HW8 Wenxiao Yang

Problem 1

(a)

iii Lasso Regression is find the $\hat{\beta}$ such that minimize $(y-X\beta)^T(y-X\beta)+\lambda\sum_j|\beta_j|$ or $(y-X\beta)^T(y-X\beta)$ subject to $\sum_j|\beta_j|\leq t$ We can find it is less flexible, and it will elimate some $\beta_i=0$ compare to OLS i.e. less predictors in this model. Less predictors will increase model bias and decrease variance. So Lasso will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

(b)

iii Ridge Regression is find the $\hat{\beta}$ such that minimize $(y-X\beta)^T(y-X\beta)+\lambda\sum_j\beta_j^2$ or $(y-X\beta)^T(y-X\beta)$ subject to $\sum_j\beta_j^2\leq t^2$ We can find it is less flexible, and it will control the β_i^2 not be too high, which will decrease variance but increase model bias. So Ridge will give improved prediction accuracy when its increase in bias is less than its decrease in variance.

Problem 2

```
library(ISLR)
data(College)
head(College)
```

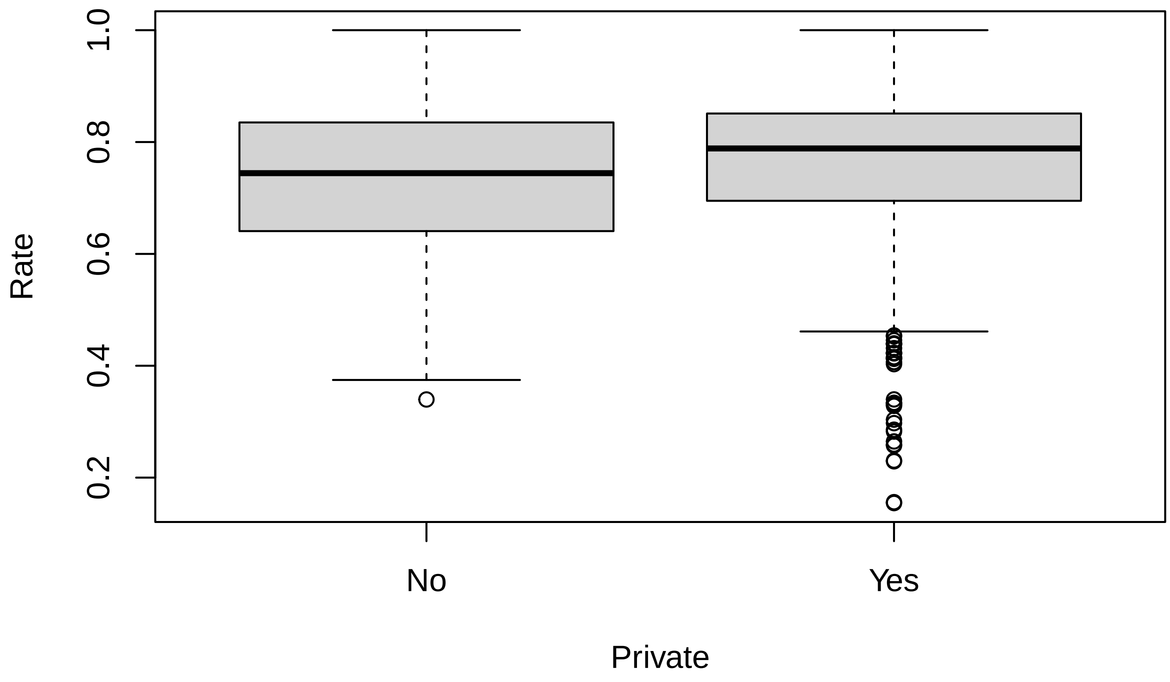
```
##                              Private Apps Accept Enroll Top10perc Top25perc
## Abilene Christian University     Yes 1660   1232    721        23        52
## Adelphi University               Yes 2186   1924    512        16        29
## Adrian College                   Yes 1428   1097    336        22        50
## Agnes Scott College              Yes  417    349    137        60        89
## Alaska Pacific University        Yes  193    146     55        16        44
## Albertson College                Yes  587    479    158        38        62
##                              F.Undergrad P.Undergrad Outstate Room.Board Books
## Abilene Christian University        2885         537     7440       3300   450
## Adelphi University                  2683        1227    12280       6450   750
## Adrian College                      1036          99    11250       3750   400
## Agnes Scott College                  510          63    12960       5450   450
## Alaska Pacific University            249         869     7560       4120   800
## Albertson College                    678          41    13500       3335   500
##                              Personal PhD Terminal S.F.Ratio perc.alumni Expend
## Abilene Christian University     2200  70       78      18.1          12   7041
## Adelphi University               1500  29       30      12.2          16  10527
## Adrian College                   1165  53       66      12.9          30   8735
## Agnes Scott College               875  92       97       7.7          37  19016
## Alaska Pacific University        1500  76       72      11.9           2  10922
## Albertson College                 675  67       73       9.4          11   9727
##                              Grad.Rate
## Abilene Christian University        60
## Adelphi University                  56
## Adrian College                      54
## Agnes Scott College                 59
## Alaska Pacific University           15
## Albertson College                   55
```

(a)

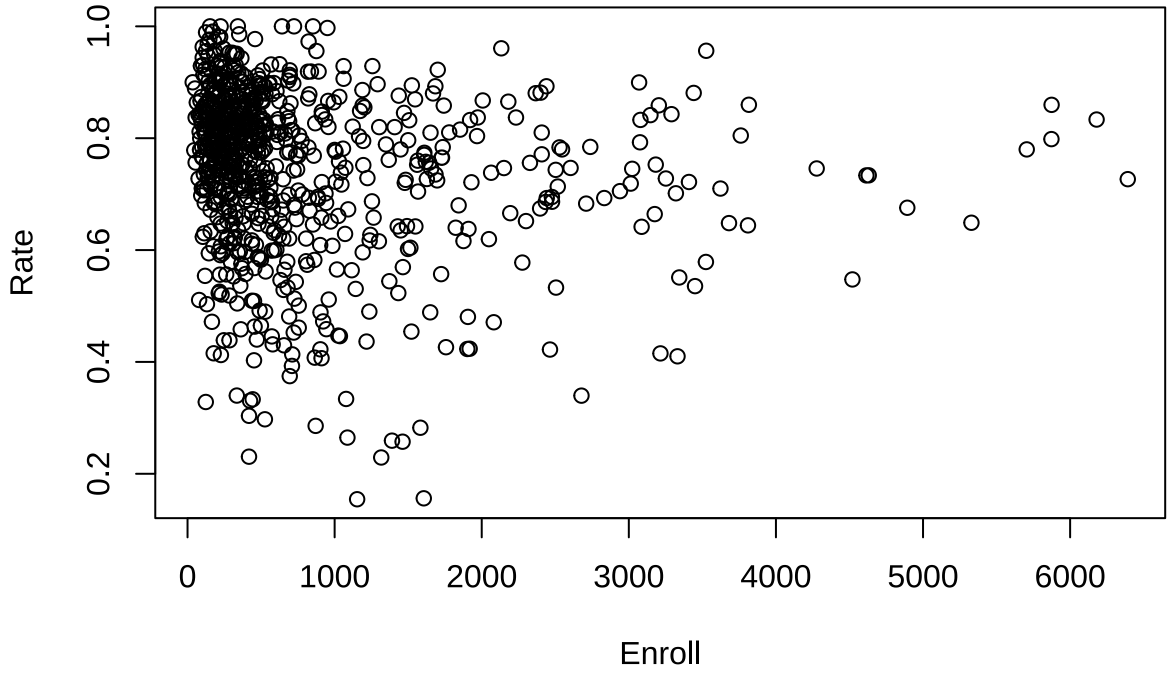
```r
College["Rate"]=College["Accept"]/College["Apps"]
head(College)
```

```
##                              Private Apps Accept Enroll Top10perc Top25perc
## Abilene Christian University     Yes 1660   1232    721        23        52
## Adelphi University               Yes 2186   1924    512        16        29
## Adrian College                   Yes 1428   1097    336        22        50
## Agnes Scott College              Yes  417    349    137        60        89
## Alaska Pacific University        Yes  193    146     55        16        44
## Albertson College                Yes  587    479    158        38        62
##                              F.Undergrad P.Undergrad Outstate Room.Board Books
## Abilene Christian University        2885         537     7440       3300   450
## Adelphi University                  2683        1227    12280       6450   750
## Adrian College                      1036          99    11250       3750   400
## Agnes Scott College                  510          63    12960       5450   450
## Alaska Pacific University            249         869     7560       4120   800
## Albertson College                    678          41    13500       3335   500
##                              Personal PhD Terminal S.F.Ratio perc.alumni Expend
## Abilene Christian University     2200  70       78      18.1          12   7041
## Adelphi University               1500  29       30      12.2          16  10527
## Adrian College                   1165  53       66      12.9          30   8735
## Agnes Scott College               875  92       97       7.7          37  19016
## Alaska Pacific University        1500  76       72      11.9           2  10922
## Albertson College                 675  67       73       9.4          11   9727
##                              Grad.Rate      Rate
## Abilene Christian University        60 0.7421687
## Adelphi University                  56 0.8801464
## Adrian College                      54 0.7682073
## Agnes Scott College                 59 0.8369305
## Alaska Pacific University           15 0.7564767
## Albertson College                   55 0.8160136
```

