Model_Selection

Import baseball database We are using statistics to predict salary of players.

```r
library(ISLR)
summary(Hitters)
```

```
##      AtBat            Hits         HmRun            Runs
##  Min.   : 16.0   Min.   :  1   Min.   : 0.00   Min.   :  0.00
##  1st Qu.:255.2   1st Qu.: 64   1st Qu.: 4.00   1st Qu.: 30.25
##  Median :379.5   Median : 96   Median : 8.00   Median : 48.00
##  Mean   :380.9   Mean   :101   Mean   :10.77   Mean   : 50.91
##  3rd Qu.:512.0   3rd Qu.:137   3rd Qu.:16.00   3rd Qu.: 69.00
##  Max.   :687.0   Max.   :238   Max.   :40.00   Max.   :130.00
##
##       RBI             Walks            Years            CAtBat
##  Min.   :  0.00   Min.   :  0.00   Min.   : 1.000   Min.   :   19.0
##  1st Qu.: 28.00   1st Qu.: 22.00   1st Qu.: 4.000   1st Qu.:  816.8
##  Median : 44.00   Median : 35.00   Median : 6.000   Median : 1928.0
##  Mean   : 48.03   Mean   : 38.74   Mean   : 7.444   Mean   : 2648.7
##  3rd Qu.: 64.75   3rd Qu.: 53.00   3rd Qu.:11.000   3rd Qu.: 3924.2
##  Max.   :121.00   Max.   :105.00   Max.   :24.000   Max.   :14053.0
##
##      CHits            CHmRun          CRuns             CRBI
##  Min.   :   4.0   Min.   :  0.00   Min.   :   1.0   Min.   :   0.00
##  1st Qu.: 209.0   1st Qu.: 14.00   1st Qu.: 100.2   1st Qu.:  88.75
##  Median : 508.0   Median : 37.50   Median : 247.0   Median : 220.50
##  Mean   : 717.6   Mean   : 69.49   Mean   : 358.8   Mean   : 330.12
##  3rd Qu.:1059.2   3rd Qu.: 90.00   3rd Qu.: 526.2   3rd Qu.: 426.25
##  Max.   :4256.0   Max.   :548.00   Max.   :2165.0   Max.   :1659.00
##
##      CWalks        League  Division    PutOuts          Assists
##  Min.   :   0.00   A:175   E:157    Min.   :   0.0   Min.   :  0.0
##  1st Qu.:  67.25   N:147   W:165    1st Qu.: 109.2   1st Qu.:  7.0
##  Median : 170.50                    Median : 212.0   Median : 39.5
##  Mean   : 260.24                    Mean   : 288.9   Mean   :106.9
##  3rd Qu.: 339.25                    3rd Qu.: 325.0   3rd Qu.:166.0
##  Max.   :1566.00                    Max.   :1378.0   Max.   :492.0
##
##      Errors          Salary       NewLeague
##  Min.   : 0.00   Min.   :  67.5   A:176
##  1st Qu.: 3.00   1st Qu.: 190.0   N:146
##  Median : 6.00   Median : 425.0
##  Mean   : 8.04   Mean   : 535.9
##  3rd Qu.:11.00   3rd Qu.: 750.0
##  Max.   :32.00   Max.   :2460.0
##                  NA's   :59
```

Remove missing values

```
#delete all rows with missing values
Hitters = na.omit(Hitters)
#check if na left
with(Hitters,sum(is.na(Salary)))
```

```
## [1] 0
```

