Homework1

February 10, 2019

*1.The data are available in BbLearn in the file, caterpillar.txt. Read the caterpillar data set into R and use R to: (i) name the columns of the resulting data frame as y, x1,…,x10, and (ii) display the first 3 and last 3 records of your data set. Show your code and output.*

caterpillar\_data<- read.table("F:/INF504/Homework/Homework1/caterpillar.txt")  
#name the columns of the resulting data frame as y,x1,...,x10  
colnames(caterpillar\_data) <- c("x1","x2","x3","x4","x5","x6","x7","x8","x9","x10","y")  
head(caterpillar\_data,3)

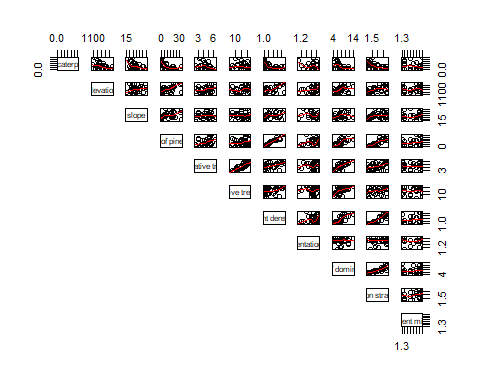
## x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 y  
## 1 1200 22 1 4.0 14.8 1.0 1.1 5.9 1.4 1.4 2.37  
## 2 1342 28 8 4.4 18.0 1.5 1.5 6.4 1.7 1.7 1.47  
## 3 1231 28 5 2.4 7.8 1.3 1.6 4.3 1.5 1.4 1.13

tail(caterpillar\_data,3)

## x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 y  
## 31 1228 31 6 5.4 21.8 1.3 1.7 7.0 1.5 1.9 0.35  
## 32 1229 21 11 5.8 16.7 1.7 1.8 10.0 2.3 2.0 0.21  
## 33 1310 36 17 5.2 17.8 2.3 1.9 10.3 2.6 2.0 0.03

*2.Create an appropriate scatter plot matrix, similar to that seen in our notes. You may use the pairs function or a similar function in R. Use the labels option of the pairs function to provide more informative variable names than given above. If you use another plot function, label your plot in a similar manner. Show your code and plot.*

labels = c("# of caterpillar","elevation","slope","# of pines","representative tree height","representative tree diameter","settlement density index","site orientation index","height of dominant tree","vegetation strata index","settlement mix index")  
pairs(y ~.,labels,upper.panel = panel.smooth, lower.panel=NULL,data = caterpillar\_data)



*3.How many linear models are posiible using all possible subsets of the inputs, including the model consisting only an intercept?*

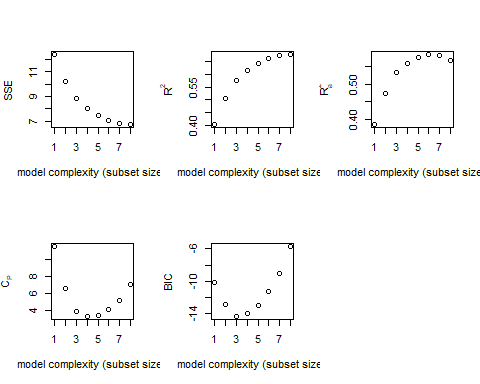
***There will be = 2048 linear models***

*4.Use the function, regsubsets, in the leaps package, to perform an exhaustive search for the best model/subset. In particular, create a matrix of plots, one for each of SSE,,,CP,and BIC, each plot showing the single best model for each of the possible sizes. For the plot of CP vs. number of covariates, add the reference line, CP = 1+|P|.Show your code and plots, appropriately annotated.*

library(leaps)  
#wold<- getOption("width")  
#options(width=75)  
caterpillar\_data.rusb <- regsubsets(y ~ ., data=caterpillar\_data, nbest=6,nvmax=9)  
(cdrsub.sum <- summary(caterpillar\_data.rusb))

