Boston Housing Prices

Shuai Zheng

Introduction

The data to be analyzed were collected by Harrison and Rubinfeld in 1978 for the purpose of discovering whether or not the value of houses in Boston. The original dataset is from CMU StatLib Datasets Archive-boston (http://lib.stat.cmu.edu/datasets/boston). **This report seeks to discover the most suitable explanatory variables to explain median price of houses in Boston. The R programming language is used to conduct this analysis.**

Exploratory Data Analysis

The data consist of 506 observations and 12 constant variables and 2 categorical variables (chas and rad). Especially, medv is the response variable while the other 13 variables are possible predictors. There is no missing value or obvious outliers in the dataset. The ultimate goal of our analysis is to fit a regression model that best explains the variation in medv.

```
> summary(df)
      crim                zn             indus            chas              nox               rm
 Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000   Min.   :0.3850   Min.   :3.561
 1st Qu.: 0.08204   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000   1st Qu.:0.4490   1st Qu.:5.886
 Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000   Median :0.5380   Median :6.208
 Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917   Mean   :0.5547   Mean   :6.285
 3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000   3rd Qu.:0.6240   3rd Qu.:6.623
 Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000   Max.   :0.8710   Max.   :8.780
      age              dis              rad              tax           ptratio          black
 Min.   :  2.90   Min.   : 1.130   Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32
 1st Qu.: 45.02   1st Qu.: 2.100   1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38
 Median : 77.50   Median : 3.207   Median : 5.000   Median :330.0   Median :19.05   Median :391.44
 Mean   : 68.57   Mean   : 3.795   Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67
 3rd Qu.: 94.08   3rd Qu.: 5.188   3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23
 Max.   :100.00   Max.   :12.127   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
```

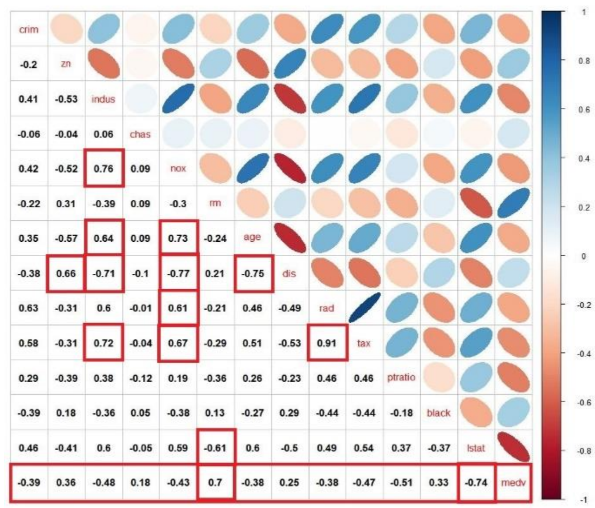
Variables	in order:
crim	per capita crime rate by town
zn	proportion of residential land zoned for lots over 25,000 sq.ft.
indus	proportion of non-retail business acres per town
chas	Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
nox	nitric oxides concentration (parts per 10 million)
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 $10,000
ptratio	pupil-teacher ratio by town
b	1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town
lstat	% lower status of the population
medv	Median value of owner-occupied homes in $1000's

Correlation checks for all the variables

From this graph, we can find the highest correlations is between indus and nox, as well as those between tax and rad and tax and indus. These correlations are reasonable that nitrogen oxide levels as well as tax levels are highest near industrial areas.

