Don’t Crack Under The Pressure. Predicting compressive strength of concrete samples.

Eric Chagnon[1], John Dinh[2], and Ignat Kulinka[3]

University of California, Davis

[1] echagnon@ucdavis.edu
[2] jndinh@ucdavis.edu
[3] ikulinka@ucdavis.edu

Abstract

Previous studies have shown that the strength of concrete is dependent not only on water-to-cement ratio, but also other characteristics such as the age of the sample. This paper is aimed at addressing the relationship between concrete attributes relative to its compressive strength. Techniques in multiple regression were used to demonstrate the influence of component quantities and age of the sample , and their interactions, on the compressive strength of concrete. When considering all models including all first order interaction terms (P = 36), the forward stepwise selection procedure reached a plateau of percent change in the BIC after 9 variables were added to the model. The percent change in BIC then increased two times at model sizes of 13 and 16 variables. These three models were then compared with the full model of 21 variables that was output from the forward stepwise procedure. After internal validation on a random 75-25 train/test split, the $MSPE - \frac{SSE}{n}$ terms were compared in order to determine the predictive power and potential overfitting issues of each model. It was found that the model with 16 variables had the best predictive power. However, based on the Principle of Parsimony ("Occam's Razor") the simpler model with 9 variables was selected to be the final model due to the overbearing complexity and diminishing predictive power of the larger models. The final selected model had 10 regression coefficients, and an adjusted R^2 of 0.845.

Introduction

While concrete is a relatively simple material made up of water, cement, aggregates (slag, crushed stone, gravel, etc.), and other additives, it is not an overstatement to say concrete is one of the most important materials in construction (Torre et al. 1). The compressive strength is the amount of force an object can take before reducing in size. Seeing how concrete is used as a load bearing material, being able to accurately measure the compressive strength is an important task. Currently, the compressive strength of concrete is assessed via a compression test. Where a sample of concrete is placed under a cylinder and compressed until fracture (Jamal). This process is not only time consuming, but also potentially wasting valuable resources. This paper attempts to determine the compressive strength of a sample of concrete given the proportion of its components as well as age, and attempts to analyze the intricate relationship between the characteristics of concrete and its compressive strength. Specifically, whether the compressive strength is heavily influenced by only a certain selection of characteristics, or if all the ingredients together are important factors in determining the compressive strength. The dataset being analyzed originated from Chung-Hua University in Taiwan. It can be found at the UCI Machine Learning Repository and has 1030 observations and 9 attributes, including compressive strength.

Methods

Upon initial inspection of the data, every column was represented in an appropriate numeric form and contained no missing values, thus no data cleaning was required. In addition to the usual summary statistics, Table 1, shows number of outliers (e.g. values outside of three standard deviations from the mean) for each of the variables in data. Close examination of the Correlation Scatter Plot Matrix, (Figure 1) shows that Concrete Strength has low to medium positive correlation with Cement, Blast Furnace Slag, Superplaticizer, and Age. On the other hand, Fine Aggregate, Coarse Aggregate, Water, and Fly Ash all have low negative correlation with Concrete Strength. Furthermore, there is potential evidence of multicollinearity among the predictors. Water and Fine Aggregate, Water and Superplasticizer, as well as, Fly Ash and Blast Furnace Slag all have moderate to severe negative correlation. Figures 2 and 3 show the distributions of each of the variables. From these distributions it is apparent that the Age, Blast Furnace Slag, Fly Ash, Cement, and Superplasticizer variables are heavily right skewed and have a large portion of the cases with value of zero. At this point the data was split into a train and test dataset with a 75-25 train-test split. Figure 4 shows the boxplots for each variable in each of the datasets having approximately equal distributions.

Initially a full model with all eight predictors was fit in order to determine if the response variable needed to be transformed via the Box-Cox procedure. In the diagnostic plots for this full model, seen in Figure 5, there are noticeable departures from linear regression assumptions. Namely, the constant variance and the approximate normal distribution of error terms assumptions are slightly overstepped. The output from the Box-Cox procedure in Figure 6 shows that λ has the largest log likelihood when $\lambda \approx 0.5$ which indicates that a square-root transformation may be appropriate for the response variable. Figures 7 and 8 show the distribution of the response variable and the response variable under the square root transformation. The new full model with $\sqrt{Concrete\ Strength}$ as the response variables was used in the model selection process. The resulting model's diagnostic plots from the forward stepwise procedure with no interaction terms can be seen in Figure 9. Since the residual vs fitted values plot depicted a nonlinear pattern, all first order interaction terms were subsequently added to the full model and the forward stepwise procedure was employed again. The diagnostic plots for the resulting model can be seen in Figure 10. This yielded similar results of having a nonlinear pattern in the fitted vs residual plot. Finally, a full model with 2nd order polynomial terms was used in the forward stepwise procedure. The diagnostic plots of the resulting model can be seen in Figure 11. This once again has a nonlinear regression relation. Since there is a strong non-linear trend in every residual vs fitted plot, transformations of the independent variables may be appropriate (Pardoe et al.).

From Figure 2 it is evident that the Age, Blast Furnace Slag, Fly Ash, Cement, and Superplasticizer variables are heavily right skewed. Distributions that are heavily right-skewed traditionally require either a square-root, natural log, or inverse transformation in order for a variable to meet the assumptions for linear regression (Watthanacheewakul 2). For each of these variables, additional plots were made to analyze their distributions under each of these three transformations. These can be seen in Figures 12, 13, 14, 15, and 16. These plots suggest the following: Blast Furnace Slag, Superplasticizer, and Cement approach a normal distribution under the square root transformation. Age approaches a normal distribution under the natural log transformation, and the distribution of Fly Ash does not improve under any of the aforementioned transformations.

After determining the form the predictors should take when they enter the model, an initial model with all of the predictors, and their appropriate transformations, was created in order to determine if the response variable needed to be transformed. The output from the Box-Cox procedure in Figure 17 shows that once again λ has the largest log likelihood when $\lambda \approx 0.5$ which indicates that a square-root transformation may be appropriate for the response variable. Finally, Figure 18 shows the plots that help visualize the assumptions required for linear regression. Since the Residual v Fitted plot shows no relationship between the Residuals and Fitted Value, and the Normal QQ Plot shows that the errors are approximately normally distributed, the assumptions required for linear regression are finally satisfied. Furthermore, none of the predictors exhibited a variance inflation factor of over 6 indicating that multicollinearity is not severe. The full initial model can be seen in Table 2.

Traditionally, variable selection is conducted using either the best possible subset procedure, or the stepwise procedure (Ahn 1). In this case the best possible subset procedure was tried first, since the model had relatively little predictors with $P = 36$. However, the best possible subset method used in the 'leaps' package was adding interaction terms to the model when the main effect terms that make up said interaction were not present. From an interpretability point of view this did not make sense, thus the stepwise procedure was used. The decision to use BIC ($nlog\frac{SSE}{n} + log(n)p$) as the main criterion was due to the fact that BIC penalizes model size more than AIC ($nlog\frac{SSE}{n} + 2p$), and having a smaller model is preferable to having a larger model that is hard to interpret. The reason why BIC penalizes model complexity more severely than AIC is because when $n \geq 8$, then $log(n) \geq 2$, and as a result, the BIC criterion grows much quicker. The output from the stepwise procedure can be seen in Table 3. The percent change in BIC as additional variables are added was included in the right-most column.

From Table 3 it is clear that the first time the BIC decreases by a value less than 1% is at model size of 10 (This model will be referred to as Model 1)[4]. The change in BIC continues to stay less than 1% until model size reaches 14 (Model 2). Here a main effect was added in the previous iteration and its most significant interaction was just added to the model. This happens one more time at model size 17 (Model 3). Finally, the final model output from the stepwise procedure had size 22 (Model 4). In addition, the diagnostic plots of these 4 candidate models (Figures 19, 20, 21, and 22) show slight departure from the constant variance assumption especially in the larger models (e.g. 3 and 4) as demonstrated by a slight "conal" shape in the residuals vs fitted plots. In addition, the same plots show little to no discernable pattern demonstrating that the regression function captures the entirety of the relationship between the response and the predictor variables. There is also an indication of a minor issue with the normality assumption as indicated by the Normal QQ plots. Nonetheless, the diagnostic plots suggest that all of the assumptions are sufficiently met. These four models were selected due to their respective % changes in the BIC, and subsequently underwent validation. The models were internally validated via $PRESS_p$ and C_p. These values can be seen in Table 8. Model 1 has the highest C_p value, which is much higher than its value of $p = 10$. This indicates heavy bias in the model. This is supported by Model 1 having the lowest $PRESS_p$. Model 4 seems to be the least biased since its $C_p \approx p$, and it has a slightly larger $PRESS_p$ than the rest. These internal validation results show heavy bias in all of the models with C_p being much greater than the p values of their respective models.

For external validation the models were tested using the hold out dataset. The $MSPE$ was found for each model and compared to the $\frac{SSE}{n}$ for the model trained on the held out dataset. If these two values are close to each other, it indicates the model is not severely overfit. These values can also be seen in Table 8. These external validation results show that these models all have good predictive power, indicated by the small values of $MSPE$, and no severe overfitting has taken place. Overall, since their predictive powers are relatively the same (seen in the $PRESS_p$ and $MSPE - \frac{SSE}{n}$), the model with the least complexity is the best option by the Principle of Parsimony.

Conclusion

The final model helps provide insight towards the goals of this paper. The diagnostic plots (Figure 23) of the final model fit on the entire dataset show initial model assumptions to be intact. The residuals vs fitted values plot shows no obvious pattern, thereby indicating the

[4] See Tables 4, 5, 6, and 7 for regression summaries for Models 1 through 4.

regression relation is linear and the residuals having constant variance. The normal QQ plots show that the residuals are normally distributed, albeit having a slightly heavy tail. The model is able to explain 84.5% of the total variance in compressive strength, indicating that the compressive strength of a concrete sample can be accurately modeled based on its characteristics. While larger models ended up including all of the variables provided in the dataset, the diminishing returns in the predictive power produced a model that contained six of the original eight predictors. From Table 9, it is evident that Age and Superplasticizer have the largest effect on compressive strength, while Coarse Aggregate and Fine Aggregate proved to be insignificant enough to be excluded from the final model. Thus, the relationship between the Compressive Strength of a concrete sample and its characteristics can be successfully modeled using only a few key variables. Furthermore, the model has good predictive power, as seen by the $MSPE$ in Table 8. Since the $MSPE - \frac{SSE}{n}$ term is close to 0, it is reasonable to say that no severe overfitting has occurred.

To assess how generalizable the model is, additional external data would be used to test the final model's predictive ability. Since the final model has a large C_p value, the model is severely underfit. As a result, using external data or a different test/train split would return different results and $PRESS_p$ values. In order to make the model more generalizable, reducing the model bias is top priority. An angle of approach for this issue is to use a different variable selection process and exhaustively check the residuals of each added variable, main effect term or interaction term, to ensure the importance of each additional variable.

One of the limitations of the analysis goes back to the quality of the data itself. Many of the predictors, namely Fly Ash, Blast Furnace Slag, and Superplasticizer, had large quantities of zeros, effectively making transformations of these variables not as effective as transformations on the other predictor variables (as seen in Figure 2). To deal with these with large amounts of zeros, a feasible transformation to make is to change the variable type to a binary variable. Another limitation is actual effectiveness of the transformations on the other variables. Based on Figure 13, the log transformation on age shows an impractical distribution, resulting in a flawed predictor. In order to side step these issues, ideas in feature engineering could have been utilised in order to design and implement more practical predictors that adhered more strictly to the model assumptions, allowing the model to be more generalizable.

An additional limitation is the amount of bias in the final model. The original goal of the study was to provide a sufficient model to predict the compressive strength of concrete, and the final model's $MSPE$ values reflected good predictive ability. However, the $MSPE$ value was derived from a single train/test split, and although showed good predictive ability, returned a C_p (seen in Table 8) value that was large compared to the number of predictors in the final model. A possible reason for this shortcoming was only performing a single train/test split and performing

external validation on this single split. This single split showed the model had good predictive ability, but a more exhaustive approach would have been testing the model on multiple splits to deduce whether or not the model's predictive ability was sufficient. In addition, testing on multiple splits would also help generalize the model on external data because the concept of bias-variance trade would have been emphasized.

In a similar vein, the final model revealed several influential points in the dataset. These points are important to note because they alter the regression function significantly. Of the 1030 observations, 74 of these observations were high influence points (7.2%). Cook's distance was calculated for each observation and was classified as a high potential influential case if the distance was greater than $\frac{4}{n-p}$. These high influence points affect the regression coefficients of the final model and may drag the fitted line in a certain direction that is away from the actual trend, thereby also affecting the model's predictive ability. Further studies could have been conducted with the removal of these points in order to see any improvements in the final model's fit and performance.

Appendix 1 Figures and Tables

Table 1: Summary Statistics for All Variables

Statistic	Min	Median	Mean	St. Dev.	Max	Outliers[1]
Cement (kg/m^3)	102	272.9	281.166	104.507	540	0
Blast Furnace Slag (kg/m^3)	0	22	73.895	86.279	359	4
Fly Ash (kg/m^3)	0	0	54.187	63.996	200	0
Water (kg/m^3)	122	185	181.566	21.356	247	2
Superplasticizer (kg/m^3)	0	6.350	6.203	5.973	32.200	10
Coarse Aggregate (kg/m^3)	801	968	972.919	77.754	1,145	0
Fine Aggregate (kg/m^3)	594	779.5	773.579	80.175	993	0
Age of Testing (days)	1	28	45.662	63.170	365	33
Concrete Strength (MPa)	2.332	34.443	35.818	16.706	82.599	0

[1] Outliers are defined as values more than three standard deviations from the mean.

Table 2: Summary of the Full Transformed Model

	Dependent variable:
	$\sqrt{\text{Concrete Strength}}$
Constant	-5.422^{***} (1.591)
$\sqrt{\text{Cement}}$	0.350^{***} (0.016)
$\sqrt{\text{Blast Furnace Slag}}$	0.115^{***} (0.008)
Fly_Ash	0.006^{***} (0.001)
Water	-0.010^{***} (0.002)
$\sqrt{\text{Superplasticizer}}$	0.110^{***} (0.024)
Coarse Aggregate	0.002^{***} (0.001)
Fine Aggregate	0.002^{***} (0.001)
log(Age)	0.769^{***} (0.018)
Observations	772
R^2	0.844
Adjusted R^2	0.842
Residual Std. Error	0.586 (df = 763)
F Statistic	514.346^{***} (df = 8; 763)
Note:	*p<0.1; **p<0.05; ***p<0.01

Table 3: Summary of the Forward Stepwise Procedure

	Step	Df	Deviance	Resid. Df	Resid. Dev	BIC	BIC % Change
1				771	1,673.762	604.057	
2	+ age	1	576.550	770	1,097.212	284.689	-52.870
3	+ cement	1	420.608	769	676.604	-81.879	-128.761
4	+ superplasticizer	1	226.606	768	449.998	-390.085	-376.416
5	+ blast_furnace_slag	1	117.749	767	332.249	-617.628	-58.332
6	+ water	1	46.268	766	285.981	-726.749	-17.668
7	+ fly_ash	1	18.152	765	267.829	-770.724	-6.051
8	+ cement:age	1	11.061	764	256.768	-796.634	-3.362
9	+ water:superplasticizer	1	7.062	763	249.706	-811.516	-1.868
10	+ blast_furnace_slag:water	1	5.173	762	244.533	-821.028	-1.172
11	+ fly_ash:superplasticizer	1	3.428	761	241.105	-825.279	-0.518
12	+ cement:blast_furnace_slag	1	3.972	760	237.133	-831.453	-0.748
13	+ coarse_aggregate	1	3.149	759	233.984	-835.124	-0.441
14	+ superplasticizer:coarse_aggregate	1	7.941	758	226.043	-855.131	-2.396
15	+ blast_furnace_slag:coarse_aggregate	1	4.928	757	221.115	-865.497	-1.212
16	+ fine_aggregate	1	4.367	756	216.749	-874.246	-1.011
17	+ fly_ash:fine_aggregate	1	7.085	755	209.664	-893.254	-2.174
18	+ coarse_aggregate:fine_aggregate	1	4.906	754	204.757	-904.886	-1.302
19	+ water:age	1	4.454	753	200.303	-915.217	-1.142
20	- blast_furnace_slag:water	1	0.097	754	200.399	-921.494	-0.686
21	+ cement:water	1	2.605	753	197.794	-924.946	-0.375
22	+ blast_furnace_slag:age	1	2.155	752	195.639	-926.756	-0.196

Table 4: Model 1 Summary

	Dependent variable:
	$\sqrt{\text{Concrete Strength}}$
Constant	1.035^{*} (0.562)
log(Age)	1.274^{***} (0.096)
$\sqrt{\text{Cement}}$	0.405^{***} (0.019)
$\sqrt{\text{Superplasticizer}}$	-0.567^{***} (0.128)
$\sqrt{\text{Blast Furnace Slag}}$	-0.047 (0.035)
Water	-0.029^{***} (0.002)
Fly Ash	0.003^{***} (0.001)
log(Age):$\sqrt{\text{Cement}}$	-0.029^{***} (0.006)
$\sqrt{\text{Superplasticizer}}$:Water	0.004^{***} (0.001)
$\sqrt{\text{Blast Furnace Slag}}$:Water	0.001^{***} (0.0002)
Observations	772
R^2	0.854
Adjusted R^2	0.852
Residual Std. Error	0.566 (df = 762)
F Statistic	494.854^{***} (df = 9; 762)
Note:	*p<0.1; **p<0.05; ***p<0.01

Table 5: Model 2 Summary

	Dependent variable:
	$\sqrt{\text{Concrete Strength}}$
Constant	0.491 (0.873)
log(Age)	1.300^{***} (0.093)
$\sqrt{\text{Cement}}$	0.460^{***} (0.021)
$\sqrt{\text{Superplasticizer}}$	-1.282^{***} (0.235)
$\sqrt{\text{Blast Furnace Slag}}$	0.063 (0.041)
Water	-0.026^{***} (0.003)
Fly Ash	0.007^{***} (0.001)
Coarse Aggregate	-0.001^{**} (0.001)
log(Age):$\sqrt{\text{Cement}}$	-0.031^{***} (0.005)
$\sqrt{\text{Superplasticizer}}$:Water	0.003^{***} (0.001)
$\sqrt{\text{Blast Furnace Slag}}$:Water	0.001^{***} (0.0002)
$\sqrt{\text{Superplasticizer}}$:Fly Ash	-0.002^{***} (0.0004)
$\sqrt{\text{Cement}}$:$\sqrt{\text{Blast Furnace Slag}}$	-0.006^{***} (0.001)
$\sqrt{\text{Superplasticizer}}$:Coarse Aggregate	0.001^{***} (0.0002)
Observations	772
R^2	0.865
Adjusted R^2	0.863
Residual Std. Error	0.546 (df = 758)
F Statistic	373.439^{***} (df = 13; 758)
Note:	*p<0.1; **p<0.05; ***p<0.01