BEST SUBSET SELECTION

go through all the predictors and select the best subset of models. For each subset size a star is put next to the important features.

```
library(leaps)
regfit.full = regsubsets(Salary~.,data=Hitters)
summary(regfit.full)
```

```
## Subset selection object
## Call: regsubsets.formula(Salary ~ ., data = Hitters)
## 19 Variables  (and intercept)
##            Forced in Forced out
## AtBat          FALSE      FALSE
## Hits           FALSE      FALSE
## HmRun          FALSE      FALSE
## Runs           FALSE      FALSE
## RBI            FALSE      FALSE
## Walks          FALSE      FALSE
## Years          FALSE      FALSE
## CAtBat         FALSE      FALSE
## CHits          FALSE      FALSE
## CHmRun         FALSE      FALSE
## CRuns          FALSE      FALSE
## CRBI           FALSE      FALSE
## CWalks         FALSE      FALSE
## LeagueN        FALSE      FALSE
## DivisionW      FALSE      FALSE
## PutOuts        FALSE      FALSE
## Assists        FALSE      FALSE
## Errors         FALSE      FALSE
## NewLeagueN     FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits CHmRun CRuns
## 1  ( 1 ) " "   " "  " "   " "  " " " "   " "   " "    " "   " "    " "
## 2  ( 1 ) " "   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "
## 3  ( 1 ) " "   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "
## 4  ( 1 ) " "   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "
## 5  ( 1 ) "*"   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "
## 6  ( 1 ) "*"   "*"  " "   " "  " " "*"   " "   " "    " "   " "    " "
## 7  ( 1 ) " "   "*"  " "   " "  " " "*"   " "   "*"    "*"   "*"    " "
## 8  ( 1 ) "*"   "*"  " "   " "  " " "*"   " "   " "    " "   "*"    "*"
##            CRBI CWalks LeagueN DivisionW PutOuts Assists Errors NewLeagueN
## 1  ( 1 ) "*"  " "    " "     " "       " "     " "     " "    " "
## 2  ( 1 ) "*"  " "    " "     " "       " "     " "     " "    " "
## 3  ( 1 ) "*"  " "    " "     " "       "*"     " "     " "    " "
## 4  ( 1 ) "*"  " "    " "     "*"       "*"     " "     " "    " "
## 5  ( 1 ) "*"  " "    " "     "*"       "*"     " "     " "    " "
## 6  ( 1 ) "*"  " "    " "     "*"       "*"     " "     " "    " "
## 7  ( 1 ) " "  " "    " "     "*"       "*"     " "     " "    " "
## 8  ( 1 ) " "  "*"    " "     "*"       "*"     " "     " "    " "
```

Default subset size is 8 but we can push it up to 19 (as many variables as we have) Cp = prediction error Pick the model with the minimum Cp

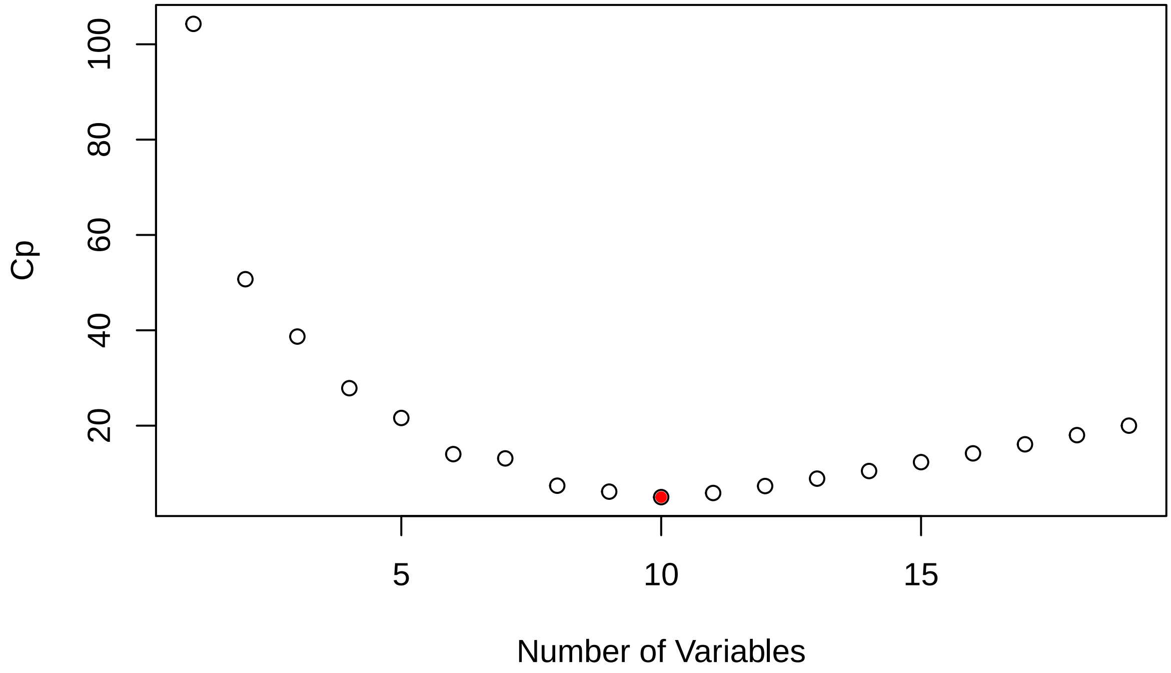
```
regfit.full=regsubsets(Salary~.,data=Hitters, nvmax=19)
reg.summary=summary(regfit.full)
names(reg.summary)
```

```
## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
```

```
plot(reg.summary$cp,xlab="Number of Variables",ylab="Cp")
which.min(reg.summary$cp)
```

```
## [1] 10
```

