# UtsavSinghi_PGPBDA_Hyd_Oct17_SL_Asmt.R

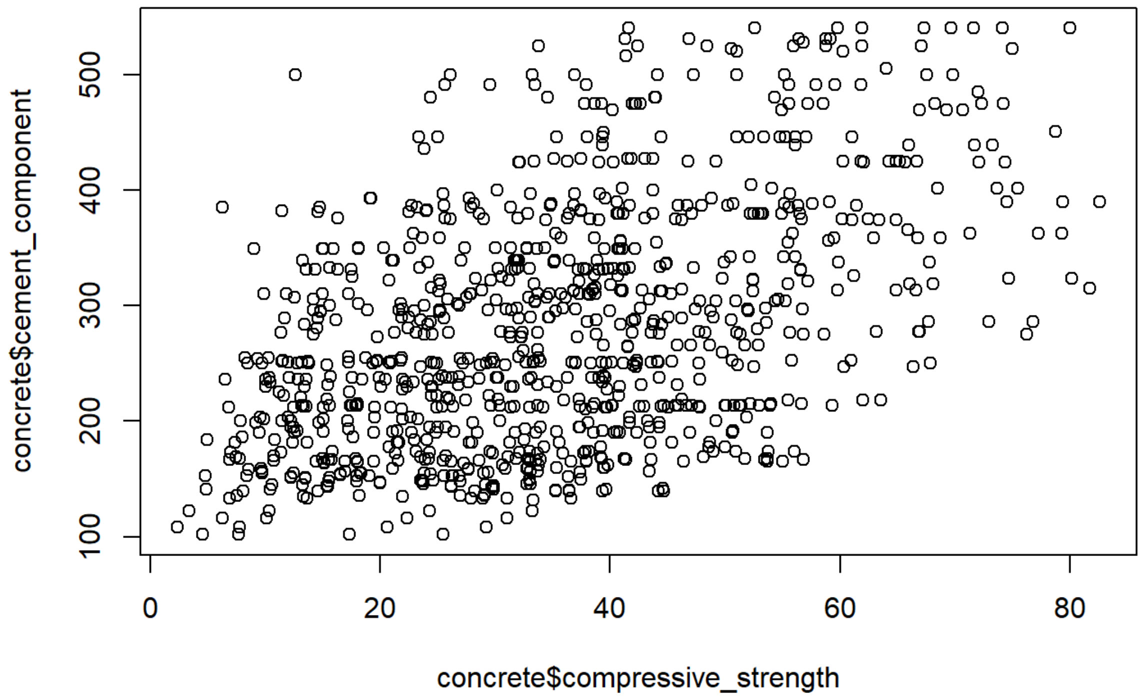
*LENEVO*

*Sat Dec 16 22:25:29 2017*

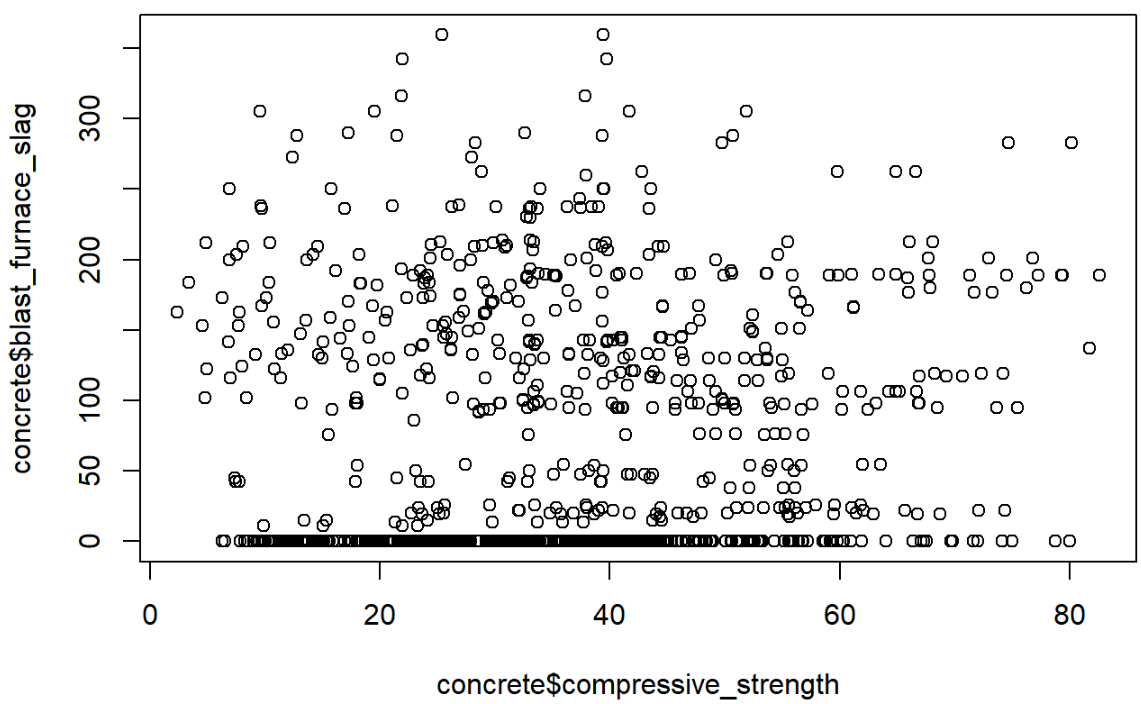
```
#Import and understand the data. Look at the range of the various attributes.
