# Statistical Models : Homework 4

Priyanshi Shah and Sourabh Prakash

2023-02-08

## Question 1

In this problem, you practice working with predictor variables that are discrete. Consider the Boston dataset in the package MASS. Take as response the median property value.

```
library(MASS)
library(ggplot2)
library(quantreg)
```

```
## Loading required package: SparseM
```

```
##
## Attaching package: 'SparseM'
```

```
## The following object is masked from 'package:base':
##
##     backsolve
```

```
head(Boston)
```

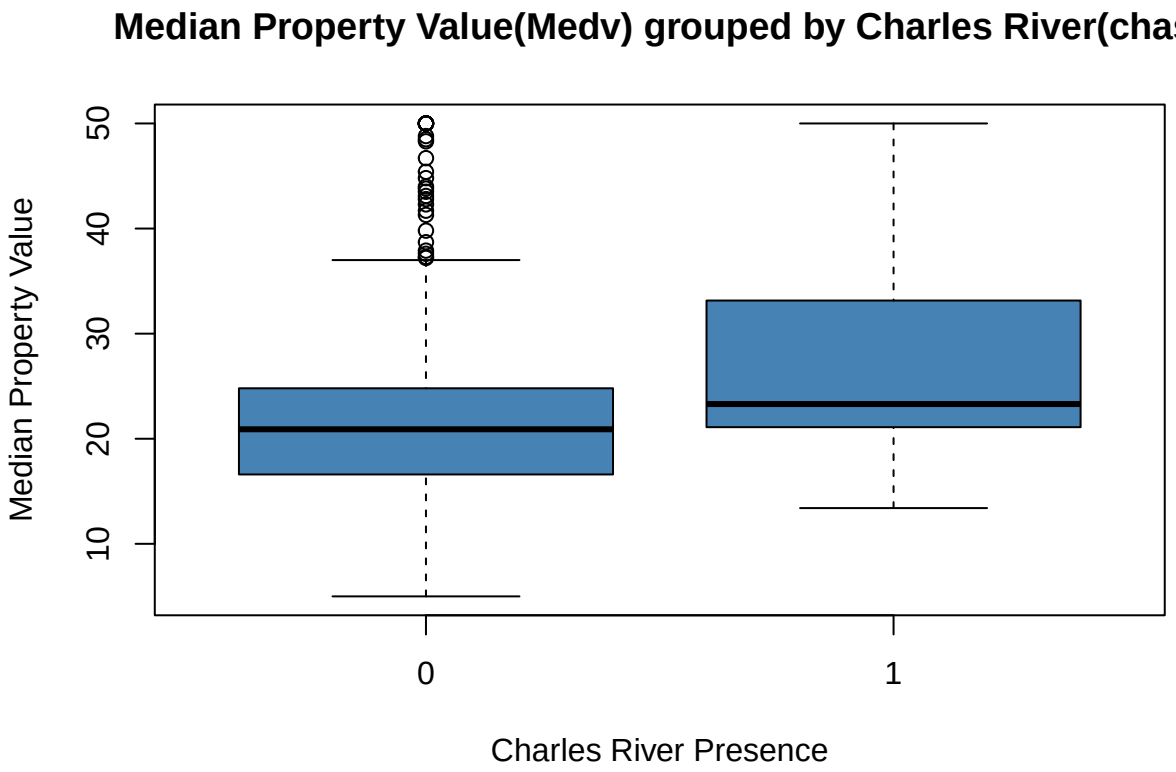
```
##      crim zn indus chas   nox    rm  age    dis rad tax ptratio  black lstat
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.90  4.98
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.90  9.14
## 3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8 392.83  4.03
## 4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3 222    18.7 394.63  2.94
## 5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3 222    18.7 396.90  5.33
## 6 0.02985  0  2.18    0 0.458 6.430 58.7 6.0622   3 222    18.7 394.12  5.21
##   medv
## 1 24.0
## 2 21.6
## 3 34.7
## 4 33.4
## 5 36.2
## 6 28.7
```

### 1(a) For chas variable(Charles river)

Look at side-by-side boxplots for medv where the groups are defined by chas. Comment on what you observe. In particular, compare the different groups visually. Then fit a model explaining medv as a function of chas. Output an ANOVA table. What is the F-test testing? Is the result consistent with the boxplots?

**Creating Boxplot**

```
#Creating a boxplot
boxplot(Boston$medv ~ Boston$chas, col='steelblue',
        main = "Median Property Value(Medv) grouped by Charles River(chas)",
        xlab = "Charles River Presence", ylab = "Median Property Value")
```



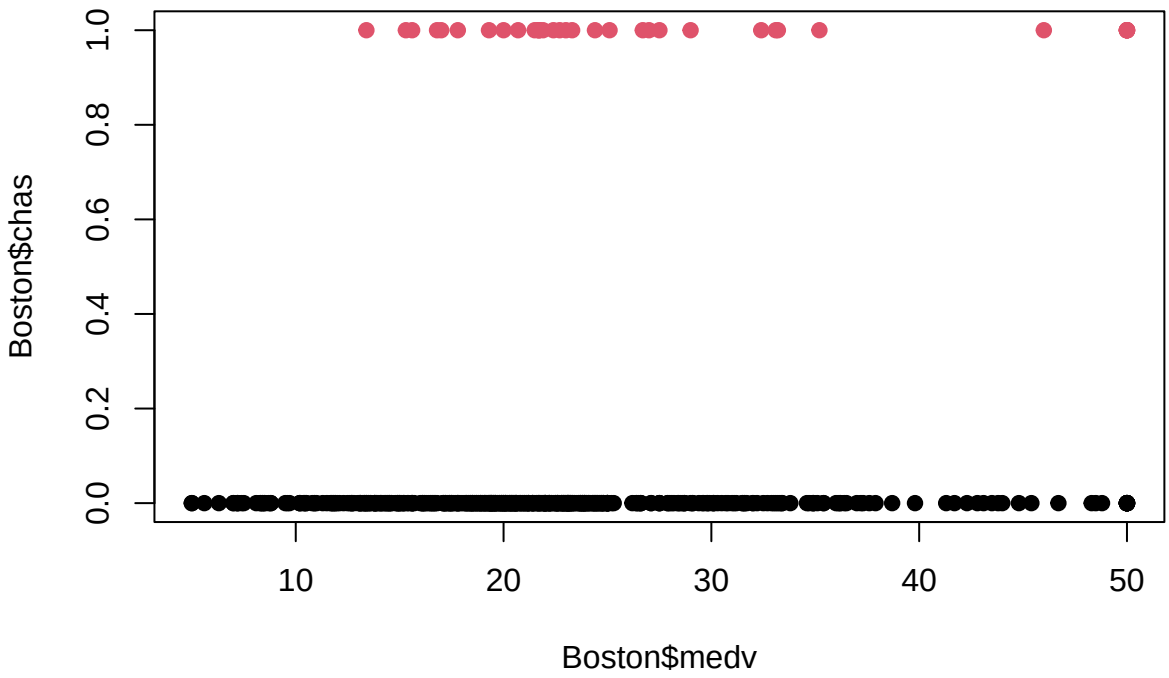

**Inference :**

- For chas = 1, the medv values lie between the 75 th and 25th quantile whereas for chas = 0, there are many values after the 75th quantile.
- The median values for both 0 and 1 category can be seen very closer to each other, approximately ranging near 20 to 25.
- We can see that median property values tend to be higher for properties near the Charles River compared to properties far from the Charles River
- This suggests that proximity to the Charles River has a positive effect on median property values.

```
#Comparing different groups visually
table(Boston$chas)
```

```
##
##   0   1
## 471  35
```

```
plot(Boston$medv,Boston$chas,pch = 19,  col=as.factor(Boston$chas))
```




Inference:

- We can also visually see from the table and graph above that there are more data points for chas = 0 than chas = 1
- We can also observe that for chas = 0 the cluster values are more spread which can also be seen from the boxplot.

