Lab 10: Model selection with the leaps package

2020-11-18

To “implement” model selection, we need:

- a criterion or benchmark to compare two models.
- a search strategy.

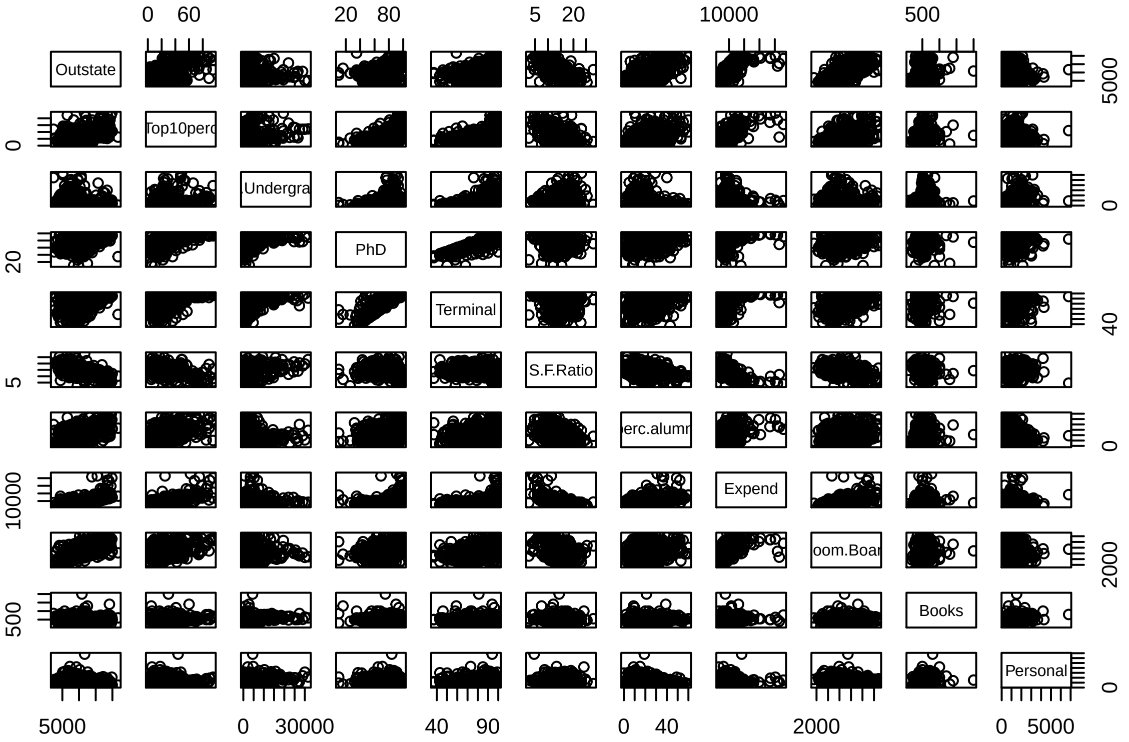
With a limited number of predictors, it is possible to search all possible models.

We’ll work with a subset of the ISLR College dataset

```
data(College)
set.seed(1)
# subset rows and columns
College %>% select(Outstate,Top10perc,F.Undergrad,PhD,Terminal,S.F.Ratio,perc.alumni,Expend,Room.Board,
# what are the variables?
names(College)
```

```
##  [1] "Outstate"    "Top10perc"   "F.Undergrad" "PhD"         "Terminal"
##  [6] "S.F.Ratio"   "perc.alumni" "Expend"      "Room.Board"  "Books"
## [11] "Personal"
```

```
# inspect pairs plot
pairs(College)
```



We will use some of the variables in the College dataset to predict Out-of-state tuition. But which variables?

Best subset selection with leaps

We have seen that we can evaluate models with R-squared and adjusted R-squared. Why not fit all the models (i.e. all possible subsets of predictors) and compare them?

This is implemented efficiently in the leaps package.

