Anthony Cerna
November 23rd, 2015
UC Berkeley

Stats 151A: Midterm #2
Predicting the Value of a Boston Home Using Neighborhood Features

I. Introduction

The purpose of this report is to analyze housing data from various census tracts of Boston from the 1970 census. Harrison and Rubinfeld collected the data in 1979 for the purpose of discovering whether or not clean air influenced the value of houses in Boston, Massachusetts. Their paper was titled "Hedonic prices and the demand for clean air" and it can be found in the *Journal of Environmental Economics and Management*, 5, 81–102.

This report attempts to answer the following two questions: which variables mainly determine the median housing price in a tract? How are the prices of houses affected by these different neighborhood characteristics?

II. Data Description

The BostonHousing dataset was obtained from the R library mlbench. It contains housing data from 506 census tracts of Boston, Massachusetts, which was obtained from the 1970 United States census. The data consists of 14 variables (i.e. features) and 506 observations. Each observation is a subdivision of a county in Boston. The 14 variables are:

1. **crim**: per capita crime rate by town
2. **zn**: proportion of residential land zoned for lots over 25,000 sq.ft
3. **indus**: proportion of non-retail business acres per town
4. **chas**: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
5. **nox**: nitric oxides concentration (parts per 10 million)
6. **rm**: average number of rooms per dwelling
7. **age**: proportion of owner-occupied units built prior to 1940
8. **dis**: weighted distances to five Boston employment centres
9. **rad**: index of accessibility to radial highways
10. **tax**: full-value property-tax rate per USD 10,000
11. **ptratio**: pupil-teacher ratio by town
12. **b**: 1000(B - 0.63)^2 where B is the proportion of blacks by town
13. **lstat**: percentage of lower status of the population
14. **medv**: median value of owner occupied homes in USD 1000's

The response variable for this analysis will be **medv**. Therefore, there are a total of 13 dependent variables and 1 dependent (response) variable that will be considered for the initial model.

Prior to fitting a model, the variables were individually analyzed in order to visually identify any potential outliers or issues with our data. To facilitate this task, the density plots of each predictor variables are plotted along with the summary statistics of each variable. Any long tails suggest the presence of extreme values relative to the rest of the group (i.e. potential *outliers*). For example, **crim**, **zn**, **dis**, and **lstat** all have very long right

tails, whereas **b** has a long left tail. The following observations had very large **crim** rates, relative to the rest of the observations: 381, 399, 405, 406, 411, 415, 419, and 428.

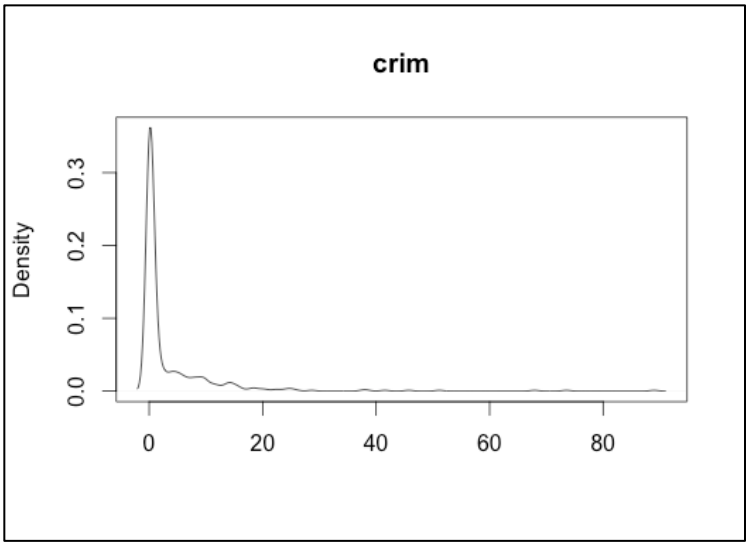


Figure 1: Density Plot for crim

However, it was very difficult to identify a small subset of potential outliers from other long-tailed variables because there were a large number of observations that were spread out on these tails. It was not just a small subset of observations that is causing these tails. This just shows the large amount of skewness for these predictors.

Another important observation from these density plots is that the variables **indus, rad**, and **tax** seem to suggest that there are two potential groups of data. Thinking back to the definition of these variables, it might be possible to split the BostonHousing data into *Urban* (larger proportion of nonretail business and closer access to highways etc.) and *Suburban tracts* and analyze them separately.

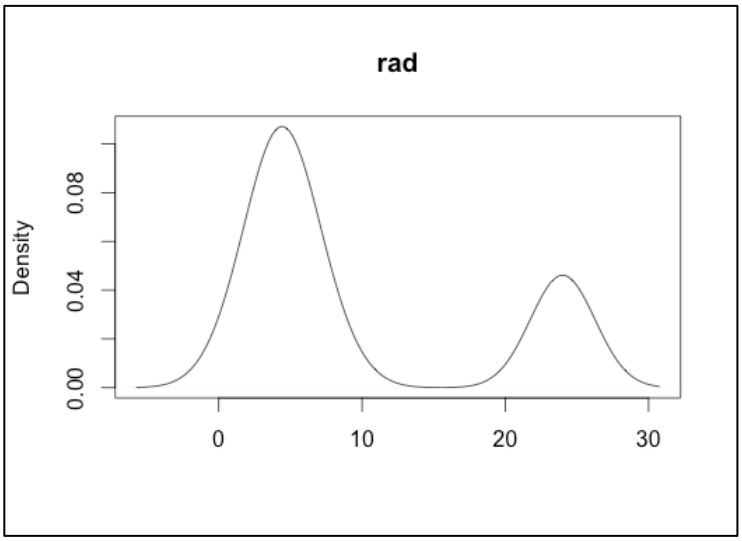


Figure 2 Density Plot for rad

Before creating a model, it is crucial to check how predictors are related with one another and more importantly, with the response variable **medv**. Looking at the last row of the correlation matrix for the predictors, **rm** is identified by its large positive correlation with a value of 0.6954. Intuitively, this makes sense because one can expect that **rm** serves as a proxy for determining the size of a house. Therefore, a larger average number of rooms can imply that the price of the house might be greater because it is a larger house. Also, **ptratio** (pupil-teacher ratio by town) and **lstat** (% of lower status of the population) have relatively large correlation coefficients of -0.5078 and -0.7377 respectively. Again, this does make some intuitive sense because one might expect the price of a house to decrease (i.e. negative correlation) if the area had a large % of lower status residents. Also, low-income areas tend to have larger classroom sizes. These are only preliminary results, so we must continue with caution. Lastly, it is important to note that **lstat** and **medv** do have a negative relationship, however, it shows a very strong non-linear relationship with the response variable, which suggests that a non-linear transformation of this predictor might be useful in our final model.

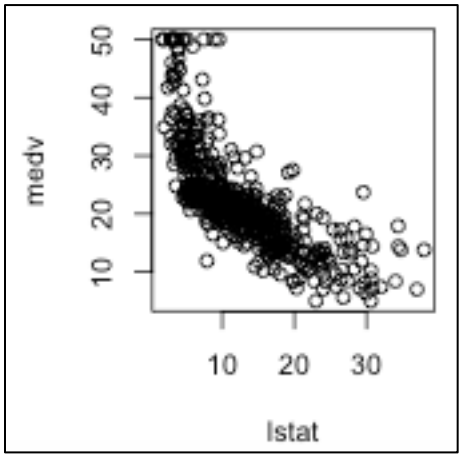


Figure 3 lstat marginal plot

III. Analysis & Results

Initial Full Regression Model

This analysis portion of the assignment involves the supervised learning method of multiple linear regressions paired with some variable selection procedures to try to hone in on the most relevant predictor variables for a model. The goal here is to try to find a model that is well balanced between *prediction* and *inference.* In other words, I want to create a model that can accurately predict **medv** that is not overly complicated in terms of large numbers of predictor transformation.

I begin by creating a *full linear model* of the form:

(1) **medv** = β_0 + β_1 (**crim**) + β_2 (**zn**) + β_3 (**indus**) + β_4 (**chas**) + β_5 (**nox**) + β_6 (**rm**) + β_7 (**age**) + β_8 (**dis**) + β_9 (**rad**) + β_{10} (**tax**) + β_{11} (**ptratio**) + β_{12} (**b**) + β_{13} **lstat** + **e.**

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	3.65E+01	5.10E+00	7.144	3.28E-12	***
crim	-1.08E-01	3.29E-02	-3.287	0.001087	**
zn	4.64E-02	1.37E-02	3.382	0.000778	***
indus	2.06E-02	6.15E-02	0.334	0.738288	
chas1	2.69E+00	8.62E-01	3.118	0.001925	**
nox	-1.78E+01	3.82E+00	-4.651	4.25E-06	***
rm	3.81E+00	4.18E-01	9.116	< 2e-16	***

age	6.92E-04	1.32E-02	0.052	0.958229	
dis	-1.48E+00	2.00E-01	-7.398	6.01E-13	***
rad	3.06E-01	6.64E-02	4.613	5.07E-06	***
tax	-1.23E-02	3.76E-03	-3.28	0.001112	**
ptratio	-9.53E-01	1.31E-01	-7.283	1.31E-12	***
b	9.31E-03	2.69E-03	3.467	0.000573	***
lstat	-5.25E-01	5.07E-02	-10.347	< 2e-16	***

This initial model has a 4.745 RSE and a R-squared of 0.7406. Notice that the variables **rm**, **lstat**, and **ptratio** appear to be the most significant coefficients when looking at the p-values. These were the exact same coefficients that we previously identified to have the largest (in absolute value) correlation coefficient with **medv**.

Numerous issues are identified when regression diagnostics is performed. For example, the "Residual vs. Fitted" and "Scale-Location" plots appear to have a slight U-shape, which provides is an indication of non-linearity in our data. The "Normal QQ" plot shows a large deviation from the straight line at the right side. This means that our standardized residuals are right-skewed implying that the normality assumption about errors is violated. The same three observations (369, 372, and 373) labeled on the "Residual vs. Fitted" plot are at the right tail end of the "Normal Q-Q" plot. These are three observations that should be tracked throughout the analysis because of their high residual values (i.e. potential outliers). From the "Residual vs. Leverage" plots, four observations are identified as being much larger than 2*(average of leverage) = 0.05533597. These observations are 411, 406, 419, and 38 (points are in ascending order).