**Fitting model**

```
#Fitting  a linear model
model_chas = lm(medv ~ chas, data = Boston)
summary(model_chas)
```

```
##
## Call:
## lm(formula = medv ~ chas, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.094  -5.894  -1.417   2.856  27.906
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.0938     0.4176  52.902  < 2e-16 ***
## chas          6.3462     1.5880   3.996 7.39e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 9.064 on 504 degrees of freedom
## Multiple R-squared:  0.03072,    Adjusted R-squared:  0.02879
## F-statistic: 15.97 on 1 and 504 DF,  p-value: 7.391e-05
```

**Analysis of Variance (ANOVA)**

```
anova(model_chas)$F
```

```
## [1] 15.97151       NA
```

```
anova(model_chas)
```

```
## Analysis of Variance Table
##
## Response: medv
##            Df Sum Sq Mean Sq F value    Pr(>F)
## chas        1   1312 1312.08  15.972 7.391e-05 ***
## Residuals 504  41404   82.15
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Inference : What is the F-test testing? Is the result consistent with the boxplots?

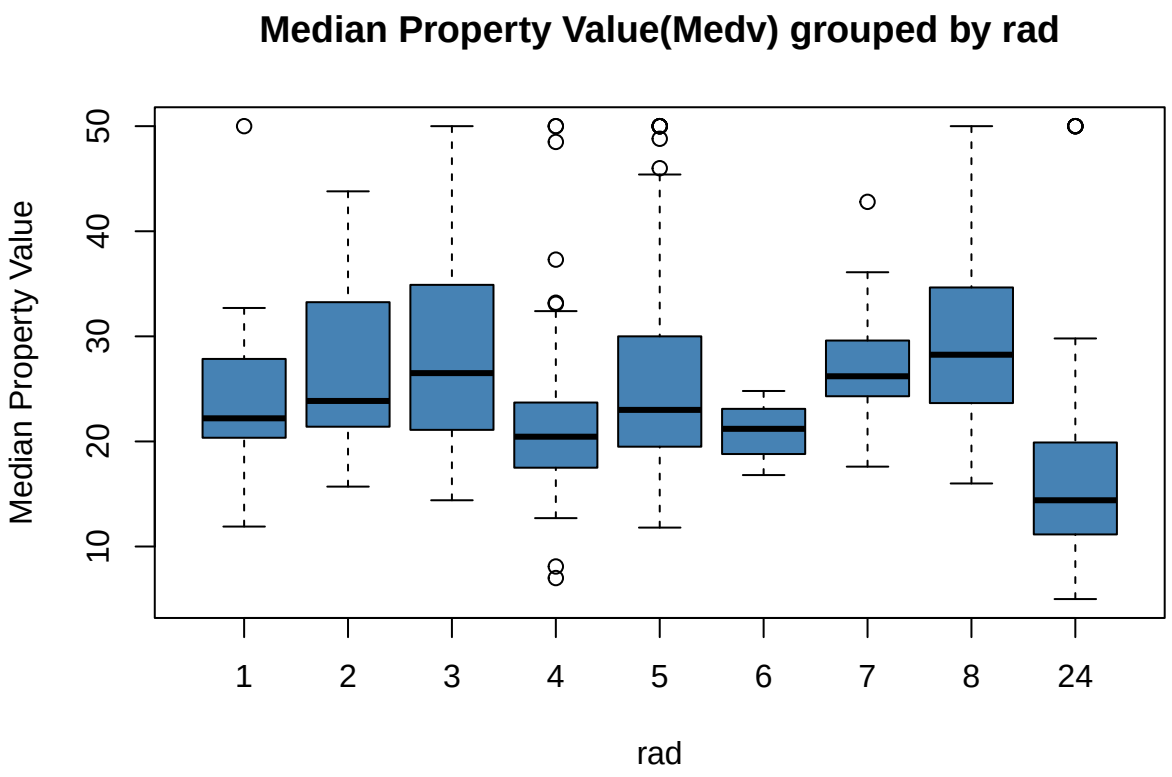
- F-test testing procedure that compares the variability of two or more sets of data
- For F test value =1 the variances are equal which is Null hypothesis and if it is not equal to 1 then there is variability
- We can also see that the F value here is 15.97 which denotes significant variability
- Null hypothesis for this case is : that the coefficients for the **chas** variable are equal to zero. This means that the **chas** variable has no effect on the median property value (**medv**). The null hypothesis assumes that the **chas** variable is not a significant predictor of **medv**. The alternative hypothesis is that the coefficients for the **chas** variable are not equal to zero, meaning that the **chas** variable does have an effect on **medv** and is a significant predictor.
- Since the p - value if smaler than the significance level (0.05), the null hypothesis fails and we conclude that the **chas** variable is significantly related to **medv** meaning that the **chas** variable does have an effect on **medv** and is a significant predictor.
- This result is also consistent with what we can see from the boxplots that Charles river has impact that proximity to the Charles River has a positive effect on median property values.

### 1(b) For Radial highway variable

Repeat with rad in place of chas.

**Creating Boxplot**

```
#Creating boxplot
boxplot(Boston$medv ~ Boston$rad, col='steelblue',
        main = "Median Property Value(Medv) grouped by rad",
        xlab = "rad", ylab = "Median Property Value")
```



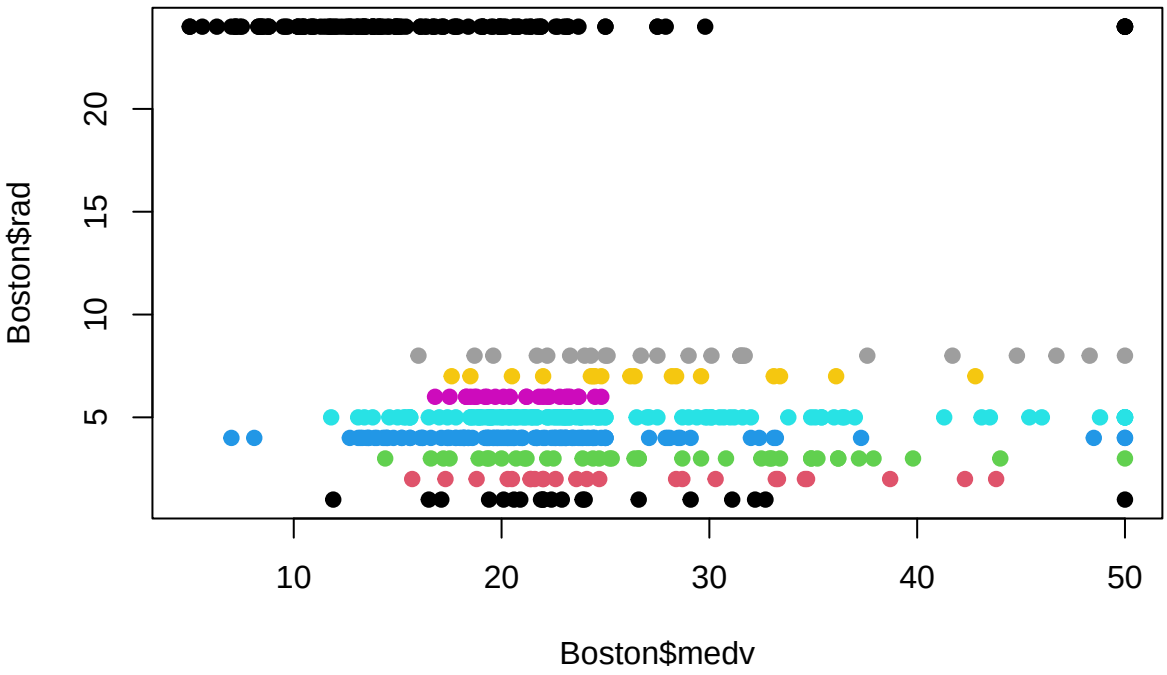

Inference:

- Since, the boxes are of different heights, we can say that the median property value is not the same across all levels of **rad.**
- We can also observe that for higher value of rad (24) , the median is less as compared to the closer rad values(1-8), that is for higher rad value (24 ) we can see that medv values is less. This means that the accessibility to radial highways generally lowers the median property values.

```
#Visually seeing the difference
table(Boston$rad)
```

```
##
##   1   2   3   4   5   6   7   8  24
##  20  24  38 110 115  26  17  24 132
```

```
plot(Boston$medv,Boston$rad,pch = 19,  col=as.factor(Boston$rad))
```




Inference:

- We can see that the data values for rad from 1-8 are quite well spread but for rad =24 are sparsely spread out with one data point above 40.

### Fitting the model

```
#Fitting the model
model_rad = lm(medv ~ rad, data = Boston)
summary(model_rad)
```

```
##
## Call:
## lm(formula = medv ~ rad, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.770  -5.199  -1.967   3.321  33.292
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 26.38213    0.56176  46.964   <2e-16 ***
## rad         -0.40310    0.04349  -9.269   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.509 on 504 degrees of freedom
## Multiple R-squared:  0.1456, Adjusted R-squared:  0.1439
## F-statistic: 85.91 on 1 and 504 DF,  p-value: < 2.2e-16
```

**Analysis of Variance (ANOVA)**

```
anova(model_rad)$F
```

```
## [1] 85.91428       NA
```

```
anova(model_rad)
```

```
## Analysis of Variance Table
##
## Response: medv
##            Df Sum Sq Mean Sq F value    Pr(>F)
## rad         1   6221  6221.1  85.914 < 2.2e-16 ***
## Residuals 504  36495    72.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Inference : What is the F-test testing? Is the result consistent with the boxplots?

