

4.4: An Example of the Backward Elimination Process

We previously identified the list of possible predictors that we can include in our models, shown in Table 4.1. We start the backward elimination process by putting all these potential predictors into a model for the `int00.dat` data frame using the `lm()` function.

```
> int00.lm <- lm(nperf ~ clock + threads + cores + transistors +
  dieSize + voltage + featureSize + channel + F04delay + L1icache +
  sqrt(L1icache) + L1dcache + sqrt(L1dcache) + L2cache + sqrt(L2cache),
  data=int00.dat)
```

This function call assigns the resulting linear model object to the variable `int00.lm`. As before, we use the suffix `.lm` to remind us that this variable is a linear model developed from the data in the corresponding data frame, `int00.dat`. The arguments in the function call tell `lm()` to compute a linear model that explains the output `nperf` as a function of the predictors separated by the “+” signs. The argument `data=int00.dat` explicitly passes to the `lm()` function the name of the data frame that should be used when developing this model. This `data=` argument is not necessary if we `attach()` the data frame `int00.dat` to the current workspace. However, it is useful to explicitly specify the data frame that `lm()` should use, to avoid confusion when you manipulate multiple models simultaneously.

The `summary()` function gives us a great deal of information about the linear model we just created:

```
> summary(int00.lm)
Call:
lm(formula = nperf ~ clock + threads + cores + transistors + dieSize +
  voltage + featureSize + channel + F04delay + L1icache + sqrt(L1icache) +
  L1dcache + sqrt(L1dcache) + L2cache + sqrt(L2cache), data = int00.dat)
Residuals:
    Min       1Q   Median       3Q      Max
-10.804   -2.702    0.000    2.285    9.809

Coefficients:
              Estimate            Std. Error      t value      Pr(>|t|)
(Intercept)  -2.108e+01         7.852e+01     -0.268      0.788
clock         2.605e-02         1.671e-03     15.594     < 2e-16
threads      -2.346e+00         2.089e+00     -1.123      0.260
cores         2.246e+00         1.782e+00      1.260      0.212
transistors   -5.580e-03         1.388e-02     -0.402      0.688
dieSize        1.021e-02         1.746e-02      0.585      0.556
voltage       -2.623e+01         7.698e+00     -3.408      0.000
featureSize    3.101e+01         1.122e+02      0.276      0.788
channel        9.496e+01         5.945e+02      0.160      0.877
F04delay      -1.765e-02         1.600e+00     -0.011      0.991
L1icache       1.102e+02         4.206e+01      2.619      0.007
sqrt(L1icache) -7.390e+02         2.980e+02     -2.480      0.007
L1dcache      -1.114e+02         4.019e+01     -2.771      0.000
sqrt(L1dcache)  7.492e+02         2.739e+02      2.735      0.000
L2cache       -9.684e-03         1.745e-03     -5.550      6.57e-06
sqrt(L2cache)   1.221e+00         2.425e-01      5.034      4.54e-06
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.632 on 61 degrees of freedom (179 observations deleted due
Multiple R-squared:  0.9652, Adjusted R-squared:  0.9566 F-statistic: 112.8 on 15 and 61
```

Notice a few things in this summary: First, a quick glance at the residuals shows that they are roughly balanced around a median of zero, which is what we like to see in our models. Also, notice the line, (179 observations deleted due to missingness). This tells us that in 179 of the rows in the data frame that is, in 179 of the processors for which performance results were reported for the Int2000 benchmark some of the values in the columns that we would like to use as potential predictors were missing. These NA values caused R to automatically remove these data rows when computing the linear model.

The total number of observations used in the model equals the number of degrees of freedom remaining 61 in this case plus the total number of predictors in the model. Finally, notice that the R^2 and adjusted R^2 values are relatively close to one, indicating that the model explains the `nperf` values well. Recall, however, that these large R^2 values may simply show us that the model is good at modeling the noise in the measurements. We must still determine whether we should retain all these potential predictors in the model.