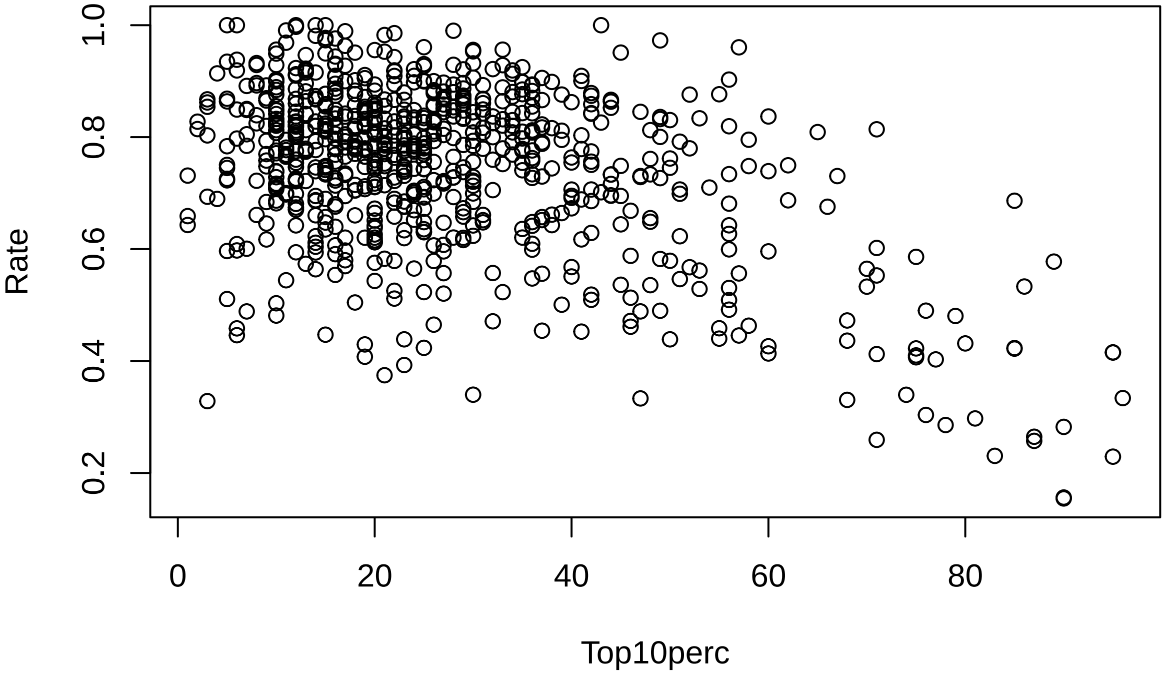
```r
plot(College$Private,College$Rate,xlab="Private",ylab="Rate")
```



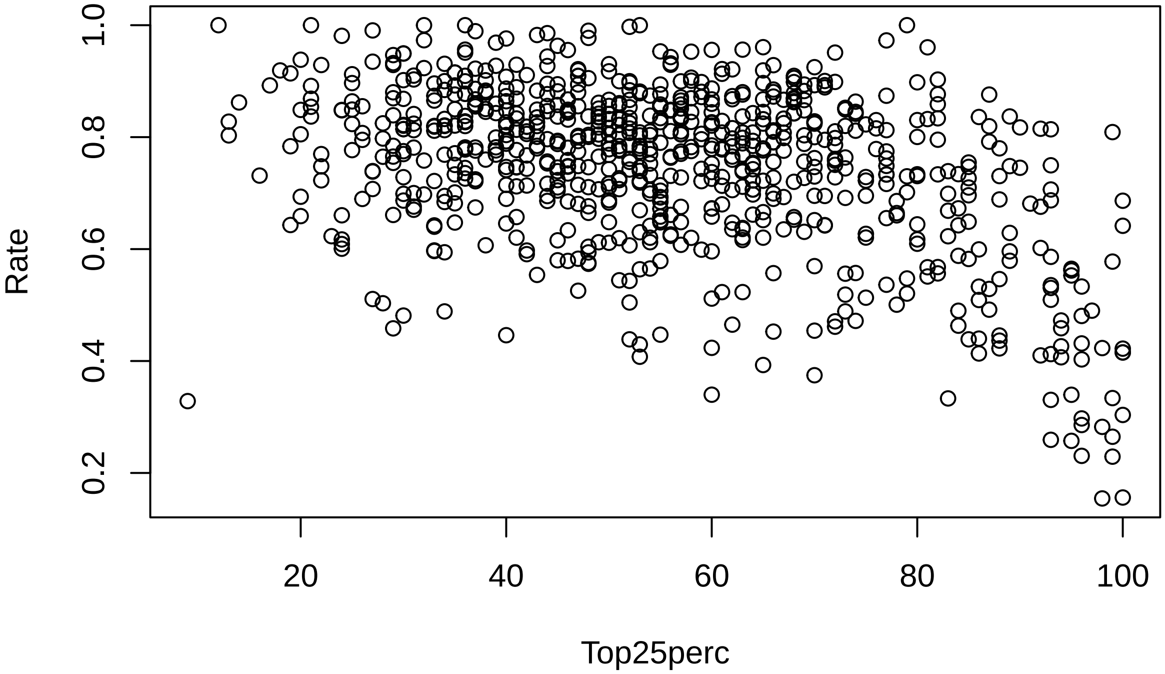
```
plot(College$Enroll,College$Rate,xlab="Enroll",ylab="Rate")
```