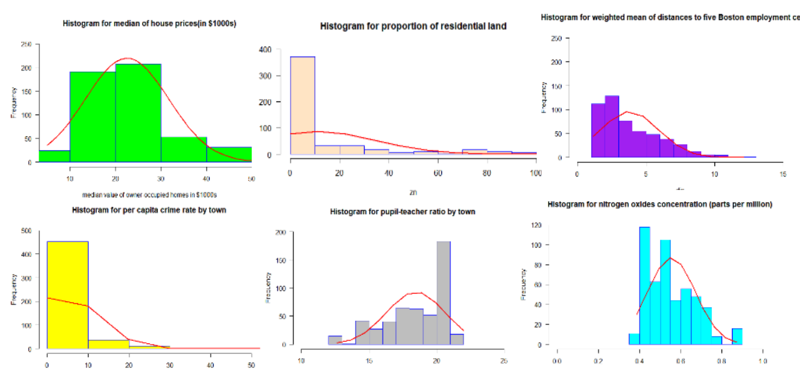
Related to medv itself, it is found that rm(average number of rooms) has the highest positive correlation, while ptratio(pupil-teacher ratio) and lstat have the highest negative correlations.

Therefore, we can remove rad which has highest correlation with tax, and we are less interested in tax in this case. In addition, we should put more efforts on rm, ptratio and lstat variables because of their stronger correlation with our target variable-medv.



Data Visualization for the raw skewed data

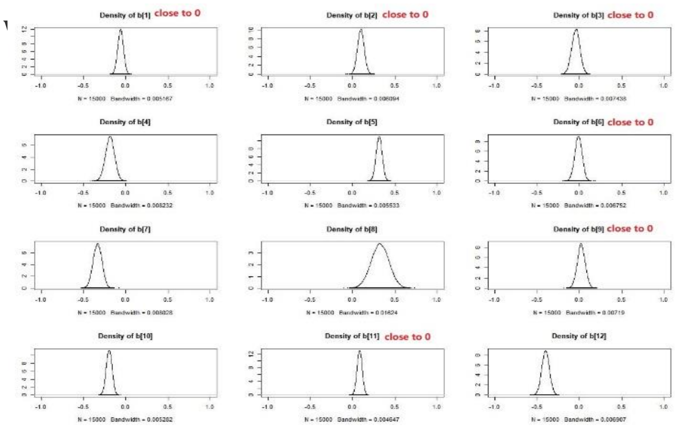
We know the skewed data will have enormous impact on the accuracy of the model. So, these variables may require transformations to better fit the model. After doing graphically examine the whole dataset to understand how it is distributed, we find the variables crim, dis, nox, zn are right skewed, making log transformations appropriate, and the left skewed distribution of ptratio suggests that squaring it will be better. We can observe accuracy variances among models with or without transformations and see how it improve models.

