

STAT3550_Lab1

Luke Rodriguez

2026-01-21

```
library(MASS)
library(ISLR2)

##
## Attaching package: 'ISLR2'

## The following object is masked from 'package:MASS':
## Boston
```

Fit a simple linear regression model using the Boston dataset, with median house value as the response variable low economic status as the explanatory variable.

```
head(Boston)

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

attach(Boston)
lm.fit <- lm(medv ~ lstat) # y ~ x
```

view the output of the model using the summary function

```
lm.fit

##
## Call:
## lm(formula = medv ~ lstat)
##
## Coefficients:
## (Intercept)      lstat
##           34.55       -0.95
```

```
summary(lm.fit)

##
## Call:
## lm(formula = medv ~ lstat)
##
## Residuals:
##    Min     1Q Median     3Q    Max 
## -15.168 -3.990 -1.318  2.034 24.500 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 34.55384   0.56263   61.41 <2e-16 ***
## lstat       -0.95005   0.03873  -24.53 <2e-16 ***
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.216 on 504 degrees of freedom
## Multiple R-squared:  0.5441, Adjusted R-squared:  0.5432 
## F-statistic: 601.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

Coefficient

```
coef(lm.fit)
```

```
## (Intercept)      lstat  
## 34.5538409 -0.9500494
```

Confidence intervals Use the confint function to create a 95% CI

```
confint(lm.fit)
```

```
##              2.5 %    97.5 %    
## (Intercept) 33.448457 35.6592247
## lstat       -1.026148 -0.8739505
```

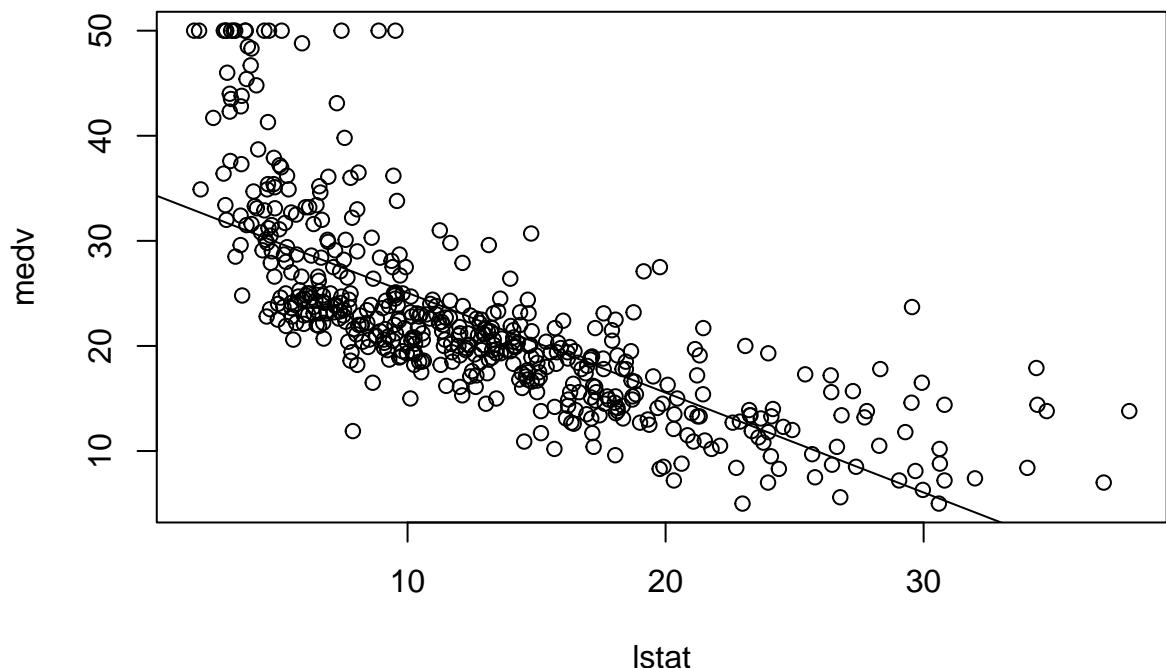
Do predictions and create prediction intervals based on 3 new values of lstat variable (5, 10, and 15).

