# Problem Set 2 - 6203

*Ardalan Mahdavieh*

## 1. Linear Regression 1 [Applied]

i.

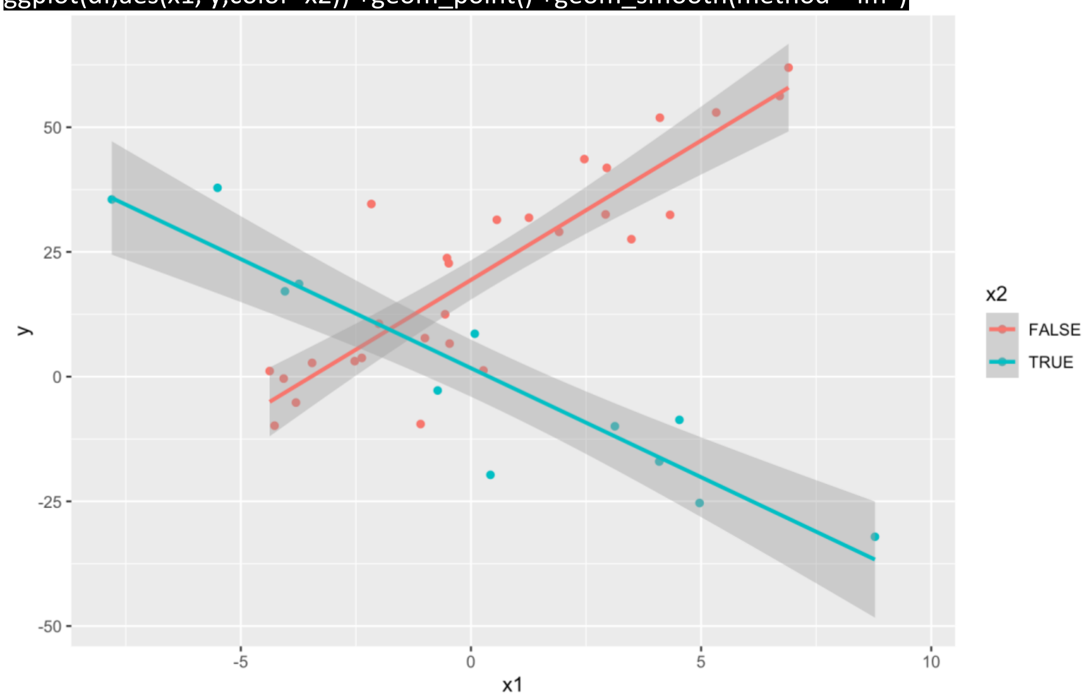
```
df <-read.csv("~/Downloads/PS2_EX1_Data Set.csv")
library(ggplot2)
fit <-lm(y ~x1 +x2, df)
summary(fit)
```

```
> summary(fit)

Call:
lm(formula = y ~ x1 + x2, data = df)

Residuals:
    Min      1Q  Median      3Q     Max 
-39.529 -15.903   0.012  16.100  42.707 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  21.0846     4.1200   5.118  9.8e-06 ***
x1            0.8601     0.9121   0.943  0.35179    
x2TRUE      -21.2042     7.4982  -2.828  0.00752 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 21.73 on 37 degrees of freedom
  (20 observations deleted due to missingness)
Multiple R-squared:  0.1938,	Adjusted R-squared:  0.1502 
F-statistic: 4.446 on 2 and 37 DF,  p-value: 0.0186
```

Adjusted R-squared is 0.1502 which means that this model explains only 15.02% of the variation in the dataset from the two variables. The p-value is very small, 0.0186.

