MSCA 31008 Assignment 1

Duo Zhou

7/5/2020

```
#Loading Data
dataPath <- "C:/Users/zd000/Desktop/MSCA/Data Mining/Assignments/week1"
GC <-read.csv(paste(dataPath, 'GermanCredit.csv', sep = '/'), header = TRUE)
GC<-GC[,-1]
```

Use GermanCredit data [Canvas or from UCI learning site or pkg Caret R Build a regression model to predict variable "Amount" as a function of other variables (choose variables that you think are necessary and required to build a good model)

```
# fit a linear regression with all variables (full model)
full.model <- lm(GC$Amount~.,data=GC)
(full.model.r.square <- summary(full.model)$r.squared)
```

```
## [1] 0.6105089
```

```
# fit linear regression with only intercept (null model)
null.model <- lm(GC$Amount~1,data=GC)
(null.model.r.square <- summary(null.model)$r.square)
```

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

Use step function to find the model with the best AIC using backwards, forwards and both Directions

```
best.AIC.model.forward <- step(null.model,scope=formula(full.model),direction="forward",trace=FALSE)
best.AIC.model.backward <- step(full.model,direction="backward",trace=FALSE)
best.AIC.model.both <- step(full.model,direction="both",trace=FALSE)
```

```
# Exact AIC and Number of predictor + Intercept
extractAIC(best.AIC.model.forward )
```

```
## [1]    18.00 15001.14
```

```r
extractAIC(best.AIC.model.backward )
```

```
## [1]    29.00 15016.99
```

```r
extractAIC(best.AIC.model.both)
```

```
## [1]    29.00 15016.99
```

The forward method gives the best AIC and 18 predictors. The Other two generates 29 predictors. I picked forward method model. Since it is required that we select variables with no fewer than 20 coefficients, I added two more variables from the original data: Foreign.Work and Purpose.NewCar

```r
summary(best.AIC.model.forward)
```

```
##
## Call:
## lm(formula = GC$Amount ~ Duration + InstallmentRatePercentage +
##     Job.Management.SelfEmp.HighlyQualified + Personal.Male.Single +
##     Telephone + Purpose.UsedCar + Purpose.Other + Class + Property.Unknown +
##     CheckingAccountStatus.gt.200 + CreditHistory.NoCredit.AllPaid +
##     SavingsAccountBonds.Unknown + OtherDebtorsGuarantors.CoApplicant +
##     Purpose.Radio.Television + CheckingAccountStatus.lt.0 + Property.RealEstate +
##     EmploymentDuration.gt.7, data = GC)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5286.2 -1000.5  -144.7   714.3 10599.1
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          3118.121    259.723  12.006  < 2e-16 ***
## Duration                              124.428      5.175  24.046  < 2e-16 ***
## InstallmentRatePercentage            -799.107     52.790 -15.137  < 2e-16 ***
## Job.Management.SelfEmp.HighlyQualified 1312.673    180.179   7.285 6.58e-13 ***
## Personal.Male.Single                  491.871    119.591   4.113 4.23e-05 ***
## Telephone                            -486.386    128.067  -3.798 0.000155 ***
## Purpose.UsedCar                       739.836    201.734   3.667 0.000258 ***
## Purpose.Other                        1782.341    542.951   3.283 0.001064 **
## ClassGood                            -365.361    137.124  -2.664 0.007838 **
## Property.Unknown                      432.586    171.459   2.523 0.011794 *
## CheckingAccountStatus.gt.200         -669.439    237.967  -2.813 0.005004 **
## CreditHistory.NoCredit.AllPaid        808.174    296.328   2.727 0.006499 **
## SavingsAccountBonds.Unknown           398.435    151.107   2.637 0.008502 **
## OtherDebtorsGuarantors.CoApplicant    633.940    292.270   2.169 0.030321 *
## Purpose.Radio.Television             -226.119    134.039  -1.687 0.091927 .
## CheckingAccountStatus.lt.0           -240.180    135.027  -1.779 0.075589 .
## Property.RealEstate                  -219.683    135.823  -1.617 0.106110
## EmploymentDuration.gt.7              -207.189    136.633  -1.516 0.129741
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1793 on 982 degrees of freedom
## Multiple R-squared:  0.6034, Adjusted R-squared:  0.5965
## F-statistic: 87.88 on 17 and 982 DF,  p-value: < 2.2e-16
```

Choose all the variables list above in the forward method model plus Foreign.Work and Purpose.NewCar.

```
#GC1<- GC[,c(1,2,3,8,10,11,13,15,20,21,22,23,24,25,26,28,29,31,32,33,39,43,47,49,50,51,59,60,61)]
GC2<- GC[,c(1,2,3,62,43,8,21,30,10,52,13,15,35,47,23,11,49,39,9,20)]
extractAIC(lm(Amount~.,data=GC2))
```

```
## [1]    20.00 15003.96
```

We can see that with 20 variables, the model still generates less AIC than backward and both direction models.