## Subset selection object  
## Call: regsubsets.formula(y ~ ., data = caterpillar\_data, nbest = 6,   
## nvmax = 9)  
## 10 Variables (and intercept)  
## Forced in Forced out  
## x1 FALSE FALSE  
## x2 FALSE FALSE  
## x3 FALSE FALSE  
## x4 FALSE FALSE  
## x5 FALSE FALSE  
## x6 FALSE FALSE  
## x7 FALSE FALSE  
## x8 FALSE FALSE  
## x9 FALSE FALSE  
## x10 FALSE FALSE  
## 6 subsets of each size up to 9  
## Selection Algorithm: exhaustive  
## x1 x2 x3 x4 x5 x6 x7 x8 x9 x10  
## 1 ( 1 ) " " " " " " " " " " " " " " " " "\*" " "  
## 1 ( 2 ) " " " " " " " " " " "\*" " " " " " " " "  
## 1 ( 3 ) " " " " "\*" " " " " " " " " " " " " " "  
## 1 ( 4 ) " " " " " " " " " " " " " " "\*" " " " "  
## 1 ( 5 ) "\*" " " " " " " " " " " " " " " " " " "  
## 1 ( 6 ) " " "\*" " " " " " " " " " " " " " " " "  
## 2 ( 1 ) "\*" " " " " " " " " " " " " " " "\*" " "  
## 2 ( 2 ) " " "\*" " " " " " " " " " " " " "\*" " "  
## 2 ( 3 ) "\*" "\*" " " " " " " " " " " " " " " " "  
## 2 ( 4 ) " " " " " " " " " " " " "\*" " " "\*" " "  
## 2 ( 5 ) "\*" " " " " " " " " " " " " "\*" " " " "  
## 2 ( 6 ) " " "\*" " " " " " " "\*" " " " " " " " "  
## 3 ( 1 ) "\*" "\*" " " " " " " " " " " " " "\*" " "  
## 3 ( 2 ) "\*" " " " " " " " " "\*" " " " " "\*" " "  
## 3 ( 3 ) "\*" " " "\*" " " " " " " " " " " "\*" " "  
## 3 ( 4 ) "\*" "\*" " " " " " " " " " " "\*" " " " "  
## 3 ( 5 ) " " "\*" " " " " " " " " "\*" " " "\*" " "  
## 3 ( 6 ) "\*" " " " " " " " " " " "\*" " " "\*" " "  
## 4 ( 1 ) "\*" "\*" " " " " " " "\*" " " " " "\*" " "  
## 4 ( 2 ) "\*" "\*" "\*" " " " " " " " " " " "\*" " "  
## 4 ( 3 ) "\*" "\*" " " "\*" "\*" " " " " " " " " " "  
## 4 ( 4 ) "\*" "\*" " " " " " " " " "\*" " " "\*" " "  
## 4 ( 5 ) "\*" "\*" " " " " "\*" " " " " " " "\*" " "  
## 4 ( 6 ) "\*" "\*" " " " " "\*" " " " " "\*" " " " "  
## 5 ( 1 ) "\*" "\*" " " " " " " "\*" "\*" " " "\*" " "  
## 5 ( 2 ) "\*" "\*" " " "\*" "\*" " " " " " " "\*" " "  
## 5 ( 3 ) "\*" "\*" "\*" " " " " " " "\*" " " "\*" " "  
## 5 ( 4 ) "\*" "\*" " " "\*" "\*" " " " " " " " " "\*"  
## 5 ( 5 ) "\*" "\*" " " "\*" "\*" " " " " "\*" " " " "  
## 5 ( 6 ) "\*" "\*" "\*" " " " " " " " " " " "\*" "\*"  
## 6 ( 1 ) "\*" "\*" " " "\*" "\*" "\*" " " " " "\*" " "  
## 6 ( 2 ) "\*" "\*" "\*" "\*" "\*" " " " " " " "\*" " "  
## 6 ( 3 ) "\*" "\*" " " "\*" "\*" " " " " " " "\*" "\*"  
## 6 ( 4 ) "\*" "\*" " " "\*" "\*" " " "\*" " " "\*" " "  
## 6 ( 5 ) "\*" "\*" " " "\*" "\*" " " " " "\*" "\*" " "  
## 6 ( 6 ) "\*" "\*" " " " " "\*" "\*" "\*" " " "\*" " "  
## 7 ( 1 ) "\*" "\*" "\*" "\*" "\*" " " " " " " "\*" "\*"  
## 7 ( 2 ) "\*" "\*" " " "\*" "\*" "\*" "\*" " " "\*" " "  
## 7 ( 3 ) "\*" "\*" " " "\*" "\*" "\*" " " " " "\*" "\*"  
## 7 ( 4 ) "\*" "\*" "\*" "\*" "\*" " " "\*" " " "\*" " "  
## 7 ( 5 ) "\*" "\*" "\*" "\*" "\*" "\*" " " " " "\*" " "  
## 7 ( 6 ) "\*" "\*" " " "\*" "\*" "\*" " " "\*" "\*" " "  
## 8 ( 1 ) "\*" "\*" "\*" "\*" "\*" " " "\*" " " "\*" "\*"  
## 8 ( 2 ) "\*" "\*" "\*" "\*" "\*" " " " " "\*" "\*" "\*"  
## 8 ( 3 ) "\*" "\*" " " "\*" "\*" "\*" "\*" " " "\*" "\*"  
## 8 ( 4 ) "\*" "\*" "\*" "\*" "\*" "\*" " " " " "\*" "\*"  
## 8 ( 5 ) "\*" "\*" " " "\*" "\*" "\*" "\*" "\*" "\*" " "  
## 8 ( 6 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" " " "\*" " "  
## 9 ( 1 ) "\*" "\*" "\*" "\*" "\*" " " "\*" "\*" "\*" "\*"  
## 9 ( 2 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" " " "\*" "\*"  
## 9 ( 3 ) "\*" "\*" "\*" "\*" "\*" "\*" " " "\*" "\*" "\*"  
## 9 ( 4 ) "\*" "\*" " " "\*" "\*" "\*" "\*" "\*" "\*" "\*"  
## 9 ( 5 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*" " "  
## 9 ( 6 ) "\*" "\*" "\*" " " "\*" "\*" "\*" "\*" "\*" "\*"

