Regression

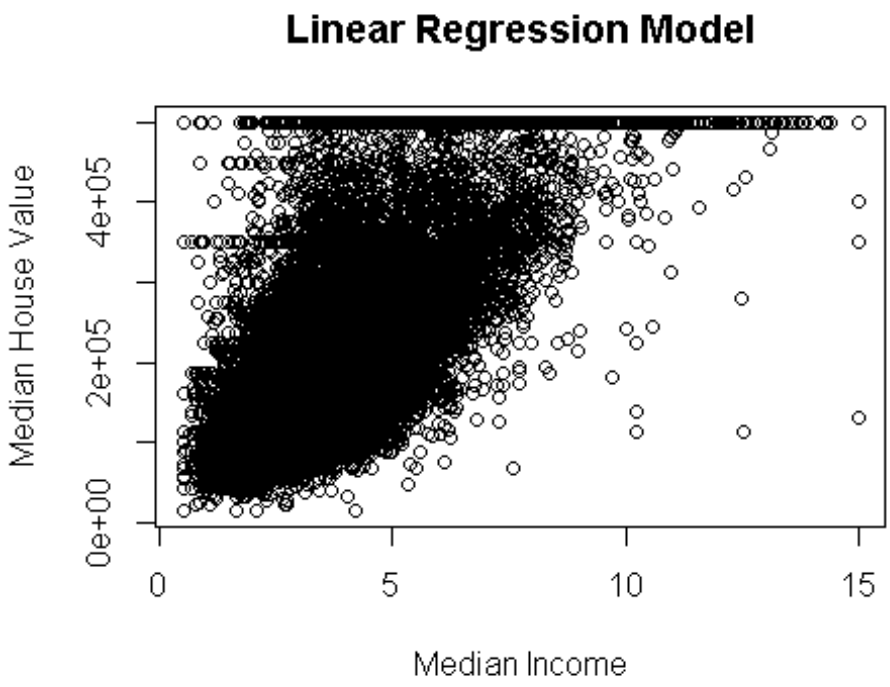
Matthew Naughton

CS 4375.003

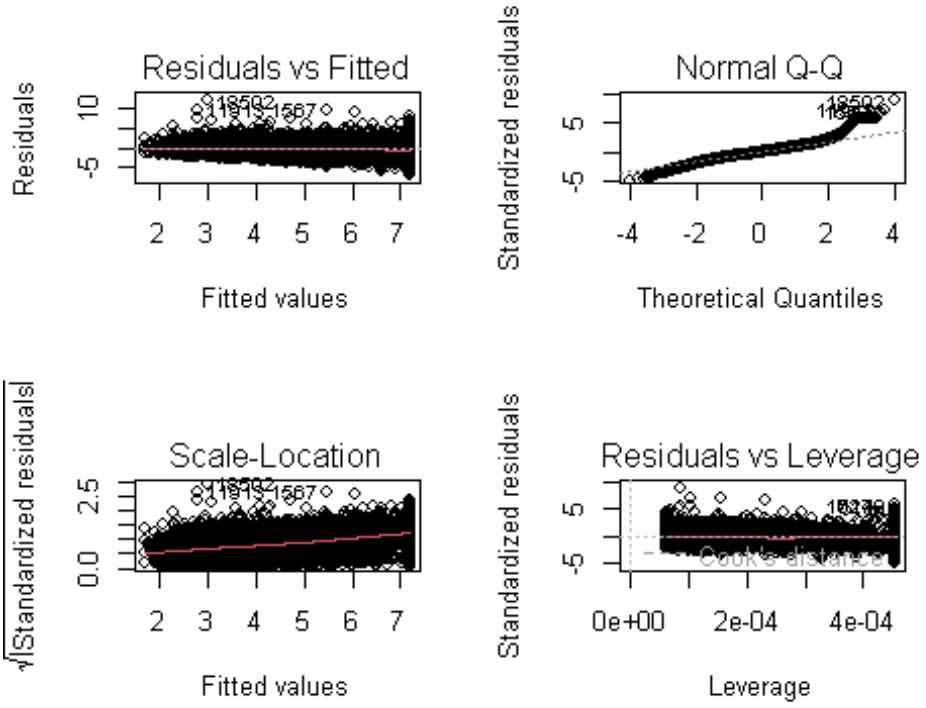
Portfolio: Searching for Similarity

```
Housing <- read.csv(file = 'housing.csv')

Housing <- na.omit(Housing)
#Divide the data into train and test data 80/20
i <- sample(1:nrow(Housing), nrow(Housing)*0.80, replace=FALSE)
train <- Housing[i,]
test <- Housing[-i,]

# Explore the data
head(train)

##       longitude latitude housing_median_age total_rooms total_bedrooms
## 2731    -115.56    32.80                 25        1311            375
## 10672   -117.85    33.62                 18         729            105
## 18979   -122.01    38.25                 16        1081            181
## 7212    -118.17    34.03                 42         882            292
## 3809    -118.46    34.20                 13        2926            816
## 8376    -118.35    33.95                 28        4770           1328
##       population households median_income median_house_value
ocean_proximity
## 2731        1193        351        2.1979              63900
INLAND
## 10672        316        108       10.3893             500001       <1H
OCEAN
## 18979        792        184        4.6779             131300
INLAND
## 7212        1248        281        2.7610             120000       <1H
OCEAN
## 3809        1867        802        3.5255             202700       <1H
OCEAN
## 8376        3201       1196        2.6810             147700       <1H
OCEAN

names(train)

##  [1] "longitude"          "latitude"           "housing_median_age"
##  [4] "total_rooms"        "total_bedrooms"     "population"
##  [7] "households"         "median_income"      "median_house_value"
## [10] "ocean_proximity"

range(train$total_rooms)
```

```
## [1]     2 39320

mean(train$total_rooms, na.rm=TRUE)

## [1] 2644.087

range(train$total_bedrooms, na.rm=TRUE)

## [1]    1 6445

mean(train$total_bedrooms, na.rm=TRUE)

## [1] 539.5609

str(train)

## 'data.frame':    16346 obs. of  10 variables:
##  $ longitude         : num  -116 -118 -122 -118 -118 ...
##  $ latitude          : num  32.8 33.6 38.2 34 34.2 ...
##  $ housing_median_age: num  25 18 16 42 13 28 18 15 15 30 ...
##  $ total_rooms       : num  1311 729 1081 882 2926 ...
##  $ total_bedrooms    : num  375 105 181 292 816 ...
##  $ population        : num  1193 316 792 1248 1867 ...
##  $ households        : num  351 108 184 281 802 ...
##  $ median_income     : num  2.2 10.39 4.68 2.76 3.53 ...
##  $ median_house_value: num  63900 500001 131300 120000 202700 ...
##  $ ocean_proximity   : chr  "INLAND" "<1H OCEAN" "INLAND" "<1H OCEAN" ...
##  - attr(*, "na.action")= 'omit' Named int [1:207] 291 342 539 564 697 739
1098 1351 1457 1494 ...
