CSC591: Foundations of Data Science
HW1: R Mini Project

Released: 9/06/18
Due: 9/17/18 (23:55pm). (One day late: -25%; -100% after that).

Instructor: Dr. Ranga Raju Vatsavai

Notes

- Submission filename: lastname_StudentID.R (last name should be all small characters).
- This h/w is worth 1% of total grade.
- All submissions must be through Moodle (you can email to TA with cc to Instructor – only if these is a problem – if not received on time, then standard late submission rules apply.)
- No makeups or bonus; for regarding policies, refer to syllabus and 1st day lecture slides.
- You are encouraged to do research, study online materials; discuss with fellow students; BUT ANSWERS SHOULD BE YOUR OWN. Any kind of copying will result in 0 grade (minimum penalty), serious cases will be referred to appropriate authority.

R Mini Project: Data loading, cleaning and transformation

In this project, you will try and explore the principles of data loading, cleaning, transformation and visualization. We will be using 'train.csv' from the Titanic Dataset (hosted on Kaggle.com -> you will need to create an account if you don't have one and download the data).

Link to data: https://www.kaggle.com/c/titanic/data

Once you download your data, perform the following steps (entire code for all the questions together should be submitted in the format specified above).

- Please note, that if you do this right, each of these sub-questions do not need more than 2-3 lines of code (absolutely no need of any loops).

Questions:

(1) **(1 point) Data Loading:** Load the data from train.csv into R. Store this in a variable named data.df
    a. **NOTE:** When submitting your solution, please specify path to this file as just 'train.csv' DONOT specify paths to files on your own machines. Remember, the TA has no access to files on your own machine.
(2) **(1 point) Data Summarization:** Count the number of rows and columns in the data. Save these values in the variables data.df.n_rows and data.df.n_cols respectively.
(3) **(1 point) Data subsetting:** Subset the data to contain only the following columns: PassengerId, Age, Fare and Embarked. Store this in the data frame named data.df.subset
(4) **(3 points) Data cleaning:** You often encounter missing/empty values in your dataset. **(a) For the column 'Age',** in data.df.subset, replace the missing/NA values with the

median.

(b) For the column 'Embarked' in data.df.subset, replace the missing/NA values with mode.

(c) Did you notice any other columns in data.df.subset with missing variables? If yes, use the most appropriate measure (specify why that measure is appropriate in comments) to replace the missing/NA values

At the end of (4), your data.df.subset should not contain any missing/incorrect values.

(5) **(1 point) Data visualization:**
   a. Create a histogram plot based on the 'Age' variable from data.df.subset
   b. Using data.df.subset, create a scatter plot with 'Age' on x-axis and 'Fare' on y-axis.

(6) **(1 point) Anomaly detection:** Recall from the theory part of the homework where we defined an anomaly *as mean + or – 2 std. deviations*. Use that definition here to identify anomalies in the 'Age' variable. Store the corresponding PassengerId values in the variable(vector) 'anomalous_indices'.

(7) (1 point) Data subsetting part 2: Subset/Filter the data.df.subset with the following conditions:
   a. Age >=25 and Age <=80
   b. contain only the following columns: Age, Fare and Embarked.

Store the outcome after (a) and (b) in the data frame named data.df.subset.v2

(8) **(1 point) Data transformation:** Rescale the column 'Fare' in data.df.subset.v2 into the 0-100 range. Store it as a new column named 'Fare_Rescaled' in data.df.subset.v2

IMPORTANT INSTRUCTIONS:

- Submit your solutions in lastname_StudentID.R (last name should be all small characters) file as instructed above
- You should only submit the R file (and a readme if necessary as instructed below). No other files will be entertained.
- DO NOT include the following in your script:
  - rm()
  - setwd()
  - install.packages(): If you want the TA to install specific packages, add a readme.txt file with one package per line. TA will take a look and install if necessary.
- Remember, you are submitting a script, not single lines copied from your R Console. TAs will be using R Studio and a custom script to autograde your solutions. Make sure that your code runs without errors when you do source('lastname_StudentID.R') in RStudio.
- You are not allowed to hard code your answers in (i.e. calculate using some other means and just store your final in variables – TA may check with a different subset of the same dataset to see if your results are valid).
- When submitting your solution, please specify path to the file as just 'train.csv' DONOT

specify paths to files on your own machines. Remember, the TA has no access to files on your own machine.

- Most importantly, clearly document/comment every section of your code.