Applied Regression Project

Charlie Krebs

Data

```
dat <- read.csv("concrete_data_final.csv")
```

Data Summary

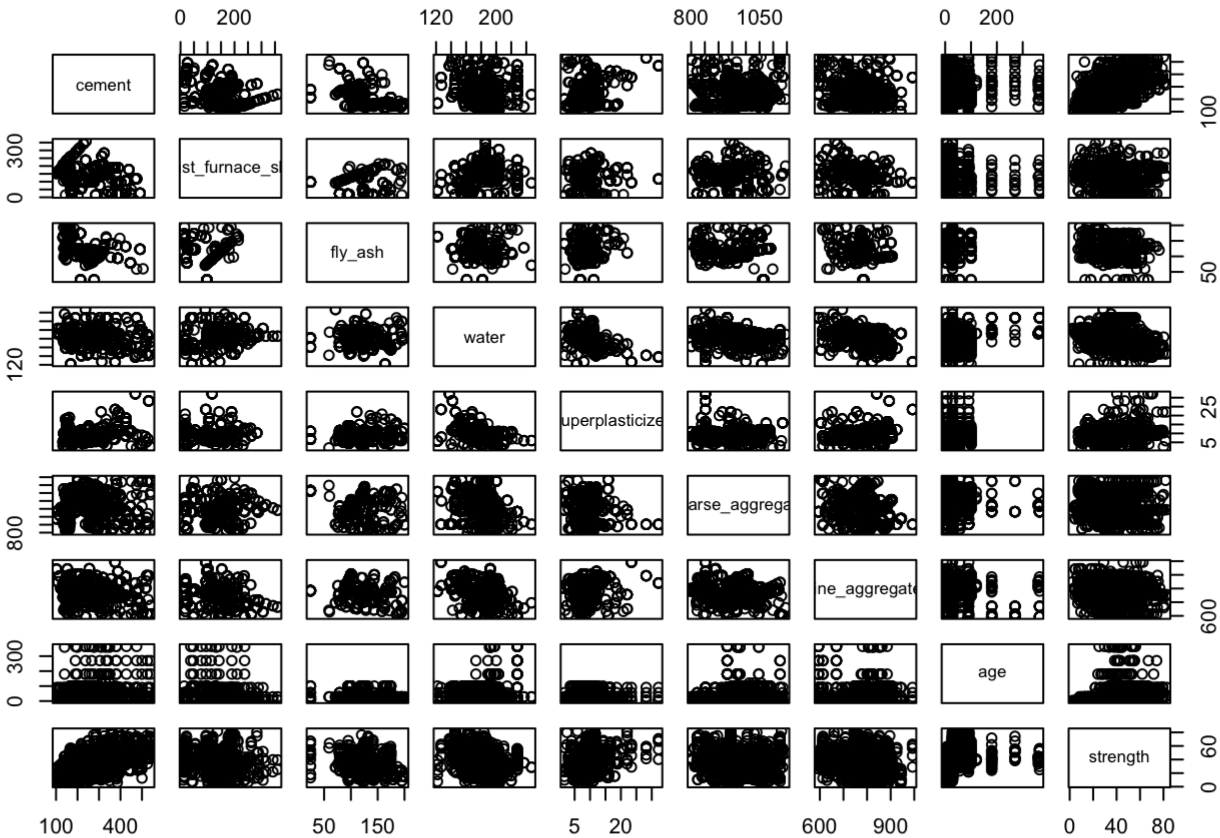
```
summary(dat)
```

```
##      cement      blast_furnace_slag    fly_ash           water      
##  Min.   :102.0   Min.   : 11.0      Min.   : 24.50   Min.   :121.8  
##  1st Qu.:192.4   1st Qu.: 95.0      1st Qu.: 97.85   1st Qu.:164.9  
##  Median :272.9   Median :135.7      Median :121.40   Median :185.0  
##  Mean   :281.2   Mean   :136.2      Mean   :120.29   Mean   :181.6  
##  3rd Qu.:350.0   3rd Qu.:189.0      3rd Qu.:141.00   3rd Qu.:192.0  
##  Max.   :540.0   Max.   :359.4      Max.   :200.10   Max.   :247.0  
##                  NA's   :471        NA's   :566                     
##  superplasticizer coarse_aggregate fine_aggregate       age        
##  Min.   : 1.700   Min.   : 801.0   Min.   :594.0   Min.   :  1.00  
##  1st Qu.: 6.950   1st Qu.: 932.0   1st Qu.:731.0   1st Qu.:  7.00  
##  Median : 9.400   Median : 968.0   Median :779.5   Median : 28.00  
##  Mean   : 9.817   Mean   : 972.9   Mean   :773.6   Mean   : 45.66  
##  3rd Qu.:11.600   3rd Qu.:1029.4   3rd Qu.:824.0   3rd Qu.: 56.00  
##  Max.   :32.200   Max.   :1145.0   Max.   :992.6   Max.   :365.00  
##  NA's   :379                                                       
##     strength    
##  Min.   : 2.33  
##  1st Qu.:23.71  
##  Median :34.45  
##  Mean   :35.82  
##  3rd Qu.:46.13  
##  Max.   :82.60  
## 
```