##   ..- attr(*, "names")= chr [1:207] "291" "342" "539" "564" ...

# Explore the data Graphically
plot(train$median_income, train$median_house_value, xlab="Median Income",
ylab="Median House Value", main="Linear Regression Model")
```




```
# Linear model:
lm1 <- lm(median_income~median_house_value, data=train)
summary(lm1)

##
## Call:
## lm(formula = median_income ~ median_house_value, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.6851 -0.7989 -0.0568  0.7705 11.9765
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        1.542e+00  2.220e-02   69.46   <2e-16 ***
## median_house_value 1.129e-05  9.351e-08  120.70   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.382 on 16344 degrees of freedom
## Multiple R-squared:  0.4713, Adjusted R-squared:  0.4712
## F-statistic: 1.457e+04 on 1 and 16344 DF,  p-value: < 2.2e-16

# Residuals:
par(mfrow=c(2,2))
plot(lm1)
```




```
# other data:
pred1 <- predict(lm1, newdata=test)
cor1 <- cor(pred1, test$median_income)
mse1 <- mean((pred1-test$median_income))
rmse1 <- sqrt(mse1)
print(paste("correlation: ", cor1))
```

```
## [1] "correlation:  0.695799310223036"
```

```
print(paste("mse: ", mse1))
```

```
## [1] "mse:  0.0265313690206695"
```

```
print(paste("rmse: ", rmse1))
```

```
## [1] "rmse:  0.162884526645932"
```

```
# kNN Regression
#install.packages("caret")
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
fit <- knnreg(train[,1:7],train[,1],k=3)
pred2 <- predict(fit, test[,1:7])
cor_knn1 <- cor(pred2, test$median_income)
```

```r
mse_knn1 <- mean((pred2 - test$median_income)^2)
print(paste("cor= ", cor_knn1))
```

```
## [1] "cor=  -0.0719237889009831"
```

```r
print(paste("mse= ", mse_knn1))
```

```
## [1] "mse=  15243.9508082911"
```

```r
#Our values are much worse than linear regression

# Decision Trees

library(tree)
tree1 <- tree(train$median_income~train$median_house_value)
pred3 <- predict(tree1, newdata=test)
```

```
## Warning: 'newdata' had 4087 rows but variables found have 16346 rows
```

```r
cor_tree <- cor(pred3, train$median_income)
print(paste("correlation: ", cor_tree))
```

```
## [1] "correlation:  0.687151457807674"
```

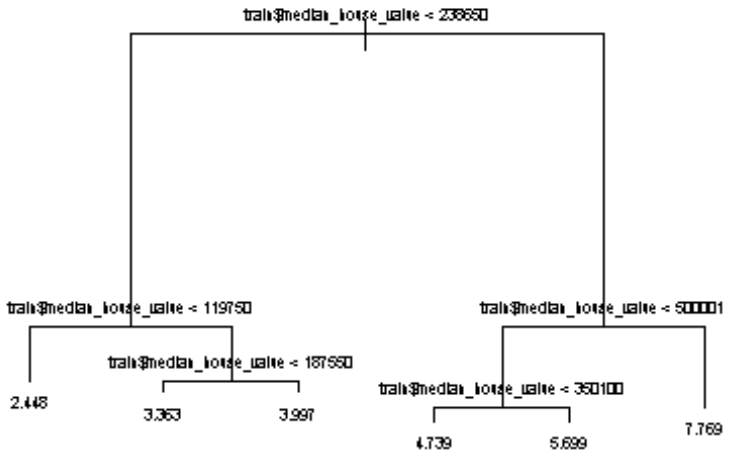
```r
rmse_tree <- sqrt(mean((pred3-test$median_income)^2))
```

```
## Warning in pred3 - test$median_income: longer object length is not a
multiple of
## shorter object length
```

