Statistical Learning (5454) - Assignment 1

Matthias Hochholzer, Lukas Pirnbacher, Anne Valder

Due: 2024-03-25

Exercise 1

We use the diabetes data set from “lars” to fit several linear models. First we load and prepare the data and look at the summary statistics.

```
##        y              age                 sex                 bmi
##  Min.   : 25.0   Min.   :-0.107226   Min.   :-0.04464   Min.   :-0.090275
##  1st Qu.: 87.0   1st Qu.:-0.037299   1st Qu.:-0.04464   1st Qu.:-0.034229
##  Median :140.5   Median : 0.005383   Median :-0.04464   Median :-0.007284
##  Mean   :152.1   Mean   : 0.000000   Mean   : 0.00000   Mean   : 0.000000
##  3rd Qu.:211.5   3rd Qu.: 0.038076   3rd Qu.: 0.05068   3rd Qu.: 0.031248
##  Max.   :346.0   Max.   : 0.110727   Max.   : 0.05068   Max.   : 0.170555
##       map                  tc                  ldl
##  Min.   :-0.112400   Min.   :-0.126781   Min.   :-0.115613
##  1st Qu.:-0.036656   1st Qu.:-0.034248   1st Qu.:-0.030358
##  Median :-0.005671   Median :-0.004321   Median :-0.003819
##  Mean   : 0.000000   Mean   : 0.000000   Mean   : 0.000000
##  3rd Qu.: 0.035644   3rd Qu.: 0.028358   3rd Qu.: 0.029844
##  Max.   : 0.132044   Max.   : 0.153914   Max.   : 0.198788
##       hdl                 tch                 ltg
##  Min.   :-0.102307   Min.   :-0.076395   Min.   :-0.126097
##  1st Qu.:-0.035117   1st Qu.:-0.039493   1st Qu.:-0.033249
##  Median :-0.006584   Median :-0.002592   Median :-0.001948
##  Mean   : 0.000000   Mean   : 0.000000   Mean   : 0.000000
##  3rd Qu.: 0.029312   3rd Qu.: 0.034309   3rd Qu.: 0.032433
##  Max.   : 0.181179   Max.   : 0.185234   Max.   : 0.133599
##       glu
##  Min.   :-0.137767
##  1st Qu.:-0.033179
##  Median :-0.001078
##  Mean   : 0.000000
##  3rd Qu.: 0.027917
##  Max.   : 0.135612
```

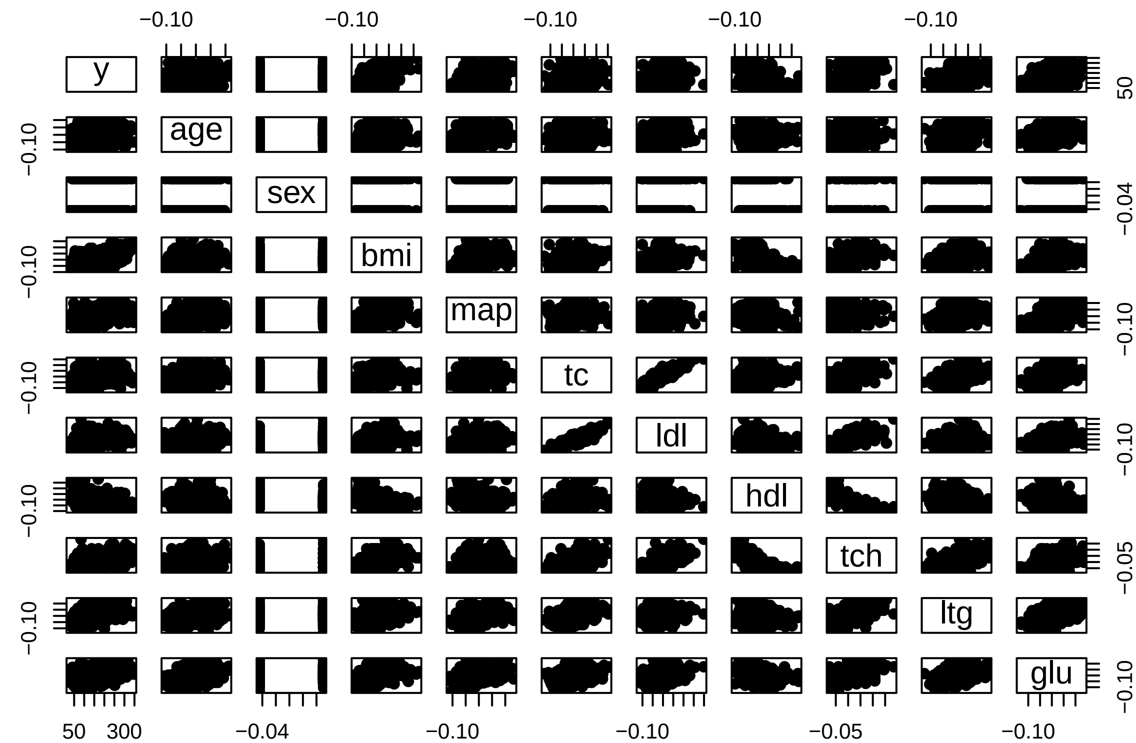
Next, we set a random seed and split the data into train and test data set such that 400 observations (approx. 95%) are used for training and the remaining ones for testing. Selecting the observations for the training set randomly has several reasons. First, we prevent sample bias since the data may be ordered or have patterns based on how the data was collected. If the data has a temporal, spatial, or any systematic order, the first 400 observations might not represent the overall variability in the data set. Second, we mitigate overfitting and improve model robustness. Overall this leads to increased generalizability of our results.

The covariates are only available in standardized form. This can have several implications, both positive and negative. On the one hand, standardization allows us to compare the relative importance of coefficients directly in a regression model, as they are on the same scale. This can be particularly useful in identifying which variables have the most significant effects on the outcome variable. This can in some cases also improve the numerical stability of the estimation process, especially when the variables were measured on vastly different scales. This can lead to more reliable and faster convergence in some algorithms. On the other hand, while standardized coefficients facilitate comparison, they can complicate the interpretation of the model. The coefficients of standardized variables represent the change in the outcome variable for a one-standard-deviation change in the predictor variable, which may not be as intuitive as the original units. Moreover, when transforming back to the usual units, the question is whether effects are captured correctly.

Next, we analyze the pairwise correlation structure between the covariates as well as the covariates and the dependent variable y. These correlations impact model selection as we can get a first impression of whether or not a linear model would be a good assumption through the correlation matrix and the correlation scatter plot. We can see that *sex* is a categorical variable and *tch* seems to be discrete. We observe a clear linear relationship between *tc* and *ldl* with a correlation coefficient of 0.9. Therefore we might ask ourselves if these two variables are really independent predictors. Adding only one to the regression instead of both comes with a slight omitted variable bias, but can make sense for dependent variables in terms of variance reduction. Also the correlation between *tch* and *hdl* lies above 0.7. In general, however, a (strong) linear relationship among the covariates is not clearly observable.

Table 1: Correlation Matrix

	y	age	sex	bmi	map	tc	ldl	hdl	tch	ltg	glu
y	1.00	0.19	0.04	0.59	0.44	0.21	0.17	-0.39	0.43	0.57	0.38
age	0.19	1.00	0.17	0.19	0.34	0.26	0.22	-0.08	0.20	0.27	0.30
sex	0.04	0.17	1.00	0.09	0.24	0.04	0.14	-0.38	0.33	0.15	0.21
bmi	0.59	0.19	0.09	1.00	0.40	0.25	0.26	-0.37	0.41	0.45	0.39
map	0.44	0.34	0.24	0.40	1.00	0.24	0.19	-0.18	0.26	0.39	0.39
tc	0.21	0.26	0.04	0.25	0.24	1.00	0.90	0.05	0.54	0.52	0.33
ldl	0.17	0.22	0.14	0.26	0.19	0.90	1.00	-0.20	0.66	0.32	0.29
hdl	-0.39	-0.08	-0.38	-0.37	-0.18	0.05	-0.20	1.00	-0.74	-0.40	-0.27
tch	0.43	0.20	0.33	0.41	0.26	0.54	0.66	-0.74	1.00	0.62	0.42
ltg	0.57	0.27	0.15	0.45	0.39	0.52	0.32	-0.40	0.62	1.00	0.46
glu	0.38	0.30	0.21	0.39	0.39	0.33	0.29	-0.27	0.42	0.46	1.00



Now, we fit a linear regression model containing all explanatory variables and evaluate its performance using the in-sample mean squared error (MSE) and the out of sample (oos) MSE.

```
##
## Call:
## lm(formula = y ~ ., data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -154.436  -37.748   -1.375   37.421  153.466
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  152.706      2.711  56.319  < 2e-16 ***
## age            9.856     62.721   0.157 0.875213
## sex         -240.347     64.936  -3.701 0.000245 ***
## bmi          499.266     70.415   7.090 6.35e-12 ***
## map          354.976     70.187   5.058 6.55e-07 ***
## tc          -861.163    436.264  -1.974 0.049095 *
## ldl          541.190    354.923   1.525 0.128119
## hdl          116.045    221.425   0.524 0.600518
## tch          166.516    166.601   0.999 0.318178
## ltg          773.896    179.728   4.306 2.11e-05 ***
## glu           63.631     68.817   0.925 0.355729
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 54.18 on 389 degrees of freedom
## Multiple R-squared:  0.5258, Adjusted R-squared:  0.5136
## F-statistic: 43.13 on 10 and 389 DF,  p-value: < 2.2e-16
```

```
##     in_sample_MSE out_of_sample_MSE
##          2854.869          2945.384
```

As expected, the in-sample MSE (2854.869) is lower than the oos MSE on the test data (2945.384).

In the next part, we fit a smaller model where only the covariates are contained, which according to a t-test are significant at the 5% significance level conditional on all other variables being included (see model summary for the full model). This leaves us with the following covariates: *sex*, *bmi*, *map*, *tc* and *ltg*. Again, we evaluate the performance in-sample as well as on the test data.

```
##
## Call:
## lm(formula = y ~ sex + bmi + map + tc + ltg, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -154.487  -39.583   -2.167   36.677  143.460
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  152.676      2.744  55.634  < 2e-16 ***
## sex         -143.624     60.008  -2.393  0.01716 *
## bmi          580.467     67.332   8.621  < 2e-16 ***
## map          344.751     68.041   5.067 6.23e-07 ***
## tc          -218.311     67.313  -3.243  0.00128 **
## ltg          657.293     75.344   8.724  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.85 on 394 degrees of freedom
## Multiple R-squared:  0.5077, Adjusted R-squared:  0.5014
## F-statistic: 81.26 on 5 and 394 DF,  p-value: < 2.2e-16
```

```
##     in_sample_MSE out_of_sample_MSE
##          2963.644          3022.301
```

```
## Analysis of Variance Table
##
## Model 1: y ~ sex + bmi + map + tc + ltg
## Model 2: y ~ age + sex + bmi + map + tc + ldl + hdl + tch + ltg + glu
##   Res.Df     RSS Df Sum of Sq      F  Pr(>F)
## 1    394 1185458
## 2    389 1141947  5     43510 2.9643 0.01221 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The in-sample MSE is now 2963.644 and the oos MSE is 3022.301. Hence, we once again observe a higher out-of-sample MSE. When comparing this smaller model to the full model using an F-test, we see that the full model, which includes more predictors, provides a significantly better fit to the data compared to the small model, as evidenced by the p-value (0.01221) being less than 0.05.

In the following, we use stepwise regression based on the AIC to select a suitable model. We use the *step()* function, which checks whether the AIC decreases when dropping variables in a stepwise procedure and stops as soon as it does not decrease any further. In a similar manner to before, we evaluate the performance in-sample as well as oos on the test data and compare this model to the full model using an F-test.

