Concrete Compressive Strength Evaluation

Alberto Beltran, Nathan Jones, Tanar Hernandez-Wroblewski, & Brandon Shin

Abstract

In this project we will be analyzing a dataset with the aim to produce a linear regression model that will aid our inquiry regarding the realtionship between fixed and relevant variables. Our dataset pertains to the Compressive Strength in Cement. We are interested in answering the following questions using our model:

\* What are relevant contributing components towards the Compressive Strength of Concrete?

\* Do strategic investments in statistically significant components make a noteworthy difference in the Compressive Strength?

\* Does elimination of extreme values affect our previous conclusion?

This regression analysis highlighted the expendability of the Coarse aggregate and fine aggregate components within the model of overall compressive strength. The analysis also led to the conclusion that the predictors cement, superplasticzer, and age were the most promenent components in our prediction of overall strength. Using this, we ultimately determined that when an increase in monetary investment towards these significant components takes place, substantial improvement in the compressive strength of the concrete is achieved.

Problem and Motivation

Whatever the product may be, as individuals, we seek to make the most cost-efficient choice. In terms of US construction, there are very few projects that can take place without the presence/implementation of concrete. In recent years (2011 to 2012) we saw an estimated growth of investment of $.9 billion dollars($6.6 to $7.5) which is only expected to increase. The average price of cement per metric ton(907.185kg), in the year 2008(data source year), as reported by statista.com was $103.5 US Dollars. Concrete is (becoming) incredibly popular around the world because it is incredibly durable. The question then becomes, can this durablility be measured? Knowing this, can we optimize our costs in order to make certain that we get the most out of our product how can we best allocate our funds in order to produce the most resilient and durable concrete? And finally, can this prediction be altered, significantly, in any manner? In other words, can we eliminate some of our data points in order to achieve a more robust model that allows us to attack our questions of interest more effectively as well as make the smartest investment when producing concrete?

Questions of Interest

When analyzing the Concrete Compressive Strength dataset, we are interested in the following:

\* What components constitute a suitable linear model test for Compressive Strength?

\* Predicting the Compressive Strength of a hypothetical scenario and compare this result with the average in order to ascertain a strategic investment.

\* Does the elimination of a handful of extreme data entries(outliers) have a significant effect on our prediction interval for the hypthetical scenario?

Data

The Compressive Strength dataset includes data gathered from Cheng Yehs’, “Modeling of strength of high performance concrete using artificial neural networks,” Cement and Concrete Research. We implement the components in our regression as follows:

X1: Cement (kg in a m3 mixture)

X2: Blast Furnace Slag (kg in a m3 mixture)

X3: Fly Ash (kg in a m3 mixture)

X4: Water (kg in a m3 mixture)

X5: Superplasticizer (kg in a m3 mixture)

X6: Coarse Aggregate (kg in a m3 mixture)

X7: Fine Aggregate (kg in a m3 mixture)

X8: Age (Day (1~365))

Y: Concrete compressive strength (MPa)

Regression Methods

In order to make sure that we have the best fit model to test Compressive Strength we start with a model build that has all varibales available in the dataset included.

\* We interpret the quantified p-values and use this reasoning to construct a reduced model. \* We run a partial F test to test the validity of certain coefficents being equal to 0.

\* Ascertain that the change in explainable variablilty is insignifcant and conclude that our reduced model is the best for linear regression.

\* Develop intuition for Model Selection using Added Variable Plots

\* Check our reduced model with the result from running a forward AIC test (This is done because the R^2 value is a measure of explainable variablility with respect to the addition of variables while the AIC test indicates the quality of the model fit in terms of data generating function. In other words, it makes sure that the variables included in the final model fit are significant).

\* We conduct an AIC to confirm our findings that the most powerful model excludes X6 and X7. The reason why we check again using AIC because AIC tests the power of a model to predict a response given new predictors while adjusted R^2 shows how well our current model explains our current data. Since both show that the best model leaves out X6 and X7, we can move forward with confidence that we have the most powerful model.