setwd('E:/PGPBDA/R Programming/Stat Assignment 2')
#Read The Data
concrete<- read.csv('Concrete_Data.csv',header = TRUE)
str(concrete)
```

```
## 'data.frame':    1030 obs. of  9 variables:
##  $ Cement..component.1..kg.in.a.m.3.mixture.            : num  540 540 332 332 199 ...
##  $ Blast.Furnace.Slag..component.2..kg.in.a.m.3.mixture.: num  0 0 142 142 132 ...
##  $ Fly.Ash..component.3..kg.in.a.m.3.mixture.           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Water...component.4..kg.in.a.m.3.mixture.            : num  162 162 228 228 192 228 228 228 228 228
...
##  $ Superplasticizer..component.5..kg.in.a.m.3.mixture.  : num  2.5 2.5 0 0 0 0 0 0 0 0 ...
##  $ Coarse.Aggregate...component.6..kg.in.a.m.3.mixture. : num  1040 1055 932 932 978 ...
##  $ Fine.Aggregate..component.7..kg.in.a.m.3.mixture.    : num  676 676 594 594 826 ...
##  $ Age..day.                                            : num  28 28 270 365 360 90 365 28 28 28 ...
##  $ Concrete.compressive.strength.MPa..megapascals..     : num  80 61.9 40.3 41 44.3 ...
```

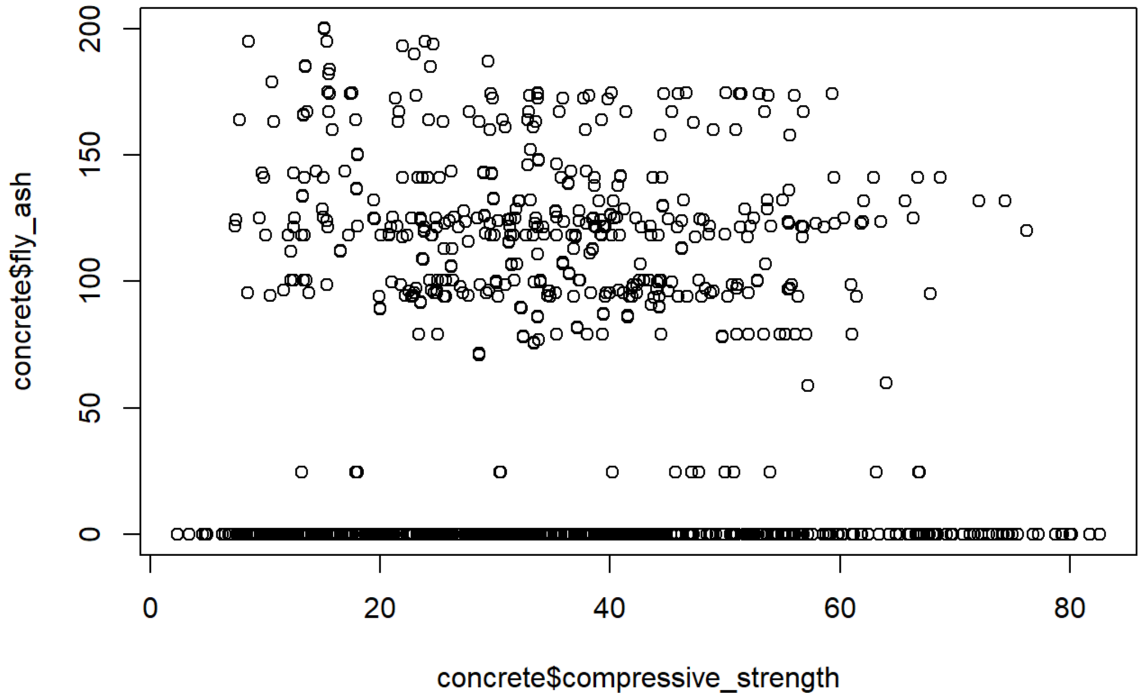
```
# Making the name of heading shorter
names(concrete) <- c('cement_component','blast_furnace_slag','fly_ash','water','superplasticizer',
                    'coarse_aggregate','fine_aggregate','age','compressive_strength')