```
## Start:  AIC=3204.71
## y ~ age + sex + bmi + map + tc + ldl + hdl + tch + ltg + glu
##
##        Df Sum of Sq     RSS    AIC
## - age   1        72 1142020 3202.7
## - hdl   1       806 1142754 3203.0
## - glu   1      2510 1144457 3203.6
## - tch   1      2933 1144880 3203.7
## <none>              1141947 3204.7
## - ldl   1      6825 1148773 3205.1
## - tc    1     11438 1153386 3206.7
## - sex   1     40216 1182164 3216.6
## - ltg   1     54429 1196377 3221.3
## - map   1     75090 1217038 3228.2
## - bmi   1    147581 1289529 3251.3
##
## Step:  AIC=3202.74
## y ~ sex + bmi + map + tc + ldl + hdl + tch + ltg + glu
##
##        Df Sum of Sq     RSS    AIC
## - hdl   1       824 1142844 3201.0
## - glu   1      2656 1144676 3201.7
## - tch   1      2916 1144936 3201.8
## <none>              1142020 3202.7
## - ldl   1      6890 1148910 3203.1
## - tc    1     11478 1153497 3204.7
## - sex   1     40274 1182294 3214.6
## - ltg   1     54900 1196920 3219.5
## - map   1     79224 1221244 3227.6
## - bmi   1    147570 1289590 3249.3
##
## Step:  AIC=3201.03
## y ~ sex + bmi + map + tc + ldl + tch + ltg + glu
##
##        Df Sum of Sq     RSS    AIC
## - tch   1      2185 1145029 3199.8
## - glu   1      2705 1145549 3200.0
## <none>              1142844 3201.0
## - ldl   1      8808 1151653 3202.1
## - tc    1     27555 1170400 3208.6
## - sex   1     40811 1183656 3213.1
## - map   1     78720 1221564 3225.7
## - ltg   1     92523 1235368 3230.2
## - bmi   1    147071 1289915 3247.4
##
## Step:  AIC=3199.79
## y ~ sex + bmi + map + tc + ldl + ltg + glu
##
##        Df Sum of Sq     RSS    AIC
## - glu   1      3071 1148100 3198.9
## <none>              1145029 3199.8
```

```
## - ldl   1     36551 1181580 3210.4
## - sex   1     39159 1184188 3211.2
## - tc    1     61374 1206403 3218.7
## - map   1     76944 1221973 3223.8
## - bmi   1    146794 1291823 3246.0
## - ltg   1    239636 1384665 3273.8
##
## Step:  AIC=3198.86
## y ~ sex + bmi + map + tc + ldl + ltg
##
##        Df Sum of Sq     RSS    AIC
## <none>              1148100 3198.9
## - sex   1     37042 1185142 3209.6
## - ldl   1     37358 1185458 3209.7
## - tc    1     61253 1209352 3217.7
## - map   1     84790 1232890 3225.4
## - bmi   1    158343 1306443 3248.5
## - ltg   1    262231 1410331 3279.1
```

Consequently, stepwise regression based on the AIC yields the following model:

```
##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept)   152.72       2.70   56.48 < 2.2e-16 ***
## sex          -225.95      63.45   -3.56 < 2.2e-16 ***
## bmi           509.71      69.23    7.36 < 2.2e-16 ***
## map           362.15      67.22    5.39 < 2.2e-16 ***
## tc           -775.93     169.46   -4.58 < 2.2e-16 ***
## ldl           554.53     155.07    3.58 < 2.2e-16 ***
## ltg           805.25      84.99    9.47 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##     in_sample_MSE out_of_sample_MSE
##          2870.250          2966.798
```

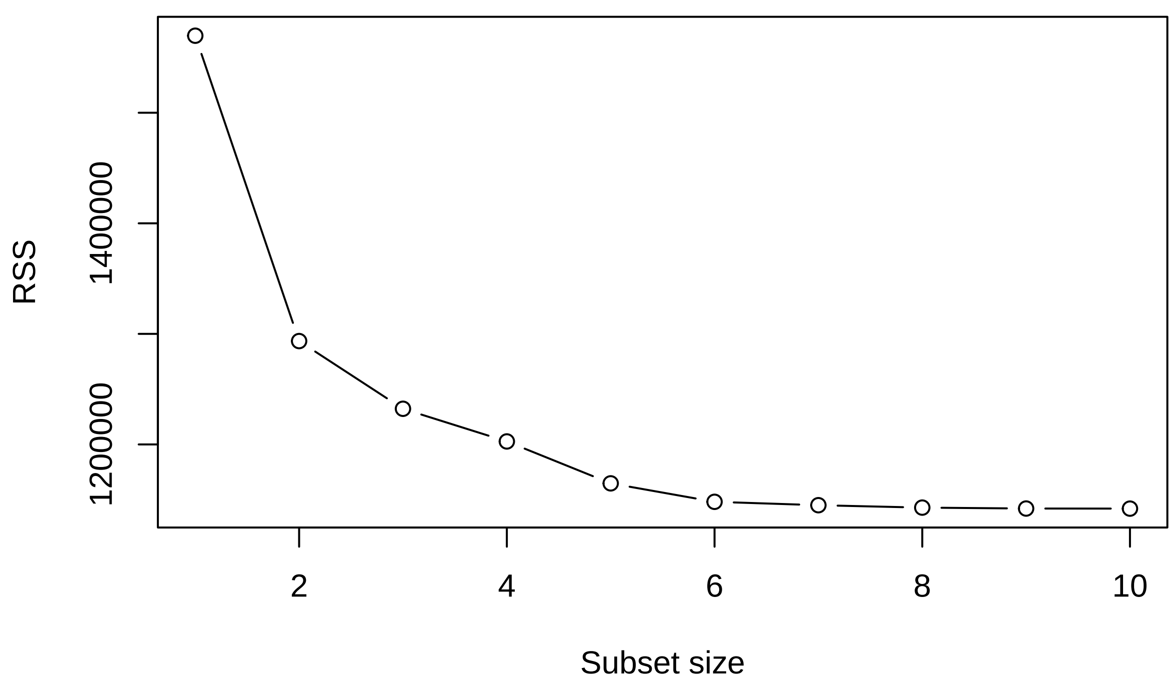
```
## Analysis of Variance Table
##
## Model 1: y ~ age + sex + bmi + map + tc + ldl + hdl + tch + ltg + glu
## Model 2: y ~ sex + bmi + map + tc + ldl + ltg
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1    389 1141947
## 2    393 1148100 -4   -6152.4 0.5239 0.7182
```

The in-sample MSE is now 2870.25 and the oos MSE is 2966.798. The p-value of the F-test (0.7182) is much greater than the typical α-level of 0.05, suggesting there's no significant evidence to favor the full model over the step model, regarding how well they explain the variability in y. In other words, the additional predictors in the full model (*age*, *hdl*, *tch*, and *glu*) do not significantly improve the model's explanatory power compared to the model suggested by stepwise regression based on the AIC.

Next, we use best subset selection to select a suitable model. In order to do so, we make use of the *regsubsets()* function from the **leaps** package.

```
## Subset selection object
## Call: regsubsets.formula(y ~ ., data = train, nvmax = 12, really.big = TRUE)
## 10 Variables  (and intercept)
##     Forced in Forced out
## age     FALSE      FALSE
```

```
## sex     FALSE      FALSE
## bmi     FALSE      FALSE
## map     FALSE      FALSE
## tc      FALSE      FALSE
## ldl     FALSE      FALSE
## hdl     FALSE      FALSE
## tch     FALSE      FALSE
## ltg     FALSE      FALSE
## glu     FALSE      FALSE
## 1 subsets of each size up to 10
## Selection Algorithm: exhaustive
##           age sex bmi map tc  ldl hdl tch ltg glu
## 1  ( 1 )  " " " " "*" " " " " " " " " " " " " " "
## 2  ( 1 )  " " " " "*" " " " " " " " " " " "*" " "
## 3  ( 1 )  " " " " "*" "*" " " " " " " " " "*" " "
## 4  ( 1 )  " " " " "*" "*" "*" " " " " " " "*" " "
## 5  ( 1 )  " " "*" "*" "*" " " " " "*" " " "*" " "
## 6  ( 1 )  " " "*" "*" "*" "*" "*" " " " " "*" " "
## 7  ( 1 )  " " "*" "*" "*" "*" "*" " " " " "*" "*"
## 8  ( 1 )  " " "*" "*" "*" "*" "*" " " "*" "*" "*"
## 9  ( 1 )  " " "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 10  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
```



So far, the algorithm has helped us to determine the best-fitting model for different levels of model complexity, i.e. different numbers of covariates. As can be seen in the plot above, the residual sum of squares (RSS) decreases with increasing model complexity. However, the incremental reduction in RSS from including additional explanatory variables diminishes rather quickly. Hence, we now want to find the optimal model complexity based on the AIC.

```
##            Adj.R2 BIC AIC
## best_model      7   5   6
```

The AIC suggests that model 6 (with covariates sex, bmi, map, tc, ldl, ltg) is the best model, while Adjusted R^2 and BIC recommend a slightly larger and smaller model, respectively.

We now evaluate the performance of the chosen model in-sample as well as on the test data and compare it to the full model using an F-test.