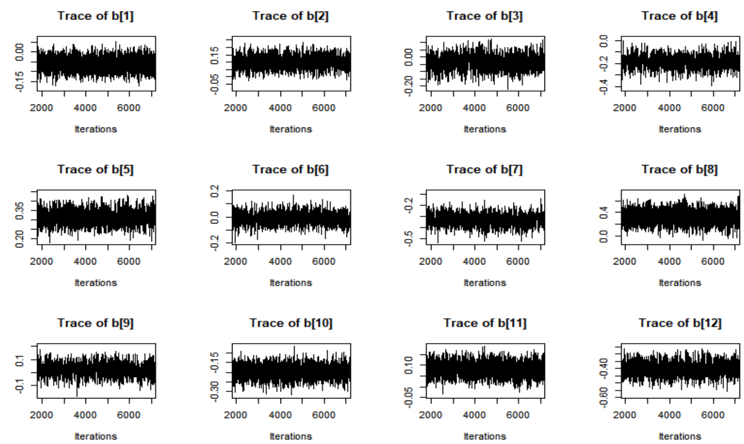


Variable selection

We scale the raw continuous variables (standardization: for each continuous variable, subtract the mean and divide by the standard deviation for that variable in the original data set used to fit the model) Then use a linear model where the priors for the β coefficients is the double exponential (or Laplace) distribution and if there is not a strong signal for a parameter.

```
> autocorr.diag(mod1_sim)
              b[1]         b[2]       b[3]      b[4]       b[5]        b[6]       b[7]          b[8]
Lag 0  1.000000000  1.000000000 1.00000000 1.0000000 1.00000000  1.00000000 1.00000000  1.0000000000
Lag 1  0.544487835  0.658480729 0.77436648 0.8176165 0.60232029  0.73511724 0.80881349  0.3220621321
Lag 5  0.078464014  0.180696156 0.25835121 0.3567220 0.17912223  0.22549667 0.33840123  0.0001970667
Lag 10 0.001512004  0.002491190 0.04451964 0.1023769 0.02458552  0.02543538 0.09098848 -0.0074293605
Lag 50 0.003903964 -0.002803374 0.00748289 0.0238622 0.01516400 -0.00197702 0.03240703  0.0113107255
              b[9]        b[10]       b[11]        b[12]          int         sig
Lag 0   1.00000000  1.000000000 1.000000000  1.000000000  1.000000000 1.000000000
Lag 1   0.77107766  0.558538206 0.421912126  0.741017475  0.084053423 0.026967019
Lag 5   0.27720824  0.103291946 0.037645170  0.242395194 -0.009418733 0.015194328
Lag 10  0.06092542  0.020636083 0.001523861  0.024574081  0.004535693 0.003751386
Lag 50 -0.01230903 -0.009863547 0.006181736 -0.001594584  0.008270651 0.006541836
```






```
> dic1
Mean deviance:  798.8
penalty 14
Penalized deviance: 812.8
> summary(mod1_sim)

Iterations = 2001:7000
Thinning interval = 1
Number of chains = 3
Sample size per chain = 5000

1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:

          Mean      SD  Naive SE Time-series SE
b[1]  -0.06167 0.03336 0.0002724      0.0005237
b[2]   0.09800 0.03941 0.0003218      0.0007655
b[3]  -0.04340 0.04803 0.0003921      0.0010709
b[4]  -0.18730 0.05367 0.0004382      0.0013707
b[5]   0.31445 0.03592 0.0002933      0.0006979
b[6]  -0.01129 0.04375 0.0003572      0.0009224
b[7]  -0.33504 0.05182 0.0004231      0.0013037
b[8]   0.32506 0.10483 0.0008559      0.0012108
b[9]   0.02144 0.04693 0.0003832      0.0010610
b[10] -0.19716 0.03410 0.0002784      0.0005513
b[11]  0.08328 0.02999 0.0002449      0.0003846
b[12] -0.40088 0.04459 0.0003640      0.0009693
int   -0.02292 0.02696 0.0002201      0.0002368
sig    0.58056 0.01855 0.0001515      0.0001587
```

Full variable model:

```
library("rjags")
mod1_string = " model {
for (i in 1:length(y)) {
#likelihood of the data
y[i] ~ dnorm(mu[i], prec)
mu[i] = int + b[1]*crim[i] + b[2]*zn[i] + b[3]*indus[i] + b[4]*nox[i] + b[5]*rm[i] + b[6]*age[i]+ b[7]*dis[i]
+ b[8]*chas[i]+ b[9]*tax[i]+ b[10]*ptratio[i]+b[11]*black[i]+ b[12]*lstat[i]}

int ~ dnorm(0.0, 1.0/1.0e6) #non-informative prior with large variance
for (j in 1:12) {
b[j] ~ ddexp(0.0, sqrt(2.0)) # has variance 1.0
}

prec ~ dgamma(3/2.0, 3*10.0/2.0)
sig2 = 1.0 / prec
#gave a prior to the precision: a prior for sigma squared
sig = sqrt(sig2)
} "
```

From the results above, we notice that the absolute beta parameter values of crim, zn, indus, age, tax and black are less t han 0.1. Therefore, we can remove these predictors that are not statistically significant from the model. Apparently, this original full model is not a better choice since it has huge penalized deviance(DIC: 812.8).