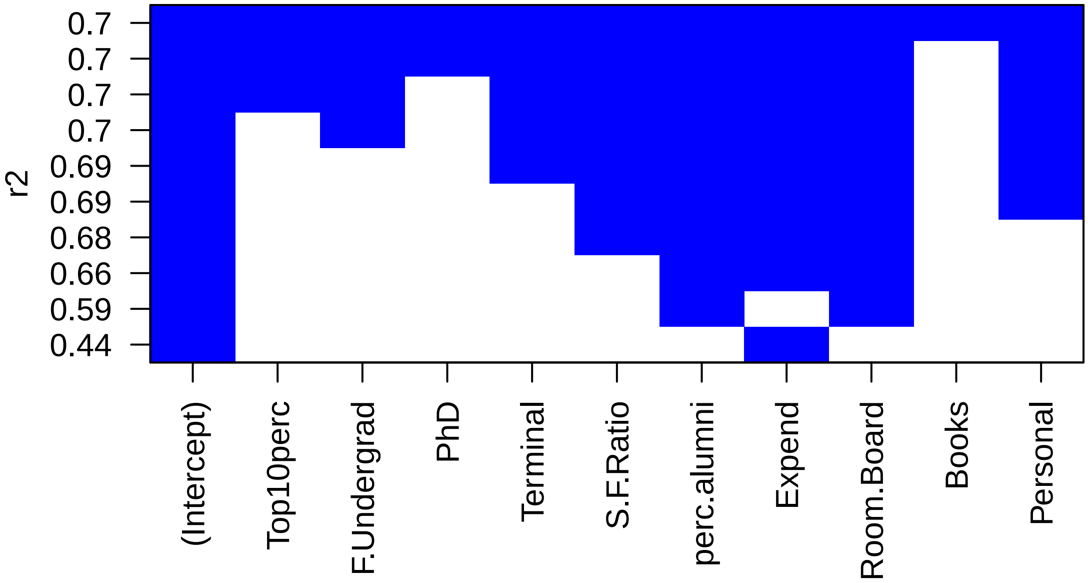
```r
# import package
library(leaps)

# regsubsets takes data in matrix/vector format
y <- College$Outstate
X <- data.matrix(College)[,-1]

# basic syntax is regsubsets(X,y,...)
# note that resubsets has many optional parameters, which can be
# viewed with ?regsubsets
leap.output <- regsubsets(X,y,method="exhaustive",nbest=2,nvmax=10)

# inspect leaps output
summary(leap.output)
```

```
## Subset selection object
## 10 Variables  (and intercept)
##              Forced in Forced out
## Top10perc       FALSE      FALSE
## F.Undergrad     FALSE      FALSE
## PhD             FALSE      FALSE
## Terminal        FALSE      FALSE
## S.F.Ratio       FALSE      FALSE
## perc.alumni     FALSE      FALSE
## Expend          FALSE      FALSE
## Room.Board      FALSE      FALSE
## Books           FALSE      FALSE
## Personal        FALSE      FALSE
## 2 subsets of each size up to 10
## Selection Algorithm: exhaustive
##           Top10perc F.Undergrad PhD Terminal S.F.Ratio perc.alumni Expend
## 1  ( 1 )  " "       " "         " " " "      " "       " "         "*"   
## 1  ( 2 )  " "       " "         " " " "      " "       " "         " "   
## 2  ( 1 )  " "       " "         " " " "      " "       "*"         " "   
## 2  ( 2 )  " "       " "         " " " "      " "       " "         "*"   
## 3  ( 1 )  " "       " "         " " " "      " "       "*"         "*"   
## 3  ( 2 )  " "       " "         " " " "      "*"       "*"         " "   
## 4  ( 1 )  " "       " "         " " " "      "*"       "*"         "*"   
## 4  ( 2 )  " "       "*"         " " " "      " "       "*"         "*"   
## 5  ( 1 )  " "       " "         " " " "      "*"       "*"         "*"   
## 5  ( 2 )  " "       "*"         " " " "      "*"       "*"         "*"   
## 6  ( 1 )  " "       " "         " " "*"      "*"       "*"         "*"   
## 6  ( 2 )  " "       "*"         " " "*"      "*"       "*"         "*"   
## 7  ( 1 )  " "       "*"         " " "*"      "*"       "*"         "*"   
## 7  ( 2 )  " "       "*"         "*" " "      "*"       "*"         "*"   
## 8  ( 1 )  "*"       "*"         " " "*"      "*"       "*"         "*"   
## 8  ( 2 )  "*"       "*"         "*" " "      "*"       "*"         "*"   
## 9  ( 1 )  "*"       "*"         "*" "*"      "*"       "*"         "*"   
## 9  ( 2 )  "*"       "*"         " " "*"      "*"       "*"         "*"
## 10  ( 1 ) "*"       "*"         "*" "*"      "*"       "*"         "*"
##           Room.Board Books Personal
## 1  ( 1 )  " "        " "   " "
## 1  ( 2 )  "*"        " "   " "
## 2  ( 1 )  "*"        " "   " "
## 2  ( 2 )  "*"        " "   " "
## 3  ( 1 )  "*"        " "   " "
## 3  ( 2 )  "*"        " "   " "
## 4  ( 1 )  "*"        " "   " "
## 4  ( 2 )  "*"        " "   " "
## 5  ( 1 )  "*"        " "   "*"
## 5  ( 2 )  "*"        " "   " "
## 6  ( 1 )  "*"        " "   "*"
## 6  ( 2 )  "*"        " "   " "
## 7  ( 1 )  "*"        " "   "*"
## 7  ( 2 )  "*"        " "   "*"
## 8  ( 1 )  "*"        " "   "*"
## 8  ( 2 )  "*"        " "   "*"
## 9  ( 1 )  "*"        " "   "*"
## 9  ( 2 )  "*"        "*"   "*"
## 10  ( 1 ) "*"        "*"   "*"
```

