Untitled

Fabian Abrego

4/22/2020

##Part 1

For this part use the prostate dataset from the faraway package. Use ?prosate to learn about the dataset. The goal of this exercise is to find a model that is useful for explaining the response lpsa.

Fit a total of five models.

One must use all possible predictors. One must use only lcavol as a predictor. The remaining three you must choose. The models you choose must be picked in a way such that for any two of the five models, one is nested inside the other. Argue that one of the five models is the best among them for explaining the response. Use appropriate methods covered and justify your answer.

```r
prostate = faraway::prostate
model_all = lm(lpsa ~ ., data = prostate)
model_lcavol = lm(lpsa ~ lcavol, data = prostate)
model_5 = lm(lpsa ~ lcavol + lweight + age + lbph + lcp, data = prostate)
model_4 = lm(lpsa ~ lcavol + lweight + age + lbph, data = prostate)
model_2 = lm(lpsa ~ lcavol + lweight, data = prostate)
summary(model_all)
```

```
##
## Call:
## lm(formula = lpsa ~ ., data = prostate)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7331 -0.3713 -0.0170  0.4141  1.6381
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.669337   1.296387   0.516  0.60693
## lcavol       0.587022   0.087920   6.677 2.11e-09 ***
## lweight      0.454467   0.170012   2.673  0.00896 **
## age         -0.019637   0.011173  -1.758  0.08229 .
## lbph         0.107054   0.058449   1.832  0.07040 .
## svi          0.766157   0.244309   3.136  0.00233 **
## lcp         -0.105474   0.091013  -1.159  0.24964
## gleason      0.045142   0.157465   0.287  0.77503
## pgg45        0.004525   0.004421   1.024  0.30886
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7084 on 88 degrees of freedom
## Multiple R-squared:  0.6548, Adjusted R-squared:  0.6234
## F-statistic: 20.86 on 8 and 88 DF,  p-value: < 2.2e-16
```

```r
summary(model_lcavol)
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.67625 -0.41648  0.09859  0.50709  1.89673
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.50730    0.12194   12.36   <2e-16 ***
## lcavol       0.71932    0.06819   10.55   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7875 on 95 degrees of freedom
## Multiple R-squared:  0.5394, Adjusted R-squared:  0.5346
## F-statistic: 111.3 on 1 and 95 DF,  p-value: < 2.2e-16
```

```r
summary(model_5)
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + lcp, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.51922 -0.39020  0.00317  0.47268  1.75943
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.80750    0.87692   0.921  0.35957
## lcavol       0.62519    0.09017   6.933 5.78e-10 ***
## lweight      0.46363    0.17576   2.638  0.00981 **
## age         -0.01345    0.01134  -1.186  0.23870
## lbph         0.08493    0.06064   1.400  0.16477
## lcp          0.09044    0.07392   1.223  0.22432
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.746 on 91 degrees of freedom
## Multiple R-squared:  0.6041, Adjusted R-squared:  0.5823
## F-statistic: 27.77 on 5 and 91 DF,  p-value: < 2.2e-16
```

```r
summary(model_4)
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph, data = prostate)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4885 -0.4241 -0.0001  0.4031  1.8073
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.73074    0.87703   0.833   0.4069
## lcavol       0.69854    0.06754  10.343   <2e-16 ***
## lweight      0.45770    0.17617   2.598   0.0109 *
## age         -0.01371    0.01137  -1.206   0.2309
## lbph         0.08404    0.06080   1.382   0.1702
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.748 on 92 degrees of freedom
## Multiple R-squared:  0.5976, Adjusted R-squared:  0.5801
## F-statistic: 34.15 on 4 and 92 DF,  p-value: < 2.2e-16
```

```r
summary(model_2)
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight, data = prostate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.61965 -0.50778 -0.02095  0.52291  1.89885
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.30262    0.56904  -0.532  0.59612
## lcavol       0.67753    0.06626  10.225  < 2e-16 ***
## lweight      0.51095    0.15726   3.249  0.00161 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7506 on 94 degrees of freedom
## Multiple R-squared:  0.5859, Adjusted R-squared:  0.5771
## F-statistic: 66.51 on 2 and 94 DF,  p-value: < 2.2e-16
```

```r
#RMSE
RMSE = function(test)
{sqrt((summary(test)$sigma^2)*test$df.residual/length(test$fitted.values))}


RMSE(model_all)
```

```
## [1] 0.674751
```

```r
RMSE(model_lcavol)
```

```
## [1] 0.7793386
```

```r
RMSE(model_5)
```

```
## [1] 0.7225684
```

```r
RMSE(model_4)
```

```
## [1] 0.7284869
```

```r
RMSE(model_2)
```

```
## [1] 0.7389478
```

```r
#Testing Residuals
#Normality Assumption
shapiro.test(resid(model_all))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(model_all)
## W = 0.99113, p-value = 0.7721
```

```r
shapiro.test(resid(model_lcavol))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(model_lcavol)
## W = 0.97985, p-value = 0.1419
```

```r
shapiro.test(resid(model_5))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(model_5)
## W = 0.98824, p-value = 0.5486
```

```r
shapiro.test(resid(model_4))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(model_4)
## W = 0.98684, p-value = 0.4491
```

```r
shapiro.test(resid(model_2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(model_2)
## W = 0.99043, p-value = 0.718
```

```r
#Constant Variance Assumption
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
bptest(model_all)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_all
## BP = 10.08, df = 8, p-value = 0.2594
```

```r
bptest(model_lcavol)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_lcavol
## BP = 0.12623, df = 1, p-value = 0.7224
```

```r
bptest(model_5)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_5
## BP = 4.2837, df = 5, p-value = 0.5093
```

```r
bptest(model_4)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_4
## BP = 2.0293, df = 4, p-value = 0.7304
```

```r
bptest(model_2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_2
## BP = 3.3046, df = 2, p-value = 0.1916
```

