STA 5207: Homework 5

Due: Friday, February 23 by 11:59 PM

Include your R code in an R chunks as part of your answer. In addition, your written answer to each exercise should be self-contained so that the grader can determine your solution without reading your code or deciphering its output.

Exercise 1 (Using `step`) [40 points]

For this exercise we will use the `prostate` data set from the `faraway` package. You can also find the data in `prostate.csv` on Canvas. The data set comes from a study on 97 men with prostate cancer who were due to receive a radical prostatectomy. The variables in the data set are

- `lcavol`: log(cancer volume).
- `lweight`: log(prostate weight).
- `age`: The patient's age in years.
- `lbph`: log(benign prostatic hyperplasia amount).
- `svi`: Seminal vesicle invasion.
- `lcp`: log(capsular penetration).
- `gleason`: Gleason score.
- `pgg45`: percentage Gleason score 4 or 5.
- `lpsa`: log(prostate specific antigen).

In the following exercises, use `lpsa` as the response and the other variables as predictors.

1. (6 points) Identify the best model based on AIC and BIC using forward selection. Create a table listing each quality criterion (AIC, BIC) and the subset of variables chosen by the method.

```
data(prostate, package = 'faraway')
mod_start <- lm(lpsa ~ 1, data=prostate)
mod_forwd_aic <- step(mod_start, scope=lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
```

```
## Start:  AIC=28.84
## lpsa ~ 1
##
##           Df Sum of Sq     RSS     AIC
## + lcavol   1    69.003  58.915 -44.366
## + svi      1    41.011  86.907  -6.658
## + lcp      1    38.528  89.389  -3.926
## + pgg45    1    22.814 105.103  11.783
## + gleason  1    17.416 110.501  16.641
## + lweight  1    16.041 111.876  17.840
## + lbph     1     4.136 123.782  27.650
## + age      1     3.679 124.238  28.007
## <none>                127.918  28.837
```

```
##
## Step:  AIC=-44.37
## lpsa ~ lcavol
##
##           Df Sum of Sq    RSS     AIC
## + lweight  1    5.9485 52.966 -52.690
## + svi      1    5.2375 53.677 -51.397
## + lbph     1    3.2658 55.649 -47.898
## + pgg45    1    1.6980 57.217 -45.203
## <none>                 58.915 -44.366
## + lcp      1    0.6562 58.259 -43.453
## + gleason  1    0.4156 58.499 -43.053
## + age      1    0.0025 58.912 -42.370
##
## Step:  AIC=-52.69
## lpsa ~ lcavol + lweight
##
##           Df Sum of Sq    RSS     AIC
## + svi      1    5.1814 47.785 -60.676
## + pgg45    1    1.9489 51.017 -54.327
## <none>                 52.966 -52.690
## + lcp      1    0.8371 52.129 -52.236
## + gleason  1    0.7810 52.185 -52.131
## + lbph     1    0.6751 52.291 -51.935
## + age      1    0.4200 52.546 -51.463
##
## Step:  AIC=-60.68
## lpsa ~ lcavol + lweight + svi
##
##           Df Sum of Sq    RSS     AIC
## + lbph     1   1.30006 46.485 -61.352
## <none>                 47.785 -60.676
## + pgg45    1   0.57347 47.211 -59.847
## + age      1   0.40251 47.382 -59.497
## + gleason  1   0.38901 47.396 -59.469
## + lcp      1   0.06412 47.721 -58.806
##
## Step:  AIC=-61.35
## lpsa ~ lcavol + lweight + svi + lbph
##
##           Df Sum of Sq    RSS     AIC
## + age      1   0.95924 45.526 -61.374
## <none>                 46.485 -61.352
## + pgg45    1   0.35332 46.131 -60.092
## + gleason  1   0.21256 46.272 -59.796
## + lcp      1   0.10230 46.383 -59.565
##
## Step:  AIC=-61.37
## lpsa ~ lcavol + lweight + svi + lbph + age
##
##           Df Sum of Sq    RSS     AIC
## <none>                 45.526 -61.374
## + pgg45    1   0.65896 44.867 -60.789
## + gleason  1   0.45601 45.070 -60.351
```

```
## + lcp      1   0.12927 45.396 -59.650
```

```
n <- nrow(prostate)
mod_forwd_bic <- step(mod_start, scope=lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
```

```
## Start:  AIC=31.41
## lpsa ~ 1
##
##           Df Sum of Sq     RSS     AIC
## + lcavol   1    69.003  58.915 -39.217
## + svi      1    41.011  86.907  -1.508
## + lcp      1    38.528  89.389   1.224
## + pgg45    1    22.814 105.103  16.932
## + gleason  1    17.416 110.501  21.790
## + lweight  1    16.041 111.876  22.990
## <none>                 127.918  31.412
## + lbph     1     4.136 123.782  32.799
## + age      1     3.679 124.238  33.156
##
## Step:  AIC=-39.22
## lpsa ~ lcavol
##
##           Df Sum of Sq    RSS     AIC
## + lweight  1    5.9485 52.966 -44.966
## + svi      1    5.2375 53.677 -43.673
## + lbph     1    3.2658 55.649 -40.174
## <none>                 58.915 -39.217
## + pgg45    1    1.6980 57.217 -37.479
## + lcp      1    0.6562 58.259 -35.728
## + gleason  1    0.4156 58.499 -35.329
## + age      1    0.0025 58.912 -34.646
##
## Step:  AIC=-44.97
## lpsa ~ lcavol + lweight
##
##           Df Sum of Sq    RSS     AIC
## + svi      1    5.1814 47.785 -50.377
## <none>                 52.966 -44.966
## + pgg45    1    1.9489 51.017 -44.028
## + lcp      1    0.8371 52.129 -41.937
## + gleason  1    0.7810 52.185 -41.833
## + lbph     1    0.6751 52.291 -41.636
## + age      1    0.4200 52.546 -41.164
##
## Step:  AIC=-50.38
## lpsa ~ lcavol + lweight + svi
##
##           Df Sum of Sq    RSS     AIC
## <none>                 47.785 -50.377
## + lbph     1   1.30006 46.485 -48.478
## + pgg45    1   0.57347 47.211 -46.974
## + age      1   0.40251 47.382 -46.623
## + gleason  1   0.38901 47.396 -46.596
## + lcp      1   0.06412 47.721 -45.933
```

```r
print(coef(mod_forwd_aic))
```

```
## (Intercept)      lcavol     lweight         svi         lbph         age
##  0.95099742  0.56560801  0.42369200  0.72095499  0.11183992 -0.01489225
```

```r
print(coef(mod_forwd_bic))
```

```
## (Intercept)      lcavol     lweight         svi
##  -0.2680926   0.5516380   0.5085413   0.6661584
```

```r
quality_criterion <- c('AIC', 'BIC')
variables <- c('lcavol,lweight,svi,lbph,age', 'lcavol,lweight,svi')
criterion_values <- c(extractAIC(mod_forwd_aic)[2], extractAIC(mod_forwd_bic, k=log(n))[2])
data.frame(quality_criterion, variables, criterion_values)
```

```
##   quality_criterion                   variables criterion_values
## 1               AIC lcavol,lweight,svi,lbph,age        -61.37439
## 2               BIC          lcavol,lweight,svi        -50.37736
```

Answer: The best model using forward selection based on AIC was the model with predictors "lcavol", "lweight", "svi", "lbph", and "age" with a final AIC of -61.37439. The best model using forward selection based on BIC was the model with predictors "lcavol", "lweight", and "svi" with a final BIC of -50.37736. The table above shows the quality criterion used, variables selected, and the criterion values of each model.

