A Step-By-Step Guide for Running a Complete Multilinear Regression Analysis in R

The point of this guide is to give new data scientists a step-by-step approach running a complete multilinear regression (MLE) analysis without needing a deep background in statistics. Just note that you should have a basic understanding of linear regression. At each step, I try and explain the theory (when necessary), concepts, and possible difficulties and pain points that may arise in your own data set. I recommend R for regression analysis over Python due to its simplicity (one-liners) and ease is plotting the required graphs. That being said, the same analysis can be completed in Python using a library like statsmodels. I break the analysis into five separate steps:

1. A brief description of the data
2. Checking for multicollinearity
3. Performing a risidual analysis
4. Variable selection
5. Model validation

By the end, if the model has some explanatory power (explains the variance), it should be good to use for inference and prediction. For this guide, I am using a sample data set suitable for regression analysis available from Kaggle called Real Estate Price Predication (https://www.kaggle.com/quantbruce/real-estate-price-prediction). Obviously, for a domain like this, we are looking to predict and infer the price of a house.

A Brief Description of the Data

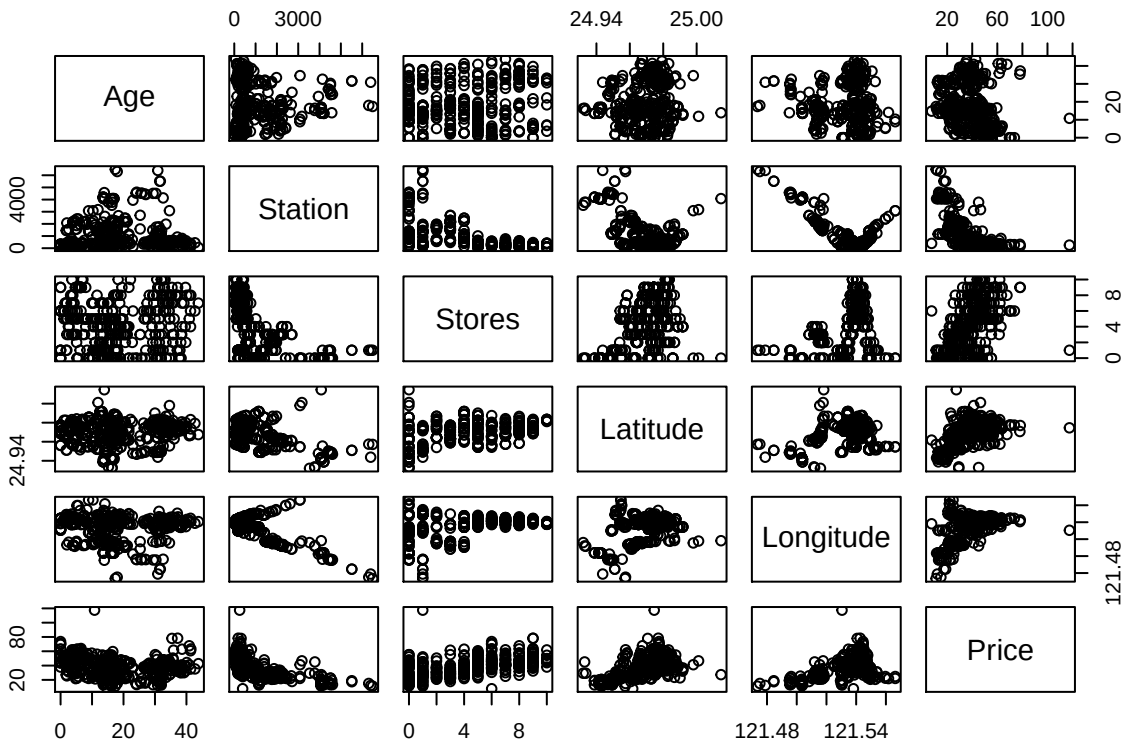
You should already be familiar with the process of going through descriptive statistics in order better understand the data. However, there is one very useful plot for regression analysis that should be incorporated. This is the pairs plot.

```
library(data.table)

# First read in the data
df <- read.csv("real_estate.csv")
df$No <- NULL

# Quick cleanup, rename the column headers
setnames(df,
         old = c('X1.transaction.date',
                 'X2.house.age',
                 'X3.distance.to.the.nearest.MRT.station',
                 'X4.number.of.convenience.stores',
                 'X5.latitude',
                 'X6.longitude',
                 'Y.house.price.of.unit.area'),
         new = c('Transaction',
                 'Age',
                 'Station',
                 'Stores',
                 'Latitude',
                 'Longitude',
                 'Price')
         )
```

```
#Pairs plot
pairs(df[,c(2:7)])
```



The pairs plot summarizes the relationships between all our variables. If you haven't encountered a pairs plot before, all you have to do is start at the diagonal, and line up the intersection between any two variables. You only need to look at the pairs on one side of the diagonal, since the other side will be mirrored. Through the pairs plot, we can get general feeling for:

1. Any colinearity between predictors
2. Predictors that have a weak relationship and may be phased out in variable selection
3. The general strength of the relationship between the response and the predictors

In our example, we can tell that there is probably some overall linear relationship between price and the predictors. The number of stores appears to have the strongest relationship, so we can expect it to remain in the model and have a significant linear relationship. If you get a data set that shows no relationship on that bottom row, and you are building a model for predication, now might be a good time to stop and move on to some data that is more useful. There are some interesting relationships between latitude (lat), longitude (lon), and the station (distance). One likely possibility is that the lat and lon near the mean have a low distance to MRT stations because those coordinates are in the city center. It is not clear whether this will cause an issue for the model. However, if you see near diagonal set of data points for any two predictors, multicollinearity will definitely be an issue that should be resolved.

```
write.csv(df,'real_estate2.csv', row.names = FALSE)
```