\* We then further investigate what kind of transformations an absolutely optimal model would consist of. Through the use of the powerTransform function, the multivariate Box-Cox method, we found optimal lambda values for power transformations on each predictor. However, our untransformed model validates our diagnostic assumptions and remained robust. Therefore, in order to maintain a higher level of interpretablility in our model, we opted to move forward with the model excluding the transformations. (see A3 for code)

\* From here we turn our attention to the Residual vs Fitted, Q-Q, and Scale-Location plots to verify that our data upholds diagnostic assumptions: Linearity, Normality, Independence, and Constant Variance

\* Using Cook’s Distance and Leverage values to guide out investigation as to whether outlier removal is needed. Ultimately concluding that no outliers exist in our model.

\* We then use the model to predict the compressive strength of cement in a hypothetical situation and compare to the value computed for average values across all components (since we are interested in the effect of particular components, we will hold the non-uniquely defined varibles to be the mean of that respective component).

Regression Analysis, Reuslts, and Interpretation

# We fit a model with all of the available components and assess what we are working with  
model1 <- lm(Y~X1+X2+X3+X4+X5+X6+X7+X8)  
summary(model1)

##   
## Call:  
## lm(formula = Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -28.653 -6.303 0.704 6.562 34.446   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -23.163756 26.588421 -0.871 0.383851   
## X1 0.119785 0.008489 14.110 < 2e-16 \*\*\*  
## X2 0.103847 0.010136 10.245 < 2e-16 \*\*\*  
## X3 0.087943 0.012585 6.988 5.03e-12 \*\*\*  
## X4 -0.150298 0.040179 -3.741 0.000194 \*\*\*  
## X5 0.290687 0.093460 3.110 0.001921 \*\*   
## X6 0.018030 0.009394 1.919 0.055227 .   
## X7 0.020154 0.010703 1.883 0.059968 .   
## X8 0.114226 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

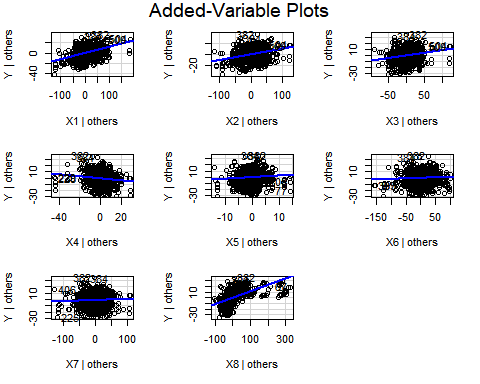
# The respective p-values of X6 and X7 seem to be inadequate at a signifance level of .05. Implying that these respective data are not statistically significant against the null hypthesis:  
# b1=b2=b3=b4=b5=b6=b7=b8=0  
model2 <- lm(Y~X1+X2+X3+X4+X5+X8)  
anova(model2, model1)

## Analysis of Variance Table  
##   
## Model 1: Y ~ X1 + X2 + X3 + X4 + X5 + X8  
## Model 2: Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 1023 110856   
## 2 1021 110428 2 427.81 1.9777 0.1389

From the anova table we can deduce that for the hypothesis test H0: beta6 = beta7 = 0 vs HA: not H0 ; the null hypothesis is accepted and deduce that X6, X7 are not significant enough to be included in our model. We note an insignificant loss of explainable variablilty from model1 to model2. Hence, those variables do not add to the explanatory power of the model and would be best left out.

To develop some intuition on the model selection process we examine the added variable plots for the full model

avPlots(model



The slope of each respective added variable plot shows the effect of each predictor against the repsonse (compressive strength) while controlling for the effects of the other predictors. From these plots we can guess that X6 and X7 will be removed by AIC model selection due to their practically horizontal slopes.