2. (6 points) Identify the best model based on AIC and BIC using backward selection. Create a table listing each quality criterion (AIC, BIC) and the subset of variables chosen by the method.

```r
mod_all_preds <- lm(lpsa ~ ., data=prostate)
mod_back_aic <- step(mod_all_preds, direction = 'backward')
```

```
## Start:  AIC=-58.32
## lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
##     pgg45
##
##           Df Sum of Sq    RSS     AIC
## - gleason  1    0.0412 44.204 -60.231
## - pgg45    1    0.5258 44.689 -59.174
## - lcp      1    0.6740 44.837 -58.853
## <none>                 44.163 -58.322
## - age      1    1.5503 45.713 -56.975
## - lbph     1    1.6835 45.847 -56.693
## - lweight  1    3.5861 47.749 -52.749
## - svi      1    4.9355 49.099 -50.046
## - lcavol   1   22.3721 66.535 -20.567
##
## Step:  AIC=-60.23
## lpsa ~ lcavol + lweight + age + lbph + svi + lcp + pgg45
##
##           Df Sum of Sq    RSS     AIC
## - lcp      1    0.6623 44.867 -60.789
## <none>                 44.204 -60.231
```

```
## - pgg45    1    1.1920 45.396 -59.650
## - age      1    1.5166 45.721 -58.959
## - lbph     1    1.7053 45.910 -58.560
## - lweight  1    3.5462 47.750 -54.746
## - svi      1    4.8984 49.103 -52.037
## - lcavol   1   23.5039 67.708 -20.872
##
## Step:  AIC=-60.79
## lpsa ~ lcavol + lweight + age + lbph + svi + pgg45
##
##           Df Sum of Sq    RSS     AIC
## - pgg45    1    0.6590 45.526 -61.374
## <none>                 44.867 -60.789
## - age      1    1.2649 46.131 -60.092
## - lbph     1    1.6465 46.513 -59.293
## - lweight  1    3.5647 48.431 -55.373
## - svi      1    4.2503 49.117 -54.009
## - lcavol   1   25.4189 70.285 -19.248
##
## Step:  AIC=-61.37
## lpsa ~ lcavol + lweight + age + lbph + svi
##
##           Df Sum of Sq    RSS     AIC
## <none>                 45.526 -61.374
## - age      1    0.9592 46.485 -61.352
## - lbph     1    1.8568 47.382 -59.497
## - lweight  1    3.2251 48.751 -56.735
## - svi      1    5.9517 51.477 -51.456
## - lcavol   1   28.7665 74.292 -15.871
```

```
n <- nrow(prostate)
mod_back_bic <- step(mod_all_preds, direction = 'backward', k=log(n))
```

```
## Start:  AIC=-35.15
## lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
##     pgg45
##
##           Df Sum of Sq    RSS     AIC
## - gleason  1    0.0412 44.204 -39.634
## - pgg45    1    0.5258 44.689 -38.576
## - lcp      1    0.6740 44.837 -38.255
## - age      1    1.5503 45.713 -36.377
## - lbph     1    1.6835 45.847 -36.095
## <none>                 44.163 -35.149
## - lweight  1    3.5861 47.749 -32.151
## - svi      1    4.9355 49.099 -29.448
## - lcavol   1   22.3721 66.535   0.030
##
## Step:  AIC=-39.63
## lpsa ~ lcavol + lweight + age + lbph + svi + lcp + pgg45
##
##           Df Sum of Sq    RSS     AIC
## - lcp      1    0.6623 44.867 -42.766
## - pgg45    1    1.1920 45.396 -41.627
```

```
## - age      1    1.5166 45.721 -40.936
## - lbph     1    1.7053 45.910 -40.537
## <none>                 44.204 -39.634
## - lweight  1    3.5462 47.750 -36.723
## - svi      1    4.8984 49.103 -34.014
## - lcavol   1   23.5039 67.708  -2.849
##
## Step:  AIC=-42.77
## lpsa ~ lcavol + lweight + age + lbph + svi + pgg45
##
##           Df Sum of Sq    RSS     AIC
## - pgg45    1    0.6590 45.526 -45.926
## - age      1    1.2649 46.131 -44.644
## - lbph     1    1.6465 46.513 -43.844
## <none>                 44.867 -42.766
## - lweight  1    3.5647 48.431 -39.925
## - svi      1    4.2503 49.117 -38.561
## - lcavol   1   25.4189 70.285  -3.800
##
## Step:  AIC=-45.93
## lpsa ~ lcavol + lweight + age + lbph + svi
##
##           Df Sum of Sq    RSS     AIC
## - age      1    0.9592 46.485 -48.478
## - lbph     1    1.8568 47.382 -46.623
## <none>                 45.526 -45.926
## - lweight  1    3.2251 48.751 -43.862
## - svi      1    5.9517 51.477 -38.583
## - lcavol   1   28.7665 74.292  -2.997
##
## Step:  AIC=-48.48
## lpsa ~ lcavol + lweight + lbph + svi
##
##           Df Sum of Sq    RSS     AIC
## - lbph     1    1.3001 47.785 -50.377
## <none>                 46.485 -48.478
## - lweight  1    2.8014 49.286 -47.377
## - svi      1    5.8063 52.291 -41.636
## - lcavol   1   27.8298 74.315  -7.542
##
## Step:  AIC=-50.38
## lpsa ~ lcavol + lweight + svi
##
##           Df Sum of Sq    RSS     AIC
## <none>                 47.785 -50.377
## - svi      1    5.1814 52.966 -44.966
## - lweight  1    5.8924 53.677 -43.673
## - lcavol   1   28.0445 75.829 -10.160
```

```
print(coef(mod_back_aic))
```

```
## (Intercept)      lcavol     lweight         age        lbph         svi
##  0.95099742  0.56560801  0.42369200 -0.01489225  0.11183992  0.72095499
```

```r
print(coef(mod_back_bic))
```

```
## (Intercept)      lcavol     lweight         svi
##  -0.2680926   0.5516380   0.5085413   0.6661584
```

```r
quality_criterion <- c('AIC', 'BIC')
variables <- c('lcavol,lweight,svi,lbph,age', 'lcavol,lweight,svi')
criterion_values <- c(extractAIC(mod_back_aic)[2], extractAIC(mod_back_bic, k=log(n))[2])
data.frame(quality_criterion, variables, criterion_values)
```

```
##   quality_criterion                   variables criterion_values
## 1               AIC lcavol,lweight,svi,lbph,age        -61.37439
## 2               BIC          lcavol,lweight,svi        -50.37736
```

Answer: The best model using backward selection based on AIC was the model with predictors "lcavol", "lweight", "svi", "lbph", and "age" with a final AIC of -61.37439. The best model using backward selection based on BIC was the model with predictors "lcavol", "lweight", and "svi" with a final BIC of -50.37736. The table above shows the quality criterion used, variables selected, and the criterion values of each model.