ii.

```
ggplot(df,aes(x1, y,color=x2)) +geom_point() +geom_smooth(method="lm")
```



iii. predict(fit, df[is.na(df$y),],interval="prediction",level=0.95)

```
> predict(fit, df[is.na(df$y),],interval="prediction",level=0.95)
         fit       lwr      upr
41 -5.622415 -53.11907 41.87424
42 21.059512 -23.75861 65.87764
43  2.027875 -43.97468 48.03043
44 -6.836206 -55.08314 41.41073
45  0.509351 -45.32708 46.34578
46 25.046804 -20.44798 70.54159
47  8.175549 -40.76907 57.12017
48 -6.318698 -54.22983 41.59243
49 18.529359 -26.70352 63.76224
50  3.536420 -42.85833 49.93117
51 22.372925 -22.48888 67.23473
52 14.559949 -32.59630 61.71620
53 25.247372 -20.32375 70.81849
54 -4.813607 -51.88396 42.25674
55 -3.247319 -49.66629 43.17166
56 -1.656302 -47.65688 44.34428
57 20.234041 -24.64751 65.11559
58 -3.699678 -50.28275 42.88339
59  3.752380 -42.71675 50.22151
60 19.840260 -25.09618 64.77670
```

Not confident about the predictions because the sample size is too small and therefore the confidence interval is very large.

## 2. Linear Regression 2 [Applied]

i.

```
set.seed(02115)
var1 <-rnorm(1000,0,1)
var2 <-rnorm(1000,0,1)
summary(lm(var1 ~ var2))
```

```
> summary(lm( var1 ~var2))

Call:
lm(formula = var1 ~ var2)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.2702 -0.6878  0.0456  0.6820  3.1209 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.005953   0.031499  -0.189    0.850
var2         0.045793   0.030915   1.481    0.139

Residual standard error: 0.996 on 998 degrees of freedom
Multiple R-squared:  0.002194,	Adjusted R-squared:  0.001194 
F-statistic: 2.194 on 1 and 998 DF,  p-value: 0.1389
```

The multiple R-squared and adjusted R square are very close to being 0. This is due to the fact that the model is unable to make any predictions since var1 and var2 are independent random variables. Moreover no significant coefficient slope was detected as var1 and var2 are independent random variables.

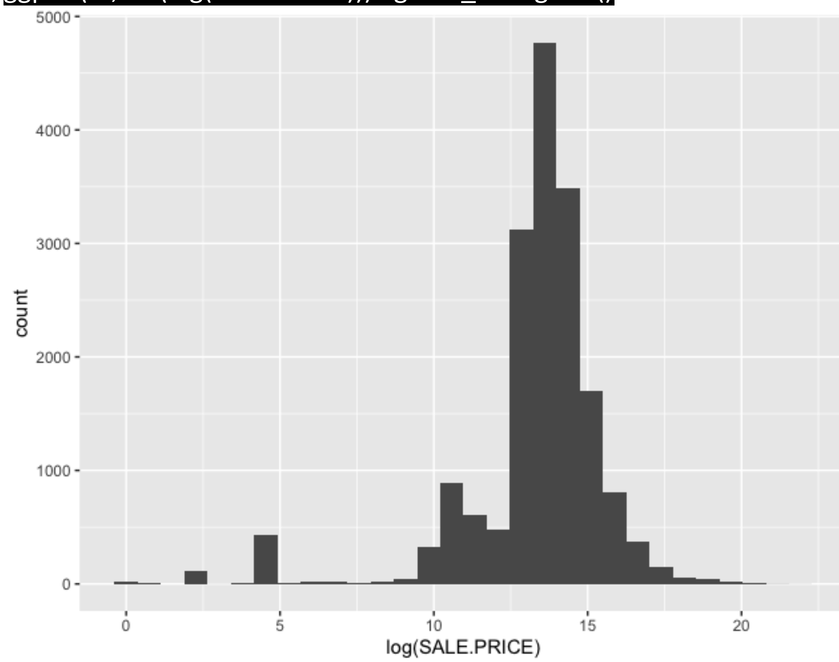
## 3. Linear Regression 3 [Applied]

```
df <-read.csv("~/Downloads/Rolling_Sales_Manhattan_Data Set.csv",na.strings="0")
dim(df)
```

```
> df <-read.csv("~/Downloads/Rolling_Sales_Manhattan_Data Set.csv",na.strings="0")
> dim(df)
[1] 22746    21
```

