anginal pain Asymptomatic = Asymptomatic pain
TRESTBPS	Resting blood pressure (in $mmHg$)	$94 - 200$
CHOL	Serum cholesterol in mg/dl	$126 - 564$
FBS	Fasting blood sugar $> 120mg/dl$?	<120 = No >120 = Yes
RESTECG	Resting electrocardiographic results	Normal = Normal ST.T.Wave = ST wave abnormality Hypertrophy = showing probable hypertrophy
THALACH	Maximum heart rate achieved	$71 - 202$
EXANG	Exercise induced angina?	N = No Y = Yes
OLDPEAK	Exercise induced ST depression relative to rest	$0 - 6.2$
SLOPE	Slope of the peak exercise ST segment	Up = Up-sloping Flat = Flat Down = Down-sloping
CA	Number of major vessels colored by flourosopy	$0 - 3$
THAL	Thallium scanning results	Normal = Normal Fixed.Defect = Fixed fluid transfer defect Reversible.Defect = Reversible fluid transfer defect
HD	Presence of heart disease	N = No Y = Yes

Table 2: Heart Disease Data Dictionary. ST depression refers to a particular type of feature in an electrocardiograph (ECG) signal during periods of exercise. Thallium scanning refers to the use of radioactive Thallium to check the fluid transfer capability of the heart.

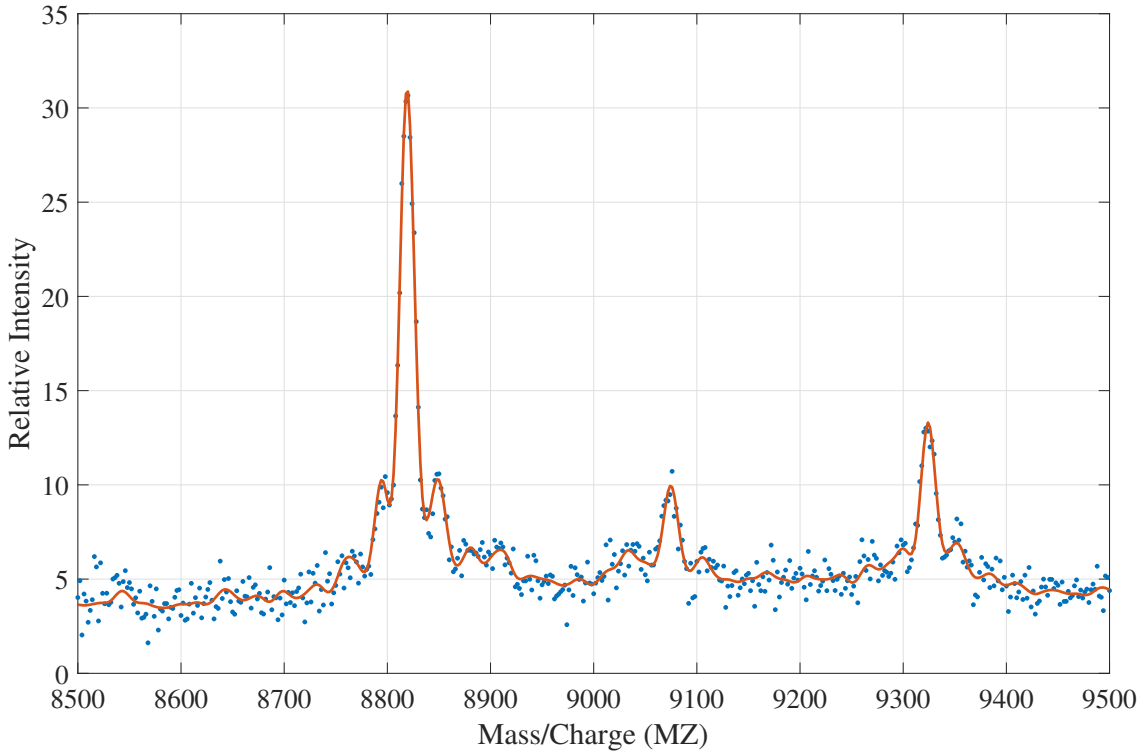

Figure 1: Noisy measurements from a subsection of a (simulated) mass spectrometry reading. The "true" (unknown) measurements are shown in orange, and the noisy measurements are shown in blue.

Question 3 (14 marks)

Data Smoothing

Data "smoothing" is a very common problem in data science and statistics. We are often interested in examining the unknown relationship between a dependent variable (y) and an independent variable (x), under the assumption that the dependent variable has been imperfectly measured and has been contaminated by measurement noise. The model of reality that we use is

$$y = f(x) + \varepsilon$$

where $f(x)$ is some unknown, "true", potentially non-linear function of x, and $\varepsilon \sim N(0, \sigma^2)$ is a random disturbance or error. This is called the problem of function estimation, and the process of estimating $f(x)$ from the noisy measurements y is sometimes called "smoothing the data" (even if the resulting curve is not "smooth" in a traditional sense, it is less rough than the original data).

