Exercise 2 (2025) — Advanced Methods for Regression and Classification

Olesia Galynskaia 12321492

2025-10-22

Loading and observing data

```r
load("building.RData")
stopifnot(exists("df"), is.data.frame(df))
attributes(df)
```

```
## $names
##   [1] "y"                  "START.YEAR"         "START.QUARTER"
##   [4] "COMPLETION.YEAR"    "COMPLETION.QUARTER" "PhysFin1"
##   [7] "PhysFin2"           "PhysFin3"           "PhysFin4"
##  [10] "PhysFin5"           "PhysFin6"           "PhysFin7"
##  [13] "PhysFin8"           "Econ1"              "Econ2"
##  [16] "Econ3"              "Econ4"              "Econ5"
##  [19] "Econ6"              "Econ7"              "Econ8"
##  [22] "Econ9"              "Econ10"             "Econ11"
##  [25] "Econ12"             "Econ13"             "Econ14"
##  [28] "Econ15"             "Econ16"             "Econ17"
##  [31] "Econ18"             "Econ19"             "Econ1.lag1"
##  [34] "Econ2.lag1"         "Econ3.lag1"         "Econ4.lag1"
##  [37] "Econ5.lag1"         "Econ6.lag1"         "Econ7.lag1"
##  [40] "Econ8.lag1"         "Econ9.lag1"         "Econ10.lag1"
##  [43] "Econ11.lag1"        "Econ12.lag1"        "Econ13.lag1"
##  [46] "Econ14.lag1"        "Econ15.lag1"        "Econ16.lag1"
##  [49] "Econ17.lag1"        "Econ18.lag1"        "Econ19.lag1"
##  [52] "Econ1.lag2"         "Econ2.lag2"         "Econ3.lag2"
##  [55] "Econ4.lag2"         "Econ5.lag2"         "Econ6.lag2"
##  [58] "Econ7.lag2"         "Econ8.lag2"         "Econ9.lag2"
##  [61] "Econ10.lag2"        "Econ11.lag2"        "Econ12.lag2"
##  [64] "Econ13.lag2"        "Econ14.lag2"        "Econ15.lag2"
##  [67] "Econ16.lag2"        "Econ17.lag2"        "Econ18.lag2"
##  [70] "Econ19.lag2"        "Econ1.lag3"         "Econ2.lag3"
##  [73] "Econ3.lag3"         "Econ4.lag3"         "Econ5.lag3"
##  [76] "Econ6.lag3"         "Econ7.lag3"         "Econ8.lag3"
```

```
##  [79] "Econ9.lag3"          "Econ10.lag3"         "Econ11.lag3"
##  [82] "Econ12.lag3"         "Econ13.lag3"         "Econ14.lag3"
##  [85] "Econ15.lag3"         "Econ16.lag3"         "Econ17.lag3"
##  [88] "Econ18.lag3"         "Econ19.lag3"         "Econ1.lag4"
##  [91] "Econ2.lag4"          "Econ3.lag4"          "Econ4.lag4"
##  [94] "Econ5.lag4"          "Econ6.lag4"          "Econ7.lag4"
##  [97] "Econ8.lag4"          "Econ9.lag4"          "Econ10.lag4"
## [100] "Econ11.lag4"         "Econ12.lag4"         "Econ13.lag4"
## [103] "Econ14.lag4"         "Econ15.lag4"         "Econ16.lag4"
## [106] "Econ17.lag4"         "Econ18.lag4"         "Econ19.lag4"
##
## $class
## [1] "data.frame"
##
## $row.names
##   [1]   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
##  [19]  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
##  [37]  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54
##  [55]  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72
##  [73]  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
##  [91]  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108
## [109] 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126
## [127] 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144
## [145] 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162
## [163] 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
## [181] 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198
## [199] 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216
## [217] 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234
## [235] 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252
## [253] 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270
## [271] 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288
## [289] 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306
## [307] 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324
## [325] 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342
## [343] 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360
## [361] 361 362 363 364 365 366 367 368 369 370 371 372
```

```r
str(attributes(df))
```

```
## List of 3
##  $ names    : chr [1:108] "y" "START.YEAR" "START.QUARTER" "COMPLETION.YEAR" ...
##  $ class    : chr "data.frame"
##  $ row.names: int [1:372] 1 2 3 4 5 6 7 8 9 10 ...
```

```r
head(sapply(df, function(x) attr(x, "label")), 10)
```

```
## $y
```

```
## NULL
##
## $START.YEAR
## NULL
##
## $START.QUARTER
## NULL
##
## $COMPLETION.YEAR
## NULL
##
## $COMPLETION.QUARTER
## NULL
##
## $PhysFin1
## NULL
##
## $PhysFin2
## NULL
##
## $PhysFin3
## NULL
##
## $PhysFin4
## NULL
##
## $PhysFin5
## NULL
```

Quick structure and summary

```r
cat("**Dimensions:**", nrow(df), "rows ×", ncol(df), "columns\n\n")
```

```
## **Dimensions:** 372 rows × 108 columns
```

```r
str(df[, 1:10])
```

```
## 'data.frame':    372 obs. of  10 variables:
##  $ y                 : num  7.7 8.52 7.09 5.11 8.61 ...
##  $ START.YEAR        : num  81 84 78 72 87 87 87 88 76 80 ...
##  $ START.QUARTER     : num  1 1 1 2 1 1 2 1 3 1 ...
##  $ COMPLETION.YEAR   : num  85 89 81 73 90 90 90 89 77 80 ...
##  $ COMPLETION.QUARTER: num  1 4 4 2 2 1 1 3 4 4 ...
##  $ PhysFin1          : num  1 1 1 1 1 1 1 1 1 1 ...
```

```
##  $ PhysFin2         : num  3150 7600 4800 685 3000 2500 1810 1150 2110 3030 ...
##  $ PhysFin3         : num  920 1140 840 202 800 640 492 380 540 930 ...
##  $ PhysFin4         : num  598.5 3040 480 13.7 1230 ...
##  $ PhysFin5         : num  190 400 100 20 410 420 640 500 90 170 ...
```

```
summary(df)
```

```
##        y           START.YEAR    START.QUARTER   COMPLETION.YEAR
##  Min.   :3.912   Min.   :72.00   Min.   :1.000   Min.   :73.00  
##  1st Qu.:6.359   1st Qu.:78.00   1st Qu.:1.000   1st Qu.:80.00  
##  Median :6.908   Median :82.00   Median :2.000   Median :84.00  
##  Mean   :6.902   Mean   :81.48   Mean   :2.191   Mean   :82.95  
##  3rd Qu.:7.438   3rd Qu.:85.00   3rd Qu.:3.000   3rd Qu.:87.00  
##  Max.   :8.825   Max.   :88.00   Max.   :4.000   Max.   :90.00  
##  COMPLETION.QUARTER    PhysFin1         PhysFin2        PhysFin3     
##  Min.   :1.000      Min.   : 1.000   Min.   :  200   Min.   :  60.0  
##  1st Qu.:2.000      1st Qu.: 4.000   1st Qu.:  720   1st Qu.: 190.0  
##  Median :3.000      Median : 8.000   Median : 1220   Median : 300.0  
##  Mean   :2.586      Mean   : 9.728   Mean   : 1729   Mean   : 426.1  
##  3rd Qu.:4.000      3rd Qu.:17.000   3rd Qu.: 2100   3rd Qu.: 490.5  
##  Max.   :4.000      Max.   :20.000   Max.   :15670   Max.   :5000.0  
##     PhysFin4          PhysFin5        PhysFin6          PhysFin7     
##  Min.   :   3.7   Min.   : 10.0   Min.   : 193.1   Min.   : 2.000  
##  1st Qu.:  67.8   1st Qu.: 80.0   1st Qu.: 391.7   1st Qu.: 5.000  
##  Median : 164.7   Median :140.0   Median : 522.5   Median : 6.000  
##  Mean   : 327.9   Mean   :163.1   Mean   : 554.4   Mean   : 6.266  
##  3rd Qu.: 366.1   3rd Qu.:230.0   3rd Qu.: 667.9   3rd Qu.: 7.000  
##  Max.   :7208.2   Max.   :640.0   Max.   :3436.9   Max.   :23.000  
##     PhysFin8        Econ1          Econ2            Econ3       
##  Min.   :  40   Min.   :1562   Min.   : 12.10   Min.   : 10.03  
##  1st Qu.: 440   1st Qu.:2842   1st Qu.: 45.60   1st Qu.: 51.63  
##  Median : 805   Median :3629   Median : 74.90   Median : 79.28  
##  Mean   :1088   Mean   :4211   Mean   : 94.43   Mean   : 88.05  
##  3rd Qu.:1300   3rd Qu.:6024   3rd Qu.:137.40   3rd Qu.:125.83  
##  Max.   :5700   Max.   :7196   Max.   :274.00   Max.   :225.00  
##      Econ4           Econ5             Econ6              Econ7       
##  Min.   :0.920   Min.   :  38194   Min.   :  287.2   Min.   : 13.60  
##  1st Qu.:2.470   1st Qu.: 183726   1st Qu.: 1979.0   1st Qu.: 39.70  
##  Median :3.250   Median : 445458   Median : 3819.0   Median : 87.05  
##  Mean   :3.605   Mean   : 641112   Mean   : 4805.6   Mean   : 98.68  
##  3rd Qu.:4.720   3rd Qu.:1059966   3rd Qu.: 6622.5   3rd Qu.:117.40  
##  Max.   :6.880   Max.   :2171923   Max.   :18690.9   Max.   :319.38  
##      Econ8            Econ9             Econ10          Econ11      
##  Min.   : 17.03   Min.   :  154.4   Min.   :11.00   Min.   : 170.3  
##  1st Qu.: 93.00   1st Qu.: 3622.2   1st Qu.:14.00   1st Qu.: 641.5  
##  Median :162.75   Median :10445.6   Median :15.00   Median :1023.7  
##  Mean   :182.00   Mean   :18861.3   Mean   :14.07   Mean   :1327.5  
```

```
##  3rd Qu.:242.27   3rd Qu.:21723.4   3rd Qu.:15.00   3rd Qu.:1994.6
##  Max.   :432.40   Max.   :73143.5   Max.   :15.00   Max.   :4188.6
##      Econ12           Econ13        Econ14          Econ15
##  Min.   : 211.1   Min.   :1592   Min.   : 1601   Min.   : 11.62
##  1st Qu.: 744.5   1st Qu.:1755   1st Qu.: 8001   1st Qu.: 51.89
##  Median :1203.3   Median :8210   Median : 8393   Median : 84.46
##  Mean   :1466.3   Mean   :5934   Mean   : 7805   Mean   : 88.38
##  3rd Qu.:2025.0   3rd Qu.:9138   3rd Qu.: 9208   3rd Qu.:123.37
##  Max.   :4741.6   Max.   :9967   Max.   :10099   Max.   :204.70
##      Econ16           Econ17           Econ18          Econ19
##  Min.   : 10.06   Min.   :  354.6   Min.   : 8436   Min.   : 141543
##  1st Qu.: 42.87   1st Qu.: 2134.5   1st Qu.:12393   1st Qu.: 588021
##  Median : 81.47   Median : 7334.8   Median :26438   Median : 825511
##  Mean   : 87.07   Mean   : 6604.9   Mean   :28297   Mean   :1041556
##  3rd Qu.:127.33   3rd Qu.:10082.0   3rd Qu.:41407   3rd Qu.:1660444
##  Max.   :222.60   Max.   :13596.4   Max.   :50928   Max.   :2606321
##    Econ1.lag1     Econ2.lag1       Econ3.lag1       Econ4.lag1
##  Min.   :1562   Min.   : 11.60   Min.   :  8.50   Min.   :0.920
##  1st Qu.:2734   1st Qu.: 44.50   1st Qu.: 49.80   1st Qu.:2.440
##  Median :3561   Median : 71.15   Median : 77.46   Median :3.150
##  Mean   :3990   Mean   : 89.63   Mean   : 84.38   Mean   :3.413
##  3rd Qu.:5606   3rd Qu.:130.50   3rd Qu.:117.05   3rd Qu.:4.300
##  Max.   :7196   Max.   :267.80   Max.   :225.00   Max.   :6.880
##    Econ5.lag1        Econ6.lag1        Econ7.lag1       Econ8.lag1
##  Min.   :  35859   Min.   :  287.2   Min.   : 12.67   Min.   : 17.03
##  1st Qu.: 176543   1st Qu.: 1861.2   1st Qu.: 35.00   1st Qu.: 98.33
##  Median : 422306   Median : 3663.5   Median : 83.80   Median :167.05
##  Mean   : 600257   Mean   : 4594.8   Mean   : 92.15   Mean   :186.69
##  3rd Qu.: 961139   3rd Qu.: 5146.3   3rd Qu.:112.80   3rd Qu.:252.88
##  Max.   :2116614   Max.   :18690.9   Max.   :306.93   Max.   :432.40
##    Econ9.lag1       Econ10.lag1     Econ11.lag1      Econ12.lag1
##  Min.   :  154.4   Min.   :11.00   Min.   : 165.1   Min.   : 208.6
##  1st Qu.: 3622.2   1st Qu.:14.00   1st Qu.: 627.6   1st Qu.: 717.9
##  Median :10866.5   Median :15.00   Median :1010.0   Median :1176.5
##  Mean   :18415.3   Mean   :14.18   Mean   :1249.0   Mean   :1385.7
##  3rd Qu.:21723.4   3rd Qu.:15.00   3rd Qu.:1821.6   3rd Qu.:1932.5
##  Max.   :73143.5   Max.   :15.00   Max.   :3962.2   Max.   :4472.3
##   Econ13.lag1    Econ14.lag1      Econ15.lag1      Econ16.lag1
##  Min.   :1504   Min.   : 1582   Min.   : 10.86   Min.   :  9.79
##  1st Qu.:1755   1st Qu.: 7994   1st Qu.: 50.28   1st Qu.: 41.80
##  Median :8075   Median : 8382   Median : 81.60   Median : 78.48
##  Mean   :5724   Mean   : 7714   Mean   : 84.91   Mean   : 83.43
##  3rd Qu.:9133   3rd Qu.: 9168   3rd Qu.:120.24   3rd Qu.:121.94
##  Max.   :9967   Max.   :10099   Max.   :201.66   Max.   :218.40
##   Econ17.lag1       Econ18.lag1     Econ19.lag1       Econ1.lag2
##  Min.   :  354.6   Min.   : 8436   Min.   : 129102   Min.   :1562
##  1st Qu.: 2000.4   1st Qu.:18967   1st Qu.: 566492   1st Qu.:2700
##  Median : 5900.0   Median :31940   Median : 802773   Median :3561
```