This data frame contains 22,746 rows and 21 columns.

i. 
```
ggplot(df,aes(log(SALE.PRICE))) +geom_histogram()
```




```
summary(lm(log(SALE.PRICE) ~YEAR.BUILT, df))
```

```
> summary(lm(log(SALE.PRICE) ~YEAR.BUILT, df))

Call:
lm(formula = log(SALE.PRICE) ~ YEAR.BUILT, data = df)

Residuals:
     Min       1Q   Median       3Q      Max 
-14.0121  -0.6730   0.0472   0.9117   8.3013 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 30.3573402  0.8064069   37.65   <2e-16 ***
YEAR.BUILT  -0.0086027  0.0004132  -20.82   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.762 on 15002 degrees of freedom
  (7742 observations deleted due to missingness)
Multiple R-squared:  0.02809,	Adjusted R-squared:  0.02802 
F-statistic: 433.5 on 1 and 15002 DF,  p-value: < 2.2e-16
```

By looking at the t-value, standard error, significance and R-squared valued we can conclude that the log-normal distribution above, indicated that YEAR.BUILT is a good predictor for SALE.PRICE. The coefficients indicate that as they buildings get older, their prices decrease.

ii. ```
df$ZIP.CODE <-factor(df$ZIP.CODE)
summary(lm(log(SALE.PRICE) ~ ZIP.CODE, df))
```

```
Call:
lm(formula = log(SALE.PRICE) ~ ZIP.CODE, data = df)

Residuals:
     Min       1Q   Median       3Q      Max
-14.6437  -0.5980   0.0069   0.6652  10.6562

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    14.3257     0.1261 113.622  < 2e-16 ***
ZIP.CODE10002  -0.8236     0.1667  -4.939 7.92e-07 ***
ZIP.CODE10003  -0.5150     0.1429  -3.604 0.000315 ***
ZIP.CODE10004  -0.4942     0.2319  -2.131 0.033111 *
ZIP.CODE10005  -0.5082     0.1975  -2.573 0.010103 *
ZIP.CODE10006  -0.1673     0.2218  -0.755 0.450549
ZIP.CODE10007   0.4536     0.2171   2.089 0.036695 *
ZIP.CODE10009  -0.5550     0.2056  -2.700 0.006951 **
ZIP.CODE10010  -0.1495     0.1537  -0.972 0.330947
ZIP.CODE10011  -0.2776     0.1407  -1.973 0.048534 *
ZIP.CODE10012   0.3311     0.1762   1.879 0.060223 .
ZIP.CODE10013   0.3180     0.1553   2.048 0.040586 *
ZIP.CODE10014  -0.1239     0.1559  -0.795 0.426762
ZIP.CODE10016  -0.7683     0.1423  -5.401 6.73e-08 ***
ZIP.CODE10017  -0.8333     0.1581  -5.270 1.38e-07 ***
ZIP.CODE10018   0.5268     0.2447   2.153 0.031331 *
ZIP.CODE10019  -3.9885     0.1309 -30.464  < 2e-16 ***
ZIP.CODE10021  -0.2060     0.1439  -1.432 0.152075
ZIP.CODE10022  -0.5637     0.1427  -3.950 7.86e-05 ***
ZIP.CODE10023  -0.5221     0.1386  -3.768 0.000165 ***
ZIP.CODE10024  -0.2684     0.1437  -1.868 0.061813 .
ZIP.CODE10025  -0.5915     0.1456  -4.063 4.87e-05 ***
ZIP.CODE10026  -0.7184     0.1743  -4.122 3.77e-05 ***
ZIP.CODE10027  -1.1323     0.1726  -6.558 5.59e-11 ***
ZIP.CODE10028  -0.3112     0.1515  -2.055 0.039940 *
ZIP.CODE10029  -0.2677     0.1957  -1.368 0.171374
ZIP.CODE10030  -0.8265     0.2118  -3.903 9.54e-05 ***
ZIP.CODE10031  -1.1173     0.1841  -6.070 1.31e-09 ***
ZIP.CODE10032  -0.9386     0.1946  -4.823 1.43e-06 ***
ZIP.CODE10033  -0.9920     0.1916  -5.177 2.28e-07 ***
ZIP.CODE10034  -1.4456     0.2047  -7.063 1.69e-12 ***
ZIP.CODE10035  -0.6195     0.2364  -2.620 0.008803 **
ZIP.CODE10036  -0.2758     0.1714  -1.609 0.107609
ZIP.CODE10037  -1.6487     0.2447  -6.739 1.65e-11 ***
ZIP.CODE10038  -0.1163     0.1815  -0.641 0.521492
ZIP.CODE10039  -1.2118     0.2659  -4.558 5.19e-06 ***
ZIP.CODE10040  -1.1594     0.1968  -5.892 3.89e-09 ***
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.887 on 17473 degrees of freedom
  (5227 observations deleted due to missingness)
Multiple R-squared:  0.3394,    Adjusted R-squared:  0.3377
F-statistic: 199.5 on 45 and 17473 DF,  p-value: < 2.2e-16
```

Based on the analysis above we can conclude that the most affordable location in Manhattan is located within ZIP.CODE 10019. By looking specifically at the
```