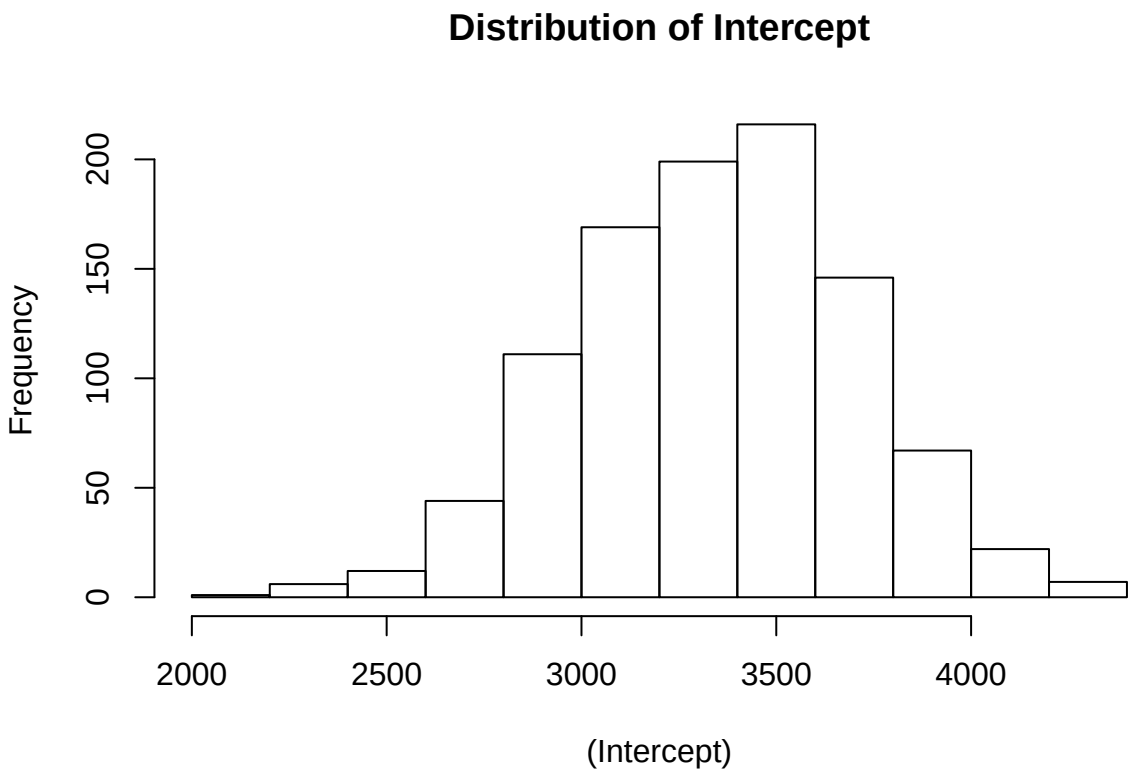
Split the sample randomly into training-test using a 632:368 ratio, and compute the coefficient and r-squares in training and test samples. Run the process 1000 times and save the results.

```
rsquare_train <- matrix(NA,1000)
rsquare_test <- matrix(NA,1000)
coefficients <- matrix(NA,20,1000)
set.seed(1234)
for (i in 1:1000){
  train_ind <- sample(nrow(GC2), size = 0.632 * nrow(GC2))
  train <- GC2[train_ind, ]
  test <- GC2[-train_ind, ]
  fit.lm <- lm(train$Amount~.,data=train)
  coefficients[,i] <- coef(fit.lm)
  rsquare_train[i] <- summary(fit.lm)$r.squared
  predited.value <- predict(fit.lm,newdata=test,type="response")
  rsquare_test[i] <- cor(test$Amount,predited.value)^2
}
```

Plot the distributions of all coefficients, test R^2, and % fall in R^2.