All models meet assumptions.

```r
#AIC similar to Mallows CP in comparing models - the smaller the better
AIC(model_all)
```

```
## [1] 218.9522
```

```r
AIC(model_lcavol)
```

```
## [1] 232.908
```

```r
AIC(model_5)
```

```
## [1] 226.2351
```

```r
AIC(model_4)
```

```
## [1] 225.8177
```

```r
AIC(model_2)
```

```
## [1] 224.5837
```

```r
#BIC the smaller the better
BIC(model_all)
```

```
## [1] 244.6993
```

```r
BIC(model_lcavol)
```

```
## [1] 240.6321
```

```r
BIC(model_5)
```

```
## [1] 244.2581
```

```r
BIC(model_4)
```

```
## [1] 241.2659
```

```r
BIC(model_2)
```

```
## [1] 234.8825
```

```r
#Anova Testing
#Model All - at 5 percent significance- reject the null for all- suggesting linear relationship between
anova(model_2,model_all)
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol + lweight
## Model 2: lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
##     pgg45
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     94 52.966
## 2     88 44.163  6    8.8032 2.9236 0.01199 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(model_4,model_all)
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol + lweight + age + lbph
## Model 2: lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
##     pgg45
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     92 51.477
## 2     88 44.163  4    7.3142 3.6436 0.00855 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(model_5,model_all)
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol + lweight + age + lbph + lcp
## Model 2: lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
##     pgg45
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     91 50.644
## 2     88 44.163  3    6.4812 4.3048 0.00699 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(model_lcavol,model_all)
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol
## Model 2: lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
##     pgg45
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     95 58.915
## 2     88 44.163  7    14.752 4.1992 0.0004916 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Model 5

anova(model_4,model_5) # no significant difference between models - the smaller the better
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol + lweight + age + lbph
## Model 2: lpsa ~ lcavol + lweight + age + lbph + lcp
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     92 51.477
## 2     91 50.644  1   0.83304 1.4968 0.2243
```

```r
anova(model_2,model_5) # no significant difference between models - the smaller the better
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol + lweight
## Model 2: lpsa ~ lcavol + lweight + age + lbph + lcp
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     94 52.966
## 2     91 50.644  3    2.3221 1.3908 0.2507
```

```r
anova(model_lcavol,model_5) # sigificant - model 5 preferred to lcavol
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol
## Model 2: lpsa ~ lcavol + lweight + age + lbph + lcp
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     95 58.915
## 2     91 50.644  4    8.2706 3.7152 0.007575 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Model 4

anova(model_lcavol,model_4) # significant - Model 4 preferred to Lcavol
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol
## Model 2: lpsa ~ lcavol + lweight + age + lbph
```

```
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     95 58.915
## 2     92 51.477  3    7.4375 4.4308 0.005902 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(model_2,model_4) # not significant - the smaller the better
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol + lweight
## Model 2: lpsa ~ lcavol + lweight + age + lbph
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     94 52.966
## 2     92 51.477  2     1.489 1.3306 0.2694
```

```r
#Model 2

anova(model_lcavol,model_2) #significant- Model 2 preferred against lcavol.
```

```
## Analysis of Variance Table
##
## Model 1: lpsa ~ lcavol
## Model 2: lpsa ~ lcavol + lweight
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     95 58.915