neighborhoods we can conclude that the most affordable neighborhood in Manhattan is Midtown West which is also located within the 10019 zip code.

```
summary(lm(log(SALE.PRICE) ~ NEIGHBORHOOD, df))
Residuals:
     Min       1Q   Median       3Q      Max 
-14.3161  -0.6046   0.0063   0.6807  10.5728 

Coefficients:
                                     Estimate Std. Error t value Pr(>|t|)    
(Intercept)                          13.47443    0.19536  68.973  < 2e-16 ***
NEIGHBORHOODCHELSEA                   0.56931    0.20707   2.749 0.005979 ** 
NEIGHBORHOODCHINATOWN                 0.64702    0.29523   2.192 0.028424 *  
NEIGHBORHOODCIVIC CENTER              0.84166    0.23885   3.524 0.000426 ***
NEIGHBORHOODCLINTON                   0.10199    0.23561   0.433 0.665120    
NEIGHBORHOODEAST VILLAGE              0.34343    0.23136   1.484 0.137723    
NEIGHBORHOODFASHION                   1.62697    0.25619   6.351 2.20e-10 ***
NEIGHBORHOODFINANCIAL                 0.52119    0.21367   2.439 0.014731 *  
NEIGHBORHOODFLATIRON                  1.00543    0.22083   4.553 5.33e-06 ***
NEIGHBORHOODGRAMERCY                  0.23493    0.21265   1.105 0.269260    
NEIGHBORHOODGREENWICH VILLAGE-CENTRAL  0.57302    0.21194   2.704 0.006863 ** 
NEIGHBORHOODGREENWICH VILLAGE-WEST    0.69899    0.21447   3.259 0.001119 ** 
NEIGHBORHOODHARLEM-CENTRAL           -0.15500    0.20776  -0.746 0.455662    
NEIGHBORHOODHARLEM-EAST               0.48115    0.25066   1.920 0.054934 .  
NEIGHBORHOODHARLEM-UPPER             -0.21508    0.24903  -0.864 0.387798    
NEIGHBORHOODHARLEM-WEST               0.27071    0.40946   0.661 0.508522    
NEIGHBORHOODINWOOD                   -0.58773    0.25385  -2.315 0.020607 *  
NEIGHBORHOODJAVITS CENTER             0.45601    0.37266   1.224 0.221098    
NEIGHBORHOODKIPS BAY                  0.01689    0.23360   0.072 0.942355    
NEIGHBORHOODLITTLE ITALY              1.33554    0.36548   3.654 0.000259 ***
NEIGHBORHOODLOWER EAST SIDE          -0.03168    0.22650  -0.140 0.888749    
NEIGHBORHOODMANHATTAN VALLEY          0.18618    0.24496   0.760 0.447246    
NEIGHBORHOODMIDTOWN CBD               0.66440    0.25579   2.598 0.009398 ** 
NEIGHBORHOODMIDTOWN EAST              0.09551    0.20428   0.468 0.640100    
NEIGHBORHOODMIDTOWN WEST             -3.05389    0.19854 -15.381  < 2e-16 ***
NEIGHBORHOODMORNINGSIDE HEIGHTS      -0.56962    0.29872  -1.907 0.056551 .  
NEIGHBORHOODMURRAY HILL               0.08654    0.21078   0.411 0.681397    
NEIGHBORHOODSOHO                      1.49779    0.23108   6.482 9.32e-11 ***
NEIGHBORHOODSOUTHBRIDGE               1.12629    0.32451   3.471 0.000520 ***
NEIGHBORHOODTRIBECA                   0.94797    0.22083   4.293 1.77e-05 ***
NEIGHBORHOODUPPER EAST SIDE (59-79)   0.62381    0.20065   3.109 0.001881 ** 
NEIGHBORHOODUPPER EAST SIDE (79-96)   0.45141    0.20203   2.234 0.025471 *  
NEIGHBORHOODUPPER EAST SIDE (96-110)  0.71509    0.28090   2.546 0.010913 *  
NEIGHBORHOODUPPER WEST SIDE (59-79)   0.37124    0.20215   1.836 0.066311 .  
NEIGHBORHOODUPPER WEST SIDE (79-96)   0.53835    0.20595   2.614 0.008957 ** 
NEIGHBORHOODUPPER WEST SIDE (96-116)  0.12068    0.22022   0.548 0.583700    
NEIGHBORHOODWASHINGTON HEIGHTS LOWER -0.06039    0.24470  -0.247 0.805077    
NEIGHBORHOODWASHINGTON HEIGHTS UPPER -0.26169    0.22240  -1.177 0.239340    
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.904 on 17482 degrees of freedom
  (5226 observations deleted due to missingness)
Multiple R-squared:  0.3271,	Adjusted R-squared:  0.3256 
F-statistic: 229.6 on 37 and 17482 DF,  p-value: < 2.2e-16
```