```
##
## Call:
## lm(formula = select_model(best_model$AIC, lm_subset), data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -157.214  -38.027   -2.143   36.163  149.530
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  152.723      2.704  56.477  < 2e-16 ***
## sex         -225.947     63.453  -3.561 0.000415 ***
## bmi          509.713     69.234   7.362 1.07e-12 ***
## map          362.152     67.222   5.387 1.23e-07 ***
## tc          -775.933    169.455  -4.579 6.28e-06 ***
## ldl          554.531    155.071   3.576 0.000392 ***
## ltg          805.250     84.993   9.474  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.05 on 393 degrees of freedom
## Multiple R-squared:  0.5232, Adjusted R-squared:  0.5159
## F-statistic: 71.88 on 6 and 393 DF,  p-value: < 2.2e-16
```

```
##     in_sample_MSE out_of_sample_MSE
##          2870.250          2966.798
```

```
## Analysis of Variance Table
##
## Model 1: y ~ age + sex + bmi + map + tc + ldl + hdl + tch + ltg + glu
## Model 2: y ~ sex + bmi + map + tc + ldl + ltg
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1    389 1141947
## 2    393 1148100 -4   -6152.4 0.5239 0.7182
```

Since best subset selection has led us to the same model choice as the stepwise regression, in-sample, out-of-sample performance as well as the F-test against the full model yield the same conclusions as above.

Last, we summarize our results in the following table, containing the regression coefficients of the different models as well as the in-sample and the test data performance:

Table 2: Results for all models

	full	small	stepwise	subset
X.Intercept.	152.71	152.68	152.72	152.72
age	9.86	NA	NA	NA
sex	-240.35	-143.62	-225.95	-225.95
bmi	499.27	580.47	509.71	509.71
map	354.98	344.75	362.15	362.15
tc	-861.16	-218.31	-775.93	-775.93
ldl	541.19	NA	554.53	554.53
hdl	116.05	NA	NA	NA
tch	166.52	NA	NA	NA
ltg	773.90	657.29	805.25	805.25
glu	63.63	NA	NA	NA
MSE in-sample	2854.87	2963.64	2870.25	2870.25
MSE out-of-sample	2945.38	3022.30	2966.80	2966.80

While it is clear that the full model has the best in-sample performance in terms of MSE, it is somewhat surprising that the full model outperforms our alternative candidates on the test data as well.

Exercise 2

We use the wage data set to fit different linear models. The data set is available in the R package **ISLR2**. First, we load and prepare the data and have a look at the summary statistics.

Next, we fit a linear regression model to predict wage using age and education as predictors. We specify non-linear effects for the variable age by including a polynomial of degree 10. Moreover, we choose suitable contrasts for the variable education to compare the different levels of education in a meaningful way. Contrasts define how categorical variables are encoded into numerical values for the analysis. The default encoding in R is treatment coding (also known as dummy coding), where one level is chosen as a baseline and the other levels are compared to this baseline. For an ordinal variable like education, where the levels have a natural order, polynomial contrasts might be more appropriate as they can model the linear and non-linear relationships between the levels of education and the outcome variable.

```
##       year           age                       maritl           race
##  Min.   :2003   Min.   :18.00   1. Never Married: 648   1. White:2480
##  1st Qu.:2004   1st Qu.:33.75   2. Married      :2074   2. Black: 293
##  Median :2006   Median :42.00   3. Widowed      :  19   3. Asian: 190
##  Mean   :2006   Mean   :42.41   4. Divorced     : 204   4. Other:  37
##  3rd Qu.:2008   3rd Qu.:51.00   5. Separated    :  55
##  Max.   :2009   Max.   :80.00
##
##               education                     region               jobclass
##  1. < HS Grad      :268   2. Middle Atlantic   :3000   1. Industrial :1544
##  2. HS Grad        :971   1. New England       :   0   2. Information:1456
##  3. Some College   :650   3. East North Central:   0
##  4. College Grad   :685   4. West North Central:   0
##  5. Advanced Degree:426   5. South Atlantic    :   0
##                           6. East South Central:   0
##                           (Other)              :   0
```

```
##             health      health_ins      logwage           wage
##  1. <=Good     : 858   1. Yes:2083   Min.   :3.000   Min.   : 20.09
##  2. >=Very Good:2142   2. No : 917   1st Qu.:4.447   1st Qu.: 85.38
##                                      Median :4.653   Median :104.92
##                                      Mean   :4.654   Mean   :111.70
##                                      3rd Qu.:4.857   3rd Qu.:128.68
##                                      Max.   :5.763   Max.   :318.34
##
```

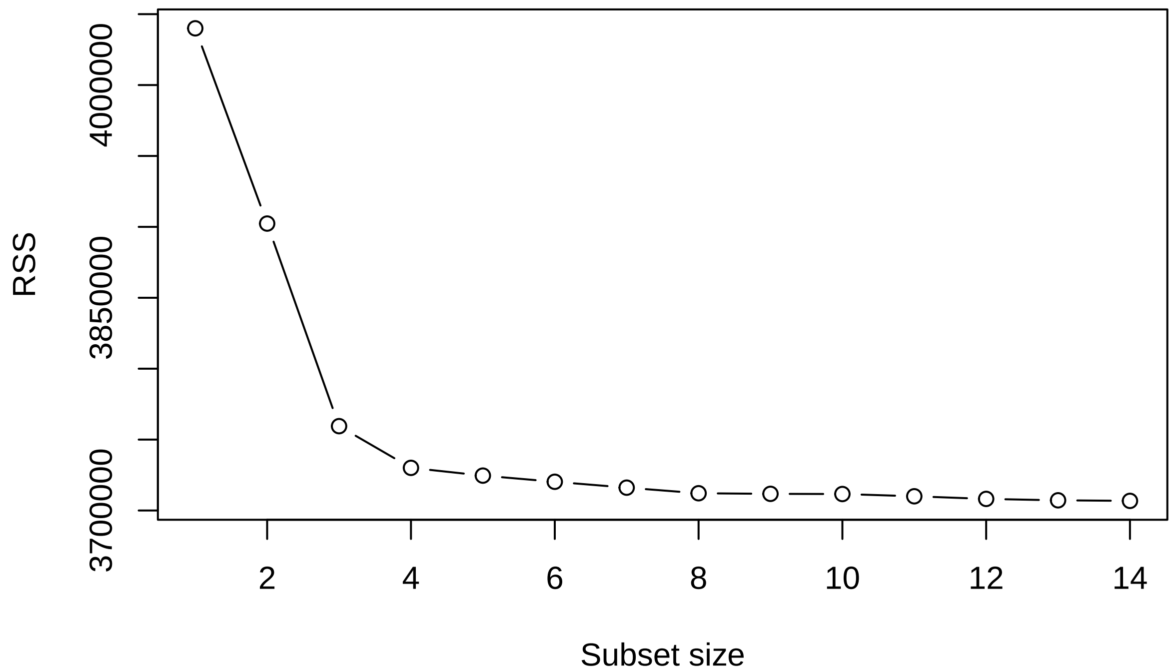
```
##       1. < HS Grad         2. HS Grad    3. Some College    4. College Grad
##                268                971                650                685
## 5. Advanced Degree
##                426
```

```
##
## Call:
## lm(formula = formula_full, data = Wage)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -117.349  -19.757   -3.136   14.504  215.200
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.138e+04  2.336e+04   1.344   0.1792
## age         -7.948e+03  6.043e+03  -1.315   0.1885
## age2         8.784e+02  6.844e+02   1.283   0.1994
## age3        -5.571e+01  4.472e+01  -1.246   0.2130
## age4         2.247e+00  1.868e+00   1.203   0.2291
## age5        -6.034e-02  5.221e-02  -1.156   0.2479
## age6         1.093e-03  9.890e-04   1.105   0.2691
## age7        -1.322e-05  1.256e-05  -1.053   0.2925
## age8         1.023e-07  1.023e-07   0.999   0.3177
## age9        -4.580e-10  4.842e-10  -0.946   0.3442
## age10        9.030e-13  1.010e-12   0.894   0.3716
## education.L  4.827e+01  1.845e+00  26.159  < 2e-16 ***
## education.Q  7.721e+00  1.721e+00   4.485 7.55e-06 ***
## education.C  2.695e+00  1.416e+00   1.904   0.0571 .
## education^4  7.541e-01  1.343e+00   0.562   0.5744
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.24 on 2985 degrees of freedom
## Multiple R-squared:  0.2902, Adjusted R-squared:  0.2868
## F-statistic: 87.16 on 14 and 2985 DF,  p-value: < 2.2e-16
```

We can see from the summaries of our full model that education has significant explanatory power with respect to wages and that wages ceteris paribus increase with higher levels of education. At the same time, none of the coefficients corresponding to the polynomial of *age* are significantly different from zero conditional on all other variables being in the model. This is not necessarily a sign that *age* cannot predict wages, but rather a sign of overfitting and some collinearity issues related to including a polynomial of such a high degree. Consequently, we now perform best subset selection to determine a suitable model. To do this, we once again use the **leaps** package in R.

```
## Subset selection object
## Call: regsubsets.formula(formula_full, data = Wage, nvmax = 20, really.big = TRUE)
```

```
## 14 Variables  (and intercept)
##              Forced in Forced out
## age              FALSE      FALSE
## age2             FALSE      FALSE
## age3             FALSE      FALSE
## age4             FALSE      FALSE
## age5             FALSE      FALSE
## age6             FALSE      FALSE
## age7             FALSE      FALSE
## age8             FALSE      FALSE
## age9             FALSE      FALSE
## age10            FALSE      FALSE
## education.L      FALSE      FALSE
## education.Q      FALSE      FALSE
## education.C      FALSE      FALSE
## education^4      FALSE      FALSE
## 1 subsets of each size up to 14
## Selection Algorithm: exhaustive
##           age age2 age3 age4 age5 age6 age7 age8 age9 age10 education.L
## 1  ( 1 )  " " " "  " "  " "  " "  " "  " "  " "  " "  " "   "*"
## 2  ( 1 )  "*" " "  " "  " "  " "  " "  " "  " "  " "  " "   "*"
## 3  ( 1 )  "*" "*"  " "  " "  " "  " "  " "  " "  " "  " "   "*"
## 4  ( 1 )  "*" "*"  " "  " "  " "  " "  " "  " "  " "  " "   "*"
## 5  ( 1 )  "*" "*"  "*"  " "  " "  " "  " "  " "  " "  " "   "*"
## 6  ( 1 )  "*" "*"  "*"  " "  " "  " "  " "  " "  " "  " "   "*"
## 7  ( 1 )  " " " "  " "  " "  "*"  "*"  "*"  "*"  "*"  " "   "*"
## 8  ( 1 )  " " " "  " "  " "  "*"  "*"  "*"  "*"  "*"  " "   "*"
## 9  ( 1 )  " " " "  " "  " "  "*"  "*"  "*"  "*"  "*"  " "   "*"
## 10  ( 1 ) " " " "  " "  " "  "*"  "*"  "*"  "*"  "*"  "*"   "*"
## 11  ( 1 ) " " " "  "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"   "*"
## 12  ( 1 ) "*" "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"  " "   "*"
## 13  ( 1 ) "*" "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"   "*"
## 14  ( 1 ) "*" "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"   "*"
##           education.Q education.C education^4
## 1  ( 1 )  " "         " "         " "
## 2  ( 1 )  " "         " "         " "
## 3  ( 1 )  " "         " "         " "
## 4  ( 1 )  "*"         " "         " "
## 5  ( 1 )  "*"         " "         " "
## 6  ( 1 )  "*"         "*"         " "
## 7  ( 1 )  "*"         " "         " "
## 8  ( 1 )  "*"         "*"         " "
## 9  ( 1 )  "*"         "*"         "*"
## 10  ( 1 ) "*"         "*"         "*"
## 11  ( 1 ) "*"         "*"         " "
## 12  ( 1 ) "*"         "*"         " "
## 13  ( 1 ) "*"         "*"         " "
## 14  ( 1 ) "*"         "*"         "*"
```



Just as in Exercise 1, we obtain the best-fitting model for different levels of complexity. Next, we want to select the model that minimizes the AIC.