## 2     94 52.966  1    5.9485 10.557 0.001606 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model All:
RSE = .7084 RMSE = .674751 R^2 = .6548 Adjusted R^2 = .6234 Beta (fail to reject): 5 predictors out of 8 AIC = 218.9522 BIC = 244.6993

Model lcavol:
RSE = .7875 RMSE = .7793386 R^2 = .5394 Adjusted R^2 = .5346 Beta fails = 0 out of 1 AIC = 232.9522 BIC = 240.6321

Model 5:
RSE = .746
RMSE =.7225684 R^2 = .6041 Adjusted R^2 = .5823 Beta fails = 3 out of 5 AIC = 226.2351 BIC = 244.2581

Model 4:
RSE = .748
RMSE = .7284869 R^2 = .5976
Adjusted R^2 = .5801 beta fails = 2 out of 4 AIC = 225.8177 BIC = 241.2659

Model 2:
RSE = .7506
RMSE = .7389478 R^2 = .5859 Adjusted R^2 = .5771 beta fails = 0 out of 2 AIC = 224.5837 BIC = 234.8825

#Explaining Test Results

On the basis of selecting a model to *explain*, it is important to keep such models small as it is easier to derive and explain relationships between the predictor variables and the explanatory variable. On that basis the smaller models considered are favored over the larger ones. Firstly, all models meet the constant variance and normality assumption of errors which is an important factor in considering models for explaining. The next important aspect to consider is the significance of regression which is measured through the F test of all predictors in a model, the T test of the individual predictors in a model, and anova tests between models at a significance level of .05. In terms of the F tests, all models would fail to accept the null suggesting a linear relationship between the model predictors and the explanatory variable. In terms of the individual T tests, models `model_2` and `model_lcavol` were the only models with betas that would fail to reject suggesting individual linear relationships between the predictors and the explanatory variable. This is an important aspect in explaining an output as we can prove the existence of a linear relationship between variables. In terms of anova testing, the model `model_all` was preferred to all other models meaning there may be a linear relationship with additional variables in the model that were not captured in the other models. Using anova on the latter models, we see no significant difference between such models except when they are compared to `model_lcavol` - all models are preferred to such. The last measures used were AIC and BIC, which are typically used in model selection. In terms of AIC the smallest value was attributed to `model_all` making it the best selection and the second smallest was with attributed to `model_2`. In terms of BIC the smallest value was attributed to `model_2` but the largest was attributed to `model_all`. Typically the AIC and BIC agree in picking the best model but since they are on two separate spectrums, we can conclude that `model_2` is the best selection based on these measures.

#Conclusion

The best model for *explaining* is `model_2`. This is due to it being a smaller model (the second smallest), the proven linear relationship between each predictor and the explanatory variable (T-test/F-test), the outcome of the AIC (second best) and BIC (best) measures. Lastly, in terms of the anova testing we saw this model as being not significantly different from `model_5` and `model_4` suggesting the `model_2` is the better option. Stacked up against `model_lcavol` we find that the additional beta in `model_2` is significant, making `model_2` the preferred choice. When using anova to compare `model_all` to all other models, we see it is a better option but given the model's performance in other areas of testing and the preference of smaller models, we dismiss the `model_all`.

In terms of prediction, RSE, RMSE, R^2 and Adjusted R^2 are better measures to consider but because we are trying to *explain*, those aspects were not as heavily considered as much as the results of testing for linearity and thus proving a relationship that can be explained.

##Part 2