In this question you will use the k-nearest neighbours machine learning technique to smooth data. This technique is used frequently in practice (think for example the 14-day rolling averages used to estimate coronavirus infection numbers). This question will explore its effectiveness as a smoothing tool.

Mass Spectrometry Data Smoothing

The file `ms.measured.2022.csv` contains $n = 501$ measurements from a mass spectrometer. Mass spectrometry is a chemical analysis tool that provides a measure of the physical composition of a material. The outputs of a mass spectrometry reading are the intensities of various ions, indexed by their mass-to-charge ratio. The resulting spectrum usually consists of a number of relatively sharp peaks that indicate a concentration of particular ions, along with an overall background level. A standard problem is that the measurement process is generally affected by noise – that is, the sensor readings are imprecise and corrupted by measurement noise. Therefore, smoothing, or removing the noise is crucial as it allows us to get a more accurate idea of the true spectrum, as well as determine the relative quantity of the ions more accurately. However, we would also *ideally* like for our smoothing procedure to not damage the important information contained in the spectrum (i.e., the heights of the peaks).

The file `ms.truth.2022.csv` contains measurements of our mass spectrometry reading. The column `ms.measured.2022$MZ` are the mass-to-charge ratios of various ions, and `ms.measured.2022$intensity` are the measured (noisy) intensities of these ions in our material. The file `ms.truth.2022.csv` contains the same $n = 501$ values of `MZ` along with the "true" intensity values (i.e., without added measurement noise), stored in `ms.truth.2022$intensity`. These true values have been found by using several advanced statistical techniques to smooth the data, and are being used here to see how close your estimated spectrum is to the truth. For reference, the samples `ms.measured.2022$intensity` and the value of the true spectrum `ms.truth.2022$intensity` are plotted in Figure 1 against their respective `MZ` values.

To answer this question, you must use the `kknn` and `boot` packages that we used in Studios 9 and 10. You will be using the k-nearest neighbours method (k-NN) to estimate the underlying spectrum from the training data. Use the `kknn` package we examined in Studio 9 to provide predictions for the MZ values in `ms.truth.2022`, using `ms.measured.2022` as the training data. You should use the `kernel = "optimal"` option when calling the `kknn()` function. This means that the predictions are formed by a weighted average of the k points nearest to the point we are trying to predict, the weights being determined by how far away the neighbours are from the point we are trying to predict.

Questions

1. For each value of $k = 1, \ldots, 25$, use k-NN to estimate the values of the spectrum associated with the MZ values in `ms.truth.2022$MZ`. Then, compute the root-mean-squared error between your estimates of the spectrum, and the true values in `ms.truth.2022$intensity`. Produce a plot of these errors against the various values of k. **[1 mark]**

2. Produce four graphs, each one showing: (i) the training data points (`ms.measured.2022$intensity`), (ii) the true spectrum (`ms.truth.2022$intensity`) and (iii) the estimated spectrum (predicted `intensity` values for the `MZ` values in `ms.truth.2022.csv`) produced by the k-NN method for four different values of k; do this for $k = 2$, $k = 6$, $k = 12$ and $k = 25$. Make sure the information presented in your graphs is clearly readable. **[3 marks]**

3. Discuss, qualitatively (i.e., visually), and quantitatively (in terms of root-mean-squared error against the true spectrum) the effect of varying k on the estimate of the spectrum. **[2 marks]**

4. Do any of the estimated spectra plotted in Q3.2 achieve our dual aims of providing a smooth, low-noise estimate of background level as well as accurate estimation of the heights of the peaks? Explain why you think the k-NN method is able to achieve, or not achieve, this aim. **[2 marks]**

5. Use the cross-validation functionality in the `kknn` package to select an estimate of the best value of k (make sure you still use the `optimal` kernel). What value of k does the method select?

How does it compare to the (in practice, unknown) value of k that would minimise the actual mean-squared error (as computed in Q3.1)? **[1 mark]**

6. Using the estimate of the spectrum produced in Q3.5 using the value of k selected by cross-validation, and the values in `ms.measured.2022$intensity`, see if you can think of a way to find an estimate of the standard deviation of the sensor/measurement noise that has corrupted our intensity measurements. **[1 mark]**

7. An important task when processing mass spectrometry signals is to locate the peaks, as this gives information on which elements are present in the material we are analysing. From the smoothed signal produced using the value of k found in Q3.5, which value of MZ corresponds to the maximum estimated intensity? **[1 mark]**

8. Using the bootstrap procedure (use at least $5,000$ bootstrap replications), write code to find a confidence interval for the k-nearest neighbours estimate of intensity at a specific MZ value. Use this code to obtain a 95% confidence interval for the estimate of the intensity at the MZ value you determined previously in Question 3.7 (i.e., the value corresponding to the highest intensity). Compute confidence intervals using the k determined in Q3.5, as well as $k = 3$ neighbours and $k = 20$ neighbours. Report these confidence intervals. Explain why you think these confidence intervals vary in size for different values of k. **[3 marks]**