To show beyond a doubt that we have the best fit model we run a forward AIC test. NOTE: We decided to use this model as opposed to our Box-Cox transformed model for reasons of interpretatability and pertenance to our questions of interest. To see the best fit model with the lowest AIC and explaination refer to Appendix 3.

model1 <- lm(Y~X1+X2+X3+X4+X5+X6+X7+X8)  
model2 <- lm(Y~X1+X2+X3+X4+X5+X8)  
anova(model2, model1)

## Analysis of Variance Table  
##   
## Model 1: Y ~ X1 + X2 + X3 + X4 + X5 + X8  
## Model 2: Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 1023 110856   
## 2 1021 110428 2 427.81 1.9777 0.1389

basemodel <- lm(Y ~ 1)  
mod <- ~ X1+X2+X3+X4+X5+X6+X7+X8  
step(basemodel, mod, direction = 'forward')

## Start: AIC=5801.44  
## Y ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + X1 1 71172 216001 5510.1  
## + X5 1 38490 248683 5655.2  
## + X8 1 31061 256112 5685.5  
## + X4 1 24087 263086 5713.2  
## + X7 1 8033 279140 5774.2  
## + X6 1 7811 279362 5775.0  
## + X2 1 5220 281953 5784.5  
## + X3 1 3212 283961 5791.9  
## <none> 287173 5801.4  
##   
## Step: AIC=5510.1  
## Y ~ X1  
##   
## Df Sum of Sq RSS AIC  
## + X5 1 29646.5 186354 5360.0  
## + X8 1 23993.8 192007 5390.8  
## + X2 1 22957.4 193043 5396.4  
## + X4 1 17926.8 198074 5422.9  
## + X6 1 3548.0 212453 5495.0  
## + X3 1 2894.4 213106 5498.2  
## + X7 1 960.2 215041 5507.5  
## <none> 216001 5510.1  
##   
## Step: AIC=5360.03  
## Y ~ X1 + X5  
##   
## Df Sum of Sq RSS AIC  
## + X8 1 37498 148857 5130.6  
## + X2 1 19456 166898 5248.5  
## + X7 1 5862 180493 5329.1  
## + X4 1 782 185572 5357.7  
## + X3 1 741 185613 5357.9  
## <none> 186354 5360.0  
## + X6 1 241 186113 5360.7  
##   
## Step: AIC=5130.63  
## Y ~ X1 + X5 + X8  
##   
## Df Sum of Sq RSS AIC  
## + X2 1 19908.5 128948 4984.7  
## + X4 1 4868.8 143988 5098.4  
## + X7 1 3385.5 145471 5108.9  
## + X3 1 323.9 148533 5130.4  
## <none> 148857 5130.6  
## + X6 1 36.9 148820 5132.4  
##   
## Step: AIC=4984.75  
## Y ~ X1 + X5 + X8 + X2  
##   
## Df Sum of Sq RSS AIC  
## + X4 1 9544.7 119403 4907.5  
## + X3 1 6524.7 122423 4933.3  
## + X6 1 1737.0 127211 4972.8  
## <none> 128948 4984.7  
## + X7 1 3.5 128945 4986.7  
##   
## Step: AIC=4907.54  
## Y ~ X1 + X5 + X8 + X2 + X4  
##   
## Df Sum of Sq RSS AIC  
## + X3 1 8547.4 110856 4833.0  
## + X7 1 1895.7 117508 4893.1  
## <none> 119403 4907.5  
## + X6 1 24.1 119379 4909.3  
##   
## Step: AIC=4833.03  
## Y ~ X1 + X5 + X8 + X2 + X4 + X3  
##   
## Df Sum of Sq RSS AIC  
## <none> 110856 4833.0  
## + X6 1 44.271 110812 4834.6  
## + X7 1 29.398 110827 4834.8

##   
## Call:  
## lm(formula = Y ~ X1 + X5 + X8 + X2 + X4 + X3)  
##   
## Coefficients:  
## (Intercept) X1 X5 X8 X2   
## 29.03022 0.10543 0.23900 0.11349 0.08649   
## X4 X3   
## -0.21829 0.06871

The AIC measures the relative quality of our model. We observe AIC=5801.44 at the intercept which tells us that we’re losing a lot of information. Our test then moves forward to include our predictor X1 (Concrete compressive strength(MPa, megapascals)) and our AIC jumps down by almost 300. Our AIC also lowers significantly as we add our next two predictors X5 (Superplasticizer (component 5)(kg in a m^3 mixture)) and X8 (Age (day)). Our model continues to improve as we add in other predictors that help improve our model but we end up leaving out our X6 and X7 as they would actually raise our AIC as it takes into account the complexity of the model as well as goodness of fit. The removal of these variables agrees with our previous intuition from the Partial F-Test and Added Varaiable plots.