3. (6 points) Identify the best model based on AIC and BIC using stepwise selection. Create a table listing each quality criterion (AIC, BIC) and the subset of variables chosen by the method.

```r
mod_start <- lm(lpsa ~ 1, data=prostate)
mod_stepwise_aic <- step(mod_start, scope=lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleasc
```

```
## Start:  AIC=28.84
## lpsa ~ 1
##
##           Df Sum of Sq     RSS     AIC
## + lcavol   1    69.003  58.915 -44.366
## + svi      1    41.011  86.907  -6.658
## + lcp      1    38.528  89.389  -3.926
## + pgg45    1    22.814 105.103  11.783
## + gleason  1    17.416 110.501  16.641
## + lweight  1    16.041 111.876  17.840
## + lbph     1     4.136 123.782  27.650
## + age      1     3.679 124.238  28.007
## <none>                127.918  28.837
##
## Step:  AIC=-44.37
## lpsa ~ lcavol
##
##           Df Sum of Sq     RSS     AIC
## + lweight  1     5.949  52.966 -52.690
## + svi      1     5.237  53.677 -51.397
## + lbph     1     3.266  55.649 -47.898
## + pgg45    1     1.698  57.217 -45.203
## <none>                  58.915 -44.366
## + lcp      1     0.656  58.259 -43.453
## + gleason  1     0.416  58.499 -43.053
## + age      1     0.003  58.912 -42.370
```

```
## - lcavol   1    69.003 127.918  28.837
##
## Step:  AIC=-52.69
## lpsa ~ lcavol + lweight
##
##           Df Sum of Sq     RSS     AIC
## + svi      1     5.181  47.785 -60.676
## + pgg45    1     1.949  51.017 -54.327
## <none>                   52.966 -52.690
## + lcp      1     0.837  52.129 -52.236
## + gleason  1     0.781  52.185 -52.131
## + lbph     1     0.675  52.291 -51.935
## + age      1     0.420  52.546 -51.463
## - lweight  1     5.949  58.915 -44.366
## - lcavol   1    58.910 111.876  17.840
##
## Step:  AIC=-60.68
## lpsa ~ lcavol + lweight + svi
##
##           Df Sum of Sq    RSS     AIC
## + lbph     1    1.3001 46.485 -61.352
## <none>                 47.785 -60.676
## + pgg45    1    0.5735 47.211 -59.847
## + age      1    0.4025 47.382 -59.497
## + gleason  1    0.3890 47.396 -59.469
## + lcp      1    0.0641 47.721 -58.806
## - svi      1    5.1814 52.966 -52.690
## - lweight  1    5.8924 53.677 -51.397
## - lcavol   1   28.0445 75.829 -17.884
##
## Step:  AIC=-61.35
## lpsa ~ lcavol + lweight + svi + lbph
##
##           Df Sum of Sq    RSS     AIC
## + age      1    0.9592 45.526 -61.374
## <none>                 46.485 -61.352
## - lbph     1    1.3001 47.785 -60.676
## + pgg45    1    0.3533 46.131 -60.092
## + gleason  1    0.2126 46.272 -59.796
## + lcp      1    0.1023 46.383 -59.565
## - lweight  1    2.8014 49.286 -57.676
## - svi      1    5.8063 52.291 -51.935
## - lcavol   1   27.8298 74.315 -17.841
##
## Step:  AIC=-61.37
## lpsa ~ lcavol + lweight + svi + lbph + age
##
##           Df Sum of Sq    RSS     AIC
## <none>                 45.526 -61.374
## - age      1    0.9592 46.485 -61.352
## + pgg45    1    0.6590 44.867 -60.789
## + gleason  1    0.4560 45.070 -60.351
## + lcp      1    0.1293 45.396 -59.650
## - lbph     1    1.8568 47.382 -59.497
```

```
## - lweight  1    3.2251 48.751 -56.735
## - svi      1    5.9517 51.477 -51.456
## - lcavol   1   28.7665 74.292 -15.871
```

```
n <- nrow(prostate)
mod_stepwise_bic <- step(mod_start, scope=lpsa ~ lcavol + lweight + age + lbph + svi + lcp + gleas
```

```
## Start:  AIC=31.41
## lpsa ~ 1
##
##           Df Sum of Sq     RSS     AIC
## + lcavol   1    69.003  58.915 -39.217
## + svi      1    41.011  86.907  -1.508
## + lcp      1    38.528  89.389   1.224
## + pgg45    1    22.814 105.103  16.932
## + gleason  1    17.416 110.501  21.790
## + lweight  1    16.041 111.876  22.990
## <none>                 127.918  31.412
## + lbph     1     4.136 123.782  32.799
## + age      1     3.679 124.238  33.156
##
## Step:  AIC=-39.22
## lpsa ~ lcavol
##
##           Df Sum of Sq     RSS     AIC
## + lweight  1     5.949  52.966 -44.966
## + svi      1     5.237  53.677 -43.673
## + lbph     1     3.266  55.649 -40.174
## <none>                  58.915 -39.217
## + pgg45    1     1.698  57.217 -37.479
## + lcp      1     0.656  58.259 -35.728
## + gleason  1     0.416  58.499 -35.329
## + age      1     0.003  58.912 -34.646
## - lcavol   1    69.003 127.918  31.412
##
## Step:  AIC=-44.97
## lpsa ~ lcavol + lweight
##
##           Df Sum of Sq     RSS     AIC
## + svi      1     5.181  47.785 -50.377
## <none>                  52.966 -44.966
## + pgg45    1     1.949  51.017 -44.028
## + lcp      1     0.837  52.129 -41.937
## + gleason  1     0.781  52.185 -41.833
## + lbph     1     0.675  52.291 -41.636
## + age      1     0.420  52.546 -41.164
## - lweight  1     5.949  58.915 -39.217
## - lcavol   1    58.910 111.876  22.990
##
## Step:  AIC=-50.38
## lpsa ~ lcavol + lweight + svi
##
##           Df Sum of Sq    RSS     AIC
## <none>                 47.785 -50.377
```

```
## + lbph     1    1.3001 46.485 -48.478
## + pgg45    1    0.5735 47.211 -46.974
## + age      1    0.4025 47.382 -46.623
## + gleason  1    0.3890 47.396 -46.596
## + lcp      1    0.0641 47.721 -45.933
## - svi      1    5.1814 52.966 -44.966
## - lweight  1    5.8924 53.677 -43.673
## - lcavol   1   28.0445 75.829 -10.160
```

```
print(coef(mod_stepwise_aic))
```

```
## (Intercept)      lcavol     lweight         svi        lbph         age
##  0.95099742  0.56560801  0.42369200  0.72095499  0.11183992 -0.01489225
```

```
print(coef(mod_stepwise_bic))
```

```
## (Intercept)      lcavol     lweight         svi
##  -0.2680926   0.5516380   0.5085413   0.6661584
```

```
quality_criterion <- c('AIC', 'BIC')
variables <- c('lcavol,lweight,svi,lbph,age', 'lcavol,lweight,svi')
criterion_values <- c(extractAIC(mod_stepwise_aic)[2], extractAIC(mod_stepwise_bic, k=log(n))[2])
data.frame(quality_criterion, variables, criterion_values)
```

```
##   quality_criterion                   variables criterion_values
## 1               AIC lcavol,lweight,svi,lbph,age        -61.37439
## 2               BIC          lcavol,lweight,svi        -50.37736
```

Answer: The best model using stepwise selection based on AIC was the model with predictors "lcavol", "lweight", "svi", "lbph", and "age" with a final AIC of -61.37439. The best model using stepwise selection based on BIC was the model with predictors "lcavol", "lweight", and "svi" with a final BIC of -50.37736. The table above shows the quality criterion used, variables selected, and the criterion values of each model.

4. (12 points) Identify the best model based on R_a^2, AIC, and BIC using best subset selection. Create a table listing each quality criterion (R_a^2, AIC, BIC) and the subset of variables chosen by the method.

```
library(leaps)
mod_exhaustive = summary(regsubsets(lpsa ~ ., data=prostate, nvmax = 8))
bestr2 <- mod_exhaustive$which[which.max(mod_exhaustive$adjr2),]
p <- ncol(mod_exhaustive$which)
mod_aic <- n * log(mod_exhaustive$rss / n) + 2 * (2:p)
mod_bic <- n * log(mod_exhaustive$rss / n) + log(n) * (2:p)
bestaic <- mod_exhaustive$which[which.min(mod_aic),]
bestr2mod <- lm(lpsa~lcavol+lweight+age+lbph+svi+lcp+pgg45, data=prostate)
bestaicmod <- lm(lpsa~lcavol+lweight+age+lbph+svi, data=prostate)
bestbic <- mod_exhaustive$which[which.min(mod_bic),]
bestbicmod <- lm(lpsa~lcavol+lweight+svi, data=prostate)
quality_criterion <- c('R2_a', 'AIC', 'BIC')
variables_chosen <- c('lcavol, lweight, age, lbph, svi, lcp, pgg45', 'lcavol, lweight, age, lbph, s
criterion_values <- c(max(mod_exhaustive$adjr2), min(mod_aic), min(mod_bic))
data.frame(quality_criterion, variables_chosen, criterion_values)
```

```
##   quality_criterion                              variables_chosen
## 1              R2_a lcavol, lweight, age, lbph, svi, lcp, pgg45
## 2               AIC             lcavol, lweight, age, lbph, svi
## 3               BIC                          lcavol, lweight, svi
##   criterion_values
## 1        0.6272521
## 2      -61.3743920
## 3      -50.3773618
```

Answer: Using best subset selection, the best model based on R_a^2 is the model with predictors "lcavol", "lweight", "age", "lbph", "svi", "lcp", and "pgg45". Using best subset selection, the best model based on AIC is the model with predictors "lcavol", "lweight", "age", "lbph", and "svi". Using best subset selection, the best model based on BIC is the model with predictors "lcavol", "lweight", and "svi".