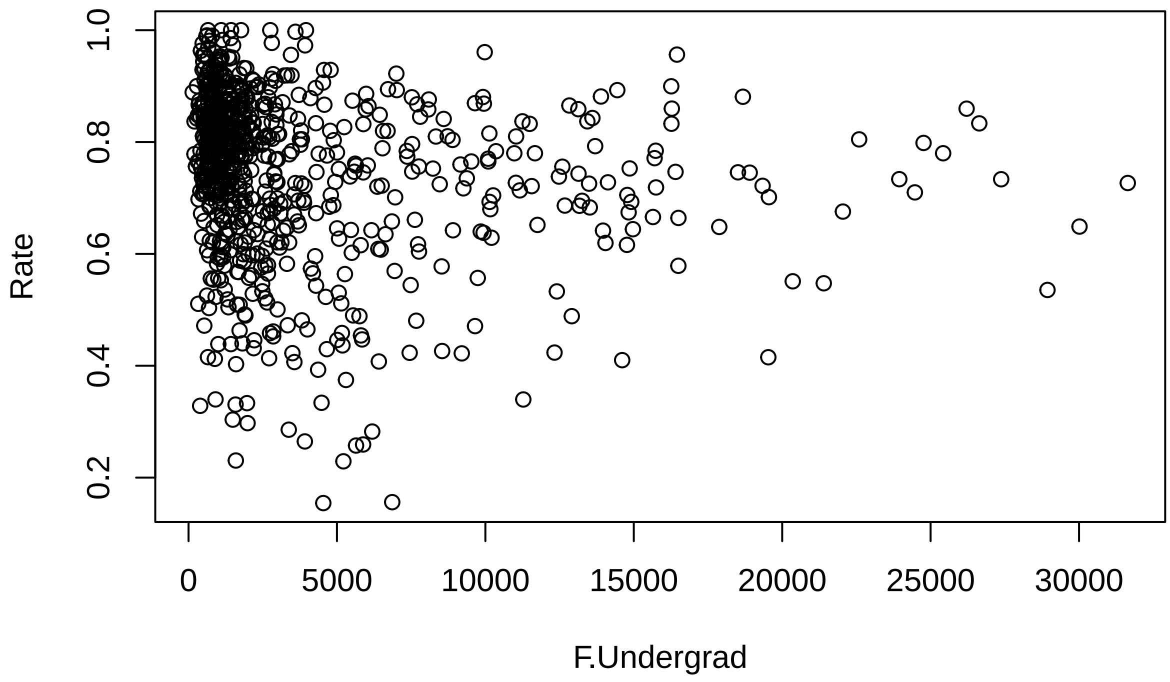
```
plot(College$Top10perc,College$Rate,xlab="Top10perc",ylab="Rate")
```



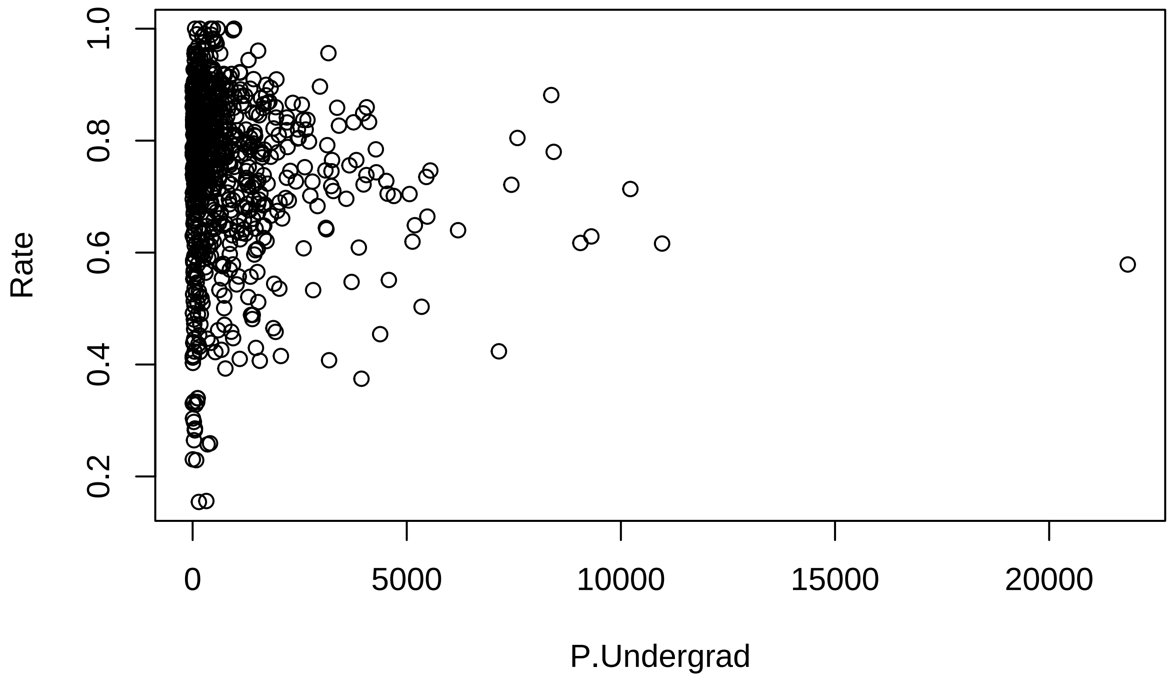
```
plot(College$Top25perc,College$Rate,xlab="Top25perc",ylab="Rate")
```



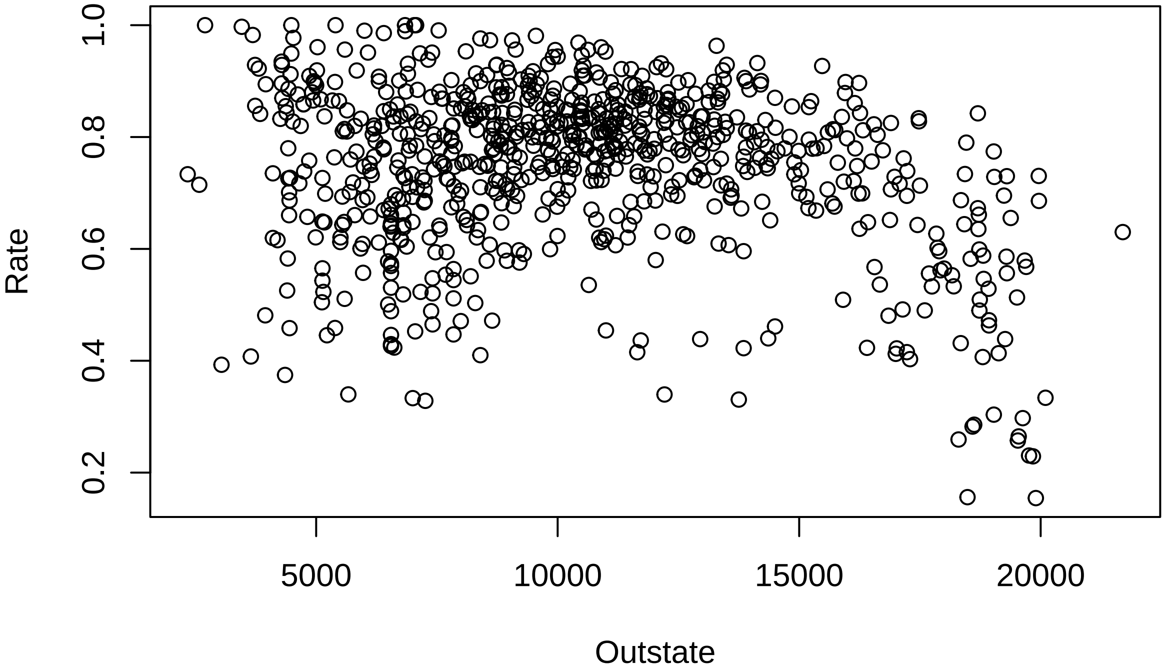
```
plot(College$F.Undergrad,College$Rate,xlab="F.Undergrad",ylab="Rate")
```