```
predict(lm.fit, data.frame(lstat = c(5, 10, 15))), interval = "prediction")
```

```
##      fit      lwr      upr  
## 1 29.80359 17.565675 42.04151
## 2 25.05335 12.827626 37.27907
## 3 20.30310  8.077742 32.52846
```

visualizing the model

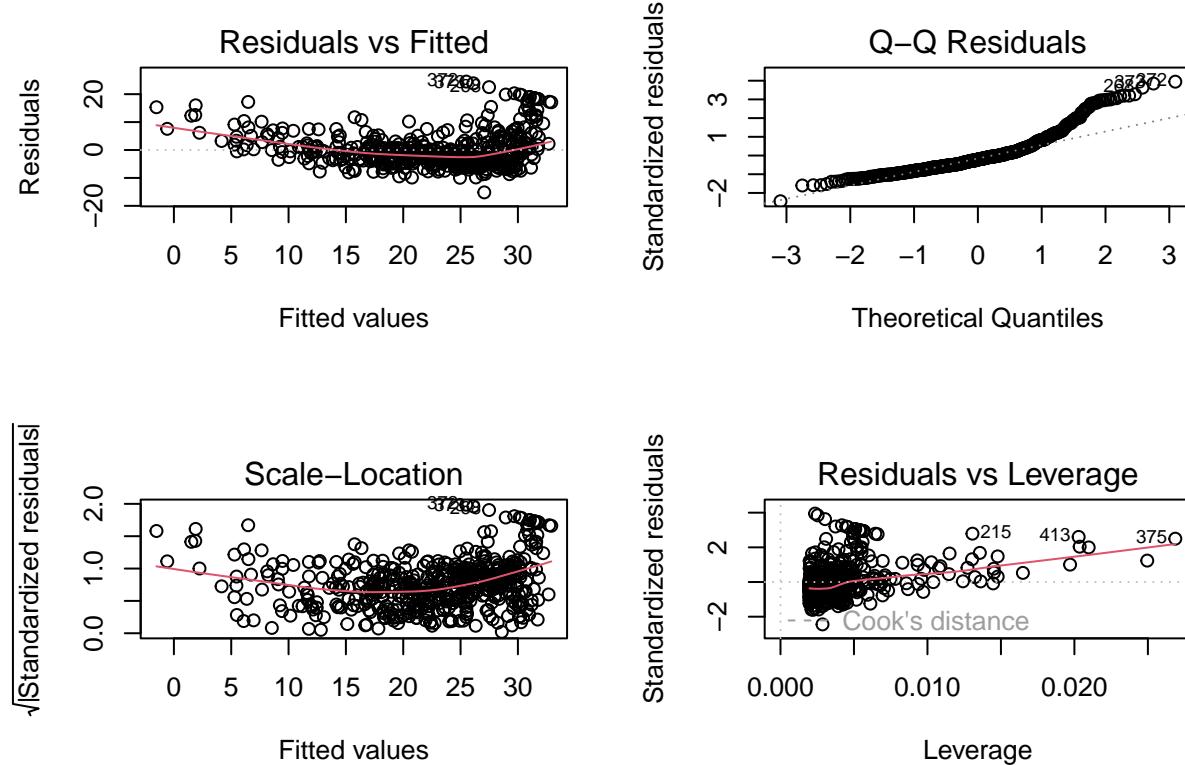
```
plot(lstat, medv)
abline(lm.fit)
```



Notice that the model seems relatively linear, but may show some nonlinearity, particularly seen in lower values values of lstat.

checking model assumptions

```
par(mfrow = c(2,2))
plot(lm.fit)
```



Notice that the residuals are not exhibiting constant variance, which might indicate that the model assumptions are not satisfied. Perhaps a different model will work better for our data.

Multiple linear regression

Fit a model with two predictors