Table 6: Model 3 Summary

	Dependent variable:
	$\sqrt{\text{Concrete Strength}}$
Constant	-5.984^{***} (1.672)
log(Age)	1.344^{***} (0.090)
$\sqrt{\text{Cement}}$	0.539^{***} (0.025)
$\sqrt{\text{Superplasticizer}}$	-1.701^{***} (0.236)
$\sqrt{\text{Blast Furnace Slag}}$	0.482^{***} (0.076)
Water	-0.017^{***} (0.003)
Fly Ash	-0.011^{***} (0.004)
Coarse Aggregate	0.001^{*} (0.001)
Fine Aggregate	0.001^{**} (0.001)
log(Age):$\sqrt{\text{Cement}}$	-0.033^{***} (0.005)
$\sqrt{\text{Superplasticizer}}$:Water	0.003^{***} (0.001)
$\sqrt{\text{Blast Furnace Slag}}$:Water	0.0003 (0.0002)
$\sqrt{\text{Superplasticizer}}$:Fly Ash	-0.001^{***} (0.0004)
$\sqrt{\text{Cement}}$:$\sqrt{\text{Blast Furnace Slag}}$	-0.009^{***} (0.001)
$\sqrt{\text{Superplasticizer}}$:Coarse Aggregate	0.001^{***} (0.0002)
$\sqrt{\text{Blast Furnace Slag}}$:Coarse Aggregate	-0.0003^{***} (0.00005)
Fly Ash:Fine Aggregate	0.00002^{***} (0.00000)
Observations	772
R^2	0.875
Adjusted R^2	0.872
Residual Std. Error	0.527 (df = 755)
F Statistic	329.514^{***} (df = 16; 755)
Note:	*p<0.1; **p<0.05; ***p<0.01

Table 7: Model 4 Summary

	Dependent variable:
	$\sqrt{\text{Concrete Strength}}$
Constant	1.090 (2.879)
log(Age)	1.754^{***} (0.157)
$\sqrt{\text{Cement}}$	0.777^{***} (0.076)
$\sqrt{\text{Superplasticizer}}$	-1.660^{***} (0.239)
$\sqrt{\text{Blast Furnace Slag}}$	0.620^{***} (0.080)
Water	0.020^{**} (0.008)
Fly Ash	-0.020^{***} (0.004)
Coarse Aggregate	-0.012^{***} (0.003)
Fine Aggregate	-0.016^{***} (0.003)
log(Age):$\sqrt{\text{Cement}}$	-0.028^{***} (0.005)
$\sqrt{\text{Superplasticizer}}$:Water	0.002^{***} (0.001)
$\sqrt{\text{Blast Furnace Slag}}$:Water	-0.0003 (0.0002)
$\sqrt{\text{Superplasticizer}}$:Fly_Ash	-0.001^{**} (0.0004)
$\sqrt{\text{Cement}}$:$\sqrt{\text{Blast Furnace Slag}}$	-0.011^{***} (0.001)
$\sqrt{\text{Superplasticizer}}$:Coarse Aggregate	0.001^{***} (0.0002)
$\sqrt{\text{Blast Furnace Slag}}$:Coarse Aggregate	-0.0003^{***} (0.00005)
Fly Ash:Fine Aggregate	0.00003^{***} (0.00001)
Coarse Aggregate:Fine Aggregate	0.00002^{***} (0.00000)
log(Age):Water	-0.003^{***} (0.001)
$\sqrt{\text{Cement}}$:Water	-0.001^{***} (0.0004)
log(Age):$\sqrt{\text{Blast Furnace Slag}}$	0.008^{***} (0.003)
Observations	772
R^2	0.883
Adjusted R^2	0.880
Residual Std. Error	0.510 (df = 751)
F Statistic	284.658^{***} (df = 20; 751)
Note:	*p<0.1; **p<0.05; ***p<0.01

Table 8: Summary of Internal And External Validation Statistics

	p	C_p	$PRESS_p$	$MSPE_v$	$MSPE_v - \frac{SSE}{n}$
Model 1	10	226.775	251.987	0.342	0.025
Model 2	14	160.766	235.954	0.302	0.009
Model 3	17	101.206	220.767	0.271	-0.001
Model 4	22	50.751	208.309	0.276	0.023

Table 9: Summary of Model 1 Fit on All Data

	Dependent variable:
	$\sqrt{\text{Concrete Strength}}$
Constant	0.860^{*} (0.497)
log(Age)	1.285^{***} (0.086)
$\sqrt{\text{Cement}}$	0.405^{***} (0.017)
$\sqrt{\text{Superplasticizer}}$	-0.441^{***} (0.114)
$\sqrt{\text{Blast Furnace Slag}}$	-0.056^{*} (0.031)
Water	-0.028^{***} (0.002)
Fly Ash	0.003^{***} (0.0005)
log(Age):$\sqrt{\text{Cement}}$	-0.030^{***} (0.005)
$\sqrt{\text{Superplasticizer}}$:Water	0.003^{***} (0.001)
$\sqrt{\text{Blast Furnace Slag}}$:Water	0.001^{***} (0.0002)
Observations	1,030
R^2	0.847
Adjusted R^2	0.845
Residual Std. Error	0.570 (df = 1020)
F Statistic	625.203^{***} (df = 9; 1020)
Note:	*p<0.1; **p<0.05; ***p<0.01