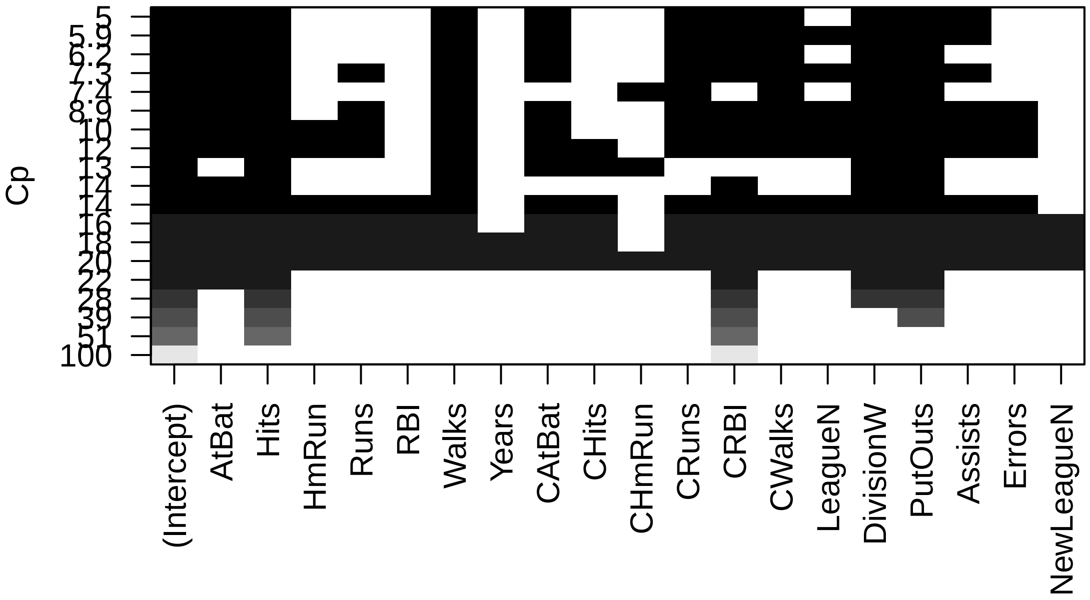
```
points(10,reg.summary$cp[10],pch=20,col="red")
```



There is a particular library to plot these graphs

Black indicates in variables and white squares are out

```
plot(regfit.full,scale="Cp")
```



```
#coefficients of the 10th model
coef(regfit.full,10)
```

```
##   (Intercept)         AtBat          Hits         Walks        CAtBat
##   162.5354420    -2.1686501     6.9180175     5.7732246    -0.1300798
##         CRuns          CRBI        CWalks     DivisionW       PutOuts
##     1.4082490     0.7743122    -0.8308264  -112.3800575     0.2973726
##       Assists
##     0.2831680
```