To continue developing the model, we apply the backward elimination procedure by identifying the predictor with the largest p-value that exceeds our predetermined threshold of $p = 0.05$. This predictor is `F04delay`, which has a p-value of 0.99123. We can use the `update()` function to eliminate a given predictor and recompute the model in one step. The notation “~.” means that `update()` should keep the left and right-hand sides of the model the same. By including “ - `F04delay` ,” we also tell it to remove that predictor from the model, as shown in the following:

```
> int00.lm <- update(int00.lm, ~. - F04delay, data = int00.dat) > summary(int00.lm)
Call:
lm(formula = nperf ~ clock + threads + cores + transistors +
    dieSize + voltage + featureSize + channel + L1icache + sqrt(L1icache) +
    L1dcache + sqrt(L1dcache) + L2cache + sqrt(L2cache), data = int00.dat)
Residuals:
    Min       1Q   Median       3Q      Max
-10.795   -2.714    0.000    2.283    9.809

Coefficients:
              Estimate      Std. Error  t value    Pr(>|t|)
(Intercept)  -2.088e+01   7.584e+01   -0.275    0.783983
clock         2.604e-02   1.563e-03   16.662    < 2e-16 ***
threads      -2.345e+00   2.070e+00   -1.133    0.261641
cores         2.248e+00   1.759e+00    1.278    0.206080
transistors   -5.556e-03   1.359e-02   -0.409    0.684020
dieSize       1.013e-02   1.571e-02    0.645    0.521488
voltage      -2.626e+01   7.302e+00   -3.596    0.000642 ***
featureSize   3.104e+01   1.113e+02    0.279    0.781232
channel       8.855e+01   1.218e+02    0.727    0.469815
L1icache      1.103e+02   4.041e+01    2.729    0.008257 **
sqrt(L1icache) -7.398e+02   2.866e+02   -2.581    0.012230 *
L1dcache     -1.115e+02   3.859e+01   -2.889    0.005311 **
sqrt(L1dcache) 7.500e+02   2.632e+02    2.849    0.005937 **
L2cache      -9.693e-03   1.494e-01   -6.488    1.64e-08 ***
sqrt(L2cache) 1.222e+00   1.975e-01    6.189    5.33e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.594 on 62 degrees of freedom (179 observations deleted due
Multiple R-squared:  0.9652, Adjusted R-squared:  0.9573 F-statistic: 122.8 on 14 and 6
```

We repeat this process by removing the next potential predictor with the largest p-value that exceeds our predetermined threshold, `featureSize`. As we repeat this process, we obtain the following sequence of possible models.

Remove `featureSize` :

```
> int00.lm <- update(int00.lm, .~. - featureSize, data=int00.dat)
> summary(int00.lm)
```

Call:

```
lm(formula = nperf ~ clock + threads + cores + transistors + dieSize +
voltage + channel + L1icache + sqrt(L1icache) + L1dcache + sqrt(L1dcache) +
L2cache + sqrt(L2cache), data = int00.dat)
```

Residuals:

Min	1Q	Median	3Q	Max
-10.5548	-2.6442	0.0937	2.2010	10.0264

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	-3.129e+01	6.554e+01	-0.477	0.634666	
clock	2.591e-02	1.471e-03	17.609	< 2e-16	***
threads	-2.447e+00	2.022e+00	-1.210	0.230755	
cores	1.901e+00	1.233e+00	1.541	0.128305	
transistors	-5.366e-03	1.347e-02	-0.398	0.691700	
dieSize	1.325e-02	1.097e-02	1.208	0.231608	
voltage	-2.519e+01	6.182e+00	-4.075	0.000131	***
channel	1.188e+02	5.504e+01	2.158	0.034735	*
L1icache	1.037e+02	3.255e+01	3.186	0.002246	**
sqrt(L1icache)	-6.930e+02	2.307e+02	-3.004	0.003818	**
L1dcache	-1.052e+02	3.106e+01	-3.387	0.001223	**
sqrt(L1dcache)	7.069e+02	2.116e+02	3.341	0.001406	**
L2cache	-9.548e-03	1.390e-03	-6.870	3.37e-09	***
sqrt(L2cache)	1.202e+00	1.821e-01	6.598	9.96e-09	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 4.56 on 63 degrees of freedom
(179 observations deleted due to missingness)
Multiple R-squared: 0.9651, Adjusted R-squared: 0.958
F-statistic: 134.2 on 13 and 63 DF, p-value: < 2.2e-16