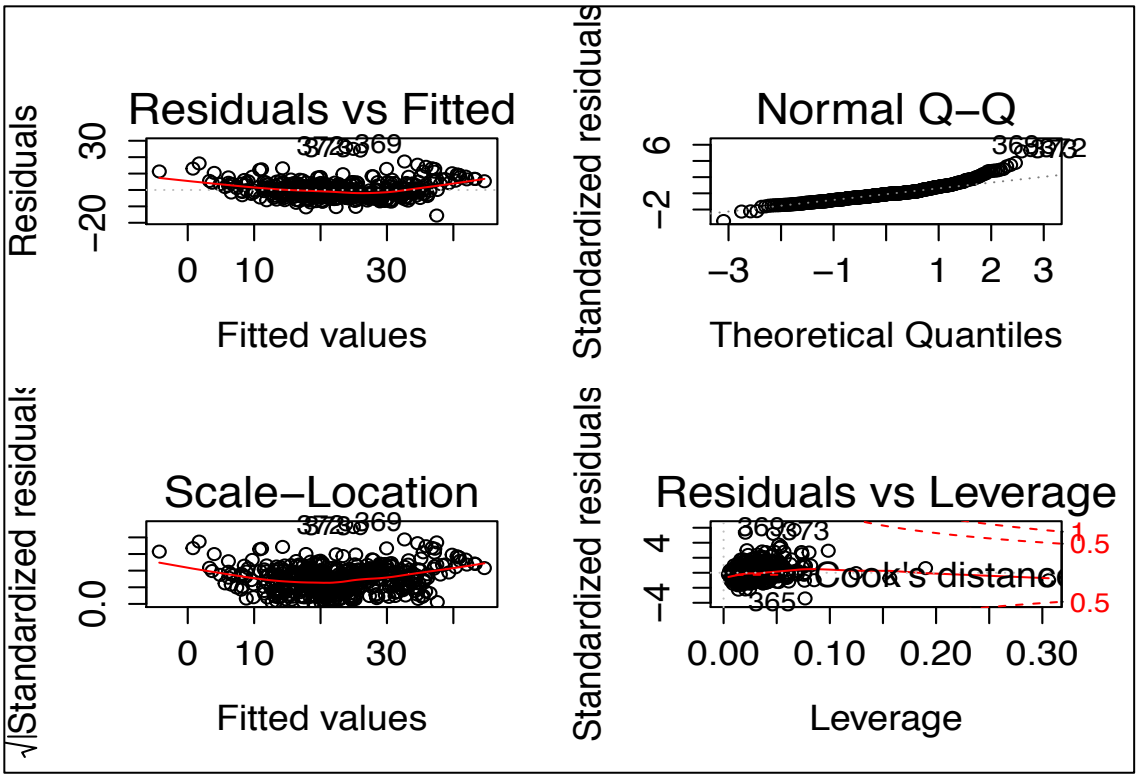


Figure 4: Regression Diagnostics Plot for Model (1)

Choosing an Optimal Model Size

In order to identify the subset of predictor variables that have a large association with the response variable, one can use *best subset selection* provided that our number of predictors *p* is not too large. Discarding those variables that have little association to our response can greatly increase the prediction accuracy of our model by getting rid of the extra noise and can lead to better model interpretability. In order to select the best model size from the best subset selection, my analysis *indirectly* estimated the test error (i.e. Mallow's Cp, AIC, BIC, and Adj R^2) and also *directly* estimated the test error via 10-fold cross- validation.

These five different methods yielded a linear model with 11 predictors, which is of the form:

(2) **medv** = β_0 + β_1 (**crim**) + β_2 (**zn**) + β_3 (**chas**) + β_4 (**nox**) + β_5 (**rm**) + β_6 (**dis**) + β_7 (**rad**) + β_8 (**tax**) + β_9 (**ptratio**) + β_{10} (**b**) + β_{11} (**lstat**) + **e.**

From this procedure, it is appears that **indus** and **age** are NOT significantly associated with **medv**. In other words, they provide very little information in trying to predict our response variable, which is why they were discarded. From the correlation matrix that was computed in 'Data Description' portion, it is important to note that both **indus** and **age** had some pretty high correlation coefficients with other predictors: **dis** vs. **age** -0.7478, **age** vs.

nox 0.7314, **tax** vs. **indus** 0.7207, **dis** vs. **indus** -0.7080, and **nox** vs. **indus** 0.7636. Because of this collinearity with other predictors, the best subset selection procedure dropped these variables because a large amount of their information was contained within other more important predictors for **medv.**

Recall that the marginal plots of our response variable **medv** against each of the predictor variables showed that **lstat** had a very strong non-linear relationship with **medv**, which suggests that a non-linear transformation of this predictor could enhance our model. This can also help to fix the issue of non-linearity of our residuals from the regression diagnostics of model (2), which was similarly present in the diagnostic plots of model (1) in Figure 4. Also, to fix the right-skewness that appeared in our "Normal Q-Q" plot, a log transformation of our response variable appears to be appropriate. The new model is now:

(3) $\log(\mathbf{medv}) = \beta_0 + \beta_1 (\mathbf{crim}) + \beta_2 (\mathbf{zn}) + \beta_3 (\mathbf{chas}) + \beta_4 (\mathbf{nox}) + \beta_5 (\mathbf{rm}) + \beta_6 (\mathbf{dis}) + \beta_7 (\mathbf{rad}) + \beta_8 (\mathbf{tax}) + \beta_9 (\mathbf{ptratio}) + \beta_{10} (\mathbf{b}) + \beta_{11} (\mathbf{lstat}) + \beta_{12} (\mathbf{lstat\text{^}2}) + \mathbf{e}.$

After these two transformations, best subset selection is performed one last time to determine the optimal model size. The final result is a model of size 11 with the predictor **zn** being dropped. After the log transformation of **medv**, notice that our R-squared improved from 0.7868 to 0.8025, despite the fact that we also dropped a predictor term of **zn**. The predictive model that is chosen is:

(4) $\log(\mathbf{medv}) = \beta_0 + \beta_1 (\mathbf{crim}) + \beta_2 (\mathbf{chas}) + \beta_3 (\mathbf{nox}) + \beta_4 (\mathbf{rm}) + \beta_5 (\mathbf{dis}) + \beta_6 (\mathbf{rad}) + \beta_7 (\mathbf{tax}) + \beta_8 (\mathbf{ptratio}) + \beta_9 (\mathbf{b}) + \beta_{10} (\mathbf{lstat}) + \beta_{11} (\mathbf{lstat\text{^}2}) + \mathbf{e}.$

The "Residual vs. Fitted" and "Scale-Location" plots for model (4) imply homoscedasticity (i.e. constant variance) because there is no 'funnel shape' in the residuals. Also, there is no longer an issue with non-linearity of our residuals because of the polynomial transformation of the **lstat** predictor. Three observations are labeled as potential outliers because of there high residual value. These points are 402, 373, and 372. Notice that the "Normal Q-Q" plot also improved after the transformations. The log transformation of the response got rid of the right-tail that was originally present in the residuals. Looking at the "Residual vs. Leverage" plot and also the "Leverage vs. Index" plot, the observations 381 and 419 are identified as high leverage points because they greatly exceed the 2*(average of leverages) mark and even the 4*(average of leverages) mark.

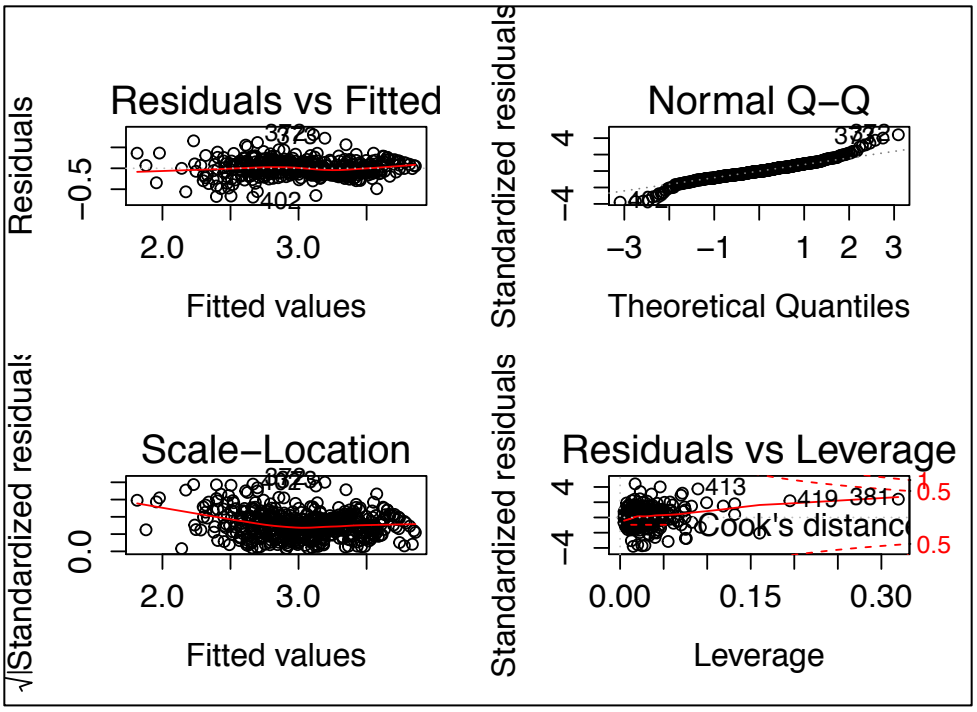


Figure 5: Regression Diagnostics Plot for Model (4)

Recall that Cook's Distance measures how much the regression would change if a point were deleted. Cook's distance is increased by leverage AND by large residuals: a point far from the centroid (of points) with a large residual can severely distort the regression. Three points with the largest Cook's Distance are 381, 413, and 419. However, none of them exceed the 0.5 Cook's distance mark, which is the rule-of-thumb boundary in trying to determine if points have significantly large Cook's Distance. The multiple t-test on the standardized predicted residual with the bonferroni correction could provide statistical proof as to whether or not some observations are outliers. Indeed, observations 401 and 402 are identified as outliers. The following is the output of model (4) after the removal of the outliers, which has an RSE of 0.1786 and an R-squared of 0.8074. Also, the calculation of our Variance Inflation Factor indicates an absence of collinearity between the chosen predictors, which is desired.

Table 1: Summary of Model (4)

	Estimate	Std. Error	t-value	Pr(>|t|)	
(Intercept)	3.6264135	0.1862759	19.468	< 2e-16	***
crim	-0.0107555	0.0012483	-8.616	< 2e-16	***
chas1	0.0986104	0.0322255	3.06	0.002334	**
nox	-0.634825	0.1337848	-4.745	2.74E-06	***
rm	0.0839802	0.0154232	5.445	8.19E-08	***
dis	-0.0448149	0.0060681	-7.385	6.55E-13	***
rad	0.0129048	0.0023872	5.406	1.01E-07	***
tax	-0.0004793	0.0001248	-3.84	0.000139	***
ptratio	-0.0351216	0.0046665	-7.526	2.51E-13	***

b	0.0004465	0.0001016	4.397	0.0000135	***
poly(lstat,2)1	-4.8346222	0.2931063	-16.494	< 2e-16	***
poly(lstat,2)2	1.200432	0.1923433	6.241	9.39E-10	***

V. General Discussion/ Conclusion

Our final chosen predictive model is:

(4) log(**medv)** = β_0 + β_1 (**crim) +** β_2 (**chas) +** β_3 (**nox) +** β_4 (**rm) +** β_5 (**dis) +** β_6 **(rad) +** β_7 (**tax) +** β_8 (**ptratio) +** β_9 (**b) +** β_{10} **(lstat) +** β_{11} **(lstat^2) + e.**

The aim of this analysis was to identify which predictor variables are the most important when trying to predict the median value of a Boston house in the 1970s. At the start of this report, the linear model was fitted using all 13 variables. Through the usage of best subset selection, our model discarded 3 variables: **indus**, **age,** and **zn**. The two other big changes that occurred in the model was the addition of the **lstat^2** predictor variable and the log transformation of the response variable: log(**medv)**.

In multiple linear regression, we interpret β_j as the average effect on Y of a one unit increase in X_j , holding all other predictors fixed. Using this interpretation, one can say that -0.0107555 is the average effect on log(**medv)** of a one unit increase in **crim**, holding all other predictors fixed. Similarly, this linear model implies that 0.0839802 is the average effect on log(**medv**) of a one unit increase in **rm**. In other words, as the crime rate of a tract increases, the log of the median value of a house will decrease, provided that all other predictors are fixed. Also, as the average number of rooms per household in a tract increases, the log of the median value of a house will decrease; provide that all other predictors are fixed. Because the log function is monotonic, it does not create huge interpretation issues in our model. Therefore, the β.hat estimates of this model provide significant amount of information about how a particular predictor relates to the response variable of the model. However, we have to proceed with caution here.