```
##            Adj.R2 BIC AIC
## best_model      8   4   8
```

Based on the AIC, we conclude that model 8 (with covariates age5, age6, age7, age8, age9, education.L, education.Q, education.C) is the best model. In contrast, the BIC would suggest model 4 (with covariates age, age2, education.L, education.Q). Let us now estimate and have a closer look at the chosen model.

```
##
## Call:
## lm(formula = wage ~ ., data = temp_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -116.022  -19.924   -3.236   14.306  214.816
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.154e+01  3.488e+00  20.511  < 2e-16 ***
## age5         1.240e-05  1.817e-06   6.828 1.04e-11 ***
## age6        -7.265e-07  1.165e-07  -6.234 5.18e-10 ***
## age7         1.618e-08  2.812e-09   5.756 9.48e-09 ***
## age8        -1.614e-10  3.008e-11  -5.365 8.73e-08 ***
## age9         6.046e-13  1.200e-13   5.038 4.99e-07 ***
## education.L  4.832e+01  1.830e+00  26.403  < 2e-16 ***
## education.Q  7.779e+00  1.712e+00   4.544 5.74e-06 ***
```

```
## education.C  2.518e+00  1.410e+00   1.785   0.0743 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.23 on 2991 degrees of freedom
## Multiple R-squared:  0.2891, Adjusted R-squared:  0.2872
## F-statistic: 152.1 on 8 and 2991 DF,  p-value: < 2.2e-16
```

Since treatment contrasts are generally easier to interpret, especially in the context of model selection, we repeat the procedure above with treatment instead of polynomial contrasts. Let us first estimate the full model.

```
##
## Call:
## lm(formula = formula_full, data = Wage)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -117.349  -19.757   -3.136   14.504  215.200
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  3.135e+04  2.336e+04   1.342    0.180
## age                         -7.948e+03  6.043e+03  -1.315    0.189
## age2                         8.784e+02  6.844e+02   1.283    0.199
## age3                        -5.571e+01  4.472e+01  -1.246    0.213
## age4                         2.247e+00  1.868e+00   1.203    0.229
## age5                        -6.034e-02  5.221e-02  -1.156    0.248
## age6                         1.093e-03  9.890e-04   1.105    0.269
## age7                        -1.322e-05  1.256e-05  -1.053    0.292
## age8                         1.023e-07  1.023e-07   0.999    0.318
## age9                        -4.580e-10  4.842e-10  -0.946    0.344
## age10                        9.030e-13  1.010e-12   0.894    0.372
## education2. HS Grad          1.118e+01  2.444e+00   4.574 4.98e-06 ***
## education3. Some College     2.358e+01  2.574e+00   9.158  < 2e-16 ***
## education4. College Grad     3.830e+01  2.557e+00  14.976  < 2e-16 ***
## education5. Advanced Degree  6.276e+01  2.773e+00  22.631  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.24 on 2985 degrees of freedom
## Multiple R-squared:  0.2902, Adjusted R-squared:  0.2868
## F-statistic: 87.16 on 14 and 2985 DF,  p-value: < 2.2e-16
```

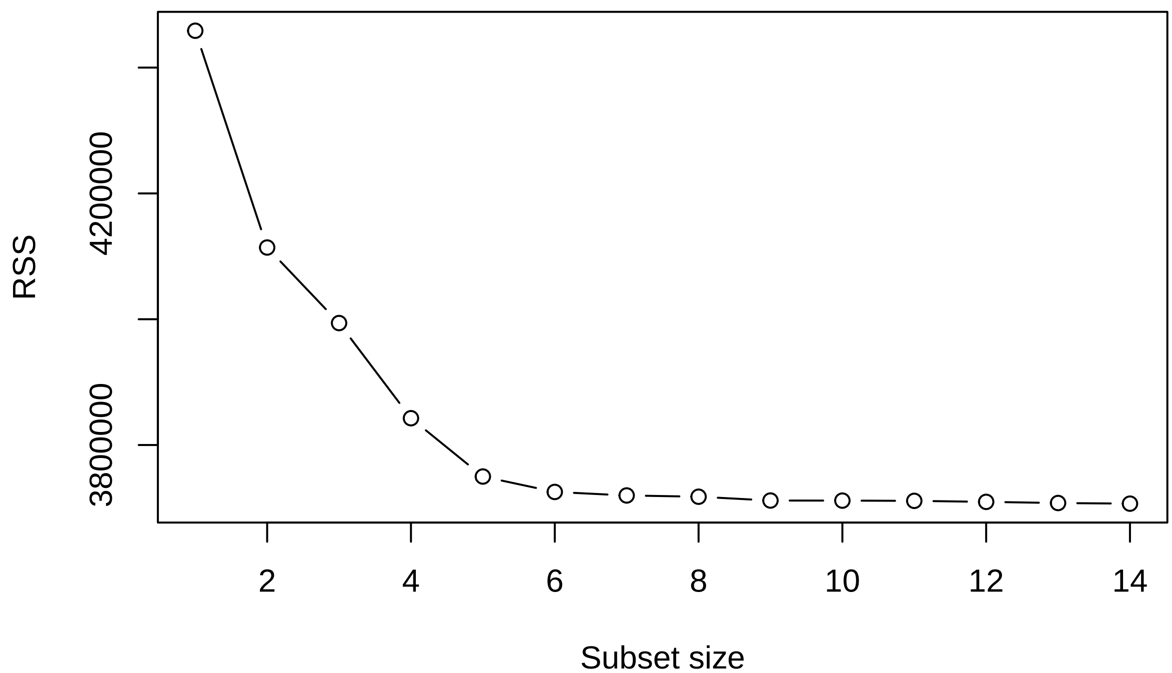
While residuals and fitted values of the full model are independent of our contrast choice, the interpretation of the estimated coefficients is different now. In particular, the estimated *education* coefficients now capture the mean difference in earnings of the respective education level to the baseline of $<$ *HS Grad*.

We continue with the best subset selection.

```
## Subset selection object
## Call: regsubsets.formula(formula_full, data = Wage, nvmax = 20, really.big = TRUE)
## 14 Variables  (and intercept)
##                             Forced in Forced out
## age                             FALSE      FALSE
## age2                            FALSE      FALSE
```

```
## age3                            FALSE      FALSE
## age4                            FALSE      FALSE
## age5                            FALSE      FALSE
## age6                            FALSE      FALSE
## age7                            FALSE      FALSE
## age8                            FALSE      FALSE
## age9                            FALSE      FALSE
## age10                           FALSE      FALSE
## education2. HS Grad             FALSE      FALSE
## education3. Some College        FALSE      FALSE
## education4. College Grad        FALSE      FALSE
## education5. Advanced Degree     FALSE      FALSE
## 1 subsets of each size up to 14
## Selection Algorithm: exhaustive
##           age age2 age3 age4 age5 age6 age7 age8 age9 age10 education2. HS Grad
## 1  ( 1 )  " " " "  " "  " "  " "  " "  " "  " "  " "  " "   " "                
## 2  ( 1 )  " " " "  " "  " "  " "  " "  " "  " "  " "  " "   " "                
## 3  ( 1 )  "*" " "  " "  " "  " "  " "  " "  " "  " "  " "   " "                
## 4  ( 1 )  "*" "*"  " "  " "  " "  " "  " "  " "  " "  " "   " "                
## 5  ( 1 )  "*" "*"  " "  " "  " "  " "  " "  " "  " "  " "   " "                
## 6  ( 1 )  "*" "*"  " "  " "  " "  " "  " "  " "  " "  " "   "*"                
## 7  ( 1 )  "*" "*"  "*"  " "  " "  " "  " "  " "  " "  " "   "*"                
## 8  ( 1 )  " " "*"  " "  "*"  "*"  "*"  " "  " "  " "  " "   "*"                
## 9  ( 1 )  " " " "  " "  " "  "*"  "*"  "*"  "*"  "*"  " "   "*"                
## 10  ( 1 ) " " " "  " "  " "  "*"  "*"  "*"  "*"  "*"  "*"   "*"                
## 11  ( 1 ) "*" "*"  "*"  "*"  "*"  "*"  "*"  " "  " "  " "   "*"                
## 12  ( 1 ) " " " "  "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"   "*"                
## 13  ( 1 ) "*" "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"  " "   "*"                
## 14  ( 1 ) "*" "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"   "*"                
##           education3. Some College education4. College Grad
## 1  ( 1 )  " "                      " "                     
## 2  ( 1 )  " "                      "*"                     
## 3  ( 1 )  " "                      "*"                     
## 4  ( 1 )  " "                      "*"                     
## 5  ( 1 )  "*"                      "*"                     
## 6  ( 1 )  "*"                      "*"                     
## 7  ( 1 )  "*"                      "*"                     
## 8  ( 1 )  "*"                      "*"                     
## 9  ( 1 )  "*"                      "*"                     
## 10  ( 1 ) "*"                      "*"                     
## 11  ( 1 ) "*"                      "*"                     
## 12  ( 1 ) "*"                      "*"                     
## 13  ( 1 ) "*"                      "*"                     
## 14  ( 1 ) "*"                      "*"                     
##           education5. Advanced Degree
## 1  ( 1 )  "*"                        
## 2  ( 1 )  "*"                        
## 3  ( 1 )  "*"                        
## 4  ( 1 )  "*"                        
## 5  ( 1 )  "*"                        
## 6  ( 1 )  "*"                        
## 7  ( 1 )  "*"                        
## 8  ( 1 )  "*"                        
## 9  ( 1 )  "*"                        
```

```
## 10  ( 1 ) "*"
## 11  ( 1 ) "*"
## 12  ( 1 ) "*"
## 13  ( 1 ) "*"
## 14  ( 1 ) "*"
```




```
##            Adj.R2 BIC AIC
## best_model      9   6   9
```

In the model selection context we start to see some differences between polynomial and treatment contrasts. With dummy coding, the best model according to the AIC is model 9, which includes all dummy variables created from *education* and the higher-order terms of *age*, i.e. $age^5, age^6, age^7, age^8$ and age^9. Let's have a look at that model:

```
##
## Call:
## lm(formula = wage ~ ., data = temp_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -116.171  -19.942   -3.328   14.470  215.061
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    4.456e+01  3.962e+00  11.248  < 2e-16 ***
## age5                           1.239e-05  1.817e-06   6.819 1.10e-11 ***
## age6                          -7.256e-07  1.166e-07  -6.225 5.48e-10 ***
## age7                           1.616e-08  2.812e-09   5.746 1.00e-08 ***
```

```
## age8                           -1.611e-10  3.009e-11  -5.354 9.23e-08 ***
## age9                            6.034e-13  1.200e-13   5.027 5.27e-07 ***
## `education2. HS Grad`           1.093e+01  2.435e+00   4.490 7.39e-06 ***
## `education3. Some College`      2.339e+01  2.563e+00   9.126  < 2e-16 ***
## `education4. College Grad`      3.820e+01  2.545e+00  15.009  < 2e-16 ***
## `education5. Advanced Degree`  6.262e+01  2.764e+00  22.658  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.23 on 2990 degrees of freedom
## Multiple R-squared:  0.2892, Adjusted R-squared:  0.2871
## F-statistic: 135.2 on 9 and 2990 DF,  p-value: < 2.2e-16
```

If we now compare the AIC of the two selected models for polynomial and treatment contrasts, we find that the former is slightly superior.