summary(leaps) returns a matrix giving the predictors used in each model. In this case it returns the two best models of each size. This can be altered with the 'nbest' parameter.

let's plot the output showing the best model of each size

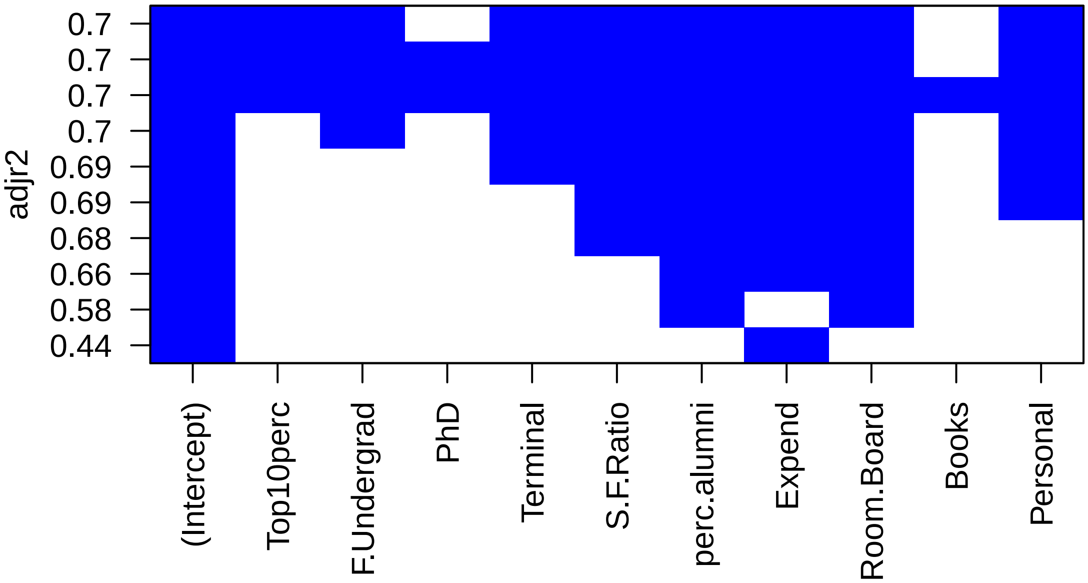
```
leap.output2 <- regsubsets(X,y,method="exhaustive",nbest=1,nvmax=10)
plot(leap.output2,scale="r2",col="blue")
```