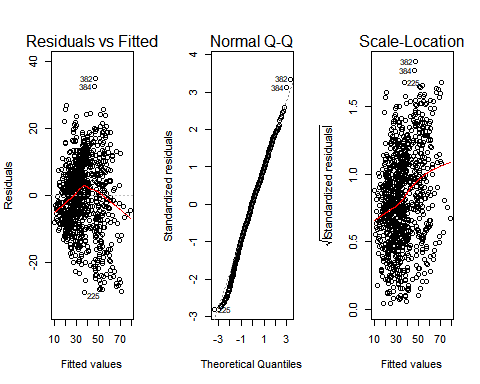
summary(model2)

##   
## Call:  
## lm(formula = Y ~ X1 + X2 + X3 + X4 + X5 + X8)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -29.014 -6.474 0.650 6.546 34.726   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 29.030224 4.212476 6.891 9.64e-12 \*\*\*  
## X1 0.105427 0.004248 24.821 < 2e-16 \*\*\*  
## X2 0.086494 0.004975 17.386 < 2e-16 \*\*\*  
## X3 0.068708 0.007736 8.881 < 2e-16 \*\*\*  
## X4 -0.218292 0.021128 -10.332 < 2e-16 \*\*\*  
## X5 0.239003 0.084586 2.826 0.00481 \*\*   
## X8 0.113495 0.005408 20.987 < 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: 0.6117   
## F-statistic: 271.2 on 6 and 1023 DF, p-value: < 2.2e-16

# differerence in R^2 from model1 to model2 is around .001

We move forward and check our data to make sure it alligns with the required diagnostics:

par(mfrow=c(1,3))  
plot(model2, which = 1:3)

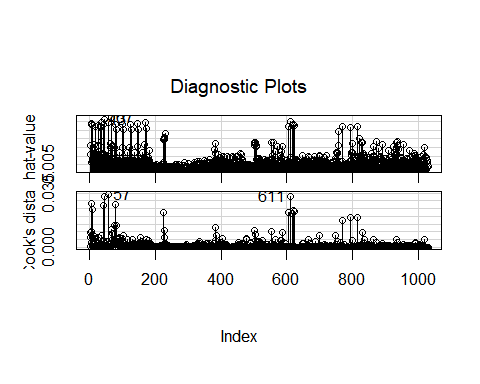


The data is never perfect, however, from the plots we can assert several aspects about our data. In the Residuals vs Fitted plot we see that our data points(regression line) more or less stay consistent with our dotted line at the horizontal axis, signifying linearity. We discuss the aspect of variance in the scale-Location plot, but we can already spot a constant trend as our data experiences a slight uptick as the predicted values increase. In the Q-Q plot we see a proper trend along a 45 degree line for the majority of the entries. Which signifies a normal distribution throughout the majority of our data. Examing our Scale-Location plot, we find a fairly horizontal line spanning the entire predictor range as well as a seemingly even and random spread of points throughout. This validates our assumption of constant variance in our residuals.

After verfying our assumptions are true and determining that no transformations are warranted, we have further reinforced the legitimacy of our model.