```
boston = MASS::Boston
library(MASS)
set.seed(42)
train_index = sample(1:nrow(Boston), 400)
train_data = boston[train_index,]
test_data = boston[-train_index,]
Model_ALL = lm(medv ~ ., data = train_data)
Model_crim = lm(medv ~ crim, data = train_data)
Model_6 = lm(medv ~ crim + indus + nox + rm + age + dis, data = train_data)
Model_5 = lm(medv ~ crim + indus + nox + rm + age, data = train_data)
Model_3 = lm(medv ~ crim + nox + dis, data = train_data)
Y = test_data[,14]

#Model All

RSE_ALL = summary(Model_ALL)$sigma
Train_RMSE_ALL = sqrt((RSE_ALL^2)*Model_ALL$df.residual/length(Model_ALL$fitted.values))

beta_ALL = as.vector(Model_ALL$coefficients)
X_ALL = cbind(1,test_data[,-14])
Y_hat_ALL = as.matrix(X_ALL) %*% beta_ALL
SSE_ALL = sum((Y - Y_hat_ALL)^2)
Test_RMSE_ALL = sqrt(SSE_ALL/length(Y))
```

```r
#Train RMSE = 4.675465 Test RMSE = 4.767746


#Model crim

RSE_crim = summary(Model_crim)$sigma
Train_RMSE_crim = sqrt((RSE_crim^2)*Model_crim$df.residual/length(Model_crim$fitted.values))

beta_crim = as.vector(Model_crim$coefficients)
X_crim = cbind(1,test_data[,1])
Y_hat_crim = as.matrix(X_crim) %*% beta_crim
SSE_crim = sum((Y - Y_hat_crim)^2)
Test_RMSE_crim = sqrt(SSE_crim/length(Y))

#Train RMSE = 8.238496  Test RMSE = 9.318085


#Model 6

RSE_6 = summary(Model_6)$sigma
Train_RMSE_6 = sqrt((RSE_6^2)*Model_6$df.residual/length(Model_6$fitted.values))

beta_6 = as.vector(Model_6$coefficients)
X_6 = cbind(1,test_data[,c("crim","indus","nox","rm","age","dis")])
Y_hat_6 = as.matrix(X_6) %*% beta_6
SSE_6 = sum((Y - Y_hat_6)^2)
Test_RMSE_6 = sqrt(SSE_6/length(Y))

#Train RMSE = 5.758958  Test RMSE = 5.95507


#Model 5

RSE_5 = summary(Model_5)$sigma
Train_RMSE_5 = sqrt((RSE_5^2)*Model_5$df.residual/length(Model_5$fitted.values))

beta_5 = as.vector(Model_5$coefficients)
X_5 = cbind(1,test_data[,c("crim","indus","nox","rm","age")])
Y_hat_5 = as.matrix(X_5) %*% beta_5
SSE_5 = sum((Y - Y_hat_5)^2)
Test_RMSE_5 = sqrt(SSE_5/length(Y))

#Train RMSE = 5.995325  Test RMSE = 6.148281


#Model 3

RSE_3 = summary(Model_3)$sigma
Train_RMSE_3 = sqrt((RSE_3^2)*Model_3$df.residual/length(Model_3$fitted.values))

beta_3 = as.vector(Model_3$coefficients)
X_3 = cbind(1,test_data[,c("crim","nox","dis")])
Y_hat_3 = as.matrix(X_3) %*% beta_3
```

```r
SSE_3 = sum((Y - Y_hat_3)^2)
Test_RMSE_3 = sqrt(SSE_3/length(Y))

#Train RMSE = 7.72839 Test RMSE = 8.643548

ALL = c(Test_RMSE_ALL, Train_RMSE_ALL) # Variance = .004258
crim = c(Test_RMSE_crim, Train_RMSE_crim) #Variance = .58276
Six = c(Test_RMSE_6, Train_RMSE_6) #Variance = .01923
Five = c(Test_RMSE_5, Train_RMSE_5) #Variance = .01170
Three = c(Test_RMSE_3, Train_RMSE_3) #Variance = .41876
summary(Model_ALL)
```