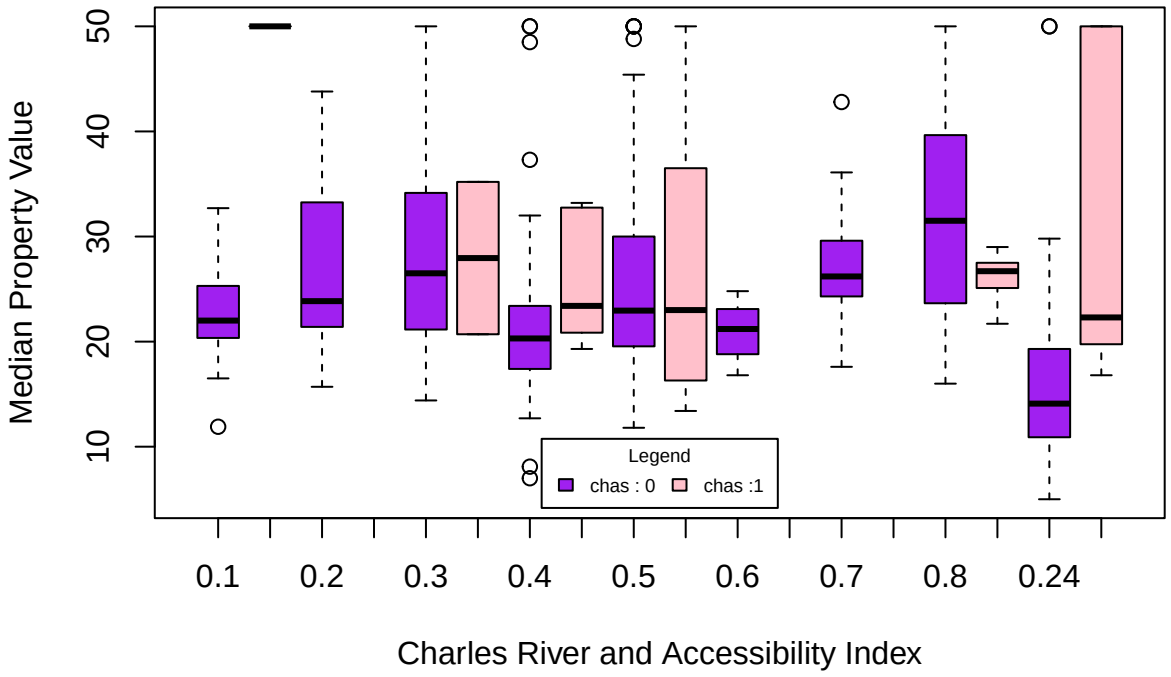
- F-test testing procedure that compares the variability of two or more sets of data
- For F test value =1 the variances are equal which is Null hypothesis and if it is not equal to 1 then there is variability
- We can also see that the F value here is 85.91 which denotes significant variability
- Null hypothesis for this case is : that the coefficients for the rad variable are equal to zero. This means that the rad variable has no effect on the median property value (**medv**). The null hypothesis assumes that the rad variable is not a significant predictor of **medv**. The alternative hypothesis is that the coefficients for the rad variable are not equal to zero, meaning that the **rad** variable does have an effect on **medv** and is a significant predictor.
- Since the p - value if smaller than the significance level (0.05), the null hypothesis fails and we conclude that the **rad** variable is significantly related to **medv** meaning that the **rad** variable does have an effect on **medv** and is a significant predictor.
- This result is also consistent with what we can see from the boxplots that less accessibility to highways has an effect on median property values

### 1(c) Boxplot

Produce a nice boxplot display of medv where the groups are defined by chas and rad jointly. Comment on what you observe. Then look at an interaction plot. Then fit a model explaining medv as a function of chas and rad with interactions. Output an ANOVA table. What are the different F-tests testing? Compare with the previous F-test as appropriate. Are the results of these tests consistent with the plots you just looked at?

```
boxplot(medv ~ chas * rad, data = Boston,
        main = "Median Property Value by Charles River and Accessibility Index",
        xlab = "Charles River and Accessibility Index",
        ylab = "Median Property Value",col=c("purple", "pink"))

legend("bottom", inset = 0.02, title = "Legend", c("chas : 0", "chas :1"),
       fill = c("purple", "pink"), horiz = TRUE, cex = 0.6)
```



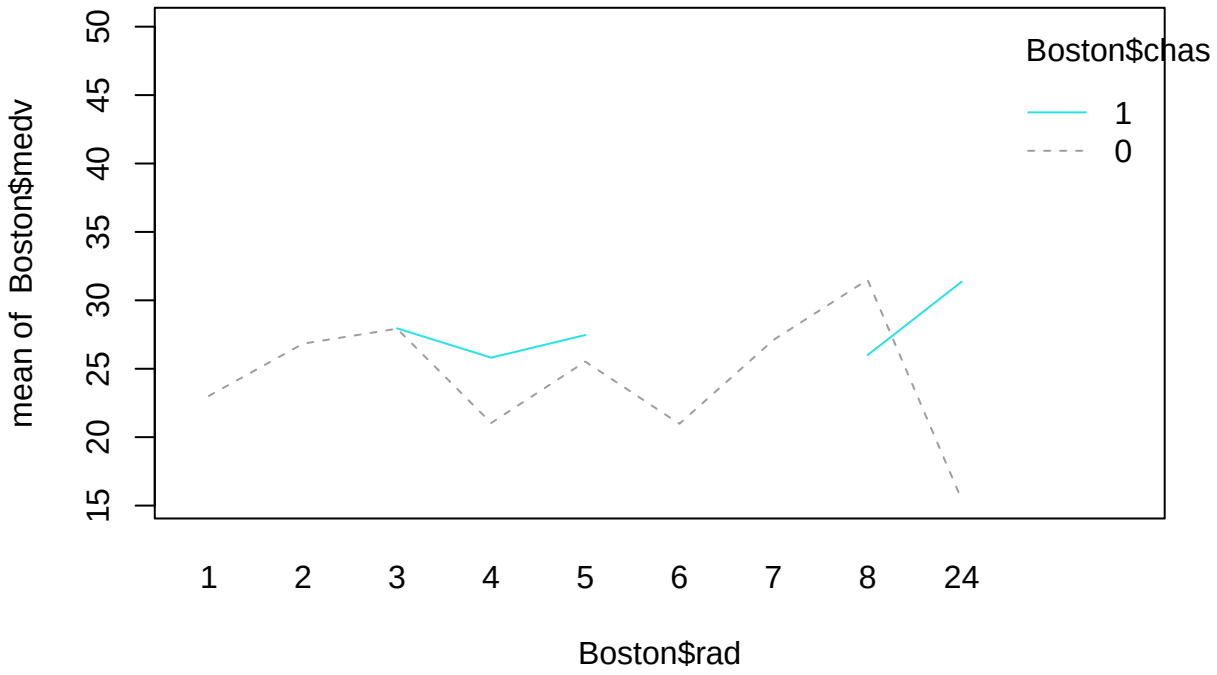

Inference:

- We can see that the grouping is done in 2 forms for chas where 1 and 0 indicate pink and purple boxes along with 9 values of rad for each, hence it creates 18 plots
- Another observation is that for some chas groups values(1) the rad values are missing.

**Interaction plot**

```
interaction.plot(Boston$rad, Boston$chas, Boston$medv, col = Boston$medv)
```

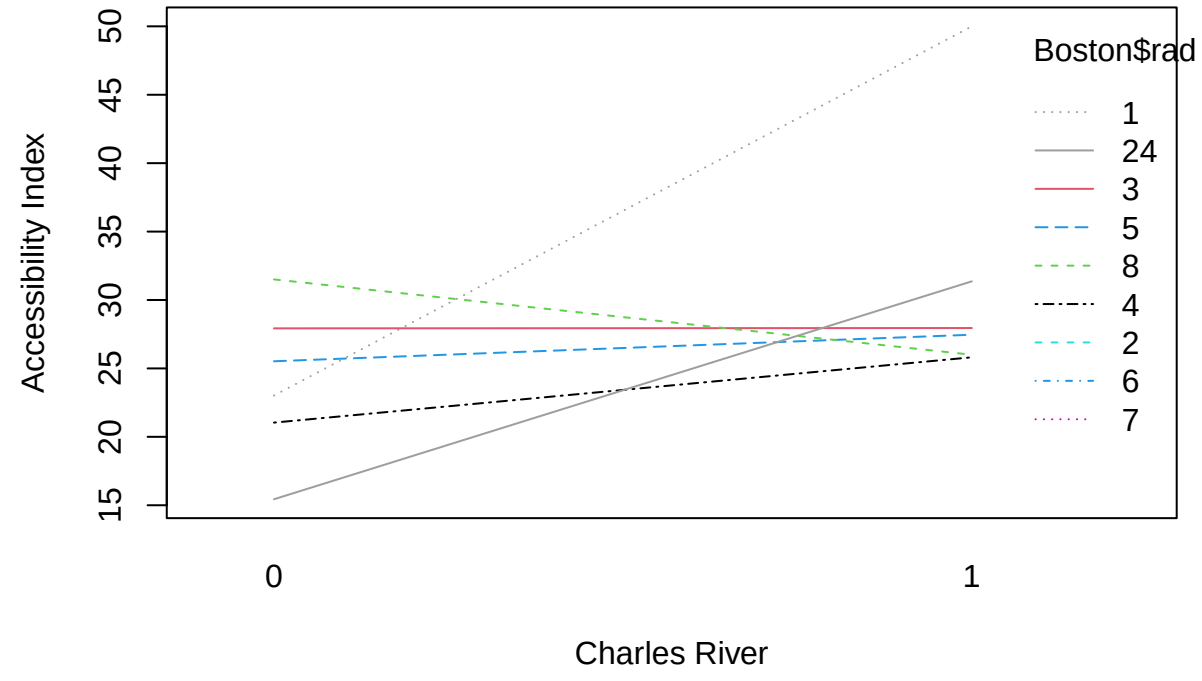
**Grouped by Chas**




Inference: We can see the variable trend for different values of rad for each of 0 and 1 groups chas.

```
interaction.plot(Boston$chas, Boston$rad, Boston$medv,
                 xlab = "Charles River",
                 ylab = "Accessibility Index", col = Boston$medv)
```