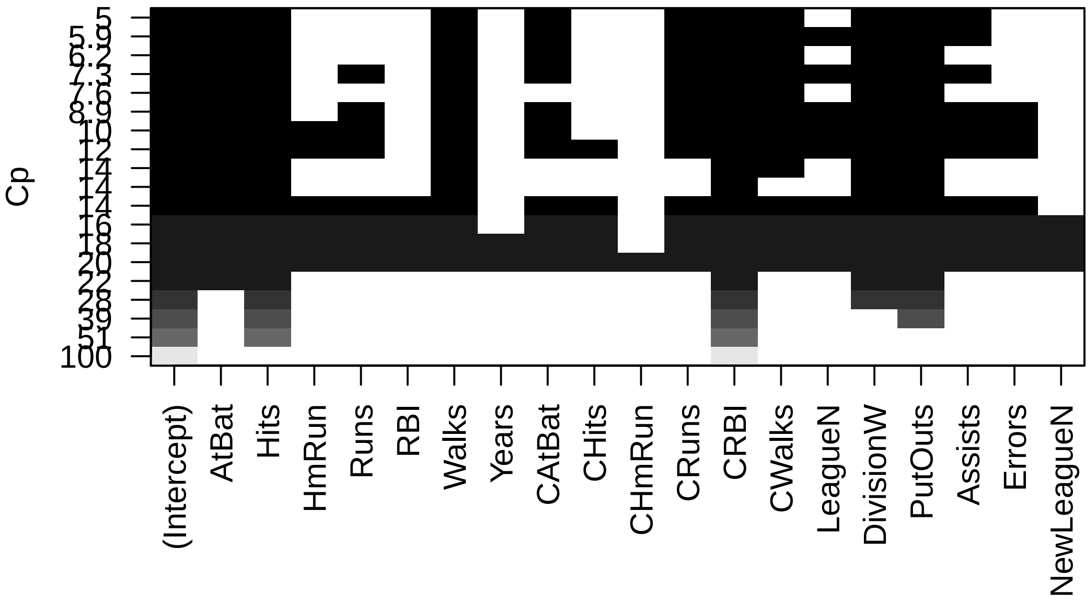
FORWARD STEPWISE SELECTION

Use regsubset with method=forward

```
regfit.fwd=regsubsets(Salary~.,data=Hitters,nvmax=19,method="forward")
summary(regfit.fwd)
```

```
## Subset selection object
## Call: regsubsets.formula(Salary ~ ., data = Hitters, nvmax = 19, method = "forward")
## 19 Variables  (and intercept)
##            Forced in Forced out
## AtBat          FALSE      FALSE
## Hits           FALSE      FALSE
## HmRun          FALSE      FALSE
## Runs           FALSE      FALSE
## RBI            FALSE      FALSE
## Walks          FALSE      FALSE
## Years          FALSE      FALSE
## CAtBat         FALSE      FALSE
## CHits          FALSE      FALSE
## CHmRun         FALSE      FALSE
## CRuns          FALSE      FALSE
## CRBI           FALSE      FALSE
## CWalks         FALSE      FALSE
## LeagueN        FALSE      FALSE
## DivisionW      FALSE      FALSE
## PutOuts        FALSE      FALSE
## Assists        FALSE      FALSE
## Errors         FALSE      FALSE
## NewLeagueN     FALSE      FALSE
## 1 subsets of each size up to 19
## Selection Algorithm: forward
##           AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits CHmRun CRuns
## 1  ( 1 )  " "   " "  " "   " "  " " " "   " "   " "    " "   " "    " "
## 2  ( 1 )  " "   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "
## 3  ( 1 )  " "   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "
## 4  ( 1 )  " "   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "
## 5  ( 1 )  "*"   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "
## 6  ( 1 )  "*"   "*"  " "   " "  " " "*"   " "   " "    " "   " "    " "
## 7  ( 1 )  "*"   "*"  " "   " "  " " "*"   " "   " "    " "   " "    " "
## 8  ( 1 )  "*"   "*"  " "   " "  " " "*"   " "   " "    " "   " "    "*"
## 9  ( 1 )  "*"   "*"  " "   " "  " " "*"   " "   "*"    " "   " "    "*"
## 10  ( 1 ) "*"   "*"  " "   " "  " " "*"   " "   "*"    " "   " "    "*"
## 11  ( 1 ) "*"   "*"  " "   " "  " " "*"   " "   "*"    " "   " "    "*"
## 12  ( 1 ) "*"   "*"  " "   "*"  " " "*"   " "   "*"    " "   " "    "*"
## 13  ( 1 ) "*"   "*"  " "   "*"  " " "*"   " "   "*"    " "   " "    "*"
## 14  ( 1 ) "*"   "*"  "*"   "*"  " " "*"   " "   "*"    " "   " "    "*"
## 15  ( 1 ) "*"   "*"  "*"   "*"  " " "*"   " "   "*"    "*"   " "    "*"
## 16  ( 1 ) "*"   "*"  "*"   "*"  "*" "*"   " "   "*"    "*"   " "    "*"
## 17  ( 1 ) "*"   "*"  "*"   "*"  "*" "*"   " "   "*"    "*"   " "    "*"
## 18  ( 1 ) "*"   "*"  "*"   "*"  "*" "*"   "*"   "*"    "*"   " "    "*"
## 19  ( 1 ) "*"   "*"  "*"   "*"  "*" "*"   "*"   "*"    "*"   "*"    "*"
##           CRBI CWalks LeagueN DivisionW PutOuts Assists Errors NewLeagueN
## 1  ( 1 )  "*"  " "    " "     " "       " "     " "     " "    " "
## 2  ( 1 )  "*"  " "    " "     " "       " "     " "     " "    " "
## 3  ( 1 )  "*"  " "    " "     " "       "*"     " "     " "    " "
## 4  ( 1 )  "*"  " "    " "     "*"       "*"     " "     " "    " "
## 5  ( 1 )  "*"  " "    " "     "*"       "*"     " "     " "    " "
## 6  ( 1 )  "*"  " "    " "     "*"       "*"     " "     " "    " "
## 7  ( 1 )  "*"  "*"    " "     "*"       "*"     " "     " "    " "
## 8  ( 1 )  "*"  "*"    " "     "*"       "*"     " "     " "    " "
## 9  ( 1 )  "*"  "*"    " "     "*"       "*"     " "     " "    " "
## 10  ( 1 ) "*"  "*"    " "     "*"       "*"     "*"     " "    " "
## 11  ( 1 ) "*"  "*"    "*"     "*"      "*"     "*"     " "    " "
## 12  ( 1 ) "*"  "*"    "*"     "*"      "*"     "*"     " "    " "
## 13  ( 1 ) "*"  "*"    "*"     "*"      "*"     "*"     "*"    " "
## 14  ( 1 ) "*"  "*"    "*"     "*"      "*"     "*"     "*"    " "
## 15  ( 1 ) "*"  "*"    "*"     "*"      "*"     "*"     "*"    " "
## 16  ( 1 ) "*"  "*"    "*"     "*"      "*"     "*"     "*"    " "
## 17  ( 1 ) "*"  "*"    "*"     "*"      "*"     "*"     "*"    "*"
## 18  ( 1 ) "*"  "*"    "*"     "*"      "*"     "*"     "*"    "*"
## 19  ( 1 ) "*"  "*"    "*"     "*"      "*"     "*"     "*"    "*"
```

```
plot(regfit.fwd,scale="Cp")
```