Comparing all our β.hat estimates of model (4) to the correlation coefficients of **medv** and each individual predictor, it is clear that all the corresponding predictors have the exact same positive or negative sign except **rad** and **dis**. Notice how **dis** vs. **medv** is positively correlated, whereas the β_5.hat estimate for rad is negative! There appears to be contradictory conclusions being made. This is NOT the case, however. Suppose I had performed a simple linear regression of log(**medv)** ~ **dis.** I would have obtained a *positive* value for the β.hat estimate of **dis** because of the already identified positive correlation between the two variables. However, in the multiple linear regression setting of model (4), we obtained a *negative* β.hat estimate. This difference stems from the fact that in the simple regression case, the slope term represents the average effect of a 1-unit increase in **dis**, *ignoring* every other predictor. In contrast, in the multiple regression setting, the β coefficient for **dis** represents the average effect of increasing **dis** by a unit while holding the rest of the predictors in model (4) fixed. Paying attention to these small details is very important when trying to determine the meaning of β coefficients in multiple linear regressions.

In conclusion, this analysis shows how **crim, chas, nox, rm, dis, rad, tax, ptratio, b, lstat, and lstat^2** are the most important variables in predicting the median value of home in Boston in the 1970s. The prices of the houses in various tracts were affected in various different ways by these predictors. For example, our model suggests that **medv** decreases as **ptratio** increases. In other words, tracts that have a low pupil to teacher ratio seem to be areas that are more expensive. It was pointed out above that the median value of households tends to decrease when crime rates are high. These interpretations are a huge oversimplification of the relationship that all of these variables have with one another. However, this is the beauty of linear modeling. Linear regression is a relatively *inflexible* model that has high levels of interpretability, like we just showed above. However, this comes at the expense of less predictive power. I avoided a large amount of complicated transformations for my predictors because I placed more weight on *inference* for this analysis (i.e. trying to determine the relationship between my predictors and my response). The final model for this analysis had a good Test MSE, without having to sacrifice much interpretability.

Table 2: Correlation of medv With Predictors

	medv
crim	-0.3883046
zn	0.3604453
indus	-0.4837252
chas	0.1752602
nox	-0.4273208
rm	0.6953599
age	-0.3769546
dis	0.2499287
rad	-0.3816262
tax	-0.4685359
ptratio	-0.5077867
b	0.3334608
lstat	-0.7376627
medv	1

Stats 151A: Midterm 2 Code (Appendix)

Anthony Cerna

November 23, 2015

Data

This dataset containes 506 observations (i.e. rows) and 14 variables (i.e. columns). Each observation represents a neighborhood in Boston.

```
library(mlbench)
data(BostonHousing)
attach(BostonHousing)
dim(BostonHousing)
```

```
## [1] 506  14
```

```
names(BostonHousing)
```

```
##  [1] "crim"    "zn"      "indus"   "chas"    "nox"     "rm"      "age"
##  [8] "dis"     "rad"     "tax"     "ptratio" "b"       "lstat"   "medv"
```

The following are a description of the varaibles: crim: per capita crime rate by town zn: proportion of residential land zoned for lots over 25,000 sq.ft indus: proportion of non-retail business acres per town chas: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise) nox: nitric oxides concentration (parts per 10 million) rm: average number of rooms per dwelling age: proportion of owner-occupied units built prior to 1940 dis: weighted distances to five Boston employment centres rad: index of accessibility to radial highways tax: full-value property-tax rate per USD 10,000 ptratio: pupil-teacher ratio by town b: 1000(B - 0.63)ˆ2 where B is the proportion of blacks by town lstat: percentage of lower status of the population Our response variable is: medv: median value of owner-occupied homes in USD 1000's

Clean Our Data

We first check to see if there are any missing values in our dataset, since these can be problematic.

```
apply(BostonHousing, 2, function(x) sum(is.na(x))) #No NA's, which is good
```

```
##    crim      zn   indus    chas     nox      rm     age     dis     rad
##       0       0       0       0       0       0       0       0       0
##     tax ptratio       b   lstat    medv
##       0       0       0       0       0
```

There are no missing NA's, so we continue with the analysis.

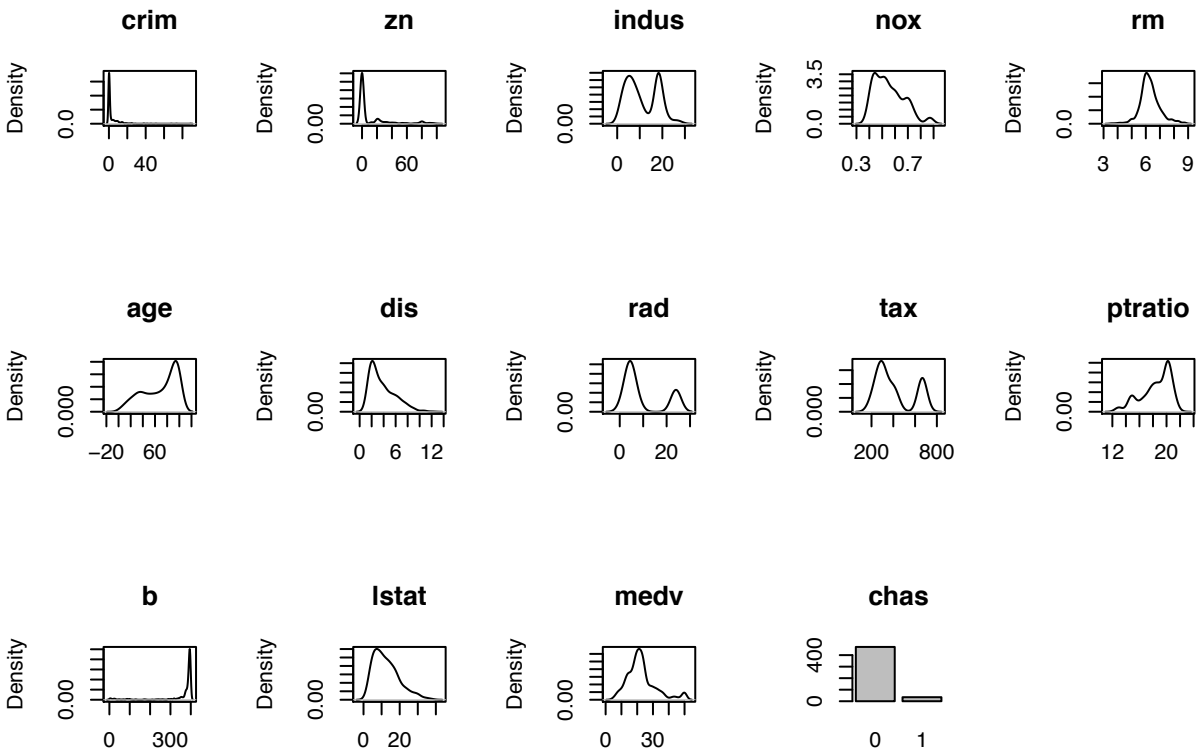
EXPLORATORY DATA ANALYSIS

PRELIMINARY SEARCH FOR OUTLIERS In this portion, we should make a note of any extreme values we might see. To do this, we do a combination of the summary function and the plotting of the densities of each variable. Any long tails suggest that there are extreme values relative to the rest of the group (i.e. potential outliers).

```
summary(BostonHousing)
```

```
##       crim                zn              indus        chas
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   0:471
##  1st Qu.: 0.08204   1st Qu.:  0.00   1st Qu.: 5.19   1: 35
##  Median : 0.25651   Median :  0.00   Median : 9.69
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74
##       nox               rm              age              dis
##  Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130
##  1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100
##  Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207
##  Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795
##  3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188
##  Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127
##       rad              tax           ptratio            b
##  Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32
##  1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38
##  Median : 5.000   Median :330.0   Median :19.05   Median :391.44
##  Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67
##  3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23
##  Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :396.90
##      lstat            medv
##  Min.   : 1.73   Min.   : 5.00
##  1st Qu.: 6.95   1st Qu.:17.02
##  Median :11.36   Median :21.20
##  Mean   :12.65   Mean   :22.53
##  3rd Qu.:16.95   3rd Qu.:25.00
##  Max.   :37.97   Max.   :50.00
```

DENSITY PLOTS



There are a couple of things to note. Notice the long tails for crim, zn, b, dis and lstat. Also notice from the density plots that there appears to be two very different groups of data by looking at the densities of indus, rad and tax.

POTENTIAL OUTLIERS

```
ou1=which(crim>30)
ou1
```

```
## [1] 381 399 405 406 411 415 419 428
```

```
ou2=which(zn>70)
ou2
```

```
##  [1]  40  41  55  56  57  58  66  67 196 197 198 199 200 201 202 203 204
## [18] 205 255 256 257 284 285 287 291 292 293 348 349 354 355 356
```

```
ou7=which(dis>9)
ou7
```

```
##  [1]  57  65 255 256 287 352 353 354 355 356
```

```
ou11=which(b<100)
ou11
```

```
##  [1] 103 156 157 411 412 413 415 416 417 419 420 424 425 426 427 428 429
## [18] 430 431 432 433 437 438 439 446 451 455 456 457 458 467
```

```
ou = c(ou1, ou2, ou7, ou11)
ou
```

```
##  [1] 381 399 405 406 411 415 419 428  40  41  55  56  57  58  66  67 196
## [18] 197 198 199 200 201 202 203 204 205 255 256 257 284 285 287 291 292
## [35] 293 348 349 354 355 356  57  65 255 256 287 352 353 354 355 356 103
## [52] 156 157 411 412 413 415 416 417 419 420 424 425 426 427 428 429 430
## [69] 431 432 433 437 438 439 446 451 455 456 457 458 467
```

From this preliminary search for outliers, it is difficult to pinpoint a small subset of potential outliers by simply looking at the tails of the density plots. "ou1" does give 6 observations with very high crim rate: 381 399 405 406 411 415 419 428. "ou7" also gives us a small subset of observations with large dis values (i.e. towns very far from five Boston Employment centers): 57 65 255 256 287 352 353 354 355 356. However,when we try to extract a small subset of extreme values from the zn and b variables, we end up with about 32 different observations for each variable. This just shows the large amount of skewness for this particular variables.

CORRELATION OF PREDICTOR VARIABLES WITH RESPONSE VARIABLES

I check to see which predictors are highly correlated with our response variable medv.