# Print out the summary statistics of all variables and arrange them in a neat tabular format.
View(summary(concrete))

#Create Scatter Plot of Compressive Strength versus the other variables, taking each predictive variable at
 a time. Clearly label the graphs.

plot(concrete$compressive_strength , concrete$cement_component)
```



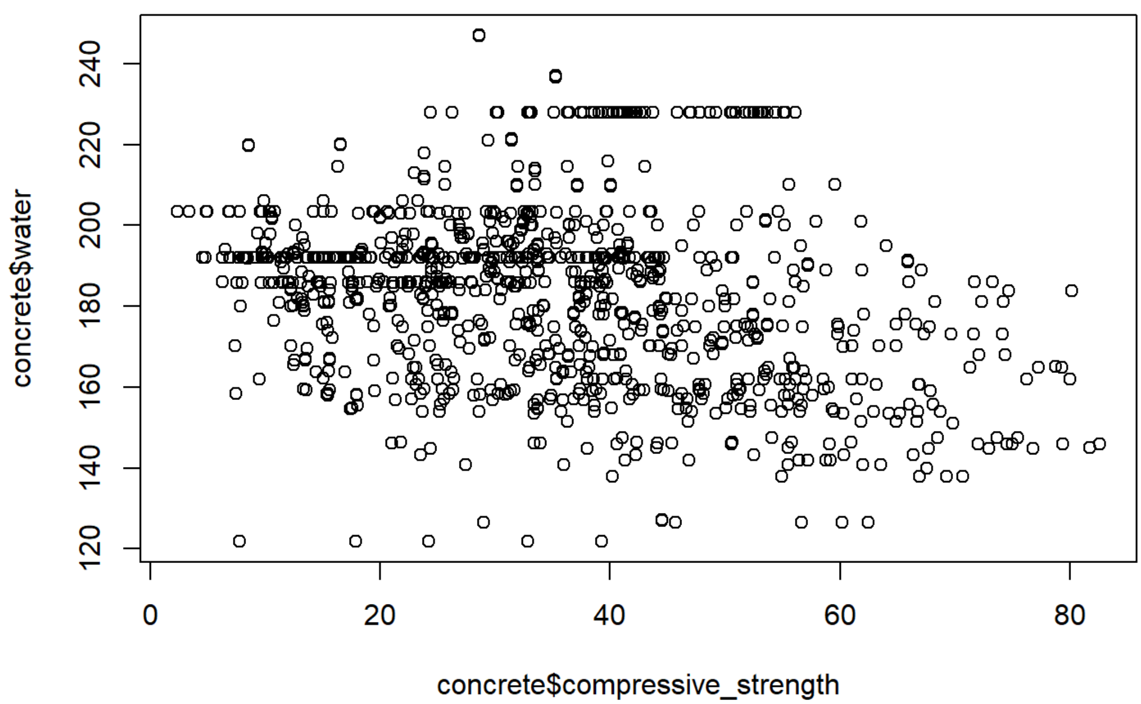

```