Looking at the summary of the data, the main point to note is the large amounts of missing data in the `blast_furnace_slag` , `fly_ash` , and `superplasticizer` variables. Another point to note is that some of the variables (`superplasticizer` for example) looked a little bit skewed, so preprocessing will be done.

Visualizations

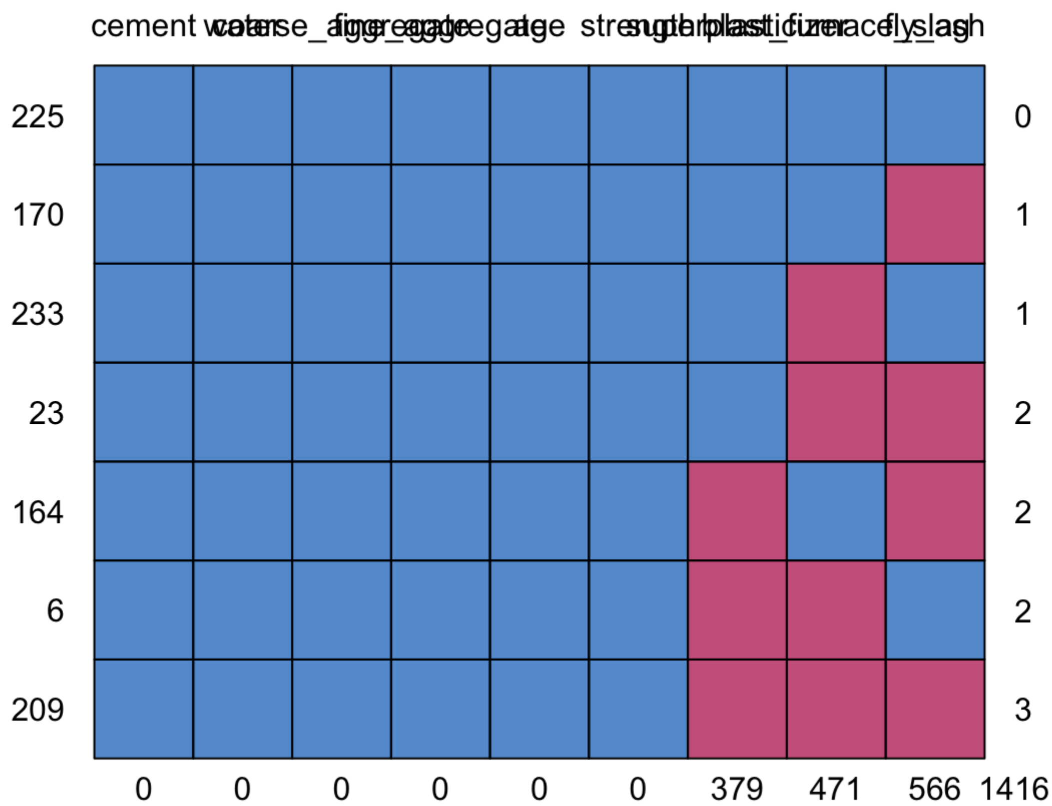
```
plot(dat)
```



We are looking at modeling `strength` on the other variables, so the plots of the strength versus the other variables can be seen along the bottom row and the far right column.

`Missing Data`

```
md.pattern(dat)
```




```
##     cement water coarse_aggregate fine_aggregate age strength superplasticizer
## 225      1     1                1              1   1        1                1
## 170      1     1                1              1   1        1                1
## 233      1     1                1              1   1        1                1
## 23       1     1                1              1   1        1                1
## 164      1     1                1              1   1        1                0
## 6        1     1                1              1   1        1                0
## 209      1     1                1              1   1        1                0
##          0     0                0              0   0        0              379
##     blast_furnace_slag fly_ash
## 225                  1       1    0
## 170                  1       0    1
## 233                  0       1    1
## 23                   0       0    2
## 164                  1       0    2
## 6                    0       1    2
## 209                  0       0    3
##                    471     566 1416
```

There is missing data in the `superplasticizer` , `blast_furnace_slag` , and `fly_ash` variables.