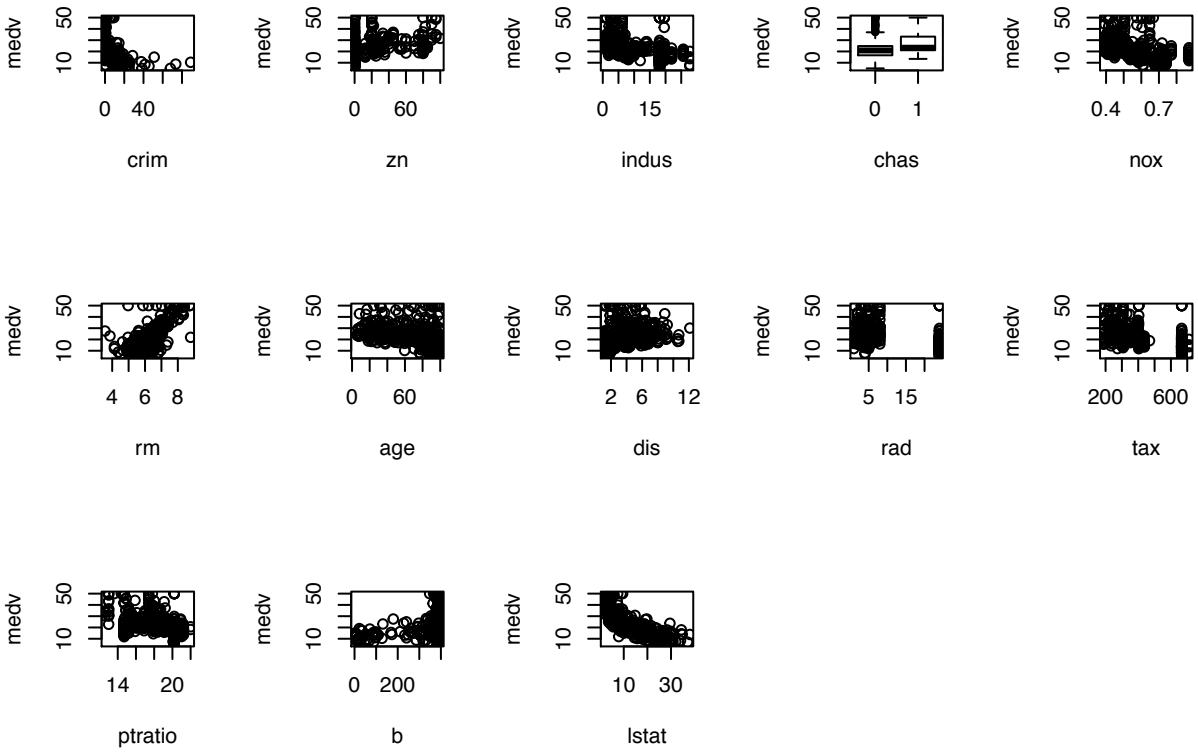
```
BostonHousing$chas <- as.numeric(chas) #change to numeric to calculate correlation
cor(BostonHousing)
```

```
##                crim          zn       indus         chas         nox
## crim     1.00000000 -0.20046922  0.40658341 -0.055891582  0.42097171
## zn      -0.20046922  1.00000000 -0.53382819 -0.042696719 -0.51660371
## indus    0.40658341 -0.53382819  1.00000000  0.062938027  0.76365145
## chas    -0.05589158 -0.04269672  0.06293803  1.000000000  0.09120281
## nox      0.42097171 -0.51660371  0.76365145  0.091202807  1.00000000
## rm      -0.21924670  0.31199059 -0.39167585  0.091251225 -0.30218819
## age      0.35273425 -0.56953734  0.64477851  0.086517774  0.73147010
## dis     -0.37967009  0.66440822 -0.70802699 -0.099175780 -0.76923011
## rad      0.62550515 -0.31194783  0.59512927 -0.007368241  0.61144056
## tax      0.58276431 -0.31456332  0.72076018 -0.035586518  0.66802320
## ptratio  0.28994558 -0.39167855  0.38324756 -0.121515174  0.18893268
## b       -0.38506394  0.17552032 -0.35697654  0.048788485 -0.38005064
## lstat    0.45562148 -0.41299457  0.60379972 -0.053929298  0.59087892
## medv    -0.38830461  0.36044534 -0.48372516  0.175260177 -0.42732077
##                  rm         age         dis          rad         tax
## crim    -0.21924670  0.35273425 -0.37967009  0.625505145  0.58276431
## zn       0.31199059 -0.56953734  0.66440822 -0.311947826 -0.31456332
## indus   -0.39167585  0.64477851 -0.70802699  0.595129275  0.72076018
## chas     0.09125123  0.08651777 -0.09917578 -0.007368241 -0.03558652
## nox     -0.30218819  0.73147010 -0.76923011  0.611440563  0.66802320
## rm       1.00000000 -0.24026493  0.20524621 -0.209846668 -0.29204783
## age     -0.24026493  1.00000000 -0.74788054  0.456022452  0.50645559
## dis      0.20524621 -0.74788054  1.00000000 -0.494587930 -0.53443158
```

```
## rad     -0.20984667  0.45602245 -0.49458793  1.000000000  0.91022819
## tax     -0.29204783  0.50645559 -0.53443158  0.910228189  1.00000000
## ptratio -0.35550149  0.26151501 -0.23247054  0.464741179  0.46085304
## b        0.12806864 -0.27353398  0.29151167 -0.444412816 -0.44180801
## lstat   -0.61380827  0.60233853 -0.49699583  0.488676335  0.54399341
## medv     0.69535995 -0.37695457  0.24992873 -0.381626231 -0.46853593
##            ptratio           b      lstat       medv
## crim     0.2899456 -0.38506394  0.4556215 -0.3883046
## zn      -0.3916785  0.17552032 -0.4129946  0.3604453
## indus    0.3832476 -0.35697654  0.6037997 -0.4837252
## chas    -0.1215152  0.04878848 -0.0539293  0.1752602
## nox      0.1889327 -0.38005064  0.5908789 -0.4273208
## rm      -0.3555015  0.12806864 -0.6138083  0.6953599
## age      0.2615150 -0.27353398  0.6023385 -0.3769546
## dis     -0.2324705  0.29151167 -0.4969958  0.2499287
## rad      0.4647412 -0.44441282  0.4886763 -0.3816262
## tax      0.4608530 -0.44180801  0.5439934 -0.4685359
## ptratio  1.0000000 -0.17738330  0.3740443 -0.5077867
## b       -0.1773833  1.00000000 -0.3660869  0.3334608
## lstat    0.3740443 -0.36608690  1.0000000 -0.7376627
## medv    -0.5077867  0.33346082 -0.7376627  1.0000000
```

```
BostonHousing$chas<- as.factor(chas) #change back to factor
```

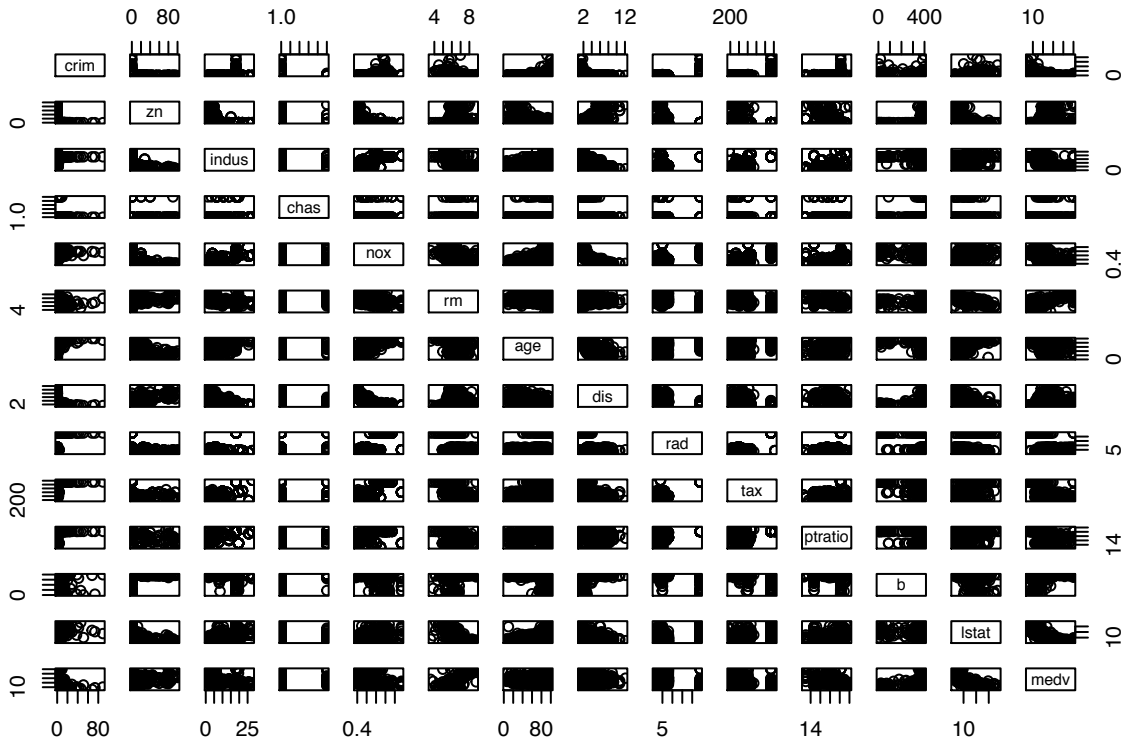
Looking at the last row of our correlation matrix, we can see that rm (average number of rooms per dwelling) has a large positive correlation with value 0.6954. Also, notice that ptratio(pupil-teacher ratio by town) and lstat (% of lower status of the population) have relatively large correlation coefficients of -0.5078 and -0.7377 respectively. What this means is that these are very good predictor variables and will almost surely be included in our most optimal predictive model for predicting medv (median value of owner-occupied homes in USD 1000's), provided that they are not highly correlated with each other (i.e. multicollinearity). The marginal plots provides a good visualization.



Notice that lstat shows a very strong non-linear relationship with our response variable medv, which suggests that a non-linear transformation of this predictor might be needed.

COLLINEARITY OF PREDICTORS

We check to see if the variables are correlated with one another. If many variables are correlated with each other, then this suggest that we can create a predictive model with a subset of our predictors. To do this, we can simply look at the upper triangular part of same correlation matrix from above. We can also create a matrix of plots to assist us.



Notice that tax and rad have a large correlation of 0.9102, dis and age -0.7478, dis and nox -0.7692, age and nox 0.7314, tax and indus 0.7207, dis and indus -0.7080, nox and indus 0.7636. Notice how age and indus tend to have the larger correlation coefficients with other predictor variables. This can pose problems in the regression context, since it can be difficult to separate out individual effects of collinear variables on the response.

INITIAL MODEL

I first fit a full model with all my predictor variables and medv as my response variable. I then obtain the summary of the model.

```
full.mod <- lm(medv~., data=BostonHousing)
summary(full.mod)
```

```
##
## Call:
## lm(formula = medv ~ ., data = BostonHousing)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.595  -2.730  -0.518   1.777  26.199
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.646e+01  5.103e+00   7.144 3.28e-12 ***
## crim        -1.080e-01  3.286e-02  -3.287 0.001087 **
## zn           4.642e-02  1.373e-02   3.382 0.000778 ***
## indus        2.056e-02  6.150e-02   0.334 0.738288
## chas1        2.687e+00  8.616e-01   3.118 0.001925 **
```