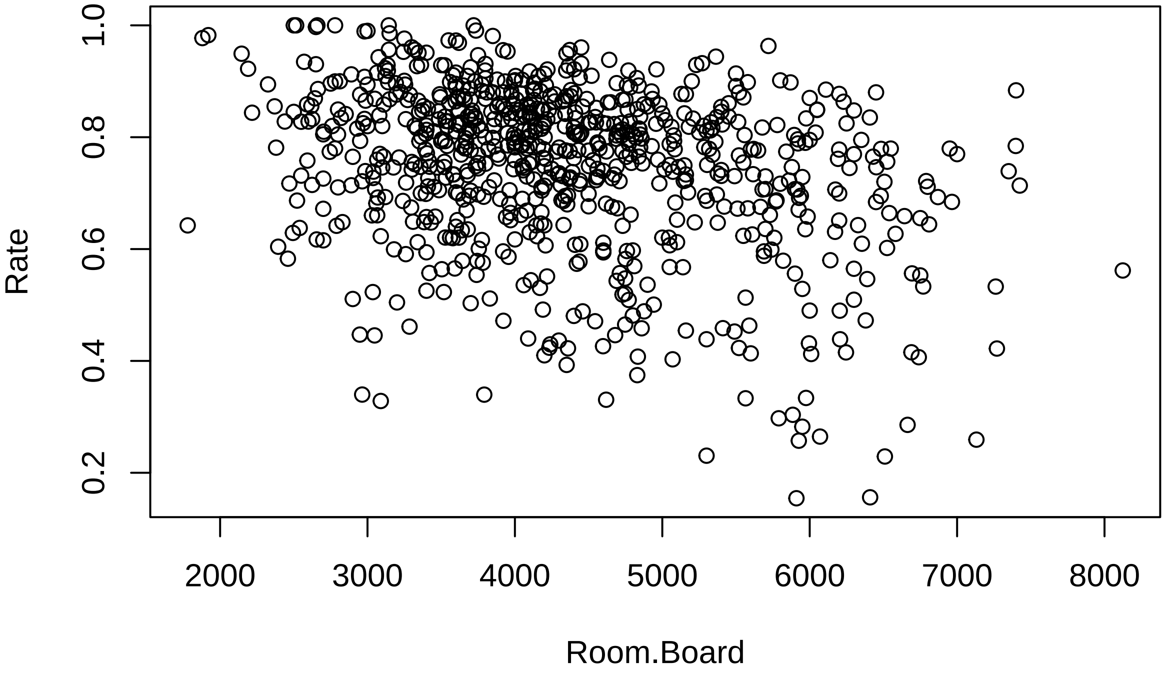
```
plot(College$P.Undergrad,College$Rate,xlab="P.Undergrad",ylab="Rate")
```



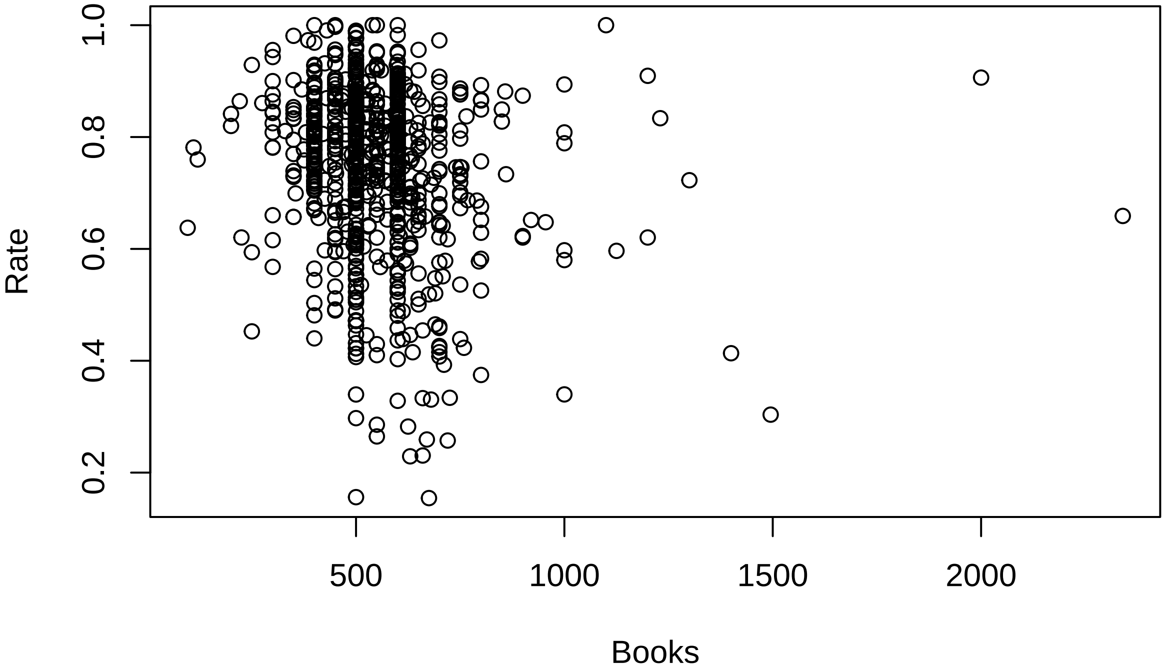
```
plot(College$Outstate,College$Rate,xlab="Outstate",ylab="Rate")
```



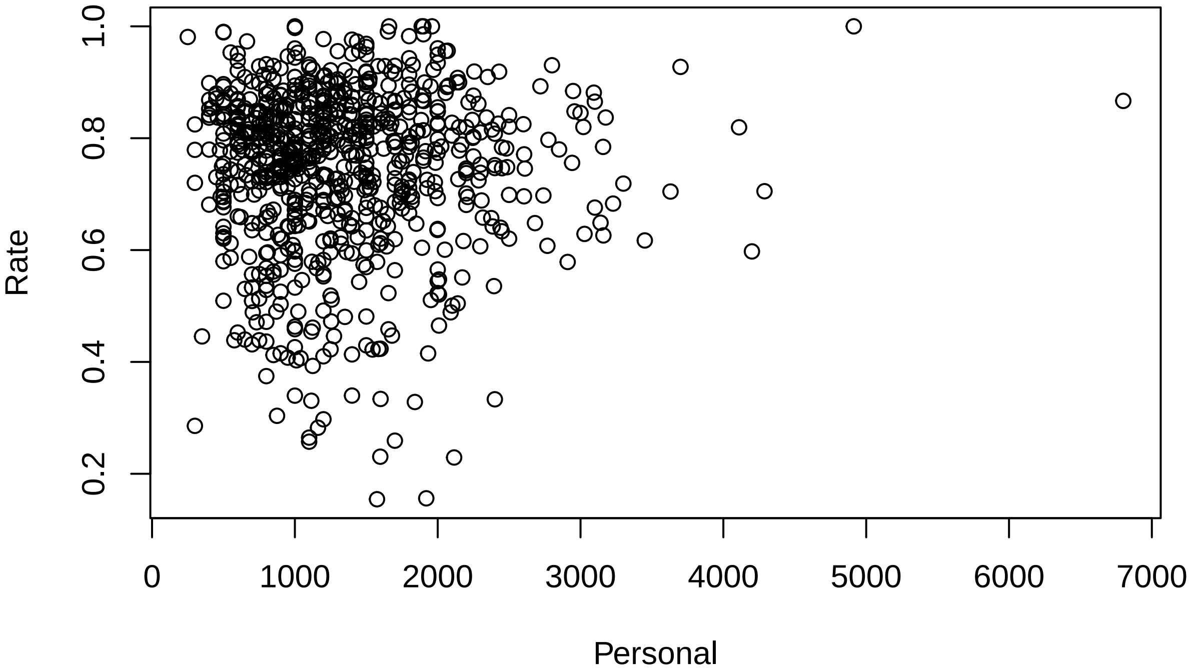
```
plot(College$Room.Board,College$Rate,xlab="Room.Board",ylab="Rate")
```