Looking at our hat-values and Cook’s Distance values we see there are several observations of interest that we need to look more into as their leverages are well above double the (p+1)/n “average.” We then run an outlier test which tells us that there aren’t any studentized residuals with significance level over 0.05, meaning although these observations may have high leverage, we don’t need need to remove them as their influence is insignificant.

influenceIndexPlot(model2, var = c('hat','Cook'))



outlierTest(model2)

## No Studentized residuals with Bonferroni p < 0.05  
## Largest |rstudent|:  
## rstudent unadjusted p-value Bonferroni p  
## 382 3.366138 0.00079066 0.81438

Now that we have a proper model at our disposal we use it note the difference that slightly above average concentration of cement and superplastizer mixture along with age can have on the Compressive Strength; when we hold all other accompanying variable in our model constant. For a proper comparison we set all variables that are not uniquely defined to the mean of that respective componenet so as to illustrate the effect of the uniquely defined components

# Prediction Interval  
summary(Concrete\_Data)

## Cement (component 1)(kg in a m^3 mixture)  
## Min. :102.0   
## 1st Qu.:192.4   
## Median :272.9   
## Mean :281.2   
## 3rd Qu.:350.0   
## Max. :540.0   
## Blast Furnace Slag (component 2)(kg in a m^3 mixture)  
## Min. : 0.0   
## 1st Qu.: 0.0   
## Median : 22.0   
## Mean : 73.9   
## 3rd Qu.:142.9   
## Max. :359.4   
## Fly Ash (component 3)(kg in a m^3 mixture)  
## Min. : 0.00   
## 1st Qu.: 0.00   
## Median : 0.00   
## Mean : 54.19   
## 3rd Qu.:118.27   
## Max. :200.10   
## Water (component 4)(kg in a m^3 mixture)  
## Min. :121.8   
## 1st Qu.:164.9   
## Median :185.0   
## Mean :181.6   
## 3rd Qu.:192.0   
## Max. :247.0   
## Superplasticizer (component 5)(kg in a m^3 mixture)  
## Min. : 0.000   
## 1st Qu.: 0.000   
## Median : 6.350   
## Mean : 6.203   
## 3rd Qu.:10.160   
## Max. :32.200   
## Coarse Aggregate (component 6)(kg in a m^3 mixture)  
## Min. : 801.0   
## 1st Qu.: 932.0   
## Median : 968.0   
## Mean : 972.9   
## 3rd Qu.:1029.4   
## Max. :1145.0   
## Fine Aggregate (component 7)(kg in a m^3 mixture) Age (day)   
## Min. :594.0 Min. : 1.00   
## 1st Qu.:731.0 1st Qu.: 7.00   
## Median :779.5 Median : 28.00   
## Mean :773.6 Mean : 45.66   
## 3rd Qu.:824.0 3rd Qu.: 56.00   
## Max. :992.6 Max. :365.00   
## Concrete compressive strength(MPa, megapascals)  
## Min. : 2.332   
## 1st Qu.:23.707   
## Median :34.443   
## Mean :35.818   
## 3rd Qu.:46.136   
## Max. :82.599

tester1 <- data.frame(X1=mean(X1), X2=mean(X2), X3=mean(X3), X4=mean(X4), X5=mean(X5), X8=mean(X8))  
tester2 <- data.frame(X1=400, X2=mean(X2), X3=mean(X3), X4=mean(X4), X5=15, X8=50)  
ans1 <- predict(model2, tester1, se.fit = TRUE, interval = 'prediction', level = .95, type = 'response')  
ans2 <- predict(model2, tester2, se.fit = TRUE, interval = 'prediction', level = .95, type = 'response')  
ans1$fit

## fit lwr upr  
## 1 35.81784 15.38095 56.25472

ans2$fit

## fit lwr upr  
## 1 50.94105 30.45538 71.42671

With the average concentrations of each component we report that we are 95% certain that this will produce a Compressive Strength in the range of (15.38, 56.25) (specifically 35.8). And with the same percentage of certainty we report that with slightly above average values of components of Cement, Superplaticizer, and Age we will have a Compressive Strength in the region of (30.46, 71.43) (specifically 50.9).