5. (10 points) For each unique candidate model chosen in parts 1 - 4, report their $\text{RMSE}_{\text{LOOCV}}$. Which model do you prefer based on this criteria?

```
calc_loocv_rmse = function(model) {
  sqrt(mean((resid(model) / (1 - hatvalues(model))) ^ 2))
}
partone_aic =  calc_loocv_rmse(mod_forwd_aic)
partone_bic = calc_loocv_rmse(mod_forwd_bic)
parttwo_aic = calc_loocv_rmse(mod_back_aic)
parttwo_bic = calc_loocv_rmse(mod_back_bic)
partthree_aic = calc_loocv_rmse(mod_stepwise_aic)
partthree_bic = calc_loocv_rmse(mod_stepwise_bic)
part_four_rsq = calc_loocv_rmse(bestr2mod)
partfour_aic = calc_loocv_rmse(bestaicmod)
partfour_bic = calc_loocv_rmse(bestbicmod)

LOOCV <- c(partone_aic, partone_bic, parttwo_aic, parttwo_bic, partthree_aic, partthree_bic, part_f
ModelNames <- c("partone_aic", "partone_bic", "parttwo_aic", "parttwo_bic", "partthree_aic", "partt
frame <- data.frame(ModelNames, LOOCV)
frame
```

```
##      ModelNames     LOOCV
## 1   partone_aic 0.7368960
## 2   partone_bic 0.7381178
## 3   parttwo_aic 0.7368960
## 4   parttwo_bic 0.7381178
## 5 partthree_aic 0.7368960
## 6 partthree_bic 0.7381178
## 7 part_four_rsq 0.7410915
## 8  partfour_aic 0.7368960
## 9  partfour_bic 0.7381178
```

```
print(frame$ModelNames[which.min(frame$LOOCV)])
```

```
## [1] "partone_aic"
```

```
print(min(frame$LOOCV))
```

```
## [1] 0.736896
```

Answer: The LOOCV values for each model are reported in the data frame above. The models we prefer are the models with the lowest LOOCV value which is the AIC model from part 1, the AIC model from part 2, the AIC model from part 3, and the AIC model from part 4 which all have predictors chosen of "lcavol", "lweight", "svi", "lbph", and "age".

Exercise 2 (`Boston` Housing Data) [40 points]

For this exercise we will use the `Boston` data set from the `ISLR2` package. You can also find the data in `Boston.csv` on Canvas. The data set contains housing values in 506 suburbs of Boston. There are a total of 12 predictors. You can type `?ISLR2::Boston` in `R` to read about the data set and the meaning of the predictors. In the following exercises, use `crim` (the per capita crime rate) as the response and the other variables as predictors.

1. (6 points) Identify the best model based on AIC and BIC using forward selection. Create a table listing each quality criterion (AIC, BIC) and the subset of variables chosen by the method.

```
data(Boston, package='ISLR2')
mod_start <- lm(crim ~ 1, data=Boston)
mod_forwd_aic <- step(mod_start, scope=crim ~ zn + indus + chas + nox + rm + age + dis + rad + tax
```

```
## Start:  AIC=2178.76
## crim ~ 1
##
##           Df Sum of Sq   RSS    AIC
## + rad      1   14618.6 22745 1929.6
## + tax      1   12689.1 24674 1970.8
## + lstat    1    7756.3 29607 2063.0
## + nox      1    6621.4 30742 2082.1
## + indus    1    6176.5 31187 2089.3
## + medv     1    5633.6 31730 2098.1
## + dis      1    5385.9 31977 2102.0
## + age      1    4648.8 32714 2113.5
## + ptratio  1    3141.1 34222 2136.3
## + rm       1    1796.0 35567 2155.8
## + zn       1    1501.5 35862 2160.0
## <none>                 37363 2178.8
## + chas     1     116.7 37247 2179.2
##
## Step:  AIC=1929.61
## crim ~ rad
##
##           Df Sum of Sq   RSS    AIC
## + lstat    1   1103.70 21641 1906.4
## + medv     1    978.68 21766 1909.3
## + rm       1    302.58 22442 1924.8
## + dis      1    244.47 22500 1926.1
## + age      1    214.87 22530 1926.8
## + chas     1     98.27 22646 1929.4
## <none>                 22745 1929.6
## + nox      1     88.51 22656 1929.6
## + indus    1     68.17 22676 1930.1
## + tax      1     39.19 22705 1930.7
## + zn       1      1.18 22743 1931.6
```

```
## + ptratio  1      0.03 22745 1931.6
##
## Step:  AIC=1906.44
## crim ~ rad + lstat
##
##           Df Sum of Sq   RSS    AIC
## + medv     1   138.075 21503 1905.2
## + zn       1    97.116 21544 1906.2
## <none>                 21641 1906.4
## + chas     1    64.158 21577 1906.9
## + indus    1    53.955 21587 1907.2
## + ptratio  1    43.610 21597 1907.4
## + nox      1    26.311 21615 1907.8
## + dis      1    22.336 21619 1907.9
## + rm       1     9.883 21631 1908.2
## + tax      1     8.847 21632 1908.2
## + age      1     3.274 21638 1908.4
##
## Step:  AIC=1905.2
## crim ~ rad + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## + ptratio  1   144.049 21359 1903.8
## + zn       1   119.115 21384 1904.4
## + rm       1    94.196 21409 1905.0
## <none>                 21503 1905.2
## + indus    1    65.402 21437 1905.7
## + dis      1    56.999 21446 1905.9
## + chas     1    33.130 21470 1906.4
## + tax      1    27.264 21476 1906.6
## + nox      1    22.157 21481 1906.7
## + age      1     0.040 21503 1907.2
##
## Step:  AIC=1903.8
## crim ~ rad + lstat + medv + ptratio
##
##         Df Sum of Sq   RSS    AIC
## <none>               21359 1903.8
## + rm     1    79.092 21280 1903.9
## + zn     1    69.140 21290 1904.2
## + nox    1    63.002 21296 1904.3
## + dis    1    61.114 21298 1904.3
## + indus  1    55.393 21303 1904.5
## + chas   1    40.935 21318 1904.8
## + tax    1    29.551 21329 1905.1
## + age    1     0.001 21359 1905.8
```

```
n <- nrow(Boston)
mod_forwd_bic <- step(mod_start, scope=crim ~ zn + indus + chas + nox + rm + age + dis + rad + tax
```

```
## Start:  AIC=2182.99
## crim ~ 1
##
##           Df Sum of Sq   RSS    AIC
```

```
## + rad      1   14618.6 22745 1938.1
## + tax      1   12689.1 24674 1979.3
## + lstat    1    7756.3 29607 2071.5
## + nox      1    6621.4 30742 2090.5
## + indus    1    6176.5 31187 2097.8
## + medv     1    5633.6 31730 2106.5
## + dis      1    5385.9 31977 2110.4
## + age      1    4648.8 32714 2122.0
## + ptratio  1    3141.1 34222 2144.8
## + rm       1    1796.0 35567 2164.3
## + zn       1    1501.5 35862 2168.5
## <none>                 37363 2183.0
## + chas     1     116.7 37247 2187.6
##
## Step:  AIC=1938.06
## crim ~ rad
##
##           Df Sum of Sq   RSS    AIC
## + lstat    1   1103.70 21641 1919.1
## + medv     1    978.68 21766 1922.0
## + rm       1    302.58 22442 1937.5
## <none>                 22745 1938.1
## + dis      1    244.47 22500 1938.8
## + age      1    214.87 22530 1939.5
## + chas     1     98.27 22646 1942.1
## + nox      1     88.51 22656 1942.3
## + indus    1     68.17 22676 1942.8
## + tax      1     39.19 22705 1943.4
## + zn       1      1.18 22743 1944.3
## + ptratio  1      0.03 22745 1944.3
##
## Step:  AIC=1919.12
## crim ~ rad + lstat
##
##           Df Sum of Sq   RSS    AIC
## <none>                 21641 1919.1
## + medv     1   138.075 21503 1922.1
## + zn       1    97.116 21544 1923.1
## + chas     1    64.158 21577 1923.8
## + indus    1    53.955 21587 1924.1
## + ptratio  1    43.610 21597 1924.3
## + nox      1    26.311 21615 1924.7
## + dis      1    22.336 21619 1924.8
## + rm       1     9.883 21631 1925.1
## + tax      1     8.847 21632 1925.1
## + age      1     3.274 21638 1925.3
```

```
print(coef(mod_forwd_aic))
```

```
## (Intercept)         rad       lstat        medv     ptratio
##   4.9361233   0.5475773   0.1435044  -0.1198319  -0.3070764
```

```r
print(coef(mod_forwd_bic))
```

```
## (Intercept)         rad       lstat
##  -4.3814053   0.5228128   0.2372846
```

```r
quality_criterion <- c('AIC', 'BIC')
variables <- c('rad,lstat,medv,ptratio', 'rad,lstat')
criterion_values <- c(extractAIC(mod_forwd_aic)[2], extractAIC(mod_forwd_bic, k=log(n))[2])
data.frame(quality_criterion, variables, criterion_values)
```

```
##   quality_criterion              variables criterion_values
## 1               AIC rad,lstat,medv,ptratio         1903.797
## 2               BIC              rad,lstat         1919.116
```