Remove transistors:

```
> int00.lm <- update(int00.lm, .~. - transistors, data=int00.dat)
> summary(int00.lm)
Call:
lm(formula = nperf ~ clock + threads + cores + dieSize + voltage + channel +
L1icache + sqrt(L1icache) + L1dcache + sqrt(L1dcache) + L2cache + sqrt(L2cache),
data = int00.dat)
Residuals:
```

Min	1Q	Median	3Q	Max
-9.8861	-3.0801	-0.1871	2.4534	10.4863

```

Coefficients:
              Estimate      Std. Error    t value    Pr(>|t|)
(Intercept)  -7.789e+01    4.318e+01    -1.804    0.075745 .
clock         2.566e-02    1.422e-03    18.040    < 2e-16 ***
threads      -1.801e+00    1.995e+00    -0.903    0.369794
cores         1.805e+00    1.132e+00     1.595    0.115496
dieSize       1.111e-02    8.807e-03     1.262    0.211407
voltage      -2.379e+01    5.734e+00    -4.148    9.64e-05 ***
channel       1.512e+02    3.918e+01     3.861    0.000257 ***
L1icache      8.159e+01    2.006e+01     4.067    0.000128 ***
sqrt(L1icache) -5.386e+02    1.418e+02    -3.798    0.000317 ***
L1dcache     -8.422e+01    1.914e+01    -4.401    3.96e-05 ***
sqrt(L1dcache) 5.671e+02    1.299e+02     4.365    4.51e-05 ***
L2cache      -8.700e-03    1.262e-03    -6.893    2.35e-09 ***
sqrt(L2cache)  1.069e+00    1.654e-01     6.465    1.36e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 4.578 on 67 degrees of freedom
(176 observations deleted due to missingness)
Multiple R-squared:  0.9657, Adjusted R-squared:  0.9596
F-statistic: 157.3 on 12 and 67 DF, p-value: < 2.2e-16
```

Remove threads:

```
> int00.lm <- update(int00.lm, .~. - threads, data=int00.dat)
> summary(int00.lm)

Call:
lm(formula = nperf ~ clock + cores + dieSize + voltage + channel + L1icache +
sqrt(L1icache) + L1dcache + sqrt(L1dcache) + L2cache + sqrt(L2cache), data = int00.d

Residuals:
    Min       1Q   Median       3Q      Max
-9.7388  -3.2326   0.1496   2.6633  10.6255

Coefficients:
                Estimate      Std. Error    t value    Pr(>|t|)
(Intercept)   -8.022e+01    4.304e+01    -1.864    0.066675 .
clock          2.552e-02    1.412e-03    18.074    <2e-16 ***
cores          2.271e+00    1.006e+00     2.257    0.027226 *
dieSize        1.281e-02    8.592e-03     1.491    0.140520
voltage       -2.299e+01    5.657e+00    -4.063    0.000128 ***
channel        1.491e+02    3.905e+01     3.818    0.000293 ***
L1icache       8.131e+01    2.003e+01     4.059    0.000130 ***
sqrt(L1icache) -5.356e+02    1.416e+02    -3.783    0.000329 ***
L1dcache       -8.388e+01    1.911e+01    -4.390    4.05e-05 ***
sqrt(L1dcache)  5.637e+02    1.297e+02     4.346    4.74e-05 ***
L2cache       -8.567e-03    1.252e-03    -6.844    2.71e-09 ***
sqrt(L2cache).  1.040e+00    1.619e-01     6.422    1.54e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.572 on 68 degrees of freedom
(176 observations deleted due to missingness)
Multiple R-squared:  0.9653, Adjusted R-squared:  0.9597
F-statistic: 172 on 11 and 68 DF, p-value: < 2.2e-16
```