```
lm.fit <- lm(medv ~ lstat + age, data = Boston)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = medv ~ lstat + age, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -15.981 -3.978 -1.283  1.968 23.158 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 33.22276   0.73085 45.458 < 2e-16 ***
## lstat       -1.03207   0.04819 -21.416 < 2e-16 ***
## age          0.03454   0.01223   2.826  0.00491 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.173 on 503 degrees of freedom
## Multiple R-squared:  0.5513, Adjusted R-squared:  0.5495 
## F-statistic: 309 on 2 and 503 DF, p-value: < 2.2e-16
```

Fit a model with all predictors

```
lm.fit <- lm(medv ~ ., data = Boston) # take all the other variables as the explanatory variables (if less than 1000 characters)
```

```
##
## Call:
## lm(formula = medv ~ ., data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -15.1304 -2.7673 -0.5814  1.9414 26.2526 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 41.617270  4.936039  8.431 3.79e-16 ***
## crim        -0.121389  0.033000 -3.678 0.000261 *** 
## zn          0.046963  0.013879  3.384 0.000772 *** 
## indus       0.013468  0.062145  0.217 0.828520  
## chas        2.839993  0.870007  3.264 0.001173 ** 
## nox         -18.758022 3.851355 -4.870 1.50e-06 *** 
## rm          3.658119  0.420246  8.705 < 2e-16 *** 
## age         0.003611  0.013329  0.271 0.786595  
## dis        -1.490754  0.201623 -7.394 6.17e-13 ***
```

```

## rad          0.289405  0.066908  4.325 1.84e-05 ***
## tax         -0.012682  0.003801 -3.337 0.000912 ***
## ptratio     -0.937533  0.132206 -7.091 4.63e-12 ***
## lstat      -0.552019  0.050659 -10.897 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.798 on 493 degrees of freedom
## Multiple R-squared:  0.7343, Adjusted R-squared:  0.7278
## F-statistic: 113.5 on 12 and 493 DF,  p-value: < 2.2e-16

```

nonlinear transformations

using the boston housing dataset, investigate whether the relationship between median house value(medv) and percentage

polynomial moel

fit a quadratic regression model

```
#linear model
lm_linear <- lm(medv ~ lstat, data = Boston)
summary(lm_linear)
```

```
##
## Call:
## lm(formula = medv ~ lstat, data = Boston)
##
## Residuals:
##    Min     1Q   Median     3Q    Max
## -15.168 -3.990 -1.318  2.034 24.500
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 34.55384   0.56263   61.41  <2e-16 ***
## lstat       -0.95005   0.03873  -24.53  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.216 on 504 degrees of freedom
## Multiple R-squared:  0.5441, Adjusted R-squared:  0.5432
## F-statistic: 601.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

```
#quadratic model
lm_quad <- lm(medv ~ lstat + I(lstat^2), data = Boston)
summary(lm_quad)
```

```
##
## Call:
## lm(formula = medv ~ lstat + I(lstat^2), data = Boston)
##
## Residuals:
##    Min     1Q   Median     3Q    Max
## -15.2834 -3.8313 -0.5295  2.3095 25.4148
```

```

## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 42.862007   0.872084  49.15 <2e-16 ***
## lstat       -2.332821   0.123803 -18.84 <2e-16 ***
## I(lstat^2)   0.043547   0.003745  11.63 <2e-16 ***
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 5.524 on 503 degrees of freedom
## Multiple R-squared:  0.6407, Adjusted R-squared:  0.6393 
## F-statistic: 448.5 on 2 and 503 DF,  p-value: < 2.2e-16

```

Run anova test to see if the model improves from the original linear model when adding the quadratic term.

```
anova(lm_linear, lm_quad)
```

```

## Analysis of Variance Table
## 
## Model 1: medv ~ lstat
## Model 2: medv ~ lstat + I(lstat^2)
##   Res.Df   RSS Df Sum of Sq    F    Pr(>F)    
## 1     504 19472
## 2     503 15347  1     4125.1 135.2 < 2.2e-16 ***
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
par(mfrow = c(2,2))
plot(lm_quad)
```