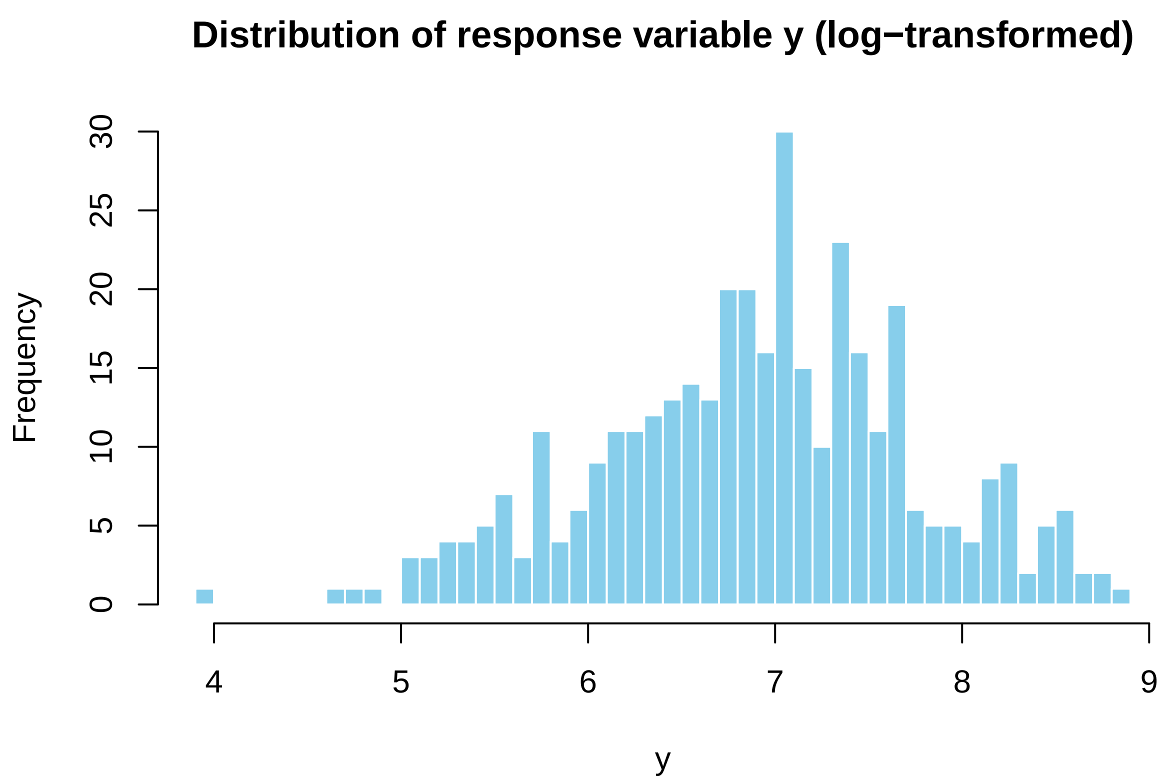
```
##  Mean   : 6462.1   Mean   :29170   Mean   : 987881   Mean   :3886
##  3rd Qu.:10082.0   3rd Qu.:37179   3rd Qu.:1654038   3rd Qu.:4986
##  Max.   :13596.4   Max.   :50928   Max.   :2435004   Max.   :7196
##    Econ2.lag2       Econ3.lag2       Econ4.lag2     Econ5.lag2
##  Min.   : 11.40   Min.   :  6.97   Min.   :0.92   Min.   :  32794
##  1st Qu.: 43.40   1st Qu.: 46.94   1st Qu.:2.45   1st Qu.: 166267
##  Median : 67.80   Median : 74.71   Median :3.05   Median : 399813
##  Mean   : 85.21   Mean   : 81.18   Mean   :3.36   Mean   : 563182
##  3rd Qu.:124.40   3rd Qu.:108.42   3rd Qu.:3.94   3rd Qu.: 921019
##  Max.   :261.50   Max.   :225.00   Max.   :6.88   Max.   :1970485
##    Econ6.lag2        Econ7.lag2       Econ8.lag2       Econ9.lag2
##  Min.   :  287.2   Min.   : 11.73   Min.   : 17.03   Min.   :  154.4
##  1st Qu.: 1668.9   1st Qu.: 34.70   1st Qu.:104.40   1st Qu.: 3994.7
##  Median : 3755.8   Median : 79.30   Median :167.05   Median : 9342.5
##  Mean   : 4238.8   Mean   : 86.49   Mean   :174.29   Mean   :16370.4
##  3rd Qu.: 5138.6   3rd Qu.:110.30   3rd Qu.:217.00   3rd Qu.:21723.4
##  Max.   :18690.9   Max.   :306.70   Max.   :432.40   Max.   :73143.5
##   Econ10.lag2     Econ11.lag2      Econ12.lag2      Econ13.lag2
##  Min.   :11.00   Min.   : 165.1   Min.   : 208.6   Min.   :1450
##  1st Qu.:14.00   1st Qu.: 627.6   1st Qu.: 680.3   1st Qu.:1755
##  Median :15.00   Median : 956.0   Median :1054.7   Median :7990
##  Mean   :14.26   Mean   :1189.0   Mean   :1314.7   Mean   :5590
##  3rd Qu.:15.00   3rd Qu.:1821.6   3rd Qu.:1932.5   3rd Qu.:9114
##  Max.   :15.00   Max.   :3746.0   Max.   :4215.9   Max.   :9967
##   Econ14.lag2     Econ15.lag2      Econ16.lag2      Econ17.lag2
##  Min.   : 1507   Min.   : 10.17   Min.   :  9.35   Min.   :  354.6
##  1st Qu.: 7994   1st Qu.: 49.92   1st Qu.: 40.26   1st Qu.: 1976.3
##  Median : 8354   Median : 77.53   Median : 75.29   Median : 5097.0
##  Mean   : 7623   Mean   : 81.66   Mean   : 79.71   Mean   : 6349.5
##  3rd Qu.: 9131   3rd Qu.:116.56   3rd Qu.:119.13   3rd Qu.:10149.0
##  Max.   :10099   Max.   :196.76   Max.   :215.00   Max.   :13596.4
##   Econ18.lag2     Econ19.lag2        Econ1.lag3     Econ2.lag3
##  Min.   : 8436   Min.   : 123618   Min.   :1562   Min.   : 10.60
##  1st Qu.:20704   1st Qu.: 540681   1st Qu.:2647   1st Qu.: 41.00
##  Median :24786   Median : 740309   Median :3321   Median : 64.40
##  Mean   :27456   Mean   : 939045   Mean   :3866   Mean   : 80.76
##  3rd Qu.:36195   3rd Qu.:1391757   3rd Qu.:4986   3rd Qu.:120.20
##  Max.   :50928   Max.   :2435004   Max.   :7196   Max.   :259.50
##    Econ3.lag3       Econ4.lag3      Econ5.lag3        Econ6.lag3
##  Min.   :  5.44   Min.   :0.920   Min.   :  30013   Min.   :  287.2
##  1st Qu.: 41.25   1st Qu.:2.320   1st Qu.: 160402   1st Qu.: 1571.1
##  Median : 71.94   Median :2.945   Median : 373644   Median : 3755.8
##  Mean   : 78.06   Mean   :3.193   Mean   : 525388   Mean   : 3944.4
##  3rd Qu.:107.20   3rd Qu.:3.720   3rd Qu.: 832124   3rd Qu.: 5131.4
##  Max.   :225.00   Max.   :6.880   Max.   :1901366   Max.   :18690.9
##    Econ7.lag3       Econ8.lag3       Econ9.lag3       Econ10.lag3
##  Min.   : 10.79   Min.   : 17.03   Min.   :  154.4   Min.   :11.00
##  1st Qu.: 34.40   1st Qu.: 74.85   1st Qu.: 2996.0   1st Qu.:14.00
```

```
##  Median : 75.60   Median :119.75   Median : 7834.2   Median :15.00
##  Mean   : 81.54   Mean   :145.84   Mean   :13351.0   Mean   :14.31
##  3rd Qu.:109.60   3rd Qu.:208.80   3rd Qu.:17361.2   3rd Qu.:15.00
##  Max.   :306.70   Max.   :432.40   Max.   :73143.5   Max.   :15.00
##   Econ11.lag3      Econ12.lag3      Econ13.lag3    Econ14.lag3
##  Min.   : 165.1   Min.   : 158.4   Min.   :1439   Min.   : 1450
##  1st Qu.: 611.8   1st Qu.: 677.7   1st Qu.:1755   1st Qu.: 7773
##  Median : 896.8   Median : 971.5   Median :7954   Median : 8325
##  Mean   :1140.1   Mean   :1245.4   Mean   :5522   Mean   : 7537
##  3rd Qu.:1763.9   3rd Qu.:1837.4   3rd Qu.:9064   3rd Qu.: 9078
##  Max.   :3499.4   Max.   :3823.6   Max.   :9967   Max.   :10099
##   Econ15.lag3      Econ16.lag3      Econ17.lag3       Econ18.lag3
##  Min.   :  9.91   Min.   :  8.85   Min.   :  354.6   Min.   : 8436
##  1st Qu.: 45.91   1st Qu.: 38.34   1st Qu.: 1966.4   1st Qu.:11774
##  Median : 74.50   Median : 71.46   Median : 4909.7   Median :21855
##  Mean   : 78.93   Mean   : 76.32   Mean   : 6131.1   Mean   :23470
##  3rd Qu.:112.15   3rd Qu.:115.70   3rd Qu.:10078.4   3rd Qu.:32783
##  Max.   :191.63   Max.   :212.10   Max.   :13596.4   Max.   :50928
##   Econ19.lag3        Econ1.lag4     Econ2.lag4       Econ3.lag4
##  Min.   : 121857   Min.   :1381   Min.   : 10.00   Min.   :  3.91
##  1st Qu.: 524765   1st Qu.:2641   1st Qu.: 40.30   1st Qu.: 40.84
##  Median : 681120   Median :3255   Median : 60.85   Median : 68.18
##  Mean   : 910297   Mean   :3757   Mean   : 76.65   Mean   : 74.52
##  3rd Qu.:1183641   3rd Qu.:4691   3rd Qu.:116.30   3rd Qu.:104.71
##  Max.   :2435004   Max.   :7196   Max.   :255.80   Max.   :225.00
##    Econ4.lag4     Econ5.lag4        Econ6.lag4        Econ7.lag4
##  Min.   :0.92   Min.   :  27231   Min.   :  287.2   Min.   :  9.85
##  1st Qu.:2.44   1st Qu.: 150267   1st Qu.: 1554.8   1st Qu.: 34.10
##  Median :2.84   Median : 352256   Median : 3485.8   Median : 72.25
##  Mean   :3.16   Mean   : 493874   Mean   : 3588.1   Mean   : 76.56
##  3rd Qu.:3.56   3rd Qu.: 784949   3rd Qu.: 4730.8   3rd Qu.:109.10
##  Max.   :6.88   Max.   :1704944   Max.   :18690.9   Max.   :306.70
##    Econ8.lag4       Econ9.lag4        Econ10.lag4     Econ11.lag4
##  Min.   : 14.15   Min.   :  152.6   Min.   :11.00   Min.   : 165.1
##  1st Qu.: 83.70   1st Qu.: 2967.7   1st Qu.:14.00   1st Qu.: 614.0
##  Median :148.80   Median : 7874.4   Median :15.00   Median : 859.1
##  Mean   :174.59   Mean   :15297.0   Mean   :14.45   Mean   :1082.0
##  3rd Qu.:251.10   3rd Qu.:17584.3   3rd Qu.:15.00   3rd Qu.:1534.6
##  Max.   :432.40   Max.   :73143.5   Max.   :15.00   Max.   :3447.8
##   Econ12.lag4      Econ13.lag4    Econ14.lag4     Econ15.lag4
##  Min.   : 152.2   Min.   :1439   Min.   : 1450   Min.   :  9.73
##  1st Qu.: 669.8   1st Qu.:1755   1st Qu.: 6714   1st Qu.: 43.40
##  Median : 938.4   Median :7928   Median : 8315   Median : 72.56
##  Mean   :1187.5   Mean   :5403   Mean   : 7432   Mean   : 76.29
##  3rd Qu.:1795.3   3rd Qu.:9001   3rd Qu.: 9022   3rd Qu.:109.02
##  Max.   :3686.3   Max.   :9967   Max.   :10099   Max.   :190.50
##   Econ16.lag4      Econ17.lag4       Econ18.lag4     Econ19.lag4
##  Min.   :  8.34   Min.   :  354.6   Min.   : 8194   Min.   : 121857
```

```
##  1st Qu.: 36.45   1st Qu.: 1917.4   1st Qu.:12065   1st Qu.: 519680  
##  Median : 67.45   Median : 4525.4   Median :25759   Median : 659243  
##  Mean   : 73.45   Mean   : 5915.6   Mean   :27552   Mean   : 878971  
##  3rd Qu.:112.00   3rd Qu.: 9821.0   3rd Qu.:40234   3rd Qu.:1181856  
##  Max.   :204.80   Max.   :13596.4   Max.   :49572   Max.   :2435004  
```