Figure 1: Correlation Scatter Plot Matrix

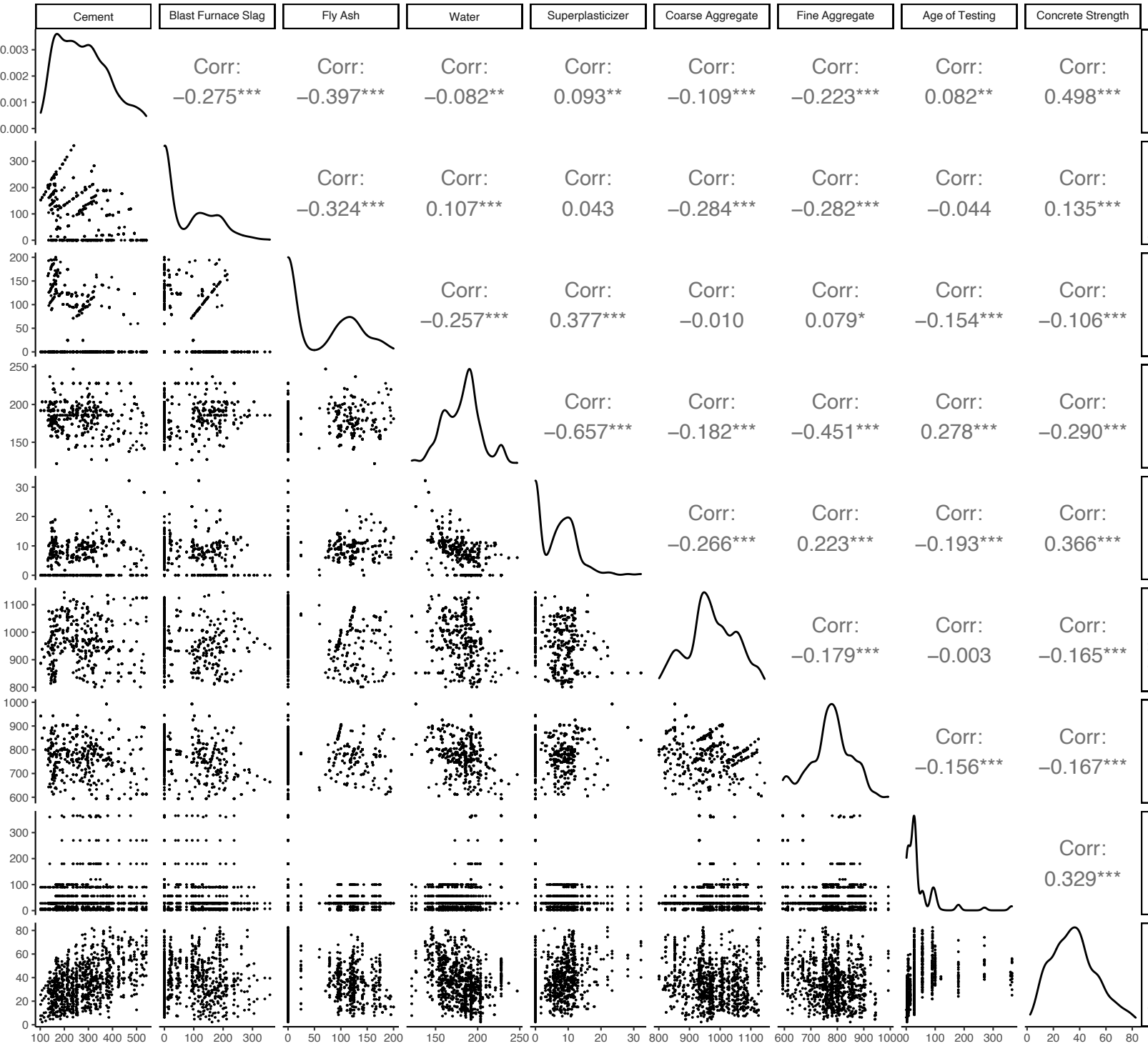


Figure 2: Histograms of Untransformed Data

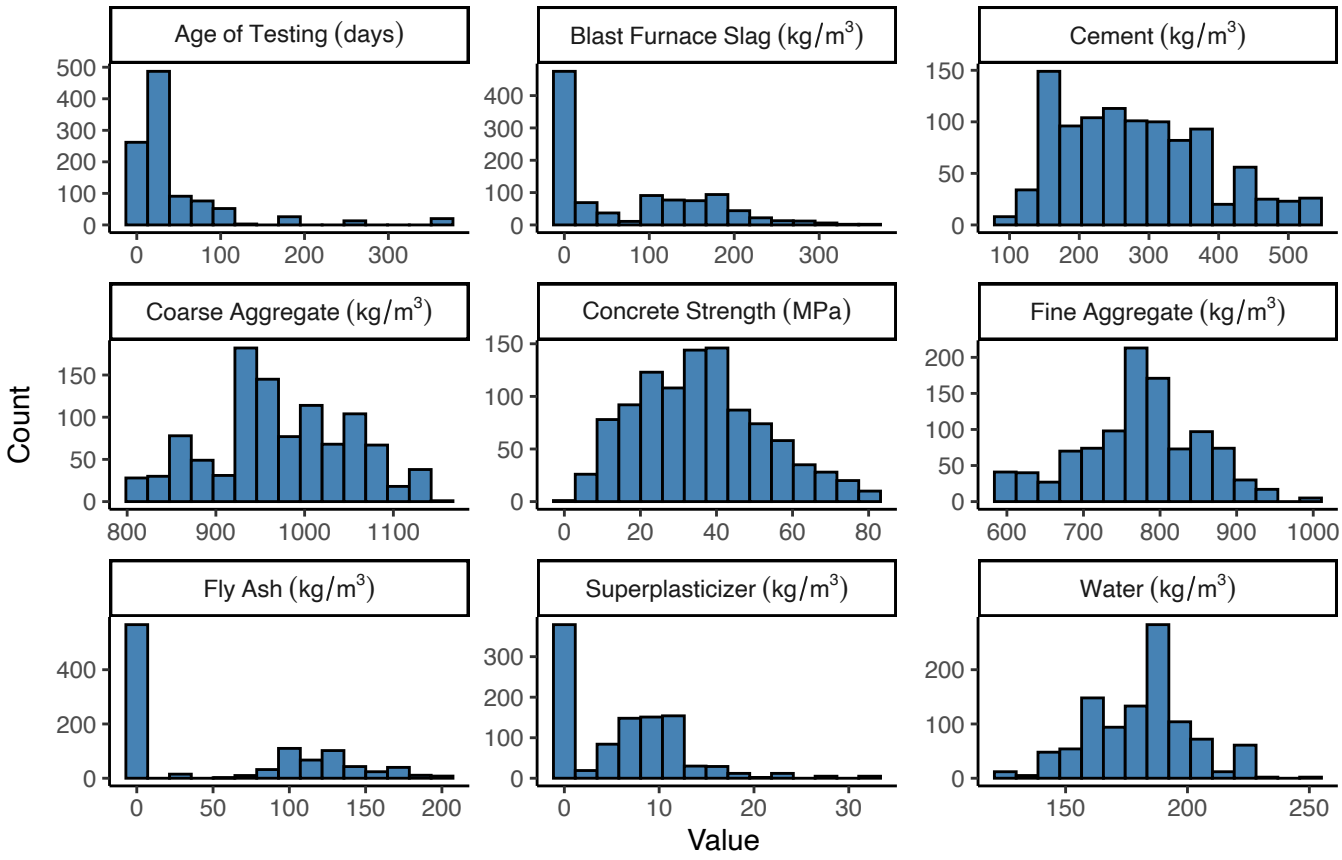


Figure 3: Box Plots of Untransformed Data

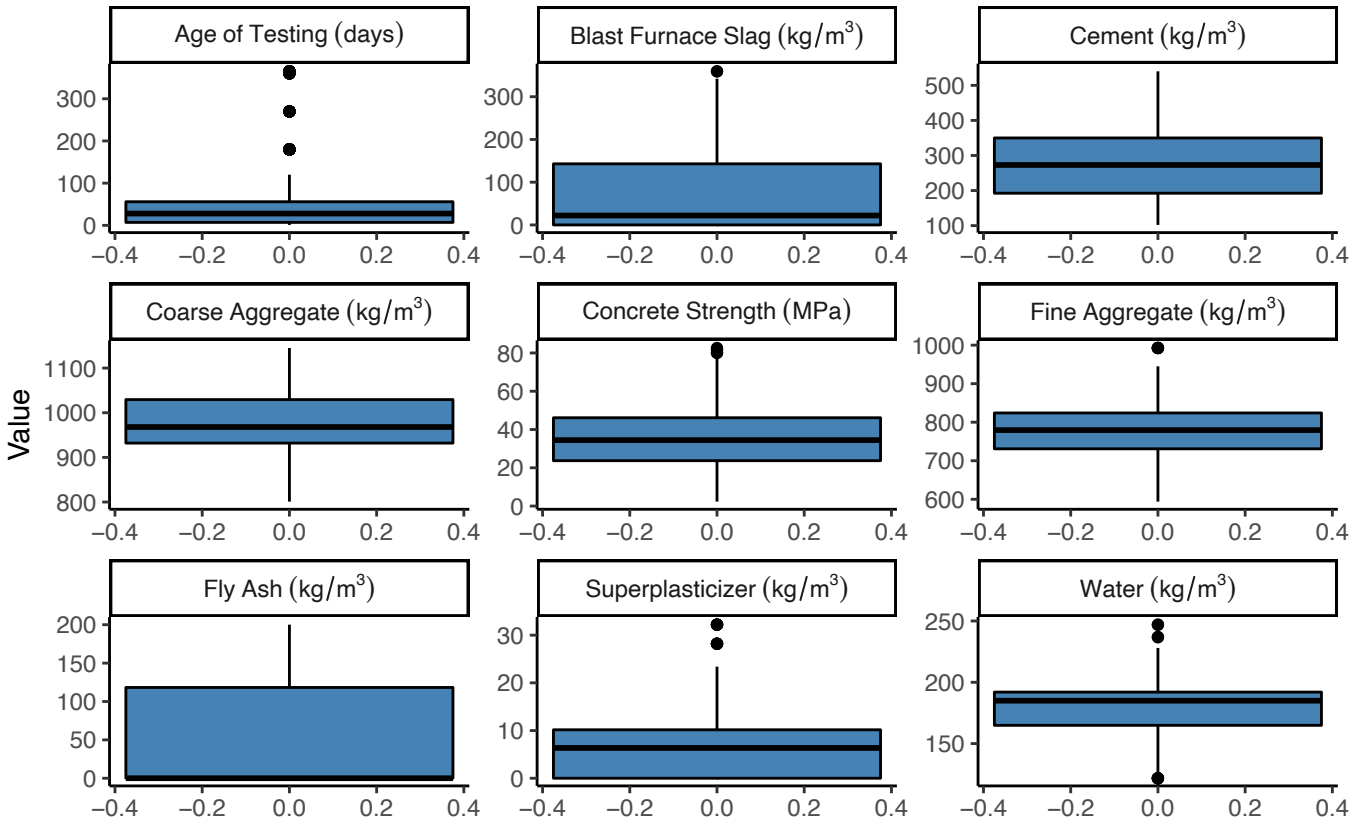


Figure 4: Comparison of Training and Test Data

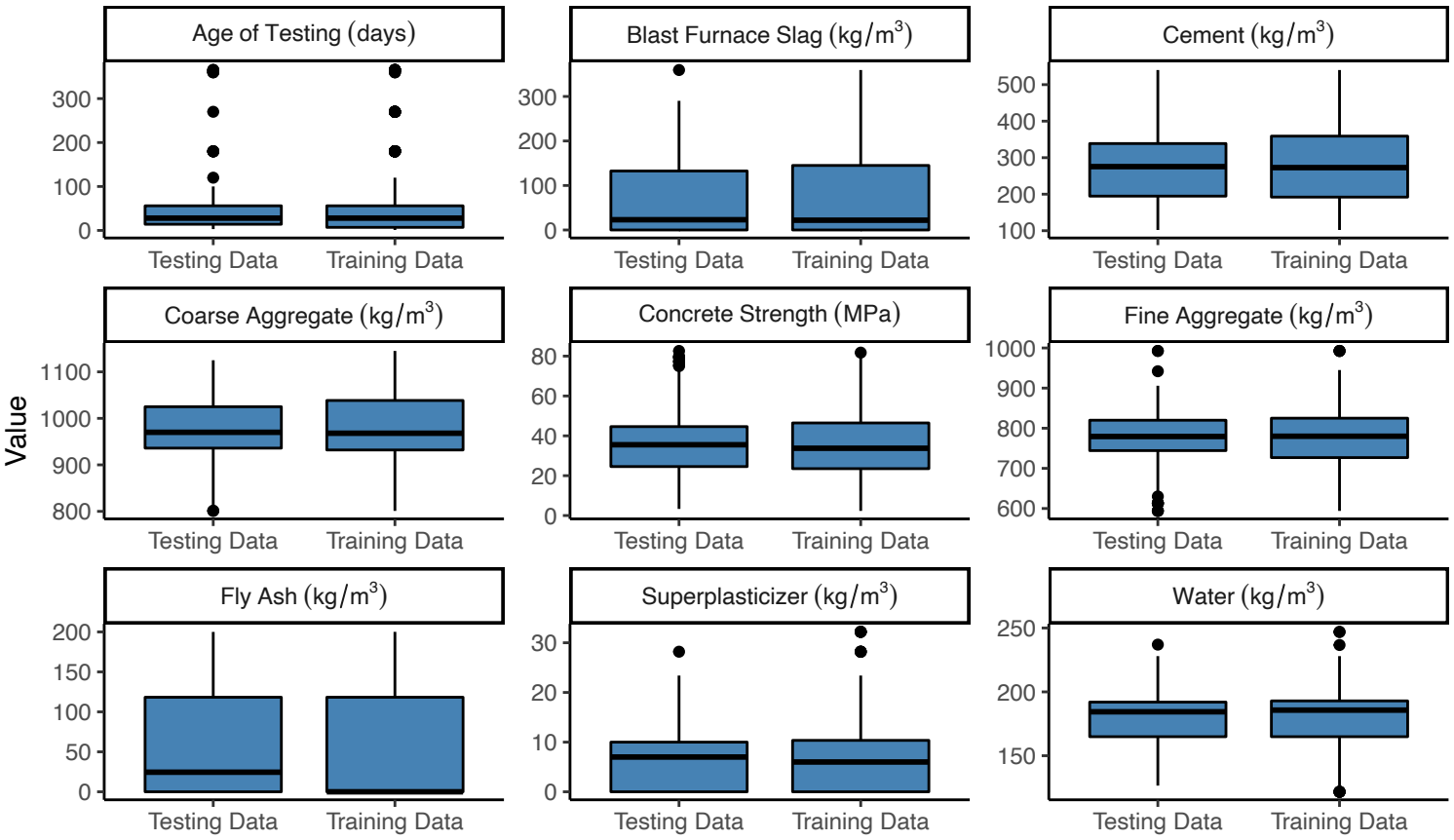


Figure 5: Diagnostic Plots For Full Untransformed Model

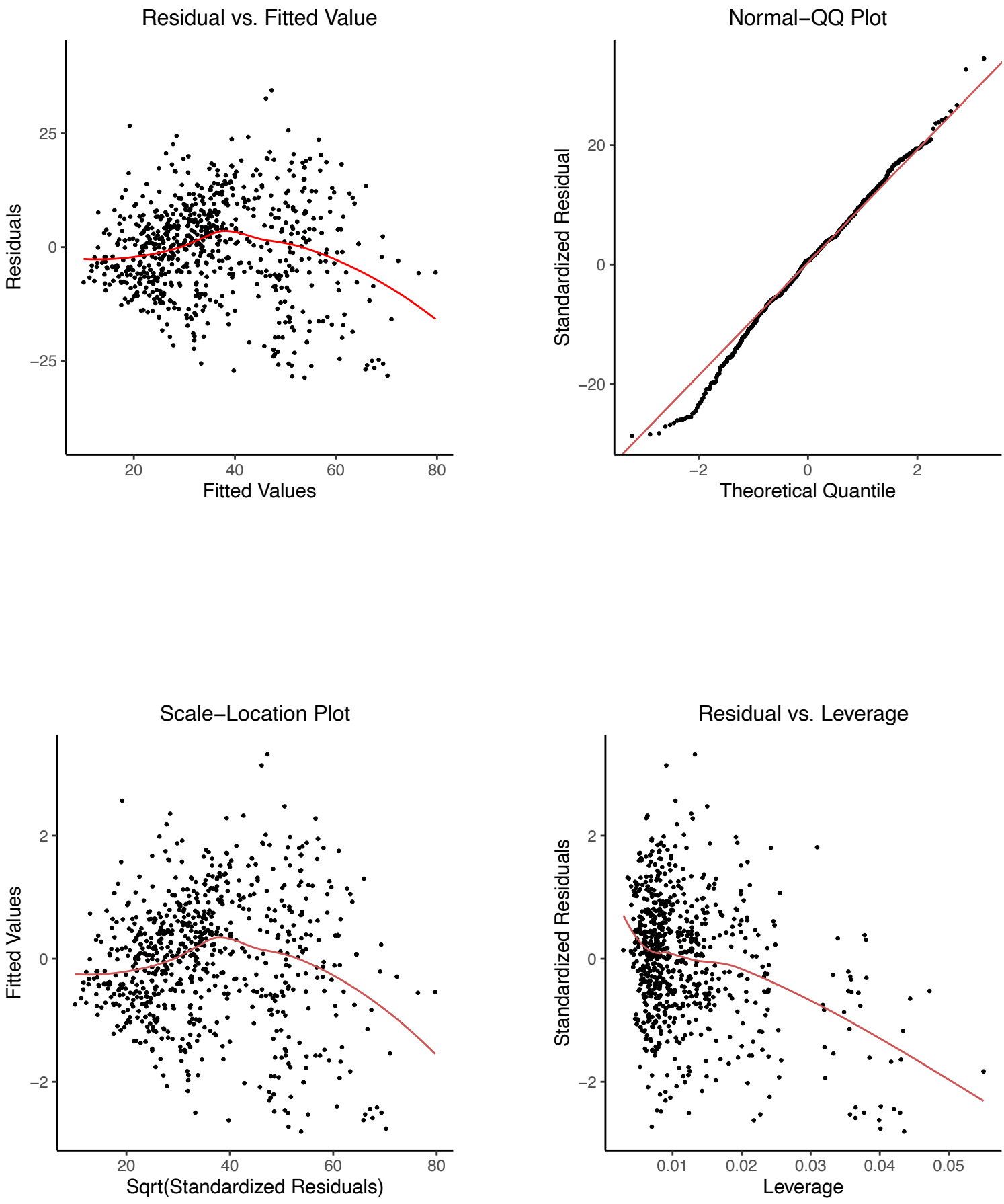


Figure 6: Box–Cox Graph for Full Initital Model

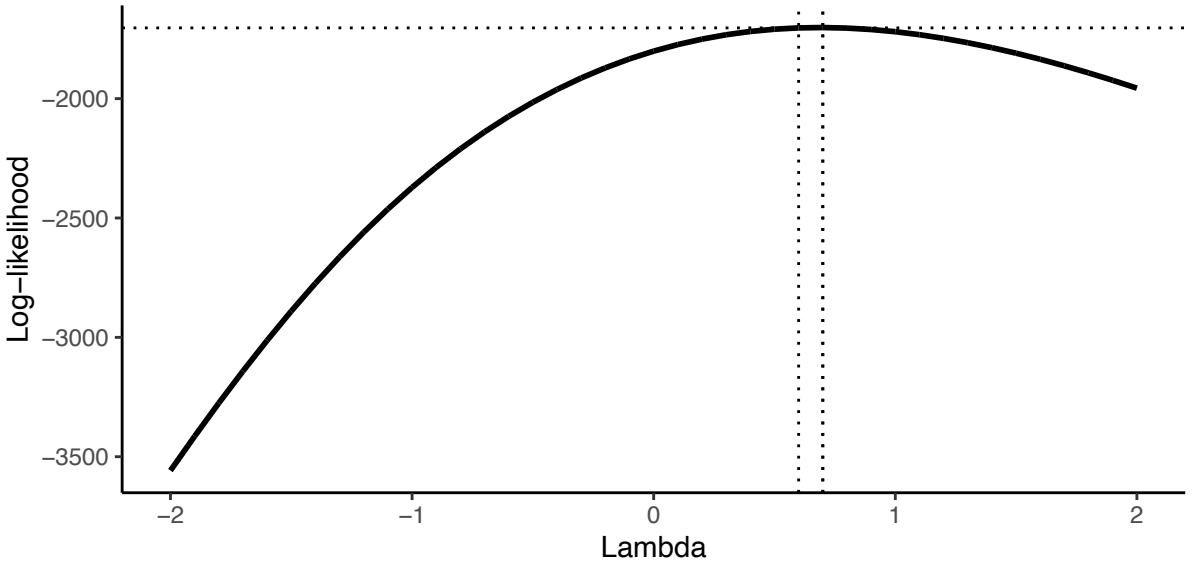


Figure 7: Histogram of Concrete Strength (MPa)

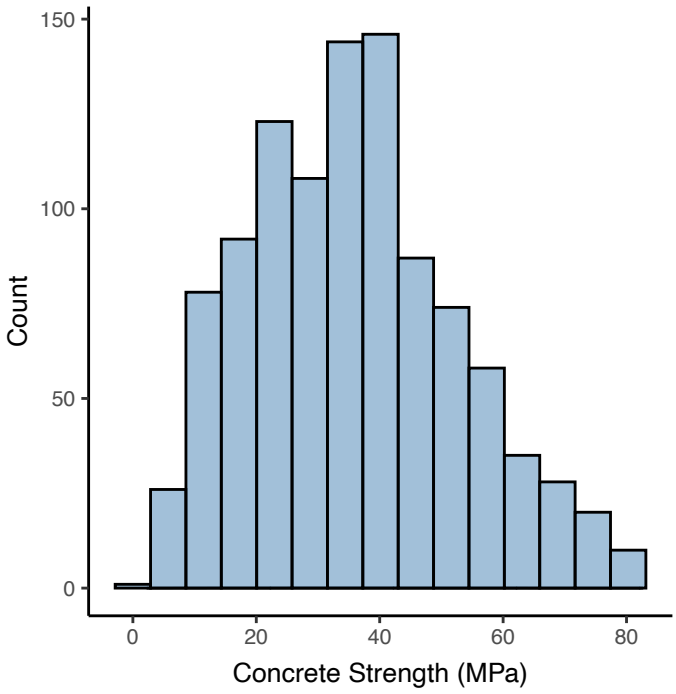


Figure 8: Histogram of $\sqrt{\text{Concrete Strength (MPa)}}$

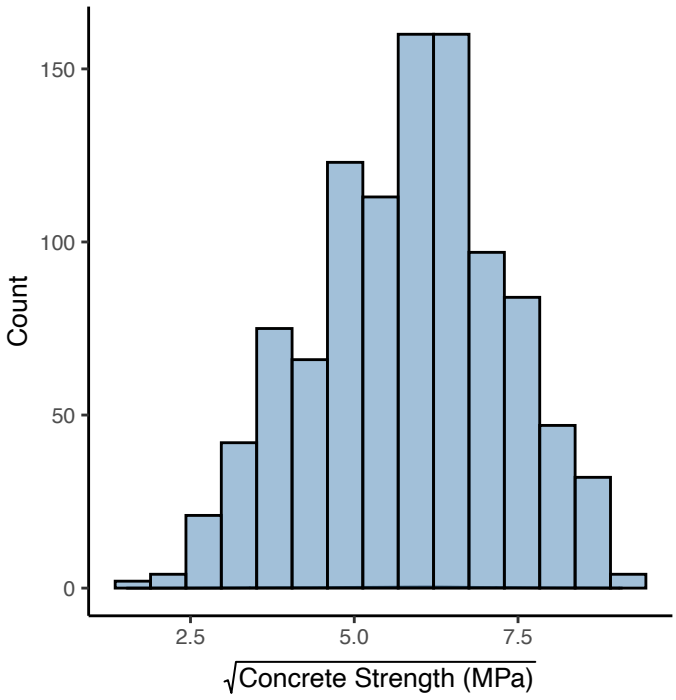


Figure 9: Diagnostic Plots For Forward Stepwise Model Without Interaction Effects

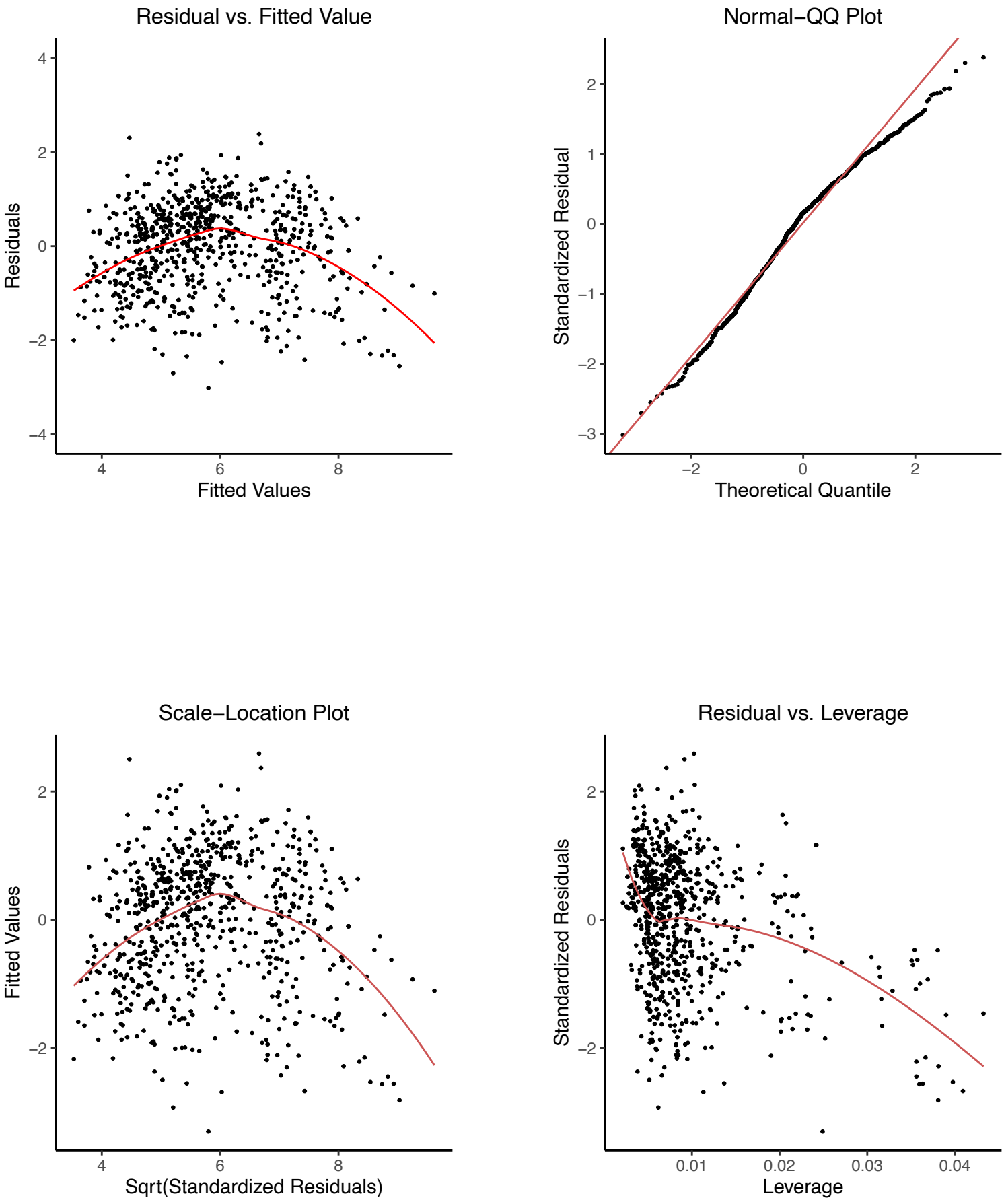


Figure 10: Diagnostic Plots For Forward Stepwise Model With First–Order Interactions

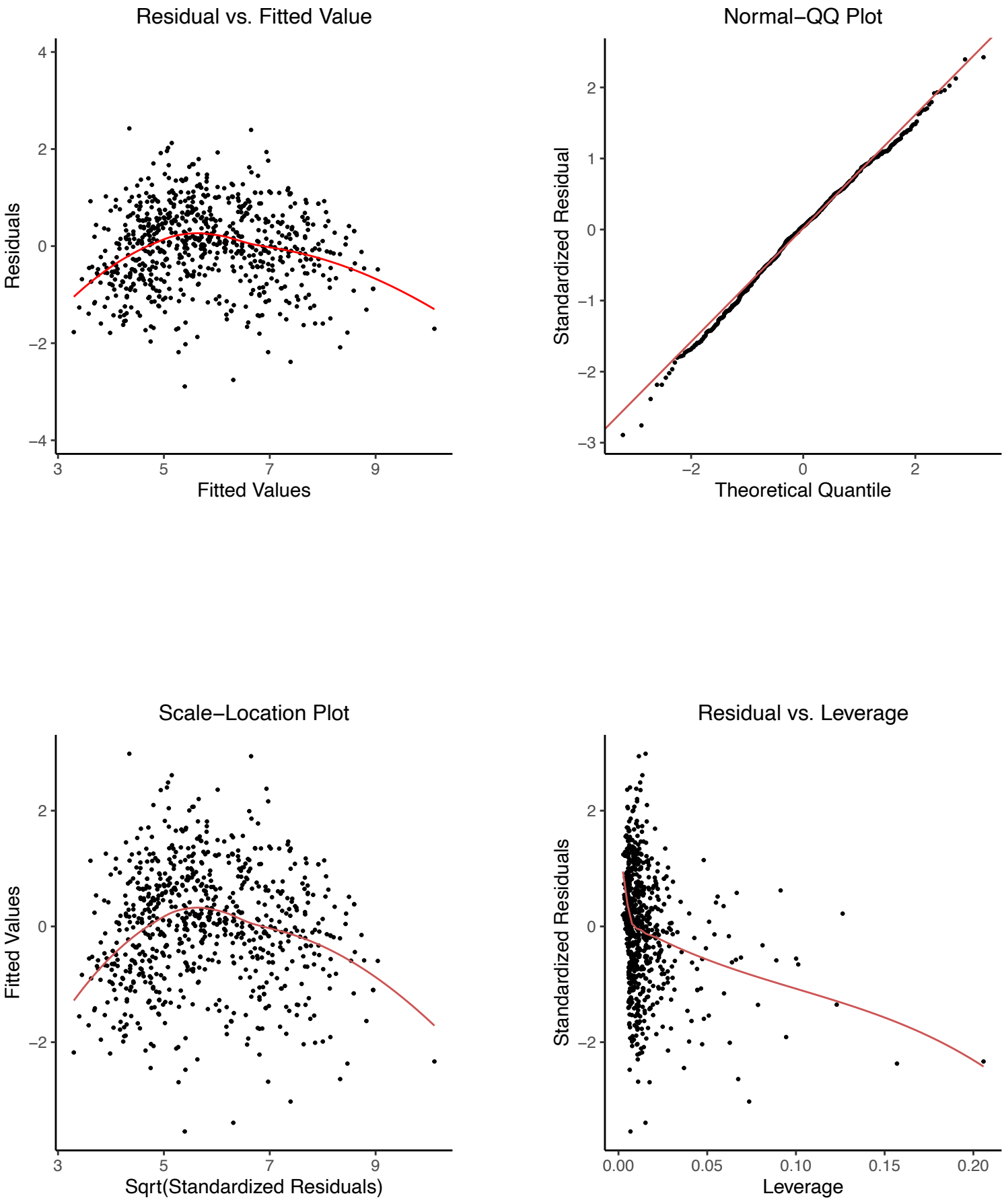


Figure 11: Diagnostic Plots For Forward Stepwise Second–Order Polynomial Model Without Interactions