par(mfrow=c(2,3))  
p1best.sum<- summary(p1best.rbest<-regsubsets(y ~ ., data=caterpillar\_data))  
#compares SSE, R2a, CP (AIC) and BIC.  
plot(p1best.sum$rss,xlab="model complexity (subset size)",ylab="SSE")  
plot(p1best.sum$rsq,xlab="model complexity (subset size)",ylab=expression(R^2))  
plot(p1best.sum$adjr2,xlab="model complexity (subset size)",ylab=expression(R[a]^2))  
plot(p1best.sum$cp,xlab="model complexity (subset size)",ylab=expression(C[P]))  
plot(p1best.sum$bic,xlab="model complexity (subset size)",ylab="BIC")



*5.Identify the best model(s), over all sizes, as selected by each of the criteria, BIC, CP, and ;this will result in 1, 2, or 3 models, depending on how these criteria (dis)agree.*

***Based on BIC, the best model is y ~ x1 + x2 + x9***

***Based on CP, the best model is y ~ x1 + x2 + x6 + x9***

***Based on Ra^2, the best model is y ~ x1 + x2 + x4 + x5 + x6 + x9***

*6.Why does it not seem plausible to use the validation set approach, using MSPR, for this data set? Please be concise.*

***We only have 33 observations in the dataset with 11 parameters including the intercept. With so few degrees of freedom (n-p), there will be greater error in our model. When we do validation, the designation of training set vs. validation set leads to high variablity in the groupings depending on which observations are chosen to be in each set.***

*7.We only briefly introduced K-fold cross-validation in our notes as a more popular and modern (ML) alternative to the validation set approach to estimate generalization error and select models. Here, we conduct K-fold CV using K = n (n-fold CV, or CV(n), or LOOCV). For this homework, we perform n-fold CV. That is, K = n folds produce n training sets of size = n -1 to predict on each of the corresponding n validation sets of size = 1.*