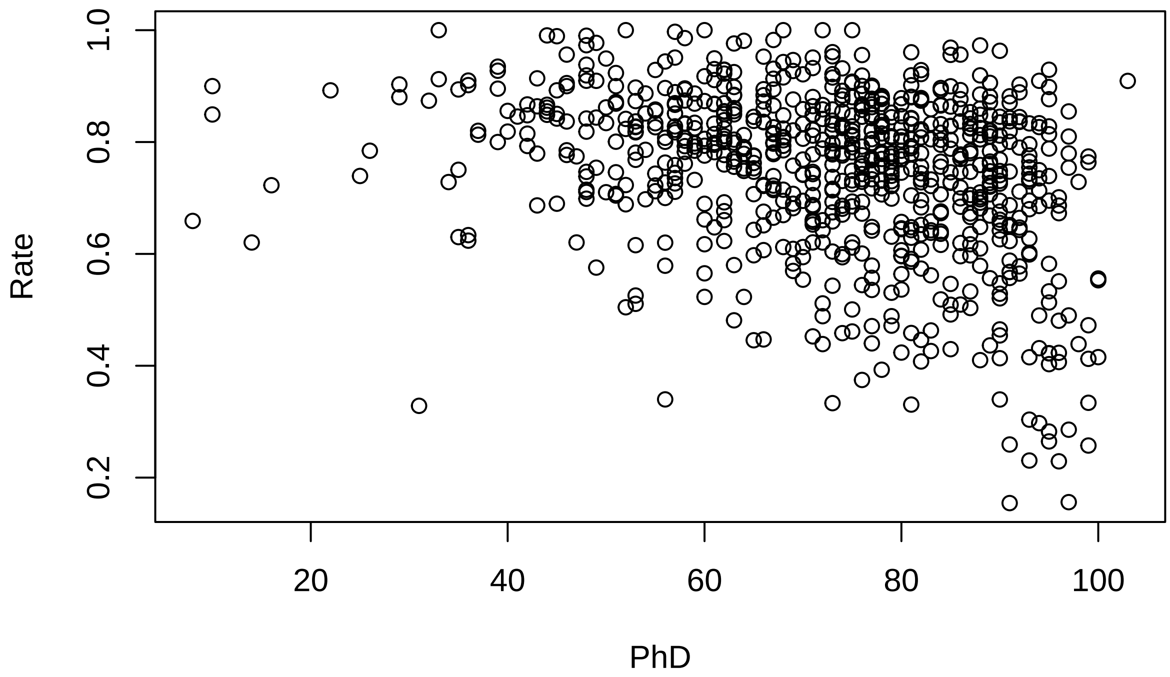
```
plot(College$Books,College$Rate,xlab="Books",ylab="Rate")
```



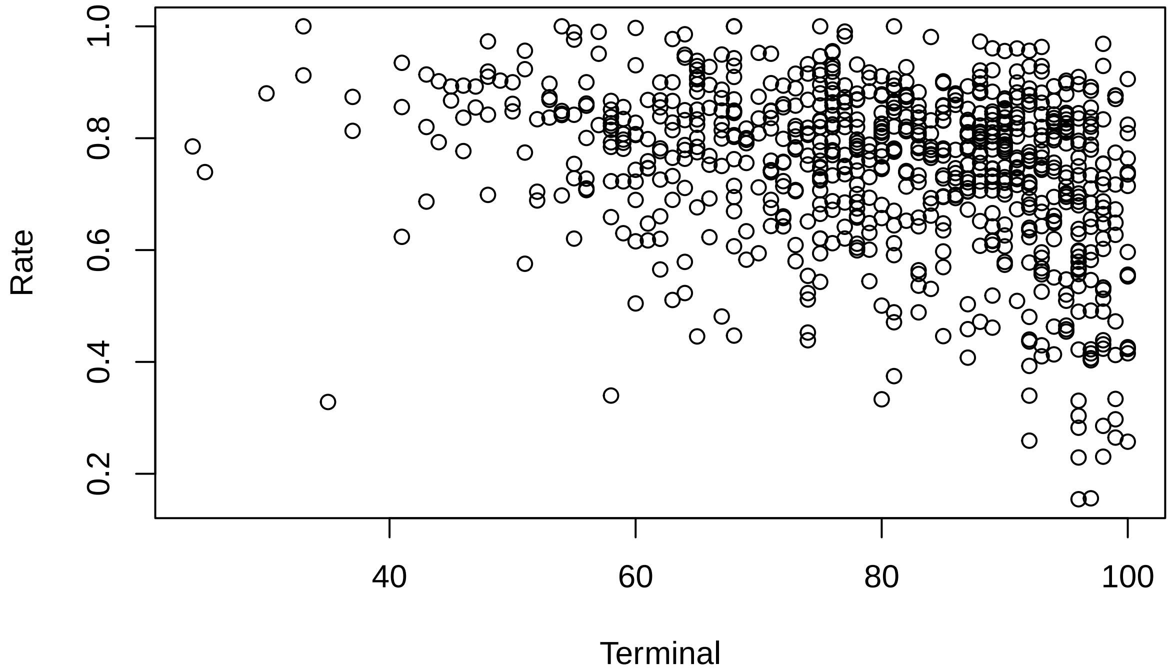
```
plot(College$Personal,College$Rate,xlab="Personal",ylab="Rate")
```



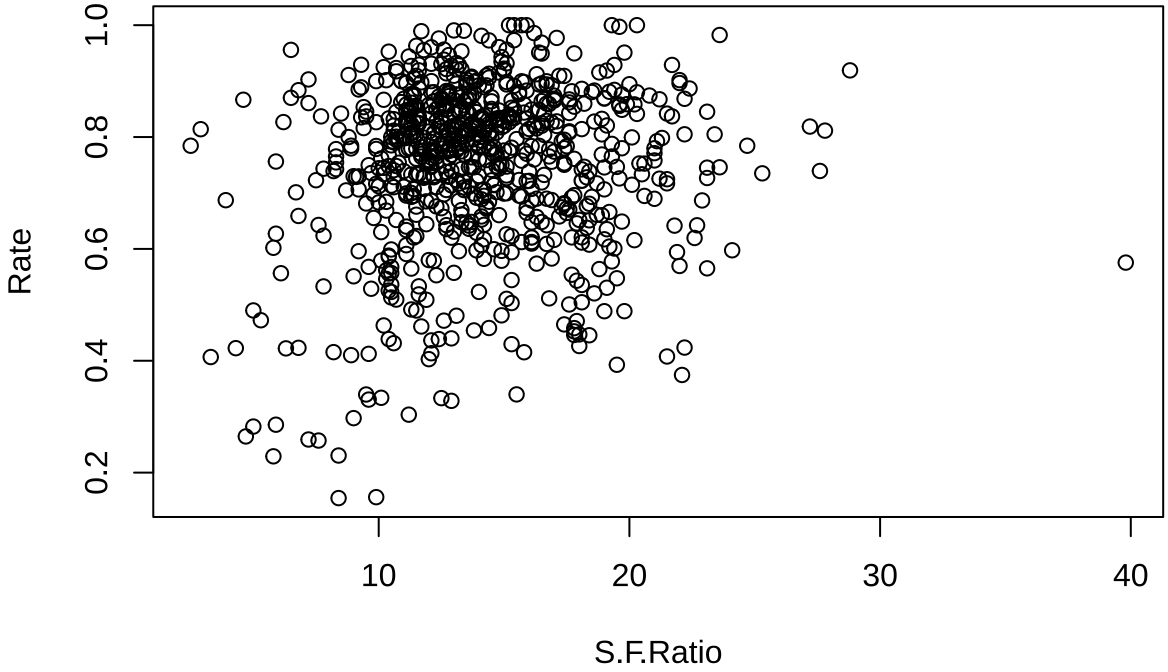
```
plot(College$PhD,College$Rate,xlab="PhD",ylab="Rate")
```