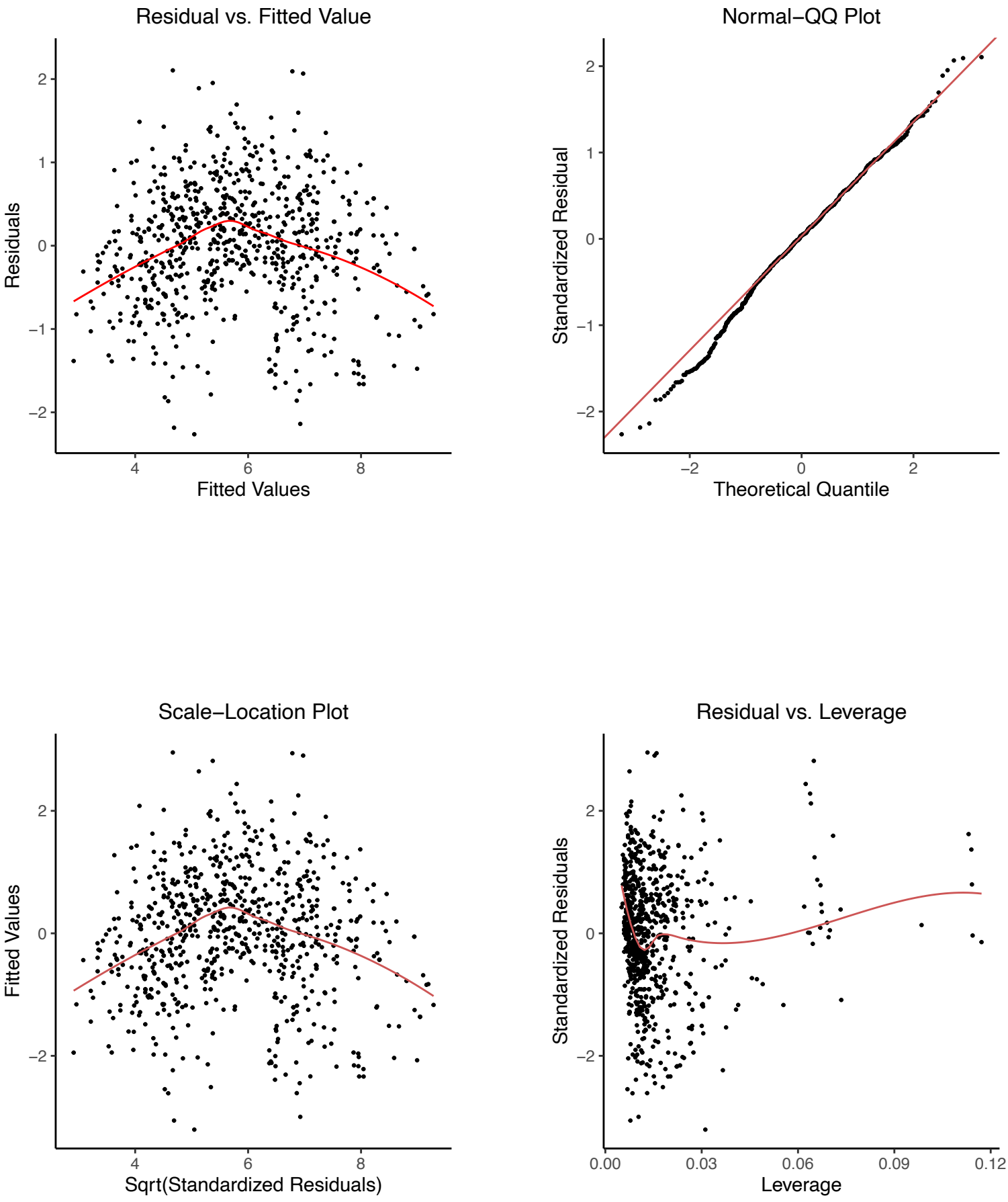


Figure 12: Cement (kg/m^3) Transformations

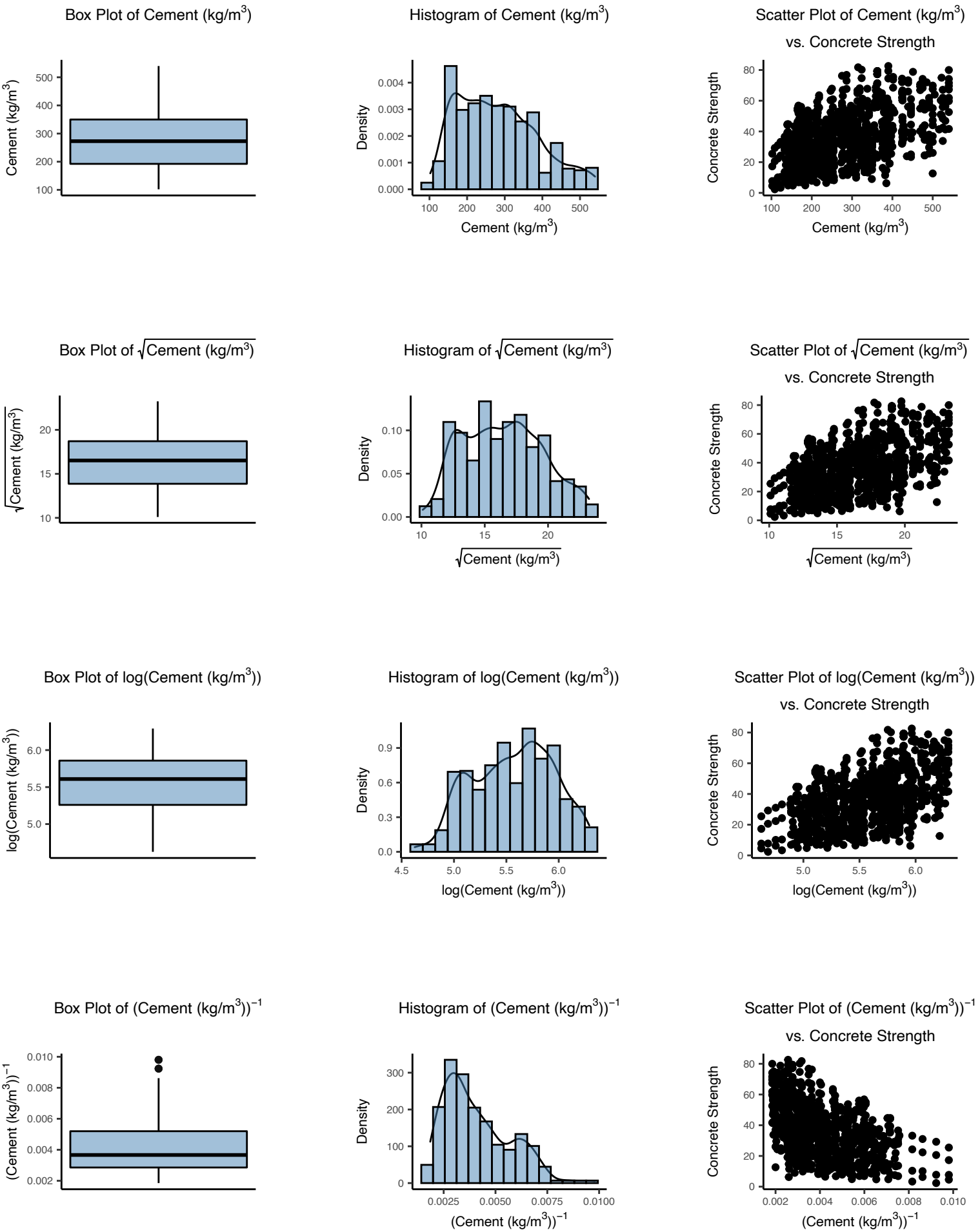
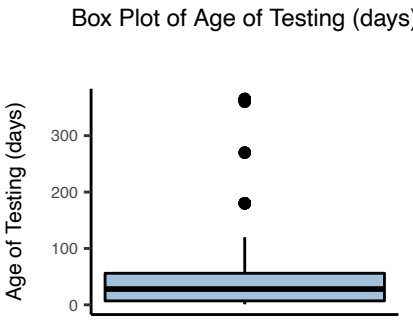
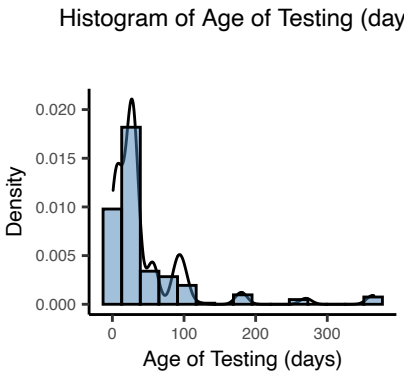
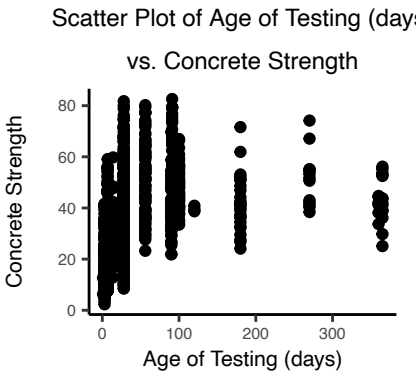
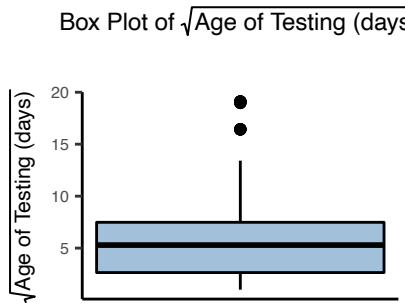


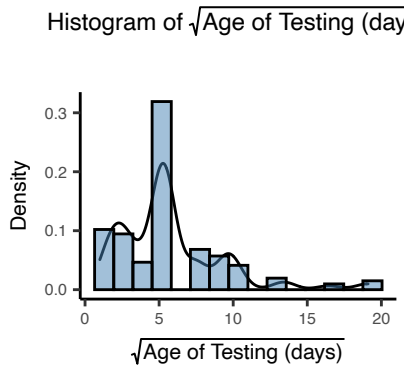
Figure 13: Age of Testing (days) Transformations

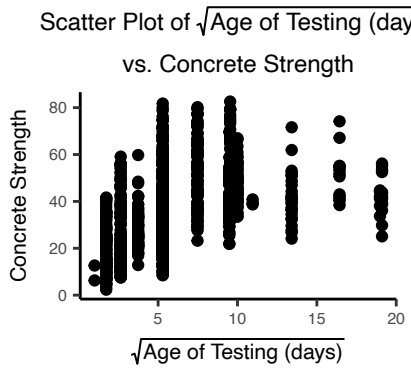


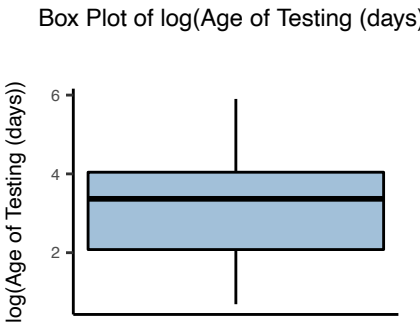


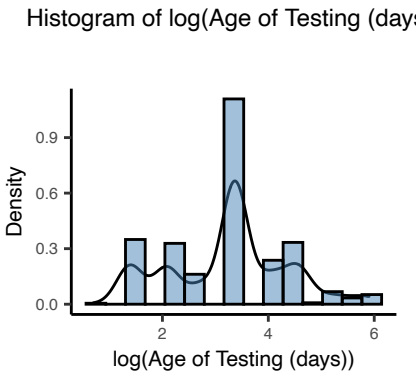


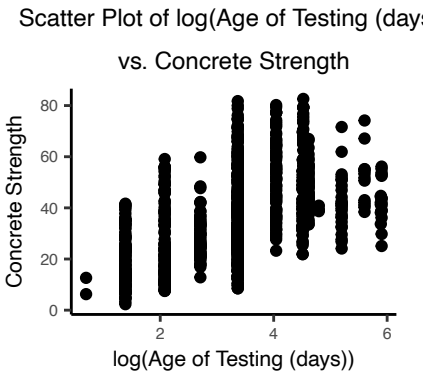


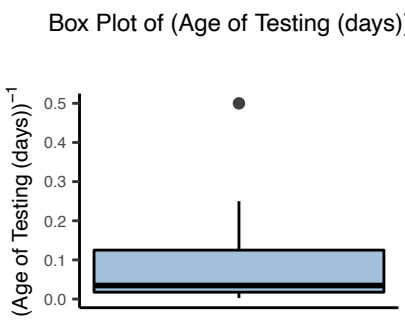


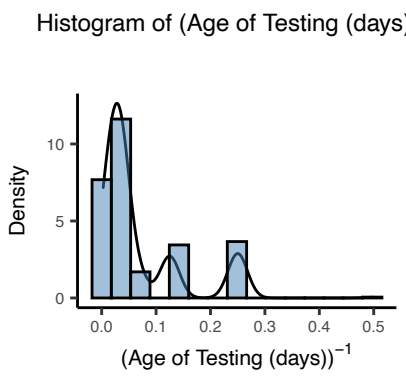












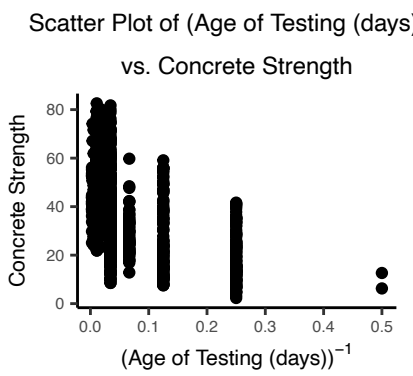
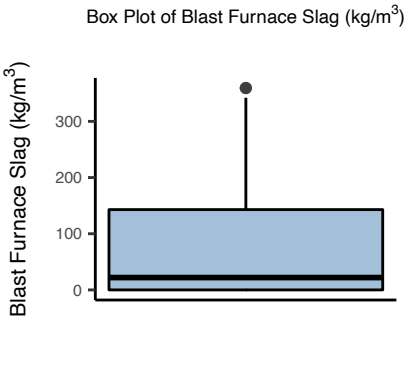
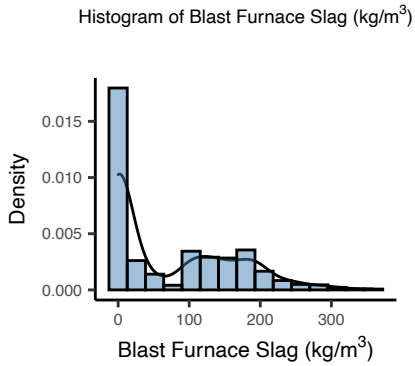
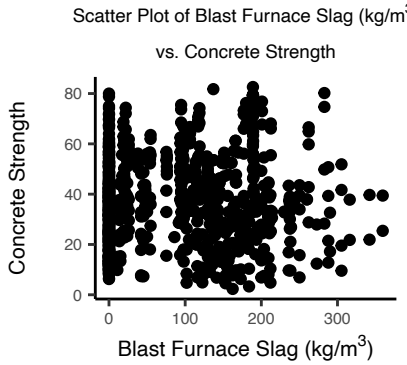
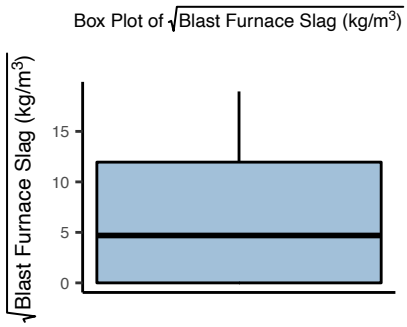


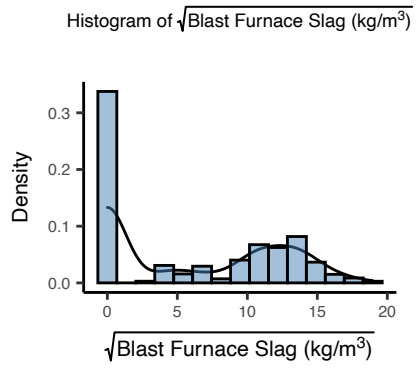
Figure 14: Blast Furnace Slag (kg/m^3) Transformations