*We’ll leave the details to the function, cv.glm, in the boot package. To use this function, we need to fit our model(s) using the function, glm, in the stats package.*

library(boot)  
eg.glm<- glm(y ~ x1 + x2 + x9, data=caterpillar\_data)  
eg.cv.n<- cv.glm(data=caterpillar\_data, glmfit=eg.glm)  
(cv.n<- eg.cv.n$delta[1]) ## CV(n)

## [1] 0.3420415

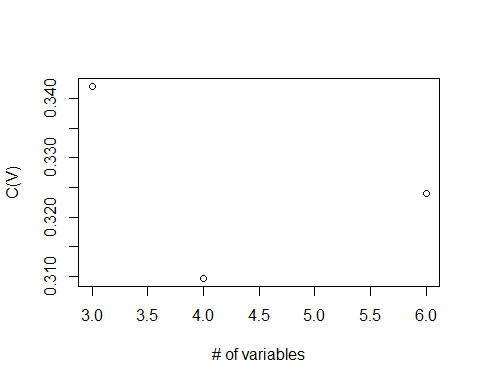
library(boot)  
eg.glm<- glm(y ~ x1 + x2 + x6 + x9, data=caterpillar\_data)  
eg.cv.n<- cv.glm(data=caterpillar\_data, glmfit=eg.glm)  
(cv.n<- eg.cv.n$delta[1]) ## CV(n)

## [1] 0.3096885

library(boot)  
eg.glm<- glm(y ~ x1 + x2 +x4 + x5+ x6 + x9, data=caterpillar\_data)  
eg.cv.n<- cv.glm(data=caterpillar\_data, glmfit=eg.glm)  
(cv.n<- eg.cv.n$delta[1]) ## CV(n)

## [1] 0.3240586

library(ggplot2)  
y <- c(0.3420415,0.3096885,0.3240586)  
x <- c(3,4,6)  
plot(x,y,xlab="# of variables",ylab = "C(V)")



*8.Discuss your findings. In particular, discuss the selection of your best model(s) using the above criteria, BIC, CP, , and CV(n). Please limit your summary to one-half page (or less).*

***No matter the best model criteria, elevation(x1), slope(x2), and vegetation strata index(x3) are powerful predictors of caterpillar nests per tree. The model with the least variables is with BIC criteria (3 variables), followed by Cp (4 variables), and then adjusted R^2 (6 variables). In addition to the base 3 powerful predictors, CP adds the variable for settlement density index (x6), while adjusted criteria also include representative tree height (x4) and representative tree diameter (x5). Based on the analysis, including the C(V) calculation, we believe the Cp criteria with 4 variables provides the best model for calculating the average number of caterpillar nests per tree.***

*9.Suppose we want to predict the (natural log of) the average number of caterpillar nests per tree () for a new observed input = .Use your best fitted model, chosen via CV(n), above, to compute the predicted value,along with a (nominal) 95% prediction interval for .To do this, use the predict function with the* ***interval=‘predict’*** *option, as illustrated in the last code chunk of our note chapter 2. (Incidentally, this assumes a normal linear model with an unbiased fitted model, which we might hope is approximately true; there are other methods to get a prediction interval in our case, but we have not talked about these methods yet.) Show your code/output and summarize briefly your prediction interval. In particular,comment on the validity of the nominal 95% coverage rate and the interval width.*

fit1<- lm(y ~ x1 + x2 + x6 + x9, data=caterpillar\_data)  
predict(fit1, data.frame(x1=1150, x2=20, x6=1.1, x9=1.5), se.fit = TRUE,interval="confidence",level=0.95)

## $fit  
## fit lwr upr  
## 1 1.739832 1.361535 2.118128  
##   
## $se.fit  
## [1] 0.1846782  
##   
## $df  
## [1] 28  
##   
## $residual.scale  
## [1] 0.5352396