THE UNIVERSITY OF AUCKLAND

TERM TEST - SEMESTER 2, 2019
Campus: City

STATISTICS

Data Science Practice

(Time allowed: 50 Minutes)

INSTRUCTIONS

- Attempt ALL questions.
- Total marks are 40.

1. **[10 marks]**

This question makes use of the R data frame `trips`, which is shown below.

```
> trips
```

```
      type duration distance hour day month year
1  scooter      187      308   18   0     7 2018
2  scooter      822     1828   20   4     7 2018
3  scooter      221      646   23   0     7 2018
4  scooter      299      626   20   0     7 2018
5  scooter      636     2612   11   0     7 2018
6  scooter      283      278   13   1     7 2018
7  scooter     2213     3351   16   0     7 2018
8  bicycle     2276     5601   15   6     7 2018
9  scooter      349      565   19   1     7 2018
10 scooter      758     1373   16   6     7 2018
```

(a) **Write an R function**, `testError()`, to perform the following steps:

(i) Randomly select *one* row of the data frame `trips` to act as a test set. The remainder of the data frame (nine rows) will act as a training set.

(ii) Fit a simple linear regression to predict `duration` from `distance` using the training set.

(iii) Use the fitted model to predict `duration` for the test set.

(iv) Calculate (and return) the squared difference between the prediction and the real `duration` in the test set.

Your function would be used like this:

```
> testError()
[1] 22399.27
```

[7 marks]

(b) **Explain** what the following R code is doing.

```
> sqrt(mean(sapply(1:100, function(i) testError())))
```

[3 marks]

2. **[10 marks]**

(a) **Explain** what the following shell code is doing **and write down the result** of running the code.

```
head -1 trips.csv > subset.csv
grep scooter trips.csv >> subset.csv
wc -l subset.csv
```

The contents of the CSV file `"trips.csv"` is shown below.

```
"type","duration","distance","hour","day","month","year"
"scooter",187,308,18,0,7,2018
"scooter",822,1828,20,4,7,2018
"scooter",221,646,23,0,7,2018
"scooter",299,626,20,0,7,2018
"scooter",636,2612,11,0,7,2018
"scooter",283,278,13,1,7,2018
"scooter",2213,3351,16,0,7,2018
"bicycle",2276,5601,15,6,7,2018
"scooter",349,565,19,1,7,2018
"scooter",758,1373,16,6,7,2018
```

[5 marks]

(b) **Explain the meaning** of the following `Makefile`. What is the purpose of each line of code?

```
report.html: report.Rmd
        Rscript -e "rmarkdown::render(\"report.Rmd\")"
```

Describe the result of running the following shell code (assuming that the `Makefile` shown above is in the current directory *and* there is also a file `report.Rmd` in the current directory).

```
touch report.Rmd
make
make
```

The content of the file `report.Rmd` is shown below.

````
# A report

```{r}
mean(read.csv("trips.csv")$distance)
```
````

[5 marks]

3. **[10 marks]**

(a) **Explain the meaning** of the following XQuery expression. What is the purpose of each line of code?

```
<months>
{
  for $i in doc("pets.xml")//row/row
    let $n := number($i/pets_adopted)
    where $n < 200
    order by $n
    return $i/month
}
</months>
```

[5 marks]

(b) Given the following XML document, `"pets.xml"`, **write down the result** of evaluating the XQuery expression above.

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <row>
    <row _uuid="00000000-0000-0000-AF9A-401551B08E58">
      <month>Jan</month>
      <pets_adopted>129</pets_adopted>
    </row>
    <row _uuid="00000000-0000-0000-F7B9-E37345BC66E7">
      <month>Mar</month>
      <pets_adopted>126</pets_adopted>
    </row>
    <row _uuid="00000000-0000-0000-ADAB-310B0A2E551C">
      <month>Feb</month>
      <pets_adopted>151</pets_adopted>
    </row>
    <row _uuid="00000000-0000-0000-D539-79AF5550719D">
      <month>Apr</month>
      <pets_adopted>128</pets_adopted>
    </row>
    <row _uuid="00000000-0000-0000-0CF1-7C7A0DE7534B">
      <month>May</month>
      <pets_adopted>143</pets_adopted>
    </row>
  </row>
</response>
```

[5 marks]

4. **[10 marks]**

This question relates to the the JSON file, `"luke.json"`, shown below.

```
{
        "name": "Luke Skywalker",
        "height": "172",
        "mass": "77",
        "hair_color": "blond",
        "skin_color": "fair",
        "eye_color": "blue",
        "gender": "male",
        "homeworld": "https://swapi.co/api/planets/1/",
        "films": [
                "https://swapi.co/api/films/2/",
                "https://swapi.co/api/films/6/",
                "https://swapi.co/api/films/3/",
                "https://swapi.co/api/films/1/",
                "https://swapi.co/api/films/7/"
        ]
}
```

(a) **Write down the result** of the following R code.

```
> library(jsonlite)
> fromJSON(readLines("luke.json"))
```

[3 marks]

(b) A MongoDB database called `starwars` contains a large number of documents for all of the characters in the Star Wars universe. Each record in the database has the same structure as the file `"luke.json"`.

Write R code to query the `starwars` database and extract the `name`, `height`, and `mass` for the first 5 records with `gender` equal to `male`.

The output of your code would look like this:

```
                name height mass
1    Luke Skywalker    172   77
2       Darth Vader    202  136
3         Owen Lars    178  120
4 Biggs Darklighter    183   84
5    Obi-Wan Kenobi    182   77
```

[7 marks]