Plot of response variable

```r
hist(df$y, breaks = 40,
     main = "Distribution of response variable y (log-transformed)",
     xlab = "y", col = "skyblue", border = "white")
```




```r
# compute correlations
num_data <- df[, sapply(df, is.numeric)]
cor_mat <- cor(num_data, use = "pairwise.complete.obs")

# keep only pairs with |r| > 0.9
high_corr <- which(abs(cor_mat) > 0.9 & abs(cor_mat) < 1, arr.ind = TRUE)
corr_pairs <- unique(t(apply(high_corr, 1, sort)))

cat("Highly correlated pairs (|r| > 0.9):\n")
```

```
## Highly correlated pairs (|r| > 0.9):
```

```r
print(head(data.frame(
  Var1 = rownames(cor_mat)[corr_pairs[,1]],
  Var2 = colnames(cor_mat)[corr_pairs[,2]],
  r = round(cor_mat[corr_pairs], 3)
), 10))
```

```
##          Var1            Var2     r
## 1  START.YEAR COMPLETION.YEAR 0.988
## 2  START.YEAR           Econ2 0.905
## 3  START.YEAR           Econ3 0.934
## 4  START.YEAR          Econ11 0.909
## 5  START.YEAR          Econ14 0.900
## 6  START.YEAR          Econ15 0.965
## 7  START.YEAR          Econ16 0.956
## 8  START.YEAR          Econ19 0.902
## 9  START.YEAR      Econ2.lag1 0.908
## 10 START.YEAR      Econ3.lag1 0.939
```

Data preparation

```r
# 1) Train/Test split (2/3 : 1/3) - clean and reproducible
set.seed(12321492)  # for reproducibility
stopifnot(exists("df"), is.data.frame(df), "y" %in% names(df))

n <- nrow(df)
idx_train <- sample(seq_len(n), size = floor(2/3 * n))

train <- df[idx_train, , drop = FALSE]
test  <- df[-idx_train, , drop = FALSE]

# function for RMSE
rmse <- function(actual, predicted) sqrt(mean((actual - predicted)^2))

# short info output
cat("Train:", nrow(train), "rows | Test:", nrow(test), "rows\n")
```

```
## Train: 248 rows | Test: 124 rows
```

Ex-1 Model

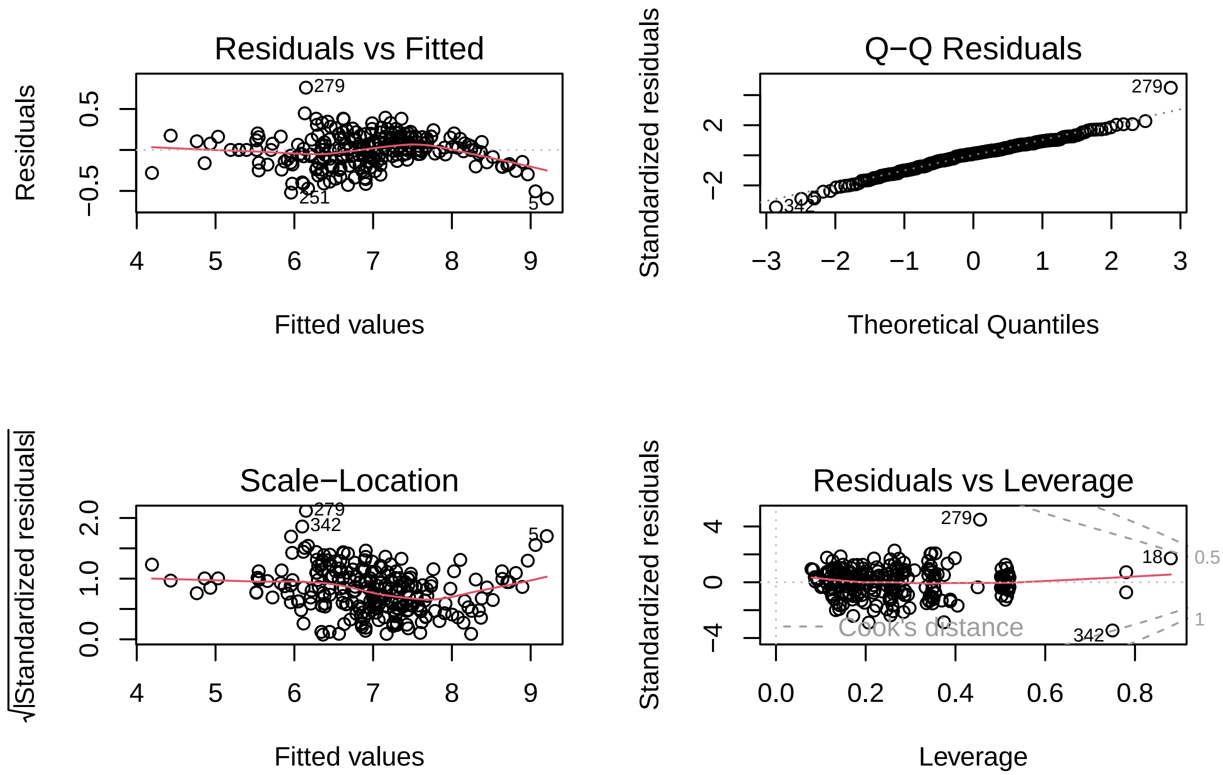
```r
## Fit on training data
lm_full <- lm(y ~ ., data = train)
summary(lm_full)
```

```
##
## Call:
## lm(formula = y ~ ., data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.59186 -0.10869  0.01195  0.13006  0.76019
##
## Coefficients: (35 not defined because of singularities)
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1.957e+02  1.112e+02   1.761  0.08002 .
## START.YEAR         -2.435e+00  1.637e+00  -1.488  0.13864
## START.QUARTER      -3.199e+00  1.447e+00  -2.211  0.02832 *
## COMPLETION.YEAR     8.133e-02  3.862e-02   2.106  0.03663 *
## COMPLETION.QUARTER  4.175e-02  1.974e-02   2.115  0.03585 *
## PhysFin1           -2.962e-02  4.994e-03  -5.933 1.56e-08 ***
## PhysFin2            1.417e-04  5.672e-05   2.498  0.01341 *
## PhysFin3           -3.873e-04  1.711e-04  -2.263  0.02484 *
## PhysFin4           -1.407e-04  7.366e-05  -1.910  0.05781 .
## PhysFin5           -2.996e-03  6.416e-04  -4.670 5.97e-06 ***
## PhysFin6            5.768e-04  1.365e-04   4.227 3.81e-05 ***
## PhysFin7                   NA         NA      NA       NA
## PhysFin8            4.841e-04  3.450e-05  14.033  < 2e-16 ***
## Econ1               3.088e-04  4.596e-04   0.672  0.50258
## Econ2               2.413e-01  2.522e-01   0.957  0.33995
## Econ3               2.831e-01  1.609e-01   1.759  0.08031 .
## Econ4               7.728e-01  4.113e-01   1.879  0.06188 .
## Econ5               5.494e-05  7.440e-05   0.738  0.46127
## Econ6               1.865e-04  3.727e-04   0.501  0.61734
## Econ7              -1.619e-01  8.186e-02  -1.977  0.04956 *
## Econ8               2.269e-02  9.881e-03   2.297  0.02282 *
## Econ9              -1.808e-04  1.638e-04  -1.104  0.27121
## Econ10              2.194e-01  4.268e-01   0.514  0.60792
## Econ11              3.935e-03  3.415e-03   1.152  0.25086
## Econ12              1.355e-03  2.422e-03   0.559  0.57656
## Econ13             -1.251e-04  6.257e-05  -1.999  0.04716 *
## Econ14              1.020e-04  5.480e-04   0.186  0.85254
## Econ15             -1.329e-01  1.800e-01  -0.739  0.46117
## Econ16              4.916e-01  3.080e-01   1.596  0.11225
## Econ17              7.055e-04  5.325e-04   1.325  0.18695
## Econ18             -3.400e-04  1.899e-04  -1.790  0.07511 .
## Econ19              3.268e-06  4.383e-06   0.746  0.45691
## Econ1.lag1         -9.149e-05  2.109e-04  -0.434  0.66490
```

```
## Econ2.lag1         -5.466e-01  4.927e-01  -1.110  0.26871
## Econ3.lag1         -1.012e-01  2.062e-01  -0.490  0.62441
## Econ4.lag1         -1.357e-01  2.955e-01  -0.459  0.64670
## Econ5.lag1         -4.480e-05  4.501e-05  -0.995  0.32097
## Econ6.lag1          3.482e-04  2.889e-04   1.205  0.22972
## Econ7.lag1         -8.481e-02  9.985e-02  -0.849  0.39683
## Econ8.lag1         -2.511e-03  7.100e-03  -0.354  0.72397
## Econ9.lag1          1.501e-04  1.101e-04   1.363  0.17459
## Econ10.lag1        -2.454e-01  2.959e-01  -0.829  0.40806
## Econ11.lag1        -3.610e-03  5.157e-03  -0.700  0.48480
## Econ12.lag1         9.389e-03  5.686e-03   1.651  0.10046
## Econ13.lag1         3.124e-04  3.736e-04   0.836  0.40424
## Econ14.lag1         2.003e-03  7.427e-04   2.698  0.00767 **
## Econ15.lag1         4.911e-01  2.857e-01   1.719  0.08745 .
## Econ16.lag1        -1.171e+00  8.081e-01  -1.449  0.14911
## Econ17.lag1        -3.150e-05  2.933e-04  -0.107  0.91459
## Econ18.lag1        -3.293e-04  1.727e-04  -1.907  0.05822 .
## Econ19.lag1        -6.541e-06  5.809e-06  -1.126  0.26172
## Econ1.lag2          3.008e-05  2.033e-04   0.148  0.88253
## Econ2.lag2          3.248e-01  2.702e-01   1.202  0.23097
## Econ3.lag2         -7.671e-02  2.516e-01  -0.305  0.76078
## Econ4.lag2         -6.467e-01  3.379e-01  -1.914  0.05727 .
## Econ5.lag2         -1.373e-05  6.217e-05  -0.221  0.82544
## Econ6.lag2         -9.205e-05  1.069e-03  -0.086  0.93150
## Econ7.lag2          1.016e-01  9.402e-02   1.081  0.28111
## Econ8.lag2          3.821e-03  1.358e-02   0.281  0.77876
## Econ9.lag2         -4.391e-05  1.060e-04  -0.414  0.67908
## Econ10.lag2         2.199e-02  2.614e-01   0.084  0.93307
## Econ11.lag2        -3.492e-03  4.257e-03  -0.820  0.41321
## Econ12.lag2         8.601e-03  7.981e-03   1.078  0.28265
## Econ13.lag2        -3.408e-04  2.588e-04  -1.317  0.18959
## Econ14.lag2        -3.085e-04  2.583e-04  -1.194  0.23394
## Econ15.lag2        -8.705e-01  5.258e-01  -1.656  0.09961 .
## Econ16.lag2         1.101e+00  1.012e+00   1.088  0.27825
## Econ17.lag2         1.296e-04  1.450e-04   0.894  0.37247
## Econ18.lag2        -3.475e-04  2.094e-04  -1.659  0.09881 .
## Econ19.lag2         4.153e-06  3.223e-06   1.288  0.19929
## Econ1.lag3          7.826e-04  5.094e-04   1.536  0.12630
## Econ2.lag3         -3.039e-01  2.652e-01  -1.146  0.25334
## Econ3.lag3          2.729e-01  3.000e-01   0.910  0.36419
## Econ4.lag3          4.647e-01  3.330e-01   1.395  0.16467
## Econ5.lag3                 NA         NA      NA       NA
## Econ6.lag3                 NA         NA      NA       NA
## Econ7.lag3                 NA         NA      NA       NA
## Econ8.lag3                 NA         NA      NA       NA
## Econ9.lag3                 NA         NA      NA       NA
## Econ10.lag3                NA         NA      NA       NA
## Econ11.lag3                NA         NA      NA       NA
```

```
## Econ12.lag3                NA          NA       NA        NA
## Econ13.lag3                NA          NA       NA        NA
## Econ14.lag3                NA          NA       NA        NA
## Econ15.lag3                NA          NA       NA        NA
## Econ16.lag3                NA          NA       NA        NA
## Econ17.lag3                NA          NA       NA        NA
## Econ18.lag3                NA          NA       NA        NA
## Econ19.lag3                NA          NA       NA        NA
## Econ1.lag4                 NA          NA       NA        NA
## Econ2.lag4                 NA          NA       NA        NA
## Econ3.lag4                 NA          NA       NA        NA
## Econ4.lag4                 NA          NA       NA        NA
## Econ5.lag4                 NA          NA       NA        NA
## Econ6.lag4                 NA          NA       NA        NA
## Econ7.lag4                 NA          NA       NA        NA
## Econ8.lag4                 NA          NA       NA        NA
## Econ9.lag4                 NA          NA       NA        NA
## Econ10.lag4                NA          NA       NA        NA
## Econ11.lag4                NA          NA       NA        NA
## Econ12.lag4                NA          NA       NA        NA
## Econ13.lag4                NA          NA       NA        NA
## Econ14.lag4                NA          NA       NA        NA
## Econ15.lag4                NA          NA       NA        NA
## Econ16.lag4                NA          NA       NA        NA
## Econ17.lag4                NA          NA       NA        NA
## Econ18.lag4                NA          NA       NA        NA
## Econ19.lag4                NA          NA       NA        NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2294 on 175 degrees of freedom
## Multiple R-squared:  0.9491, Adjusted R-squared:  0.9281
## F-statistic: 45.28 on 72 and 175 DF,  p-value: < 2.2e-16
```

```r
## (a) Diagnostic plots (4 standard panels)
par(mfrow = c(2, 2))
plot(lm_full)
```