```
##  AIC_poly_contr AIC_treat_contr
##        29895.90        29897.57
```

Finally, we assess if it makes a difference if we include the polynomial of the original variable *age* or orthogonal polynomials constructed using poly(age, k). Direct polynomials are straightforward, making them somewhat easier to interpret in terms of the direct effect of aging. However, they can be collinear, especially with higher-degree polynomials. Orthogonal polynomials deal with the potential issue of multicollinearity between the polynomial terms, leading to more stable coefficient estimates. However, the coefficients of orthogonal polynomials are a bit more challenging to interpret. For predictive accuracy, orthogonal polynomials can sometimes offer an advantage, especially in complex models. For interpretation, direct polynomials might be preferred if the primary interest is in understanding the specific nature of the relationship between age and wage.

In the full model, both the orthogonal polynomial and the direct polynomial yield the same model fit, since each design matrix spans the same space. In particular, AIC and BIC values are the same for both models using direct polynomial terms and orthogonal polynomials for age.

	direct	orthogonal
AIC	29903.56	29903.56
BIC	29999.66	29999.66

Let us now investigate whether there are any differences between orthogonal and direct polynomials when it comes to model selection.

```
## Subset selection object
## Call: regsubsets.formula(formula_full_ortho, data = Wage_ortho, nvmax = 20,
##     really.big = TRUE)
## 14 Variables  (and intercept)
##             Forced in Forced out
## age             FALSE      FALSE
## age2            FALSE      FALSE
## age3            FALSE      FALSE
## age4            FALSE      FALSE
## age5            FALSE      FALSE
## age6            FALSE      FALSE
## age7            FALSE      FALSE
## age8            FALSE      FALSE
## age9            FALSE      FALSE
## age10           FALSE      FALSE
```

```
## education.L     FALSE      FALSE
## education.Q     FALSE      FALSE
## education.C     FALSE      FALSE
## education^4     FALSE      FALSE
## 1 subsets of each size up to 14
## Selection Algorithm: exhaustive
##           age age2 age3 age4 age5 age6 age7 age8 age9 age10 education.L
## 1  ( 1 )  " " " "  " "  " "  " "  " "  " "  " "  " "  " "   "*"
## 2  ( 1 )  " " "*"  " "  " "  " "  " "  " "  " "  " "  " "   "*"
## 3  ( 1 )  "*" "*"  " "  " "  " "  " "  " "  " "  " "  " "   "*"
## 4  ( 1 )  "*" "*"  " "  " "  " "  " "  " "  " "  " "  " "   "*"
## 5  ( 1 )  "*" "*"  "*"  " "  " "  " "  " "  " "  " "  " "   "*"
## 6  ( 1 )  "*" "*"  "*"  " "  " "  " "  " "  " "  " "  " "   "*"
## 7  ( 1 )  "*" "*"  "*"  " "  " "  " "  "*"  " "  " "  " "   "*"
## 8  ( 1 )  "*" "*"  "*"  " "  " "  " "  "*"  " "  "*"  " "   "*"
## 9  ( 1 )  "*" "*"  "*"  " "  "*"  " "  "*"  " "  "*"  " "   "*"
## 10  ( 1 ) "*" "*"  "*"  " "  "*"  "*"  "*"  " "  "*"  " "   "*"
## 11  ( 1 ) "*" "*"  "*"  " "  "*"  "*"  "*"  " "  "*"  "*"   "*"
## 12  ( 1 ) "*" "*"  "*"  " "  "*"  "*"  "*"  " "  "*"  "*"   "*"
## 13  ( 1 ) "*" "*"  "*"  "*"  "*"  "*"  "*"  " "  "*"  "*"   "*"
## 14  ( 1 ) "*" "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"  "*"   "*"
##           education.Q education.C education^4
## 1  ( 1 )  " "         " "         " "
## 2  ( 1 )  " "         " "         " "
## 3  ( 1 )  " "         " "         " "
## 4  ( 1 )  "*"         " "         " "
## 5  ( 1 )  "*"         " "         " "
## 6  ( 1 )  "*"         "*"         " "
## 7  ( 1 )  "*"         "*"         " "
## 8  ( 1 )  "*"         "*"         " "
## 9  ( 1 )  "*"         "*"         " "
## 10  ( 1 ) "*"         "*"         " "
## 11  ( 1 ) "*"         "*"         " "
## 12  ( 1 ) "*"         "*"         "*"
## 13  ( 1 ) "*"         "*"         "*"
## 14  ( 1 ) "*"         "*"         "*"
```

```
##            Adj.R2 BIC AIC
## best_model     10   4   9
```

```
##
## Call:
## lm(formula = wage ~ ., data = temp_data_ortho)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -117.020  -19.823   -3.165   14.256  213.840
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  112.4313     0.6988 160.889  < 2e-16 ***
## age          362.0337    35.4051  10.225  < 2e-16 ***
## age2        -379.4059    35.3918 -10.720  < 2e-16 ***
## age3          75.0084    35.2750   2.126   0.0336 *
## age5         -53.2595    35.3370  -1.507   0.1319
```

```
## age7          61.7630    35.2541   1.752   0.0799 .
## age9         -58.7896    35.2668  -1.667   0.0956 .
## education.L   48.3330     1.8306  26.402  < 2e-16 ***
## education.Q    7.7588     1.7157   4.522 6.36e-06 ***
## education.C    2.6415     1.4107   1.872   0.0612 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.23 on 2990 degrees of freedom
## Multiple R-squared:  0.2895, Adjusted R-squared:  0.2874
## F-statistic: 135.4 on 9 and 2990 DF,  p-value: < 2.2e-16
```

Consequently, we find that the use of orthogonal polynomials leads to different conclusions when it comes to model selection. In particular, the model selected based on the AIC is now more complex and also includes the lower-order terms of *age*. The AIC of the best model with orthogonal polynomials is slightly larger than that of the best model with the direct polynomial.

```
##     AIC_direct AIC_orthogonal
##       29895.90       29896.29
```

Exercise 3

We assume the following data generating process:

$$y = f(x) + \epsilon = x + x^2 + \epsilon,$$

where $\epsilon \sim N(0, \sigma_\epsilon^2)$, $x \sim N(0, \sigma_x^2)$ and x and ϵ are independent. First, we analytically determine the test error using the squared error loss for given parameter estimates $\hat{\beta} = \left(\hat{\beta}_1, \hat{\beta}_2\right)'$.

Given a training set $\mathcal{T}$, the test error (also called generalization error) of the model $\hat{f}$ is given by

$$Err_{\mathcal{T}} = \mathbb{E}_{x,y}[L(y, \hat{f}(x)) \,|\, \mathcal{T}],$$

where $\hat{f}(x) = \hat{\beta}_1 x + \hat{\beta}_2 x^2$ and $L(y, \hat{f}(x)) = (y - \hat{f}(x))^2$ denotes the squared error loss function. It follows that

$$\begin{aligned}
Err_{\mathcal{T}} &= \mathbb{E}_{x,y}\left[\left(x + x^2 + \epsilon - \hat{\beta}_1 x - \hat{\beta}_2 x^2\right)^2 \,\middle|\, \mathcal{T}\right] \\
&= \mathbb{E}_{x,y}\Big[\big(\underbrace{(1-\hat{\beta}_1)x + (1-\hat{\beta}_2)x^2}_{=:\mathrm{red}(x)} + \epsilon\big)^2 \,\Big|\, \mathcal{T}\Big] \\
&= \mathbb{E}_{x,y}\left[\mathrm{red}(x)^2 + \epsilon^2 + 2\mathrm{red}(x)\epsilon \,|\, \mathcal{T}\right] \\
&= \mathbb{E}_{x,y}\left[(1-\hat{\beta}_1)^2 x^2 + (1-\hat{\beta}_2)^2 x^4 + 2(1-\hat{\beta}_1)(1-\hat{\beta}_2)x^3 \,|\, \mathcal{T}\right] + \mathbb{E}_{x,y}\left[\epsilon^2 \,|\, \mathcal{T}\right] + 2\mathbb{E}_{x,y}\left[\mathrm{red}(x)\epsilon \,|\, \mathcal{T}\right] \\
&= (1-\hat{\beta}_1)^2 \mathbb{E}_{x,y}\left[x^2 \,|\, \mathcal{T}\right] + (1-\hat{\beta}_2)^2 \mathbb{E}_{x,y}\left[x^4 \,|\, \mathcal{T}\right] + 2(1-\hat{\beta}_1)(1-\hat{\beta}_2)\mathbb{E}_{x,y}\left[x^3 \,|\, \mathcal{T}\right] + \sigma_\epsilon^2 \\
&= (1-\hat{\beta}_1)^2 \sigma_x^2 + (1-\hat{\beta}_2)^2 3\sigma_x^4 + \sigma_\epsilon^2,
\end{aligned}$$

where red(x) is the reducible error. In the last step we used the 2^{nd}, 3^{rd} and 4^{th} moment of the Normal distribution. In the step before, we used the fact, that x and ϵ are independent and that $\mathbb{E}[\epsilon] = 0$.

Next, we draw a sample of size $N = 40$ as training data (assuming $\sigma_\epsilon^2 = \sigma_x^2 = 1$) and estimate the regression coefficients using OLS. We then determine the test error using the analytical formula as well as simulations. For the simulations we generate a test sample of size $N_{test} = 10,000$, in order for the mean of the squared prediction errors to be a reasonable approximation of the test error.

Ultimately, we find that simulated and analytical test errors are quite similar and given by

```
## test_error_analytical  test_error_simulated
##              1.049210              1.050748
```

Finally, we want to determine the expected test error, which is defined as

$$Err = \mathbb{E}_{x,y}\left[L(y, \hat{f}(x))\right] = \mathbb{E}_{\mathcal{T}}\left[Err_{\mathcal{T}}\right].$$

We estimate the expected test error as the mean test error across $N_{\mathcal{T}} = 1000$ different training samples of size $N = 40$ and find that $Err = 1.0636976$. Consequently, the expected test error is similar to the test error we obtained above. However, a closer look at the summary statistics of the test errors computed for different sets of training data shows that there is in fact some variation in $Err_{\mathcal{T}}$, depending on the particular characteristics of the respective training data $\mathcal{T}$.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.015   1.037   1.064   1.078   1.881
```

Exercise 4

We consider the data generating process

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

where $\boldsymbol{\epsilon} \sim N(\boldsymbol{0}, \sigma_\epsilon^2 \boldsymbol{I})$ and $\boldsymbol{X} \in \mathbb{R}^{N\times p}$ is a fixed covariate matrix. First, we want to derive the in-sample error for given parameter estimates $\hat{\boldsymbol{\beta}}$ using the squared error loss.