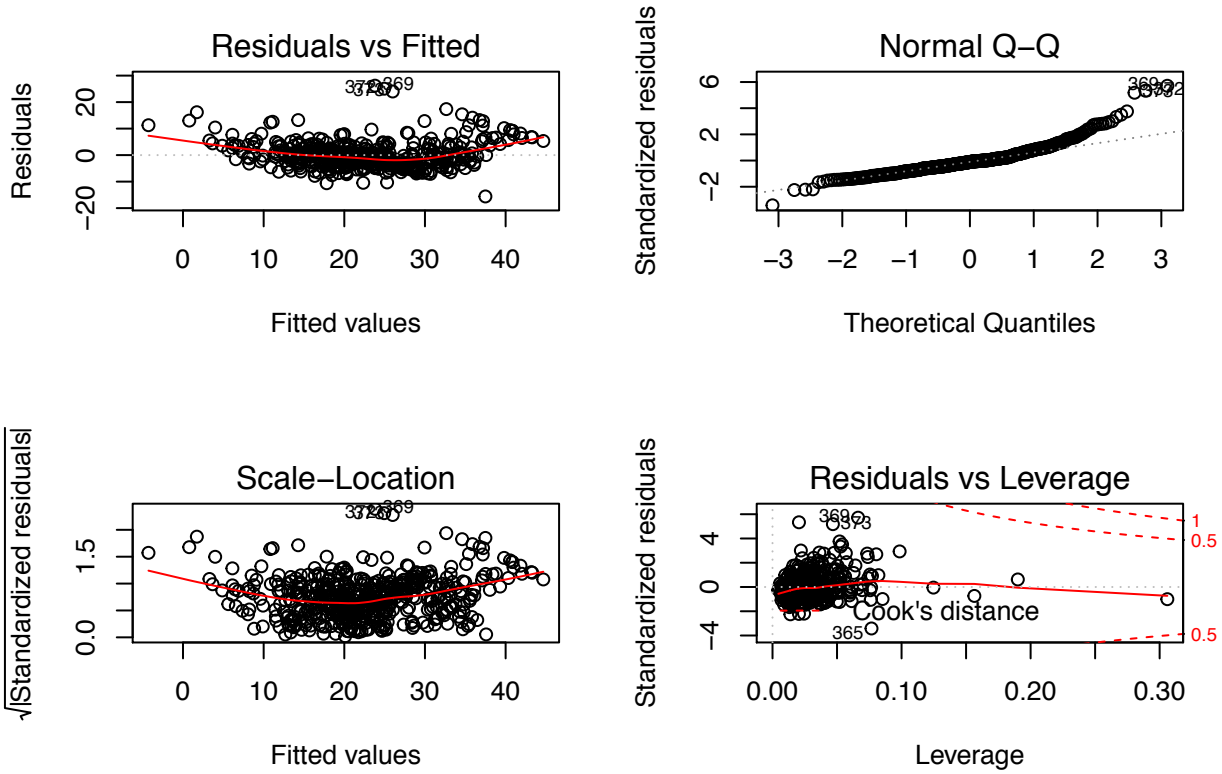
```
## nox         -1.777e+01  3.820e+00  -4.651 4.25e-06 ***
## rm           3.810e+00  4.179e-01   9.116  < 2e-16 ***
## age          6.922e-04  1.321e-02   0.052 0.958229
## dis         -1.476e+00  1.995e-01  -7.398 6.01e-13 ***
## rad          3.060e-01  6.635e-02   4.613 5.07e-06 ***
## tax         -1.233e-02  3.760e-03  -3.280 0.001112 **
## ptratio     -9.527e-01  1.308e-01  -7.283 1.31e-12 ***
## b            9.312e-03  2.686e-03   3.467 0.000573 ***
## lstat       -5.248e-01  5.072e-02 -10.347  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.745 on 492 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7338
## F-statistic: 108.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

```
round(summary(full.mod)$coef, 5)
```

```
##              Estimate Std. Error   t value Pr(>|t|)
## (Intercept)  36.45949    5.10346   7.14407  0.00000
## crim         -0.10801    0.03286  -3.28652  0.00109
## zn            0.04642    0.01373   3.38158  0.00078
## indus         0.02056    0.06150   0.33431  0.73829
## chas1         2.68673    0.86158   3.11838  0.00193
## nox         -17.76661    3.81974  -4.65126  0.00000
## rm            3.80987    0.41793   9.11614  0.00000
## age           0.00069    0.01321   0.05240  0.95823
## dis          -1.47557    0.19945  -7.39800  0.00000
## rad           0.30605    0.06635   4.61290  0.00001
## tax          -0.01233    0.00376  -3.28001  0.00111
## ptratio      -0.95275    0.13083  -7.28251  0.00000
## b             0.00931    0.00269   3.46679  0.00057
## lstat        -0.52476    0.05072 -10.34715  0.00000
```

Notice that the variables rm, lstat, and ptratio appear to be the most significant coefficients when looking at the p-values. These were the exact same coefficients that we previously identified to have the largest (both positive and negative) correlation coefficient with medv.

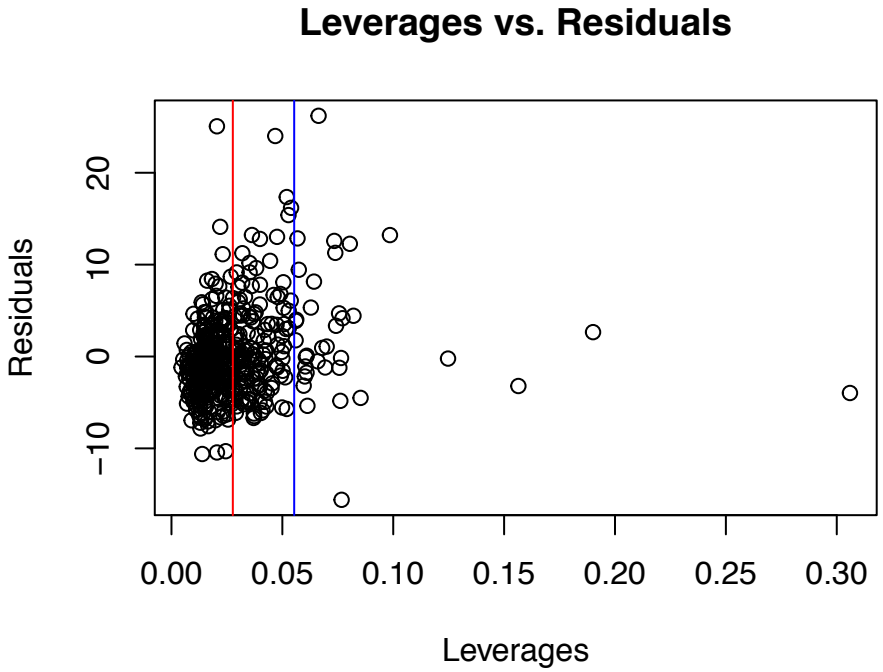
Initial Diagnostics



Recall that the "Residuals vs Fitted" and "Scale-Location" plots are essentially the same. From these two plots, it appears that the residuals have a slight U-shape which provides some indication of non-linearity in our data. This indicates that we might have to use a non-linear transformation of some predictors. Notice also that observations 369, 372, and 373 have very high residuals, which suggests that these points are potential outliers.

Looking at the "Normal QQ" plot, we see that a large deviation from the straight line at the right side of the plot. This means that our standardized residuals are right-skewed. So our normality assumption about errors seems to be violated. We can again see that the same 3 observations identified by the "Residual vs. Fitted" plot are also labeled here.

Lastly, we look at the "Residuals vs. Leverage" plots.



I calculated the (average of the leverages) = (4+1)/n=0.02766798(indicated by the red line). If a given observation has a leverage statistic that is greater than 2*(average of leverage) = 0.05533597 (indicated by the blue line) then the corresponding point has high leverage. We see that there are a number of points that are past the blueline, however there are four points in particular that are far greater than the blue line.Using the identify function, we identify observations 411, 406, 419, and 38 as having high leverage (points are in ascending order).

CHOOSING AN OPTIMAL MODEL SIZE USING ADJ R^2, BIC, AND MALLOW'S CP

I first want to find out what is the optimal model size for my regression model. I attempt to answer this question by first using the regsubsets() function in order to perform best subset selection. In other words, I fit a separate least squares regression line for every possible combination of of the 13 predictors. For each model of a particular size, I choose the best model by choosing the one that has the smallest RSS. For example, I will first look at every possible model that contains 1 predictor, choose the model with the smallest RSS, and then repeat this process for a model that contains 2 predictors. Therefore, I end up with a total of 13 'best' models (each of different size).

```
library(leaps)
regsubs.full=regsubsets(medv~.,data=BostonHousing, nvmax=13)
regsubs.sum=summary(regsubs.full)
```

To choose the most optimal model from the 13 models of different sizes,we can ESTIMATE the test error of each model by making an adjustment to our error by taking into account the fact that the models differ in size. As we have learned this semester, we can look at Mallow's Cp,BIC,Adjusted R^2, and AIC.

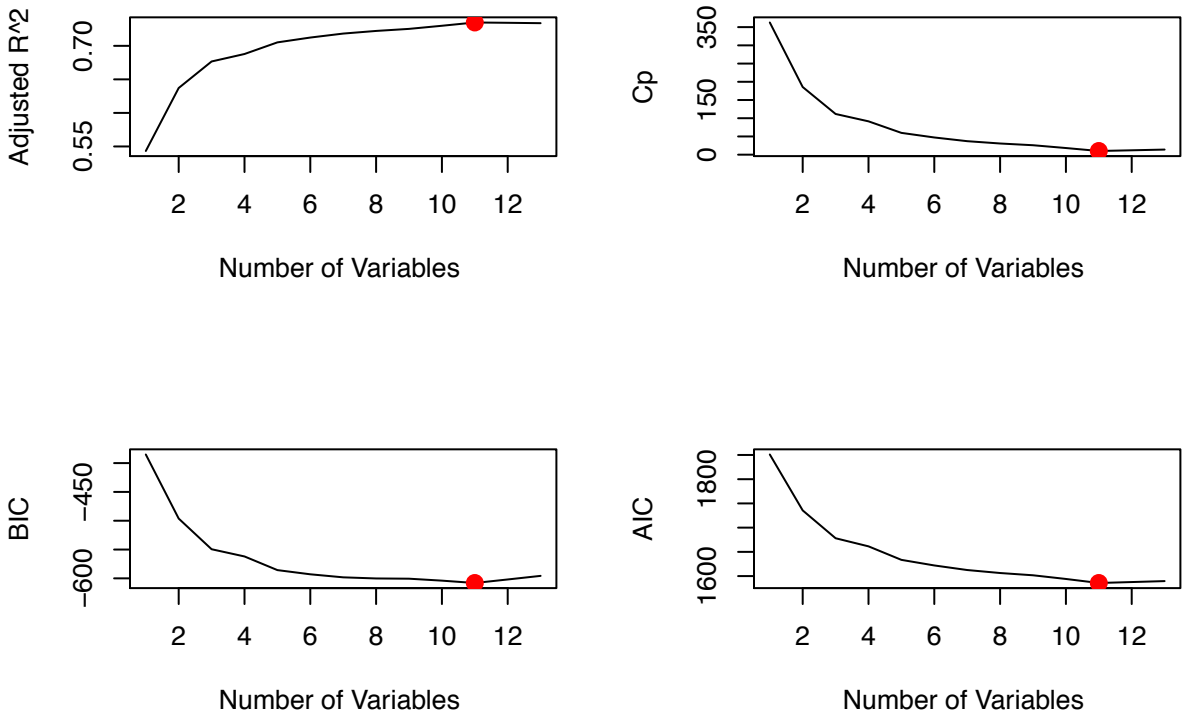
```
## [1] 11
```

```
## [1] 11
```

```
## [1] 11
```

```
##  [1] 1851.009 1735.577 1678.131 1661.393 1633.473 1621.973 1612.473
##  [8] 1606.309 1601.672 1594.031 1585.761 1587.646 1589.643
```

```
## [1] 11
```




```
##   (Intercept)          crim            zn         chas1           nox
##  36.341145004  -0.108413345   0.045844929   2.718716303 -17.376023429
##            rm           dis           rad           tax       ptratio
##   3.801578840  -1.492711460   0.299608454  -0.011777973  -0.946524570
##             b         lstat
##   0.009290845  -0.522553457
```

From these four plots, it is clear that a model of size 11 is the best model size with respect to the estimated test error. Our 11 variable model gets rid of indus and age.

CHOOSING AN OPTIMAL MODEL SIZE USING CROSS-VALIDATION

I also attempt to find an optimal model size by directly estimating the test error using the 10-fold cross-validation technique.