As we could have predicted, the best R-squared grows monotonically with the model size. Let's try the same thing with adjusted R-squared, althouh for the top 4 models the R-squared is nearly indistinguishable.

Note that we don't need to rerun regsubsets to do this

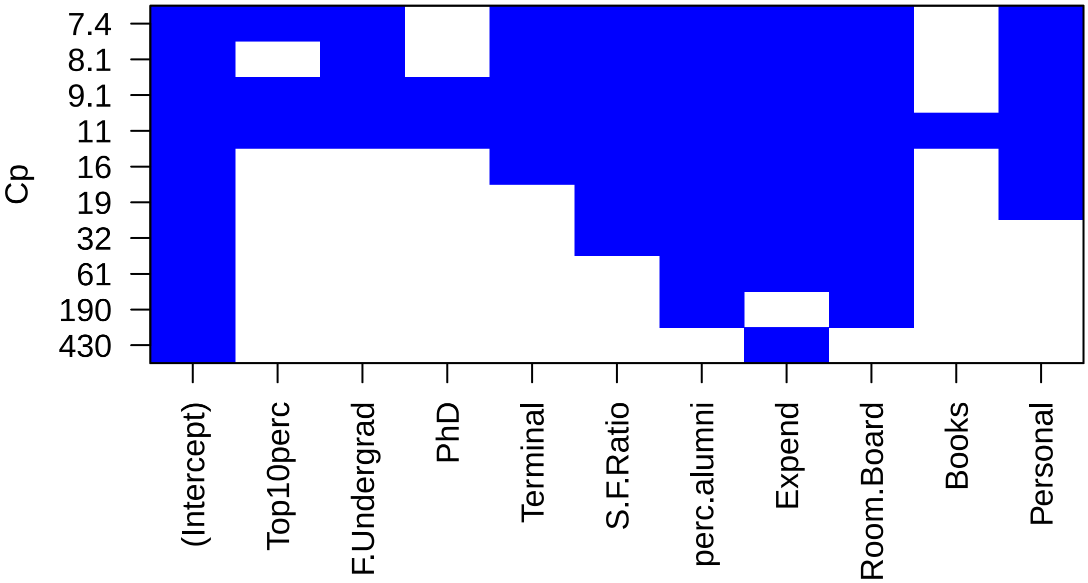
```
plot(leap.output2,scale="adjr2",col="blue")
```



Now there are models which leave out one or two predictors with better adjusted R-squared.