Let $\boldsymbol{x}_i$ denote the i-th row of $\boldsymbol{X}$, i.e. the covariates of observation $i = 1, \ldots, N$. Similarly, let y_i and ϵ_i denote the i-th entry of $\boldsymbol{y}$ and $\boldsymbol{\epsilon}$, respectively. The in-sample error for given training data $\mathcal{T}$ can then be defined as

$$Err_{in} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{y_i^0}\left[\left(y_i^0 - \hat{f}(x_i)\right)^2 \,\Big|\, \mathcal{T}\right],$$

where $y_i^0 = f(x_i) + \epsilon_i^0 = \boldsymbol{x}_i\boldsymbol{\beta} + \epsilon_i^0$ is a new response for observation $i = 1, \ldots, N$ and $\hat{f}(x_i) = \boldsymbol{x}_i\hat{\boldsymbol{\beta}}$. For arbitrary $i = 1, \ldots, N$ it now follows that

$$\begin{aligned}
\mathbb{E}_{y_i^0}\left[\left(y_i^0 - \hat{f}(x_i)\right)^2 \,\Big|\, \mathcal{T}\right] &= \mathbb{E}_{y_i^0}\left[\left(\boldsymbol{x}_i\boldsymbol{\beta} + \epsilon_i^0 - x_i\hat{\boldsymbol{\beta}})\right)^2 \,\Big|\, \mathcal{T}\right] \\
&= \mathbb{E}_{y_i^0}\left[\left(\boldsymbol{x}_i(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})\right)^2 + 2\epsilon_i^0\boldsymbol{x}_i(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}) + \left(\epsilon_i^0\right)^2 \,\Big|\, \mathcal{T}\right] \\
&= \left(\boldsymbol{x}_i(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})\right)^2 + 2\boldsymbol{x}_i(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})\underbrace{\mathbb{E}\left[\epsilon_i^0\right]}_{=0} + \underbrace{\mathbb{E}\left[(\epsilon_i^0)^2\right]}_{=\sigma_\epsilon^2} \\
&= \sigma_\epsilon^2 + \left(\boldsymbol{x}_i(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})\right)^2.
\end{aligned}$$

The in-sample error can therefore be written as

$$\begin{aligned}
Err_{in} &= \frac{1}{N}\sum_{i=1}^{N} \sigma_\epsilon^2 + \left(\boldsymbol{x}_i(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})\right)^2 \\
&= \sigma_\epsilon^2 + \frac{1}{N}\left(\boldsymbol{X}(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})\right)'\left(\boldsymbol{X}(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})\right) \\
&= \sigma_\epsilon^2 + \frac{1}{N}(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})'\boldsymbol{X}'\boldsymbol{X}(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}).
\end{aligned}$$

Next, we want to determine the expected in-sample error for OLS estimates of the regression coefficients. The expected in-sample error can be obtained by averaging the in-sample error over the distribution of training data $\mathcal{T}$. Hence, we are interested in $\mathbb{E}_{\mathcal{T}}\left[Err_{in}\right]$. Since the design matrix $\boldsymbol{X}$ is deterministic in this example, the only source of randomness in our training data are the error terms $\boldsymbol{\epsilon}^{\mathcal{T}} = \left(\epsilon_1^{\mathcal{T}}, \ldots, \epsilon_N^{\mathcal{T}}\right)'$.

Let us first consider the in-sample error for OLS estimates and fixed training data $\mathcal{T}$. The OLS estimates can be expressed as

$$
\begin{aligned}
\hat{\boldsymbol{\beta}} &= \left(\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\boldsymbol{y}^{\mathcal{T}} \\
&= \left(\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\left(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}^{\mathcal{T}}\right) \\
&= \left(\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\boldsymbol{X}'\boldsymbol{X}\boldsymbol{\beta} + \underbrace{\left(\boldsymbol{X}'\boldsymbol{X}\right)^{-1}\boldsymbol{X}'}_{=:\boldsymbol{X}^{\dagger}}\boldsymbol{\epsilon}^{\mathcal{T}} \\
&= \boldsymbol{\beta} + \boldsymbol{X}^{\dagger}\boldsymbol{\epsilon}^{\mathcal{T}}
\end{aligned}
$$

Hence, the in-sample error is given by

$$
\begin{aligned}
Err_{in} &= \sigma_\epsilon^2 + \frac{1}{N}\left(-\boldsymbol{X}^{\dagger}\boldsymbol{\epsilon}^{\mathcal{T}}\right)'\boldsymbol{X}'\boldsymbol{X}\left(-\boldsymbol{X}^{\dagger}\boldsymbol{\epsilon}^{\mathcal{T}}\right) \\
&= \sigma_\epsilon^2 + \frac{1}{N}\left(\boldsymbol{\epsilon}^{\mathcal{T}}\right)'\left(\boldsymbol{X}^{\dagger}\right)'\boldsymbol{X}'\boldsymbol{X}\boldsymbol{X}^{\dagger}\boldsymbol{\epsilon}^{\mathcal{T}} \\
&= \sigma_\epsilon^2 + \frac{1}{N}\left(\boldsymbol{\epsilon}^{\mathcal{T}}\right)'\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{\epsilon}^{\mathcal{T}} \\
&= \sigma_\epsilon^2 + \frac{1}{N}\left(\boldsymbol{\epsilon}^{\mathcal{T}}\right)'\underbrace{\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'}_{=:P_X}\boldsymbol{\epsilon}^{\mathcal{T}},
\end{aligned}
$$

where the projection matrix P_X is the orthogonal projection onto the column space of $\boldsymbol{X}$. Assuming that $\boldsymbol{X}$ has full column rank, which is a necessary and sufficient condition for $\boldsymbol{X}'\boldsymbol{X}$ to be invertible, it follows that $\operatorname{rank}(P_X) = p$.

In order to derive the expected in-sample error, we make use of the following result: If P is a projection matrix with rank r and $z \sim N(0, \boldsymbol{I})$, then the quadratic form $z'Pz$ is distributed as $\chi^2(r)$. In particular,

$$
\left(\sigma_\epsilon^{-1}\boldsymbol{\epsilon}^{\mathcal{T}}\right)' P_X \left(\sigma_\epsilon^{-1}\boldsymbol{\epsilon}^{\mathcal{T}}\right) \sim \chi^2(p).
$$

Consequently, we find that the expected in-sample error is given by

$$
\begin{aligned}
\mathbb{E}_{\mathcal{T}}\left[Err_{in}\right] &= \mathbb{E}_{\mathcal{T}}\left[\sigma_\epsilon^2 + \frac{\sigma_\epsilon^2}{N}\left(\sigma_\epsilon^{-1}\boldsymbol{\epsilon}^{\mathcal{T}}\right)' P_X \left(\sigma_\epsilon^{-1}\boldsymbol{\epsilon}^{\mathcal{T}}\right)\right] \\
&= \sigma_\epsilon^2 + \frac{\sigma_\epsilon^2}{N}\underbrace{\mathbb{E}_{\mathcal{T}}\left[\left(\sigma_\epsilon^{-1}\boldsymbol{\epsilon}^{\mathcal{T}}\right)' P_X \left(\sigma_\epsilon^{-1}\boldsymbol{\epsilon}^{\mathcal{T}}\right)\right]}_{=p} \\
&= \sigma_\epsilon^2\left(1 + \frac{p}{N}\right).
\end{aligned}
$$