```
##
## Call:
## lm(formula = medv ~ ., data = train_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.3126  -2.7134  -0.5522   1.5431  25.5431
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  40.211363   5.823305   6.905 2.07e-11 ***
## crim         -0.121911   0.034032  -3.582 0.000384 ***
## zn            0.037754   0.016166   2.335 0.020038 *
## indus         0.002787   0.069150   0.040 0.967867
## chas          1.918167   0.999327   1.919 0.055663 .
## nox         -17.987178   4.304668  -4.179 3.63e-05 ***
## rm            3.478935   0.457299   7.608 2.16e-13 ***
## age          -0.003087   0.014798  -0.209 0.834880
## dis          -1.456826   0.230828  -6.311 7.60e-10 ***
## rad           0.310637   0.074539   4.167 3.81e-05 ***
## tax          -0.011081   0.004234  -2.617 0.009212 **
## ptratio      -0.996107   0.148701  -6.699 7.45e-11 ***
## black         0.007692   0.003214   2.393 0.017194 *
## lstat        -0.533910   0.055318  -9.652  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.759 on 386 degrees of freedom
## Multiple R-squared:  0.7262, Adjusted R-squared:  0.7169
## F-statistic: 78.73 on 13 and 386 DF,  p-value: < 2.2e-16
```

```r
summary(Model_crim)
```

```
##
## Call:
## lm(formula = medv ~ crim, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.734  -5.147  -1.788   2.329  29.619
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 23.80157    0.44704  53.243  < 2e-16 ***
## crim        -0.37045    0.04425  -8.372 9.75e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.259 on 398 degrees of freedom
## Multiple R-squared:  0.1497, Adjusted R-squared:  0.1476
## F-statistic: 70.09 on 1 and 398 DF,  p-value: 9.753e-16
```

```
summary(Model_6)
```

```
##
## Call:
## lm(formula = medv ~ crim + indus + nox + rm + age + dis, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.219  -3.166  -0.600   2.097  37.638
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.93105    4.52606  -0.206 0.837125
## crim         -0.19154    0.03456  -5.543 5.47e-08 ***
## indus        -0.23006    0.07173  -3.207 0.001451 **
## nox         -12.17993    4.66224  -2.612 0.009334 **
## rm            6.90354    0.45293  15.242  < 2e-16 ***
## age          -0.06287    0.01689  -3.723 0.000226 ***
## dis          -1.45760    0.25404  -5.738 1.92e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.81 on 393 degrees of freedom
## Multiple R-squared:  0.5845, Adjusted R-squared:  0.5782
## F-statistic: 92.15 on 6 and 393 DF,  p-value: < 2.2e-16
```

```
summary(Model_5)
```

```
##
## Call:
## lm(formula = medv ~ crim + indus + nox + rm + age, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.127  -3.120  -0.847   2.174  39.005
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.64406    3.60183  -4.899 1.41e-06 ***
## crim         -0.18183    0.03589  -5.067 6.23e-07 ***
## indus        -0.11816    0.07177  -1.646    0.101
## nox          -3.78538    4.60257  -0.822    0.411
## rm            7.28048    0.46594  15.625  < 2e-16 ***
## age          -0.02127    0.01586  -1.341    0.181
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.041 on 394 degrees of freedom
## Multiple R-squared:  0.5497, Adjusted R-squared:  0.544
## F-statistic:  96.2 on 5 and 394 DF,  p-value: < 2.2e-16
```

```r
summary(Model_3)
```

```
##
## Call:
## lm(formula = medv ~ crim + nox + dis, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.488  -4.947  -2.063   2.625  29.138
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  48.09033    3.85312  12.481  < 2e-16 ***
## crim         -0.26589    0.04577  -5.809 1.29e-08 ***
## nox         -37.35806    5.35091  -6.982 1.24e-11 ***
## dis          -1.02654    0.29019  -3.537 0.000452 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.767 on 396 degrees of freedom
## Multiple R-squared:  0.2518, Adjusted R-squared:  0.2461
## F-statistic: 44.42 on 3 and 396 DF,  p-value: < 2.2e-16
```