Model Selection Using a Validation Set

Make a training and validation set, so that we can choose a good subset model.

```
dim(Hitters)
```

```
## [1] 263  20
```

```
set.seed(1)
#seq creates a sequence from 1 to n
```

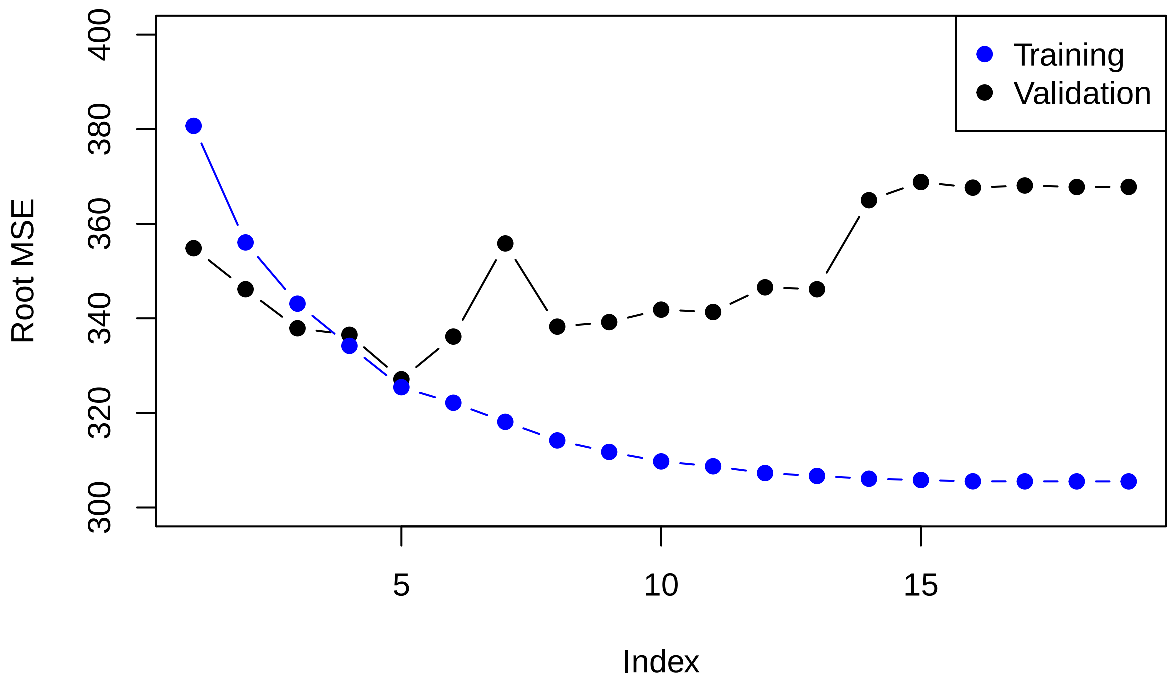
```
#180 is indexes of observations
train=sample(seq(263),180,replace=FALSE)
train
```

```
##   [1]  70  98 150 237  53 232 243 170 161  16 259  45 173  97 192 124 178
##  [18] 245  94 190 228  52 158  31  64  92   4  91 205  80 113 140 115  43
##  [35] 244 153 181  25 163  93 184 144 174 122 117 251   6 104 241 149 102
##  [52] 183 224 242  15  21  66 107 136  83 186  60 211  67 130 210  95 151
##  [69]  17 256 207 162 200 239 236 168 249  73 222 177 234 199 203  59 235
##  [86]  37 126  22 230 226  42  11 110 214 132 134  77  69 188 100 206  58
## [103]  44 159 101  34 208  75 185 201 261 112  54  65  23   2 106 254 257
## [120] 154 142  71 166 221 105  63 143  29 240 212 167 172   5  84 120 133
## [137]  72 191 248 138 182  74 179 135  87 196 157 119  13  99 263 125 247
## [154]  50  55  20  57   8  30 194 139 238  46  78  88  41   7  33 141  32
## [171] 180 164 213  36 215  79 225 229 198  76
```

```
regfit.fwd=regsubsets(Salary~.,data=Hitters[train,],nvmax=19,method="forward")
```

there are 19 models, so we set up some vectors to record the errors

```
val.errors=rep(NA,19)
#create a matrix where training data is removed from set
x.test=model.matrix(Salary~.,data=Hitters[-train,])
#for all the parameters
for(i in 1:19){
  #size of sample is i
  coefi=coef(regfit.fwd,id=i)
  pred=x.test[,names(coefi)]%*%coefi
  val.errors[i]=mean((Hitters$Salary[-train]-pred)^2)
}
plot(sqrt(val.errors),ylab="Root MSE",ylim=c(300,400),pch=19,type="b")
points(sqrt(regfit.fwd$rss[-1]/180),col="blue",pch=19,type="b")
legend("topright",legend=c("Training","Validation"),col=c("blue","black"),pch=19)
```



model prediction method for regsubset

```
predict.regsubsets=function(object,newdata,id,...){
  form=as.formula(object$call[[2]])
  mat=model.matrix(form,newdata)
  coefi=coef(object,id=id)
  mat[,names(coefi)]%*%coefi
}
```

MODEL SELECTION WITH CROSS VALIDATION

10 fold cross validation

```
set.seed(10)
#take 10 samples
folds = sample(rep(1:10,length=nrow(Hitters)))
folds
```