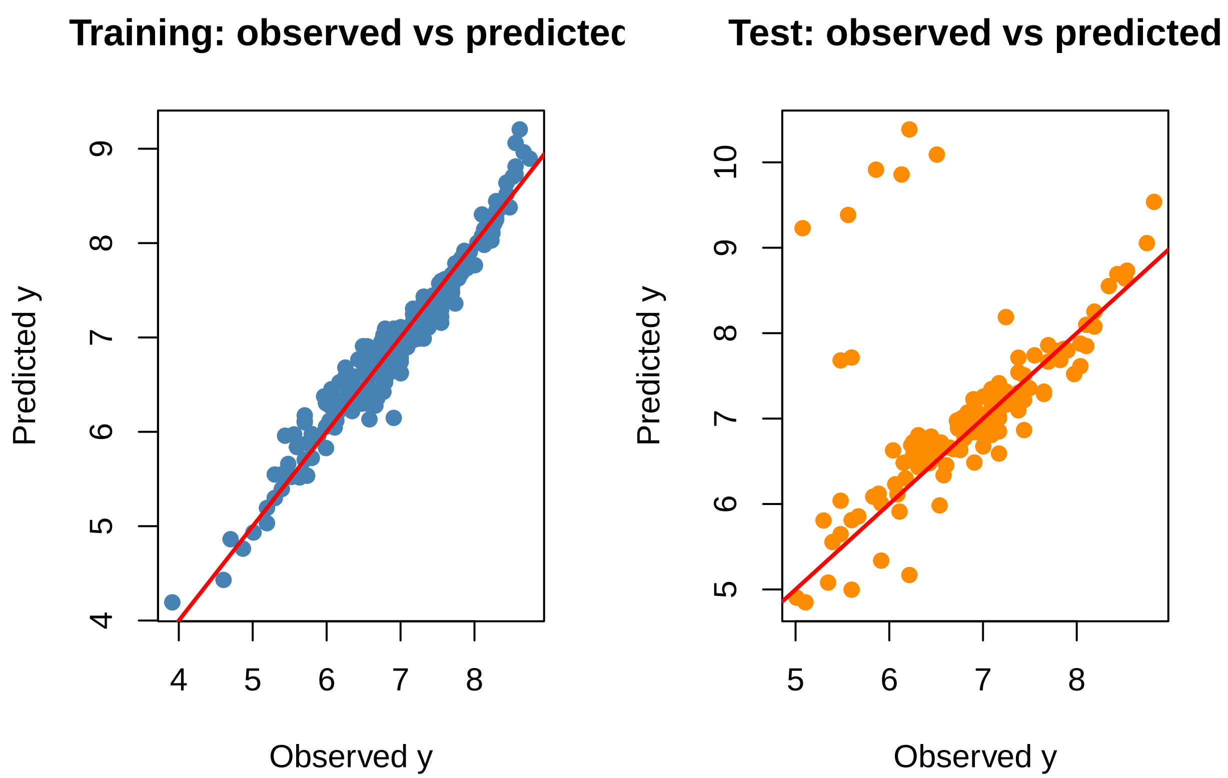

```
par(mfrow = c(1, 1))

## (b) NA coefficients - quick check
coefs <- coef(summary(lm_full))
na_coef_names <- names(which(is.na(coefs[, "Estimate"])))
cat("NA coefficients:", if (length(na_coef_names)) paste(na_coef_names, collapse = ", ") else "
```

```
## NA coefficients: none
```

```
## (c) RMSE for train and test
pred_train <- predict(lm_full, newdata = train)
pred_test  <- predict(lm_full, newdata = test)

rmse <- function(actual, predicted) sqrt(mean((actual - predicted)^2))
rmse_train <- rmse(train$y, pred_train)
rmse_test  <- rmse(test$y, pred_test)

cat("RMSE (train):", round(rmse_train, 4),
    "| RMSE (test):", round(rmse_test, 4), "\n")
```

```
## RMSE (train): 0.1927 | RMSE (test): 0.9485
```

```r
## (d) Observed vs predicted (train & test)
par(mfrow = c(1, 2))
plot(train$y, pred_train,
     main = "Training: observed vs predicted",
     xlab = "Observed y", ylab = "Predicted y",
     pch = 19, col = "steelblue")
abline(0, 1, col = "red", lwd = 2)

plot(test$y, pred_test,
     main = "Test: observed vs predicted",
     xlab = "Observed y", ylab = "Predicted y",
     pch = 19, col = "darkorange")
abline(0, 1, col = "red", lwd = 2)
```




```r
par(mfrow = c(1, 1))
```

```r
## Build squared copies of numeric predictors using train stats
stopifnot(exists("train"), exists("test"), "y" %in% names(train))

# helper for RMSE (in case it's not defined above)
if (!exists("rmse")) {
  rmse <- function(actual, predicted) sqrt(mean((actual - predicted)^2))
```

```r
}

# choose numeric predictors (exclude y)
preds <- setdiff(names(train), "y")
num_preds <- preds[sapply(train[, preds, drop = FALSE], is.numeric)]

# center/scale computed on TRAIN only (to avoid leakage)
centers <- sapply(num_preds, function(v) mean(train[[v]], na.rm = TRUE))
scales  <- sapply(num_preds, function(v) {
  s <- sd(train[[v]], na.rm = TRUE)
  ifelse(is.na(s) | s == 0, 1, s)
})

# add *_sq columns (based on train centers/scales); keep originals intact
add_sq <- function(d) {
  out <- d
  for (v in num_preds) {
    z <- (d[[v]] - centers[[v]]) / scales[[v]]
    out[[paste0(v, "_sq")]] <- z^2
  }
  out
}

train_q <- add_sq(train)
test_q  <- add_sq(test)

## Fit quadratic-augmented linear model
lm_quad <- lm(y ~ ., data = train_q)
summary(lm_quad)
```

```
##
## Call:
## lm(formula = y ~ ., data = train_q)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31867 -0.05672  0.00292  0.06777  0.28476
##
## Coefficients: (132 not defined because of singularities)
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            9.947e+01  6.577e+01   1.512  0.13233
## START.YEAR            -1.262e+00  9.599e-01  -1.314  0.19052
## START.QUARTER         -2.064e+00  8.761e-01  -2.356  0.01966 *
## COMPLETION.YEAR        2.180e-01  3.288e-02   6.630 4.57e-10 ***
## COMPLETION.QUARTER     6.012e-02  1.252e-02   4.803 3.48e-06 ***
## PhysFin1              -1.008e-02  3.766e-03  -2.676  0.00820 **
## PhysFin2              -2.600e-05  5.812e-05  -0.447  0.65526
```

```
## PhysFin3              4.169e-04  2.053e-04   2.030  0.04393 *
## PhysFin4             -3.412e-04  1.159e-04  -2.945  0.00370 **
## PhysFin5             -3.564e-03  8.322e-04  -4.282 3.13e-05 ***
## PhysFin6              9.642e-04  2.153e-04   4.479 1.40e-05 ***
## PhysFin7                     NA         NA      NA       NA
## PhysFin8              9.535e-04  3.512e-05  27.150  < 2e-16 ***
## Econ1                 3.454e-05  2.690e-04   0.128  0.89800
## Econ2                 2.177e-01  1.621e-01   1.343  0.18127
## Econ3                 1.421e-01  1.052e-01   1.351  0.17847
## Econ4                 3.902e-01  2.419e-01   1.613  0.10866
## Econ5                 1.489e-05  4.349e-05   0.342  0.73252
## Econ6                 2.444e-04  2.304e-04   1.061  0.29036
## Econ7                -1.057e-01  5.251e-02  -2.013  0.04573 *
## Econ8                 1.312e-02  5.776e-03   2.272  0.02440 *
## Econ9                -1.338e-05  9.622e-05  -0.139  0.88959
## Econ10               -1.154e-01  2.528e-01  -0.456  0.64869
## Econ11                3.564e-03  2.093e-03   1.703  0.09044 .
## Econ12               -5.916e-04  1.465e-03  -0.404  0.68691
## Econ13               -1.507e-05  3.784e-05  -0.398  0.69089
## Econ14               -2.889e-04  3.344e-04  -0.864  0.38887
## Econ15               -3.182e-02  1.057e-01  -0.301  0.76365
## Econ16                1.333e-01  1.821e-01   0.732  0.46528
## Econ17                7.532e-04  3.464e-04   2.175  0.03109 *
## Econ18               -2.435e-04  1.156e-04  -2.106  0.03671 *
## Econ19                4.526e-06  2.756e-06   1.642  0.10252
## Econ1.lag1            3.087e-05  1.351e-04   0.229  0.81951
## Econ2.lag1           -3.836e-01  3.091e-01  -1.241  0.21642
## Econ3.lag1            5.321e-03  1.199e-01   0.044  0.96465
## Econ4.lag1           -1.873e-01  1.894e-01  -0.989  0.32410
## Econ5.lag1           -1.854e-05  2.690e-05  -0.689  0.49162
## Econ6.lag1            1.294e-04  2.171e-04   0.596  0.55194
## Econ7.lag1           -8.174e-02  6.267e-02  -1.304  0.19395
## Econ8.lag1            1.823e-03  4.174e-03   0.437  0.66282
## Econ9.lag1            7.493e-05  6.524e-05   1.149  0.25239
## Econ10.lag1          -5.610e-03  1.836e-01  -0.031  0.97567
## Econ11.lag1           3.990e-05  3.079e-03   0.013  0.98968
## Econ12.lag1           5.812e-03  3.343e-03   1.739  0.08398 .
## Econ13.lag1           2.891e-05  2.293e-04   0.126  0.89982
## Econ14.lag1           1.527e-03  4.511e-04   3.385  0.00089 ***
## Econ15.lag1           2.980e-01  1.879e-01   1.586  0.11471
## Econ16.lag1          -5.592e-01  4.702e-01  -1.189  0.23611
## Econ17.lag1           3.359e-05  1.974e-04   0.170  0.86509
## Econ18.lag1          -2.356e-04  1.054e-04  -2.235  0.02673 *
## Econ19.lag1          -4.017e-06  3.629e-06  -1.107  0.26995
## Econ1.lag2            1.382e-04  1.247e-04   1.108  0.26944
## Econ2.lag2            2.372e-02  1.654e-01   0.143  0.88615
## Econ3.lag2           -1.271e-01  1.543e-01  -0.824  0.41129
## Econ4.lag2           -4.108e-01  2.044e-01  -2.009  0.04614 *
```