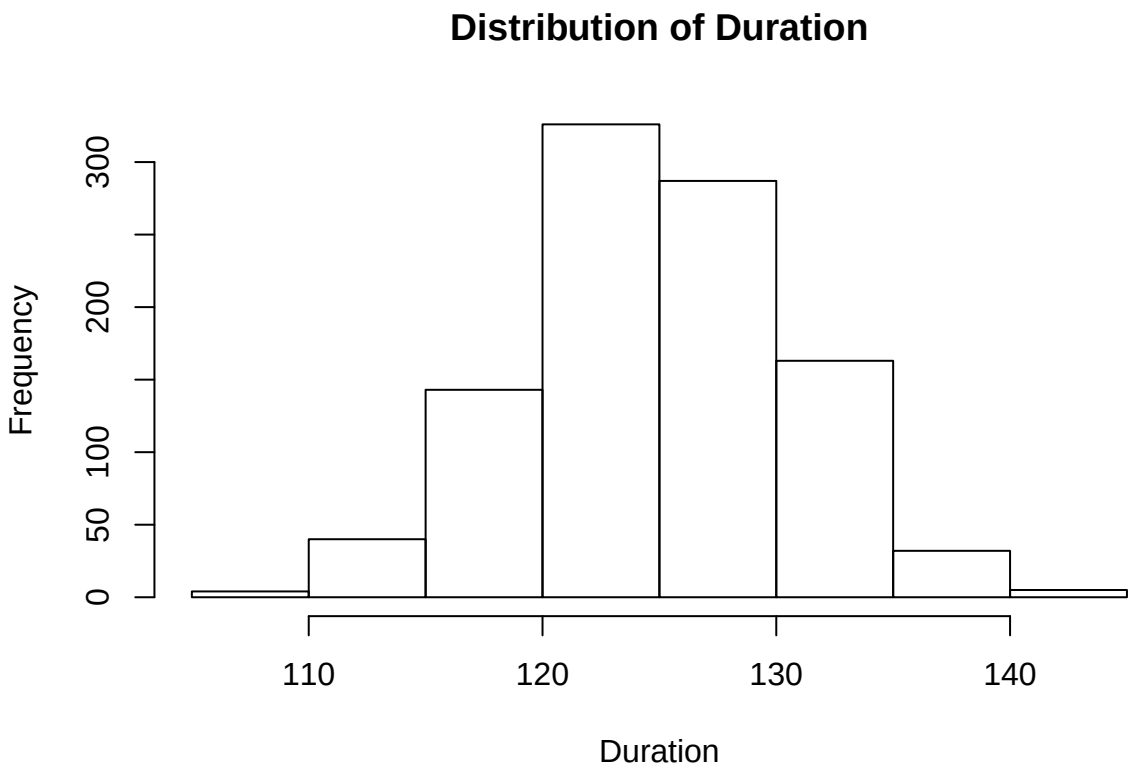
Plot the histagram of Intercept

```
rownames(coefficients)<-names(lm(Amount~.,data=GC2)$coefficients)
hist(coefficients[1,],xlab=rownames(coefficients)[1],ylab='Frequency',
     main='Distribution of Intercept')
```



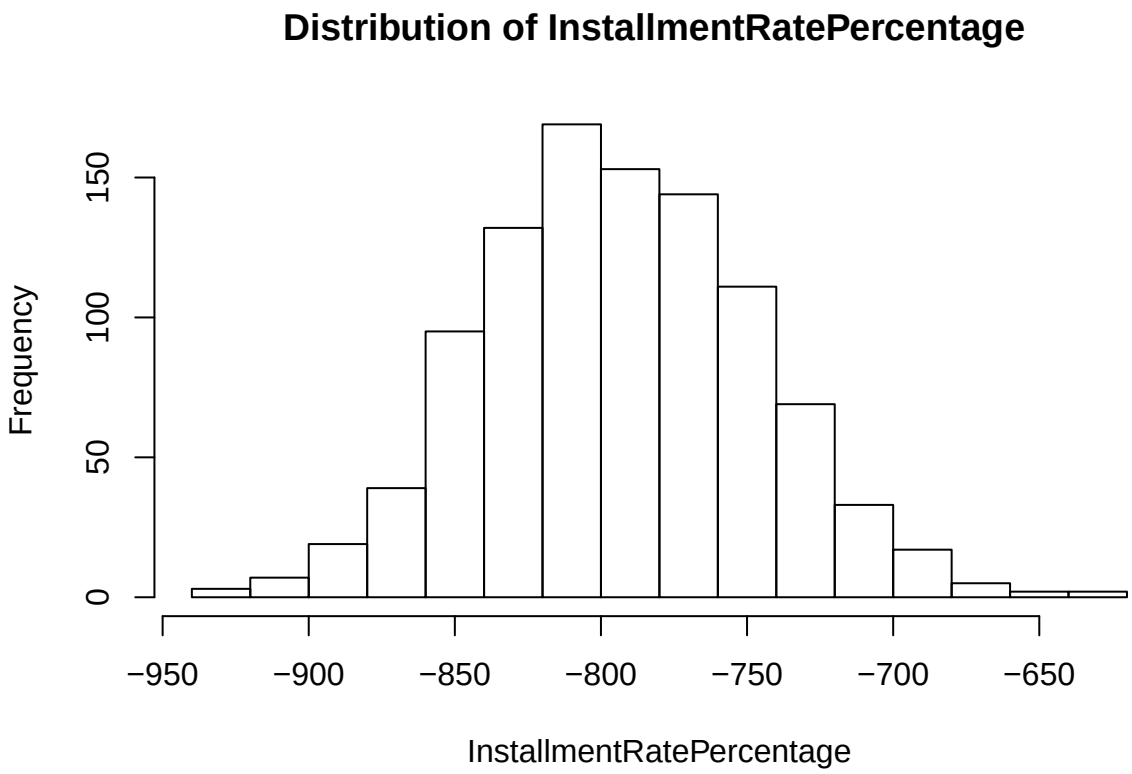
Plot the histagram of Duration

```
hist(coefficients[2,],xlab=rownames(coefficients)[2],ylab='Frequency',
     main='Distribution of Duration')
```