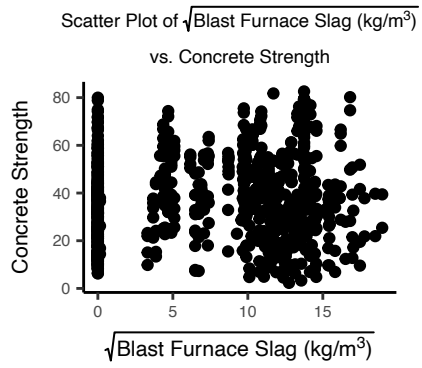


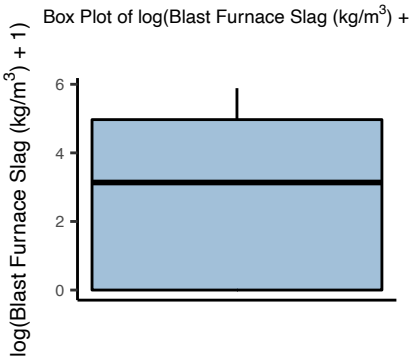


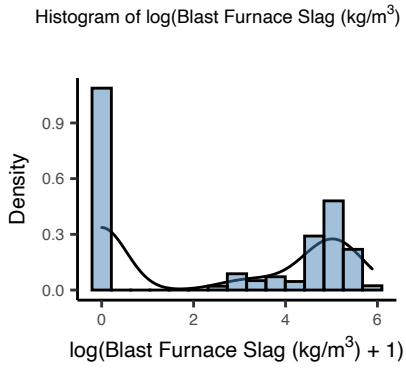


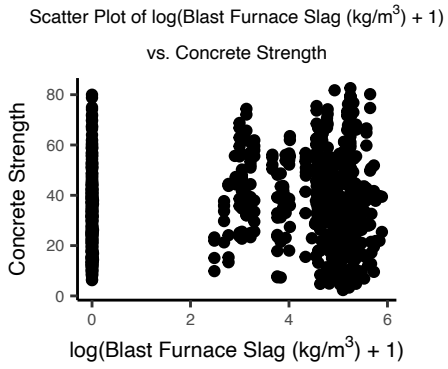


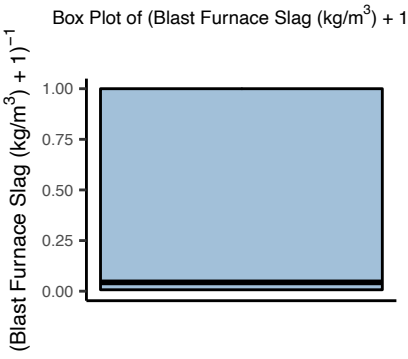


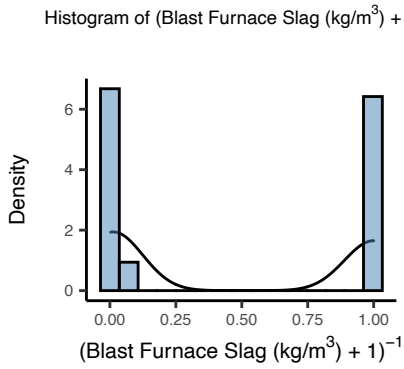












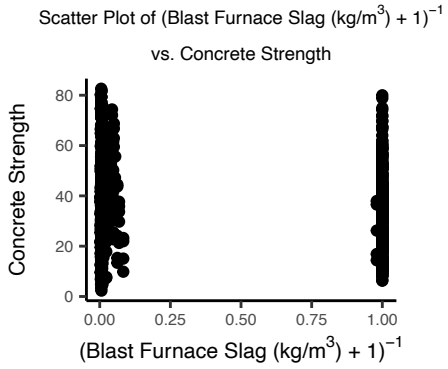


Figure 15: Superplasticizer (kg/m^3) Transformations

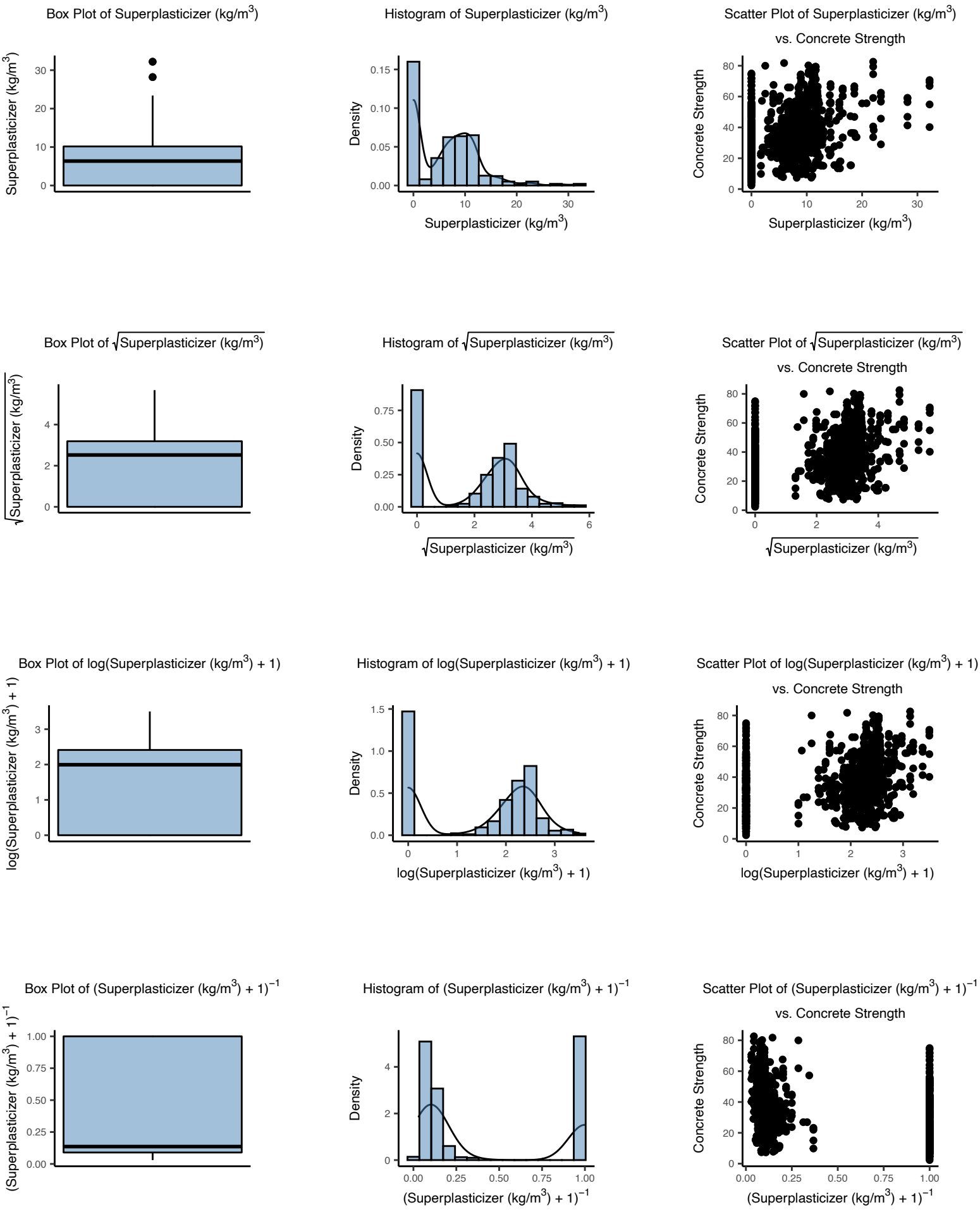


Figure 16: Fly Ash (kg/m^3) Transformations

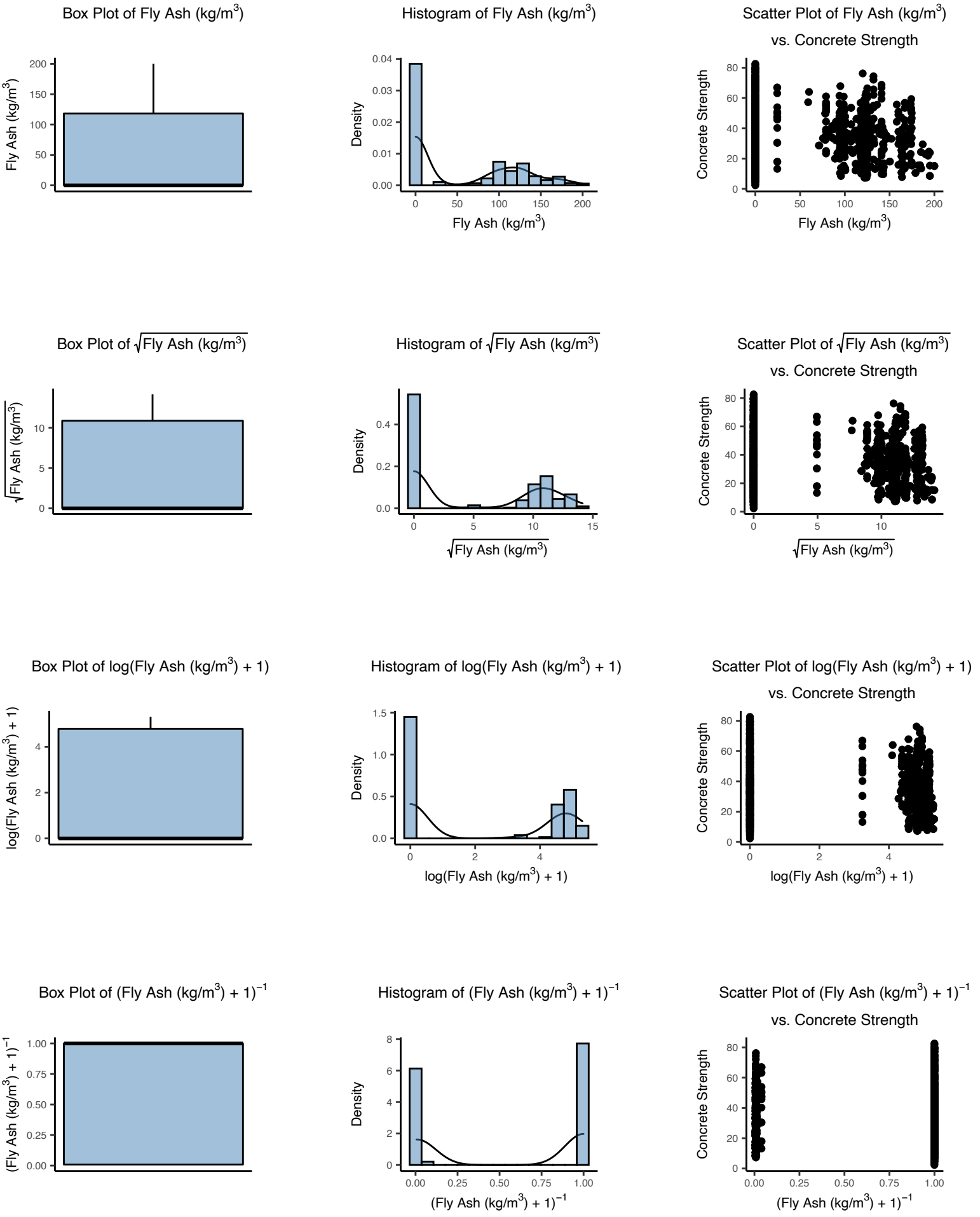


Figure 17: Box−Cox Graph for Full Initital Transformed Model

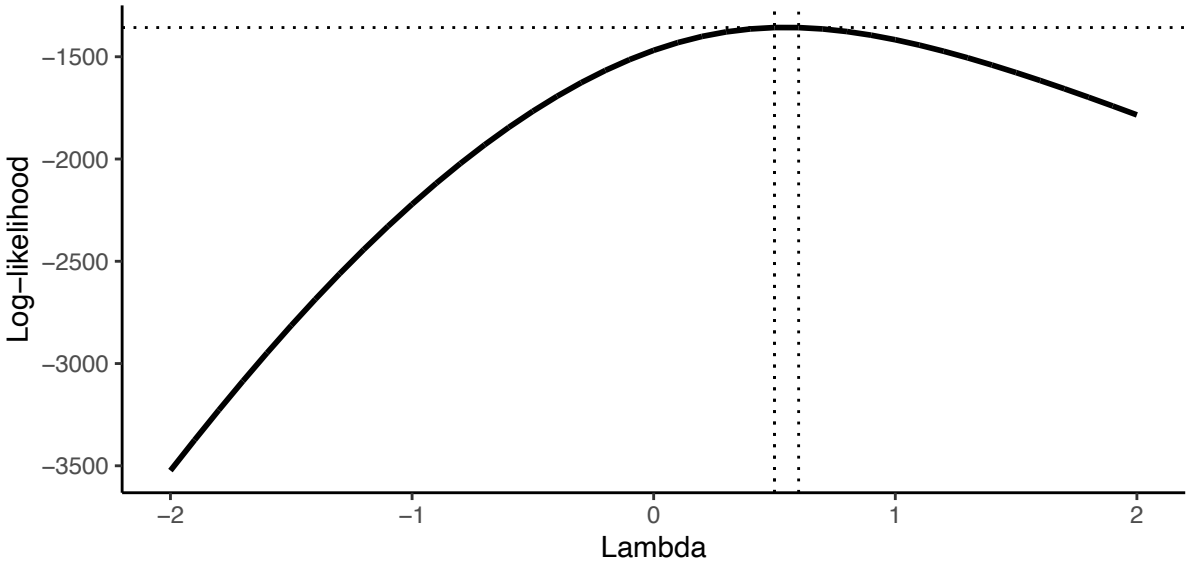


Figure 18: Diagnostic Plots For Full Transformed Model

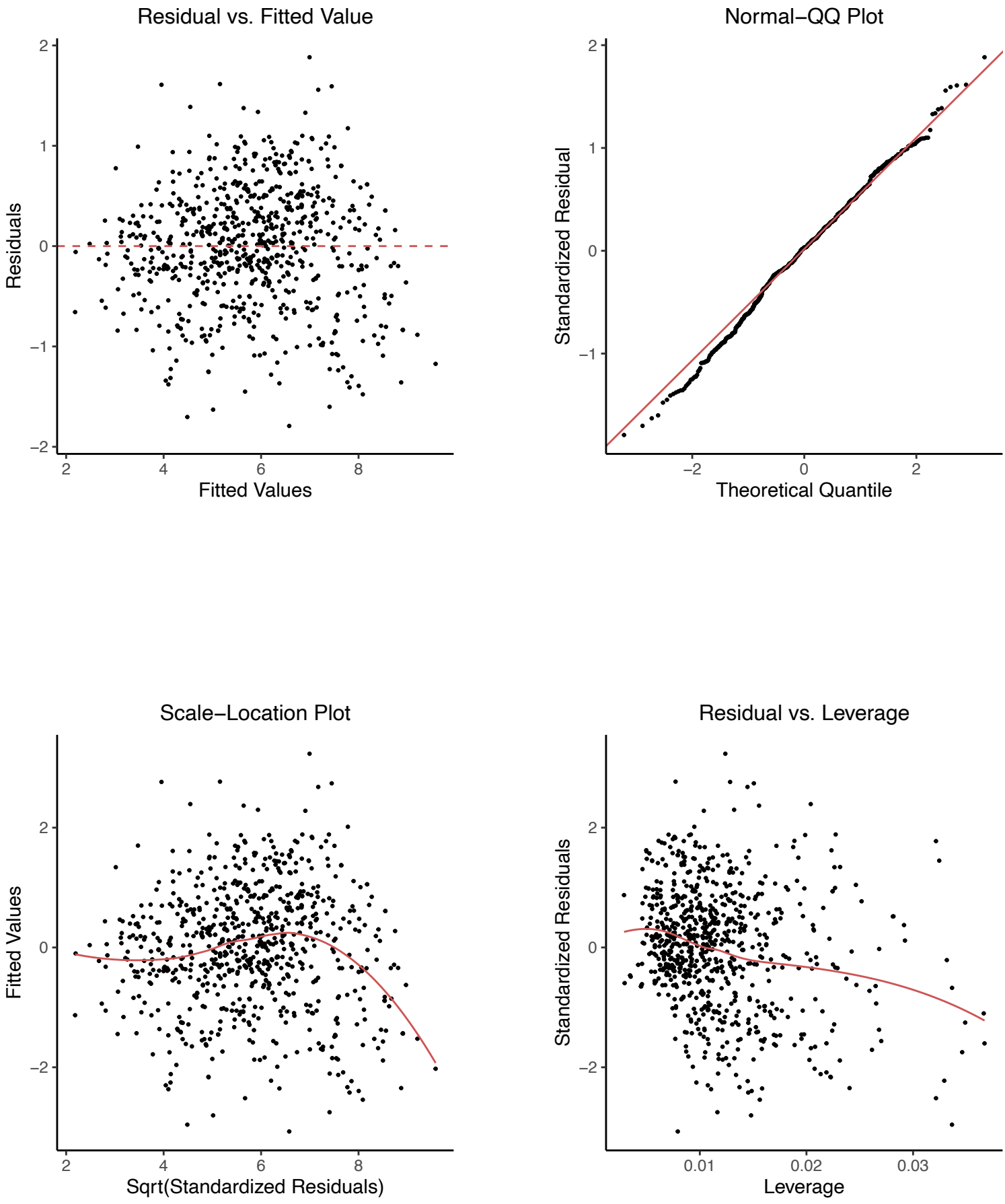
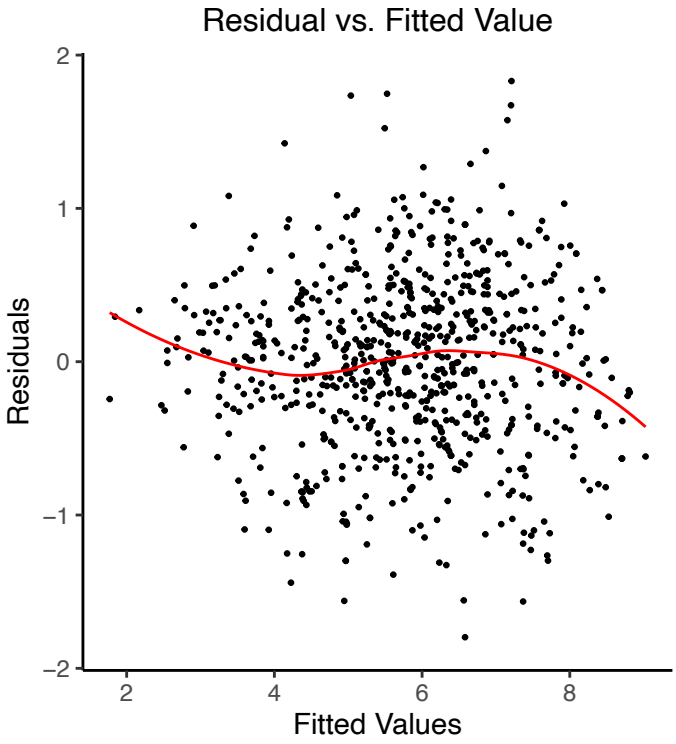
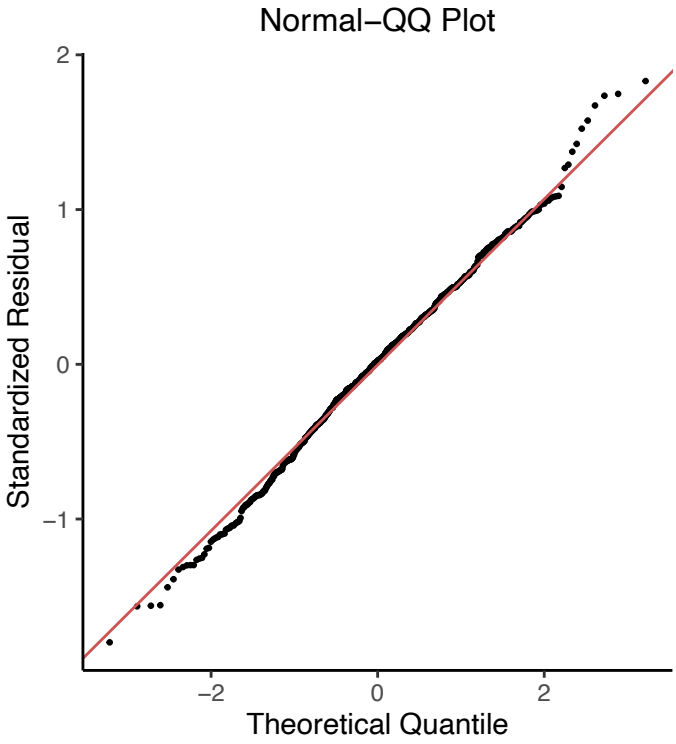
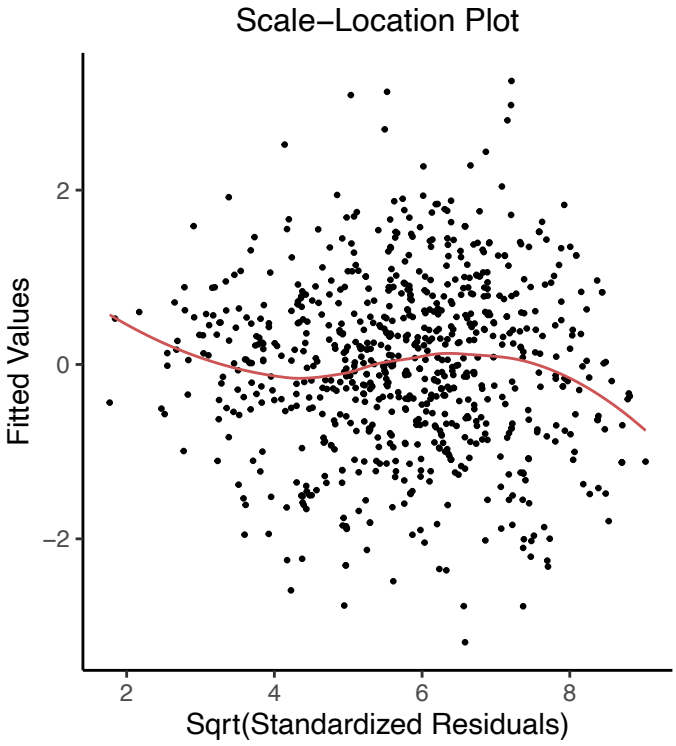