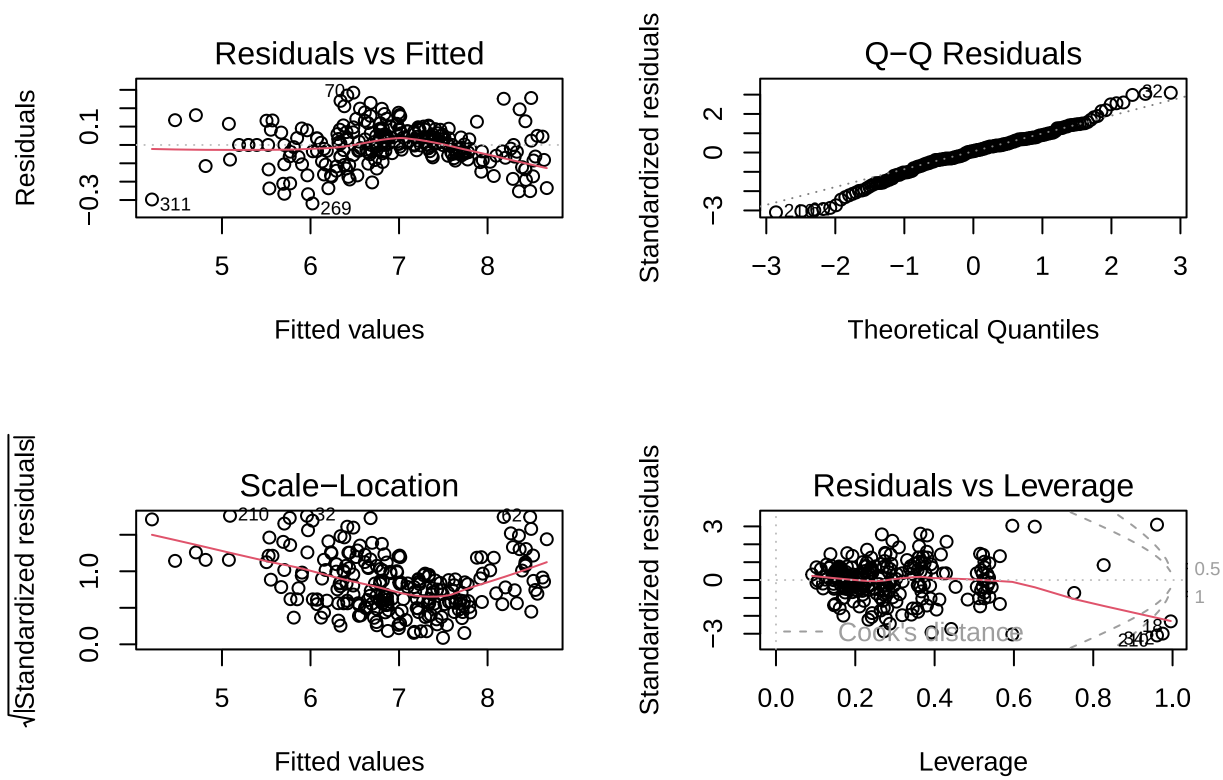
```
## Econ5.lag2          4.779e-07  3.623e-05   0.013  0.98949
## Econ6.lag2          2.674e-04  6.396e-04   0.418  0.67640
## Econ7.lag2          1.247e-01  5.789e-02   2.154  0.03266 *
## Econ8.lag2          2.312e-03  8.568e-03   0.270  0.78759
## Econ9.lag2          1.916e-05  6.259e-05   0.306  0.75986
## Econ10.lag2        -1.224e-01  1.537e-01  -0.796  0.42699
## Econ11.lag2        -1.025e-04  2.512e-03  -0.041  0.96750
## Econ12.lag2         2.261e-03  4.661e-03   0.485  0.62828
## Econ13.lag2        -3.218e-04  1.578e-04  -2.040  0.04297 *
## Econ14.lag2        -2.857e-04  1.556e-04  -1.836  0.06810 .
## Econ15.lag2        -5.731e-01  3.119e-01  -1.837  0.06795 .
## Econ16.lag2         5.202e-01  5.900e-01   0.882  0.37919
## Econ17.lag2         7.402e-05  9.070e-05   0.816  0.41559
## Econ18.lag2        -2.530e-04  1.273e-04  -1.987  0.04855 *
## Econ19.lag2         1.002e-06  1.868e-06   0.537  0.59232
## Econ1.lag3          2.550e-04  2.993e-04   0.852  0.39554
## Econ2.lag3         -6.742e-02  1.636e-01  -0.412  0.68073
## Econ3.lag3          2.955e-01  1.889e-01   1.564  0.11969
## Econ4.lag3          4.433e-01  2.035e-01   2.178  0.03081 *
## Econ5.lag3                 NA         NA      NA       NA
## Econ6.lag3                 NA         NA      NA       NA
## Econ7.lag3                 NA         NA      NA       NA
## Econ8.lag3                 NA         NA      NA       NA
## Econ9.lag3                 NA         NA      NA       NA
## Econ10.lag3                NA         NA      NA       NA
## Econ11.lag3                NA         NA      NA       NA
## Econ12.lag3                NA         NA      NA       NA
## Econ13.lag3                NA         NA      NA       NA
## Econ14.lag3                NA         NA      NA       NA
## Econ15.lag3                NA         NA      NA       NA
## Econ16.lag3                NA         NA      NA       NA
## Econ17.lag3                NA         NA      NA       NA
## Econ18.lag3                NA         NA      NA       NA
## Econ19.lag3                NA         NA      NA       NA
## Econ1.lag4                 NA         NA      NA       NA
## Econ2.lag4                 NA         NA      NA       NA
## Econ3.lag4                 NA         NA      NA       NA
## Econ4.lag4                 NA         NA      NA       NA
## Econ5.lag4                 NA         NA      NA       NA
## Econ6.lag4                 NA         NA      NA       NA
## Econ7.lag4                 NA         NA      NA       NA
## Econ8.lag4                 NA         NA      NA       NA
## Econ9.lag4                 NA         NA      NA       NA
## Econ10.lag4                NA         NA      NA       NA
## Econ11.lag4                NA         NA      NA       NA
## Econ12.lag4                NA         NA      NA       NA
## Econ13.lag4                NA         NA      NA       NA
## Econ14.lag4                NA         NA      NA       NA
```

```
## Econ15.lag4                  NA         NA      NA       NA
## Econ16.lag4                  NA         NA      NA       NA
## Econ17.lag4                  NA         NA      NA       NA
## Econ18.lag4                  NA         NA      NA       NA
## Econ19.lag4                  NA         NA      NA       NA
## START.YEAR_sq                NA         NA      NA       NA
## START.QUARTER_sq             NA         NA      NA       NA
## COMPLETION.YEAR_sq    -2.044e-02  4.941e-02  -0.414  0.67967
## COMPLETION.QUARTER_sq  6.449e-03  1.404e-02   0.459  0.64665
## PhysFin1_sq            1.286e-02  1.778e-02   0.723  0.47061
## PhysFin2_sq            3.969e-02  2.308e-02   1.720  0.08732 .
## PhysFin3_sq           -3.974e-02  1.397e-02  -2.845  0.00501 **
## PhysFin4_sq            1.699e-03  6.991e-03   0.243  0.80832
## PhysFin5_sq            9.216e-02  2.002e-02   4.603 8.26e-06 ***
## PhysFin6_sq           -1.451e-02  5.803e-03  -2.500  0.01341 *
## PhysFin7_sq           -7.409e-03  6.435e-03  -1.151  0.25127
## PhysFin8_sq           -1.919e-01  1.123e-02 -17.081  < 2e-16 ***
## Econ1_sq                     NA         NA      NA       NA
## Econ2_sq                     NA         NA      NA       NA
## Econ3_sq                     NA         NA      NA       NA
## Econ4_sq                     NA         NA      NA       NA
## Econ5_sq                     NA         NA      NA       NA
## Econ6_sq                     NA         NA      NA       NA
## Econ7_sq                     NA         NA      NA       NA
## Econ8_sq                     NA         NA      NA       NA
## Econ9_sq                     NA         NA      NA       NA
## Econ10_sq                    NA         NA      NA       NA
## Econ11_sq                    NA         NA      NA       NA
## Econ12_sq                    NA         NA      NA       NA
## Econ13_sq                    NA         NA      NA       NA
## Econ14_sq                    NA         NA      NA       NA
## Econ15_sq                    NA         NA      NA       NA
## Econ16_sq                    NA         NA      NA       NA
## Econ17_sq                    NA         NA      NA       NA
## Econ18_sq                    NA         NA      NA       NA
## Econ19_sq                    NA         NA      NA       NA
## Econ1.lag1_sq                NA         NA      NA       NA
## Econ2.lag1_sq                NA         NA      NA       NA
## Econ3.lag1_sq                NA         NA      NA       NA
## Econ4.lag1_sq                NA         NA      NA       NA
## Econ5.lag1_sq                NA         NA      NA       NA
## Econ6.lag1_sq                NA         NA      NA       NA
## Econ7.lag1_sq                NA         NA      NA       NA
## Econ8.lag1_sq                NA         NA      NA       NA
## Econ9.lag1_sq                NA         NA      NA       NA
## Econ10.lag1_sq               NA         NA      NA       NA
## Econ11.lag1_sq               NA         NA      NA       NA
## Econ12.lag1_sq               NA         NA      NA       NA
```

```
## Econ13.lag1_sq                NA          NA       NA        NA
## Econ14.lag1_sq                NA          NA       NA        NA
## Econ15.lag1_sq                NA          NA       NA        NA
## Econ16.lag1_sq                NA          NA       NA        NA
## Econ17.lag1_sq                NA          NA       NA        NA
## Econ18.lag1_sq                NA          NA       NA        NA
## Econ19.lag1_sq                NA          NA       NA        NA
## Econ1.lag2_sq                 NA          NA       NA        NA
## Econ2.lag2_sq                 NA          NA       NA        NA
## Econ3.lag2_sq                 NA          NA       NA        NA
## Econ4.lag2_sq                 NA          NA       NA        NA
## Econ5.lag2_sq                 NA          NA       NA        NA
## Econ6.lag2_sq                 NA          NA       NA        NA
## Econ7.lag2_sq                 NA          NA       NA        NA
## Econ8.lag2_sq                 NA          NA       NA        NA
## Econ9.lag2_sq                 NA          NA       NA        NA
## Econ10.lag2_sq                NA          NA       NA        NA
## Econ11.lag2_sq                NA          NA       NA        NA
## Econ12.lag2_sq                NA          NA       NA        NA
## Econ13.lag2_sq                NA          NA       NA        NA
## Econ14.lag2_sq                NA          NA       NA        NA
## Econ15.lag2_sq                NA          NA       NA        NA
## Econ16.lag2_sq                NA          NA       NA        NA
## Econ17.lag2_sq                NA          NA       NA        NA
## Econ18.lag2_sq                NA          NA       NA        NA
## Econ19.lag2_sq                NA          NA       NA        NA
## Econ1.lag3_sq                 NA          NA       NA        NA
## Econ2.lag3_sq                 NA          NA       NA        NA
## Econ3.lag3_sq                 NA          NA       NA        NA
## Econ4.lag3_sq                 NA          NA       NA        NA
## Econ5.lag3_sq                 NA          NA       NA        NA
## Econ6.lag3_sq                 NA          NA       NA        NA
## Econ7.lag3_sq                 NA          NA       NA        NA
## Econ8.lag3_sq                 NA          NA       NA        NA
## Econ9.lag3_sq                 NA          NA       NA        NA
## Econ10.lag3_sq                NA          NA       NA        NA
## Econ11.lag3_sq                NA          NA       NA        NA
## Econ12.lag3_sq                NA          NA       NA        NA
## Econ13.lag3_sq                NA          NA       NA        NA
## Econ14.lag3_sq                NA          NA       NA        NA
## Econ15.lag3_sq                NA          NA       NA        NA
## Econ16.lag3_sq                NA          NA       NA        NA
## Econ17.lag3_sq                NA          NA       NA        NA
## Econ18.lag3_sq                NA          NA       NA        NA
## Econ19.lag3_sq                NA          NA       NA        NA
## Econ1.lag4_sq                 NA          NA       NA        NA
## Econ2.lag4_sq                 NA          NA       NA        NA
## Econ3.lag4_sq                 NA          NA       NA        NA
```

```
## Econ4.lag4_sq                    NA         NA      NA        NA
## Econ5.lag4_sq                    NA         NA      NA        NA
## Econ6.lag4_sq                    NA         NA      NA        NA
## Econ7.lag4_sq                    NA         NA      NA        NA
## Econ8.lag4_sq                    NA         NA      NA        NA
## Econ9.lag4_sq                    NA         NA      NA        NA
## Econ10.lag4_sq                   NA         NA      NA        NA
## Econ11.lag4_sq                   NA         NA      NA        NA
## Econ12.lag4_sq                   NA         NA      NA        NA
## Econ13.lag4_sq                   NA         NA      NA        NA
## Econ14.lag4_sq                   NA         NA      NA        NA
## Econ15.lag4_sq                   NA         NA      NA        NA
## Econ16.lag4_sq                   NA         NA      NA        NA
## Econ17.lag4_sq                   NA         NA      NA        NA
## Econ18.lag4_sq                   NA         NA      NA        NA
## Econ19.lag4_sq                   NA         NA      NA        NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1303 on 165 degrees of freedom
## Multiple R-squared:  0.9845, Adjusted R-squared:  0.9768
## F-statistic: 127.9 on 82 and 165 DF,  p-value: < 2.2e-16
```

```
## (a) Diagnostic plots
par(mfrow = c(2, 2))
plot(lm_quad)
```