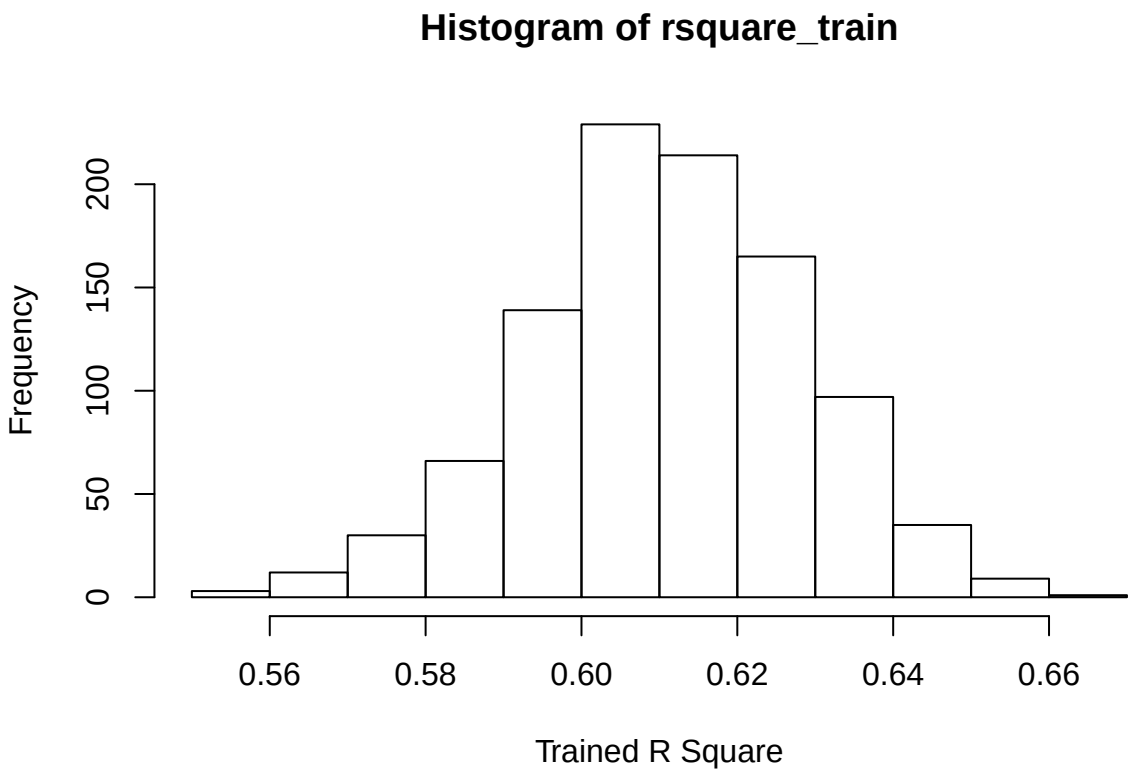
Plot the histagram of InstallmentRatePercentage

```
rownames(coefficients)<-names(lm(Amount~.,data=GC2)$coefficients)
hist(coefficients[3,],xlab=rownames(coefficients)[3],ylab='Frequency',
     main='Distribution of InstallmentRatePercentage ')
```



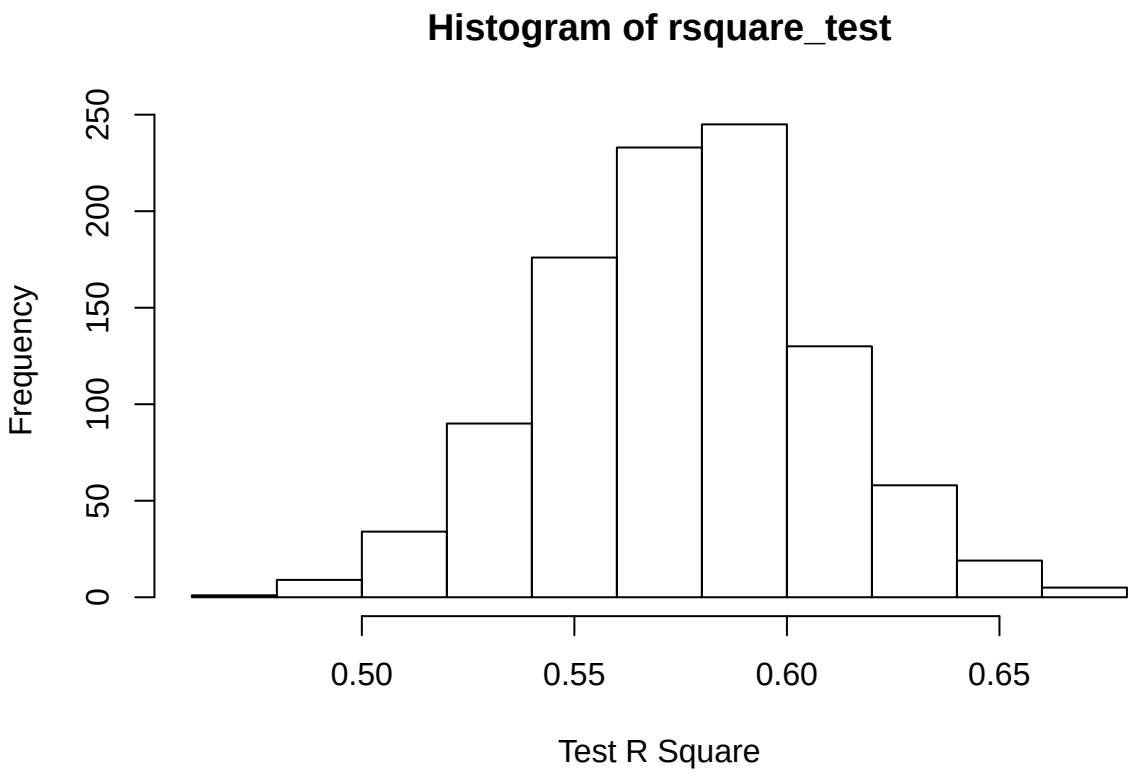
plot the distribution of Train R^2

```
hist(rsquare_train,xlab="Trained R Square",ylab='Frequency')
```