Answer: The best model using forward selection based on AIC was the model with predictors "rad", "lstat", "medv", and "ptratio" with a final AIC of 1903.797. The best model using forward selection based on BIC was the model with predictors "rad and "lstat" with a final BIC of 1919.116. The table above shows the quality criterion used, variables selected, and the criterion values of each model.

2. (6 points) Identify the best model based on AIC and BIC using backward selection. Create a table listing each quality criterion (AIC, BIC) and the subset of variables chosen by the method.

```r
data(Boston, package='ISLR2')
mod_start <- lm(crim ~ ., data=Boston)
mod_back_aic <- step(mod_start, direction = 'backward')
```

```
## Start:  AIC=1900.87
## crim ~ zn + indus + chas + nox + rm + age + dis + rad + tax +
##     ptratio + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## - age      1      0.09 20575 1898.9
## - chas     1     20.30 20595 1899.4
## - indus    1     20.31 20595 1899.4
## - tax      1     22.24 20597 1899.4
## - rm       1     44.79 20619 1900.0
## <none>                 20575 1900.9
## - ptratio  1    111.10 20686 1901.6
## - lstat    1    140.23 20715 1902.3
## - nox      1    147.88 20722 1902.5
## - zn       1    246.97 20822 1904.9
## - dis      1    535.94 21111 1911.9
## - medv     1    564.68 21139 1912.6
## - rad      1   2043.03 22618 1946.8
##
## Step:  AIC=1898.87
## crim ~ zn + indus + chas + nox + rm + dis + rad + tax + ptratio +
##     lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## - indus    1     20.31 20595 1897.4
## - chas     1     20.50 20595 1897.4
```

```
## - tax      1     22.34 20597 1897.4
## - rm       1     45.57 20620 1898.0
## <none>                 20575 1898.9
## - ptratio  1    112.36 20687 1899.6
## - lstat    1    152.46 20727 1900.6
## - nox      1    160.61 20735 1900.8
## - zn       1    251.86 20827 1903.0
## - medv     1    564.95 21140 1910.6
## - dis      1    576.47 21151 1910.8
## - rad      1   2060.74 22635 1945.2
##
## Step:  AIC=1897.37
## crim ~ zn + chas + nox + rm + dis + rad + tax + ptratio + lstat +
##     medv
##
##           Df Sum of Sq   RSS    AIC
## - chas     1     25.04 20620 1896.0
## - rm       1     51.61 20647 1896.6
## - tax      1     55.48 20650 1896.7
## <none>                 20595 1897.4
## - ptratio  1    127.12 20722 1898.5
## - lstat    1    144.65 20740 1898.9
## - nox      1    209.68 20805 1900.5
## - zn       1    271.51 20866 1902.0
## - dis      1    556.28 21151 1908.9
## - medv     1    568.32 21163 1909.1
## - rad      1   2397.21 22992 1951.1
##
## Step:  AIC=1895.98
## crim ~ zn + nox + rm + dis + rad + tax + ptratio + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## - tax      1     51.38 20671 1895.2
## - rm       1     53.90 20674 1895.3
## <none>                 20620 1896.0
## - ptratio  1    124.52 20745 1897.0
## - lstat    1    141.65 20762 1897.5
## - nox      1    229.05 20849 1899.6
## - zn       1    276.43 20896 1900.7
## - dis      1    556.72 21177 1907.5
## - medv     1    621.23 21241 1909.0
## - rad      1   2383.04 23003 1949.3
##
## Step:  AIC=1895.24
## crim ~ zn + nox + rm + dis + rad + ptratio + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## - rm       1      59.6 20731 1894.7
## <none>                 20671 1895.2
## - ptratio  1     142.2 20814 1896.7
## - lstat    1     149.6 20821 1896.9
## - zn       1     238.9 20910 1899.1
## - nox      1     285.4 20957 1900.2
## - dis      1     521.0 21192 1905.8
```

```
## - medv     1     579.4 21251 1907.2
## - rad      1    5413.8 26085 2011.0
##
## Step:  AIC=1894.7
## crim ~ zn + nox + dis + rad + ptratio + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## <none>                  20731 1894.7
## - lstat    1     113.7 20845 1895.5
## - ptratio  1     149.9 20881 1896.3
## - zn       1     259.0 20990 1899.0
## - nox      1     283.2 21014 1899.6
## - medv     1     522.7 21254 1905.3
## - dis      1     535.9 21267 1905.6
## - rad      1    5666.7 26398 2015.0
```

```
n <- nrow(Boston)
mod_back_bic <- step(mod_start, direction = 'backward', k=log(n))
```

```
## Start:  AIC=1955.81
## crim ~ zn + indus + chas + nox + rm + age + dis + rad + tax +
##     ptratio + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## - age      1      0.09 20575 1949.6
## - chas     1     20.30 20595 1950.1
## - indus    1     20.31 20595 1950.1
## - tax      1     22.24 20597 1950.1
## - rm       1     44.79 20619 1950.7
## - ptratio  1    111.10 20686 1952.3
## - lstat    1    140.23 20715 1953.0
## - nox      1    147.88 20722 1953.2
## - zn       1    246.97 20822 1955.6
## <none>                 20575 1955.8
## - dis      1    535.94 21111 1962.6
## - medv     1    564.68 21139 1963.3
## - rad      1   2043.03 22618 1997.5
##
## Step:  AIC=1949.59
## crim ~ zn + indus + chas + nox + rm + dis + rad + tax + ptratio +
##     lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## - indus    1     20.31 20595 1943.9
## - chas     1     20.50 20595 1943.9
## - tax      1     22.34 20597 1943.9
## - rm       1     45.57 20620 1944.5
## - ptratio  1    112.36 20687 1946.1
## - lstat    1    152.46 20727 1947.1
## - nox      1    160.61 20735 1947.3
## - zn       1    251.86 20827 1949.5
## <none>                 20575 1949.6
## - medv     1    564.95 21140 1957.1
## - dis      1    576.47 21151 1957.3
```

```
## - rad      1   2060.74 22635 1991.7
##
## Step:  AIC=1943.86
## crim ~ zn + chas + nox + rm + dis + rad + tax + ptratio + lstat +
##     medv
##
##           Df Sum of Sq   RSS    AIC
## - chas     1     25.04 20620 1938.2
## - rm       1     51.61 20647 1938.9
## - tax      1     55.48 20650 1939.0
## - ptratio  1    127.12 20722 1940.8
## - lstat    1    144.65 20740 1941.2
## - nox      1    209.68 20805 1942.8
## <none>                 20595 1943.9
## - zn       1    271.51 20866 1944.3
## - dis      1    556.28 21151 1951.1
## - medv     1    568.32 21163 1951.4
## - rad      1   2397.21 22992 1993.3
##
## Step:  AIC=1938.25
## crim ~ zn + nox + rm + dis + rad + tax + ptratio + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## - tax      1     51.38 20671 1933.3
## - rm       1     53.90 20674 1933.3
## - ptratio  1    124.52 20745 1935.1
## - lstat    1    141.65 20762 1935.5
## - nox      1    229.05 20849 1937.6
## <none>                 20620 1938.2
## - zn       1    276.43 20896 1938.8
## - dis      1    556.72 21177 1945.5
## - medv     1    621.23 21241 1947.0
## - rad      1   2383.04 23003 1987.4
##
## Step:  AIC=1933.28
## crim ~ zn + nox + rm + dis + rad + ptratio + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## - rm       1      59.6 20731 1928.5
## - ptratio  1     142.2 20814 1930.5
## - lstat    1     149.6 20821 1930.7
## - zn       1     238.9 20910 1932.9
## <none>                 20671 1933.3
## - nox      1     285.4 20957 1934.0
## - dis      1     521.0 21192 1939.7
## - medv     1     579.4 21251 1941.0
## - rad      1    5413.8 26085 2044.8
##
## Step:  AIC=1928.51
## crim ~ zn + nox + dis + rad + ptratio + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## - lstat    1     113.7 20845 1925.0
## - ptratio  1     149.9 20881 1925.9
```

```
## <none>                 20731 1928.5
## - zn       1     259.0 20990 1928.6
## - nox      1     283.2 21014 1929.2
## - medv     1     522.7 21254 1934.9
## - dis      1     535.9 21267 1935.2
## - rad      1    5666.7 26398 2044.6
##
## Step:  AIC=1925.05
## crim ~ zn + nox + dis + rad + ptratio + medv
##
##           Df Sum of Sq   RSS    AIC
## - ptratio  1     170.8 21015 1923.0
## - nox      1     258.1 21103 1925.0
## <none>                 20845 1925.0
## - zn       1     279.8 21125 1925.6
## - dis      1     700.2 21545 1935.5
## - medv     1    1519.5 22364 1954.4
## - rad      1    5949.3 26794 2045.9
##
## Step:  AIC=1922.96
## crim ~ zn + nox + dis + rad + medv
##
##        Df Sum of Sq   RSS    AIC
## - nox   1     151.2 21167 1920.4
## <none>              21015 1923.0
## - zn    1     454.6 21470 1927.6
## - dis   1     668.8 21684 1932.6
## - medv  1    1364.0 22379 1948.5
## - rad   1    6388.2 27404 2051.0
##
## Step:  AIC=1920.36
## crim ~ zn + dis + rad + medv
##
##        Df Sum of Sq   RSS    AIC
## <none>              21167 1920.4
## - zn    1     422.6 21589 1924.1
## - dis   1     538.2 21705 1926.8
## - medv  1    1214.7 22381 1942.4
## - rad   1    6465.2 27632 2049.0
```

```
print(coef(mod_forwd_aic))
```

```
## (Intercept)         rad       lstat        medv     ptratio
##   4.9361233   0.5475773   0.1435044  -0.1198319  -0.3070764
```

```
print(coef(mod_forwd_bic))
```

```
## (Intercept)         rad       lstat
##  -4.3814053   0.5228128   0.2372846
```

```
quality_criterion <- c('AIC', 'BIC')
variables <- c('zn,nox,dis,rad,ptratio,lstat,medv', 'zn,dis,rad,medv')
criterion_values <- c(extractAIC(mod_forwd_aic)[2], extractAIC(mod_forwd_bic, k=log(n))[2])
data.frame(quality_criterion, variables, criterion_values)
```

```
##   quality_criterion                          variables criterion_values
## 1               AIC zn,nox,dis,rad,ptratio,lstat,medv         1903.797
## 2               BIC                   zn,dis,rad,medv         1919.116
```