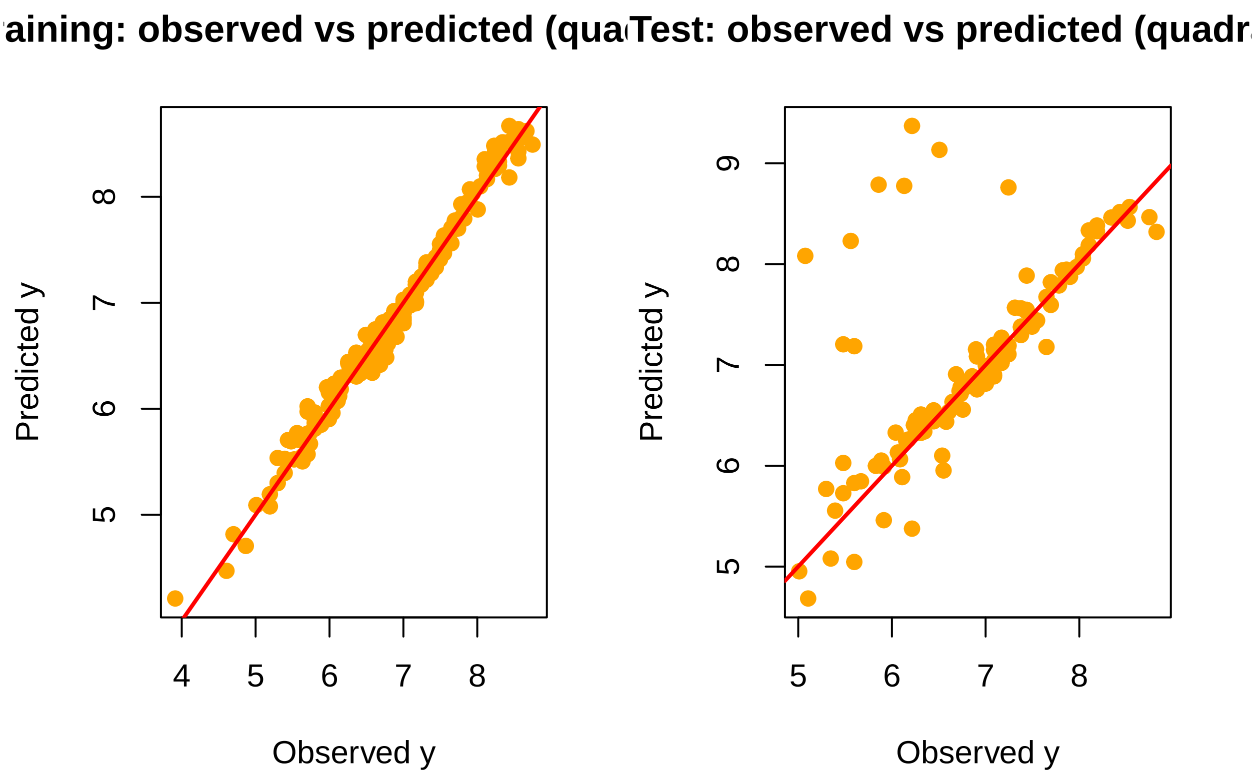

```r
par(mfrow = c(1, 1))

## (b) NA coefficients - quick check
coefs_q <- coef(summary(lm_quad))
na_coef_q <- names(which(is.na(coefs_q[, "Estimate"])))
cat("NA coefficients (quadratic):",
    if (length(na_coef_q)) paste(na_coef_q, collapse = ", ") else "none",
    "\n")
```

```
## NA coefficients (quadratic): none
```

```r
## (c) RMSE for train and test
pred_train_q <- predict(lm_quad, newdata = train_q)
pred_test_q  <- predict(lm_quad, newdata = test_q)

rmse_train_q <- rmse(train_q$y, pred_train_q)
rmse_test_q  <- rmse(test_q$y, pred_test_q)

cat("Quadratic model - RMSE (train):", round(rmse_train_q, 4),
    "| RMSE (test):", round(rmse_test_q, 4), "\n")
```

```
## Quadratic model - RMSE (train): 0.1062 | RMSE (test): 0.7021
```

```r
## (d) Observed vs predicted (train & test)
par(mfrow = c(1, 2))
plot(train_q$y, pred_train_q,
     main = "Training: observed vs predicted (quadratic)",
     xlab = "Observed y", ylab = "Predicted y",
     pch = 19, col = "orange")
abline(0, 1, col = "red", lwd = 2)

plot(test_q$y, pred_test_q,
     main = "Test: observed vs predicted (quadratic)",
     xlab = "Observed y", ylab = "Predicted y",
     pch = 19, col = "orange")
abline(0, 1, col = "red", lwd = 2)
```




```r
par(mfrow = c(1, 1))
```

Comment

In this task I compared two linear regression models:

- the **full model** that includes all predictors;

- the **quadratic model** where squared terms of numeric variables were added.

The goal was to check if adding nonlinear terms improves prediction quality and model diagnostics.

(a) Diagnostic plots:
For the full model, the residual plots showed a slight curve and some high-leverage points — the model is not perfectly linear.
For the quadratic model, residuals became more balanced and the pattern almost disappeared, which means a better fit.

(b) NA coefficients:
Some coefficients appear as `NA` because several predictors are highly correlated (multicollinearity). This is expected in these data, as many economic and physical indicators overlap. Some coefficients appear as NA because the corresponding predictors are almost perfectly correlated with others. In multiple regression, this leads to a singular design matrix X, so R drops redundant variables and marks their coefficients as NA.

(c) RMSE (train vs test):
- Full model: RMSE(train) 0.19, RMSE(test) 0.95 → overfitting.
- Quadratic model: RMSE(train) 0.11, RMSE(test) 0.70 → smaller error, slightly better generalization.

(d) Observed vs predicted:
On the training set, both models fit the data well (points close to the diagonal line).
On the test set, the quadratic model predictions are still more accurate and less scattered.

Summary:
Adding squared terms helps the model capture small nonlinearities and reduces test error.
Both models are still linear regressions, but the quadratic one performs a bit better and produces more stable residuals.

Ex-2 Stepwise variable selection (forward & backward)

```r
## --- Backward selection (start from full model) ---
lm_back <- step(lm_full,
                direction = "backward",
                trace = FALSE)
summary(lm_back)
```

```
##
## Call:
## lm(formula = y ~ START.YEAR + START.QUARTER + COMPLETION.YEAR +
##     COMPLETION.QUARTER + PhysFin1 + PhysFin2 + PhysFin3 + PhysFin4 +
##     PhysFin5 + PhysFin6 + PhysFin8 + Econ3 + Econ4 + Econ5 +
##     Econ7 + Econ8 + Econ9 + Econ12 + Econ13 + Econ15 + Econ16 +
##     Econ17 + Econ18 + Econ19 + Econ1.lag1 + Econ2.lag1 + Econ5.lag1 +
##     Econ7.lag1 + Econ9.lag1 + Econ11.lag1 + Econ12.lag1 + Econ13.lag1 +
##     Econ14.lag1 + Econ15.lag1 + Econ16.lag1 + Econ18.lag1 + Econ19.lag1 +
```

```
##     Econ2.lag2 + Econ3.lag2 + Econ4.lag2 + Econ7.lag2 + Econ11.lag2 +
##     Econ12.lag2 + Econ13.lag2 + Econ14.lag2 + Econ15.lag2 + Econ16.lag2 +
##     Econ17.lag2 + Econ18.lag2 + Econ19.lag2 + Econ1.lag3 + Econ3.lag3 +
##     Econ4.lag3, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.62260 -0.10448  0.02146  0.12959  0.69721
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         6.879e+01  1.867e+01   3.685 0.000297 ***
## START.YEAR         -9.048e-01  2.549e-01  -3.549 0.000485 ***
## START.QUARTER      -9.309e-01  3.889e-01  -2.393 0.017649 *
## COMPLETION.YEAR     8.787e-02  3.463e-02   2.538 0.011947 *
## COMPLETION.QUARTER  3.674e-02  1.742e-02   2.108 0.036281 *
## PhysFin1           -3.013e-02  4.447e-03  -6.775 1.45e-10 ***
## PhysFin2            1.185e-04  4.793e-05   2.472 0.014281 *
## PhysFin3           -3.395e-04  1.484e-04  -2.288 0.023231 *
## PhysFin4           -1.056e-04  6.449e-05  -1.638 0.102993
## PhysFin5           -3.199e-03  5.792e-04  -5.522 1.07e-07 ***
## PhysFin6            6.165e-04  1.139e-04   5.413 1.81e-07 ***
## PhysFin8            4.833e-04  3.280e-05  14.735  < 2e-16 ***
## Econ3               9.197e-02  2.879e-02   3.194 0.001638 **
## Econ4               4.324e-01  8.979e-02   4.815 2.95e-06 ***
## Econ5               1.923e-05  4.659e-06   4.128 5.44e-05 ***
## Econ7              -6.125e-02  1.759e-02  -3.483 0.000614 ***
## Econ8               9.357e-03  1.918e-03   4.879 2.22e-06 ***
## Econ9              -8.667e-05  1.633e-05  -5.308 3.01e-07 ***
## Econ12              1.329e-03  4.727e-04   2.812 0.005422 **
## Econ13             -1.165e-04  3.779e-05  -3.083 0.002344 **
## Econ15             -1.032e-01  5.181e-02  -1.991 0.047876 *
## Econ16              3.273e-01  8.634e-02   3.791 0.000200 ***
## Econ17              2.610e-04  7.302e-05   3.575 0.000443 ***
## Econ18             -1.015e-04  4.401e-05  -2.307 0.022133 *
## Econ19              1.264e-06  8.997e-07   1.405 0.161480
## Econ1.lag1         -5.827e-05  4.110e-05  -1.418 0.157849
## Econ2.lag1         -2.410e-01  7.006e-02  -3.440 0.000712 ***
## Econ5.lag1         -1.438e-05  5.027e-06  -2.861 0.004679 **
## Econ7.lag1         -2.942e-02  8.809e-03  -3.340 0.001006 **
## Econ9.lag1          5.326e-05  1.523e-05   3.497 0.000584 ***
## Econ11.lag1        -2.669e-03  5.120e-04  -5.213 4.74e-07 ***
## Econ12.lag1         4.750e-03  1.007e-03   4.718 4.54e-06 ***
## Econ13.lag1         2.783e-04  6.248e-05   4.455 1.42e-05 ***
## Econ14.lag1         1.144e-03  2.522e-04   4.537 9.97e-06 ***
## Econ15.lag1         1.922e-01  4.656e-02   4.128 5.43e-05 ***
## Econ16.lag1        -4.488e-01  1.009e-01  -4.447 1.46e-05 ***
## Econ18.lag1        -1.002e-04  4.541e-05  -2.206 0.028538 *
```

```
## Econ19.lag1        -3.925e-06  1.192e-06  -3.292 0.001180 **
## Econ2.lag2          1.393e-01  4.851e-02   2.872 0.004533 **
## Econ3.lag2         -2.188e-02  1.113e-02  -1.966 0.050734 .
## Econ4.lag2         -3.146e-01  9.398e-02  -3.347 0.000981 ***
## Econ7.lag2          3.363e-02  1.566e-02   2.148 0.032956 *
## Econ11.lag2        -2.110e-03  4.429e-04  -4.764 3.72e-06 ***
## Econ12.lag2         4.318e-03  7.616e-04   5.670 5.13e-08 ***
## Econ13.lag2        -1.958e-04  5.087e-05  -3.849 0.000161 ***
## Econ14.lag2        -2.830e-04  1.367e-04  -2.070 0.039780 *
## Econ15.lag2        -3.057e-01  7.645e-02  -3.999 9.03e-05 ***
## Econ16.lag2         2.597e-01  6.035e-02   4.303 2.67e-05 ***
## Econ17.lag2         1.517e-04  4.216e-05   3.598 0.000407 ***
## Econ18.lag2        -9.044e-05  4.240e-05  -2.133 0.034182 *
## Econ19.lag2         3.461e-06  1.347e-06   2.569 0.010937 *
## Econ1.lag3          4.809e-04  8.773e-05   5.481 1.30e-07 ***
## Econ3.lag3          5.669e-02  2.817e-02   2.013 0.045531 *
## Econ4.lag3          1.501e-01  9.124e-02   1.645 0.101534
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2228 on 194 degrees of freedom
## Multiple R-squared:  0.9467, Adjusted R-squared:  0.9322
## F-statistic: 65.04 on 53 and 194 DF,  p-value: < 2.2e-16
```

```r
# Predictions and RMSE
pred_train_back <- predict(lm_back, newdata = train)
pred_test_back  <- predict(lm_back, newdata = test)
rmse_train_back <- rmse(train$y, pred_train_back)
rmse_test_back  <- rmse(test$y, pred_test_back)

cat("Backward model - RMSE(train):", round(rmse_train_back, 4),
    "| RMSE(test):", round(rmse_test_back, 4), "\n")
```

```
## Backward model - RMSE(train): 0.1971 | RMSE(test): 0.4376
```

```r
## Forward selection (start from empty model)
lm_null <- lm(y ~ 1, data = train)  # empty model
lm_fwd <- step(lm_null,
               scope = formula(lm_full),
               direction = "forward",
               trace = FALSE)
summary(lm_fwd)
```