**Grouped by Rad**



Inference: We can see the lines for rad of different groups of chas for each of 0 and 1 groups

### Fitting model and ANOVA

```
model_combined= lm(medv ~ chas * rad, data = Boston)
summary(model_combined)
```

```
##
## Call:
## lm(formula = medv ~ chas * rad, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.527  -5.127  -1.796   3.548  34.216
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  26.2762     0.5662  46.409  < 2e-16 ***
## chas          0.7775     2.2042   0.353  0.72445
## rad          -0.4372     0.0437 -10.005  < 2e-16 ***
## chas:rad      0.5860     0.1777   3.297  0.00105 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.287 on 502 degrees of freedom
## Multiple R-squared:  0.1929, Adjusted R-squared:  0.188
## F-statistic: 39.98 on 3 and 502 DF,  p-value: < 2.2e-16
```

```
anova(model_combined)
```

```
## Analysis of Variance Table
##
## Response: medv
##            Df Sum Sq Mean Sq F value    Pr(>F)
## chas        1   1312  1312.1  19.104 1.505e-05 ***
## rad         1   6179  6179.4  89.972 < 2.2e-16 ***
## chas:rad    1    746   746.5  10.869  0.001047 **
## Residuals 502  34478    68.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Inference:

F tests are used for different purposes that is for:

- testing equality of variance to test hypothesis of equality of two population variances
- testing equality of several means to test for equality of several means is carried out by the technique called ANOVA
- for testing significance of regression is used to test the significance of the regression model

Comparision to previous F test:

As seen before, the chas and rad F test values were 15.97 and 85.91 but here we can see an increase in F test values to be 19.10 and 89.97 for degree of freedom 1

We can also see the combined effect of chas and rad where the F test values is 10.86 for degree of freedom 1. We can also see a decrease in significance of p values.

The observed values don't seem very consistent with the plots

### 1(d) Checking for chas

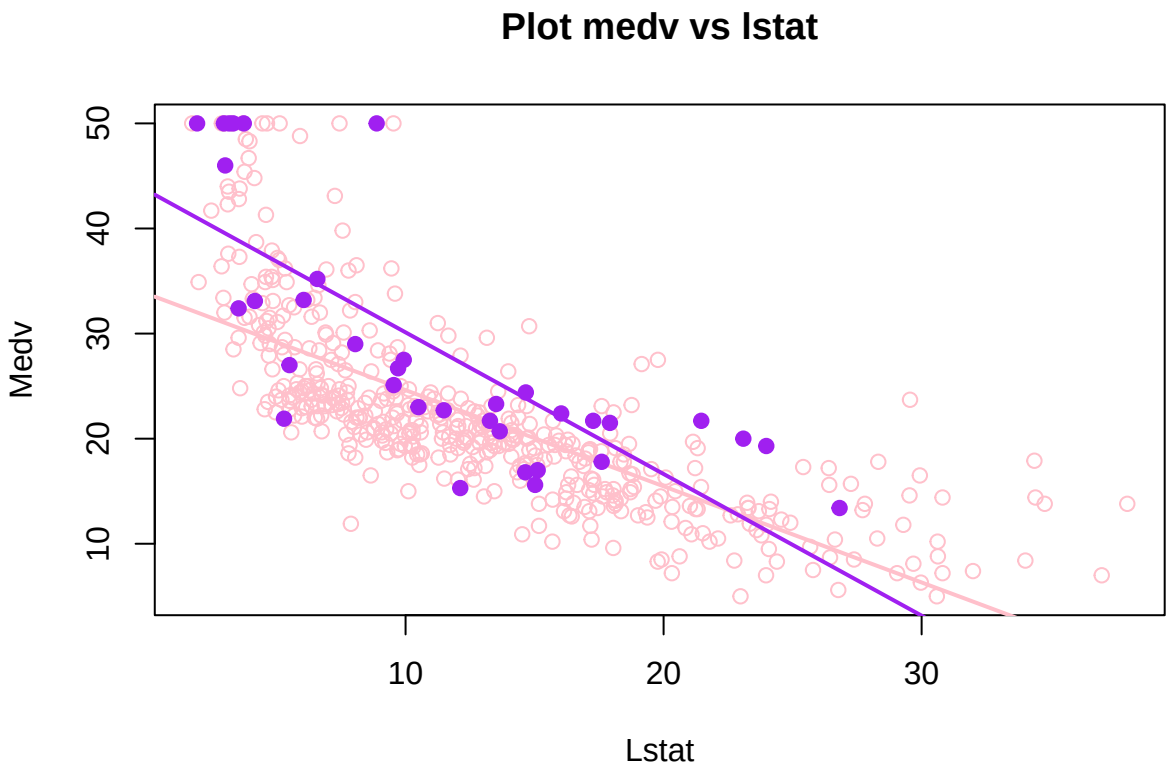
It makes sense that median property value decreases with the percentage of lower status population lstat, and this is indeed what is observed here. Does the rate of decrease depend on whether the area borders the Charles River? Produce a plot that helps answer that question.

```
plot(Boston$medv ~ Boston$lstat, col='steelblue',
     main = "Median Property Value(Medv) vs  Lstat",
     xlab = "Lstat", ylab = "Median Property Value")
m2 = lm(Boston$medv ~ Boston$lstat)
abline(m2, col="black")
```


We can observes that as the the value of lstat decreases, the values of medv increases. It is almost inversely proportional.

```
#Producing a plot to show our results
ind = (Boston$chas==0)
plot(Boston$medv[ind] ~ Boston$lstat[ind], data = Boston,
     main = "Plot medv vs lstat",col="pink",
     xlab="Lstat", ylab = "Medv")
```

```
fit_1 = lm(Boston$medv[ind] ~ Boston$lstat[ind], data = Boston)
abline(fit_1, col="pink", lwd=2)
ind = (Boston$chas==1)
points(Boston$medv[ind] ~ Boston$lstat[ind], data = Boston, col="purple", pch=19)
fit_2 = lm(Boston$medv[ind] ~ Boston$lstat[ind], data = Boston)
abline(fit_2, col="purple", lwd=2)
```




```
anova(fit_1)
```

```
## Analysis of Variance Table
##
## Response: Boston$medv[ind]
##                     Df Sum Sq Mean Sq F value    Pr(>F)
## Boston$lstat[ind]    1  20224   20224  577.23 < 2.2e-16 ***
## Residuals          469  16432      35
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit_2)
```

```
## Analysis of Variance Table
##
## Response: Boston$medv[ind]
##                     Df Sum Sq Mean Sq F value    Pr(>F)
## Boston$lstat[ind]    1 2761.8 2761.77  45.896 1.006e-07 ***
## Residuals           33 1985.8   60.17
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Inference:

- We can see from the interaction plot that the houses near the Charles river (chas =1 ) that is the purple dots has a higher median property value as compared to the ones that are far away from the Charles river(chas =0) that is the pink dots.
- To test the hypothesis that the rate of decrease of median property value depends on whether the area borders the Charles River, we can perform hypothesis testing. First, we need to formulate our null and alternative hypotheses.
- Our null hypothesis might be that the rate of decrease in median property value with increasing lstat is the same for neighborhoods that border the Charles River as it is for neighborhoods that do not. Our alternative hypothesis might be that the rate of decrease is different for these two groups.
- As we can see that both are highly significant in the hypothesis testing but the variability in both the F values is significantly different. Thus this disregards the null hypothesis and satisfies our alternate hypothesis.