Answer: The best model using backward selection based on AIC was the model with predictors "zn", "nox", "dis", "rad", "ptratio", "lstat" and "medv" with a final AIC of 1894.700. The best model using forward selection based on BIC was the model with predictors "zn", "dis", "rad", and "medv" with a final BIC of 1920.358. The table above shows the quality criterion used, variables selected, and the criterion values of each model.

3. (6 points) Identify the best model based on AIC and BIC using stepwise selection. Create a table listing each quality criterion (AIC, BIC) and the subset of variables chosen by the method.

```r
data(Boston, package='ISLR2')
mod_start <- lm(crim ~ 1, data=Boston)
mod_stepwise_aic <- step(mod_start, scope=crim ~ zn + indus + chas + nox + rm + age + dis + rad + t
```

```
## Start:  AIC=2178.76
## crim ~ 1
##
##           Df Sum of Sq   RSS    AIC
## + rad      1   14618.6 22745 1929.6
## + tax      1   12689.1 24674 1970.8
## + lstat    1    7756.3 29607 2063.0
## + nox      1    6621.4 30742 2082.1
## + indus    1    6176.5 31187 2089.3
## + medv     1    5633.6 31730 2098.1
## + dis      1    5385.9 31977 2102.0
## + age      1    4648.8 32714 2113.5
## + ptratio  1    3141.1 34222 2136.3
## + rm       1    1796.0 35567 2155.8
## + zn       1    1501.5 35862 2160.0
## <none>                 37363 2178.8
## + chas     1     116.7 37247 2179.2
##
## Step:  AIC=1929.61
## crim ~ rad
##
##           Df Sum of Sq   RSS    AIC
## + lstat    1    1103.7 21641 1906.4
## + medv     1     978.7 21766 1909.3
## + rm       1     302.6 22442 1924.8
## + dis      1     244.5 22500 1926.1
## + age      1     214.9 22530 1926.8
## + chas     1      98.3 22646 1929.4
## <none>                 22745 1929.6
## + nox      1      88.5 22656 1929.6
## + indus    1      68.2 22676 1930.1
## + tax      1      39.2 22705 1930.7
## + zn       1       1.2 22743 1931.6
## + ptratio  1       0.0 22745 1931.6
## - rad      1   14618.6 37363 2178.8
##
## Step:  AIC=1906.44
```

```
## crim ~ rad + lstat
##
##           Df Sum of Sq   RSS    AIC
## + medv     1     138.1 21503 1905.2
## + zn       1      97.1 21544 1906.2
## <none>                 21641 1906.4
## + chas     1      64.2 21577 1906.9
## + indus    1      54.0 21587 1907.2
## + ptratio  1      43.6 21597 1907.4
## + nox      1      26.3 21615 1907.8
## + dis      1      22.3 21619 1907.9
## + rm       1       9.9 21631 1908.2
## + tax      1       8.8 21632 1908.2
## + age      1       3.3 21638 1908.4
## - lstat    1    1103.7 22745 1929.6
## - rad      1    7966.0 29607 2063.0
##
## Step:  AIC=1905.2
## crim ~ rad + lstat + medv
##
##           Df Sum of Sq   RSS    AIC
## + ptratio  1     144.0 21359 1903.8
## + zn       1     119.1 21384 1904.4
## + rm       1      94.2 21409 1905.0
## <none>                 21503 1905.2
## + indus    1      65.4 21437 1905.7
## + dis      1      57.0 21446 1905.9
## + chas     1      33.1 21470 1906.4
## - medv     1     138.1 21641 1906.4
## + tax      1      27.3 21476 1906.6
## + nox      1      22.2 21481 1906.7
## + age      1       0.0 21503 1907.2
## - lstat    1     263.1 21766 1909.3
## - rad      1    7880.7 29384 2061.2
##
## Step:  AIC=1903.8
## crim ~ rad + lstat + medv + ptratio
##
##           Df Sum of Sq   RSS    AIC
## <none>                 21359 1903.8
## + rm       1      79.1 21280 1903.9
## + zn       1      69.1 21290 1904.2
## + nox      1      63.0 21296 1904.3
## + dis      1      61.1 21298 1904.3
## + indus    1      55.4 21303 1904.5
## + chas     1      40.9 21318 1904.8
## + tax      1      29.6 21329 1905.1
## - ptratio  1     144.0 21503 1905.2
## + age      1       0.0 21359 1905.8
## - lstat    1     211.6 21570 1906.8
## - medv     1     238.5 21597 1907.4
## - rad      1    7589.6 28948 2055.7
```

```r
n <- nrow(Boston)
mod_stepwise_bic <- step(mod_start, scope=crim ~ zn + indus + chas + nox + rm + age + dis + rad + t
```

```
## Start:  AIC=2182.99
## crim ~ 1
##
##           Df Sum of Sq   RSS    AIC
## + rad      1   14618.6 22745 1938.1
## + tax      1   12689.1 24674 1979.3
## + lstat    1    7756.3 29607 2071.5
## + nox      1    6621.4 30742 2090.5
## + indus    1    6176.5 31187 2097.8
## + medv     1    5633.6 31730 2106.5
## + dis      1    5385.9 31977 2110.4
## + age      1    4648.8 32714 2122.0
## + ptratio  1    3141.1 34222 2144.8
## + rm       1    1796.0 35567 2164.3
## + zn       1    1501.5 35862 2168.5
## <none>                 37363 2183.0
## + chas     1     116.7 37247 2187.6
##
## Step:  AIC=1938.06
## crim ~ rad
##
##           Df Sum of Sq   RSS    AIC
## + lstat    1    1103.7 21641 1919.1
## + medv     1     978.7 21766 1922.0
## + rm       1     302.6 22442 1937.5
## <none>                 22745 1938.1
## + dis      1     244.5 22500 1938.8
## + age      1     214.9 22530 1939.5
## + chas     1      98.3 22646 1942.1
## + nox      1      88.5 22656 1942.3
## + indus    1      68.2 22676 1942.8
## + tax      1      39.2 22705 1943.4
## + zn       1       1.2 22743 1944.3
## + ptratio  1       0.0 22745 1944.3
## - rad      1   14618.6 37363 2183.0
##
## Step:  AIC=1919.12
## crim ~ rad + lstat
##
##           Df Sum of Sq   RSS    AIC
## <none>                 21641 1919.1
## + medv     1     138.1 21503 1922.1
## + zn       1      97.1 21544 1923.1
## + chas     1      64.2 21577 1923.8
## + indus    1      54.0 21587 1924.1
## + ptratio  1      43.6 21597 1924.3
## + nox      1      26.3 21615 1924.7
## + dis      1      22.3 21619 1924.8
## + rm       1       9.9 21631 1925.1
## + tax      1       8.8 21632 1925.1
```