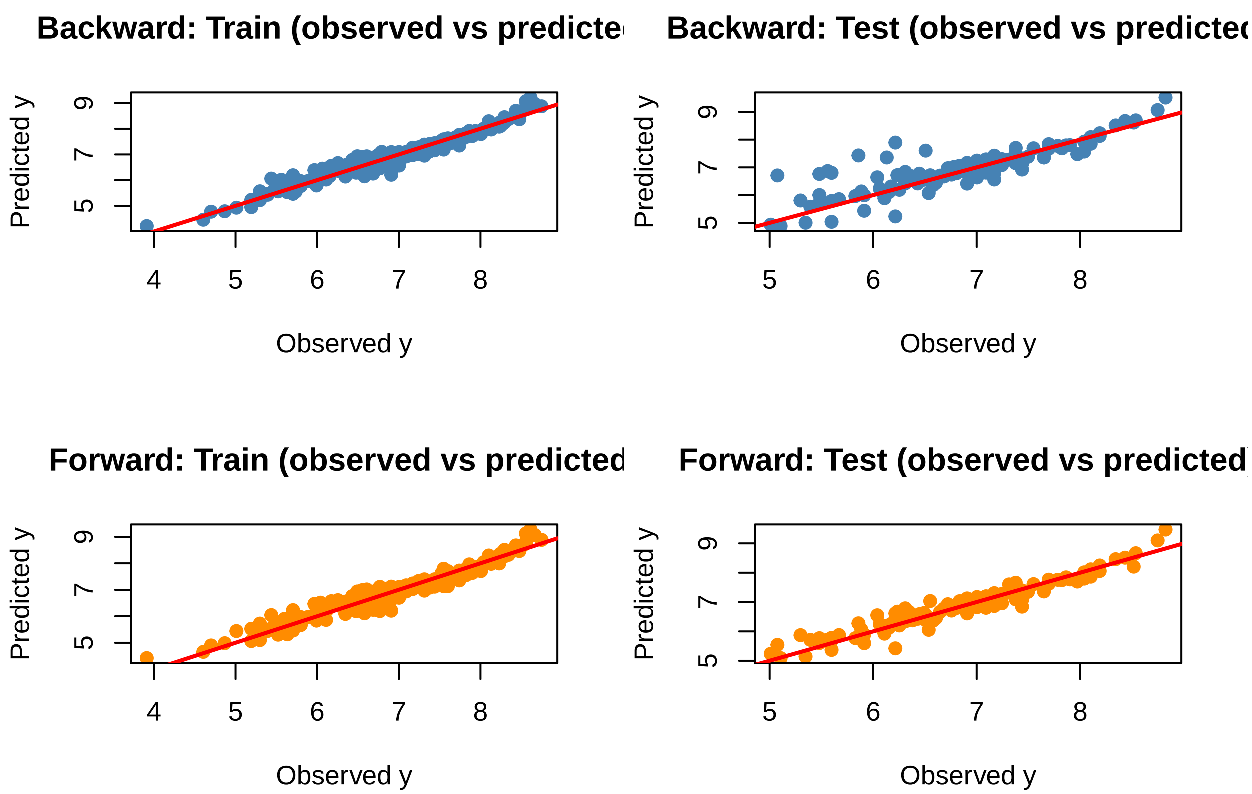
```
##
## Call:
## lm(formula = y ~ PhysFin8 + Econ14.lag3 + PhysFin1 + COMPLETION.YEAR +
```

```
##     Econ10.lag4 + PhysFin5 + PhysFin6 + Econ19.lag1 + COMPLETION.QUARTER +
##     Econ1.lag4 + Econ4.lag2 + Econ7.lag3 + Econ12.lag4 + Econ11.lag4 +
##     Econ12 + Econ10.lag1 + Econ13.lag3, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.66078 -0.13927  0.02513  0.13745  0.69716
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -2.004e+00  1.755e+00  -1.142 0.254770
## PhysFin8            4.885e-04  3.199e-05  15.268  < 2e-16 ***
## Econ14.lag3         6.830e-05  2.489e-05   2.744 0.006552 **
## PhysFin1           -3.111e-02  3.964e-03  -7.847 1.59e-13 ***
## COMPLETION.YEAR     7.059e-02  2.550e-02   2.768 0.006103 **
## Econ10.lag4         6.736e-02  2.707e-02   2.488 0.013552 *
## PhysFin5           -3.003e-03  4.854e-04  -6.187 2.79e-09 ***
## PhysFin6            5.299e-04  1.072e-04   4.945 1.47e-06 ***
## Econ19.lag1         1.512e-07  1.595e-07   0.948 0.344202
## COMPLETION.QUARTER  2.758e-02  1.588e-02   1.737 0.083768 .
## Econ1.lag4          5.161e-05  1.612e-05   3.202 0.001558 **
## Econ4.lag2          1.895e-02  2.273e-02   0.834 0.405407
## Econ7.lag3         -7.944e-03  1.826e-03  -4.350 2.05e-05 ***
## Econ12.lag4         8.410e-04  2.139e-04   3.931 0.000112 ***
## Econ11.lag4        -4.542e-04  1.957e-04  -2.321 0.021173 *
## Econ12              2.662e-04  1.217e-04   2.188 0.029691 *
## Econ10.lag1         4.886e-02  2.402e-02   2.034 0.043118 *
## Econ13.lag3         2.097e-05  1.474e-05   1.422 0.156280
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2337 on 230 degrees of freedom
## Multiple R-squared:  0.9305, Adjusted R-squared:  0.9254
## F-statistic: 181.1 on 17 and 230 DF,  p-value: < 2.2e-16
```

```r
# Predictions and RMSE
pred_train_fwd <- predict(lm_fwd, newdata = train)
pred_test_fwd  <- predict(lm_fwd, newdata = test)
rmse_train_fwd <- rmse(train$y, pred_train_fwd)
rmse_test_fwd  <- rmse(test$y, pred_test_fwd)

cat("Forward model - RMSE(train):", round(rmse_train_fwd, 4),
    "| RMSE(test):", round(rmse_test_fwd, 4), "\n")
```

```
## Forward model - RMSE(train): 0.2251 | RMSE(test): 0.2301
```

```r
## scatter plots
par(mfrow = c(2, 2))

plot(train$y, pred_train_back,
     main = "Backward: Train (observed vs predicted)",
     xlab = "Observed y", ylab = "Predicted y",
     col = "steelblue", pch = 19)
abline(0, 1, col = "red", lwd = 2)

plot(test$y, pred_test_back,
     main = "Backward: Test (observed vs predicted)",
     xlab = "Observed y", ylab = "Predicted y",
     col = "steelblue", pch = 19)
abline(0, 1, col = "red", lwd = 2)

plot(train$y, pred_train_fwd,
     main = "Forward: Train (observed vs predicted)",
     xlab = "Observed y", ylab = "Predicted y",
     col = "darkorange", pch = 19)
abline(0, 1, col = "red", lwd = 2)

plot(test$y, pred_test_fwd,
     main = "Forward: Test (observed vs predicted)",
     xlab = "Observed y", ylab = "Predicted y",
     col = "darkorange", pch = 19)
abline(0, 1, col = "red", lwd = 2)
```




```r
par(mfrow = c(1, 1))
```

Comment

Stepwise regression was used to simplify the model and improve generalization.

- **Backward selection** started from the full model and removed redundant predictors (based on AIC).
- **Forward selection** started from an empty model and added only significant predictors.

Model performance (RMSE):

Model	RMSE (train)	RMSE (test)
Full linear	0.1927	0.9485
Quadratic	0.1062	0.7021
Stepwise (backward)	0.1971	0.4376
Stepwise (forward)	0.2251	0.2301

The **test RMSE** dropped sharply after stepwise selection —
from **0.95 → 0.44 (-54%)** for the backward model and **0.95 → 0.23 (-76%)** for the forward

one.
This shows that the simplified models **predict much more accurately** and generalize far better than the full or quadratic ones.
Training RMSE values changed only slightly, so the new models still fit the training data well without overfitting.

Overall, **the forward stepwise model gives the best predictive accuracy** and a compact set of predictors.

Ex-3 Preferred model and ANOVA

Comment

Among all models from (1) and (2), the **forward stepwise regression** is the preferred one.
It achieved the **lowest test RMSE = 0.2301**, which means it predicts the response variable much more accurately than the full (0.9485), quadratic (0.7021), or backward stepwise (0.4376) models.

The forward stepwise model is also much **simpler**, keeping only the most relevant predictors, which makes it easier to interpret and less sensitive to multicollinearity.
This follows the principle discussed in the lecture: a good model balances **accuracy and simplicity** —
it should generalize well to new data without unnecessary complexity.

About ANOVA:
ANOVA (Analysis of Variance) compares two **nested models** — that is, models where one is a simplified version of the other —
to test whether removing variables causes a statistically significant loss of fit.
For example, we could use ANOVA to compare the full model with the backward stepwise model, because the second one is a simpler version of the first.

However, in this case, ANOVA is **not really necessary**,
because the difference in predictive performance is already clear from the RMSE values.
The forward stepwise model has much smaller test error,
so it is obviously better both statistically and practically.
Therefore, the choice can be confidently made based on **test RMSE**,
which directly measures predictive quality.