Figure 19: Diagnostic Plots For Model 1 (9 Variables)







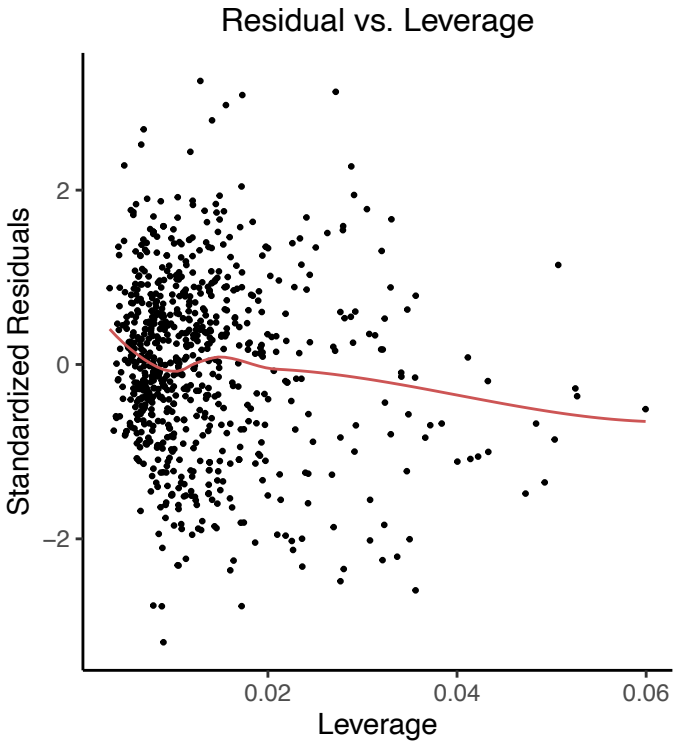


Figure 20: Diagnostic Plots For Model 2 (13 Variables)

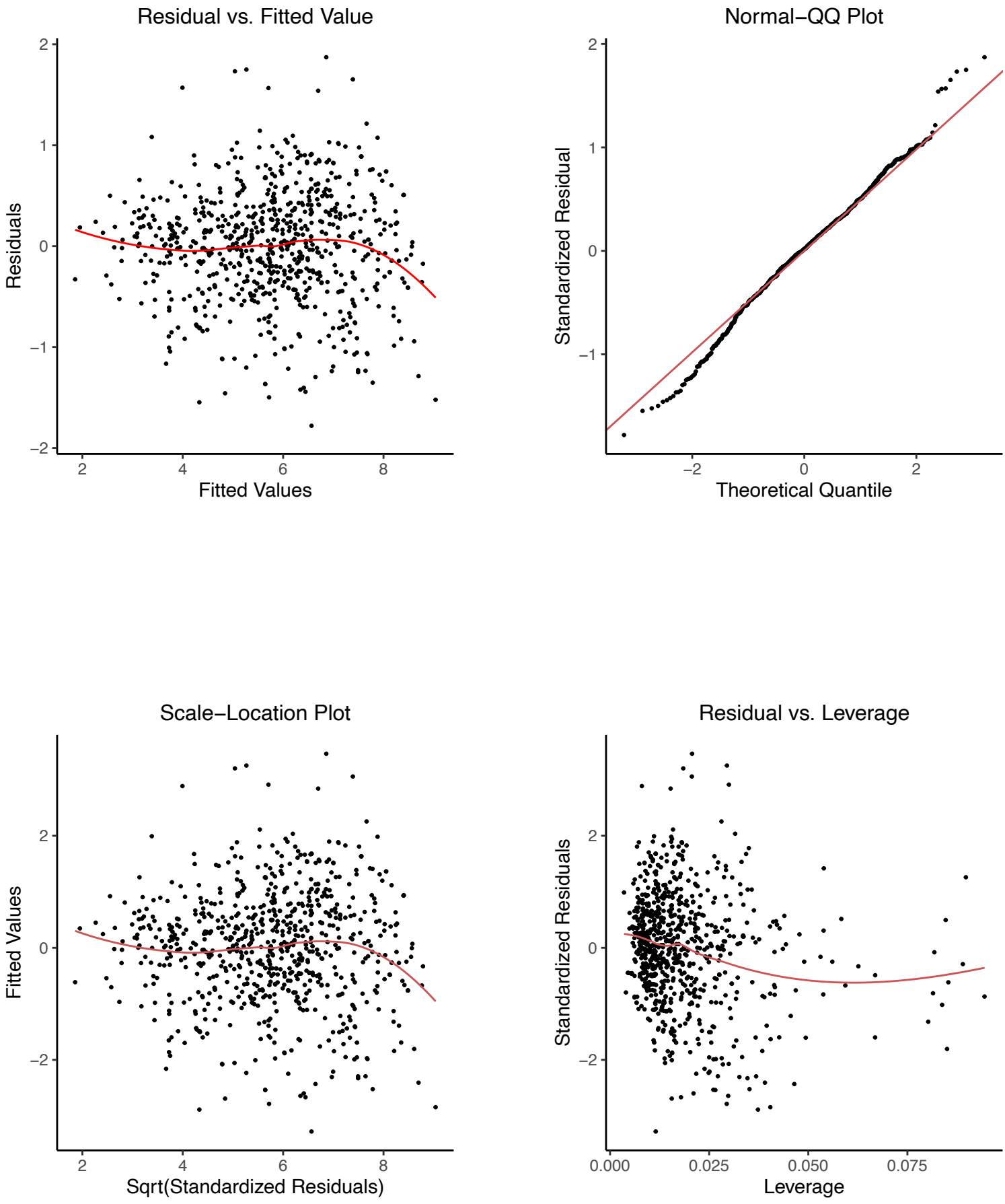
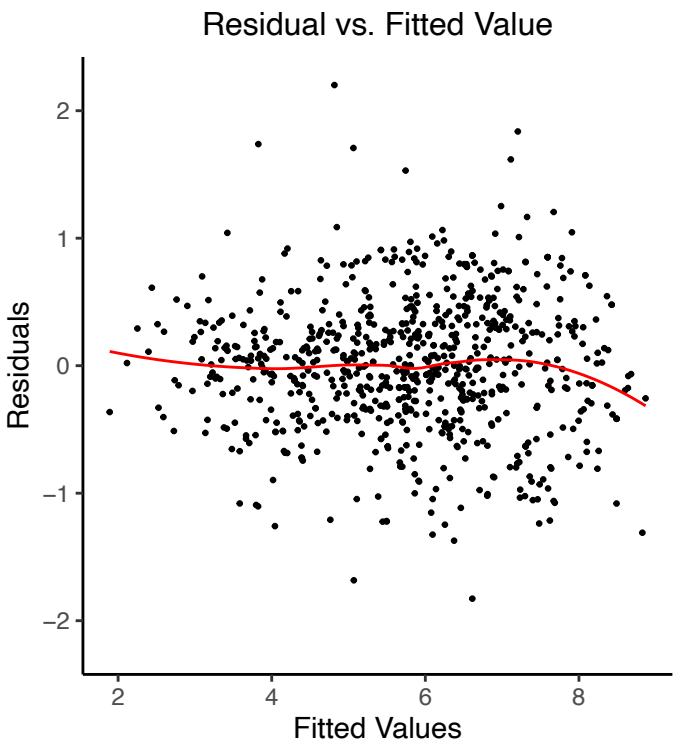
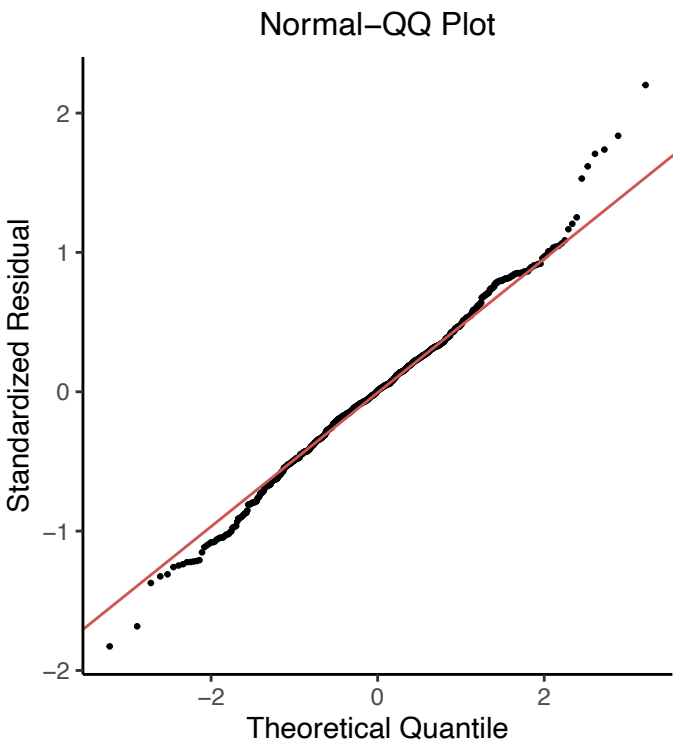
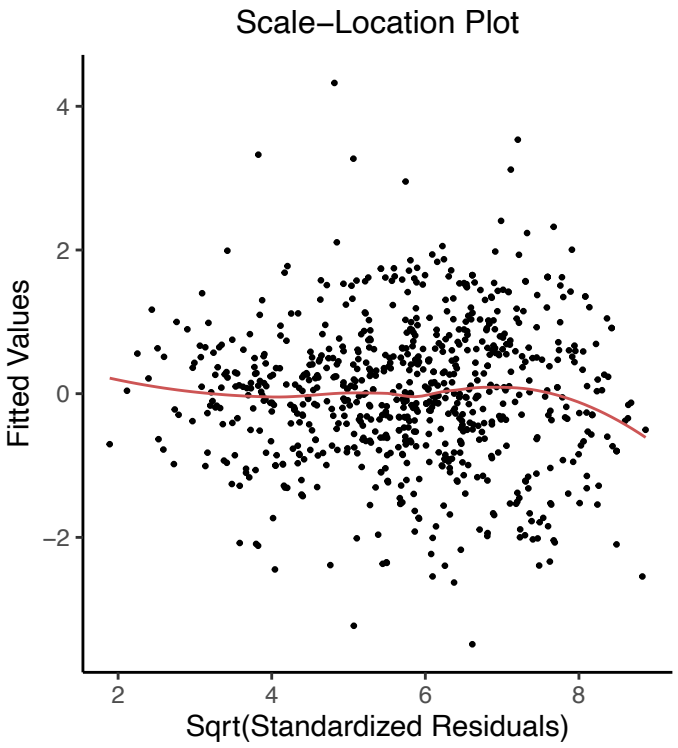


Figure 21: Diagnostic Plots For Model 3 (16 Variables)







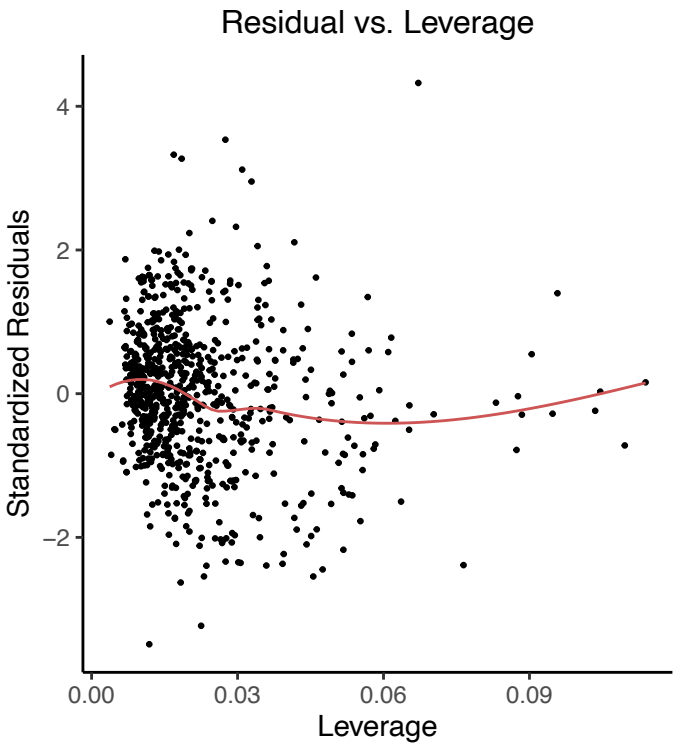


Figure 22: Diagnostic Plots For Model 4 (21 Variables)

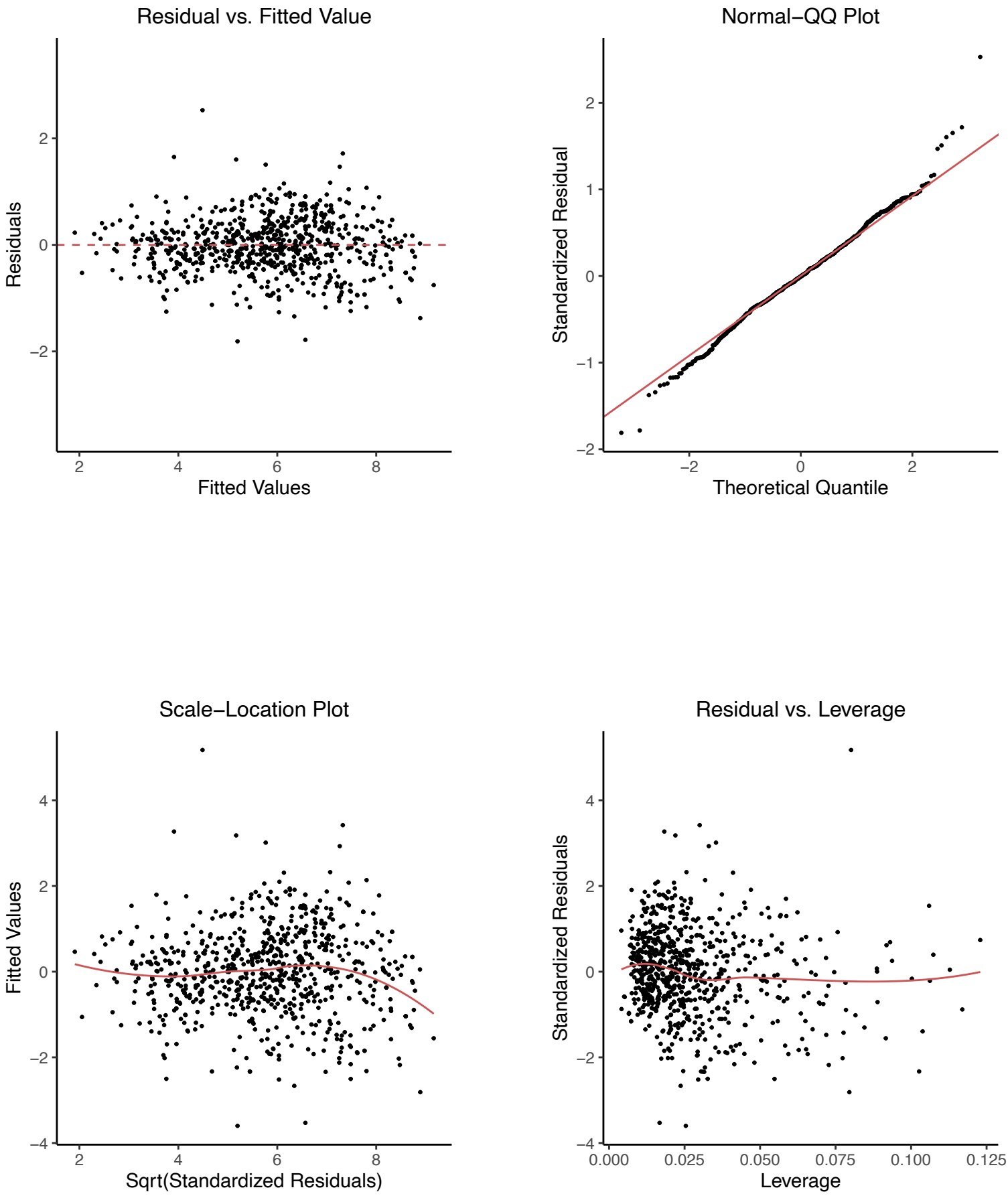
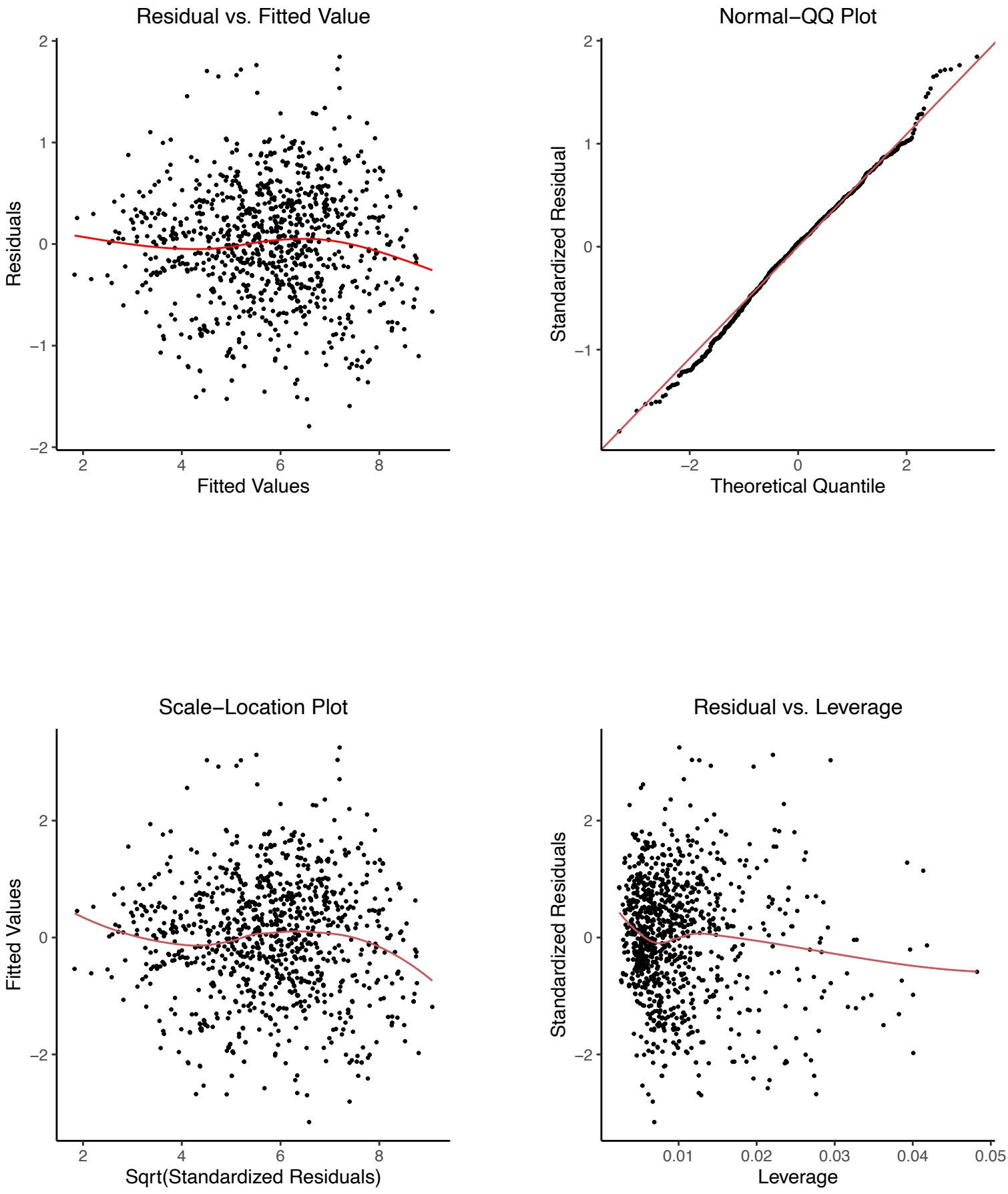