Conclusion

Our suitable model contains the components of Cement, Blast furnance, fly ash, water, superplaticizer, and age. We elected to drop the components of Coarse Aggregate and Fine Aggregate. We justified this decision through multiple processes of model selection such as AIC, Partial F tests, and Added Variable plots. Moving forward with this set of predictors, we examined the selection of this model through diagnostic residual plots, confirming that our model met all of the assumptions of a linear regression model. We then determined if any outlier removal was in order based on investigation into points with high Cook’s distance and leverage values using an outlier test. However this concluded that no such outlier removal was needed. In our prediction section we decided to illustrate the significance of the statistics at hand. We made unique alterations to Cement, Superplasticizer, and Age becasue these were deemed as the most influential positive predictors in our model. The average cost for the cement is roughly $32. We upped that investment to $45 and noted a predicted increase in compressive strength by 15 MPa. Put into perspective, the maximum Compressive strength evaluation is 82.599. We conclude that a targetted increase in significant components of a structure, such as cement, can optimize the products efficiency substantially.

Appendix

1. Defining Variables

* # Concrete Compressive Strength  
  Y <- Concrete\_Data$`Concrete compressive strength(MPa, megapascals)`  
  X1 <- Concrete\_Data$`Cement (component 1)(kg in a m^3 mixture)`  
  X2 <- Concrete\_Data$`Blast Furnace Slag (component 2)(kg in a m^3 mixture)`  
  X3 <- Concrete\_Data$`Fly Ash (component 3)(kg in a m^3 mixture)`  
  X4 <- Concrete\_Data$`Water (component 4)(kg in a m^3 mixture)`  
  X5 <- Concrete\_Data$`Superplasticizer (component 5)(kg in a m^3 mixture)`  
  X6 <- Concrete\_Data$`Coarse Aggregate (component 6)(kg in a m^3 mixture)`  
  X7 <- Concrete\_Data$`Fine Aggregate (component 7)(kg in a m^3 mixture)`  
  X8 <- Concrete\_Data$`Age (day)`

1. Dataset and Packages Used

* We used Cheng Yehs’ “Modeling of strength of high performance concrete using artificial neural networks” dataset which we obtained from the UCI Machine Learning Repository.
* Relevant Packages include
  + MASS
  + Car

1. Alternate Model Selection and Explaination

We considered a different model for our analysis but moved forward without transformations for reasons of interpretability. The following is the considered transformed model which had the lowest AIC. In this model we removed X3 after checking both the marginal plot with the optimal power transform of X3, and untransformed X3 against Y. These resulted in a null plot which indicated removal was ok.

```r  
library(car)  
X2\_c=X2+1  
X3\_c=X3+1  
X5\_c=X5+1  
conc.pt= powerTransform(cbind(X1, X2\_c, X3\_c, X4, X5\_c, X6, X7, X8)~1, family= "bcPower")  
summary(conc.pt)  
```  
  
```  
## bcPower Transformations to Multinormality   
## Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd  
## X1 0.5346 0.50 0.4015 0.6677  
## X2\_c 0.1728 0.17 0.1317 0.2139  
## X3\_c -0.0830 -0.08 -0.1282 -0.0378  
## X4 1.1093 1.00 0.8268 1.3918  
## X5\_c 0.2227 0.22 0.1591 0.2863  
## X6 1.1660 1.00 0.6126 1.7195  
## X7 1.4078 1.41 1.0214 1.7942  
## X8 0.0557 0.06 0.0101 0.1012  
##   
## Likelihood ratio test that transformation parameters are equal to 0  
## (all log transformations)  
## LRT df pval  
## LR test, lambda = (0 0 0 0 0 0 0 0) 344.0189 8 < 2.22e-16  
##   
## Likelihood ratio test that no transformations are needed  
## LRT df pval  
## LR test, lambda = (1 1 1 1 1 1 1 1) 5897.883 8 < 2.22e-16  
```  
  
```r  
X1\_new= sqrt(X1)  
X2\_new= X2^.17  
X5\_new= X5^.22  
X7\_new= X7^1.41  
X8\_new= X8^.06  
testTransform(conc.pt, lambda=c(.50, .17, -.08, 1, .22, 1, 1.41, .06))  
```  
  
```  
## LRT df pval  
## LR test, lambda = (0.5 0.17 -0.08 1 0.22 1 1.41 0.06) 1.358178 8 0.99482  
```  
  
```r  
conc.best.lm= lm(Y~ X1\_new + X2\_new + X4 + X5\_new + X6 + X7\_new + X8\_new)  
summary(conc.best.lm)  
```  
  
```  
##   
## Call:  
## lm(formula = Y ~ X1\_new + X2\_new + X4 + X5\_new + X6 + X7\_new +   
## X8\_new)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -22.0094 -4.7750 0.1668 4.3707 26.9496   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.066e+02 1.200e+01 -8.886 < 2e-16 \*\*\*  
## X1\_new 2.814e+00 9.836e-02 28.611 < 2e-16 \*\*\*  
## X2\_new 4.269e+00 2.831e-01 15.077 < 2e-16 \*\*\*  
## X4 -2.256e-01 2.117e-02 -10.654 < 2e-16 \*\*\*  
## X5\_new 3.928e+00 4.473e-01 8.781 < 2e-16 \*\*\*  
## X6 -2.472e-03 4.860e-03 -0.509 0.611189   
## X7\_new -8.325e-04 2.225e-04 -3.742 0.000192 \*\*\*  
## X8\_new 1.155e+02 2.709e+00 42.636 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7.359 on 1022 degrees of freedom  
## Multiple R-squared: 0.8073, Adjusted R-squared: 0.806   
## F-statistic: 611.6 on 7 and 1022 DF, p-value: < 2.2e-16  
```