Model Comparison and results review

1. Fit a model with removing insignificant parameters without transformation or scaling

```
> dic2
Mean deviance:  3053
penalty 9.211
Penalized deviance: 3062
> #these results are for a regression model: logarithm of infant mortality to the logarithm of income.
> (pm_coef = colMeans(mod2_csim))
     b[1]       b[2]       b[3]       b[4]       b[5]       b[6]       b[7]        sig
38.165268 -19.371448   4.054604  -1.174151   3.245671  -1.025498  -0.569720   4.934129
```

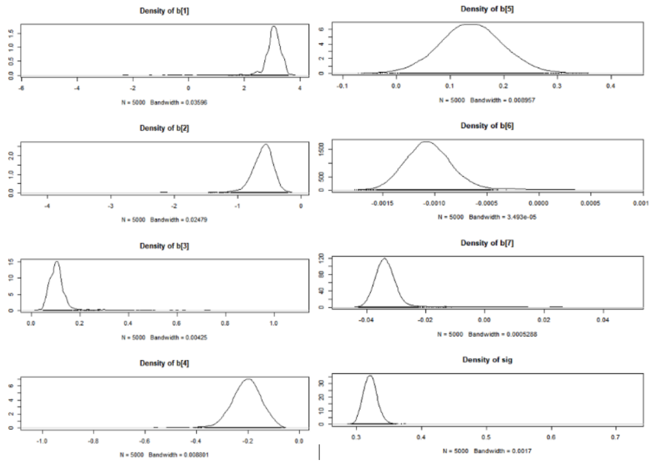
```
############Model without log and squares#####################
library("rjags")
mod2_string = " model {
for (i in 1:n) {
#likelihood of the data
y[i] ~ dnorm(mu[i], prec)
#add the linear model: mu[i] is linear
mu[i] = b[1]+ b[2]*nox[i] + b[3]*rm[i] + b[4]*dis[i]
+ b[5]*chas[i]+ b[6]*ptratio[i]+ b[7]*lstat[i]
}

for (i in 1:7) {
b[i] ~ dnorm(0.0, 1.0/1.0e6) #non-informative prior with large variance
}

prec ~ dgamma(3/2.0, 3*10.0/2.0)
sig2 = 1.0 / prec
#gave a prior to the precision: a prior for sigma squared
sig = sqrt(sig2)
} "
```

From the results above, we know the model without transformation or scaling gives us terrible results (very large DIC: 3062), although we have removed the insignificant variables.

Therefore, we should conduct the log transformation for the left skewed variables: nox, dis and take squares for the right skewed variable: ptratio.

2. Fit a model with removing insignificant parameters and log & square transformation

```
############Model with log and squares######################
df1<-df[,c(4,6,13)]
df1$logmedv = log(df$medv)
df1$lognox = log(df$nox)
df1$logdis = log(df$dis)
df1$sqrptratio = (df$ptratio)*(df$ptratio)

library("rjags")
mod2_string = " model {
for (i in 1:n) {
#likelihood of the data
y[i] ~ dnorm(mu[i], prec)
#add the linear model: mu[i] is linear
mu[i] = b[1]+ b[2]*nox[i] + b[3]*rm[i] + b[4]*dis[i]
+ b[5]*chas[i]+ b[6]*ptratio[i]+ b[7]*lstat[i]
}

# prior of the coefficients
for (i in 1:7) {
b[i] ~ dnorm(0.0, 1.0/1.0e6) #non-informative prior with large variance
}

prec ~ dgamma(3/2.0, 3*10.0/2.0)
sig2 = 1.0 / prec
#gave a prior to the precision: a prior for sigma squared
sig = sqrt(sig2)
} "
```




```
> dic2
Mean deviance:  -3.98
penalty 7.313
Penalized deviance: 3.333
> #these results are for a regression model: logarithm of infant mortality to the logarithm of income.
> (pm_coef = colMeans(mod2_csim))
        b[1]         b[2]         b[3]         b[4]         b[5]         b[6]         b[7]          sig
 2.873398711 -0.656308885  0.122254081 -0.214851576  0.140265749 -0.001028944 -0.032142626  0.326104438
```