iii. A regression model is appropriate to use because we can study the relationships between SALE.PRICE and other variables in order to analyze our dataset. By looking at the R-squared value we can determine the proportion of the variance between SALE.PRICE and other variables. Moreover by analyzing the coefficients we can understand the scale of how different variables are affecting each other.

iv. By identifying the housing prices within each neighborhood or region (zip codes), the city planning officials will have a better understanding of wealth distribution within the city. Therefore they can allocate their resources more effectively.

4. **Support-Vector Machines [Applied]**

```
?spam
install.packages("kernlab")
install.packages("e1071")
library(e1071)
library(kernlab)
data(spam)
dim(spam)
```

4601 Rows and 58 Columns.

```
head(spam)
```

```
  make address  all num3d  our over remove internet order mail receive will people report addresses
1 0.00    0.64 0.64     0 0.32 0.00   0.00     0.00  0.00 0.00    0.00 0.64   0.00   0.00      0.00
2 0.21    0.28 0.50     0 0.14 0.28   0.21     0.07  0.00 0.94    0.21 0.79   0.65   0.21      0.14
3 0.06    0.00 0.71     0 1.23 0.19   0.19     0.12  0.64 0.25    0.38 0.45   0.12   0.00      1.75
4 0.00    0.00 0.00     0 0.63 0.00   0.31     0.63  0.31 0.63    0.31 0.31   0.31   0.00      0.00
5 0.00    0.00 0.00     0 0.63 0.00   0.31     0.63  0.31 0.63    0.31 0.31   0.31   0.00      0.00
6 0.00    0.00 0.00     0 1.85 0.00   0.00     1.85  0.00 0.00    0.00 0.00   0.00   0.00      0.00
  free business email  you credit your font num000 money hp hpl george num650 lab labs telnet
1 0.32     0.00  1.29 1.93   0.00 0.96    0   0.00  0.00  0   0      0      0   0    0      0
2 0.14     0.07  0.28 3.47   0.00 1.59    0   0.43  0.43  0   0      0      0   0    0      0
3 0.06     0.06  1.03 1.36   0.32 0.51    0   1.16  0.06  0   0      0      0   0    0      0
4 0.31     0.00  0.00 3.18   0.00 0.31    0   0.00  0.00  0   0      0      0   0    0      0
5 0.31     0.00  0.00 3.18   0.00 0.31    0   0.00  0.00  0   0      0      0   0    0      0
6 0.00     0.00  0.00 0.00   0.00 0.00    0   0.00  0.00  0   0      0      0   0    0      0
  num857 data num415 num85 technology num1999 parts pm direct cs meeting original project   re  edu
1      0    0      0     0          0    0.00     0  0   0.00  0       0     0.00       0 0.00 0.00
2      0    0      0     0          0    0.07     0  0   0.00  0       0     0.00       0 0.00 0.00
3      0    0      0     0          0    0.00     0  0   0.06  0       0     0.12       0 0.06 0.06
4      0    0      0     0          0    0.00     0  0   0.00  0       0     0.00       0 0.00 0.00
5      0    0      0     0          0    0.00     0  0   0.00  0       0     0.00       0 0.00 0.00