plot(concrete$compressive_strength , concrete$blast_furnace_slag)
```



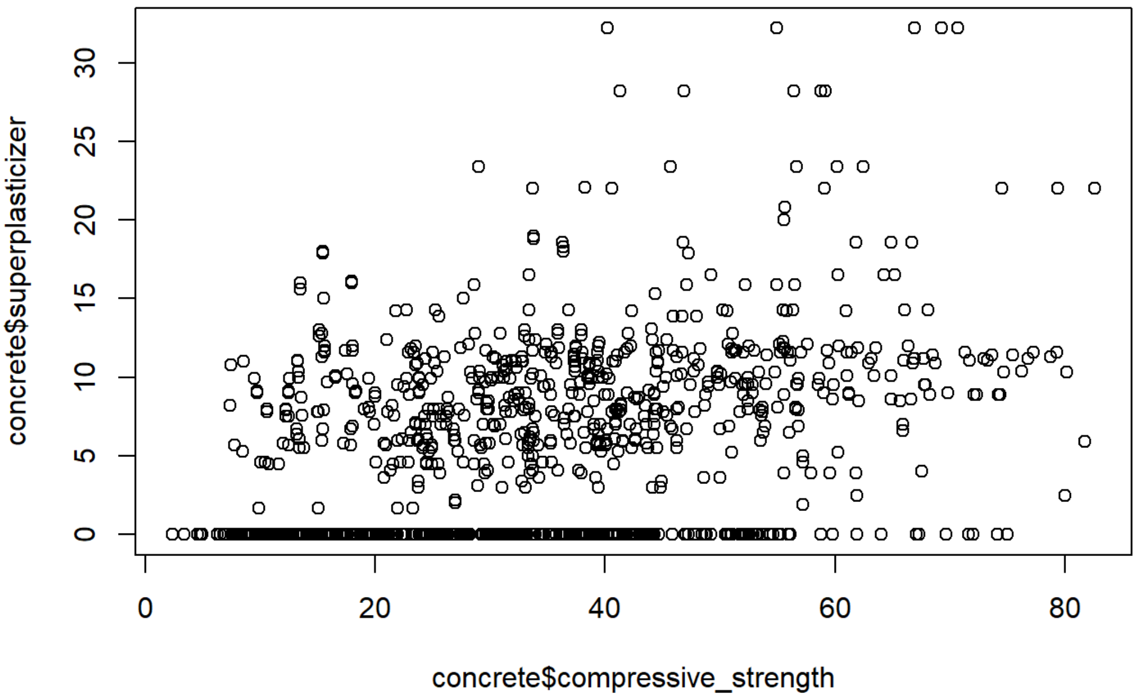

```
plot(concrete$compressive_strength , concrete$fly_ash)
```



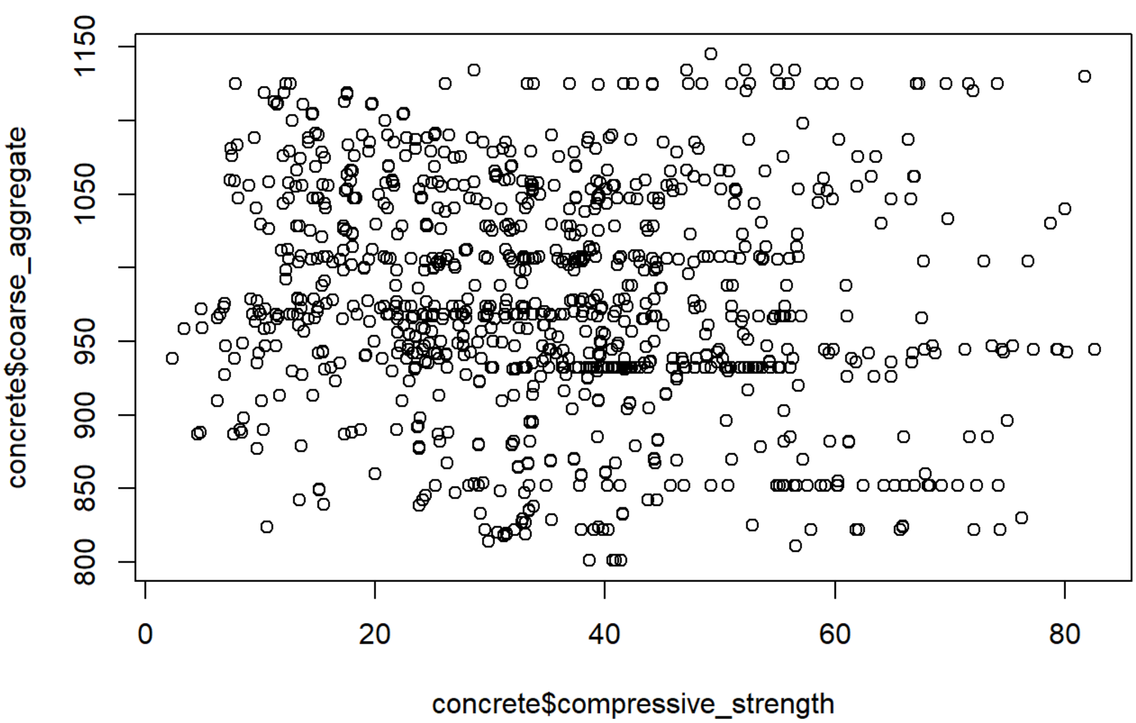

```
plot(concrete$compressive_strength , concrete$water)
```



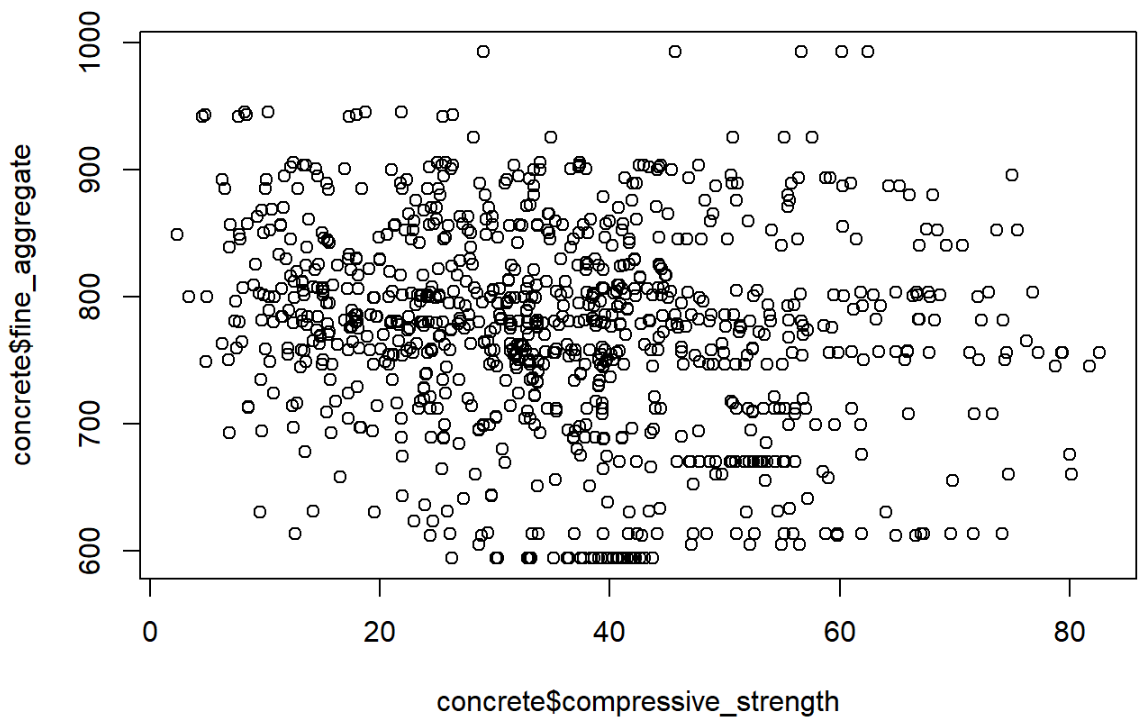

```
plot(concrete$compressive_strength , concrete$superplasticizer)
```




```
plot(concrete$compressive_strength , concrete$coarse_aggregate)
```




```
plot(concrete$compressive_strength , concrete$fine_aggregate)
```




```
plot(concrete$compressive_strength , concrete$age)

# Produce pairwise correlation coefficient table. Comment on the values of the correlations between each pr
edictor and response. Do you think there is any pairwise correlation between predictors which may cause wor
ry?