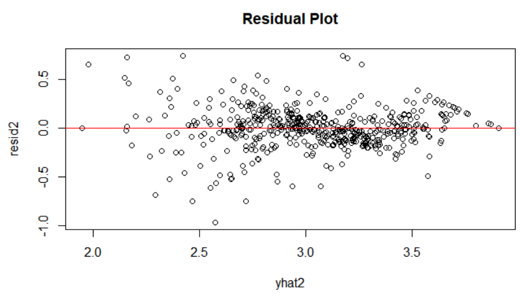
```
> mean(resid2)
[1] -4.368163e-05
> sd(resid2) # standard deviation of residuals
[1] 0.2075334
> mean(abs(resid2)>mean(resid2)+2*sd(resid2))
[1] 0.06126482
```

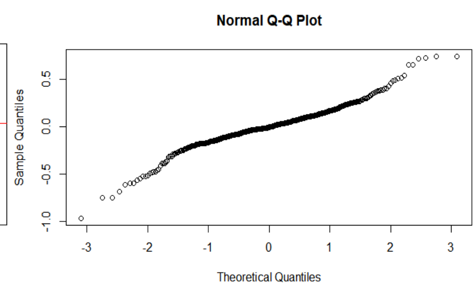
From the results shown above, we delight to observe that the DIC value of this model is significantly drop to 3.33. This has demonstrated that transformation can be used to reduce skewness and applied to improve the model.

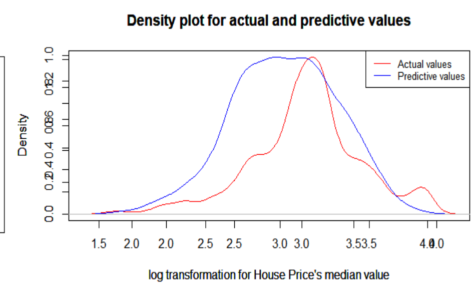
The linear equation we get for this model(This is our preferred model for mu- the normal distribution's mean of y)

$$\log(mu) = 2.873 - 0.666 * \log(nox) + 0.122 * \mathrm{rm} - 0.215 * \log(\mathrm{dis}) + 0.140 * \mathrm{chas} - 0.001 * \mathrm{ptratio}^2 - 0.032\mathrm{lstat}$$

Check the residuals



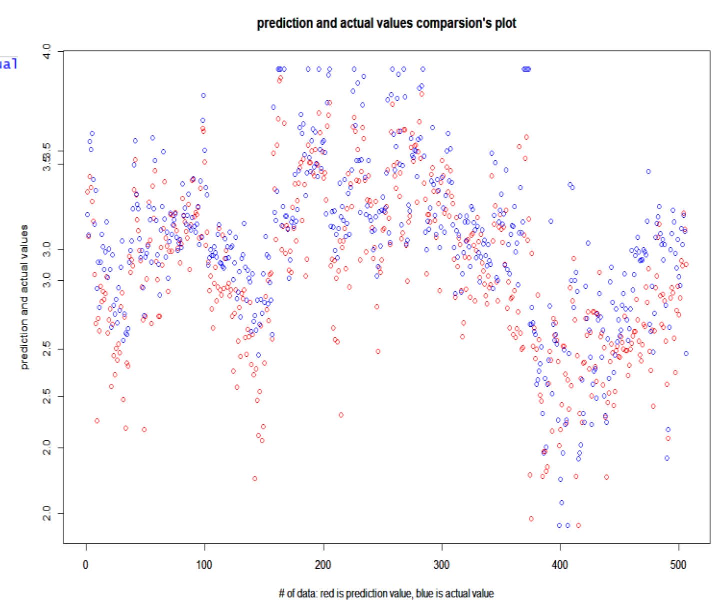




<u>Top 5 outliers</u>

```
> df1[rownames(df1)[order(abs(resid2), decreasing=TRUE)[1:5]],] # largest absolute residual
    chas    rm lstat   logcrim sqrtzn  black  logmedv     lognox    logdis sqrptratio
406    0 5.683 22.98  4.218342      0 384.97 1.609438 -0.3667253 0.3544525     408.04
402    0 6.343 20.32  2.655788      0 396.90 1.974081 -0.3667253 0.4536837     408.04
401    0 5.987 26.77  3.220718      0 396.90 1.722767 -0.3667253 0.4629790     408.04
215    0 5.412 29.55 -1.239427      0 348.93 3.165475 -0.7153928 1.2774556     345.96
372    0 6.216  9.53  2.222708      0 366.15 3.912023 -0.4604494 0.1562342     408.04
```

