CSC591: Foundations of Data Science
HW2-3-R-project: Combined R mini-projects for topics spanning HW2 and HW3.

Released: **10/020/18**
Due: **(23:55pm);** (One day late: -25%; -100% after that).

Student Name:
Student ID:

Notes – Read carefully

Submit Single zip file (Name it studentID_hw23_R.zip); follow naming convention for all files. Submit code with reasonable documentation; if needed individuals will be asked to run their code (no modifications are allowed after submission). **Don't change the folder structure given to you.**

- Your solution zip file should include only the "code" folder. (don't include "data" folder – due to big datasets).
    - Folder name 'code' should contain the following:
        - One .txt file, showing instructions on how to run R programs; libraries used, etc.; Also answer any question specific items (for example, if question asks you to submit a plot, then include it here). **Call it Readme.txt**
        - **installPackages.R:** (provided to you). In the first line, add all the R packages your code will need.
        - One R file that includes functions for each question. **Call it** studentid_hw23**.R. Fill in the missing parts of the functions in the template**(utils.R). Don't change the function format or code outside the functions. We will call the functions in this file to evaluate your results. **Never include "rm(list=ls(all=T))" in your code.**
        - One PDF file that includes plots and corresponding explanation for each question. Name it **studentID_hw23.pdf**

Q1. Using "hw23R-linear.txt" data (from the book) answer (a) and (b) (2x5 = 10 points) (see https://cran.r-project.org/web/packages/ISLR/ISLR.pdf)

(a) Fit simple linear regression (separately) for each covariate. Provide scatter plots with fitted regression line. Which covariate provides best prediction?

Note: Implement the function SLR() in studentid_hw23.R with **"mpg"** as your response variable, and **return the best covariate name(return type: string, ["cylinders", "displacement", "horsepower", "weight", "acceleration", "year", "origin"]).** Put the plots in the pdf file.

(b) Fit multiple linear regression model for the data (mpg is still the response variable). Show resulting equation. How do you compare the β's obtained with this model with corresponding β's found in (a), put your answer in the pdf file.

Note: Implement the function MLR() in studentid_hw23.R and **return the coefficients of the model(return type: list, containing coefficients for the intercept and 7 covariates).**

Q2. Fit logistic regression model for the dataset (hw23R-logistic.txt). Note that by ignoring the 'year' column, this dataset contains 6 covariates, therefore you should use multiple logistic regression which is straightforward generalization of simple logistic regression. Note that "Direction" attribute is your class label. (recall simple linear regression vs. multiple linear regression; read documentation of the function you are using in R) (5 points)

Note: Implement the function LogisticRegression() in studentid_hw23.R and **return the coefficients of the model(return type: list, containing 7 coefficients for the intercept and 6 covariates in the following order: [intercept, lag1, lag2, .., lag5, volume]). Also note the training accuracy in your pdf.**

Q3. Apply your data science skills to **improve** the model fitted in Q2. In what sense your improved model is **better** than the model found in (Q2). [Note the term "improve"; that is, you still have to use multiple logistic model only]. Show your work. (5 points)

Note: Implement the function LogisticRegressionImproved() in studentid_hw23.R and **return the training accuracy(return type: float, in range [0, 1]). Also, explain your work clearly in the pdf file.**

Q4. Big data (is everywhere now). The supplied file "slr-90m-data.csv" contain 90 million (x,y) data points. Regular lm() model typically fails on this data unless you have 16gb+ memory. So you objective is to explore alternative scalable packages to fit slr to this data, and answer the following questions:

(a) Find linear regression fit to this data (by using other than lm() or glm() functions).

Note: Implement the function BigSLR() in studentid_hw23.R and **return the coefficients**

of the model(return type: list, containing two coefficients for the intercept and the covariate)

(b) In general, how do you deal with such big data problems? One simple solution is to generate samples. Now generate 5 datasets consists of 1%, 2%, 3%, 4%, 5% random samples from the original data. Fit lm() to each data set. Plot regression lines (all five; use different colors and **provide legend**) on top of regression plot fitted to all data (in (a)). Comment on the quality of regression models generated from samples. Please use set.seed(123) otherwise random samples may change each time you run your code.

Note: Including the five resulting regression equations and the regression plot in the pdf file. Also, put the comments on the quality of regression models in the pdf file.

Q5. Implement a function ZTest to perform a simple z-test automatically on the data. Given the following parameters:

INPUTS:

x type: vector (input dataset)

test_type type: string [‘left-tailed’, ‘right-tailed’, ‘two-tailed’] (indicates left tailed/right tailed/two-tailed test)

alpha type: float (alpha value)

pop_mean type: float (population mean)

pop_sd type: float (population standard deviation)

OUTPUTS:

return a string: ‘reject’ to reject the null hypothesis, ‘not-reject’ to not reject the null hypothesis

What you are allowed to use: mean, sd, sqrt, qnorm

Not allowed to use: predefined hypothesis test functions

Example function call:

```
ZTest(x=c(50, 95, 120, 85, 45, 90, 70, 60, 70, 50, 40, 80, 70,
90, 75, 60, 90, 90, 75, 85, 80, 60, 110, 65, 80, 85, 85, 45,
60, 95, 110, 70, 75, 55, 80, 55),
test_type='left-tailed',alpha=0.1, pop_mean=80, pop_sd=19.2)
```

Q6. Implement an R program to demonstrate CLT. At the minimum, your project should implement the following elements.
(A) Should take at least these inputs:

(1) type of population distribution (e.g., uniform, normal, etc),
(2) sample size (~30),
(3) number of samples (>100).

Note: Implement the function CLT() (necessary arguments have been added in the code, their naming is self explanatory) in studentid_hw23.R to demonstrate the above question and **return the mean and standard error of the mean(return type: list, containing two variables mean and se)**. Use default settings when generating samples. Put the plot of the sampling distribution in the pdf file.

Additional instructions:

1. DONOT add rm(list=ls()) function in your code. This will mess with TA's autograding script.
2. DONOT call the functions you defined in your studentid_hw23.R file in the same file - you can call them separately in another file (which you shouldn't submit). TA will call your functions from another file.