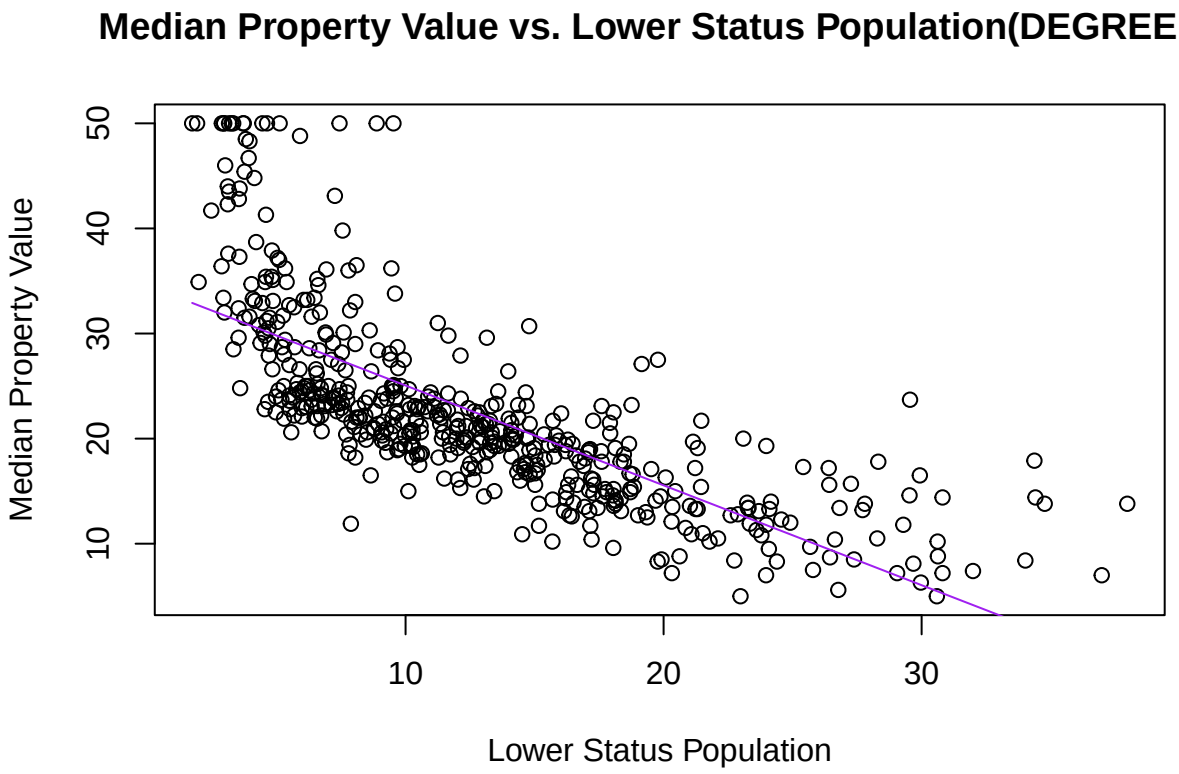
# Question 2

Consider the same dataset and turn to the problem of fitting a polynomial model explaining medv as a function of lstat.

### Fit a polynomial model of degree 1

First we will fit a model of degree of freedom = 1

```
#Fit a polynomial model of degree 1
fit <- lm(medv ~ poly(lstat, 1), data = Boston)
plot(Boston$lstat, Boston$medv,
     main = "Median Property Value vs. Lower Status Population(DEGREE 1)",
     xlab = "Lower Status Population",
     ylab = "Median Property Value")
lines(sort(Boston$lstat), predict(fit,
      newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple")
```

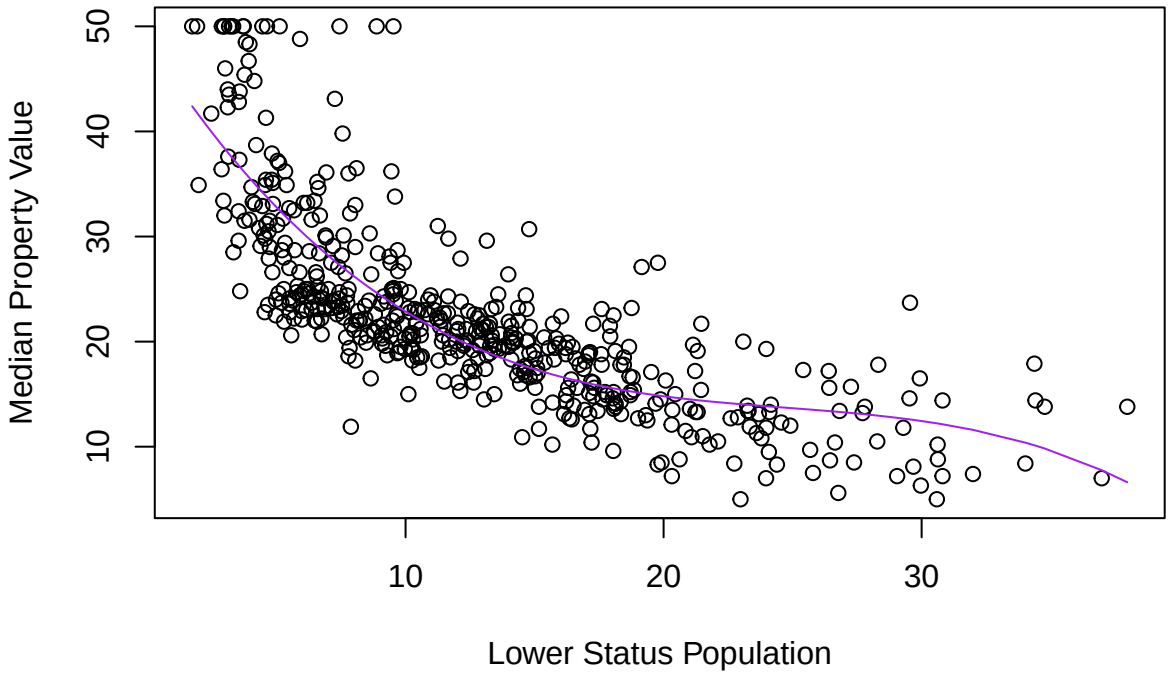



```
#summary(fit)
```

**2a** Fit a polynomial model of degree 3 by least squares.

**Fit a polynomial model of degree 3 by least squares.**

```
#Fit a polynomial model of degree 3 by least squares.
fit <- lm(medv ~ poly(lstat, 3), data = Boston)
plot(Boston$lstat, Boston$medv,
     main = "Median Property Value vs. Lower Status Population (DEGREE 3)",
     xlab = "Lower Status Population",
     ylab = "Median Property Value")
lines(sort(Boston$lstat), predict(fit, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple")
```




```
summary(fit)
```

```
##
## Call:
## lm(formula = medv ~ poly(lstat, 3), data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.5441  -3.7122  -0.5145   2.4846  26.4153
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       22.5328     0.2399  93.937  < 2e-16 ***
## poly(lstat, 3)1 -152.4595     5.3958 -28.255  < 2e-16 ***
## poly(lstat, 3)2   64.2272     5.3958  11.903  < 2e-16 ***
## poly(lstat, 3)3  -27.0511     5.3958  -5.013 7.43e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.396 on 502 degrees of freedom
## Multiple R-squared:  0.6578, Adjusted R-squared:  0.6558
## F-statistic: 321.7 on 3 and 502 DF,  p-value: < 2.2e-16
```

Similar plot with different library:

```
#Fit a polynomial model of degree 3 by regression qualtile library - rq
fit <- rq(medv ~ poly(lstat, 3), data = Boston, tau = 0.5)
plot(Boston$lstat, Boston$medv,
```

```
     main = "Median Property Value vs. Lower Status Population (DEGREE 3)",
     xlab = "Lower Status Population",
     ylab = "Median Property Value")
lines(sort(Boston$lstat),
      predict(fit, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple")
```


```
#summary(fit)
```

Inference:

As we can see from the above plots of degree of polynomial = 1 and degree of polynomial = 3, we can see a better fit in degree of polynomial = 3 rather than degree of polynomial = 1 .