`Imputation`

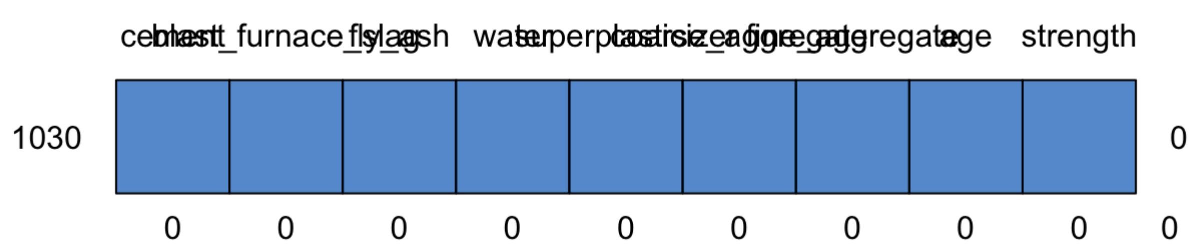
```
dat_miss <- mice(dat, m = 1)
```

```
##
##  iter imp variable
##   1   1  blast_furnace_slag  fly_ash  superplasticizer
##   2   1  blast_furnace_slag  fly_ash  superplasticizer
##   3   1  blast_furnace_slag  fly_ash  superplasticizer
##   4   1  blast_furnace_slag  fly_ash  superplasticizer
##   5   1  blast_furnace_slag  fly_ash  superplasticizer
```

```
dat_imp <- complete(dat_miss)

md.pattern(dat_imp)
```

```
##  /\     /\
## {  `---'  }
## {  O   O  }
## ==>  V <==  No need for mice. This data set is completely observed.
##  \  \|/  /
##   `-----'
```



```
##      cement blast_furnace_slag fly_ash water superplasticizer coarse_aggregate
## 1030      1                  1       1     1                1                1
##           0                  0       0     0                0                0
##      fine_aggregate age strength
## 1030              1   1        1 0
##                   0   0        0 0
```

Preprocessing

```
pre_process_mod <- preProcess(dat_imp, method = c("YeoJohnson", "center", "scale"))
dat_processed <- predict(pre_process_mod, newdata = dat_imp)
```

The data was preprocessed by centering, scaling, and Yeo Johnson transforming after imputation.