To choose the transformations for our predictors we use the multivariate version of Box-Cox transformation. We had to add a small constant to the vectors X2, X3, and X5 because Box-Cox transformation requires strictly positive entries and these vectors contained 0 values. After transforming our predictors we need to check the transformation for our response Y.

```r  
comp.strength.pt= powerTransform(Y~., Concrete\_Data)  
```  
  
```  
## Warning in estimateTransform.default(X, Y, weights, family, ...):  
## Convergence failure: return code = 52  
```  
  
```r  
summary(comp.strength.pt)  
```  
  
```  
## bcPower Transformation to Normality   
## Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd  
## Y1 1 1 1 1  
##   
## Likelihood ratio test that transformation parameter is equal to 0  
## (log transformation)  
## LRT df pval  
## LR test, lambda = (0) 24513.33 1 < 2.22e-16  
##   
## Likelihood ratio test that no transformation is needed  
## LRT df pval  
## LR test, lambda = (1) -43186.86 1 1  
```

This tells us that we should leave Y untransformed. Now we can proceed with AIC model selection using the newly transformed model. We ran the AIC selection process using the optimal transformation powers but omit it here because the optimal powers provide little real life interpretation and application. Instead we considered rounding the transformations to their closest interpretable values and doing AIC selection process again, (0, .5, 1 represent a log(x), sqrt(x), and no transformation on x, respectively).

```r  
smallest.mod = lm(Y~1)  
conc.best.lm1= lm(Y ~ sqrt(X1) + log(X2\_c) + X4 + log(X5\_c) + X6 + X7 + log(X8))  
trsf\_mod.step1= step(smallest.mod, scope= list(lower=smallest.mod, upper=conc.best.lm1, test= F))  
```  
  
```  
## Start: AIC=5801.44  
## Y ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + log(X8) 1 87561 199612 5428.8  
## + sqrt(X1) 1 70799 216374 5511.9  
## + log(X5\_c) 1 32627 254546 5679.2  
## + X4 1 24087 263086 5713.2  
## + log(X2\_c) 1 12849 274324 5756.3  
## + X7 1 8033 279140 5774.2  
## + X6 1 7811 279362 5775.0  
## <none> 287173 5801.4  
##   
## Step: AIC=5428.82  
## Y ~ log(X8)  
##   
## Df Sum of Sq RSS AIC  
## + sqrt(X1) 1 70699 128912 4980.5  
## + X4 1 42921 156691 5181.5  
## + log(X5\_c) 1 37056 162556 5219.3  
## + log(X2\_c) 1 13337 186275 5359.6  
## + X6 1 5862 193750 5400.1  
## + X7 1 3168 196444 5414.3  
## <none> 199612 5428.8  
## - log(X8) 1 87561 287173 5801.4  
##   
## Step: AIC=4980.46  
## Y ~ log(X8) + sqrt(X1)  
##   
## Df Sum of Sq