cor(concrete$compressive_strength , concrete$cement_component)
```

```
## [1] 0.4978319
```

```
cor(concrete$compressive_strength , concrete$blast_furnace_slag)
```

```
## [1] 0.1348293
```

```
cor(concrete$compressive_strength , concrete$fly_ash)
```

```
## [1] -0.1057549
```

```
cor(concrete$compressive_strength , concrete$water)
```

```
## [1] -0.2896334
```

```
cor(concrete$compressive_strength , concrete$superplasticizer)
```

```
## [1] 0.3660788
```

```
cor(concrete$compressive_strength , concrete$coarse_aggregate)
```

```
## [1] -0.1649346
```

```
cor(concrete$compressive_strength , concrete$fine_aggregate)
```

```
## [1] -0.1672412
```

```
cor(concrete$compressive_strength , concrete$age)
```

```
## [1] 0.328873
```

```
# As per our analysis,  pairwise correlation between predictors is not a  cause worry
cor(concrete)
```

```
##                      cement_component blast_furnace_slag      fly_ash
## cement_component           1.00000000        -0.27521591 -0.397467341
## blast_furnace_slag        -0.27521591         1.00000000 -0.323579901
## fly_ash                   -0.39746734        -0.32357990  1.000000000
## water                     -0.08158675         0.10725203 -0.256984023
## superplasticizer           0.09238617         0.04327042  0.377503146
## coarse_aggregate          -0.10934899        -0.28399861 -0.009960828
## fine_aggregate            -0.22271785        -0.28160267  0.079108491
## age                        0.08194602        -0.04424602 -0.154370516
## compressive_strength       0.49783192         0.13482926 -0.105754916
##                            water superplasticizer coarse_aggregate
## cement_component     -0.08158675       0.09238617     -0.109348994
## blast_furnace_slag    0.10725203       0.04327042     -0.283998612
## fly_ash              -0.25698402       0.37750315     -0.009960828
## water                 1.00000000      -0.65753291     -0.182293602
## superplasticizer     -0.65753291       1.00000000     -0.265999148
## coarse_aggregate     -0.18229360      -0.26599915      1.000000000
## fine_aggregate       -0.45066117       0.22269123     -0.178480957
## age                   0.27761822      -0.19270003     -0.003015880
## compressive_strength -0.28963338       0.36607883     -0.164934614
##                      fine_aggregate         age compressive_strength
## cement_component        -0.22271785  0.08194602            0.4978319
## blast_furnace_slag      -0.28160267 -0.04424602            0.1348293
## fly_ash                  0.07910849 -0.15437052           -0.1057549
## water                   -0.45066117  0.27761822           -0.2896334
## superplasticizer         0.22269123 -0.19270003            0.3660788
## coarse_aggregate        -0.17848096 -0.00301588           -0.1649346
## fine_aggregate           1.00000000 -0.15609470           -0.1672412
## age                     -0.15609470  1.00000000            0.3288730
## compressive_strength    -0.16724125  0.32887300            1.0000000
```

```
#Build Multiple Linear Regression model of Compressive Strength on ALL the predictors. Report multiple R2 o
f the model.
lm.model <- lm(compressive_strength~.,data=concrete)
summary(lm.model)
```

```
##
## Call:
## lm(formula = compressive_strength ~ ., data = concrete)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.654  -6.302   0.703   6.569  34.450
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -23.331214  26.585504  -0.878 0.380372
## cement_component      0.119804   0.008489  14.113  < 2e-16 ***
## blast_furnace_slag    0.103866   0.010136  10.247  < 2e-16 ***
## fly_ash               0.087934   0.012583   6.988 5.02e-12 ***
## water                -0.149918   0.040177  -3.731 0.000201 ***
## superplasticizer      0.292225   0.093424   3.128 0.001810 **
## coarse_aggregate      0.018086   0.009392   1.926 0.054425 .
## fine_aggregate        0.020190   0.010702   1.887 0.059491 .
## age                   0.114222   0.005427  21.046  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.4 on 1021 degrees of freedom
## Multiple R-squared:  0.6155, Adjusted R-squared:  0.6125
## F-statistic: 204.3 on 8 and 1021 DF,  p-value: < 2.2e-16
```

```
#Sum of square
lm.modelsum<- sum((lm.model$residuals)^2)
lm.modelsum
```

```
## [1] 110413.2
```

```
#As we can see the p-values of the t-statistic of all the coefficients are close to zero
#except for that of coarse_aggregate and fine_aggregate.