plot the distribution of Test R^2

```
hist(rsquare_test,ylab='Frequency',xlab="Test R Square")
```



plot the distribution of % fall in R^2

```
Pct.fall.r.sqrt<-(rsquare_train-rsquare_test)/rsquare_train
hist(Pct.fall.r.sqrt,
     ylab='Frequency', xlab="% Fall in R Square")
```

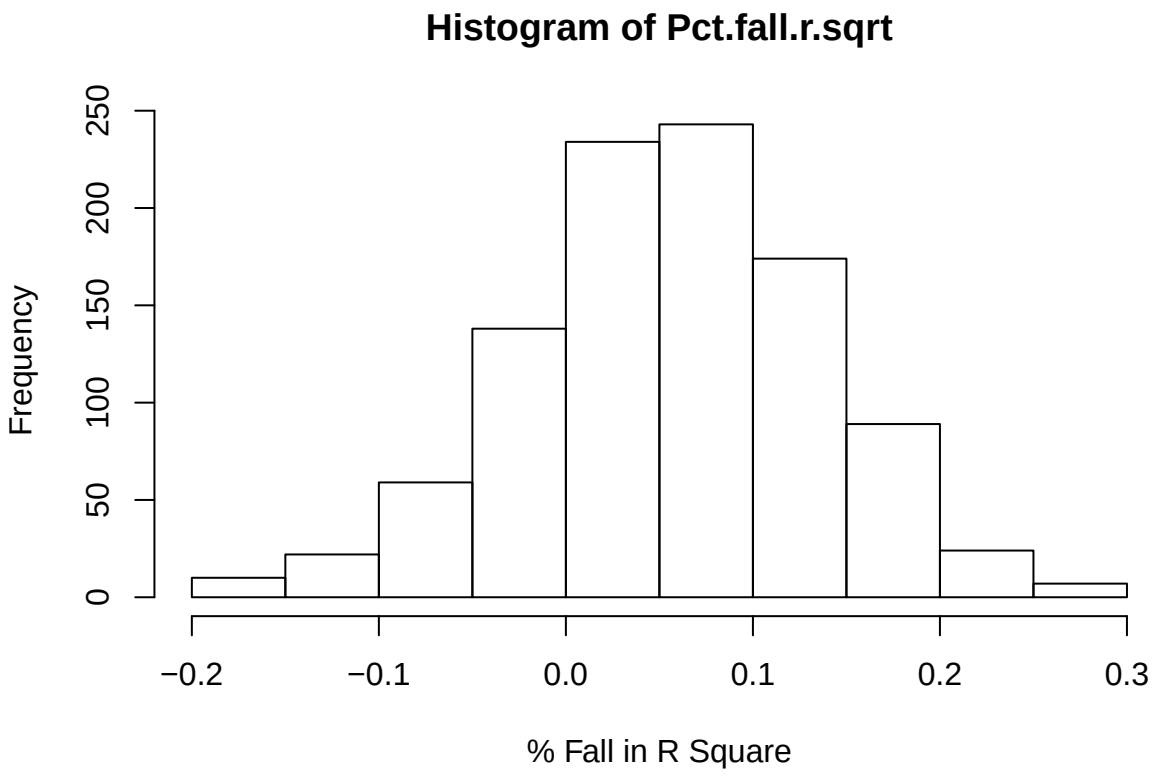


Table of Train, Test and % Fall R^2

```
r.square.table<-cbind(rsquare_train,rsquare_test,Pct.fall.r.sqrt)
colnames(r.square.table)<-c("Train.R.Square","Test.R.Square","%Fall.R.Square")
head(r.square.table)
```

```
##      Train.R.Square Test.R.Square %Fall.R.Square
## [1,]      0.6152178     0.5616678     0.08704238
## [2,]      0.6187729     0.5590161     0.09657306
## [3,]      0.6351336     0.5289136     0.16724038
## [4,]      0.6172584     0.5556888     0.09974686
## [5,]      0.6045357     0.5895642     0.02476527
## [6,]      0.6089067     0.5729312     0.05908217
```

```
# Comparison of Average Train and Test R Square
mean.r.square.table<-cbind(mean(rsquare_train),mean(rsquare_test),
            (mean(rsquare_train)-mean(rsquare_test))/mean(rsquare_train))

colnames(mean.r.square.table)<-c("Avg.Train.R.Square","Avg.Test.R.Square","Avg.%Fall.R.Square")
mean.r.square.table
```

```
##      Avg.Train.R.Square Avg.Test.R.Square Avg.%Fall.R.Square
## [1,]          0.6109545         0.5754504         0.05811258
```

The above graphes show that the test R Square is on average smaller than train R square. The percentage fall of R square range betweem -0.3 to 0.3. Over 80% of the runs generate %Fall of R Sqaure between -0.05 and 0.15. One average, We expect the R square to lose 5.62% from Train to Test.