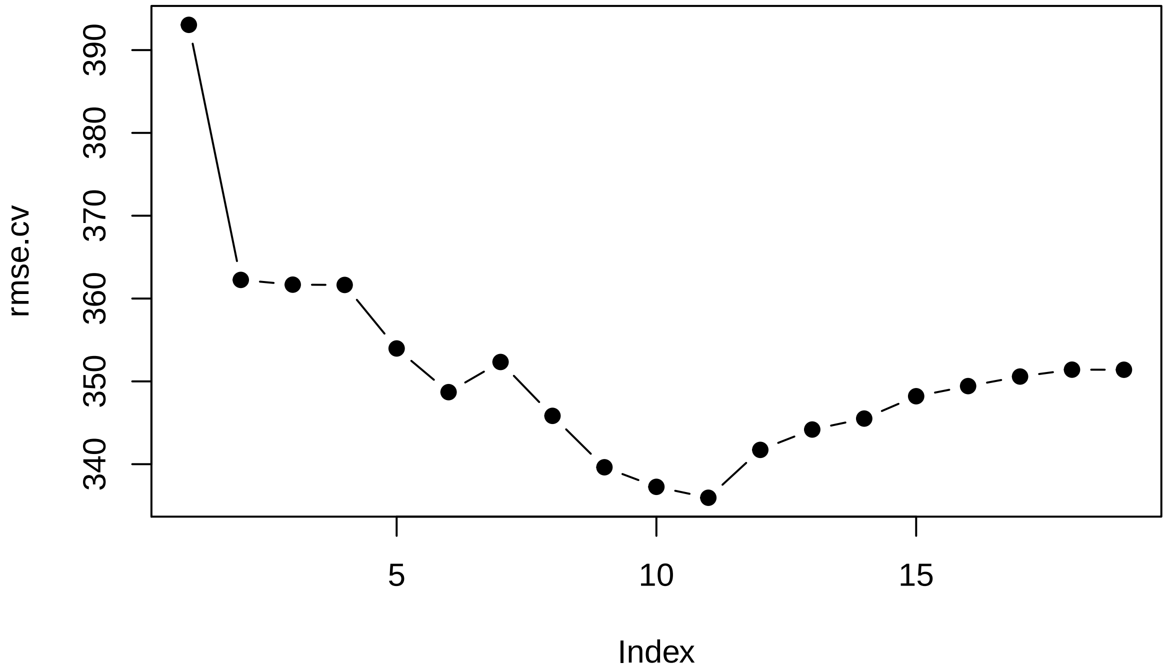
```
##   [1]  4  1  2  1  3  9  1 10  8 10  5  4  9  9 10  7  3  5  8  5  1 10  7
##  [24]  6  7  9  9  7  2  4  5  2 10  7  9  1  1  6  5  3  2  1  4  1  5  6
##  [47]  1  5  9  2  6  7  2 10  1  2  6  9  2  4  7  4  5 10  6  1  4  8  7
##  [70]  3  5  3  3 10  8  2  6  9  8  9  5  5  9  7  7  7  2  9 10  5  5  1
##  [93]  2  6  6  1  2  9  4  5  6 10 10  6  6  8  5  5  8 10 10  9  8 10  6
## [116]  6  4  6  2  1  6  5  6 10  1  1  7  7  9  8  9  3  2  4 10  6  1  9
## [139]  4  9 10  9  2  1  8  5  6  2  9  4 10  5  5  3  3 10  3  8  9  1  6
## [162]  3  7  1  3  6  8  1  4  6  8  7  8 10 10  1  4  1  9 10  8  7  5  8
## [185] 10  7  7  3  9  3  6  3  8  6  3  6  3  9  8  3  3  4  2  2  9  7  4
## [208]  4  3  8  7  2  8 10  3  7  3  5  7  5  4  3  8  7  7  8  4  9  1  4
## [231]  4  6  7  2  1  3  5  3 10  3  8 10  2  2  1  5  7  2  2  4  8  8  4
## [254]  2  5  8  4  2  1  2  4  4  3
```

```r
table(folds)
```

```
## folds
##  1  2  3  4  5  6  7  8  9 10
## 27 27 27 26 26 26 26 26 26 26
```

```r
#make a matrix to store errors for each model on a fold
cv.errors=matrix(NA,10,19)

#two loops
for(afold in 1:10){
  #fit a regsubset model
  best.fit=regsubsets(Salary~.,data=Hitters[folds!=afold,],nvmax=19,method="forward")
  for(param in 1:19){
    #predict the best fit from selected
    pred=predict(best.fit,Hitters[folds==afold,],id=param)
    cv.errors[afold,param]=mean( (Hitters$Salary[folds==afold]-pred)^2)
  }
}
rmse.cv=sqrt(apply(cv.errors,2,mean))
plot(rmse.cv,pch=19,type="b")
```



its not as jittery as the validation test curve