Finally we do this with Cp.

```
plot(leap.output2,scale="Cp",col="blue")
```

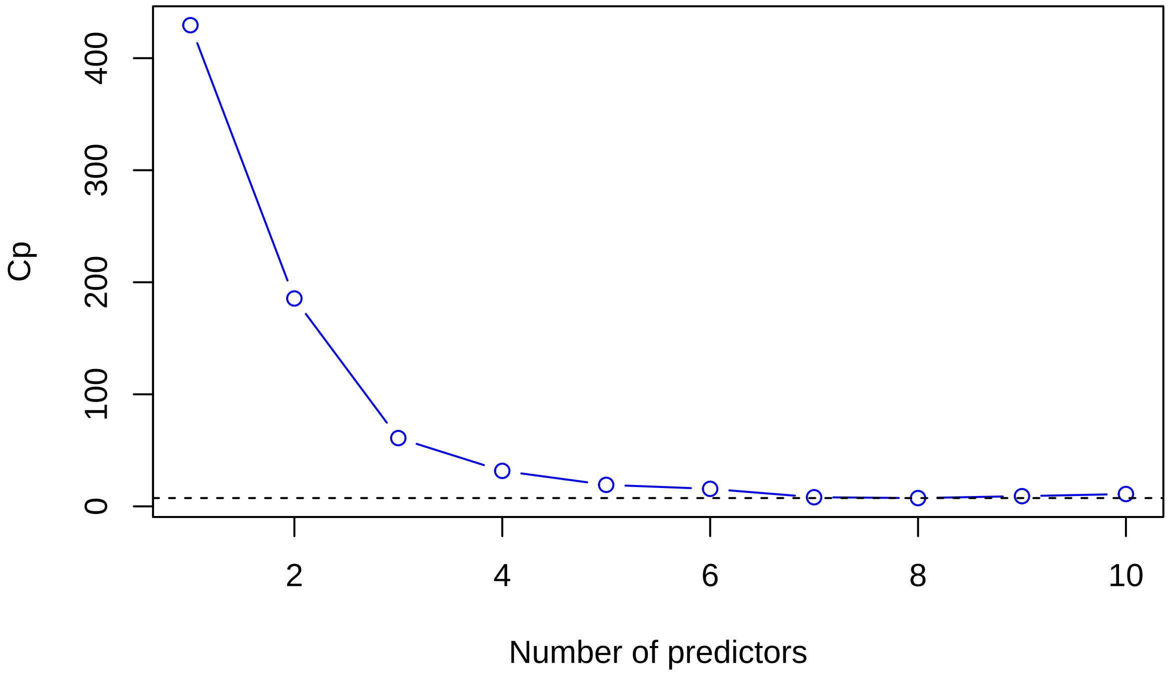


Cp gives the same best model, but the penalty on complexit is greater that adjusted R-squared.

Note that the models are not nested. The variable 'Expend' enters the model, then leaves again.

As a final plot, we look at the optimal Cp vs model size (recall that for Cp, lower is better).

```
cp <- summary(leap.output2)$cp
plot(1:10,cp,col="blue",type="b",
     xlab="Number of predictors",ylab="Cp")
abline(h=min(cp),lty=2)
```



How do we extract the best model(s)? (Note we might be interested in the best model of a given size)

```
# find the model with the best Cp
leap.summary <- summary(leap.output2)

# what is the best Cp

index.best <- which.min(leap.summary$cp)
# size of the best model
index.best
```

```
## [1] 8
```

```
# variables of the best model
var.best <- leap.summary$which[index.best,-1]

colnames(X)
```

```
##  [1] "Top10perc"   "F.Undergrad" "PhD"         "Terminal"    "S.F.Ratio"
##  [6] "perc.alumni" "Expend"      "Room.Board"  "Books"       "Personal"
```

```
colnames(X)[var.best]
```

```
## [1] "Top10perc"   "F.Undergrad" "Terminal"    "S.F.Ratio"   "perc.alumni"
## [6] "Expend"      "Room.Board"  "Personal"
```

```
# find the model with 3 predictors with the best adj Cp (note for
# a fixed size all the metrics are equivalent)

# variables
var.best.three <- leap.summary$which[3,-1]

colnames(X)[var.best.three]
```

```
## [1] "perc.alumni" "Expend"      "Room.Board"
```

So the best model according to Cp includes all variables besides “PhD” and “Books”.

The best model with three predictors contains the predictors “perc.alumni”,“Expend” and “Room.Board”.

Stepwise variable selection with leaps

Subset selection can be computationally expensive when there are a lot of predictors in the model (2^p possible subsets).

regsubsets can also be used to select models in a stepwise manner.

Stepwise selection is a greedy procedure that can operate in a forward or backward direction.