```
k=10
set.seed(1)
folds=sample(x=1:k, size=nrow(BostonHousing), replace=TRUE) #Note: NOT equal-sized folds
cv.errors=matrix(data=NA, nrow=k, ncol=13, dimnames=list(NULL, paste(1:13)))
```

```
#I create a for loop to perform CV. In the ith fold, the elements of folds that equal i are in the test
for(i in 1:k){
  best= regsubsets(medv~., data=BostonHousing[folds!=i,], nvmax=13)
  for(j in 1:13) {
    coefi=coef(best, id=j) #Beta.hat vector
    nam=names(coefi)
    replace(nam, nam=="chas1", "chas")
    test.mat=model.matrix(medv~., data=BostonHousing[folds==i,], nvmax=13)
    pred=test.mat[,nam]%*%coefi #Y.hat
    cv.errors[i,j]=mean((BostonHousing$medv[folds==i]-pred)^2)
  }
}
cv.errors
```

```
##              1        2        3        4        5        6        7
##  [1,] 35.44616 24.51926 19.30059 17.56207 18.54459 18.55069 16.68644
##  [2,] 29.51830 23.13902 21.34213 21.51456 17.73440 17.67907 16.77428
##  [3,] 42.80247 29.40477 26.49839 28.20283 26.07464 27.87814 25.86151
##  [4,] 28.56110 16.63569 13.04672 12.73908 12.83084 13.28999 12.60347
##  [5,] 41.14179 33.50183 32.31408 31.45725 30.08384 29.19085 29.61660
##  [6,] 51.82067 43.80377 40.52366 45.54283 33.82245 37.72433 37.89960
##  [7,] 37.19266 38.06643 36.18826 35.77112 36.15917 37.20492 36.26603
##  [8,] 53.43504 42.69138 33.77638 38.35374 37.55944 34.07379 32.76613
##  [9,] 34.10846 35.40319 29.87003 27.13465 26.38915 25.20487 24.12834
## [10,] 37.90551 30.77866 28.67872 27.30445 26.75491 25.32572 25.45368
##              8        9       10       11       12       13
##  [1,] 16.33680 20.64012 20.01157 18.11008 18.31962 18.48291
##  [2,] 17.49269 17.30694 16.80800 15.49213 15.47092 15.48336
##  [3,] 24.74010 24.89166 24.27559 22.22565 22.24301 22.35099
##  [4,] 12.98786 13.37136 12.93109 11.86230 11.94018 12.03613
##  [5,] 29.20652 29.94815 27.74203 25.86101 25.98271 25.98338
##  [6,] 36.29564 36.77980 35.91484 33.94954 34.07186 34.05015
##  [7,] 37.54791 37.45065 37.10872 36.38257 36.70569 36.67819
##  [8,] 32.01112 32.65156 32.52019 31.21210 31.50163 31.50205
##  [9,] 24.18409 24.06871 24.06596 23.46169 23.56592 23.64684
## [10,] 23.09811 24.26575 23.51774 21.61191 21.60193 21.60890
```

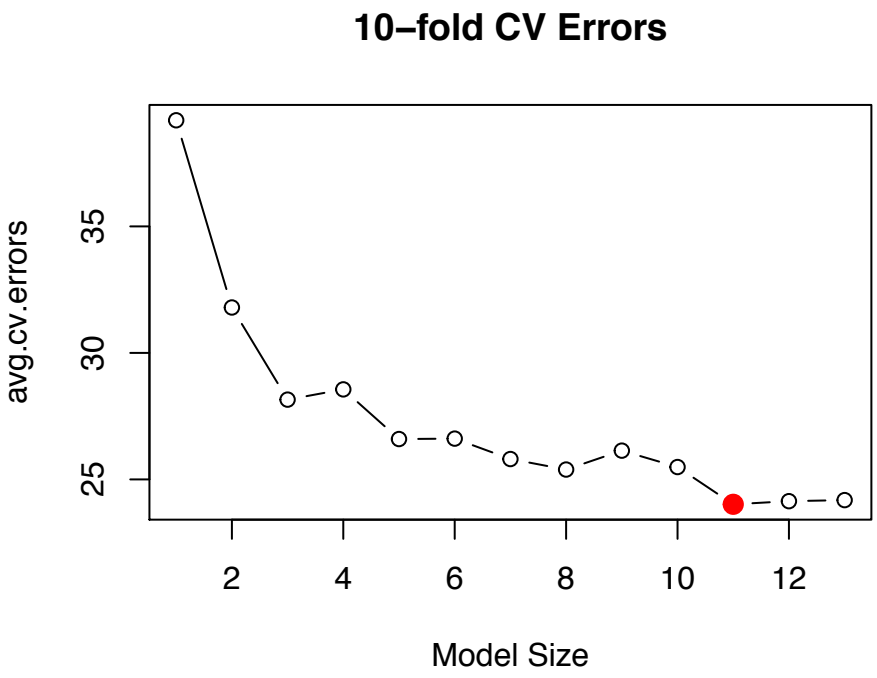
```
#The cv.errors is a 10X13 matrix, where the i,jth entry is the test MSE for the ith fold and the jth si
avg.cv.errors=apply(cv.errors, 2, mean)
avg.cv.errors
```

```
##        1        2        3        4        5        6        7        8
## 39.19322 31.79440 28.15390 28.55826 26.59534 26.61224 25.80561 25.39008
##        9       10       11       12       13
## 26.13747 25.48957 24.01690 24.14035 24.18229
```

```
par(mfrow=c(1,1))
plot(avg.cv.errors, type='b', main="10-fold CV Errors", xlab="Model Size")
which.min(avg.cv.errors)
```

```
## 11
## 11
```

```
points(11,avg.cv.errors[11],col="red",cex=2,pch=20)
```




```
#We see that cross-validation selects an 11-variable model as well.
coef(regsubs.full, 11)
```

```
##   (Intercept)          crim            zn         chas1           nox
##  36.341145004  -0.108413345   0.045844929   2.718716303 -17.376023429
##            rm           dis           rad           tax       ptratio
##   3.801578840  -1.492711460   0.299608454  -0.011777973  -0.946524570
##             b         lstat
##   0.009290845  -0.522553457
```

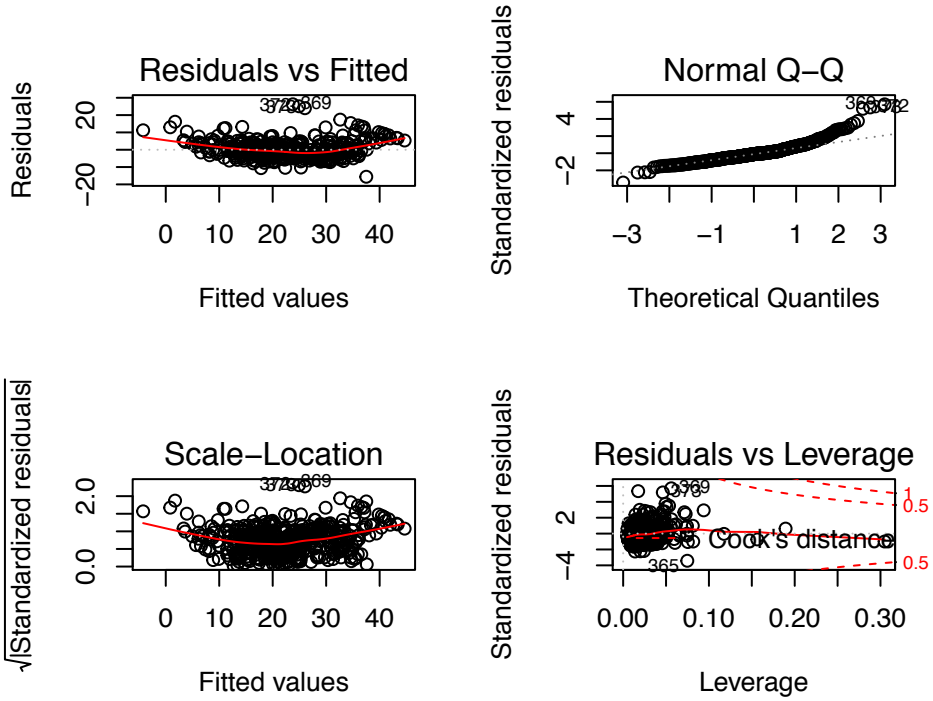
From looking at Mallow's Cp,BIC,Adjusted R^2, AIC, and cross-validation, I end up with an 11-variable model. This model was shown to have the lowest estimated test error.

```
sub.mod<-lm(medv~crim+zn+chas+nox+rm+dis+rad+tax+ptratio+b+lstat, data=BostonHousing)
summary(sub.mod)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + b + lstat, data = BostonHousing)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.5984  -2.7386  -0.5046   1.7273  26.2373
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  36.341145   5.067492   7.171 2.73e-12 ***
```

```
## crim         -0.108413   0.032779  -3.307 0.001010 **
## zn            0.045845   0.013523   3.390 0.000754 ***
## chas1         2.718716   0.854240   3.183 0.001551 **
## nox         -17.376023   3.535243  -4.915 1.21e-06 ***
## rm            3.801579   0.406316   9.356  < 2e-16 ***
## dis          -1.492711   0.185731  -8.037 6.84e-15 ***
## rad           0.299608   0.063402   4.726 3.00e-06 ***
## tax          -0.011778   0.003372  -3.493 0.000521 ***
## ptratio      -0.946525   0.129066  -7.334 9.24e-13 ***
## b             0.009291   0.002674   3.475 0.000557 ***
## lstat        -0.522553   0.047424 -11.019  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.736 on 494 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7348
## F-statistic: 128.2 on 11 and 494 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2, 2))
plot(sub.mod)
```



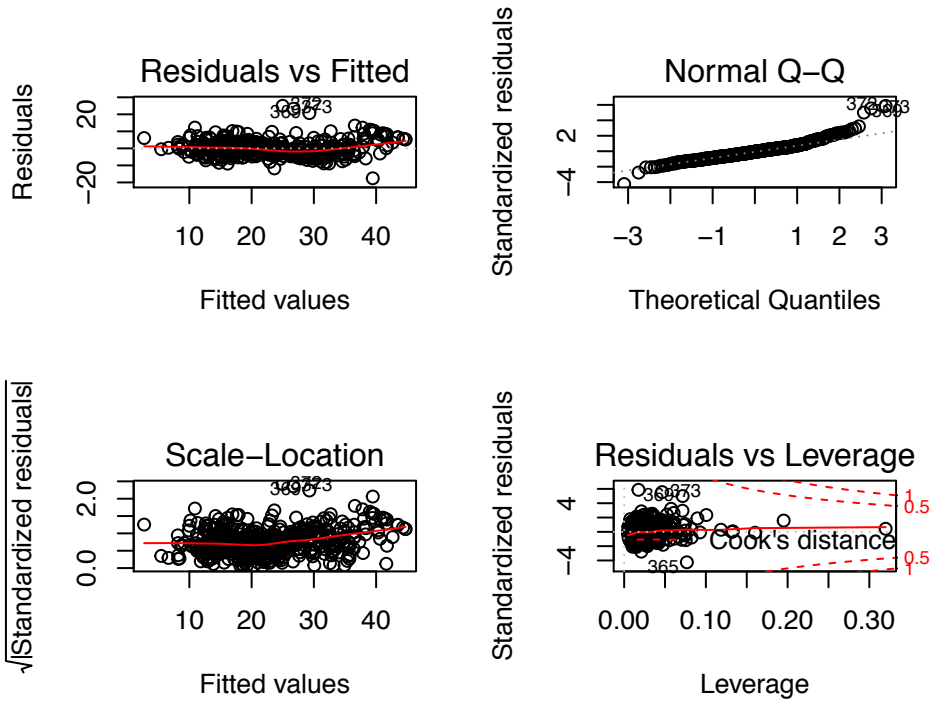
Also, recall that the above analysis of the marginal plots of our response variable medv against each of the predictor variables showed that lstat had a very strong non-linear relationship with medv, which suggests that a non-linear transformation of this predictor could enhance our model. I add a second degree polynomialterm of lstat to the model.