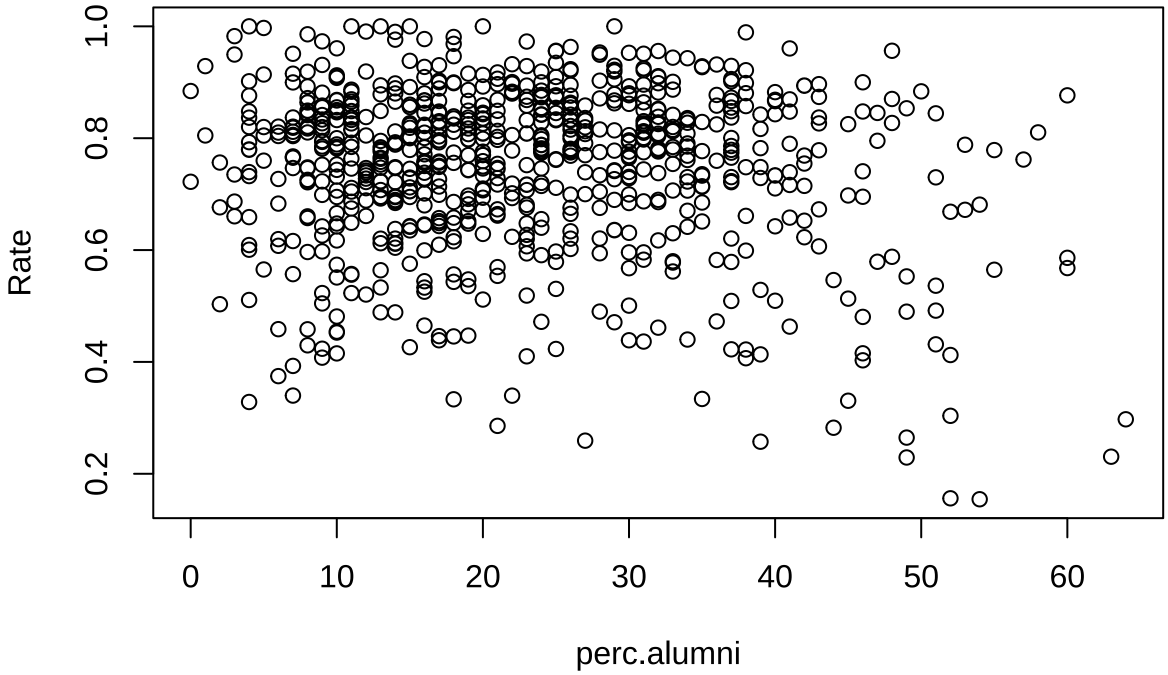
```
plot(College$Terminal,College$Rate,xlab="Terminal",ylab="Rate")
```



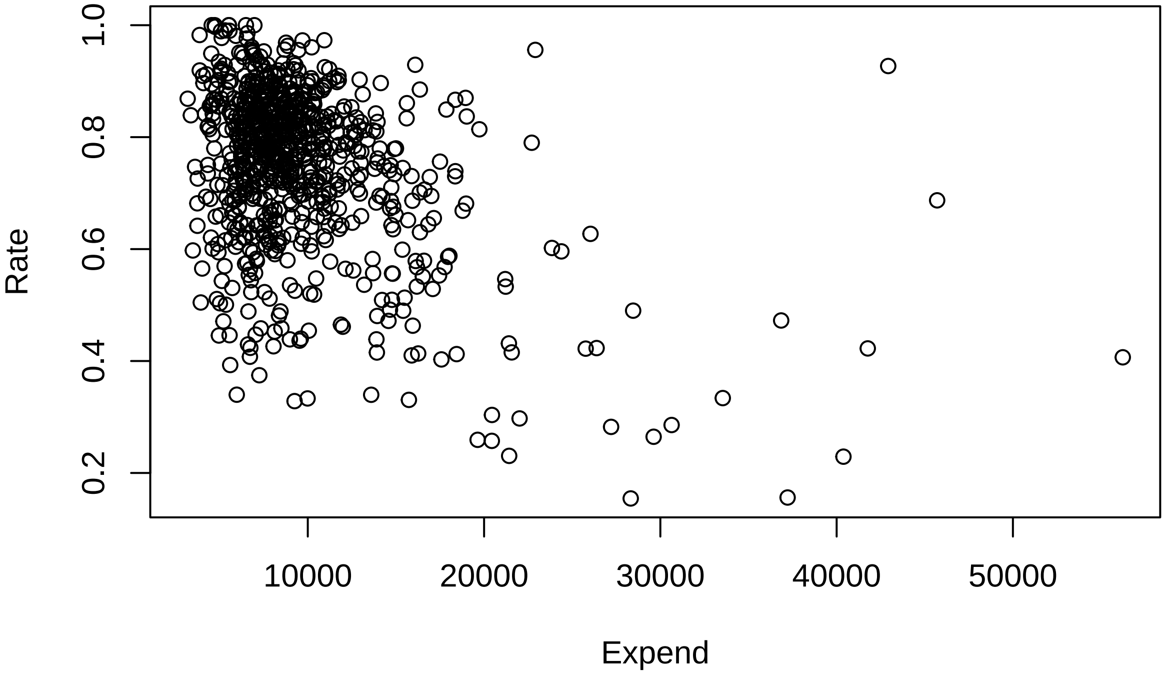
```
plot(College$S.F.Ratio,College$Rate,xlab="S.F.Ratio",ylab="Rate")
```



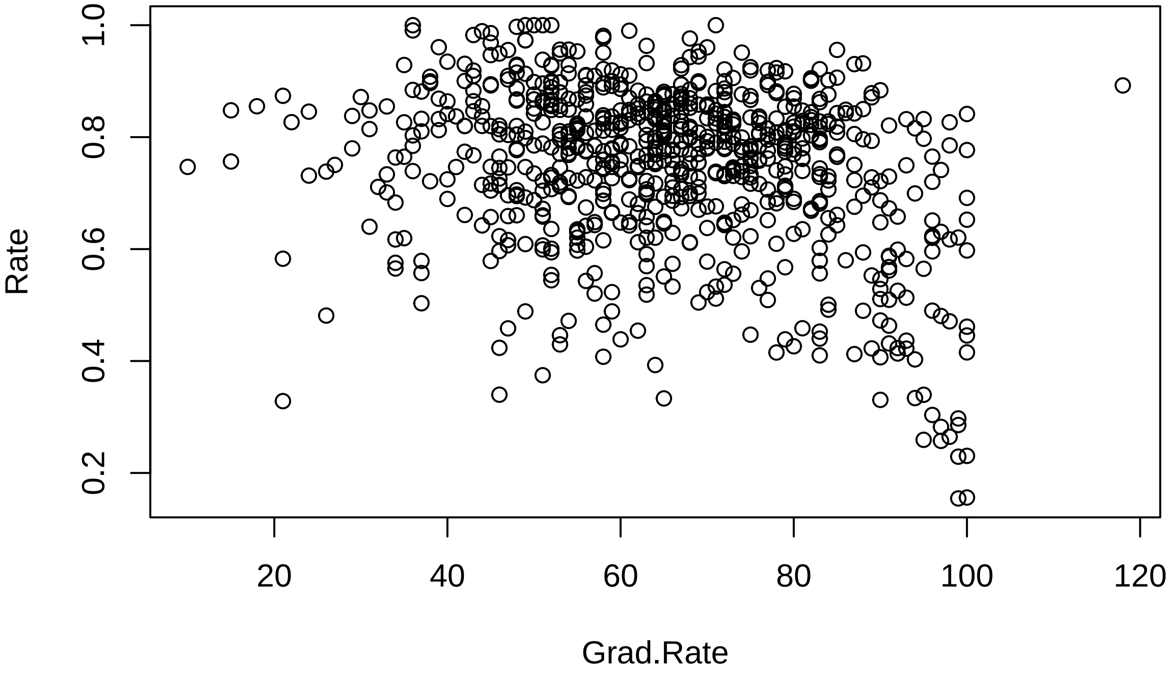
```
plot(College$perc.alumni,College$Rate,xlab="perc.alumni",ylab="Rate")
```



```
plot(College$Expend,College$Rate,xlab="Expend",ylab="Rate")
```



```
plot(College$Grad.Rate,College$Rate,xlab="Grad.Rate",ylab="Rate")
```



```
set.seed(425)
n=dim(College)[1]
train.index=sample(n,0.7*n)
data.training=College[train.index,c(-2,-3)]
data.testing=College[-train.index,c(-2,-3)]
```