In forward stepwise selection, variables are added to the model until to give the maximal decrease in RSS.

In backward stepwise selection, variables are removed from the model to give the minimal increase in RSS.

Let’s select a model for the College dataset using AIC and forward selection

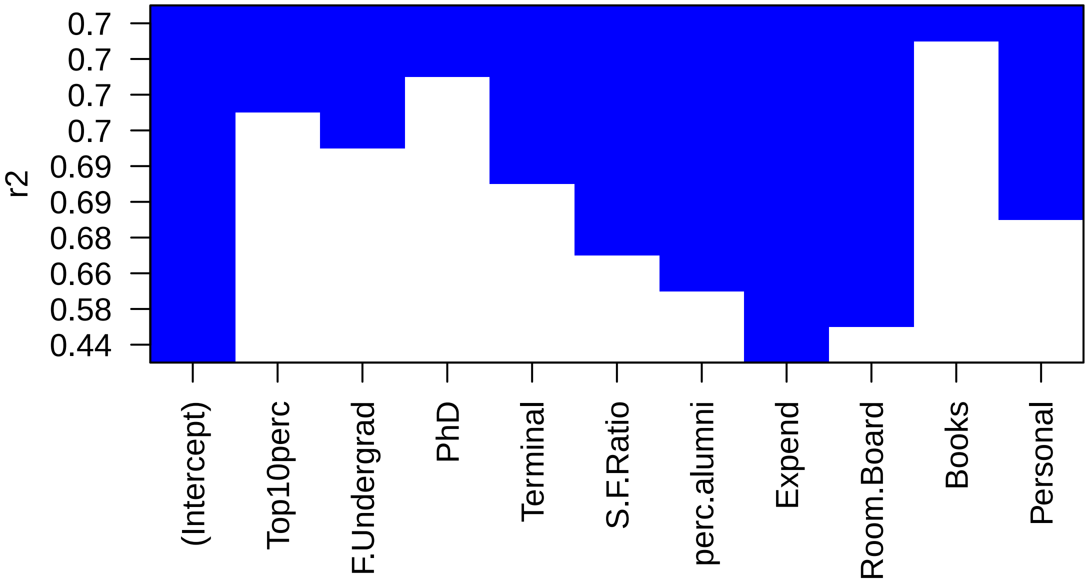
```r
leaps.stepforward <- regsubsets(X,y,method="forward",nvmax=10)
summary(leaps.stepforward)
```

```
## Subset selection object
## 10 Variables  (and intercept)
##             Forced in Forced out
## Top10perc       FALSE      FALSE
## F.Undergrad     FALSE      FALSE
## PhD             FALSE      FALSE
## Terminal        FALSE      FALSE
## S.F.Ratio       FALSE      FALSE
## perc.alumni     FALSE      FALSE
## Expend          FALSE      FALSE
## Room.Board      FALSE      FALSE
## Books           FALSE      FALSE
## Personal        FALSE      FALSE
## 1 subsets of each size up to 10
## Selection Algorithm: forward
##           Top10perc F.Undergrad PhD Terminal S.F.Ratio perc.alumni Expend
## 1  ( 1 )  " "       " "         " " " "      " "       " "         "*"   
## 2  ( 1 )  " "       " "         " " " "      " "       " "         "*"   
## 3  ( 1 )  " "       " "         " " " "      " "       "*"         "*"   
## 4  ( 1 )  " "       " "         " " " "      "*"       "*"         "*"   
## 5  ( 1 )  " "       " "         " " " "      "*"       "*"         "*"   
## 6  ( 1 )  " "       " "         " " "*"      "*"       "*"         "*"   
## 7  ( 1 )  " "       "*"         " " "*"      "*"       "*"         "*"   
## 8  ( 1 )  "*"       "*"         " " "*"      "*"       "*"         "*"   
## 9  ( 1 )  "*"       "*"         "*" "*"      "*"       "*"         "*"   
## 10  ( 1 ) "*"       "*"         "*" "*"      "*"       "*"         "*"   
##           Room.Board Books Personal
## 1  ( 1 )  " "        " "   " "     
## 2  ( 1 )  "*"        " "   " "     
## 3  ( 1 )  "*"        " "   " "     
## 4  ( 1 )  "*"        " "   " "     
## 5  ( 1 )  "*"        " "   "*"     
## 6  ( 1 )  "*"        " "   "*"     
## 7  ( 1 )  "*"        " "   "*"     
## 8  ( 1 )  "*"        " "   "*"     
## 9  ( 1 )  "*"          " "    "*"
## 10  ( 1 ) "*"          "*"    "*"
```