6      0    0      0     0          0    0.00     0  0   0.00  0       0     0.00       0 0.00 0.00
  table conference charSemicolon charRoundbracket charSquarebracket charExclamation charDollar
1     0          0          0.00            0.000                 0           0.778      0.000
2     0          0          0.00            0.132                 0           0.372      0.180
3     0          0          0.01            0.143                 0           0.276      0.184
4     0          0          0.00            0.137                 0           0.137      0.000
5     0          0          0.00            0.135                 0           0.135      0.000
6     0          0          0.00            0.223                 0           0.000      0.000
  charHash capitalAve capitalLong capitalTotal type
1    0.000      3.756          61          278 spam
2    0.048      5.114         101         1028 spam
3    0.010      9.821         485         2259 spam
4    0.000      3.537          40          191 spam
5    0.000      3.537          40          191 spam
6    0.000      3.000          15           54 spam
```

```
set.seed(02115)
sample <- sample( c(TRUE, FALSE), nrow(spam), replace=TRUE)
train <- spam[sample,]
test <- spam[!sample,]
```

i. svmfit <-svm( type ~.,data=train,kernel="linear",cost=1,scale=FALSE)

```
> svmfit

Call:
svm(formula = type ~ ., data = train, kernel = "linear", cost = 1, scale = FALSE)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  linear
       cost:  1

Number of Support Vectors:  402
```

ii. pred <-predict(svmfit, test)

table(pred)

```
> table(pred)
pred
nonspam    spam
   1222    1043
```

table(Predict=pred,Truth=test$type)

```
> table(Predict=pred,Truth=test$type)
         Truth
Predict   nonspam spam
  nonspam    1169   53
  spam        189  854
```

The model wrongfully classified 189 non-spam emails as spam and 53 spam emails as non-spam. The total classification error is 242.

iii. svmfit <-svm( type ~.,data=train,kernel="linear",cost=.01,scale=FALSE)

pred <-predict(svmfit, test)

table(Predict=pred,Truth=test$type)

```
> table(Predict=pred,Truth=test$type)
         Truth
Predict   nonspam spam
  nonspam    1273  134
  spam         85  773
```

By lowering the cost from 1 to 0.01, the classification accuracy improved. Our total classification error dropped to 219 from 242.

iv. Compared to the regression model, the support vector machines (SVM) are much hard to interpret. By using the regression model we are able to have a better understanding of which dependent and independent variables are important in our predictions and analyses by looking at the provided information (residuals, coefficients, standard error, R-squared). By using the SVM method, I had less control of the prediction models as I was not able to see which variables are factored into the prediction model.