The residuals look pretty good (no obvious patterns, shapes, straight lines for Q-Q plot) except for several outliers. After double check them, we find the values are correct and these outliers are part of data and should not be removed, we can try to use e t distribution which is similar to the normal distribution, but it has thicker tails which can accommodate outliers. We can build a model with a likelihood contributing to t distribution. We might assign the degrees of freedom to prior exponential distribution plus 2 to guarantee existence of mean and variance.



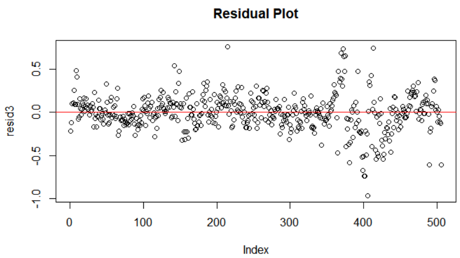
Bulid a model with likelihood contributes to t-distribution

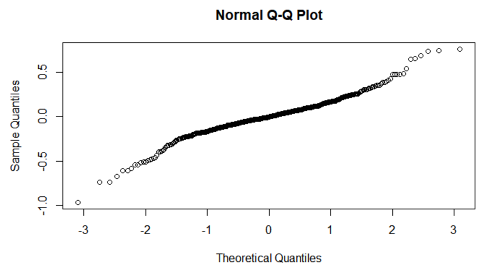
```
mod3_string = " model {
for (i in 1:length(y)) {
y[i] ~ dt( mu[i], tau, df )
mu[i] = b[1]+ b[2]*nox[i] + b[3]*rm[i] + b[4]*dis[i]
+ b[5]*chas[i]+ b[6]*ptratio[i]+ b[7]*lstat[i]
}

for (i in 1:7) {
b[i] ~ dnorm(0.0, 1.0/1.0e6)
}

df = nu + 2.0 # we want degrees of freedom > 2 to guarantee existence of mean and variance
nu ~ dexp(1.0)

tau ~ dgamma(3/2.0, 3*10.0/2.0) # tau is close to, but not equal to the precision
sig = sqrt( 1.0 / tau * df / (df - 2.0) ) # standard deviation of errors
} "
```

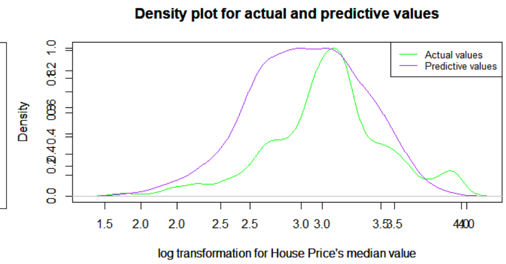
```
> dic3
Mean deviance:  7.1
penalty 6.917
Penalized deviance: 14.02
> pm_coef = colMeans(mod3_csim)
> pm_coef3 = colMeans(mod3_csim)
> pm_coef3
       b[1]         b[2]         b[3]         b[4]         b[5]         b[6]         b[7]           nu
3.038643293 -0.568225826  0.106709223 -0.198585629  0.133814135 -0.001065789 -0.033665150 10.073978997
        sig
0.347258805
```

```
> mean(resid3)
[1] -0.0007534469
> sd(resid3) # standard deviation of residuals
[1] 0.2070802
> mean(abs(resid3)>mean(resid3)+2*sd(resid3))
[1] 0.06126482
```