# If we look at summary data, after p-value there is dot for coarse_aggregate and fine_aggregate
#Usually it means probablity between 0.5 and 0.10 and we do not include these inmode;
#This indicates that other than the coefficients of coarse_aggregate and fine_aggregate,
#all the other coefficients of the model are significant for the accuracy of the fit.



  #Removing the coarse_aggregate and fine_aggregate attributes from the model
  lm.model2 <- lm(compressive_strength~.-coarse_aggregate-fine_aggregate,data=concrete)
summary(lm.model2)
```

```
##
## Call:
## lm(formula = compressive_strength ~ . - coarse_aggregate - fine_aggregate,
##     data = concrete)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.987  -6.469   0.653   6.547  34.732
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        28.992982   4.213202   6.881 1.03e-11 ***
## cement_component    0.105413   0.004246  24.825  < 2e-16 ***
## blast_furnace_slag  0.086472   0.004974  17.385  < 2e-16 ***
## fly_ash             0.068660   0.007735   8.877  < 2e-16 ***
## water              -0.218088   0.021129 -10.322  < 2e-16 ***
## superplasticizer    0.240311   0.084567   2.842  0.00458 **
## age                 0.113492   0.005407  20.988  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.41 on 1023 degrees of freedom
## Multiple R-squared:  0.614,  Adjusted R-squared:  0.6118
## F-statistic: 271.2 on 6 and 1023 DF,  p-value: < 2.2e-16
```

```
#Sum of square
lm.modelsum2<- sum((lm.model2$residuals)^2)
lm.modelsum
```

```
## [1] 110413.2
```

```
#Removal of "coarse_aggregate" and "fine_aggregate" attributes
#doesn't improve the R-squared value of the model and the mean squared error of the model  data doesn't dec
rease.
#Therefore we will go ahead with the first model, lm.model.

#Residual Standard Error of lm.model
sqrt(deviance(lm.model)/lm.model$df.residual)
```

```
## [1] 10.39914
```

```
#Mean response of the dataset
mean(concrete$compressive_strength)
```

```
## [1] 35.81796
```

```
#The residual standard error of lm.model is around 10.340and the mean response is about 36.00.
#The error rate is as following.


  #Error rate
 10.40/36*100
```

```
## [1] 28.88889
```

```
  28.88
```

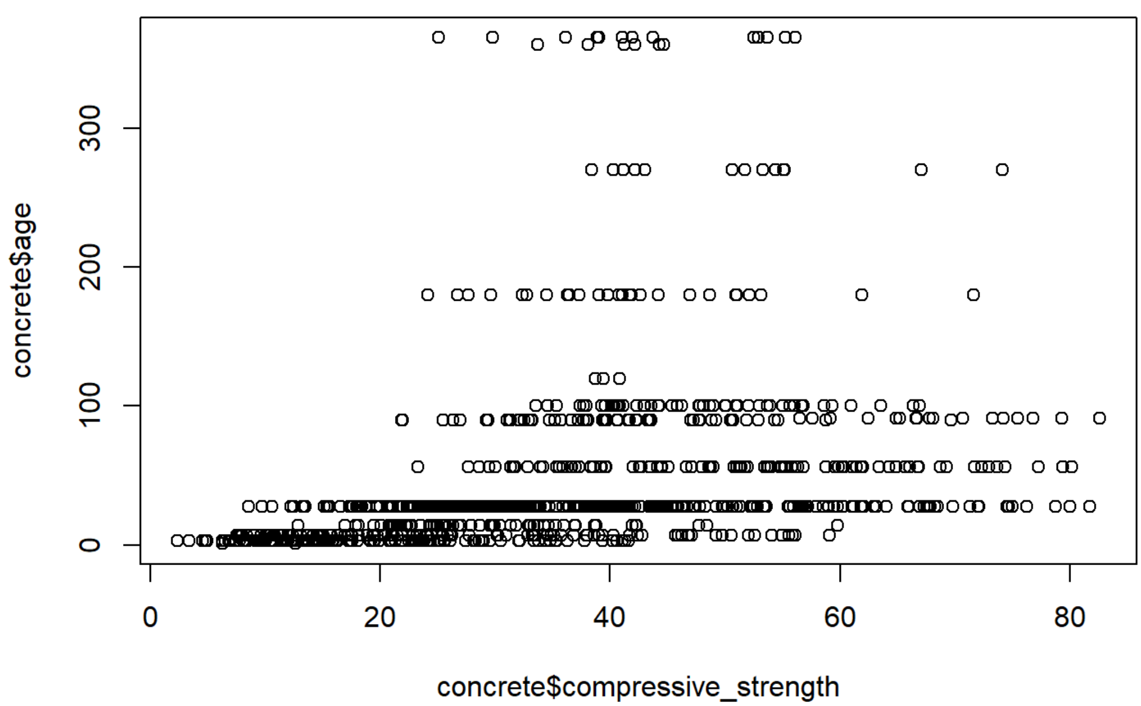
```
## [1] 28.88
```

```
# R Squared value
  summary(lm.model)$r.squared
```

```
## [1] 0.6155199
```

```
# The R-squared value of lm.model is 0.61. This indicates that 61% of the variability in the response has b
een explained by the model.