This is because degree of polynomial = 3 model adjusts more to the current data

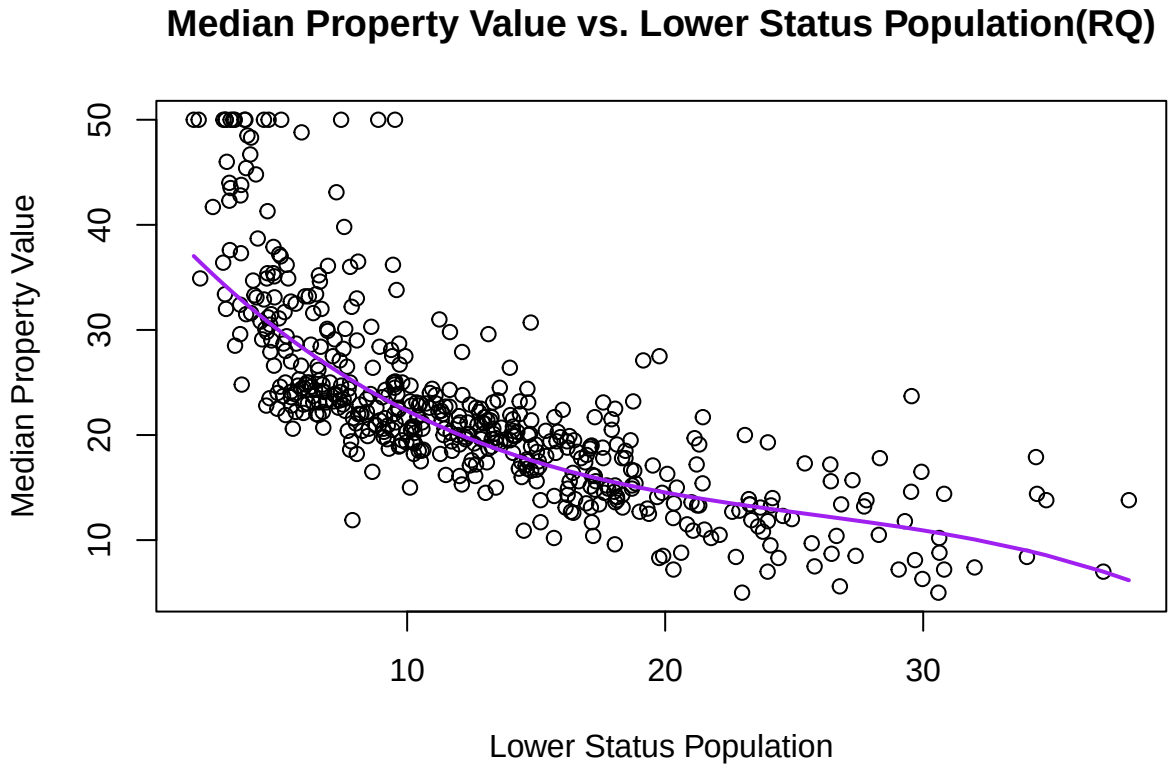
For degree of polynomial = 1 we can see high bias whereas for degree of polynomial = 3 there is much more optimal fit.

**2b Repeat with each robust method covered in the lecture notes/slides.**

**Robust methods**

```
fit_rq <- rq(medv ~ poly(lstat, 3), data = Boston, tau = 0.5)
plot(Boston$lstat, Boston$medv,
     main = "Median Property Value vs. Lower Status Population(RQ)",
     xlab = "Lower Status Population",
```

```
     ylab = "Median Property Value")
lines(sort(Boston$lstat),
      predict(fit_rq, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple", lwd = 2)
```



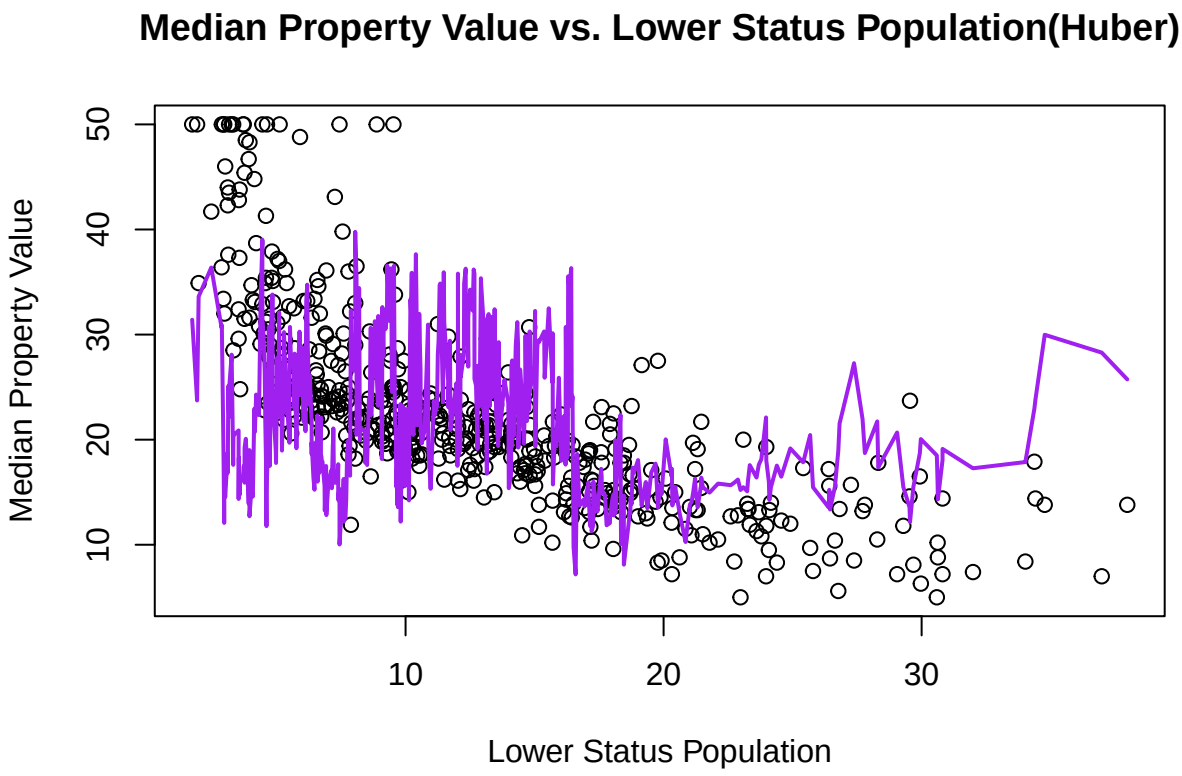

**RQ**

```
m.huber = rlm(Boston$medv ~ poly(Boston$lstat, 3), psi = psi.huber)
summary(m.huber)
```

**Huber's M-estimation:**

```
##
## Call: rlm(formula = Boston$medv ~ poly(Boston$lstat, 3), psi = psi.huber)
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.8373  -3.0157  -0.2219   2.7329  26.8409
##
## Coefficients:
##                          Value     Std. Error t value
## (Intercept)               21.9774    0.2037   107.8773
## poly(Boston$lstat, 3)1 -143.0378    4.5827   -31.2126
## poly(Boston$lstat, 3)2   55.9111    4.5827    12.2005
## poly(Boston$lstat, 3)3  -21.7848    4.5827    -4.7537
##
## Residual standard error: 4.323 on 502 degrees of freedom
```

```r
plot(Boston$lstat, Boston$medv,
     main = "Median Property Value vs. Lower Status Population(Huber)",
     xlab = "Lower Status Population",
     ylab = "Median Property Value")
lines(sort(Boston$lstat),
      predict(m.huber, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple", lwd = 2)
```



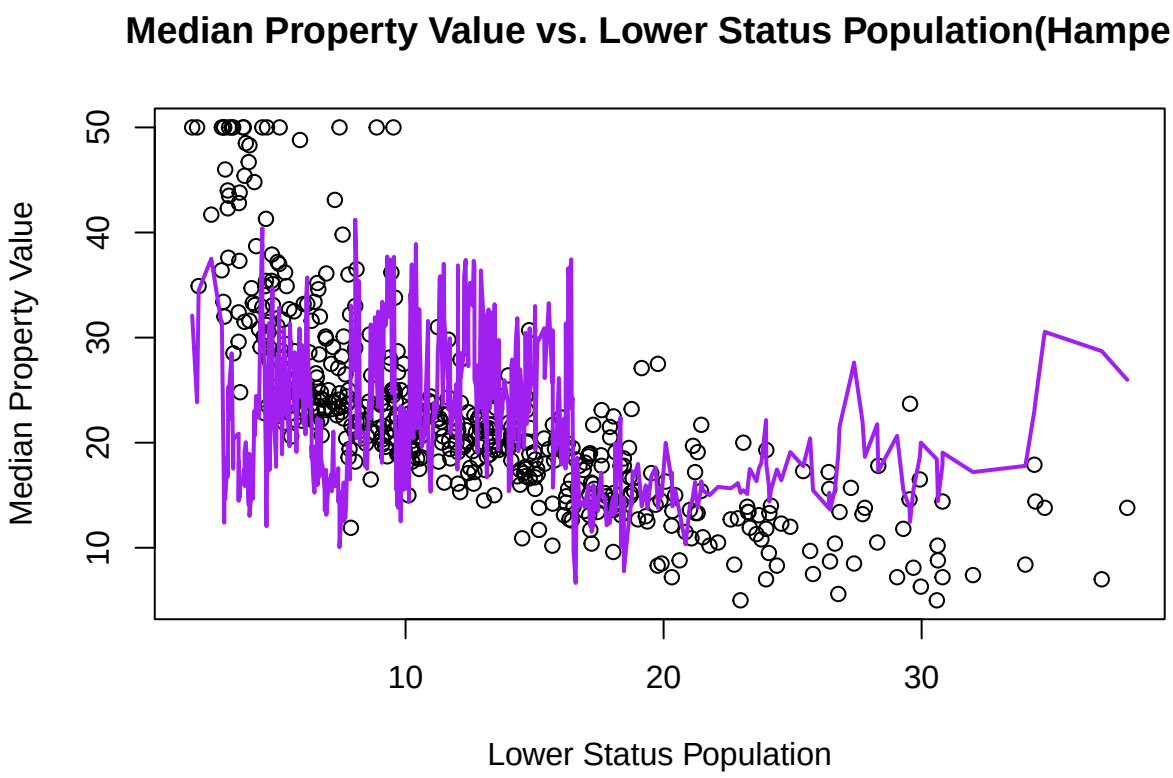

```r
m.hampel = rlm(Boston$medv ~ poly(Boston$lstat, 3), psi = psi.hampel)
summary(m.hampel)
```