RSS AIC  
## + log(X5\_c) 1 40400 88512 4595.2  
## + X4 1 34608 94304 4660.5  
## + log(X2\_c) 1 31869 97044 4690.0  
## + X6 1 2501 126412 4962.3  
## <none> 128912 4980.5  
## + X7 1 0 128912 4982.5  
## - sqrt(X1) 1 70699 199612 5428.8  
## - log(X8) 1 87462 216374 5511.9  
##   
## Step: AIC=4595.18  
## Y ~ log(X8) + sqrt(X1) + log(X5\_c)  
##   
## Df Sum of Sq RSS AIC  
## + log(X2\_c) 1 23550 64962 4278.6  
## + X4 1 6090 82422 4523.8  
## + X7 1 1467 87045 4580.0  
## <none> 88512 4595.2  
## + X6 1 2 88510 4597.2  
## - log(X5\_c) 1 40400 128912 4980.5  
## - sqrt(X1) 1 74044 162556 5219.3  
## - log(X8) 1 92093 180604 5327.8  
##   
## Step: AIC=4278.58  
## Y ~ log(X8) + sqrt(X1) + log(X5\_c) + log(X2\_c)  
##   
## Df Sum of Sq RSS AIC  
## + X4 1 9633 55329 4115.3  
## + X6 1 3634 61328 4221.3  
## + X7 1 238 64724 4276.8  
## <none> 64962 4278.6  
## - log(X2\_c) 1 23550 88512 4595.2  
## - log(X5\_c) 1 32081 97044 4690.0  
## - sqrt(X1) 1 89433 154395 5168.3  
## - log(X8) 1 92283 157245 5187.1  
##   
## Step: AIC=4115.26  
## Y ~ log(X8) + sqrt(X1) + log(X5\_c) + log(X2\_c) + X4  
##   
## Df Sum of Sq RSS AIC  
## + X7 1 1002 54327 4098.4  
## + X6 1 391 54939 4110.0  
## <none> 55329 4115.3  
## - log(X5\_c) 1 6008 61337 4219.4  
## - X4 1 9633 64962 4278.6  
## - log(X2\_c) 1 27092 82422 4523.8  
## - sqrt(X1) 1 82833 138162 5055.8  
## - log(X8) 1 100232 155561 5178.0  
##   
## Step: AIC=4098.43  
## Y ~ log(X8) + sqrt(X1) + log(X5\_c) + log(X2\_c) + X4 + X7  
##   
## Df Sum of Sq RSS AIC  
## <none> 54327 4098.4  
## + X6 1 4 54323 4100.4  
## - X7 1 1002 55329 4115.3  
## - log(X5\_c) 1 5503 59830 4195.8  
## - X4 1 10397 64724 4276.8  
## - log(X2\_c) 1 21142 75469 4435.0  
## - sqrt(X1) 1 65968 120295 4915.2  
## - log(X8) 1 99259 153586 5166.8  
```  
  
```r  
trsf\_mod.step1  
```  
  
```  
##   
## Call:  
## lm(formula = Y ~ log(X8) + sqrt(X1) + log(X5\_c) + log(X2\_c) +   
## X4 + X7)  
##   
## Coefficients:  
## (Intercept) log(X8) sqrt(X1) log(X5\_c) log(X2\_c)   
## 4.51096 8.39772 2.85458 2.61305 2.07163   
## X4 X7   
## -0.21767 -0.01537  
```

This model resulted in the lowest AIC of any other model that we tested (AIC=4098.4). Also notice that the selection process for the interpretable transformed model removed X6 (Course Aggregate), but kept X7 (Fine Aggregate). If we had pursued this model, we would have continued with the same procedure as the untransformed model by checking the diagnostics again and then applying our model to the questions of interest.

## End Appendix