Remove dieSize:

```
> int00.lm <- update(int00.lm, .~. - dieSize, data=int00.dat)
> summary(int00.lm)
Call:
lm(formula = nperf ~ clock + cores + voltage + channel + L1icache + sqrt(L1icache) +
L1dcache + sqrt(L1dcache) + L2cache + sqrt(L2cache), data = int00.dat)

Residuals:
    Min       1Q   Median       3Q      Max
-10.0240  -3.5195   0.3577   2.5486  12.0545

Coefficients:
              Estimate      Std. Error    t value    Pr(>|t|)
(Intercept)  -5.822e+01    3.840e+01    -1.516    0.133913
clock         2.482e-02    1.246e-03    19.922    < 2e-16 ***
cores         2.397e+00    1.004e+00     2.389    0.019561 *
voltage      -2.358e+01    5.495e+00    -4.291    5.52e-05 ***
channel       1.399e+02    3.960e+01     3.533    0.000726 ***
L1icache      8.703e+01    1.972e+01     4.412    3.57e-05 ***
sqrt(L1icache) -5.768e+02    1.391e+02    -4.146    9.24e-05 ***
L1dcache      -8.903e+01    1.888e+01    -4.716    1.17e-05 ***
sqrt(L1dcache)  5.980e+02    1.282e+02     4.665    1.41e-05 ***
L2cache      -8.621e-03    1.273e-03    -6.772    3.07e-09 ***
sqrt(L2cache)  1.085e+00    1.645e-01     6.598    6.36e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.683 on 71 degrees of freedom
(174 observations deleted due to missingness)
Multiple R-squared:  0.9641, Adjusted R-squared:  0.959
F-statistic: 190.7 on 10 and 71 DF, p-value: < 2.2e-16
```

At this point, the p-values for all of the predictors are less than 0.02, which is less than our predetermined threshold of 0.05. This tells us to stop the backward elimination process. Intuition and experience tell us that ten predictors are a rather large number to use in this type of model. Nevertheless, all of these predictors have p-values below our significance threshold, so we have no reason to exclude any specific predictor. We decide to include all ten predictors in the final model:

$$\begin{aligned} \text{nperf} = & -58.22 + 0.02482 \text{clock} + 2.397 \text{cores} \\ & -23.58 \text{voltage} + 139.9 \text{channel} + 87.03 \text{L1icache} \\ & -576.8 \sqrt{\text{L1icache}} - 89.03 \text{L1dcache} + 598 \sqrt{\text{L1dcache}} \\ & -0.008621 \text{L2cache} + 1.085 \sqrt{\text{L2cache}} \end{aligned}$$

Looking back over the sequence of models we developed, notice that the number of degrees of freedom in each subsequent model increases as predictors are excluded, as expected. In some cases, the number of degrees of freedom increases by more than one when only a single predictor is eliminated from the model. To understand how an increase of more than one is possible, look at the sequence of values in the lines labeled `the number of observations dropped due to missingness`. These values show how many rows the `update()` function dropped because the value for one of the predictors in those rows was missing and had the NA value. When the backward elimination process removed that predictor from the model, at least some of those rows became ones we can use in computing the next version of the model, thereby increasing the number of degrees of freedom.

Also notice that, as predictors drop from the model, the R^2 values stay very close to 0.965. However, the adjusted R^2 value tends to increase very slightly with each dropped predictor. This increase indicates that the model with fewer predictors and more degrees of freedom tends to explain the data slightly better than the previous model, which had one more predictor. These changes

in R^2 values are very small, though, so we should not read too much into them. It is possible that these changes are simply due to random data fluctuations. Nevertheless, it is nice to see them behaving as we expect.

Roughly speaking, the F-test compares the current model to a model with one fewer predictor. If the current model is better than the reduced model, the p-value will be small. In all of our models, we see that the p-value for the F-test is quite small and consistent from model to model. As a result, this F-test does not particularly help us discriminate between potential models.

This page titled [4.4: An Example of the Backward Elimination Process](#) is shared under a [CC BY-NC 4.0](#) license and was authored, remixed, and/or curated by [David Lilja](#) ([University of Minnesota Libraries Publishing](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.