```
sub.mod<-lm(medv~crim+zn+chas+nox+rm+dis+rad+tax+ptratio+b+poly(lstat,2), data=BostonHousing)
summary(sub.mod)
```

```
##
## Call:
```

```
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + b + poly(lstat, 2), data = BostonHousing)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.5603  -2.6709  -0.3071   1.9469  25.0085
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      27.973362   4.476864   6.248 8.97e-10 ***
## crim             -0.149072   0.030006  -4.968 9.34e-07 ***
## zn                0.021281   0.012500   1.703 0.089291 .
## chas1             2.589821   0.775307   3.340 0.000900 ***
## nox             -13.534522   3.229619  -4.191 3.30e-05 ***
## rm                3.233174   0.372802   8.673  < 2e-16 ***
## dis              -1.357892   0.169051  -8.032 7.09e-15 ***
## rad               0.271744   0.057599   4.718 3.11e-06 ***
## tax              -0.009546   0.003068  -3.111 0.001970 **
## ptratio          -0.790820   0.118089  -6.697 5.82e-11 ***
## b                 0.008174   0.002429   3.365 0.000824 ***
## poly(lstat, 2)1 -99.489182   7.069954 -14.072  < 2e-16 ***
## poly(lstat, 2)2  48.752080   4.716090  10.337  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.298 on 493 degrees of freedom
## Multiple R-squared:  0.7868, Adjusted R-squared:  0.7816
## F-statistic: 151.6 on 12 and 493 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2, 2))
plot(sub.mod)
```



Notice that the R-squared value improved from 0.7406 to 0.7868 after adding the polynomial term. Notice also that the U-shape from the "Residual vs. Fitted" plot disappears after adding this polynomial term. However, "Normal Q-Q" plot still has a right tail so we consider a log transformation of our response.

We perform regsubsets one last time to check if adding the polynomial lstat term and the addition of the log term changes the number of our desired predictors.

```
library(leaps)
regsubs.full=regsubsets(log(medv)~crim+zn+chas+nox+rm+dis+rad+tax+ptratio+b+poly(lstat,2),data=BostonHou
regsubs.sum=summary(regsubs.full)

par(mfrow=c(1,2))
plot(regsubs.sum$adjr2 ,xlab="Number of Variables ", ylab="Adjusted R^2",type="l")
which.max(regsubs.sum$adjr2)
```

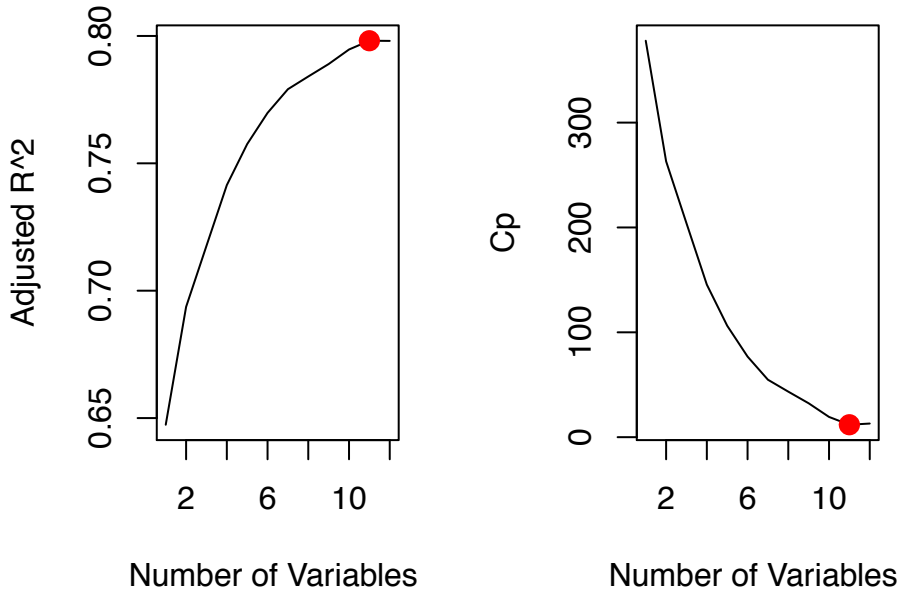
```
## [1] 11
```

```
points(x=11,y=regsubs.sum$adjr2[11], col="red",cex=2,pch=20)

plot(regsubs.sum$cp, xlab="Number of Variables ",ylab="Cp", type='l')
which.min(regsubs.sum$cp)
```

```
## [1] 11
```

```
points(11,regsubs.sum$cp[11], col="red", cex=2, pch=20)
```



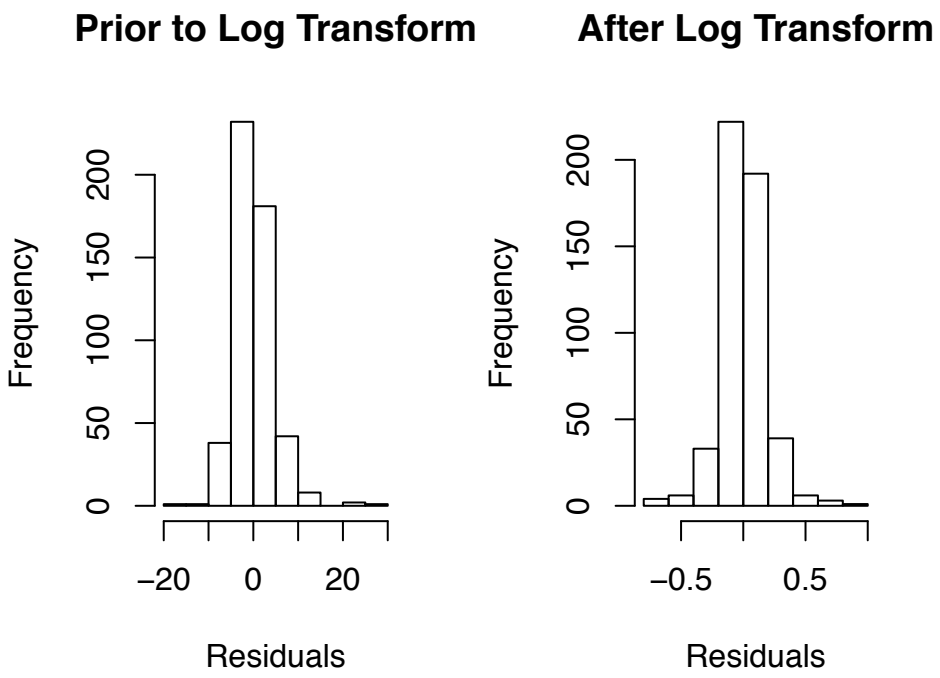

```
coef(regsubs.full, 11)
```

```
##     (Intercept)            crim            chas1              nox
##    3.6852225648   -0.0112268084    0.1018626185   -0.6389554481
##              rm             dis             rad             tax
##    0.0785847782   -0.0450835960    0.0125929148   -0.0004791116
##         ptratio               b poly(lstat, 2)1 poly(lstat, 2)2
##   -0.0350966618    0.0003855614   -4.9789459686    1.2157645104
```

```r
#We get the 11 variable model as our most optimal model, so drop zn.
M<-lm(log(medv)~crim+chas+nox+rm+dis+rad+tax+ptratio+b+poly(lstat,2), data=BostonHousing)
summary(M)
```

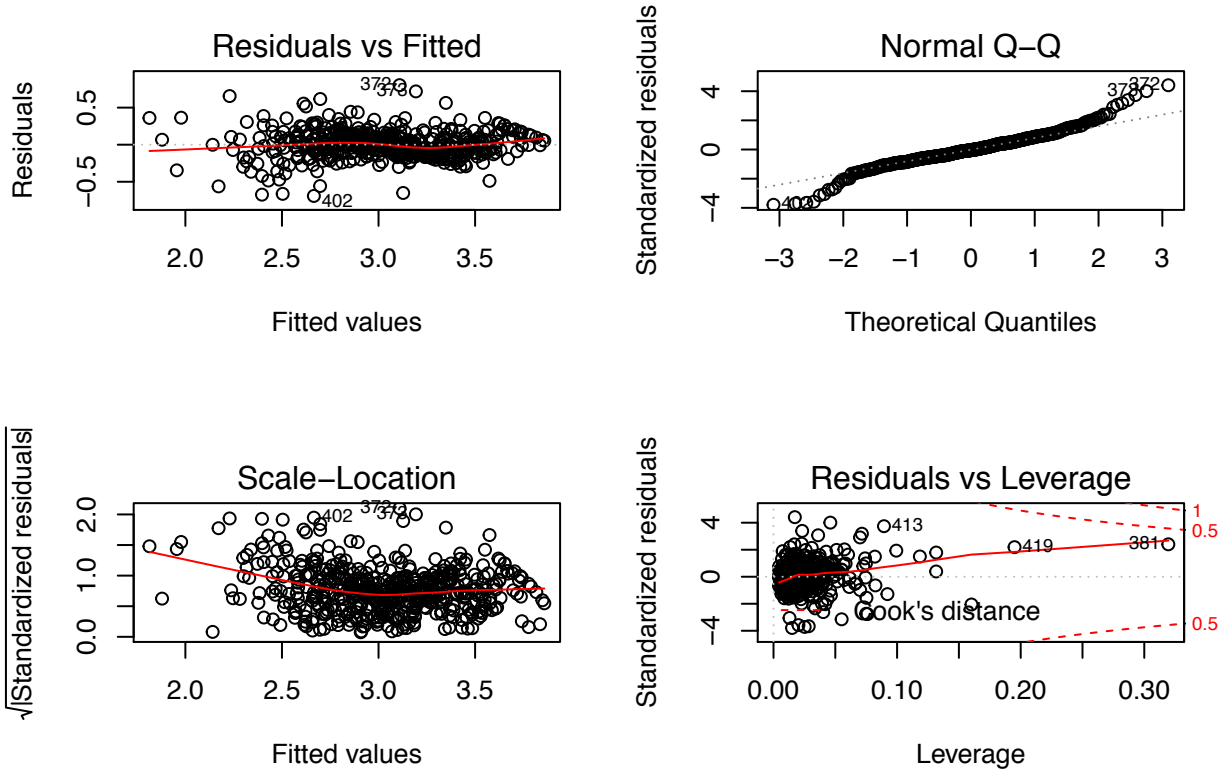
```
##
## Call:
## lm(formula = log(medv) ~ crim + chas + nox + rm + dis + rad +
##     tax + ptratio + b + poly(lstat, 2), data = BostonHousing)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.69111 -0.09974 -0.00730  0.09518  0.80291
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.6852226  0.1911889  19.275  < 2e-16 ***
## crim             -0.0112268  0.0012795  -8.775  < 2e-16 ***
## chas1             0.1018626  0.0331288   3.075 0.002223 **
## nox              -0.6389554  0.1375574  -4.645 4.37e-06 ***
## rm                0.0785848  0.0158262   4.965 9.45e-07 ***
## dis              -0.0450836  0.0062391  -7.226 1.90e-12 ***
## rad               0.0125929  0.0024537   5.132 4.12e-07 ***
## tax              -0.0004791  0.0001283  -3.734 0.000211 ***
## ptratio          -0.0350967  0.0047981  -7.315 1.05e-12 ***
## b                 0.0003856  0.0001038   3.715 0.000227 ***
## poly(lstat, 2)1 -4.9789460  0.3018995 -16.492  < 2e-16 ***
## poly(lstat, 2)2  1.2157645  0.1978460   6.145 1.65e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1837 on 494 degrees of freedom
## Multiple R-squared:  0.8025, Adjusted R-squared:  0.7981
## F-statistic: 182.5 on 11 and 494 DF,  p-value: < 2.2e-16
```

After the long transformation of medv, notice that our R-squared improved from 0.7868 to 0.8025, despite the fact that we also dropped a predictor term of zn.



Notice also that the histogram of our residual more closely resembles a normal distribution.

DIAGNOSTICS OF CHOSEN MODEL

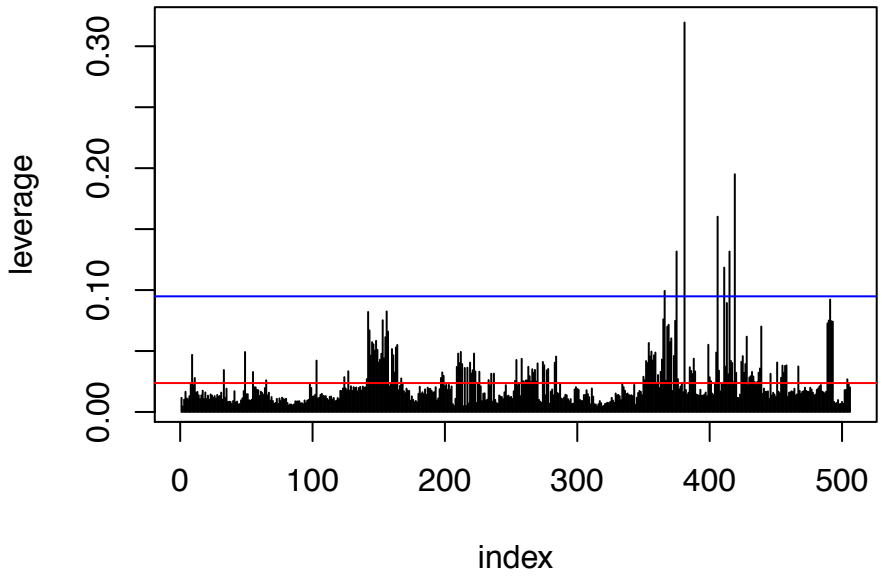


From the "Residual vs. Fitted" and "Scale-Location" plots, we can say that our data is homoscedastic (i.e. constant variance) because there is no 'funnel shape' in the residual plot. The non-linearity issue

encountered earlier has also been fixed with the polynomial. term. We can see that there are 3 points that have a large fitted value, relative to all the other fitted values. These points are 402, 373, and 372. For the most part, the "Normal Q-Q" plot follows a straight line with the exception of some light tails at the end. This "Normal Q-Q" plot is an improvement from the one we had from our full model in the beginning of the analysis, which had a large right tail. Looking at the "Residual vs. Leverage" plot, we see a couple of observations with large leverage values. I plot these leverages along with their average. Those that greatly exceed 2*(average of the leverages) can be labeled as high leverage points. These points are 419 and 381.

Leverages