Build linear model using entire sample.

```
lm.entire <- lm(GC2$Amount~.,data=GC2)
```

Compute the mean and standard deviation of all 1000 coefficients (for each beta)

```
coef.mean <-  apply(coefficients,1,mean)
coef.sd <-  apply(coefficients,1,sd)
names(coef.mean) <- names(lm.entire$coefficients)
names(coef.sd) <- names(lm.entire$coefficients)
cbind(coef.mean=coef.mean,coef.sd=coef.sd)
```

```
##                                          coef.mean    coef.sd
## (Intercept)                             3344.87851  358.74507
## Duration                                 124.99360    5.67158
## InstallmentRatePercentage               -793.54899   46.56111
## Job.Management.SelfEmp.HighlyQualified 1303.16337  189.37109
## Personal.Male.Single                     487.74399   93.95457
## Telephone                               -496.34141  100.61692
## Purpose.UsedCar                          770.83182  200.92048
## Purpose.Other                           1817.87013 1038.77520
## ClassGood                               -371.13224  131.21422
## Property.Unknown                         438.86247  174.44749
## CheckingAccountStatus.gt.200            -680.37834  121.50075
## CreditHistory.NoCredit.AllPaid           814.71453  318.40430
## SavingsAccountBonds.Unknown              394.32153  136.10228
## OtherDebtorsGuarantors.CoApplicant       612.13581  256.09755
## Purpose.Radio.Television                -189.22556   97.98155
## CheckingAccountStatus.lt.0              -251.54930  114.10193
## Property.RealEstate                     -227.17332   81.96834
## EmploymentDuration.gt.7                 -200.30510  108.74483
## ForeignWorker                           -275.15602  237.97138
## Purpose.NewCar                            72.69814  118.66175
```

Compare average across 1000 to single model built using entire sample

```
cbind(Averaged.Coef=coef.mean,Entire.Model.Coef=lm.entire$coefficients,
      "% Difference"=(lm.entire$coefficients-coef.mean)/lm.entire$coefficients)
```

```
##                                        Averaged.Coef Entire.Model.Coef
## (Intercept)                              3344.87851        3360.40732
## Duration                                  124.99360         125.08749
## InstallmentRatePercentage                -793.54899        -795.14442
```

```
## Job.Management.SelfEmp.HighlyQualified    1303.16337        1307.92879
## Personal.Male.Single                        487.74399         481.96905
## Telephone                                  -496.34141        -497.18601
## Purpose.UsedCar                             770.83182         770.77364
## Purpose.Other                              1817.87013        1792.56967
## ClassGood                                  -371.13224        -370.55206
## Property.Unknown                            438.86247         431.47662
## CheckingAccountStatus.gt.200               -680.37834        -679.22520
## CreditHistory.NoCredit.AllPaid              814.71453         808.60346
## SavingsAccountBonds.Unknown                 394.32153         391.55914
## OtherDebtorsGuarantors.CoApplicant          612.13581         618.46243
## Purpose.Radio.Television                   -189.22556        -190.25455
## CheckingAccountStatus.lt.0                 -251.54930        -250.79721
## Property.RealEstate                        -227.17332        -232.40142
## EmploymentDuration.gt.7                    -200.30510        -203.12810
## ForeignWorker                              -275.15602        -283.99638
## Purpose.NewCar                               72.69814          71.03331
##                                          % Difference
## (Intercept)                              4.621109e-03
## Duration                                 7.505688e-04
## InstallmentRatePercentage                2.006463e-03
## Job.Management.SelfEmp.HighlyQualified   3.643487e-03
## Personal.Male.Single                    -1.198197e-02
## Telephone                                1.698765e-03
## Purpose.UsedCar                         -7.547891e-05
## Purpose.Other                           -1.411407e-02
## ClassGood                               -1.565714e-03
## Property.Unknown                        -1.711761e-02
## CheckingAccountStatus.gt.200            -1.697728e-03
## CreditHistory.NoCredit.AllPaid          -7.557566e-03
## SavingsAccountBonds.Unknown             -7.054827e-03
## OtherDebtorsGuarantors.CoApplicant       1.022959e-02
## Purpose.Radio.Television                 5.408477e-03
## CheckingAccountStatus.lt.0              -2.998807e-03
## Property.RealEstate                      2.249600e-02
## EmploymentDuration.gt.7                  1.389765e-02
## ForeignWorker                            3.112841e-02
## Purpose.NewCar                          -2.343736e-02
```

The coefficients from the Model fitted using entire sample is very similar to the Averaged Coefficients of the 1000 training sample runs. The difference on average is less that 1%.