(b)

```
lm.fit=lm(Rate~.,data=data.training)
mean(lm.fit$res^2)
```

```
## [1] 0.01450403
```

MSE of training data is 0.01450403

```
lm.pre=predict(lm.fit,data=data.testing)
mean((data.testing$Rate-lm.pre)^2)
```

```
## Warning in data.testing$Rate - lm.pre: longer object length is not a multiple of
## shorter object length
```

```
## [1] 0.02839502
```

MSE of testing data is 0.02839502

(c)

```
library(leaps)
C.leaps=regsubsets(Rate~.,data=College[,c(-2,-3)],nvmax=16)
rs=summary(C.leaps)
rs
```

```
## Subset selection object
## Call: regsubsets.formula(Rate ~ ., data = College[, c(-2, -3)], nvmax = 16)
## 16 Variables  (and intercept)
##             Forced in Forced out
## PrivateYes      FALSE      FALSE
## Enroll          FALSE      FALSE
## Top10perc       FALSE      FALSE
## Top25perc       FALSE      FALSE
## F.Undergrad     FALSE      FALSE
## P.Undergrad     FALSE      FALSE
## Outstate        FALSE      FALSE
## Room.Board      FALSE      FALSE
## Books           FALSE      FALSE
## Personal        FALSE      FALSE
## PhD             FALSE      FALSE
## Terminal        FALSE      FALSE
## S.F.Ratio       FALSE      FALSE
## perc.alumni     FALSE      FALSE
## Expend          FALSE      FALSE
## Grad.Rate       FALSE      FALSE
## 1 subsets of each size up to 16
## Selection Algorithm: exhaustive
##           PrivateYes Enroll Top10perc Top25perc F.Undergrad P.Undergrad
## 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 ) "*"        "*"    "*"       "*"       "*"         "*"
##           Outstate Room.Board Books Personal PhD Terminal S.F.Ratio perc.alumni
## 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 ) "*"      "*"        "*"   "*"      "*" "*"      "*"       "*"
##           Expend Grad.Rate
## 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 ) "*"    "*"
```

adjusted R^2

```
rs$adjr2
```

```
##  [1] 0.2281355 0.2547006 0.2951031 0.3039859 0.3109264 0.3175412 0.3242976
##  [8] 0.3302143 0.3374867 0.3440776 0.3449889 0.3448224 0.3446644 0.3445000
## [15] 0.3439173 0.3430753
```

```
rs$which[which.max(rs$adjr2),]
```

```
## (Intercept)  PrivateYes      Enroll   Top10perc   Top25perc F.Undergrad
##        TRUE        TRUE        TRUE        TRUE       FALSE       FALSE
## P.Undergrad    Outstate  Room.Board       Books    Personal         PhD
##        TRUE        TRUE        TRUE        TRUE       FALSE       FALSE
##    Terminal   S.F.Ratio perc.alumni      Expend   Grad.Rate
##       FALSE        TRUE        TRUE        TRUE        TRUE
```

The model adjusted R^2 chooses

$$Rate = \beta_0+\beta_1 PrivateYes+\beta_2 Enroll+\beta_3 Top10perc+\beta_4 P.Undergrad+\beta_5 Outstate+\beta_6 Room.Board+\beta_7 Books+\beta_8 S.F.Ratio$$

```
lm.fit1=lm(Rate~.,data=data.training[,c(-4,-5,-10,-11,-12)])
mean(lm.fit1$res^2)
```

```
## [1] 0.01462862
```

MSE of training data is 0.01462862

```
lm.pre1=predict(lm.fit1,data=data.testing)
mean((data.testing$Rate-lm.pre1)^2)
```

```
## Warning in data.testing$Rate - lm.pre1: longer object length is not a multiple
## of shorter object length

## [1] 0.0283374
```

MSE of testing data is 0.0283374

AIC

```
m=2:17
Aic=n*log(rs$rss/n)+2*m
Aic
```

```
##  [1] -3177.625 -3203.841 -3246.151 -3255.011 -3261.805 -3268.308 -3275.049
##  [8] -3280.893 -3288.388 -3295.171 -3295.266 -3294.085 -3292.915 -3291.739
## [15] -3290.069 -3288.094
```

```
rs$which[which.min(Aic),]
```

```
## (Intercept)  PrivateYes      Enroll   Top10perc   Top25perc F.Undergrad
##        TRUE        TRUE        TRUE        TRUE       FALSE       FALSE
## P.Undergrad    Outstate  Room.Board       Books    Personal         PhD
##        TRUE        TRUE        TRUE        TRUE       FALSE       FALSE
##    Terminal   S.F.Ratio perc.alumni      Expend   Grad.Rate
##       FALSE        TRUE        TRUE        TRUE        TRUE
```

The model AIC chooses

$$Rate = \beta_0+\beta_1 PrivateYes+\beta_2 Enroll+\beta_3 Top10perc+\beta_4 P.Undergrad+\beta_5 Outstate+\beta_6 Room.Board+\beta_7 Books+\beta_8 S.F.Ratio$$

Which is same as the model derived by adjusted R^2, so MSE of training data is 0.01462862 MSE of testing data is 0.0283374

BIC

```
rs$bic
```

```
##  [1] -188.8923 -210.4529 -248.1077 -252.3118 -254.4504 -256.2982 -258.3833
##  [8] -259.5726 -262.4121 -264.5390 -259.9788 -254.1423 -248.3172 -242.4859
## [15] -236.1604 -229.5300
```

```
rs$which[which.min(rs$bic),]
```

```
## (Intercept)  PrivateYes      Enroll   Top10perc   Top25perc F.Undergrad
##        TRUE        TRUE        TRUE        TRUE       FALSE       FALSE
## P.Undergrad    Outstate  Room.Board       Books    Personal         PhD
##        TRUE        TRUE        TRUE        TRUE       FALSE       FALSE
##    Terminal   S.F.Ratio perc.alumni      Expend   Grad.Rate
##       FALSE        TRUE       FALSE        TRUE        TRUE
```

The model BIC chooses

$$Rate = \beta_0+\beta_1 PrivateYes+\beta_2 Enroll+\beta_3 Top10perc+\beta_4 P.Undergrad+\beta_5 Outstate+\beta_6 Room.Board+\beta_7 Books+\beta_8 S.F.Ratio$$

```
lm.fit2=lm(Rate~.,data=data.training[,c(-4,-5,-10,-11,-12,-14)])
mean(lm.fit2$res^2)
```

```
## [1] 0.0146697
```

MSE of training data is 0.0146697

```
lm.pre2=predict(lm.fit2,data=data.testing)
mean((data.testing$Rate-lm.pre2)^2)
```

```
## Warning in data.testing$Rate - lm.pre2: longer object length is not a multiple
## of shorter object length
```

```
## [1] 0.02836103
```

MSE of testing data is 0.02836103

(d) Ridge

```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-3
```

```
ridge.fit=cv.glmnet(model.matrix(Rate~., data=data.training), data.training$Rate,nfolds=10, alpha=0)
ridge.lambda=ridge.fit$lambda.min
ridge.lambda
```

```
## [1] 0.01240732
```

```
ridge.fit
```

```
##
## Call:  cv.glmnet(x = model.matrix(Rate ~ ., data = data.training), y = data.training$Rate,      nfol
##
## Measure: Mean-Squared Error
##
##      Lambda Index Measure       SE Nonzero
## min 0.01241    94 0.01569 0.001143      16
## 1se 0.18424    65 0.01682 0.001057      16
```