Notice that the third step in the stepwise selection is the same as the best model of size 3 selected by adjusted R-squared. The greedy approach is typically a good approximation of the exhaustive subset selection approach.

We can plot the output as above

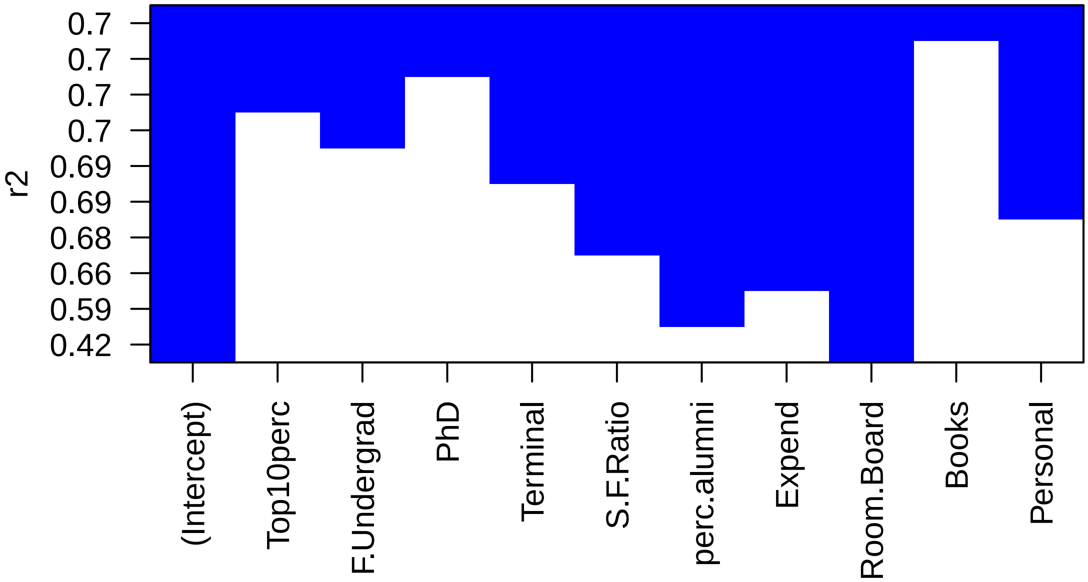
```
plot(leaps.stepforward,col="blue",scale="r2")
```



Now the models are nested.

What about if we use backward selection instead?

```
leaps.stepbackward <- regsubsets(X,y,method="backward",nvmax=10)
plot(leaps.stepbackward,col="blue",scale="r2")
```