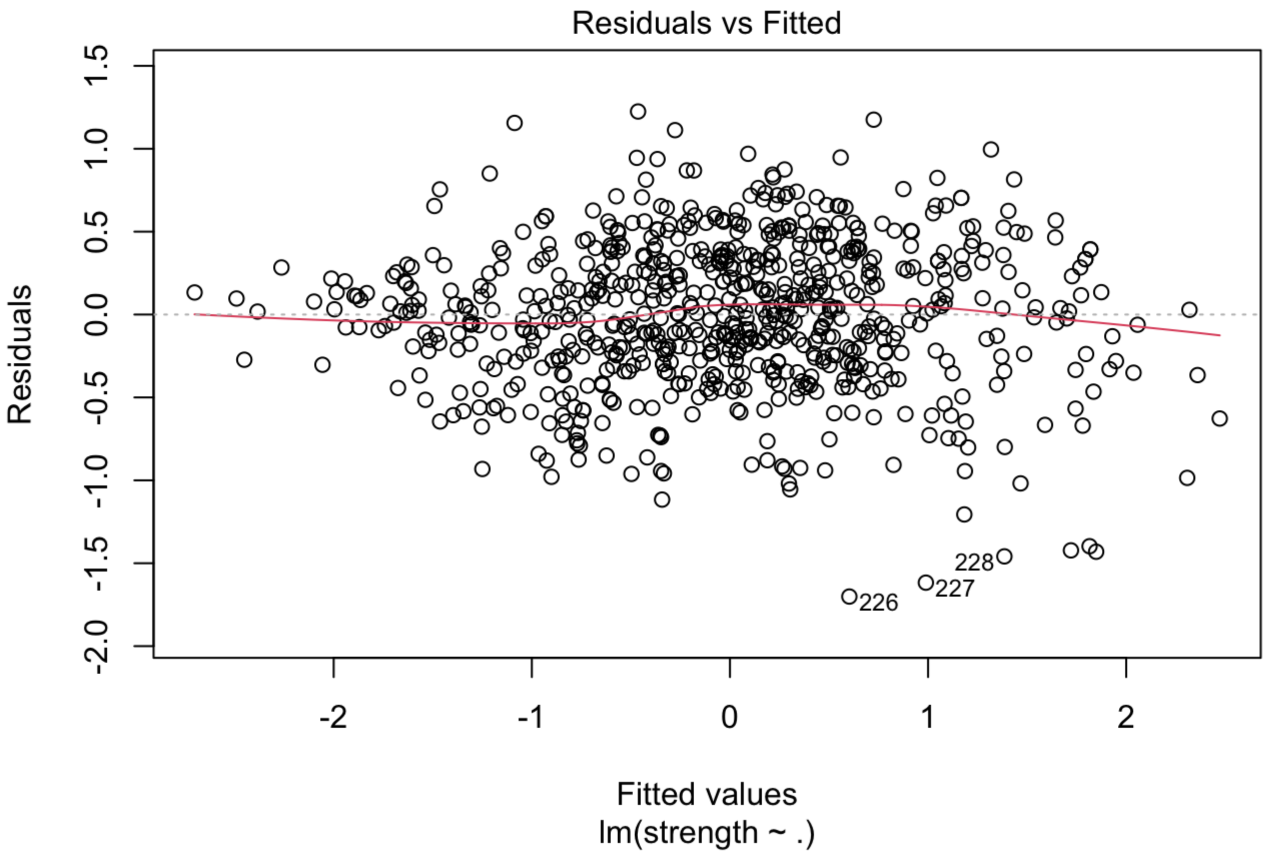
Test/Train Sets

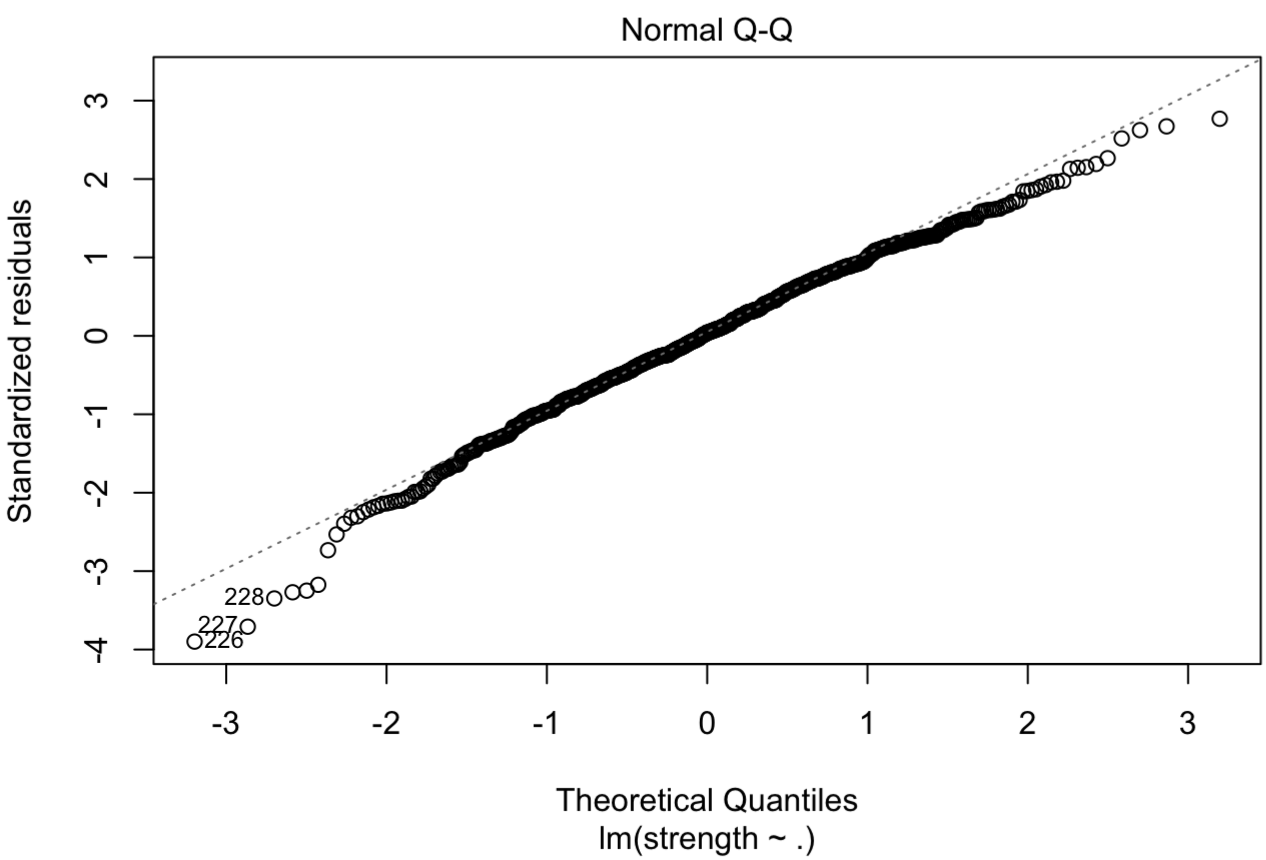
```
ind_train <- sample(1:1030, .7 * 1030)
dat_train <- dat_processed[ind_train,]
dat_test <- dat_processed[-ind_train,]
```