Figure 23: Diagnostic Plots For Model 1 Fit on All Data



Appendix 2 Code

STA 206: Final Project

John, Eric, and Ignat

12/2/2021

EDA and Data Visualizations

Step 1: Load in the data

```
# A. Step 1: Load in the data
#  1. Read the data in from the Excel file
concrete <- read_xls("Concrete_Data.xls")

#  2. Fix column names
names(concrete) <- c("cement", "blast_furnace_slag", "fly_ash", "water", "superplasticiz
er", "coarse_aggregate", "fine_aggregate", "age", "concrete_strength")

# B. Checks
#  1. Variable types
sapply(concrete, class)

#  2. Check for NA's
sapply(concrete, function(x) sum(is.na(x)))
```

Step 2: Summary Statistics

```
# A. Summary statistics
# stargazer(as.data.frame(concrete), header=FALSE, type='latex', add.lines = )

# B. Correlation matrix
concrete2 <- data.frame(concrete)
names(concrete2) <- c("Cement","Blast~Furnace~Slag","Fly~Ash","Water","Superplasticizer"
,

"Coarse~Aggregate","Fine~Aggregate", "Age~of~Testing", "Concrete~Strength")

corr_m <- ggpairs(concrete2, lower = list(continuous = wrap("points",size=0.01)), labell
er = label_parsed,
                                                                        title = "Figure
 1: Correlation Scatter Plot Matrix") + theme_classic(base_size = 8)  +
    theme(plot.title = element_text(hjust = 0.5, size = 16),
                             plot.margin = margin(4,0,4,0, "cm"))
# ggsave("plots_pg1.pdf", corr_m, width = 8.5, height = 11)

# C. Check for outliers within each variable defined as outside of three standard deviat
ions
#  1. Define function
outlier_func <- function(x) {
    return(x[(x > (mean(x) + 3 * sd(x))) | (x < (mean(x) - 3 * sd(x)))])
}

outlier_func2 <- function(x) {
    return((x > (mean(x) + 3 * sd(x))) | (x < (mean(x) - 3 * sd(x))))
}

#  2. Count
outliers_cnt <- data.frame(cnt_outliers = sapply(concrete, function(x) length(outlier_fu
nc(x))))

#  3. Review rows # check for overlap
nms <- names(concrete)
outlier_tbl <- concrete %>%
    mutate(across(all_of(nms),  list(flg = outlier_func2), .names = "{.col}.{.fn}")) %>%
    filter(if_any(ends_with("flg"), ~. >=1)) %>%
    melt(nms) %>%
    filter(value == TRUE) %>%
    dplyr::select(-value) %>%
    mutate(outlier_cat = paste0("outlier in ", str_extract(variable, ".*(?=\\.flg)")))
```

Step 3: Distibutions of Variables

```
# B. Histograms for all variables
#  1. Prep data
plot_dt <- gather(concrete)
plot_dt$key2 <- factor(plot_dt$key, labels = c("Age~of~Testing~(days)","Blast~Furnace~Sl
ag~(kg/m^3)","Cement~(kg/m^3)",

"Coarse~Aggregate~(kg/m^3)","Concrete~Strength~(MPa)","Fine~Aggregate~(kg/m^3)",

"Fly~Ash~(kg/m^3)","Superplasticizer~(kg/m^3)","Water~(kg/m^3)"))
#  2. Plot data
hist_plot <- ggplot(plot_dt, aes(value)) +
    geom_histogram(bins = 15, fill="steelblue", color="black") +
    labs(title = "Figure 2: Histograms of Untransformed Data") +
    facet_wrap(~key2, scales = 'free', labeller = label_parsed) +
    ylab("Count") +
    xlab("Value") +
    theme_classic() +
    theme(plot.title = element_text(hjust = 0.5, size = 16),
                            plot.margin = margin(1,0,1,0, "cm"))
# text=element_text(family = "Arial"),

# C. Box plots
box_plot <- ggplot(plot_dt, aes(value)) +
    geom_boxplot(fill="steelblue", color="black") +
    coord_flip() +
    facet_wrap(~key2, scales = 'free', labeller = label_parsed) +
    theme_classic() +
    labs(title = "Figure 3: Box Plots of Untransformed Data") +
    xlab("Value") +
    theme(
                            plot.title = element_text(hjust = 0.5, size = 16),
                            plot.margin = margin(1,0,1,0, "cm"))
# text=element_text(family = "Arial"),
# D. Combine the above two plots to one page
# ggsave("plots_pg2.pdf", ggarrange(hist_plot, box_plot, ncol=1, align = "hv") +
                                # theme(plot.margin = margin(0,2,0,2, "cm")), width = 8.
5, height = 11)
```

Step 4: Transformations

Part a: Full Model

```
# A. Train/Test 75/25 split
#  1. Split the data
set.seed(1215385763)
idx <- sample(1:nrow(concrete), nrow(concrete)*.75, replace = FALSE)
concrete_train <- concrete[idx, ]
concrete_test <- concrete[-idx, ]

# B. Compare the two data sets
#   i. Prep data
comp_tbl <- concrete %>%
    mutate(train_test_cat = ifelse(1:n() %in% idx, "Training Data", "Testing Data"))
plot_dt2 <- pivot_longer(comp_tbl, cols = names(comp_tbl)[names(comp_tbl) != "train_test
_cat"])
plot_dt2$name2 <- factor(plot_dt2$name, labels = c("Age~of~Testing~(days)","Blast~Furnac
e~Slag~(kg/m^3)","Cement~(kg/m^3)",

"Coarse~Aggregate~(kg/m^3)","Concrete~Strength~(MPa)","Fine~Aggregate~(kg/m^3)",

"Fly~Ash~(kg/m^3)","Superplasticizer~(kg/m^3)","Water~(kg/m^3)"))

#   ii. Make side by side boxplots
box_plot2 <- ggplot(plot_dt2, aes(value, train_test_cat)) +
    geom_boxplot(fill="steelblue", color="black") +
    coord_flip() +
    facet_wrap(~name2, scales = 'free', labeller = label_parsed) +
    theme_classic() +
    labs(title = "Figure 4: Comparison of Training and Test Data") +
    xlab("Value") +
    theme(axis.title.x = element_blank(),
                            plot.title = element_text(hjust = 0.5, size = 16),
                            plot.margin = margin(1,1,1,1, "cm"))
# ggsave("plots_pg3.pdf", ggarrange(box_plot2, ggally_blank(), nrow = 2), width = 8.5, h
eight = 11)


# C. Initial full model
#  1. Fit model
full.initial_model <- lm(concrete_strength~., data=concrete_train)
summary(full.initial_model)

#  2. Save plot
#   i. Given a gg_diagnose plot remove the hline and adds
fix_resid_plt <- function(x) {
    # remove the hline
    x[[2]][2] <- NULL

    # return with loess
    return(x+geom_smooth(color = "red", size = 0.5, se = FALSE))
}

#   ii. Plot
diag1 <- gg_diagnose(full.initial_model, theme = theme_classic(), plot.all = FALSE)
```

```
diag1_gg <- ggarrange(fix_resid_plt(diag1[10]$res_fitted) +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag1$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag1$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag1$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)
# ggsave("plots_pg4.pdf", annotate_figure(diag1_gg, top = text_grob(bquote("Figure 5: Di
agnostic Plots For Full Untransformed Model"))), width = 8.5, height = 11)

# C. Perform Box-Cox procedure to check if response needs to transformed
#  1. Perform the procedure
bc <- boxcox(full.initial_model)
(lambda <- bc$x[which.max(bc$y)])
# While lambda close 1 with square root being the closest potential transformation at 0.
5

#  2. Box Cox Graph
boxcox <- gg_boxcox(full.initial_model, scale.factor = 1, showlambda = F) +
    theme_classic() +
    labs(title = "Figure 6: Box-Cox Graph for Full Initital Model") +
    theme(plot.title = element_text(hjust = 0.5, size = 12),
                                plot.margin = margin(3,3,3,3, "cm"))

#  3. Side-by-Side histograms of the response variable
cs_plot <- ggplot(concrete, aes(concrete_strength)) +
        geom_density() +
        geom_histogram(bins = 15, fill = "steelblue", color = "black", alpha = 0.5) +
        labs(title = TeX("Figure 7: Histogram of Concrete Strength (MPa)")) +
        ylab("Count") +
        xlab(TeX("Concrete Strength (MPa)")) +
        theme_classic() +
        theme(plot.title = element_text(hjust = 0.5, size = 11),
                                plot.subtitle = element_text(hjust = 0.5, size = 11),
```

```
                                text = element_text(size=10),
                                plot.margin = margin(2,1,2,1, "cm"))

cs_plot_sqrt <- ggplot(concrete, aes(sqrt(concrete_strength))) +
        geom_density() +
        geom_histogram(bins = 15, fill = "steelblue", color = "black", alpha = 0.5) +
        labs(title = TeX("Figure 8: Histogram of \\sqrt{Concrete Strength (MPa)}")) +
        ylab("Count") +
        xlab(TeX("\\sqrt{Concrete Strength (MPa)}")) +
        theme_classic() +
        theme(plot.title = element_text(hjust = 0.5, size = 11),
                                plot.subtitle = element_text(hjust = 0.5, size = 11),
                                text = element_text(size=10),
                                plot.margin = margin(2,1,2,1, "cm"))

cs_cmb <- ggarrange(boxcox, ggarrange(cs_plot, cs_plot_sqrt, nrow = 1, align = "hv"), nr
ow = 2)
# ggsave("plots_pg5.pdf", cs_cmb, width = 8.5, height = 11)

# E. Run additional models and save diagnostic plots
#  1. Forward no interaction
#   i. Fit model
full.model <- lm(concrete_strength~., concrete_train)
null.model <- lm(sqrt(concrete_strength)~1, concrete_train)
n <- nrow(concrete_train)
step.f.sqrt <- stepAIC(null.model, scope=list(upper=full.model, lower=~1), trace=F, dire
ction="forward", k=log(n))
step.f.sqrt.m <- lm(sqrt(concrete_strength) ~ cement + age + superplasticizer + blast_fu
rnace_slag +
    water + fly_ash, concrete_train)

#   ii. Diagnostics plots
diag2 <- gg_diagnose(step.f.sqrt.m, theme = theme_classic(), plot.all = FALSE)
diag2_gg <- ggarrange(fix_resid_plt(diag2$res_fitted) +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2$resleverage +
```

```
theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)

# ggsave("plots_pg6.pdf", annotate_figure(diag2_gg, top = text_grob(bquote("Figure 9: Di
agnostic Plots For Forward Stepwise Model Without Interaction Effects"))), width = 8.5,
 height = 11)

#  2. Forward with interaction
#   i. Fit model
full.model <- lm(sqrt(concrete_strength)~.^2, concrete_train)
null.model <- lm(sqrt(concrete_strength)~1, concrete_train)
n <- nrow(concrete_train)
step.f.sqrt <- stepAIC(null.model, scope=list(upper=full.model, lower=~1), trace=F, dire
ction="forward", k=log(n))
step.f.sqrt.m <- lm(sqrt(concrete_strength) ~ cement + age + superplasticizer + blast_fu
rnace_slag +
    fly_ash + water + age*superplasticizer + age*fly_ash + superplasticizer*water +
    cement*age, concrete_train)

#   ii. Diagnostics plots
diag2 <- gg_diagnose(step.f.sqrt.m, theme = theme_classic(), plot.all = FALSE)
diag2_gg <- ggarrange(fix_resid_plt(diag2$res_fitted) +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)

ggsave("plots_pg7.pdf", annotate_figure(diag2_gg, top = text_grob(bquote("Figure 10: Dia
```

```
gnostic Plots For Forward Stepwise Model With First-Order Interactions"))), width = 8.5,
height = 11)

#  3. Forward with second degree polynomial no interaction
#   i. Fit model
full.model <- lm(concrete_strength~cement+I(cement^2)+blast_furnace_slag+I(blast_furnace
_slag^2)+fly_ash+I(fly_ash^2)+water+I(water^2) + superplasticizer + I(superplasticizer^2
) + coarse_aggregate + I(coarse_aggregate^2) + fine_aggregate + I(fine_aggregate^2) + ag
e + I(age^2), concrete_train)
null.model <- lm(sqrt(concrete_strength)~1, concrete_train)
n <- nrow(concrete_train)
step.f.sqrt.poly <- stepAIC(null.model, scope=list(upper=full.model, lower=~1), trace=F,
direction="forward", k=log(n))
step.f.sqrt.poly.m <- lm(sqrt(concrete_strength) ~ cement + age + I(age^2) + superplasti
cizer +
    blast_furnace_slag + I(superplasticizer^2) + water + fly_ash +
    I(fly_ash^2) + I(cement^2) + I(blast_furnace_slag^2), concrete_train)

#   ii. Diagnostics plots
diag3 <- gg_diagnose(step.f.sqrt.poly.m, theme = theme_classic(), plot.all = FALSE)
diag3_gg <- ggarrange(fix_resid_plt(diag3$res_fitted) +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag3$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag3$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag3$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)

ggsave("plots_pg8.pdf", annotate_figure(diag3_gg, top = text_grob(bquote("Figure 11: Dia
gnostic Plots For Forward Stepwise Second-Order Polynomial Model Without Interactions"
))), width = 8.5, height = 11)
```