# In our model multicollinearity is not there, as there is no independent variable which are highly co rela
ted
# There s no definate cut off but we are considering any correlation greater than or less than 0.7 is cause
 for concern.
# So we will sticking to model 1

#Check for Multi-Collinearity and report all VIF values.
library(car)
```

```
## Warning: package 'car' was built under R version 3.4.3
```



```
lm.modelmulti1 <- lm(compressive_strength~.,data=concrete)
summary(lm.modelmulti1)
```

```
##
## Call:
## lm(formula = compressive_strength ~ ., data = concrete)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.654  -6.302   0.703   6.569  34.450
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -23.331214  26.585504  -0.878 0.380372
## cement_component       0.119804   0.008489  14.113  < 2e-16 ***
## blast_furnace_slag     0.103866   0.010136  10.247  < 2e-16 ***
## fly_ash                0.087934   0.012583   6.988 5.02e-12 ***
## water                 -0.149918   0.040177  -3.731 0.000201 ***
## superplasticizer       0.292225   0.093424   3.128 0.001810 **
## coarse_aggregate       0.018086   0.009392   1.926 0.054425 .
## fine_aggregate         0.020190   0.010702   1.887 0.059491 .
## age                    0.114222   0.005427  21.046  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.4 on 1021 degrees of freedom
## Multiple R-squared:  0.6155, Adjusted R-squared:  0.6125
## F-statistic: 204.3 on 8 and 1021 DF,  p-value: < 2.2e-16
```

```
View(vif(lm.modelmulti1))

# If we see Multi-Collinearity, is very large in cement_component, so we will be building complete
#new multi linear regression model and removing cement_component

lm.modelmulti2 <- lm(compressive_strength~.-cement_component,data=concrete)
View(vif(lm.modelmulti2))
summary(lm.modelmulti2)
```

```
##
## Call:
## lm(formula = compressive_strength ~ . - cement_component, data = concrete)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -41.329  -7.679  -0.080   7.832  35.877
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        301.532014  14.533501  20.747  < 2e-16 ***
## blast_furnace_slag  -0.023371   0.005061  -4.618 4.37e-06 ***
## fly_ash             -0.068234   0.006546 -10.424  < 2e-16 ***
## water               -0.549056   0.031181 -17.609  < 2e-16 ***
## superplasticizer     0.279643   0.102076   2.740  0.00626 **
## coarse_aggregate    -0.085002   0.006451 -13.176  < 2e-16 ***
## fine_aggregate      -0.109407   0.006005 -18.220  < 2e-16 ***
## age                  0.109522   0.005919  18.504  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.36 on 1022 degrees of freedom
## Multiple R-squared:  0.5405, Adjusted R-squared:  0.5374
## F-statistic: 171.7 on 7 and 1022 DF,  p-value: < 2.2e-16
```

```
# So if we see, after removing water our model is more significant then previous models also new r square

#value is 54.05%

#final Model is below given model and if we see we have all  significant at alpha value of 0.05
View(lm.modelmulti2$coefficients)
```