Cross-validated: MSE of $\lambda = 0.0103$ is 0.001123

```
train.ridge.pred=predict(ridge.fit,s=ridge.lambda,newx=model.matrix(Rate~., data=data.training))
mean((train.ridge.pred-data.training$Rate)^2)
```

```
## [1] 0.01466132
```

Training: MSE is 0.01463213

```
test.ridge.pred=predict(ridge.fit,s=ridge.lambda,newx=model.matrix(Rate~., data=data.testing))
mean((test.ridge.pred-data.testing$Rate)^2)
```

```
## [1] 0.01283903
```

Testing: MSE is 0.01281851

(e) lasso

```
lasso.fit=cv.glmnet(model.matrix(Rate~., data=data.training), data.training$Rate,nfolds=10, alpha=1)
lasso.lambda=lasso.fit$lambda.min
lasso.lambda
```

```
## [1] 0.001079125
```

```
lasso.fit
```

```
##
## Call:  cv.glmnet(x = model.matrix(Rate ~ ., data = data.training), y = data.training$Rate,      nfol
##
## Measure: Mean-Squared Error
##
##       Lambda Index Measure        SE Nonzero
## min 0.001079    46 0.01555 0.0009641      14
## 1se 0.011045    21 0.01643 0.0011548       6
```

Cross-validated: MSE of $\lambda = 0.000168$ is 0.001143

```
train.lasso.pred=predict(lasso.fit,s=lasso.lambda,newx=model.matrix(Rate~., data=data.training))
mean((train.lasso.pred-data.training$Rate)^2)
```

```
## [1] 0.01459302
```

Training: MSE is 0.01450827

```
test.lasso.pred=predict(lasso.fit,s=lasso.lambda,newx=model.matrix(Rate~., data=data.testing))
mean((test.lasso.pred-data.testing$Rate)^2)
```

```
## [1] 0.01281276
```

Testing: MSE is 0.01283959

(f)PCR

```
library(pls)
```

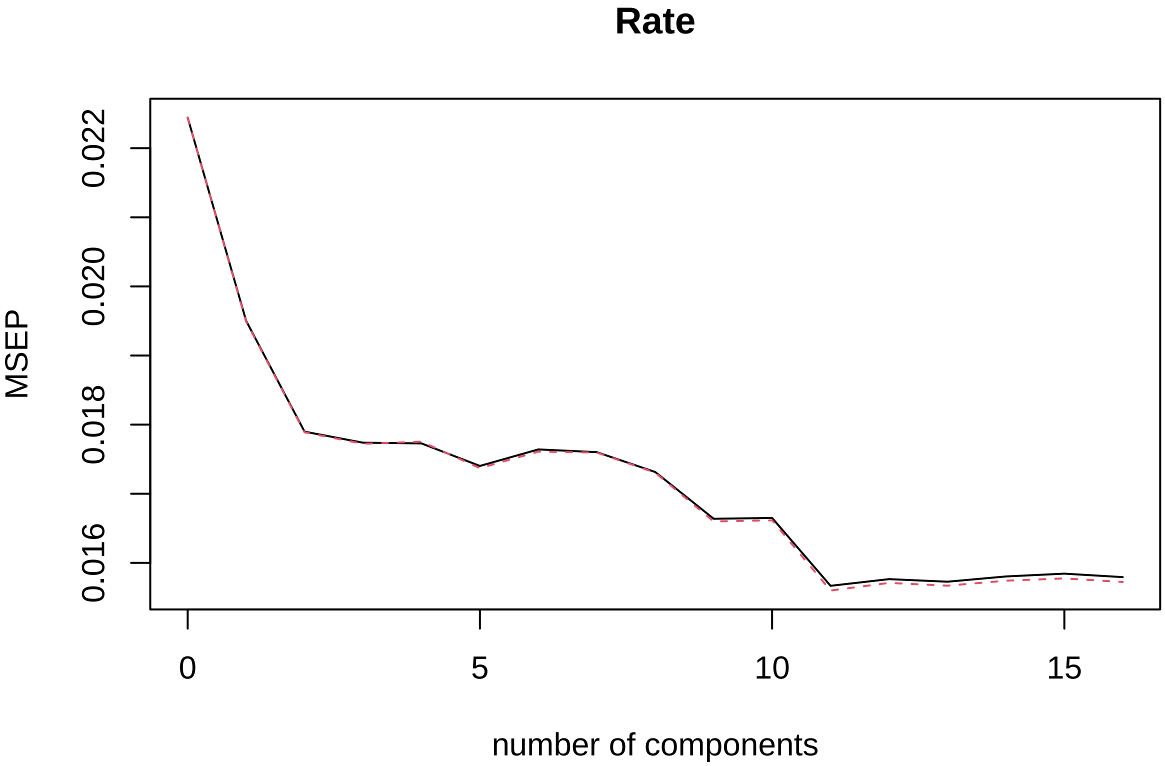
```
##
## Attaching package: 'pls'
```

```
## The following object is masked from 'package:stats':
##
##     loadings
```

```
pcr.fit=pcr(Rate~.,data=data.training,scale=TRUE,validation="CV")
summary(pcr.fit)
```

```
## Data:    X dimension: 543 16
##  Y dimension: 543 1
## Fit method: svdpc
## Number of components considered: 16
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV          0.1498   0.1397   0.1338   0.1332   0.1331   0.1319   0.1328
## adjCV       0.1498   0.1396   0.1337   0.1331   0.1332   0.1318   0.1327
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV      0.1327   0.1316   0.1290    0.1290    0.1252    0.1256    0.1254
## adjCV   0.1326   0.1316   0.1288    0.1289    0.1249    0.1253    0.1252
##        14 comps  15 comps  16 comps
## CV       0.1257    0.1259    0.1257
## adjCV    0.1255    0.1256    0.1254
##
## TRAINING: % variance explained
##       1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
## X       34.09    56.78    64.16    69.90    75.37    80.05    83.84    87.55
## Rate    13.35    20.76    22.12    22.36    24.02    24.02    24.38    25.42
##       9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps
## X       90.63     93.21     95.36     97.29     98.44     99.33     99.84
## Rate    29.07     29.39     33.86     33.89     34.28     34.28     34.70
##       16 comps
## X       100.00
## Rate     35.13
```

```
validationplot(pcr.fit,val.type="MSEP")
```



I will use 12 components accoring to the plot. M=12

```
pcr.pred=predict(pcr.fit,data.testing,ncomp=12)
mean((pcr.pred-data.testing$Rate)^2)
```

```
## [1] 0.01287599
```

The MSE of test is 0.01287599.

(g)

```
T=mean((mean(data.testing$Rate)-data.testing$Rate)^2)
```

Adjusted R^2, AIC

```
1-mean((data.testing$Rate-lm.pre1)^2)/T
```

```
## Warning in data.testing$Rate - lm.pre1: longer object length is not a multiple
## of shorter object length
```

```
## [1] -0.4309355
```

BIC

```
1-mean((data.testing$Rate-lm.pre2)^2)/T
```

```
## Warning in data.testing$Rate - lm.pre2: longer object length is not a multiple
## of shorter object length
```

```
## [1] -0.4321288
```

Ridge

```
1-mean((test.ridge.pred-data.testing$Rate)^2)/T
```

```
## [1] 0.3516755
```

Lasso

```
1-mean((test.lasso.pred-data.testing$Rate)^2)/T
```

```
## [1] 0.3530021
```

PCR

```
1-mean((pcr.pred-data.testing$Rate)^2)/T
```

```
## [1] 0.3498095
```

We can explain 0.352712 at most. All methods don't work well. I would recommend Ridge which is relative good.