```
## + age      1       3.3 21638 1925.3
## - lstat    1    1103.7 22745 1938.1
## - rad      1    7966.0 29607 2071.5
```

```
print(coef(mod_stepwise_aic))
```

```
## (Intercept)         rad       lstat        medv     ptratio
##   4.9361233   0.5475773   0.1435044  -0.1198319  -0.3070764
```

```
print(coef(mod_stepwise_bic))
```

```
## (Intercept)         rad       lstat
##  -4.3814053   0.5228128   0.2372846
```

```
quality_criterion <- c('AIC', 'BIC')
variables <- c('rad,lstat,medv,ptratio', 'rad,lstat')
criterion_values <- c(extractAIC(mod_stepwise_aic)[2], extractAIC(mod_stepwise_bic, k=log(n))[2])
data.frame(quality_criterion, variables, criterion_values)
```

```
##   quality_criterion              variables criterion_values
## 1               AIC rad,lstat,medv,ptratio         1903.797
## 2               BIC              rad,lstat         1919.116
```

Answer: The best model using stepwise selection based on AIC was the model with predictors "rad", "lstat", "medv", and "ptratio" with a final AIC of 1903.797. The best model using forward selection based on BIC was the model with predictors "rad and "lstat" with a final BIC of 1919.116. The table above shows the quality criterion used, variables selected, and the criterion values of each model.

4. (12 points) Identify the best model based on R_a^2, AIC, and BIC using best subset selection. Note that you have to set `nvmax = 12` when calling `regsubsets`, since there are 12 predictors. Create a table listing each quality criterion (R_a^2, AIC, and BIC) and the subset of the variables chosen by the method.

```
library(leaps)
mod_exhaustive = summary(regsubsets(crim ~ ., data=Boston, nvmax = 12))
bestr2 <- mod_exhaustive$which[which.max(mod_exhaustive$adjr2),]
p <- ncol(mod_exhaustive$which)
mod_aic <- n * log(mod_exhaustive$rss / n) + 2 * (2:p)
mod_bic <- n * log(mod_exhaustive$rss / n) + log(n) * (2:p)
bestaic <- mod_exhaustive$which[which.min(mod_aic),]
bestr2mod <- lm(crim~zn+dis+rad+ptratio+lstat+medv, data=Boston)
bestaicmod <- lm(crim~rad+lstat, data=Boston)
bestbic <- mod_exhaustive$which[which.min(mod_bic),]
bestbicmod <- lm(crim~zn+indus+nox+rm+dis+rad+ptratio+lstat+medv, data=Boston)
quality_criterion <- c('R2_a', 'AIC', 'BIC')
variables_chosen <- c('lcavol, lweight, age, lbph, svi, lcp, pgg45', 'lcavol, lweight, age, lbph, s
criterion_values <- c(max(mod_exhaustive$adjr2), min(mod_aic), min(mod_bic))
data.frame(quality_criterion, variables_chosen, criterion_values)
```

```
##   quality_criterion                            variables_chosen
## 1             R2_a lcavol, lweight, age, lbph, svi, lcp, pgg45
## 2               AIC            lcavol, lweight, age, lbph, svi
## 3               BIC                        lcavol, lweight, svi
```

```
##   criterion_values
## 1        0.4382783
## 2     1894.6997871
## 3     1919.1164600
```

Answer:

5. (10 points) For each unique candidate model chosen in parts 1 - 4, report their $\text{RMSE}_{\text{LOOCV}}$. Which model do you prefer based on this criteria?

```
calc_loocv_rmse = function(model) {
  sqrt(mean((resid(model) / (1 - hatvalues(model))) ^ 2))
}
partone_aic =  calc_loocv_rmse(mod_forwd_aic)
partone_bic = calc_loocv_rmse(mod_forwd_bic)
parttwo_aic = calc_loocv_rmse(mod_back_aic)
parttwo_bic = calc_loocv_rmse(mod_back_bic)
partthree_aic = calc_loocv_rmse(mod_stepwise_aic)
partthree_bic = calc_loocv_rmse(mod_stepwise_bic)
part_four_rsq = calc_loocv_rmse(bestr2mod)
partfour_aic = calc_loocv_rmse(bestaicmod)
partfour_bic = calc_loocv_rmse(bestbicmod)

LOOCV <- c(partone_aic, partone_bic, parttwo_aic, parttwo_bic, partthree_aic, partthree_bic, part_f
ModelNames <- c("partone_aic", "partone_bic", "parttwo_aic", "parttwo_bic", "partthree_aic", "partt
frame <- data.frame(ModelNames, LOOCV)
frame
```

```
##      ModelNames    LOOCV
## 1   partone_aic 6.576221
## 2   partone_bic 6.601046
## 3   parttwo_aic 6.497268
## 4   parttwo_bic 6.532880
## 5 partthree_aic 6.576221
## 6 partthree_bic 6.601046
## 7 part_four_rsq 6.532891
## 8  partfour_aic 6.601046
## 9  partfour_bic 6.509403
```

```
print(frame$ModelNames[which.min(frame$LOOCV)])
```

```
## [1] "parttwo_aic"
```

```
print(min(frame$LOOCV))
```

```
## [1] 6.497268
```

Answer: The LOOCV values for each model are reported in the data frame above. The model we prefer is the model with the lowest LOOCV value which is the AIC model from part 2 which has predictors chosen of “zn”, “nox”, “dis”, “rad”, “ptratio”, “lstat”, and “medv”.