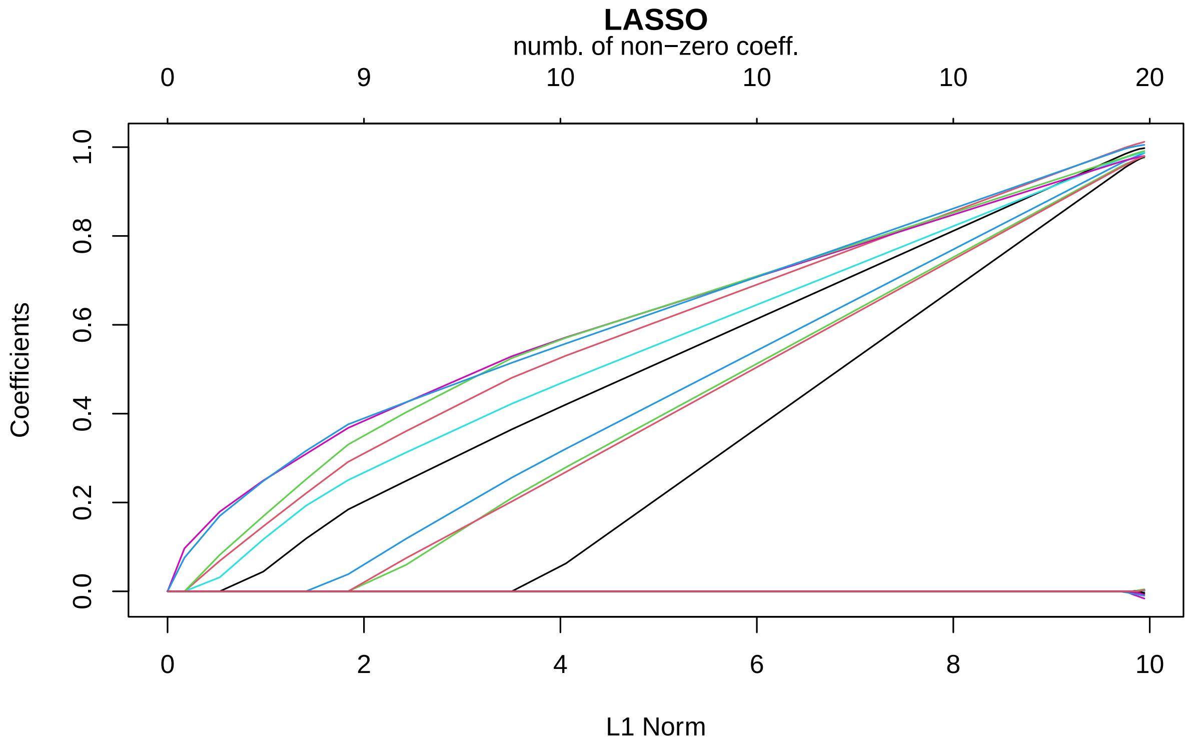
Exercise 5

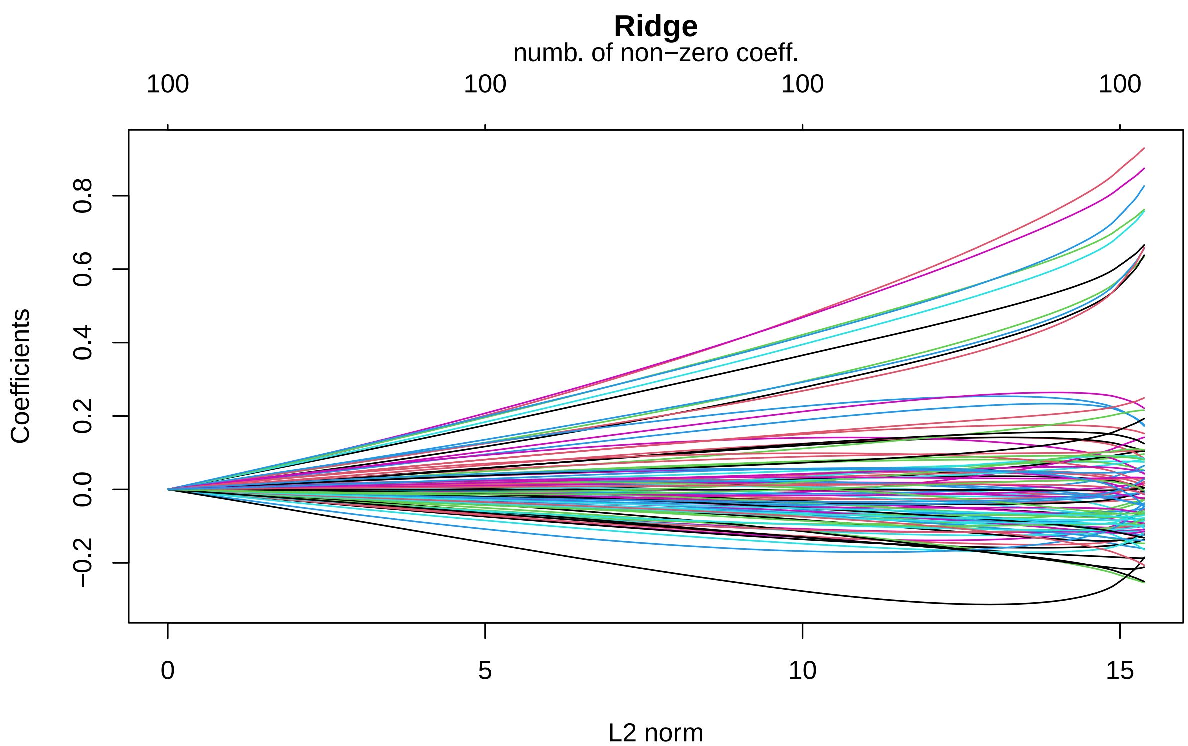
We use artificial data to perform LASSO and ridge regression. First, we set a random seed before the analysis. Then, we draw 100 observations from a 100-dimensional standard multivariate normal distribution. This is the matrix of covariates X of dimension 100×100. Next, we draw 100 observations for the dependent variable given by

$$
y = \sum_{i=1}^{10} x_i + \epsilon, \quad \text{with } \epsilon \sim N(0, 0.1)
$$

Now, we fit LASSO and Ridge models with different values of λ using function *glmnet* from package **glmnet**.

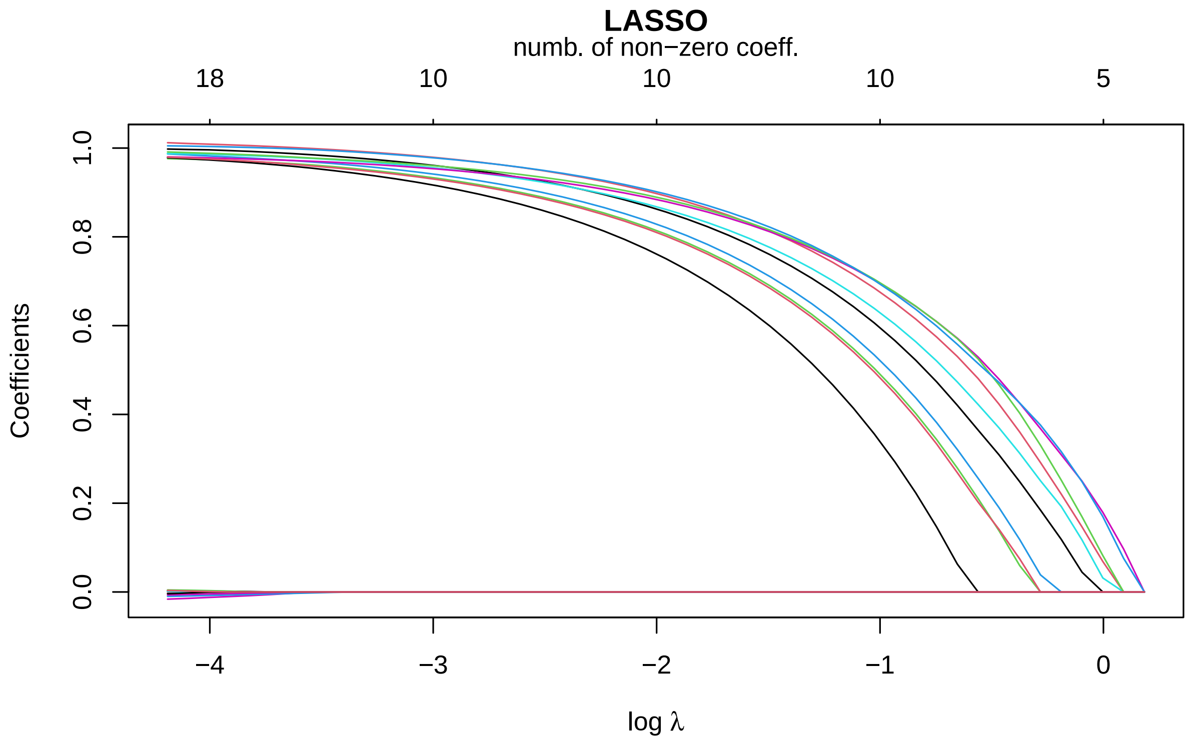
We plot the default plots for the returned objects.

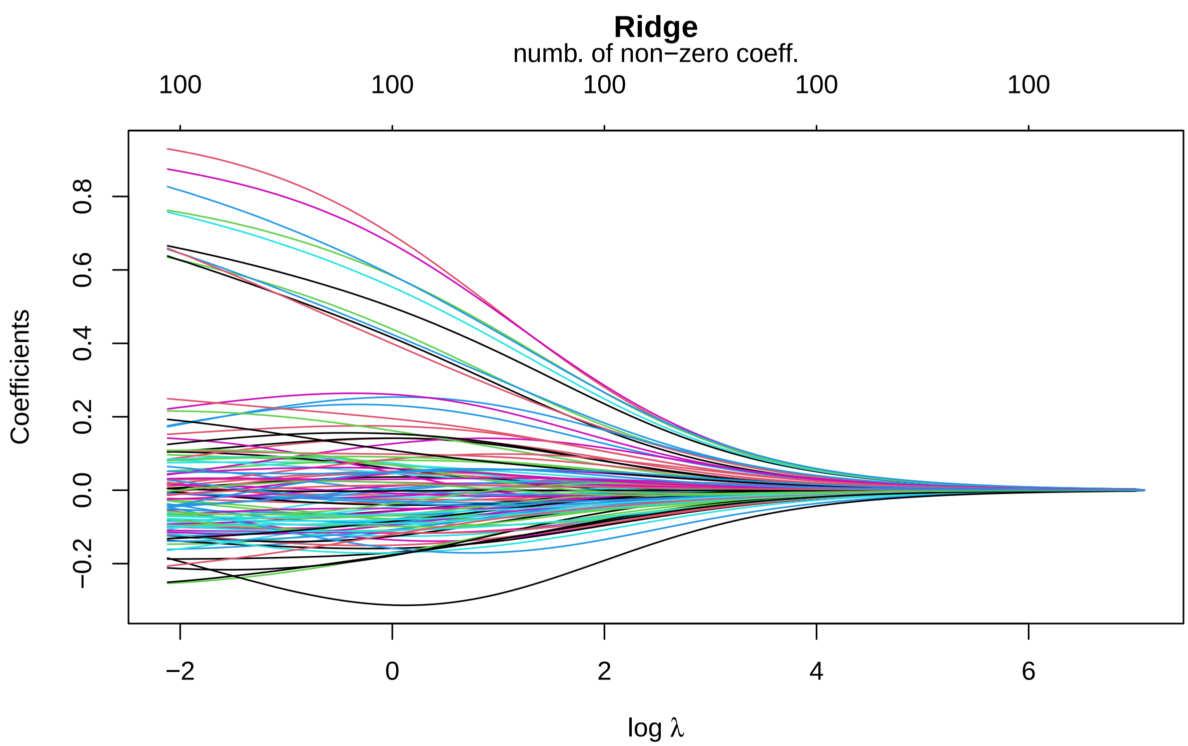




Each line represents one variable. L1 norm is the regularization term of LASSO, and L2 the regularization term of Ridge. A small L1 or L2 norm represent a lot of regularization. On the other hand, a high L1 or L2 norm represent low regularization. For LASSO, an L1 norm of zero gives the null model. Variables enter the model with increasing L1 norm, as their coefficients take non-zero values. On the top axis, we see the number of non-zero coefficients. For Ridge, an L2 norm of zero also gives the null model. But compared to Ridge, all variables enter right away. Also note, that some are negative.