Check the residuals

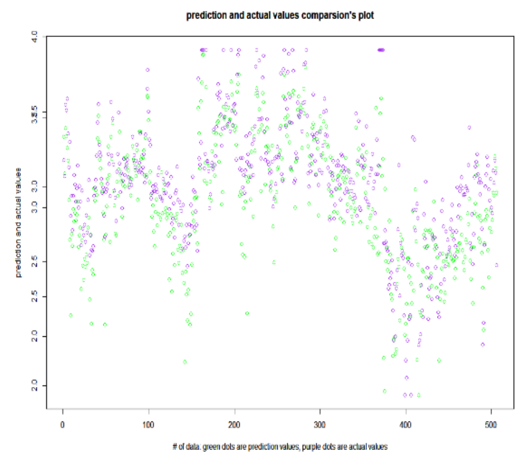






<u>Top 5 outliers</u>

```
> df1[rownames(df1)[order(abs(resid3), decreasing=TRUE)[1:5]],] # largest absolute residual value
    chas    rm lstat   logcrim sqrtzn  black  logmedv     lognox    logdis sqrptratio
406    0 5.683 22.98  4.218342      0 384.97 1.609438 -0.3667253 0.3544525     408.04
215    0 5.412 29.55 -1.239427      0 348.93 3.165475 -0.7153928 1.2774556     345.96
402    0 6.343 20.32  2.655788      0 396.90 1.974081 -0.3667253 0.4536837     408.04
413    0 4.628 34.37  2.934442      0  28.79 2.884801 -0.5158382 0.4407679     408.04
372    0 6.216  9.53  2.222708      0 366.15 3.912023 -0.4604494 0.1562342     408.04
```



By checking the DIC for the model, we find the model with likelihood contributes to t distribution doesn't get improved. Therefore, we might add additional covariates that may explain the outliers. We cannot show the results for them due to the limitation of the report length.

Conclusion

Our preferred model represents that there are positive correlations to median house price if the house is next to the Charles River and the average number of rooms. And there are also negative correlations to median house price if the nitric oxides concentration, pupil-teacher ratios, percentage of lower status of population around house increase and the house is close to five Boston employment centers. This will be explored further in the conclusion.

Among those predictors with negative correlations, we know nitric oxides concentration in the air have the most negative impact on median house price. That is while holding other predictors constant, a one unit change in log(nitric oxides concentration)results in 0.666 decreasing in log(median house price). After conducting several statistical techniques were used to eliminate predictors and checking the residuals, our preferred model($\log(mu) = 2.873 - 0.666 * \log(nox) + 0.122 * \mathrm{rm} - 0.215 * \log(\mathrm{dis}) + 0.140 * \mathrm{chas} - 0.001 * \mathrm{ptratio}^2 - 0.032\mathrm{lstat}$) means median house prices are higher in areas with lower nitric oxides concentration, pupil-teacher ratios, and lower density of lower status of population. House prices also tend to be higher closer to the Charles River, and houses with more rooms are pricier.

The most interesting factors to consider are nitrogen oxide levels and distance to the main employment centers. The result shows that people prefer to live far away their place of employment which might be the center of the town with higher nitrogen oxide levels. This makes sense because it is reasonable to suggest that pollution levels are higher as one moves closer to these main employment centers. Moreover, the pollution here is not just nitrogen oxide, but also includes others such as noise or water pollutions. The linear model shows that higher levels of pollution decrease house prices more significantly than distance to employment centers. This suggests that people would prefer to live further away from their work place because the environment there have lower levels of pollution.

Suggestions for current preferred model's further improvement

We can add additional covariates such as log(crim) or log(zn) that may be able to explain the outliers. Besides, we also can build and run more models with more suitable distribution for the models' priors and likelihoods.

In terms of the timeliness of data, the data used for this analysis was collected in 1978 and the pollution levels have risen as time goes by, so we can conduct more research on examining which factors that affect median house pricing in Boston today.