Exercise 3 (Post-Selection Inference and Data Splitting) [20 points]

For this exercise, we will use the `prostate_fake_train.csv` and `prostate_fake_test.csv` data sets on Canvas. These data sets are subsets of the `prostate` data set you analyzed in Exercise 1; however, I replaced the `lpsa` column with a column of noise drawn from a uniform distribution on $[-1, 1]$. I then split the data set into a training subset and a testing subset. I ran the following code:

```
library(tidyverse)

data(prostate, package = 'faraway')

# set random seed for reproducability
set.seed(123456)

# replace the lpsa column with pure noise
prostate_fake = prostate |>
    select(-lpsa) |>
    mutate(noise = runif(nrow(prostate), min = -1, max = 1))

# train/test split
n = nrow(prostate)
train = sample(1:n, size = 49)
test = !(1:n %in% train)

# write data to a file
write_csv(prostate_fake[train,], 'prostate_fake_train.csv')
write_csv(prostate_fake[test,], 'prostate_fake_test.csv')
```

For this exercise, use `noise` as the response and the remaining variables as predictors. Note that by design there is no relationship between `noise` and any of the predictors.

1. (6 points) Identify the best model using AIC and backward selection based on the data in `prostate_fake_train.csv`. Report the subset of the variables chosen by this method.

```
train_data <- read.csv('prostate_fake_train.csv')
mod_start <- lm(noise ~ ., data=train_data)
mod_back_aic <- step(mod_start, direction = 'backward')
```

```
## Start:  AIC=-52.07
## noise ~ lcavol + lweight + age + lbph + svi + lcp + gleason +
##     pgg45
##
##           Df Sum of Sq    RSS     AIC
## - lcavol   1   0.01129 11.737 -54.023
## - lbph     1   0.03342 11.759 -53.931
## - svi      1   0.10170 11.828 -53.647
## - lcp      1   0.14823 11.874 -53.455
## - age      1   0.15526 11.881 -53.426
## <none>                 11.726 -52.070
## - lweight  1   0.63579 12.362 -51.483
## - gleason  1   1.63336 13.359 -47.680
## - pgg45    1   2.26554 13.992 -45.415
##
```

```
## Step:  AIC=-54.02
## noise ~ lweight + age + lbph + svi + lcp + gleason + pgg45
##
##           Df Sum of Sq    RSS     AIC
## - lbph     1   0.04001 11.777 -55.857
## - svi      1   0.12596 11.863 -55.500
## - lcp      1   0.13695 11.874 -55.455
## - age      1   0.16143 11.899 -55.354
## <none>                 11.737 -54.023
## - lweight  1   0.73645 12.474 -53.041
## - gleason  1   1.70302 13.440 -49.384
## - pgg45    1   2.26081 13.998 -47.392
##
## Step:  AIC=-55.86
## noise ~ lweight + age + svi + lcp + gleason + pgg45
##
##           Df Sum of Sq    RSS     AIC
## - lcp      1   0.13186 11.909 -57.311
## - svi      1   0.13496 11.912 -57.298
## - age      1   0.18079 11.958 -57.110
## <none>                 11.777 -55.857
## - lweight  1   0.72303 12.500 -54.937
## - gleason  1   1.68323 13.461 -51.311
## - pgg45    1   2.22327 14.001 -49.383
##
## Step:  AIC=-57.31
## noise ~ lweight + age + svi + gleason + pgg45
##
##           Df Sum of Sq    RSS     AIC
## - svi      1   0.04735 11.957 -59.117
## - age      1   0.11571 12.025 -58.837
## <none>                 11.909 -57.311
## - lweight  1   0.62381 12.533 -56.809
## - pgg45    1   2.09927 14.008 -51.356
## - gleason  1   2.19225 14.101 -51.032
##
## Step:  AIC=-59.12
## noise ~ lweight + age + gleason + pgg45
##
##           Df Sum of Sq    RSS     AIC
## - age      1   0.14110 12.098 -60.542
## <none>                 11.957 -59.117
## - lweight  1   0.77575 12.732 -58.036
## - gleason  1   2.17252 14.129 -52.936
## - pgg45    1   2.49167 14.448 -51.841
##
## Step:  AIC=-60.54
## noise ~ lweight + gleason + pgg45
##
##           Df Sum of Sq    RSS     AIC
## <none>                 12.098 -60.542
## - lweight  1   0.66979 12.767 -59.901
## - gleason  1   2.41864 14.516 -53.611
## - pgg45    1   2.46213 14.560 -53.464
```

```
mod_back_aic$coefficients
```

```
## (Intercept)     lweight     gleason       pgg45
## -1.74783020 -0.27382468  0.45676205 -0.01312037
```

Answer: The best model using AIC and backward selection is the model which uses "lweight", "gleason", and "pgg48" as predictors.

2. (7 points) Using your model from part 1, perform a t-test at the $\alpha = 0.05$ significance level for each predictor. Report the predictors that are significant according to this test. Should we trust the results of this test? Why or why not?

```
train_data <- read.csv('prostate_fake_train.csv')
model <- lm(noise ~ lweight + gleason + pgg45, data=train_data)
summary(model)
```

```
##
## Call:
## lm(formula = noise ~ lweight + gleason + pgg45, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8305 -0.4107  0.0453  0.3612  1.1647
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.747830   1.190764  -1.468  0.14911
## lweight     -0.273825   0.173479  -1.578  0.12147
## gleason      0.456762   0.152282   2.999  0.00440 **
## pgg45       -0.013120   0.004335  -3.026  0.00408 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5185 on 45 degrees of freedom
## Multiple R-squared:  0.2341, Adjusted R-squared:  0.183
## F-statistic: 4.585 on 3 and 45 DF,  p-value: 0.00695
```

Answer: If we use a significance level of 0.05, the predictors that are significant according to this test are "gleason" and "pgg45" because their p-values are less than 0.05. The results found have been trained on a certain section of data and we are looking at the p-values of that very data. We don't know how this model will work against unseen data, so we can trust that the data we are seeing is accurate for the data we tested it on, but cannot say that we trust that unseen data will behave this same way.

3. (7 points) Using the predictors you selected in part 1, fit a multiple linear regression model on the data in `prostate_fake_test.csv`. Perform a t-test at the $\alpha = 0.05$ significance level for each predictor. Report the predictors that are significant according to this test. Do the results match the results from part 2? Should we trust these results? Why or why not?

```
test_data <- read.csv('prostate_fake_test.csv')
model <- lm(noise ~ lweight + gleason + pgg45, data=test_data)
summary(model)
```

```
##
## Call:
## lm(formula = noise ~ lweight + gleason + pgg45, data = test_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2342 -0.5276 -0.1722  0.6073  1.0393
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.510717   1.551436  -0.974    0.336
## lweight      0.235278   0.174727   1.347    0.185
## gleason      0.095962   0.225074   0.426    0.672
## pgg45       -0.004486   0.004844  -0.926    0.359
##
## Residual standard error: 0.6528 on 44 degrees of freedom
## Multiple R-squared:  0.05677,    Adjusted R-squared:  -0.007542
## F-statistic: 0.8827 on 3 and 44 DF,  p-value: 0.4574
```

Answer: There are no predictors that are significant in this test and the results do not match the results found in part 2. Going along with our explanation in part 2, we are now seeing that the model, in fact, did not respond well with unseen data. We can "trust" that the values we are seeing are correct for the given data it has been tested on, but cannot say these numbers will be true for other subsets of data.