Week 2 Assignment Report
Student Name/ Number: Gabriel Wee Kiat Lim/ 180703634

Part A

1)

The experiment looks at the effect of increasing nicotine concentration on the nervous system. From the input data plot of the response voltage of the control and the three concentrations of nicotine, it appears that increase in nicotine concentration reduces the activation threshold required to excite the axons.

Predict function:

The predict function sets down a gate parameter of sorts.

The inputs are stimulation voltage, which is a list random generated uniform variate between the range of -100 to 40 mV, and activation threshold which is the set by nicotine level infused into the axon the control used.

Activation initialises excitation voltages to -70 mV and this value is replicated to the number of values of the stimulation_voltage input

When the input stimulation_voltage is more than -70, the axons will fire and the activation excitation will be converted to output 40mV. If the stimulation_voltage is less than or equal to -70, the activation excitation output will remain at -70

Calculate_errors function:

This function generates the sum of squares residuals from inputs of observed values from controls, and predicted values generated by the model.

Fit_threshold function:

This function takes input of input_value_stimulation, input_values_response and threshold and combines the predict and calculate_errors functions created above to output the sum of square residuals (fit errors)

2)

The data is modelled on a regression model where the input stimulation voltage is the explanatory variable and the excitation response voltage is the response variable. It is based on the least squares method which minimises the sum of squares deviation between the observed and predicted values to establish a relationship between these sets of data. [1]

3)

Degrees of freedom present in model =1

Degree of freedom present in residuals = 14

4)

When the code is executed, fitted threshold generates a random univariate value between -100 to 40. The code is looped 20 times to generate a set of threshold and errors values till the minimal error value is obtained, which will provide a best fit model.

5)

```
for(i in 1:20){
  new_threshold = runif(1,-100,40)
  new_error     =
fit_threshold(stimulation_potential,response_voltage_control,new_threshold)
  if(new_error < error){
    fitted_threshold = new_threshold
    error = new_error
  }
}
```

The model is optimised by the loop above by generating and storing a new error and fitted threshold value whenever the error is less than the current/last stored error. At the end of the loop run, the best fit values of error and fitted threshold is being output.

6) How could you improve the optimisation?

The optimisation can be improved by increasing the number of iterations of the loop from 20 to 100.

A proposal mechanism can be introduced to progressively narrow down the spread of random values generated for the threshold.

7)

a)

See “Week 2 Assignment markdown”

b)

Treatment	Control	Conc 1	Conc 2	Conc 3
Mean Threshold Voltage	-54.5	-63.9	-74.9	-82.3

A t-test is done for control with each nicotine treatment, as the p-values are all below 0.05, the difference in the mean of the threshold voltages for each nicotine treatment is significant.

The p values are:

Control with Conc_1 : 1.333e-12, df=198

Control with Conc_2 : < 2.2e-16, df=196

Control with Conc_3 : < 2.2e-16, df=198

c) what is the null hypothesis in this case?

The null hypothesis is that there is no difference in the activation threshold of the control and the three nicotine concentrations, and thus nicotine concentrations will not have an effect on the activation threshold.

PART B

1.

First build an image into a docker container , push it into a repository on Docker Hub.
Pull the container from Docker Hub into apocrita, then pull the image. An image file will be created, which you can then run on the grid, by writing a job script file. Make sure the file has module load singularity, relevant file name and parameters. Then submitting by qsub to the queue to run on the grid.

2.

```
#$ -cwd
#$ -S /bin/bash
#$ -j y
#$ -t 1-42:1 #run as an array with 42 replicates
#$ -pe smp 1
#$ -l h_vmem=2G

echo ${SGE_TASK_ID}

module load singularity
singularity run -B /usr/local/src/myscripts:
/usr/local/src/myscripts/output activationContainer.img input_data.csv -
p[integer]
```

References

1. Grafen, A; Hails, R (2002). *Modern statistics for the life sciences*. London: Oxford University Press.