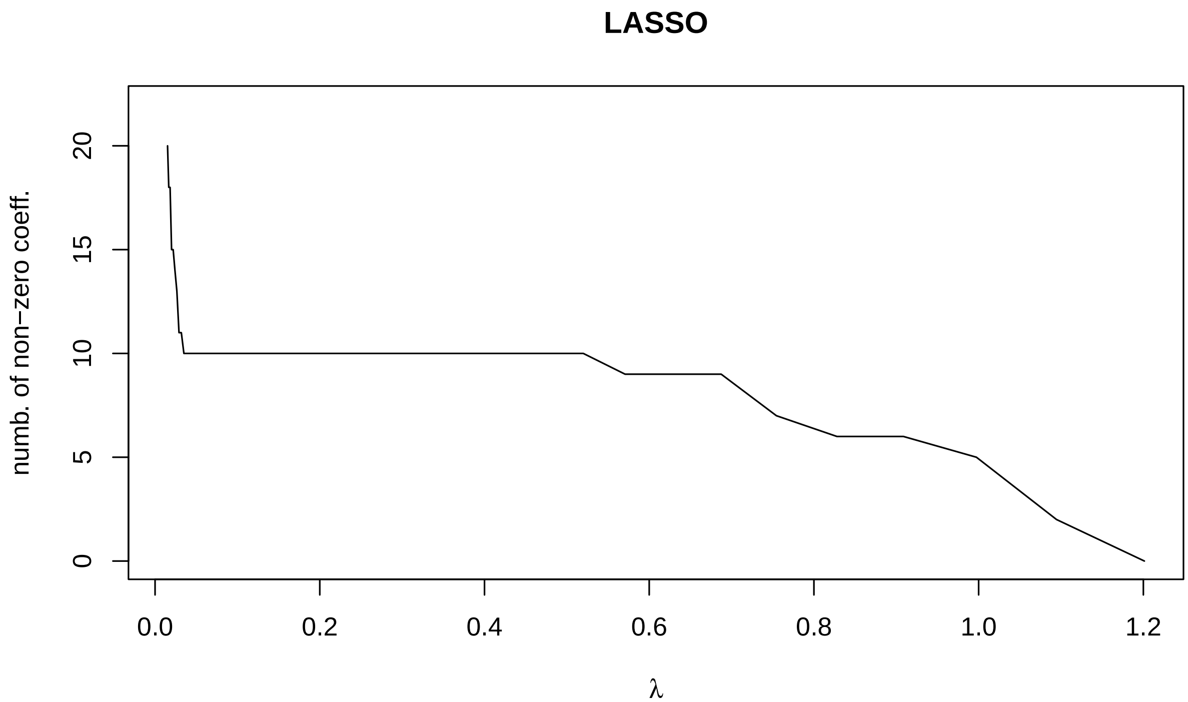
Additionally, we plot the plots where the argument *xvar* is set to *"lambda"*.

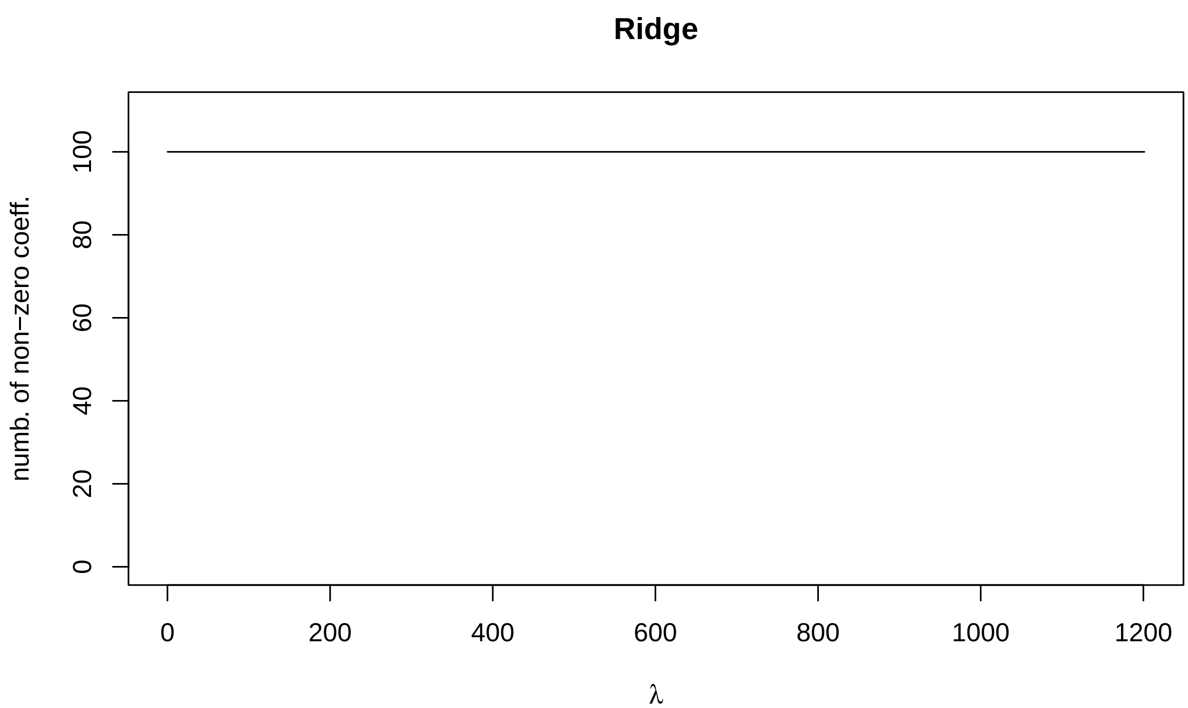




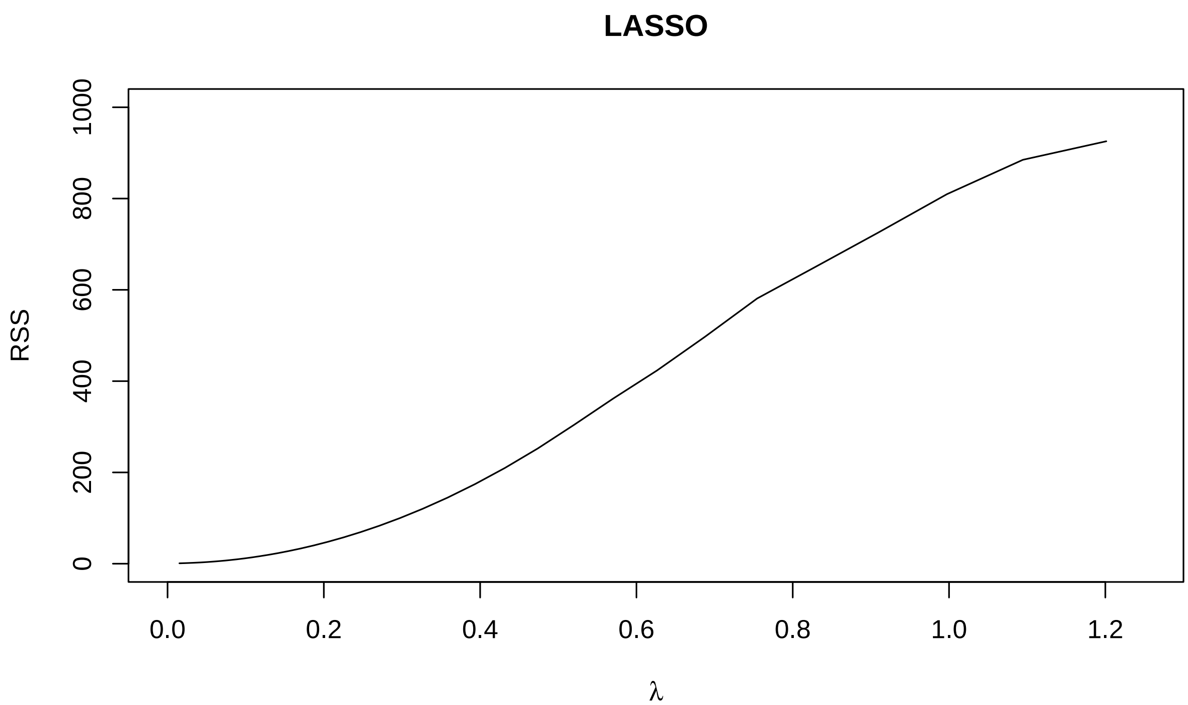
We basically see the same as before, just on another scale. This time the x-axis is log λ, the logarithm of the weight given to the regularization. λ is therefore the complexity parameter. For $\lambda = 0$, the solution is the OLS solution.

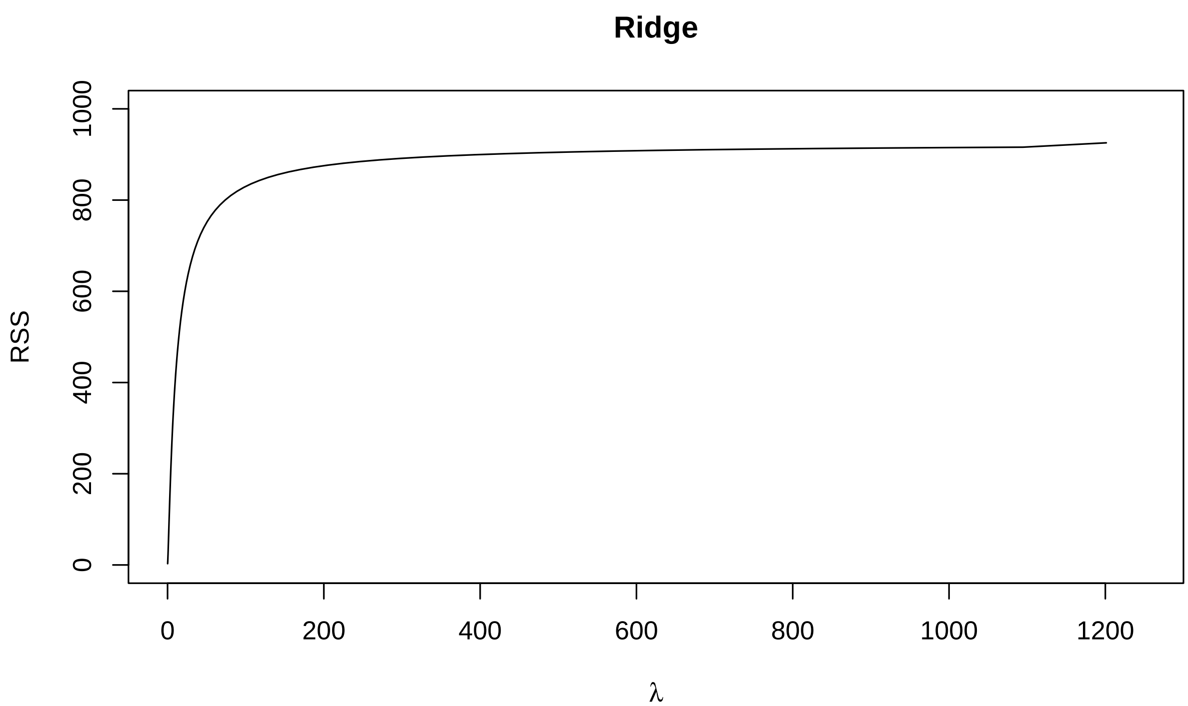
Next, we determine the number of non-zero coefficients in dependence of λ for LASSO and ridge.





We can see, that for LASSO, the number of non-zero coefficients decreases with λ. We actually stay at 10 non-zero coefficents for λ values between $\sim 0.05 - 0.55$. For Ridge all variables are in the model for all values of λ. Finally, we find the model fit as measured by the deviance() (= RSS) in dependence of λ for LASSO and Ridge.





In general, a low λ gives a better fit (lower RSS). Compared to LASS0, for Ridge the RSS increases faster with increasing λ. This could be a result of all variables entering already with low levels of λ.