**Hampel's M-estimation:**

```
##
## Call: rlm(formula = Boston$medv ~ poly(Boston$lstat, 3), psi = psi.hampel)
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.0819  -3.3423  -0.3086   2.6869  26.7536
##
## Coefficients:
##                          Value     Std. Error t value
## (Intercept)               22.2032    0.2161   102.7330
## poly(Boston$lstat, 3)1 -147.0125    4.8616   -30.2395
## poly(Boston$lstat, 3)2   60.7791    4.8616    12.5019
## poly(Boston$lstat, 3)3  -25.7454    4.8616    -5.2957
```

```
##
## Residual standard error: 4.547 on 502 degrees of freedom
```

```
plot(Boston$lstat, Boston$medv,
     main = "Median Property Value vs. Lower Status Population(Hampel)",
     xlab = "Lower Status Population",
     ylab = "Median Property Value")
lines(sort(Boston$lstat),
      predict(m.hampel, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple", lwd = 2)
```



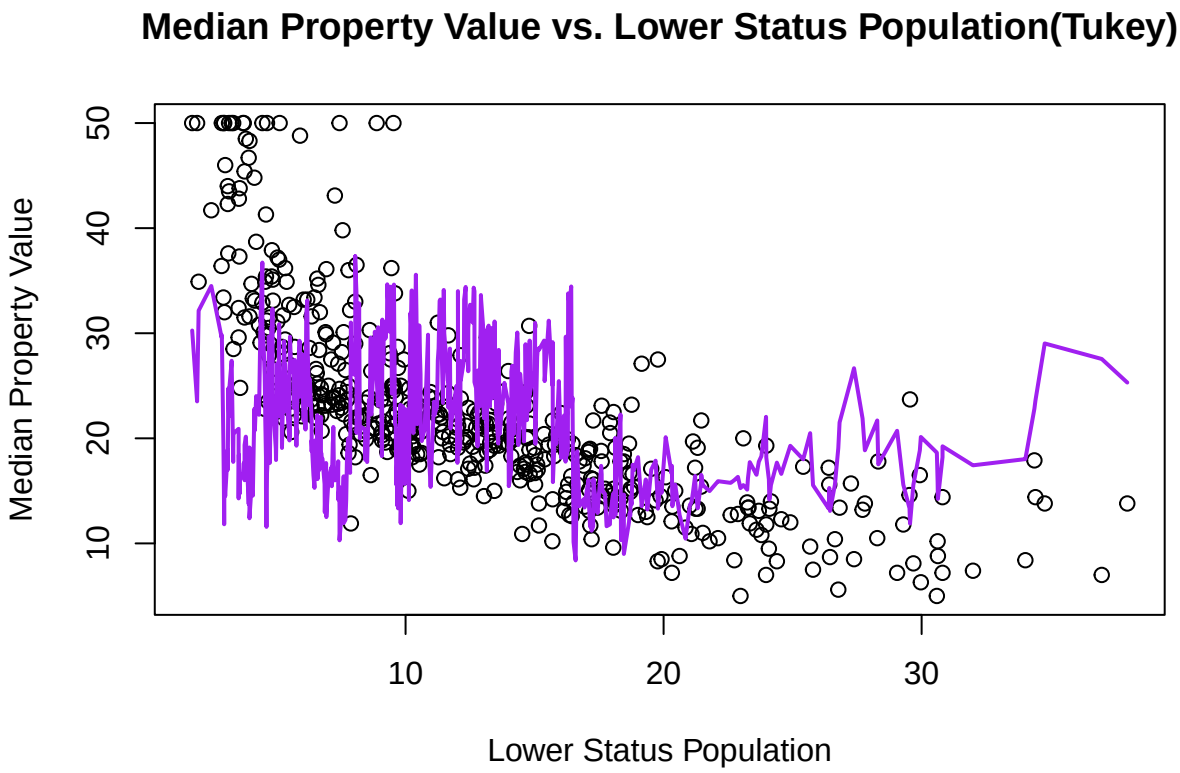

```
m.tukey = rlm(Boston$medv ~ poly(Boston$lstat, 3), psi = psi.bisquare)
summary(m.tukey)
```

**Tukey's M-estimation:**

```
##
## Call: rlm(formula = Boston$medv ~ poly(Boston$lstat, 3), psi = psi.bisquare)
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4084  -2.7037  -0.1377   2.9575  27.0035
##
## Coefficients:
##                         Value     Std. Error t value
## (Intercept)              21.6329    0.1962   110.2426
```

```
## poly(Boston$lstat, 3)1 -134.8470    4.4141   -30.5493
## poly(Boston$lstat, 3)2   48.3681    4.4141    10.9577
## poly(Boston$lstat, 3)3  -15.4625    4.4141    -3.5030
##
## Residual standard error: 4.172 on 502 degrees of freedom
```

```
plot(Boston$lstat, Boston$medv,
     main ="Median Property Value vs. Lower Status Population(Tukey)",
     xlab = "Lower Status Population",
     ylab = "Median Property Value")
lines(sort(Boston$lstat),
      predict(m.tukey, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple", lwd = 2)
```



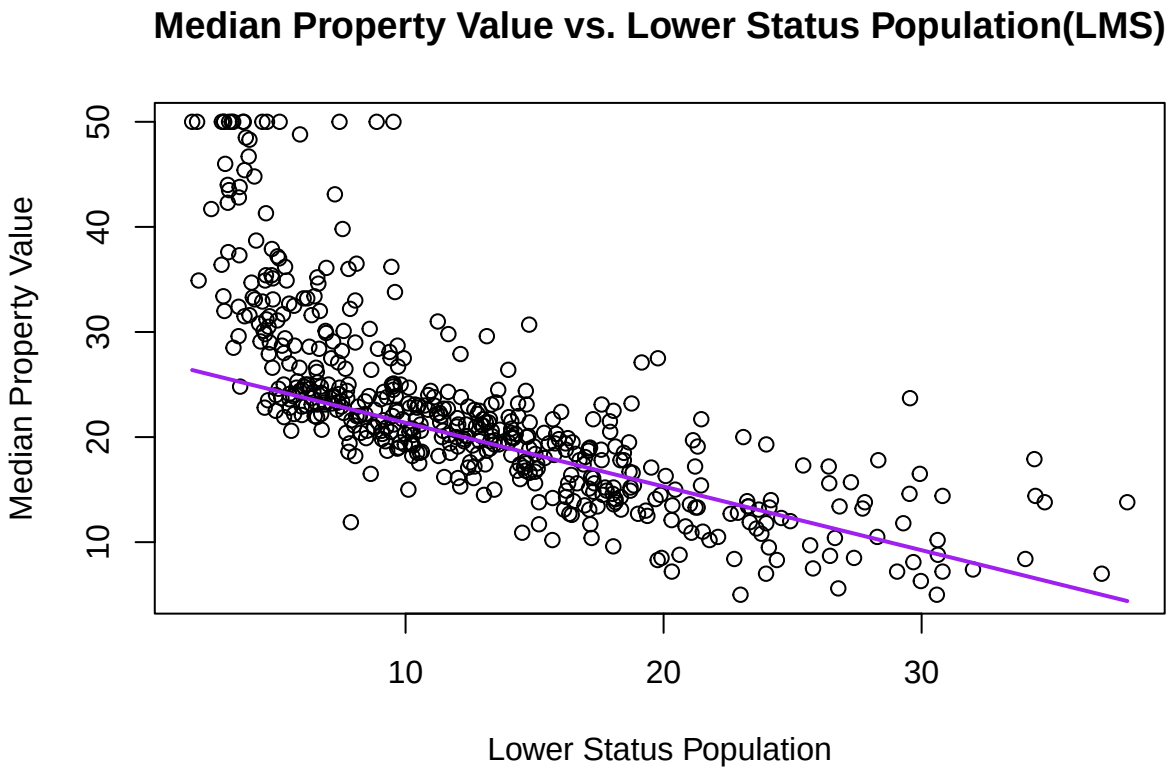

```
# high breakdown point methods
fit.lms = lmsreg(medv ~ lstat, data = Boston)
summary(fit.lms)
```

