From previous weeks assignment I have learnt how to use ‘read.csv’ and ‘str’ commands. By using ‘read.csv’ R command I have loaded the dataset into Rstudio as below.

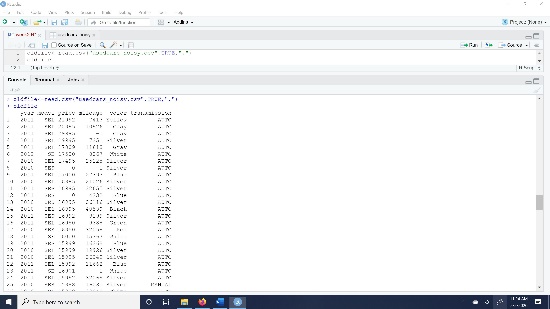
**Syntax to read.csv:**

Oldfile<- read.csv (“usedcars\_noisy.csv”, header = true, ‘,’).

**To display the data in the .csv file:**

Oldfile run this R command in RStudio.

**Implementation in Rstudio:**



**To find the mean value:**

In the price column, I found that the noisy data as ‘0’ price values. To clean up the data I need to replace this noisy data. So, as given in the requirement, I am replacing all the noisy data with mean values of the cars in the dataset. I have found the mean values using R as below:

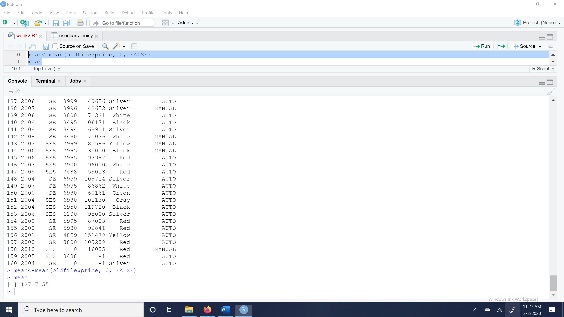
**Syntax to find mean value:**

Mean<- mean (oldfile$price, 0, FALSE)

**To display the value of mean:**

Run Mean in Rstudio.

**Implementation in Rstudio:**



**To find the median value:**

In the Mileage column, I found that the noisy data as ‘-1’ mileage values. To clean up this noisy data I need to replace this ‘-1’ values in mileage column with the median mileage of all cars in the dataset as per the given requirement. To find that median value of mileage of all cars in the dataset we need to use this R command below:

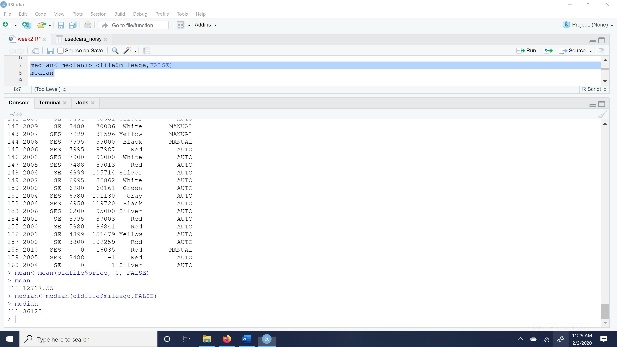
**Syntax to find the median value:**

Median <- median (oldfile$mileage, FALSE)

**To display the value of median:**

Run Median in Rstudio.

**Implementation in Rstudio:**



**To clean the noisy data:**

For cleaning up noisy data in dataset, we need to replace all the ‘0’ values in price column with the mean value of the price column. To do so we need to use the R command shown below:

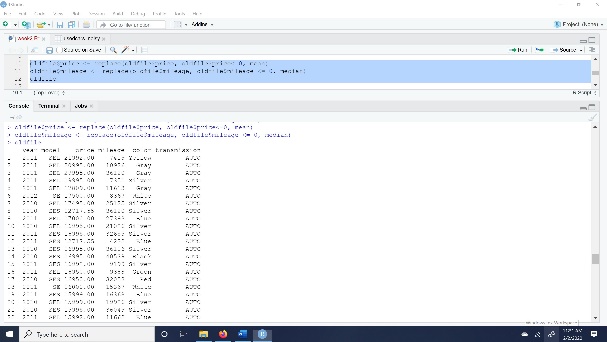
Oldfile$price <- replace (oldfile$price, Oldfile$price <= 0, mean)

In the same way as followed above we can replace all the ‘-1’ values in mileage column with median value of mileage column. To do so we need to use the following R command:

Oldfile$mileage <- replace (oldfile$mileage, Oldfile$mileage <= 0, median)

After executing the above R commands, we need to check if the values of both columns have replaced or not, to do so we need to run this R command ‘oldfile’ in R studio. Then we get see the cleaned up noisy data in the console.

**Implementation in Rstudio:**



**To save the cleaned-up data in a file:**

We need to follow the below R command to save the new cleaned file.

**Syntax to save a file:**

Write.csv (df, file= “define the path of the file where it needed to be saved”)

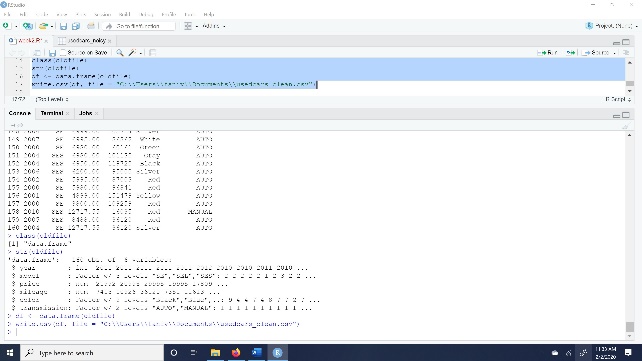
We define the data frame as below:

df<- data.frame (oldfile)

Write.csv (df, file= “C:\\Users\\faniv\\Documents\\usedcars\_clean.csv”)

Now we use str(oldfile) command in Rstudio to know the structure of the file. After following all these steps above, we can save the file in the local computer. Then the file named “usedcars\_clean.csv” has saved.

**Implementation in Rstudio:**



Finally, to summarize my experience on cleaning data by using R commands. It has been easy for the three fourth part i.e., till replacing the values, after that to save that cleaned data into new file is bit challenging i.e., finding the ‘df’ in ‘write.csv’. But finally, I cracked it. After executing this assignment, I have learned that, I don’t think there will be any challenges to face this sort of dataset either it is massive or a small dataset.