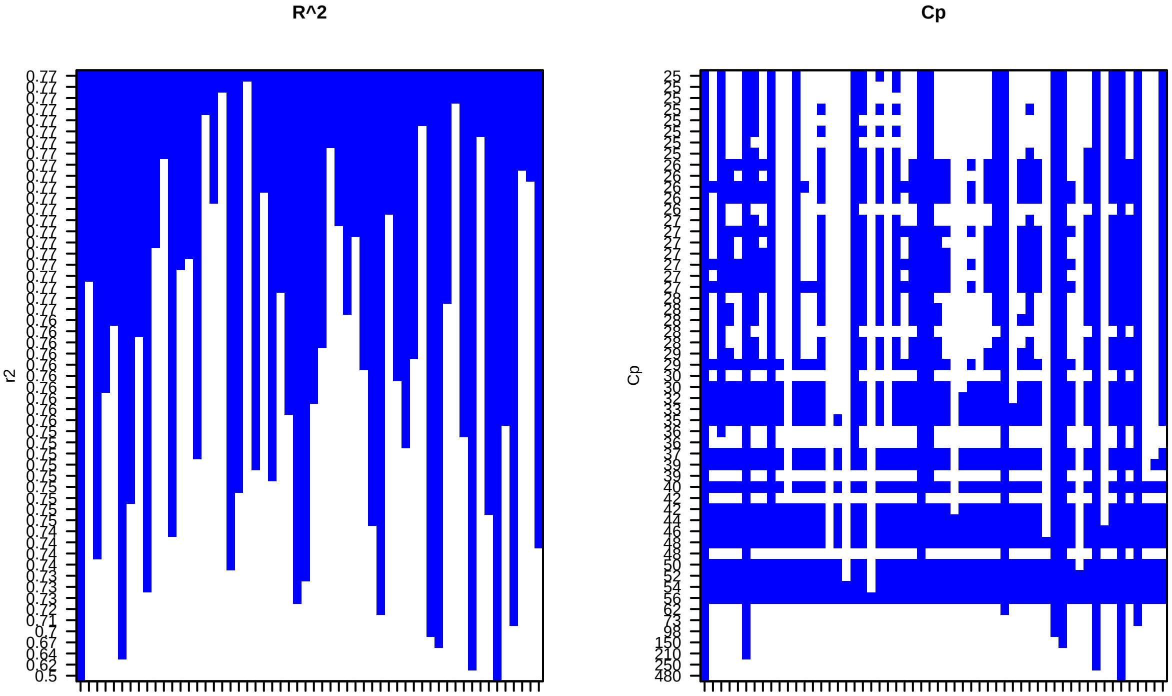


We can use code similar to above to get the best model (according to Cp) in both cases.

Here we see that both forward and backward selection will choose the same best model as the exhaustive method (since they both find the overall best model with 8 predictors).

The big advantage of stepwise selection is that it speeds up calculations when the number of predictors is very large. For instance, if we want to fit models which consider all 2-way interactions of the 10 predictors. This is 55 possible variables, so in subset selection we would have to consider 2^{55} models.

```
# get the new matrix of predictors with 2-way interactions
# a fast way to do this is with model.matrix
Xint <- model.matrix(lm(Outstate~.^2,data=College))[,-1]
# -1 to remove the intercept

leaps.bigforward <- regsubsets(Xint,y,nvmax=55,method="forward")

par(mfrow=c(1,2),cex=.5)
plot(leaps.bigforward,scale="r2",labels=rep("",56),col="blue",main="R^2")
plot(leaps.bigforward,scale="Cp",labels=rep("",56),col="blue",main="Cp")
```



```
par(mfrow=c(1,1),cex=1)
# suppressed x-axis labels, unfortunately no good way to suppress y-axis
# tick labels
```

As before, R-squared increases monotonically with model complexity, while Cp penalizes model complexity. Thus Cp chooses some smaller model as the best model.

```
index.best.interactions <- which.min(summary(leaps.bigforward)$cp)
# size of the best model
index.best.interactions
```

```
## [1] 20
```

```
# variables of the best model
var.best <- summary(leaps.bigforward)$which[index.best.interactions,-1]

# how many main effects? (first 10 columns)
```

```r
sum(var.best[1:10])
```

```
## [1] 4
```

```r
# how many interactions? (columns 11-55)
sum(var.best[11:55])
```

```
## [1] 16
```

Considering all 2-way interactions, forward stepwise selection chooses a model with 20 predictors - 4 main effects and 16 interaction effects. This model will be much more difficult to interpret but may perform better for prediction tasks.