Part b: Potential Predictor Transformations

```
# A. Univariate series of plots for each predictor - 3 charts: box plot + histogram (inc
l density overlay) + scatter vs strength
plot_fn <- function(x, lab, title_size=9) {
    # first draw the boxplot
    p1 <- ggplot(concrete, aes(x)) +
        geom_boxplot(color = "black", fill = "steelblue", alpha = 0.5) +
        coord_flip() +
        xlab(TeX(lab)) +
        labs(title = TeX(paste("Box Plot of ", lab))) +
        theme_classic() +
        theme(axis.text.x.bottom = element_blank(),
                                axis.ticks.x = element_blank(),
                                text = element_text(size=8),
                                plot.title = element_text(hjust = 0.5, size = title_siz
e),
                                plot.subtitle = element_text(hjust = 0.5, size = title_s
ize),
                                plot.margin = margin(1,1,1,1, "cm"))

    # then the histogram with the overlay
    p2 <- ggplot(concrete, aes(x)) +
        geom_density() +
        geom_histogram(aes(y = ..density..), bins = 15, fill = "steelblue", color = "bla
ck", alpha = 0.5) +
        labs(title = TeX(paste("Histogram of ", lab))) +
        ylab("Density") +
        xlab(TeX(lab)) +
        theme_classic() +
        theme(plot.title = element_text(hjust = 0.5, size = title_size),
                                plot.subtitle = element_text(hjust = 0.5, size = title_s
ize),
                                text = element_text(size=8),
                                plot.margin = margin(1,0,1,0, "cm"))

    # scatter last
    p3 <- ggplot(concrete, aes(x=x, y=concrete_strength)) +
        geom_point() +
        labs(title = TeX(paste("Scatter Plot of ", lab)),
                                subtitle = "vs. Concrete Strength") +
        ylab("Concrete Strength") +
        xlab(TeX(lab)) +
        theme_classic() +
        theme(plot.title = element_text(hjust = 0.5, size = title_size),
                                plot.subtitle = element_text(hjust = 0.5, size = title_s
ize),
                                text = element_text(size=8),
                                plot.margin = margin(1,1,1,1, "cm"))

    # return plots
    ggarrange(p1, p2, p3, nrow = 1, align = "hv")
}
```

```
# B. Tranformed Predictors
#  1. Cement
plots_cement <- ggarrange(plot_fn(concrete$cement, "Cement (kg/m^3)"),
                                        plot_fn(sqrt(concrete$cement), "\\sqrt{Cement (k
g/m^3)}"),
                                        plot_fn(log(concrete$cement), "log(Cement (kg/m^
3))"),
                                        plot_fn(1/(concrete$cement), "(Cement (kg/m^3))^
{-1}"), nrow = 4)
# ggsave("plots_pg9.pdf", annotate_figure(plots_cement, top = text_grob(bquote("Figure 1
2: Cement "*(kg/m^3)*" Transformations"))), width = 8.5, height = 11)

#  2. Age
plots_age <- ggarrange(plot_fn(concrete$age, "Age of Testing (days)"),
                                        plot_fn(sqrt(concrete$age), "\\sqrt{Age of Testi
ng (days)}"),
                                        plot_fn(log(concrete$age + 1), "log(Age of Testi
ng (days))"),
                                        plot_fn(1/(concrete$age + 1), "(Age of Testing
 (days))^{-1}"), nrow = 4)
# ggsave("plots_pg10.pdf", annotate_figure(plots_age, top = text_grob(bquote("Figure 13:
Age of Testing (days) Transformations"))), width = 8.5, height = 11)


#  3. Blast~Furnace~Slag~(kg/m^3)
plots_slag <- ggarrange(plot_fn(concrete$blast_furnace_slag, "Blast Furnace Slag (kg/m^
3)", 7),
                                        plot_fn(sqrt(concrete$blast_furnace_slag), "\\sq
rt{Blast Furnace Slag (kg/m^3)}", 7),
                                        plot_fn(log(concrete$blast_furnace_slag + 1), "l
og(Blast Furnace Slag (kg/m^3) + 1)", 7),
                                        plot_fn(1/(concrete$blast_furnace_slag + 1), "(B
last Furnace Slag (kg/m^3) + 1)^{-1}", 7), nrow = 4)
# ggsave("plots_pg11.pdf", annotate_figure(plots_slag, top = text_grob(bquote("Figure 1
4: Blast Furnace Slag "*(kg/m^3)*" Transformations"))), width = 8.5, height = 11)



#  4. Superplasticizer (kg/m^3)
plots_super <- ggarrange(plot_fn(concrete$superplasticizer, "Superplasticizer (kg/m^3)",
8),
                                        plot_fn(sqrt(concrete$superplasticizer), "\\sqrt
{Superplasticizer (kg/m^3)}",8),
                                        plot_fn(log(concrete$superplasticizer + 1), "log
(Superplasticizer (kg/m^3) + 1)",8),
                                        plot_fn(1/(concrete$superplasticizer + 1), "(Sup
erplasticizer (kg/m^3) + 1)^{-1}",8), nrow = 4)
# ggsave("plots_pg12.pdf", annotate_figure(plots_super, top = text_grob(bquote("Figure 1
5: Superplasticizer "*(kg/m^3)*" Transformations"))), width = 8.5, height = 11)


#  5. Fly Ash
```

```
plots_fly <- ggarrange(plot_fn(concrete$fly_ash, "Fly Ash (kg/m^3)"),
                                        plot_fn(sqrt(concrete$fly_ash), "\\sqrt{Fly Ash
 (kg/m^3)}"),
                                        plot_fn(log(concrete$fly_ash + 1), "log(Fly Ash
 (kg/m^3) + 1)"),
                                        plot_fn(1/(concrete$fly_ash + 1), "(Fly Ash (kg/
m^3) + 1)^{-1}"), nrow = 4)
# ggsave("plots_pg13.pdf", annotate_figure(plots_fly, top = text_grob(bquote("Figure 16:
Fly Ash "*(kg/m^3)*" Transformations"))), width = 8.5, height = 11)

# C. Fit regression model on all predictors with transformations
full.transformed_model <- lm(concrete_strength ~ sqrt(cement) + sqrt(blast_furnace_slag)
+ fly_ash + water + sqrt(superplasticizer) + coarse_aggregate + fine_aggregate + log(ag
e), concrete_train)

# D. Review
#  1. Diagnostic plots
diag2 <- gg_diagnose(full.transformed_model, theme = theme_classic(), plot.all = FALSE)
diag2_gg <- ggarrange(diag2[10]$res_fitted +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2[11]$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2[12]$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)
# ggsave("plots_pg14.pdf", annotate_figure(diag2_gg, top = text_grob(bquote("Figure 17:
 Diagnostic Plots For Full Transformed Model"))), width = 8.5, height = 11)

# E. Perform Box-Cox procedure to check if response needs to transformed
#  1. Perform the procedure
bc <- boxcox(full.transformed_model)
(lambda <- bc$x[which.max(bc$y)])
# While lambda close 1 with square root being the closest potential transformation at 0.
5
```

```
#  2. Box Cox Graph
boxcox <- gg_boxcox(full.transformed_model, scale.factor = 1, showlambda = F) +
    theme_classic() +
    labs(title = "Figure 17: Box-Cox Graph for Full Initital Transformed Model") +
    theme(plot.title = element_text(hjust = 0.5, size = 12),
                                plot.margin = margin(3,3,3,3, "cm"))
# ggsave("plots_pg14.pdf", ggarrange(boxcox, ggally_blank(), nrow = 2), width = 8.5, hei
ght = 11)

# F. Re-fit full transformed model with Y transformed
#  1. Fit
full.transformed_model <- lm(sqrt(concrete_strength) ~ sqrt(cement) + sqrt(blast_furnace
_slag) + fly_ash + water + sqrt(superplasticizer) + coarse_aggregate + fine_aggregate +
 log(age), concrete_train)

summary(full.transformed_model)
# stargazer(full.transformed_model, intercept.bottom = FALSE, single.row = TRUE)

#  2. Diagnostic plots
diag2 <- gg_diagnose(full.transformed_model, theme = theme_classic(), plot.all = FALSE)
diag2_gg <- ggarrange(diag2[10]$res_fitted +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2[11]$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2[12]$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag2$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)
# ggsave("plots_pg15.pdf", annotate_figure(diag2_gg, top = text_grob(bquote("Figure 18:
 Diagnostic Plots For Full Transformed Model"))), width = 8.5, height = 11)
```

Part 5: Model Selection

```
# A. Apply predictor transformations
#  1. Training data
concrete_train <- concrete_train %>%
    mutate(concrete_strength=sqrt(concrete_strength),
                                age=log(age),
                                blast_furnace_slag=sqrt(blast_furnace_slag),
                                superplasticizer=sqrt(superplasticizer),
                                cement=sqrt(cement))

#  2. Testing data
concrete_test <- concrete_test %>%
    mutate(concrete_strength=sqrt(concrete_strength),
                                age=log(age),
                                blast_furnace_slag=sqrt(blast_furnace_slag),
                                superplasticizer=sqrt(superplasticizer),
                                cement=sqrt(cement))

# B. Perform stepwise model selection
#  1. Define full (all variables + all two way interactions) and null models
null.model <- lm(concrete_strength~1, concrete_train)
full.model <- lm(concrete_strength~.^2, concrete_train)

#  2. Perform procedure with BIC (change header!!!)
n <- nrow(concrete_train)
step.f <- stepAIC(null.model, scope=list(upper=full.model, lower=~1), trace=F, direction
="both", k=log(n))
step.f_smry <- step.f$anova

#  3. Review steps
step.f_smry2 <- step.f_smry %>%
    mutate(prc_chg = (AIC - lag(AIC))/abs(lag(AIC)) * 100)
# stargazer(step.f_smry2, summary = FALSE)

#  4. Plot percent change # add annotations for models 1 through 4
# prc_chng <- tibble(p=1:22, amount=step.f_smry2$prc_chg)
# prc_chng_plot <- ggplot(prc_chng, aes(x=p, y=amount/100)) +
#   geom_line() +
#   scale_y_continuous(labels = scales::percent) +
#   xlab("Number of Predictors (p)") +
#   ylab("BIC Percent Change (%)") +
#   labs(title = "Figure 14: BIC Percent Change Over Number of Parameters") +
#   theme_classic() +
#   theme(plot.title = element_text(hjust = 0.5, size = 12),
#                           plot.margin = margin(1,1,1,1, "cm"))
# ggsave("plots_pg9.pdf", ggarrange(prc_chng_plot, ggally_blank(), nrow = 2), width = 8.
5, height = 11)
```

Step 6: Validation

```
# A. Prepare validation set
newdata <- concrete_test[,-9]

# B. Fit potential models on training data
#  1. Model with 9 predictors
#   i. Fit model
model.10 <- lm(concrete_strength~age + cement + superplasticizer + blast_furnace_slag +
 water + fly_ash + cement*age + water*superplasticizer + blast_furnace_slag*water, concr
ete_train)
# stargazer(model.10, intercept.bottom = FALSE, single.row = TRUE)

#   ii. Diagnostic Plots
diag_m10 <- gg_diagnose(model.10, theme = theme_classic(), plot.all = FALSE)
diag_m10_gg <- ggarrange(fix_resid_plt(diag_m10$res_fitted) +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m10$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m10$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m10$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)
# ggsave("plots_pg16.pdf", annotate_figure(diag_m10_gg, top = text_grob(bquote("Figure 1
9: Diagnostic Plots For Model 1 (9 Variables)"))), width = 8.5, height = 11)


#  2. Model with 13 predictors
#   i. Fit model
model.14 <- lm(concrete_strength~age + cement + superplasticizer + blast_furnace_slag +
 water + fly_ash + cement*age + water*superplasticizer + blast_furnace_slag*water + fly_
ash*superplasticizer + cement*blast_furnace_slag + coarse_aggregate + superplasticizer*c
oarse_aggregate, concrete_train)
# stargazer(model.14, intercept.bottom = FALSE, single.row = TRUE)

#   ii. Diagnostic Plots
diag_m14 <- gg_diagnose(model.14, theme = theme_classic(), plot.all = FALSE)
```

```
diag_m14_gg <- ggarrange(fix_resid_plt(diag_m14$res_fitted) +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m14$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m14$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m14$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)
# ggsave("plots_pg17.pdf", annotate_figure(diag_m14_gg, top = text_grob(bquote("Figure 2
0: Diagnostic Plots For Model 2 (13 Variables)"))), width = 8.5, height = 11)


#  3. Model with 16 predictors
#   i. Fit model
model.17 <- lm(concrete_strength~age + cement + superplasticizer + blast_furnace_slag +
 water + fly_ash + cement*age + water*superplasticizer + blast_furnace_slag*water + fly_
ash*superplasticizer + cement*blast_furnace_slag + coarse_aggregate + superplasticizer*c
oarse_aggregate + blast_furnace_slag*coarse_aggregate + fine_aggregate + fly_ash*fine_ag
gregate, concrete_train)
# stargazer(model.17, intercept.bottom = FALSE, single.row = TRUE)

#   ii. Diagnostic Plots
diag_m17 <- gg_diagnose(model.17, theme = theme_classic(), plot.all = FALSE)
diag_m17_gg <- ggarrange(fix_resid_plt(diag_m17$res_fitted) +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m17$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),
```

```
 diag_m17$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m17$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)
# ggsave("plots_pg18.pdf", annotate_figure(diag_m17_gg, top = text_grob(bquote("Figure 2
1: Diagnostic Plots For Model 3 (16 Variables)"))), width = 8.5, height = 11)

#  4. Model with 21 predictors
#   i. Fit model
model.22 <- lm(concrete_strength~age + cement + superplasticizer + blast_furnace_slag +
 water + fly_ash + cement*age + water*superplasticizer + blast_furnace_slag*water + fly_
ash*superplasticizer + cement*blast_furnace_slag + coarse_aggregate + superplasticizer*c
oarse_aggregate + blast_furnace_slag*coarse_aggregate + fine_aggregate + fly_ash*fine_ag
gregate + coarse_aggregate*fine_aggregate + water*age + blast_furnace_slag*water + cemen
t*water + blast_furnace_slag*age, concrete_train)
# stargazer(model.22, intercept.bottom = FALSE, single.row = TRUE)


#   ii. Diagnostic Plots
diag_m22 <- gg_diagnose(model.22, theme = theme_classic(), plot.all = FALSE)
diag_m22_gg <- ggarrange(diag_m22$res_fitted +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m22$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m22$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m22$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),
```

```
 ncol =2, nrow = 2)
# ggsave("plots_pg19.pdf", annotate_figure(diag_m22_gg, top = text_grob(bquote("Figure 2
2: Diagnostic Plots For Model 4 (21 Variables)"))), width = 8.5, height = 11)

# C. Calculate Y_hat using potential models on the testing data set
y.hat.10 <- predict(model.10, newdata)
y.hat.14 <- predict(model.14, newdata)
y.hat.17 <- predict(model.17, newdata)
y.hat.22 <- predict(model.22, newdata)


# D. Calculate MSPE for each of the potential models
mspe.10 <- mean((concrete_test$concrete_strength-y.hat.10)^2)
mspe.14 <- mean((concrete_test$concrete_strength-y.hat.14)^2)
mspe.17 <- mean((concrete_test$concrete_strength-y.hat.17)^2)
mspe.22 <- mean((concrete_test$concrete_strength-y.hat.22)^2)



# E. Calculate SSE from train
sse_t.10 <- sum(model.10$residuals^2)
sse_t.14 <- sum(model.14$residuals^2)
sse_t.17 <- sum(model.17$residuals^2)
sse_t.22 <- sum(model.22$residuals^2)

# F. Calculate MSPE to SSE(from model)/N ratios
mspe.10 - (sse_t.10 / n)
mspe.14 - (sse_t.14 / n)
mspe.17 - (sse_t.17 / n)
mspe.22 - (sse_t.22 / n)

# G. Internal Validation (PRESSp and Cp)
PRESS_10 <- sum( (model.10$residuals/(1-influence(model.10)$hat))^2)
PRESS_14 <- sum( (model.14$residuals/(1-influence(model.14)$hat))^2)
PRESS_17 <- sum( (model.17$residuals/(1-influence(model.17)$hat))^2)
PRESS_22 <- sum( (model.22$residuals/(1-influence(model.22)$hat))^2)


Cp_10 = ols_mallows_cp(model.10, full.model)
Cp_14 = ols_mallows_cp(model.14, full.model)
Cp_17 = ols_mallows_cp(model.17, full.model)
Cp_22 = ols_mallows_cp(model.22, full.model)

# create a summary table
out_tbl <- data.frame(p = c(10, 14, 17, 22),
                                          cp = c(Cp_10, Cp_14, Cp_17, Cp_22),
                                          press <- c(PRESS_10, PRESS_14, PRESS_17, PRE
SS_22),
                                          mspe = c(mspe.10, mspe.14, mspe.17, mspe.2
2),
                                          mspe_sse =  c(mspe.10 - (sse_t.10 / n), msp
```

```
e.14 - (sse_t.14 / n),

mspe.17 - (sse_t.17 / n),mspe.22 - (sse_t.22 / n)))
# stargazer(out_tbl, summary = FALSE)
```

Step 7: Final Model

```
# A. Fit the best model on all of the data
#  1. Apply transformations
concrete_full_transformed <- data.frame(concrete)
concrete_full_transformed$concrete_strength <- sqrt(concrete$concrete_strength)
concrete_full_transformed$age <- log(concrete$age)
concrete_full_transformed$blast_furnace_slag <- sqrt(concrete$blast_furnace_slag)
concrete_full_transformed$superplasticizer <- sqrt(concrete$superplasticizer)
concrete_full_transformed$cement <- sqrt(concrete$cement)

#  2. Fit model
model.10.full <- lm(concrete_strength~age + cement + superplasticizer + blast_furnace_sl
ag + water + fly_ash + cement*age + water*superplasticizer + blast_furnace_slag*water, c
oncrete_full_transformed)

#  3. Review model diagnostics
#   i. Diagnostic Plots
diag_m10_full <- gg_diagnose(model.10.full, theme = theme_classic(), plot.all = FALSE)
diag_m10_full_gg <- ggarrange(fix_resid_plt(diag_m10_full$res_fitted) +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m10_full$qqplot +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m10_full$scalelocation +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 diag_m10_full$resleverage +

theme(plot.title = element_text(hjust = 0.5, size = 12),

plot.margin = margin(2,1,2,1, "cm")),

 ncol =2, nrow = 2)
# ggsave("plots_pg20.pdf", annotate_figure(diag_m10_full_gg, top = text_grob(bquote("Fig
ure 23: Diagnostic Plots For Model 1 Fit on All Data"))), width = 8.5, height = 11)

#  4. Model summary
summary(model.10.full)
# stargazer(model.10.full, intercept.bottom = FALSE, single.row = TRUE)

#  5. Find outliers and HIFs
#   i. Hat values
hats <- data.frame(val=hatvalues(model.10.full))
```

```r
#   ii. Greater than 2p/n (slide 18 week 9)
length(hats[hats["val"] > 2*9/1030, ])
# 103 (10% of the data is outlying in X)

hats[hats["val"] > (2*9)/1030, ]

#   iii. Cook's Distance
c_dist <- cooks.distance(model.10.full)

#   iv. Filter to gt 4/(n-p)
length(c_dist[c_dist > (4/(1030-9))])
# 74 (7.2% of the data is highly influential)

c_dist[c_dist > (4/(1030-9))]

c_dist[c_dist > 1]
```

Works Cited

Ahn, Kwang W. "Modern variable selection techniques." *Medical College of Wisconsin Division of Biostatistics Newsletter*, vol. 22, no. 1, 2016, https://www.mcw.edu/-/media/MCW/Departments/Biostatistics/vol22no1ahn.pdf.

Jamal, Haseeb. "Compressive Strength of Concrete | Definition, Importance, Applications." *AboutCivil.Org*, 29 January 2017, https://www.aboutcivil.org/compressive-strength-of-concrete.html.

Pardoe, Iain, et al. "More on Data Transformations." *STAT 501*, Penn State University, https://online.stat.psu.edu/stat501/lesson/9/9.5.

Torre, A., et al. "Prediction of compression strength of high performance concrete using artificial neural networks." *Journal of Physics: Conference Series*, vol. 582, no. 1, 2015, http://dx.doi.org/10.1088/1742-6596/582/1/012010.

Watthanacheewakul, Lakhana. "Transformations for Left Skewed Data." *Proceedings of the World Congress on Engineering*, 2021, http://www.iaeng.org/publication/WCE2021/WCE2021_pp101-106.pdf.