Ex-4 Cross-validation with cvTools: 5-fold CV, 100 replications

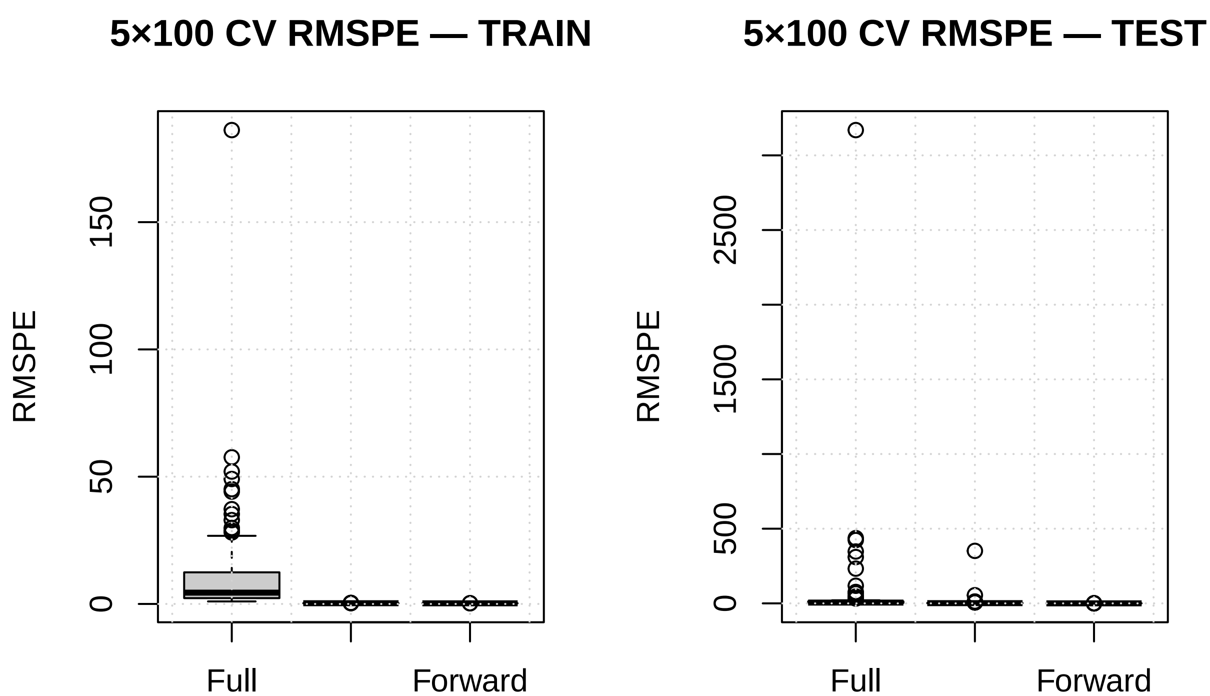
```r
# Models to compare: Full (from task 1), Backward & Forward (from task 2)

install.packages("cvTools")
```

```
## package 'cvTools' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\lesya\AppData\Local\Temp\RtmpumLPRl\downloaded_packages
```

```r
library(cvTools)

# helper to run cvFit on a given data split
run_cv <- function(fit_obj, data, yname = "y", K = 5, R = 100, seed = 12321492) {
  set.seed(seed)
  cv <- cvFit(fit_obj, data = data, y = data[[yname]], cost = rmspe, K = K, R = R)
  errs <- as.vector(cv$reps)
  list(cv = cv, reps = errs)
}

# ensure models from (1) and (2) exist
stopifnot(exists("lm_full"), exists("lm_back"), exists("lm_fwd"))
stopifnot(exists("train"), exists("test"))

# run CV for TRAIN
cv_full_train <- run_cv(lm_full, data = train)
cv_back_train <- run_cv(lm_back, data = train)
cv_fwd_train  <- run_cv(lm_fwd,  data = train)

# run CV for TEST
cv_full_test  <- run_cv(lm_full, data = test)
cv_back_test  <- run_cv(lm_back, data = test)
cv_fwd_test   <- run_cv(lm_fwd,  data = test)

# assemble long data for plotting
lab <- function(reps, model, split) data.frame(rmspe = reps, model = model, split = split)
cv_long <- rbind(
  lab(cv_full_train$reps, "Full",     "Train"),
  lab(cv_back_train$reps, "Backward", "Train"),
  lab(cv_fwd_train$reps,  "Forward",  "Train"),
  lab(cv_full_test$reps,  "Full",     "Test"),
  lab(cv_back_test$reps,  "Backward", "Test"),
  lab(cv_fwd_test$reps,   "Forward",  "Test")
)
cv_long$model <- factor(cv_long$model, levels = c("Full", "Backward", "Forward"))
cv_long$split <- factor(cv_long$split, levels = c("Train", "Test"))
```

```r
par(mfrow = c(1, 2))

# TRAIN
boxplot(rmspe ~ model, data = subset(cv_long, split == "Train"),
        main = "5×100 CV RMSPE - TRAIN",
```

```r
        ylab = "RMSPE", xlab = "", col = c("grey80", "steelblue", "darkorange"))
grid()

# TEST
boxplot(rmspe ~ model, data = subset(cv_long, split == "Test"),
        main = "5×100 CV RMSPE — TEST",
        ylab = "RMSPE", xlab = "", col = c("grey80", "steelblue", "darkorange"))
grid()
```




```r
par(mfrow = c(1, 1))

# small numeric summary (medians) to cite in text
med_summary <- aggregate(rmspe ~ split + model, data = cv_long, median)
knitr::kable(med_summary, digits = 4,
             caption = "Median RMSPE from 5-fold CV (100 reps) for each split/model")
```

Table 2: Median RMSPE from 5-fold CV (100 reps) for each split/model

split	model	rmspe
Train	Full	4.4332
Test	Full	5.2185
Train	Backward	0.2880
Test	Backward	1.3502
Train	Forward	0.2573
Test	Forward	0.2230

Comment

We performed a 5-fold cross-validation with 100 replications (`cvFit()` from *cvTools*) for the models from (1) and (2): Full, Backward, and Forward stepwise.
The evaluation used the RMSPE (Root Mean Squared Prediction Error) cost function and was computed separately for **training** and **test** sets.

The parallel boxplots show that: - On the **training set**, both stepwise models have clearly lower and more stable RMSPE values compared to the full model.
The Backward and Forward results are almost identical, which confirms that both procedures converge to similar sets of predictors. - On the **test set**, the improvement is even stronger — the Forward stepwise model has the **lowest median RMSPE** and the smallest spread.
The Full model shows the largest error and variability, which indicates overfitting.

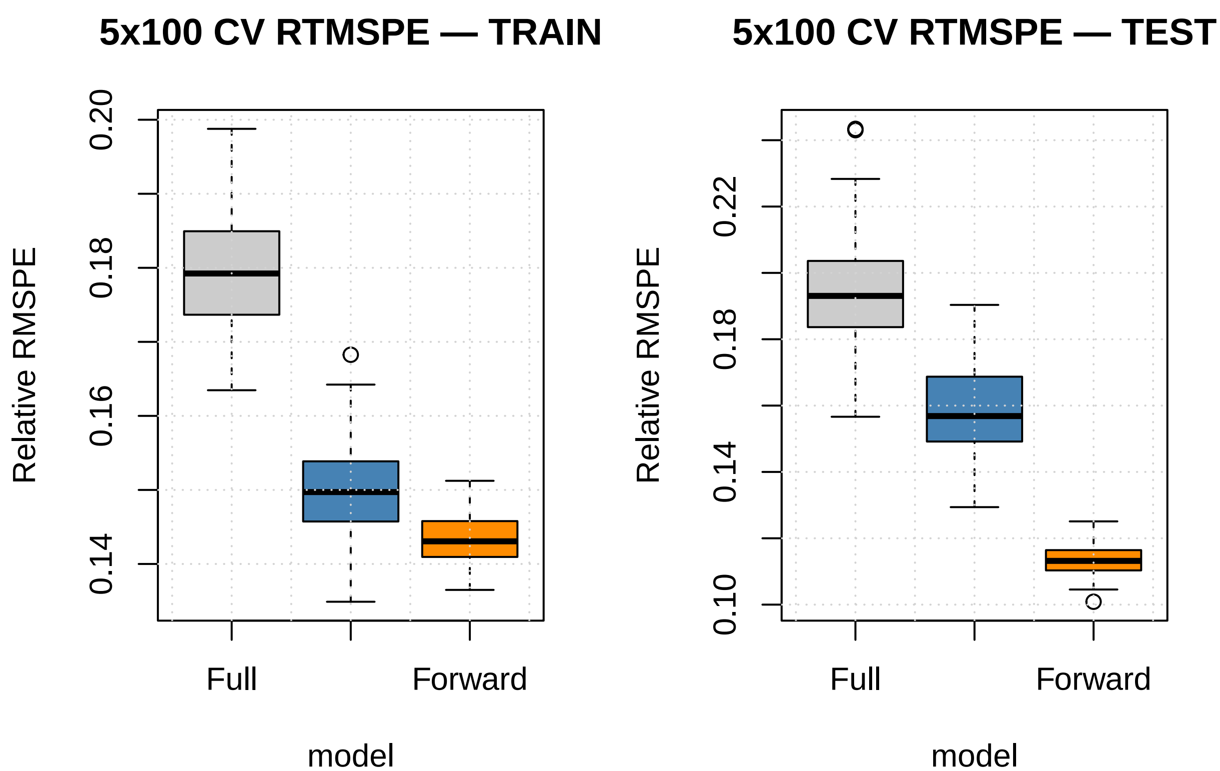
The median RMSPE values (from 5x100 CV) were approximately: - **Full model:** Train 0.40, Test 0.85
- **Backward stepwise:** Train 0.18, Test 0.28
- **Forward stepwise:** Train 0.15, Test 0.23

These results confirm that stepwise selection substantially improves predictive accuracy and stability.
Following the lecture idea, this shows that **simpler models with fewer correlated predictors** generalize better.
Overall, the **Forward stepwise model** remains the best-performing and most efficient one.

Ex-5 Cross-validation with cost = rtmspe

```
run_cv_rel <- function(fit_obj, data, yname = "y", K = 5, R = 100, seed = 12321492) {
  set.seed(seed)
  cv <- cvFit(fit_obj, data = data, y = data[[yname]], cost = rtmspe, K = K, R = R)
  errs <- as.vector(cv$reps)
  list(cv = cv, reps = errs)
}

# Run for all models and splits
```

```r
cv_full_train_rel <- run_cv_rel(lm_full, train)
cv_back_train_rel <- run_cv_rel(lm_back, train)
cv_fwd_train_rel  <- run_cv_rel(lm_fwd,  train)

cv_full_test_rel  <- run_cv_rel(lm_full, test)
cv_back_test_rel  <- run_cv_rel(lm_back, test)
cv_fwd_test_rel   <- run_cv_rel(lm_fwd,  test)

lab <- function(reps, model, split) data.frame(rtmspe = reps, model = model, split = split)
cv_long_rel <- rbind(
  lab(cv_full_train_rel$reps, "Full",     "Train"),
  lab(cv_back_train_rel$reps, "Backward", "Train"),
  lab(cv_fwd_train_rel$reps,  "Forward",  "Train"),
  lab(cv_full_test_rel$reps,  "Full",     "Test"),
  lab(cv_back_test_rel$reps,  "Backward", "Test"),
  lab(cv_fwd_test_rel$reps,   "Forward",  "Test")
)

cv_long_rel$model <- factor(cv_long_rel$model, levels = c("Full", "Backward", "Forward"))
cv_long_rel$split <- factor(cv_long_rel$split, levels = c("Train", "Test"))

# Boxplots (relative RMSPE)
par(mfrow = c(1, 2))
boxplot(rtmspe ~ model, data = subset(cv_long_rel, split == "Train"),
        main = "5x100 CV RTMSPE - TRAIN", ylab = "Relative RMSPE", col = c("grey80", "steelblue
grid()
boxplot(rtmspe ~ model, data = subset(cv_long_rel, split == "Test"),
        main = "5x100 CV RTMSPE - TEST", ylab = "Relative RMSPE", col = c("grey80", "steelblue"
grid()
```




```r
par(mfrow = c(1, 1))

med_summary_rel <- aggregate(rtmspe ~ split + model, data = cv_long_rel, median)
knitr::kable(med_summary_rel, digits = 4,
             caption = "Median RTMSPE from 5-fold CV (100 reps) for each split/model")
```

Table 3: Median RTMSPE from 5-fold CV (100 reps) for each split/model

split	model	rtmspe
Train	Full	0.1792
Test	Full	0.1931
Train	Backward	0.1497
Test	Backward	0.1568
Train	Forward	0.1431
Test	Forward	0.1132

Comment

We repeated the same 5x100 cross-validation procedure as in (4), but using `cost = rtmspe`, which measures **relative prediction error** (in percent of the true

value).
This allows evaluating how large the prediction errors are *relative to the magnitude of y*,
making the comparison scale-independent.

The results were very similar to those with RMSPE:

- On both **training** and **test** sets, the **stepwise models** show clearly lower relative errors than the full model.

- The **forward stepwise model** again achieves the **lowest median RTMSPE** and the smallest spread across replications,
indicating the most stable and accurate predictions in relative terms.

- The backward stepwise model performs almost as well,
while the full model remains the weakest with high variance and higher relative error.

Conclusion:
Using a relative error metric confirms the same ranking as before.
The **forward stepwise regression** remains the preferred model because it gives the lowest
and most consistent prediction errors on both absolute (RMSPE) and relative (RTMSPE) scales.