Sort each coefficient's 1000 values.

```
rownames(coefficients)<- names(lm.entire$coefficients)
head.matrix(apply(coefficients, 1, sort))
```

```
##      (Intercept) Duration InstallmentRatePercentage
## [1,]    2127.556 106.8617                 -929.9321
## [2,]    2295.899 107.4811                 -923.3796
## [3,]    2298.879 109.9179                 -921.8891
## [4,]    2312.445 109.9668                 -914.0322
```

```
## [5,]    2326.570 110.2650                -912.8539
## [6,]    2348.756 110.5111                -907.9924
##      Job.Management.SelfEmp.HighlyQualified Personal.Male.Single Telephone
## [1,]                               751.1365             174.5096 -787.7366
## [2,]                               758.6072             192.1113 -742.2757
## [3,]                               759.2652             233.9465 -741.6085
## [4,]                               811.1159             256.0219 -740.6347
## [5,]                               817.2111             257.9914 -736.4498
## [6,]                               853.7072             260.3686 -735.8702
##      Purpose.UsedCar Purpose.Other ClassGood Property.Unknown
## [1,]        193.5125    -1963.4986 -824.6305      -148.230144
## [2,]        229.7797    -1645.9737 -785.5316       -74.168817
## [3,]        234.1691    -1346.5477 -779.1397       -72.933384
## [4,]        237.8505     -991.6591 -737.6472       -40.278737
## [5,]        258.9848     -956.4395 -726.9683       -28.385163
## [6,]        287.7420     -873.4023 -703.4471        -1.808855
##      CheckingAccountStatus.gt.200 CreditHistory.NoCredit.AllPaid
## [1,]                   -1073.7580                     -79.391731
## [2,]                   -1057.3574                     -60.162141
## [3,]                   -1022.9706                       9.894927
## [4,]                    -995.8146                      14.787211
## [5,]                    -992.5778                      31.426746
## [6,]                    -980.9877                      35.867560
##      SavingsAccountBonds.Unknown OtherDebtorsGuarantors.CoApplicant
## [1,]                    6.981326                        -397.137196
## [2,]                   20.016948                        -112.777058
## [3,]                   33.099360                         -76.738049
## [4,]                   48.540347                         -43.403584
## [5,]                   49.368345                         -20.040170
## [6,]                   51.911052                           6.303004
##      Purpose.Radio.Television CheckingAccountStatus.lt.0 Property.RealEstate
## [1,]                -541.6766                  -603.9301           -447.7224
## [2,]                -506.8825                  -601.1782           -435.9066
## [3,]                -474.1053                  -586.5064           -432.9310
## [4,]                -467.5366                  -578.7260           -430.0273
## [5,]                -437.4051                  -557.3095           -429.3911
## [6,]                -421.3411                  -556.0122           -423.4472
##      EmploymentDuration.gt.7 ForeignWorker Purpose.NewCar
## [1,]               -581.3833    -1016.8565      -317.8334
## [2,]               -531.1611     -970.7009      -251.9308
## [3,]               -500.4448     -958.8807      -250.2042
## [4,]               -498.6059     -955.7553      -240.0529
## [5,]               -495.7275     -909.8780      -236.7991
## [6,]               -491.6535     -890.4494      -230.1969
```

Compute 2.5%-97.5% Confidence Intervals (CI). Scale these CI's down by a factor of .632^0.5

```
CI_lower<-rep(NA,20)
CI_upper<-rep(NA,20)
for (i in 1:20){
    CI_lower[i] <- coef.mean[i]+qnorm(0.025)*(coef.sd[i])
    CI_upper[i] <- coef.mean[i]+qnorm(0.975)*(coef.sd[i])
```

```
}
scaled.CI <- cbind("Scaled 2.5%"=CI_lower," Scaled 97.5"=CI_upper,
                   "Scaled With"=(CI_upper-CI_lower)*sqrt(0.632))
rownames(scaled.CI)<-names(lm.entire$coefficients)
```

compute single model's CIs

```
single.model.CI <- confint(lm.entire,names(lm.entire$coefficients),level=0.95)

single.model.CI.width<-cbind(single.model.CI,'width'=(single.model.CI[,2]-single.model.CI[,1]))
```