**High breakdown point methods - Least Median of Squares (LMS)**

```
##              Length Class      Mode
## crit           1    -none-     numeric
## sing           1    -none-     character
## coefficients   2    -none-     numeric
## bestone        2    -none-     numeric
```

```
## fitted.values 506    -none-     numeric
## residuals     506    -none-     numeric
## scale           2    -none-     numeric
## terms           3    terms      call
## call            4    -none-     call
## xlevels         0    -none-     list
## model           2    data.frame list
```

```
plot(Boston$lstat, Boston$medv,
     main =  "Median Property Value vs. Lower Status Population(LMS)",
     xlab = "Lower Status Population", ylab = "Median Property Value")
lines(sort(Boston$lstat),
      predict(fit.lms,
      newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple", lwd = 2)
```



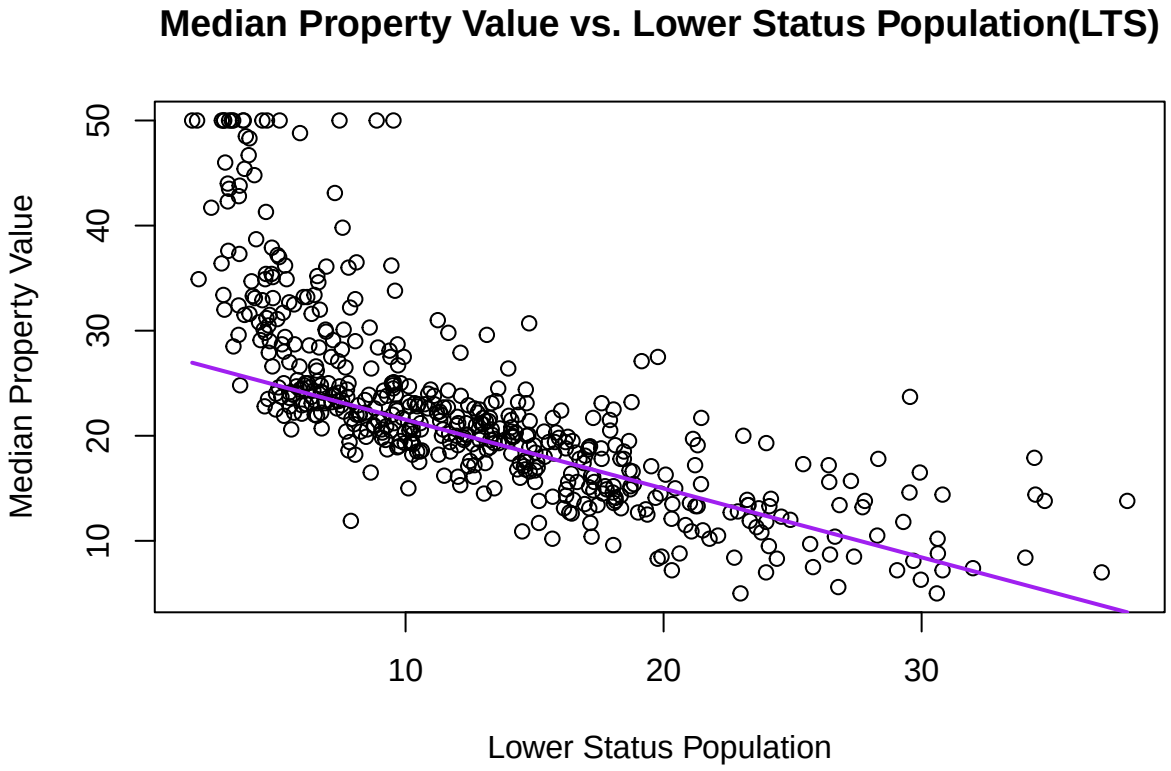

```
fit.lts = ltsreg(medv ~ lstat, data = Boston)
summary(fit.lts)
```

**High breakdown point methods - Least Trimmed Squares (LTS)**

```
##               Length Class      Mode
## crit            1    -none-     numeric
## sing            1    -none-     character
## coefficients    2    -none-     numeric
```

```
## bestone         2    -none-     numeric
## fitted.values 506    -none-     numeric
## residuals     506    -none-     numeric
## scale           2    -none-     numeric
## terms           3    terms      call
## call            4    -none-     call
## xlevels         0    -none-     list
## model           2    data.frame list
```

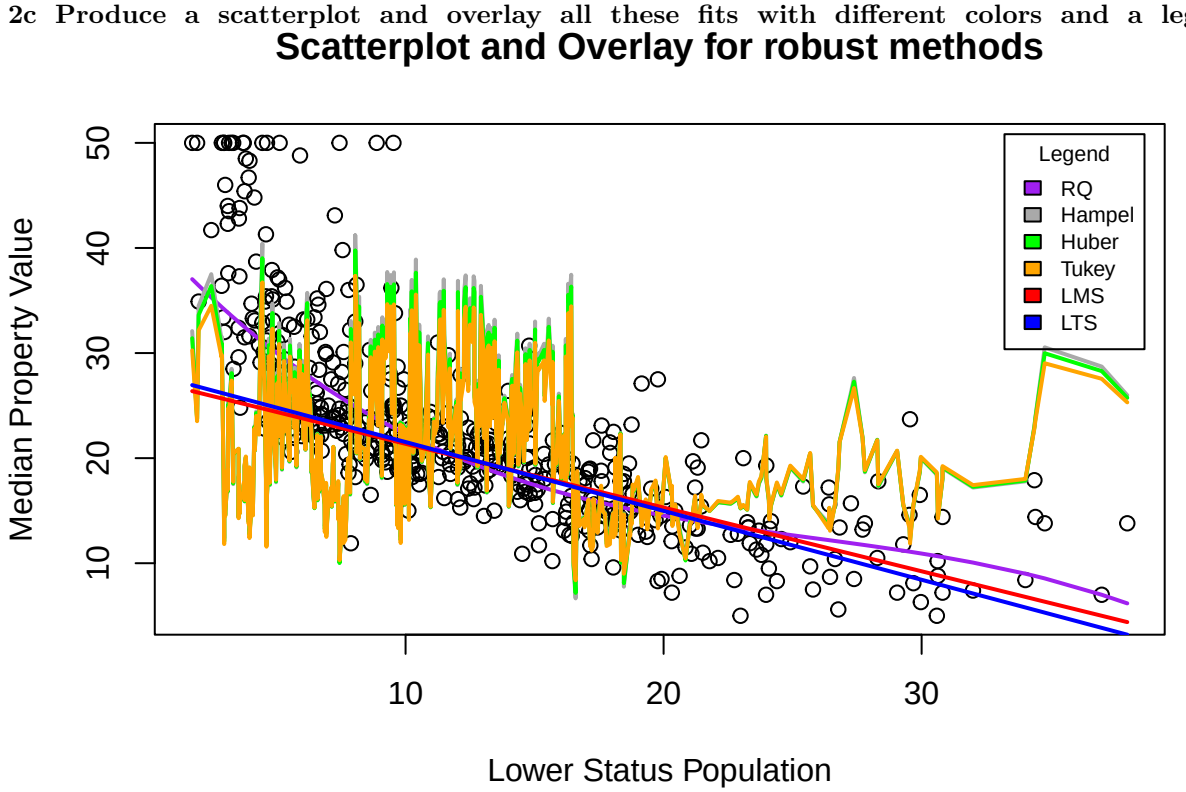
```
plot(Boston$lstat, Boston$medv,
     main =  "Median Property Value vs. Lower Status Population(LTS)",
     xlab = "Lower Status Population", ylab = "Median Property Value")
lines(sort(Boston$lstat),
      predict(fit.lts,
      newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple", lwd = 2)
```




```
plot(Boston$lstat, Boston$medv,
     main = "Scatterplot and Overlay for robust methods",
     xlab = "Lower Status Population", ylab = "Median Property Value")
lines(sort(Boston$lstat), predict(fit_rq, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "purple", lwd = 2)
lines(sort(Boston$lstat), predict(m.hampel, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "darkgrey", lwd = 2)
lines(sort(Boston$lstat), predict(m.huber, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "green", lwd = 2)
```

```
lines(sort(Boston$lstat), predict(m.tukey, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "orange", lwd = 2)
lines(sort(Boston$lstat), predict(fit.lms, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "red", lwd = 2)
lines(sort(Boston$lstat), predict(fit.lts, newdata = data.frame(lstat = sort(Boston$lstat))),
      col = "blue", lwd = 2)
legend("topright", inset = 0.02,
       title = "Legend", c("RQ", "Hampel","Huber","Tukey","LMS","LTS"),
       fill = c("purple","darkgrey","green","orange","red","blue"),
       horiz = FALSE, cex = 0.7)
```

**2c Produce a scatterplot and overlay all these fits with different colors and a legend.**




Team Contributions :

Both the team members Sourabh Prakash and Priyanshi Shah have contributed equally to the homework by discussing the key points and logic together and doing pair programming. For the implementation part question 1 a,b,c was contributed by Priyanshi Shah and question 1d and 2a,b,c by Sourabh Prakash. The conclusions were written together for all the parts of question 1 and 2