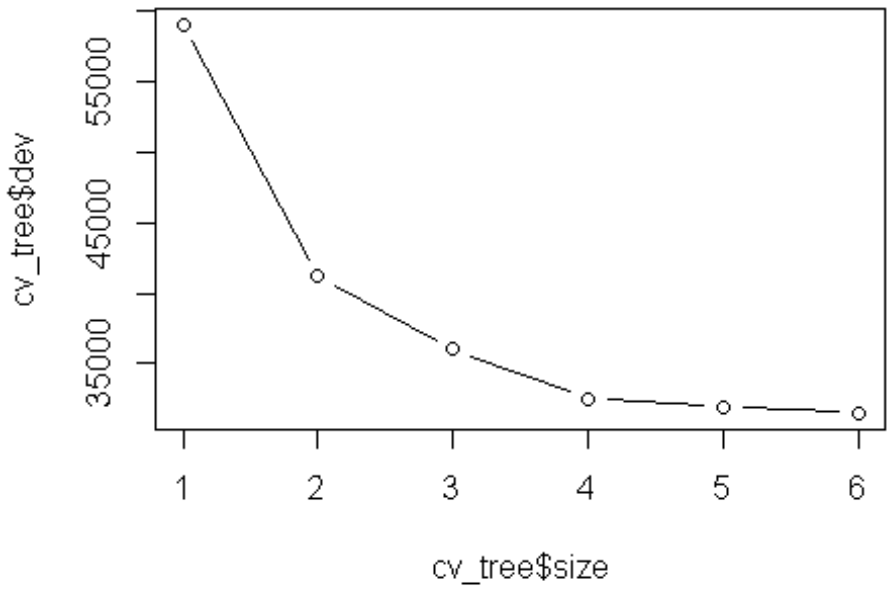
```r
print(paste("rmse: ", rmse_tree))
```

```
## [1] "rmse:  2.28893730869035"
```

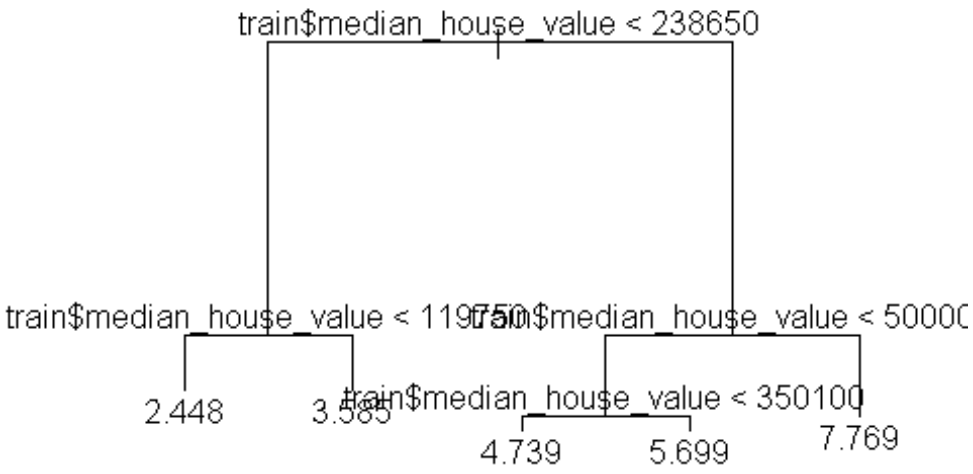
```r
# this data shows us that the decision is closer than the linear regression
model

plot(tree1)
text(tree1, cex=0.5, pretty=0)
```




```r
#cv tree
cv_tree <- cv.tree(tree1)
plot(cv_tree$size, cv_tree$dev, type='b')
```



```r
# pruned tree
tree_pruned <- prune.tree(tree1, best=5)
plot(tree_pruned)
text(tree_pruned, pretty=0)
```




```r
# pruned tree values
pred_pruned <- predict(tree_pruned, newdata=test)

## Warning: 'newdata' had 4087 rows but variables found have 16346 rows

cor_pruned <- cor(pred_pruned, train$median_income)
rmse_pruned <- sqrt(mean((pred_pruned-test$median_income)^2))

## Warning in pred_pruned - test$median_income: longer object length is not a
## multiple of shorter object length

print(paste("cor of pruned tree: ", cor_pruned))

## [1] "cor of pruned tree:  0.67897656669672"

print(paste("rmse of pruned tree: ", rmse_pruned))

## [1] "rmse of pruned tree:  2.2819394457373"

# The data of the pruned tree is worse than the linear regression model, but
not by much
```