```
Minf = influence(M)
#sort(Minf$hat)
par(mfrow=c(1,1))
plot(Minf$hat, xlab="index", ylab="leverage", type = "h")
abline(h= (11+1) /nrow(BostonHousing), col="red") #Average leverage
abline(h=4*(12/nrow(BostonHousing)), col="blue")
```

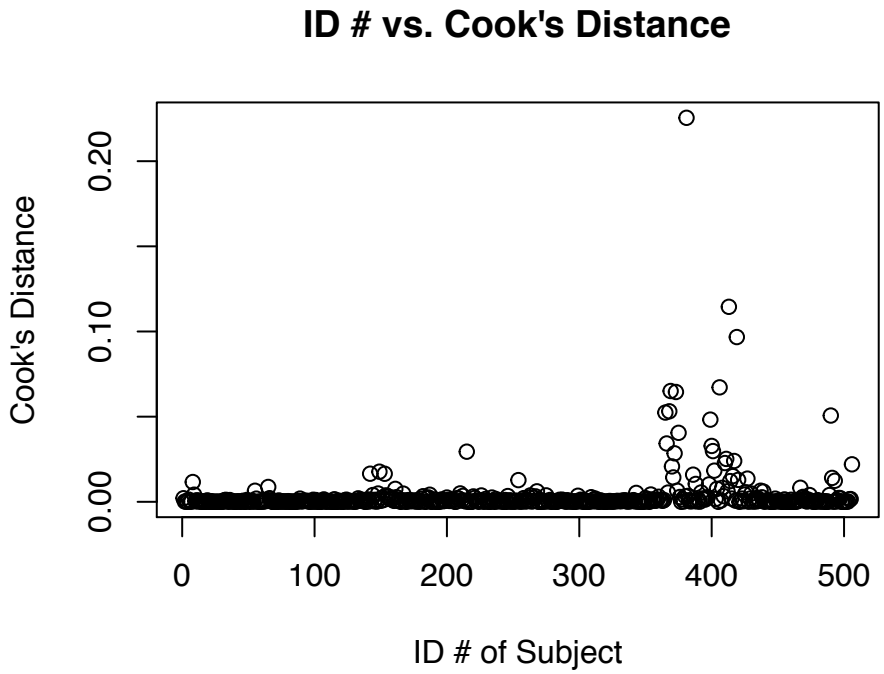



```
which(Minf$hat > 4*(12/nrow(BostonHousing)))
```

```
## 366 375 381 406 411 415 419
## 366 375 381 406 411 415 419
```

```
#We see that observations 366 375 381 406 411 415 419 have very high leverages.
```

Cook's Distance We plot the Cook's Distance of each observation and see that none of the observations exceed the 0.5 mark. Recall that because the regression must pass through the centroid, points that lie far from the centroid have greater leverage, and their leverage increases if there are fewer points nearby. As a result, leverage reflects both the distance from the centroid and the isolation of a point. The Cook's distance measures how much the regression would change if a point was deleted. Cook's distance is increased by leverage AND by large residuals: a point far from the centroid with a large residual can severely distort the regression.




```
##     373     369     406     419     413     381
## 0.0645 0.0651 0.0672 0.0967 0.1145 0.2255
```

Notice that the points 381, 413, and 419 stand out. However, none of them exceed the 0.5 Cook's distance mark.

Correlation of Errors

No time series structure to the data so no point in checking for correlated errors.

Outlier t-test

Next, for each observation, I calculate the p-value for testing whether the ith subject is an outlier based on the standardized predicted residual.

```
jack=rstudent(M)
#Compute the p-value for all standardized predicted residuals using the t-distribution
p.val=pt(jack, df=nrow(BostonHousing)-11-2)

#How many outliers do we get with alpha=0.05
sum(p.val< 0.05) #17 outlier
```

```
## [1] 17
```

```
which(p.val< 0.05)
```

```
##  343  365  386  388  398  399  400  401  402  404  406  416  417  420  427  490  506
##  343  365  386  388  398  399  400  401  402  404  406  416  417  420  427  490  506
```

```
#Use Bonferroni correction because of the issue with multiple testing.
sum(p.val< (0.05/nrow(BostonHousing))) #2 outlier
```

```
## [1] 2
```

```
which(p.val< (0.05/ nrow(BostonHousing))) #Observation 401 and 402.
```

```
## 401 402
## 401 402
```

```
#We get that observation 401 and 402 is an outlier, after the Bonferroni correction.
BostonHousing[401:402,]
```

```
##        crim zn indus chas   nox    rm age    dis rad tax ptratio     b
## 401 25.0461  0  18.1    0 0.693 5.987 100 1.5888  24 666    20.2 396.9
## 402 14.2362  0  18.1    0 0.693 6.343 100 1.5741  24 666    20.2 396.9
##     lstat medv
## 401 26.77  5.6
## 402 20.32  7.2
```

```
#Remove the Identified Outlier 401, 402
updated.M=lm(log(medv)~crim+chas+nox+rm+dis+rad+tax+ptratio+b+poly(lstat,2),
             data=BostonHousing[-c(401:402),])
summary(updated.M)
```

```
##
## Call:
## lm(formula = log(medv) ~ crim + chas + nox + rm + dis + rad +
##     tax + ptratio + b + poly(lstat, 2), data = BostonHousing[-c(401:402),
##     ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68208 -0.09937 -0.00994  0.09272  0.79519
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.6264135  0.1862759  19.468  < 2e-16 ***
## crim             -0.0107555  0.0012483  -8.616  < 2e-16 ***
## chas1             0.0986104  0.0322255   3.060 0.002334 **
## nox              -0.6348250  0.1337848  -4.745 2.74e-06 ***
## rm                0.0839802  0.0154232   5.445 8.19e-08 ***
## dis              -0.0448149  0.0060681  -7.385 6.55e-13 ***
## rad               0.0129048  0.0023872   5.406 1.01e-07 ***
## tax              -0.0004793  0.0001248  -3.840 0.000139 ***
## ptratio          -0.0351216  0.0046665  -7.526 2.51e-13 ***
## b                 0.0004465  0.0001016   4.397 1.35e-05 ***
## poly(lstat, 2)1 -4.8346222  0.2931063 -16.494  < 2e-16 ***
## poly(lstat, 2)2  1.2004320  0.1923433   6.241 9.39e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1786 on 492 degrees of freedom
## Multiple R-squared:  0.8074, Adjusted R-squared:  0.8031
## F-statistic: 187.6 on 11 and 492 DF,  p-value: < 2.2e-16
```


```
library(MASS)
#No more suggested variables to delete according to Backward Selection Method
stepAIC(updated.M, direction = "backward")
```

```
## Start:  AIC=-1724.43
## log(medv) ~ crim + chas + nox + rm + dis + rad + tax + ptratio +
##     b + poly(lstat, 2)
##
##                  Df Sum of Sq    RSS     AIC
## <none>                        15.697 -1724.4
## - chas            1    0.2987 15.996 -1716.9
## - tax             1    0.4705 16.168 -1711.5
## - b               1    0.6168 16.314 -1707.0
## - nox             1    0.7184 16.415 -1703.9
## - rad             1    0.9324 16.629 -1697.3
## - rm              1    0.9459 16.643 -1696.9
## - dis             1    1.7401 17.437 -1673.4
## - ptratio         1    1.8073 17.504 -1671.5
## - crim            1    2.3685 18.065 -1655.6
## - poly(lstat, 2)  2    8.9325 24.630 -1501.4

##
## Call:
## lm(formula = log(medv) ~ crim + chas + nox + rm + dis + rad +
##     tax + ptratio + b + poly(lstat, 2), data = BostonHousing[-c(401:402),
##     ])
##
## Coefficients:
##     (Intercept)             crim            chas1              nox
##       3.6264135       -0.0107555        0.0986104       -0.6348250
##              rm              dis              rad              tax
```

```
##       0.0839802       -0.0448149        0.0129048       -0.0004793
##         ptratio                b  poly(lstat, 2)1  poly(lstat, 2)2
##      -0.0351216        0.0004465       -4.8346222        1.2004320
```

```
#Confidence Intervals of Parameters
confint(full.mod)
```

```
##                     2.5 %        97.5 %
## (Intercept)  26.432226009  46.486750761
## crim         -0.172584412  -0.043438304
## zn            0.019448778   0.073392139
## indus        -0.100267941   0.141385193
## chas1         0.993904193   4.379563446
## nox         -25.271633564 -10.261588893
## rm            2.988726773   4.631003640
## age          -0.025262320   0.026646769
## dis          -1.867454981  -1.083678710
## rad           0.175692169   0.436406789
## tax          -0.019723286  -0.004945902
## ptratio      -1.209795296  -0.695699168
## b             0.004034306   0.014589060
## lstat        -0.624403622  -0.425113133
```

```
confint(updated.M)
```

```
##                         2.5 %        97.5 %
## (Intercept)      3.2604191040  3.9924078421
## crim            -0.0132081976 -0.0083028355
## chas1            0.0352939598  0.1619269360
## nox             -0.8976850384 -0.3719650143
## rm               0.0536767985  0.1142835432
## dis             -0.0567375263 -0.0328922225
## rad              0.0082144968  0.0175951055
## tax             -0.0007245020 -0.0002340643
## ptratio         -0.0442902147 -0.0259529298
## b                0.0002469892  0.0006460505
## poly(lstat, 2)1 -5.4105167724 -4.2587276768
## poly(lstat, 2)2  0.8225163657  1.5783476555
```