```
#R code: Discussion 9.                          Sta108, Fall 2007, Utts

#today's topics:
#Troubleshooting with R
#Model selection
#Homework help

#Troubleshooting with R
#If you reach an error message because you forgot how to use a certain function/command,
#Type: a question-mark, followed by the name of the function/command
#This will open the help manual file for that function/command
?plot
?lm
?leaps
?update
#It is helpful to scroll to the end to see examples how to use such commands

#Model selection
#Goal: Choose the most parsimonious (best) model from candidate sub-models based on a chosen
Criterion
  #Choose Maximum R-Square from candidate sub-models
  #Choose Maximum Adjusted-R-Square from candidate sub-models
  #Choose Minimum Mallows' Cp from candidate sub-models
  #Choose Minimum AICp from candidate sub-models
  #Choose Minimum BICp from candidate sub-models
  #Choose Minimum PRESSp from candidate sub-models

#Example: Grocery Retailer: Problem 6.9
Data = read.table("CH06PR09.txt")
names(Data) = c("Hours","Cases","Costs","Holiday")

#To obtain the AICp criterion for any sub-model,
#1.Obtain a linear fit involving just the predictors for that sub-model, call it Fit
#2.Use extractAIC() function:
Fit = lm( Hours ~ Cases + Costs + Holiday, data=Data)
extractAIC(Fit)

#To obtain the SBCp criterion (also called BICp):
extractAIC(Fit, k = log(n))

#To obtain the PRESSp criterion for each sub-model:
sum( (Fit$residuals / ( 1-hatvalues( Fit ) ))^2 )
  #Be careful with the parentheses


###Stepwise regression

#Possible choices: forward selection, backward elimination, or combination of both (called "forward
#stepwise regression" in text)
#Method 1: function step() - uses AICp criterion at each step, automatic procedure
#Method 2: function summary() - read P-values, manually update
#Method 3: functions addterm(),dropterm() - read F-statistics/P-values, manually update


#Method 1:
#Forward selection
#1.Fit initial/base model (with one predictor)
#2.Fit full model (with all the predictors you wish to consider)
#3.Use step() function
Base = lm( Hours ~ Holiday, data=Data )
Full = lm( Hours ~ Cases + Costs + Holiday, data=Data )
step(Base, scope = list( upper=Full, lower=~1 ), direction = "forward", trace=FALSE)
###Input:
  #the first parameter is the initial model in stepwise search, (I called it Base)
  #score: defines the range of models examined in the stepwise search
     #upper: defines the full model
     #lower: defines the most simple model, (in this case: just the intercept term)
  #direction: mode of stepwise search, can be one of "forward", "backward", or "both"
  #trace: FALSE gives only the final model, TRUE gives intermediate results at each step
###Output:
  #step() identifies and fits the model which produced the lowest value of AIC
```

```
#Backward elimination
#1.Fit initial/base model, which is the full model (with all the predictors you wish to consider)
#2.Use step() function
step( Full, direction = "backward", trace=FALSE )

#Both Forward and Backward stepwise regression ("Forward stepwise regression" in text)
step(Base, scope = list( upper=Full, lower=~1 ), direction = "both", trace=FALSE)


#Method 2:
#Backward elimination using P-values to delete predictors one-at-a-time
#0.Choose significance level Alpha before you begin
#1.START with fitting full model,
  #a. look at model summary(),
  #b. identify the predictor (if any) with the largest P-value above your Alpha-level
#2.DROP. Fit a new linear model with that predictor deleted
  #use the update() function to make this easier
  #a. look at model summary(),
  #b. identify the predictor (if any) with the largest P-value above your Alpha-level
#3.Repeat Step #2 if predictor was identified, or
  #STOP stepwise regression if all remaining P-values are below your Alpha-level

Full = lm( Hours ~ Cases + Costs + Holiday, data=Data )
summary(Full)
NewMod = update( Full, .~. - Costs )
summary(NewMod)


#Method 3:
#Backward elimination using R function dropterm() in the MASS package
library(MASS)
  #addterm(), dropterm() functions use an F-test criterion or a P-value criterion
#0.Choose F limit or level Alpha before you begin
#1.START with fitting full model,
  #a. use dropterm() function
  #b. identify (to delete) the predictor (if any) with the smallest F-value below your F limit, or
     #the largest P-value above your Alpha-level
#2.DROP. Fit a new linear model with that predictor deleted
  #use the update() function to make this easier
  #a. use dropterm() function
  #b. identify (to delete) the predictor (if any) with the smallest F-value below your F limit, or
     #the largest P-value above your Alpha-level
#3.Repeat Step #2 if predictor was identified, or
  #STOP stepwise regression if all remaining P-values are below your Alpha-level

Full = lm( Hours ~ Cases + Costs + Holiday, data=Data )
dropterm( Full, test = "F" )
NewMod = update( Full, .~. - Costs )
dropterm( NewMod, test = "F" )


#Forward selection using R function addterm() in the MASS package
library(MASS)
  #addterm(), dropterm() functions use an F-test criterion or a P-value criterion
#0.Choose F limit or level Alpha before you begin
#1.START with fitting null model, say, no predictors but only intercept
  #a. use addterm() function
  #b. identify (to admit) the predictor (if any) with the largest value above your F limit, or
     #the smallest P-value below your Alpha-level.
#2.ADD. Fit a new linear model with that predictor deleted
  #use the update() function to make this easier
  #a. use addterm() function
  #b. identify (to admit) the predictor (if any) with the largest value above your F limit, or
     #the smallest P-value below your Alpha-level.
#3.Repeat Step #2 if predictor was identified, or
  #STOP stepwise regression if all F-values are larger than your F limit, or
     #all P-values are  below your Alpha-level

Null = lm( Hours ~ 1, data=Data )
addterm( Null, scope = Full, test="F" )
NewMod = update( Null, .~. + Holiday )
addterm( NewMod, scope = Full, test="F" )
```

```
NewMod = update( NewMod, .~. + Cases )
addterm( NewMod, scope = Full, test="F" )



#Homework help:
#Example: Grocery Retailer: Problem 6.9
Data = read.table("CH06PR09.txt")
names(Data) = c("Hours","Cases","Costs","Holiday")

DataX=Data[,2:4]
DataY=Data[,1]
names(Data)

library(leaps)
leaps( x=DataX, y=DataY, names=c("Cases","Costs","Holiday"), method="Cp")

#To automatically print models in the increasing order of Cp criterion:
ModelSel = leaps( x=DataX, y=DataY, names=c("Cases","Costs","Holiday"), method="Cp")
ModelSel$which[ order( ModelSel$Cp ), ]

#To print Cp criterion in increasing order
sort( ModelSel$Cp )
```