```r
#LOOCV RMSE

calc_loocv_rmse = function(model) {sqrt(mean((resid(model) / (1 - hatvalues(model))) ^ 2))}
calc_loocv_rmse(Model_ALL)
```

```
## [1] 4.908037
```

```r
calc_loocv_rmse(Model_crim)
```

```
## [1] 8.310566
```

```r
calc_loocv_rmse(Model_6)
```

```
## [1] 5.875081
```

```r
calc_loocv_rmse(Model_5)
```

```
## [1] 6.107382
```

```r
calc_loocv_rmse(Model_3)
```

```
## [1] 7.805229
```

```r
#Train RMSE
Train_RMSE_ALL
```

```
## [1] 4.675465
```

```r
Train_RMSE_crim
```

```
## [1] 8.238496
```

```r
Train_RMSE_6
```

```
## [1] 5.758958
```

```r
Train_RMSE_5
```

```
## [1] 5.995325
```

```r
Train_RMSE_3
```

```
## [1] 7.72839
```

```r
#Test RMSE
Test_RMSE_ALL
```

```
## [1] 4.767746
```

```r
Test_RMSE_crim
```

```
## [1] 9.318085
```

```r
Test_RMSE_6
```

```
## [1] 5.95507
```

```r
Test_RMSE_5
```

```
## [1] 6.148281
```

```r
Test_RMSE_3
```

```
## [1] 8.643548
```

```r
#AIC and BIC
AIC(Model_ALL)
```

```
## [1] 2399.014
```

```r
AIC(Model_crim)
```

```
## [1] 2828.205
```

```r
AIC(Model_6)
```

```
## [1] 2551.756
```

```r
AIC(Model_5)
```

```
## [1] 2581.935
```

```r
AIC(Model_3)
```

```
## [1] 2781.071
```

```r
BIC(Model_ALL)
```

```
## [1] 2458.886
```

```r
BIC(Model_crim)
```

```
## [1] 2840.179
```

```r
BIC(Model_6)
```

```
## [1] 2583.688
```

```r
BIC(Model_5)
```

```
## [1] 2609.875
```

```r
BIC(Model_3)
```

```
## [1] 2801.029
```

Model All Train RMSE = 4.675465 Test RMSE = 4.767746
LOOCV RMSE = 4.908037 Variance = .004258
R^2= .7262 Adjusted R^2 = .7169 AIC = 2399.014 BIC = 2458.886

Model crim Train RMSE = 8.238496 Test RMSE = 9.318085
LOOCV RMSE = 8.310566 Variance = .58276
R^2 = .1497 Adjusted R^2 = .1476 AIC = 2828.205 BIC = 2840.179

Model 6 Train RMSE = 5.758958 Test RMSE = 5.95507 LOOCV RMSE = 5.875081 Variance = .01923
R^2 = .5845 Adjusted R^2 = .5782 AIC = 2551.756 BIC = 2583.688

Model 5 Train RMSE = 5.995325 Test RMSE = 6.148281
LOOCV RMSE = 6.107382 Variance = .01170
R^2 = .5497 Adjusted R^2 = .544 AIC = 2581.935 BIC = 2609.875

Model 3 Train RMSE = 7.72839 Test RMSE = 8.643548
LOOCV RMSE = 7.805229 Variance = .41876
R^2 = .2518 Adjusted R^2 = .2461 AIC = 2781.071 BIC = 2801.029

The model that is the best for *predicting* the response variable `medv` is the `Model_ALL` which contains all predictors. This is based on the fact that it had the smallest RMSE for train and test data and the smallest LOOCV RMSE. This is important because the smaller the RMSE, the less error is attributed to the model. It is also important in regard to the LOOCV RMSE because this RMSE measure implicity penalizes models for having more predictors. Secondly, this is due to this model having the smallest variance, or spread, between the RMSE calculated from the test data and the train data. Lastly, this choice is based on the fact that such model has the highest R^2 and Adjusted R^2 values. This is important because these values both essentially give the percentage of variation in the response variable that is described by the model. These factors are weighed heavily as they provide measures of error that are attributed to a model. The less error, the better for predicting. It is also important to note, in terms of AIC and BIC statistics, `Model_ALL` is considered the best model.