How do these Scaled CIs compare to CIs computed from single model's CIs? Tighter or broader?

```
# CI compariaion table
CI.compare<-cbind(scaled.CI=scaled.CI,single.model.CI=single.model.CI.width)
CI.compare
```

```
##                                       Scaled 2.5%  Scaled 97.5 Scaled With
## (Intercept)                            2641.75109  4048.005934  1117.95049
## Duration                                113.87751   136.109692    17.67424
## InstallmentRatePercentage              -884.80709  -702.290894   145.09751
## Job.Management.SelfEmp.HighlyQualified  932.00285  1674.323883   590.13355
## Personal.Male.Single                    303.59643   671.891562   292.78884
## Telephone                              -693.54696  -299.135870   313.55061
## Purpose.UsedCar                         377.03492  1164.628722   626.12469
## Purpose.Other                          -218.09185  3853.832100  3237.11551
## ClassGood                              -628.30739  -113.957088   408.90040
## Property.Unknown                         96.95167   780.773264   543.62742
## CheckingAccountStatus.gt.200           -918.51543  -442.241245   378.63049
## CreditHistory.NoCredit.AllPaid          190.65357  1438.775492   992.23731
## SavingsAccountBonds.Unknown             127.56596   661.077093   424.13296
## OtherDebtorsGuarantors.CoApplicant      110.19384  1114.077780   798.07194
## Purpose.Radio.Television               -381.26587     2.814749   305.33805
## CheckingAccountStatus.lt.0             -475.18497   -27.913631   355.57368
## Property.RealEstate                    -387.82832   -66.518320   255.43640
## EmploymentDuration.gt.7                -413.44104    12.830846   338.87944
## ForeignWorker                          -741.57136   191.259312   741.58571
## Purpose.NewCar                         -159.87462   305.270910   369.78338
##                                            2.5 %     97.5 %      width
## (Intercept)                           2554.03660 4166.77804 1612.74144
## Duration                               114.85291  135.32207   20.46916
## InstallmentRatePercentage             -899.03517 -691.25367  207.78150
## Job.Management.SelfEmp.HighlyQualified  953.87258 1661.98500  708.11241
## Personal.Male.Single                    246.44206  717.49604  471.05398
## Telephone                              -749.43566 -244.93637  504.49930
## Purpose.UsedCar                         358.87341 1182.67388  823.80047
## Purpose.Other                           719.61639 2865.52295 2145.90657
## ClassGood                              -641.31699  -99.78713  541.52986
```

```
## Property.Unknown                          94.67003  768.28321  673.61318
## CheckingAccountStatus.gt.200           -1146.75670 -211.69369  935.06301
## CreditHistory.NoCredit.AllPaid           224.98669 1392.22023 1167.23354
## SavingsAccountBonds.Unknown               94.55293  688.56536  594.01243
## OtherDebtorsGuarantors.CoApplicant        43.52477 1193.40009 1149.87532
## Purpose.Radio.Television                -477.00735   96.49825  573.50560
## CheckingAccountStatus.lt.0              -516.64711   15.05270  531.69981
## Property.RealEstate                     -500.07229   35.26945  535.34174
## EmploymentDuration.gt.7                 -471.68294   65.42674  537.10968
## ForeignWorker                           -898.69465  330.70189 1229.39654
## Purpose.NewCar                          -229.45227  371.51889  600.97117
```

Number of CI tighter or broader.

```
paste("Numer of CIs tightened: ",sum(CI.compare[,3] < CI.compare[,6]))
```

```
## [1] "Numer of CIs tightened:  19"
```

```
paste("Numer of CIs broadened: ",sum(CI.compare[,3] > CI.compare[,6]))
```

```
## [1] "Numer of CIs broadened:  1"
```

Standard deviation of Coefficients where CI of sampled runs tightened

```
coef.sd[CI.compare[,3] < CI.compare[,6]]
```

```
##                             (Intercept)                                Duration
##                               358.74507                                 5.67158
##             InstallmentRatePercentage Job.Management.SelfEmp.HighlyQualified
##                                46.56111                                189.37109
##                  Personal.Male.Single                                Telephone
##                                93.95457                                100.61692
##                       Purpose.UsedCar                                ClassGood
##                                200.92048                                131.21422
##                       Property.Unknown          CheckingAccountStatus.gt.200
##                                174.44749                                121.50075
##         CreditHistory.NoCredit.AllPaid           SavingsAccountBonds.Unknown
##                                318.40430                                136.10228
##     OtherDebtorsGuarantors.CoApplicant              Purpose.Radio.Television
##                                256.09755                                 97.98155
##             CheckingAccountStatus.lt.0                   Property.RealEstate
##                                114.10193                                 81.96834
##                EmploymentDuration.gt.7                          ForeignWorker
##                                108.74483                                237.97138
##                         Purpose.NewCar
##                                118.66175
```

Standard deviation of Coefficient where CI of sampled runs broadened

```
coef.sd[CI.compare[,3] > CI.compare[,6]]
```

```
## Purpose.Other
##      1038.775
```

According to the tables above, we find 19 variables generated smaller CIs and 1 variable generated larger CI when using repeated sample models. The Coefficents of varaibles with tighter CIs tend to vary much less (smaller variance/sd) than the coefficient with broader CI when generated by random sampling model. Hence we could conclude that repeating the model fitting process with random smaples multiple times can help improving the precision of our model coefficients. Increasing number of repeatitions will improve CIs; howerer once a critical value of repetition is acheive, the improvement of CIs becomes unnoticeable. In this assignment, the result of random sample modeling 10,000 times is very similar to the result of modeling 1000 times.