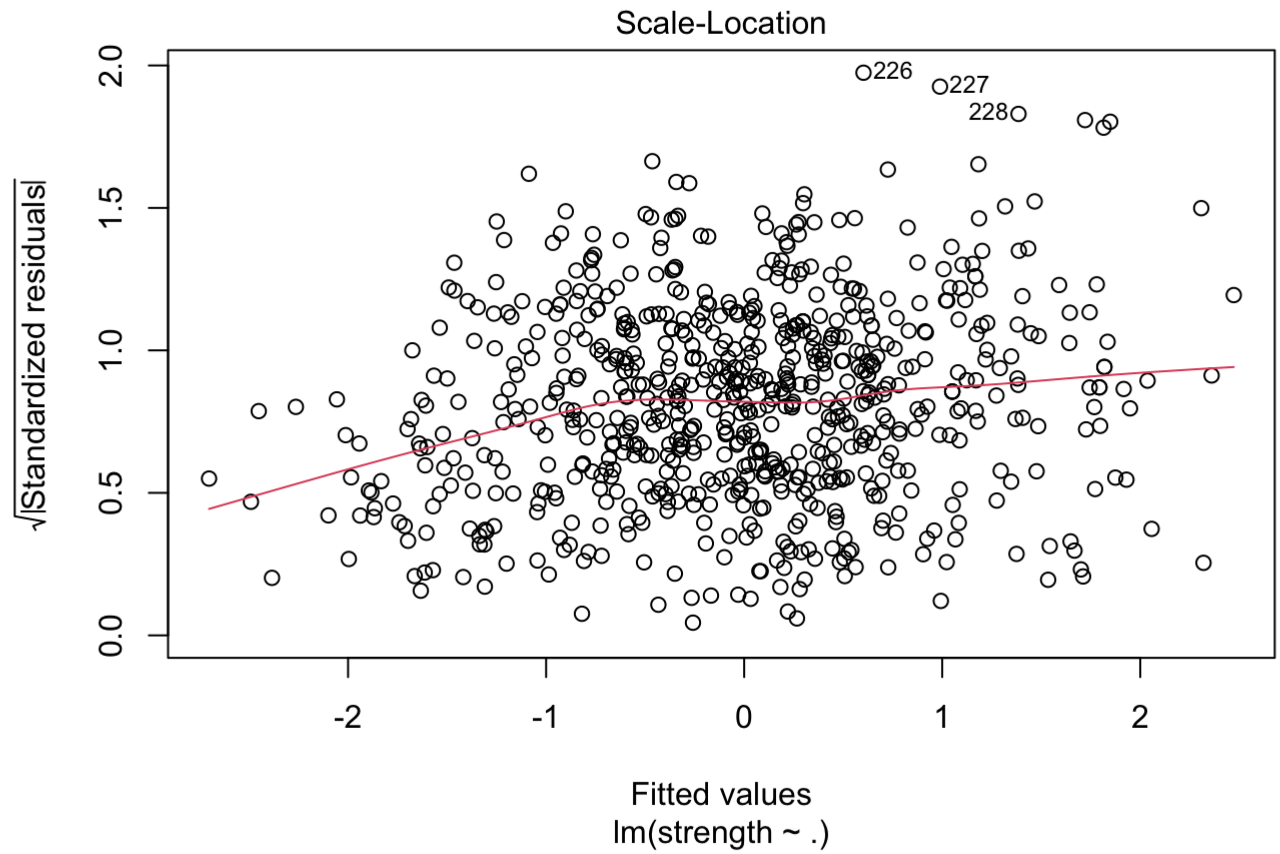
Identifying Outliers

```
train_mod <- lm(strength ~ ., data = dat_train)
test_mod <- lm(strength ~ ., data = dat_test)

plot(train_mod)
```

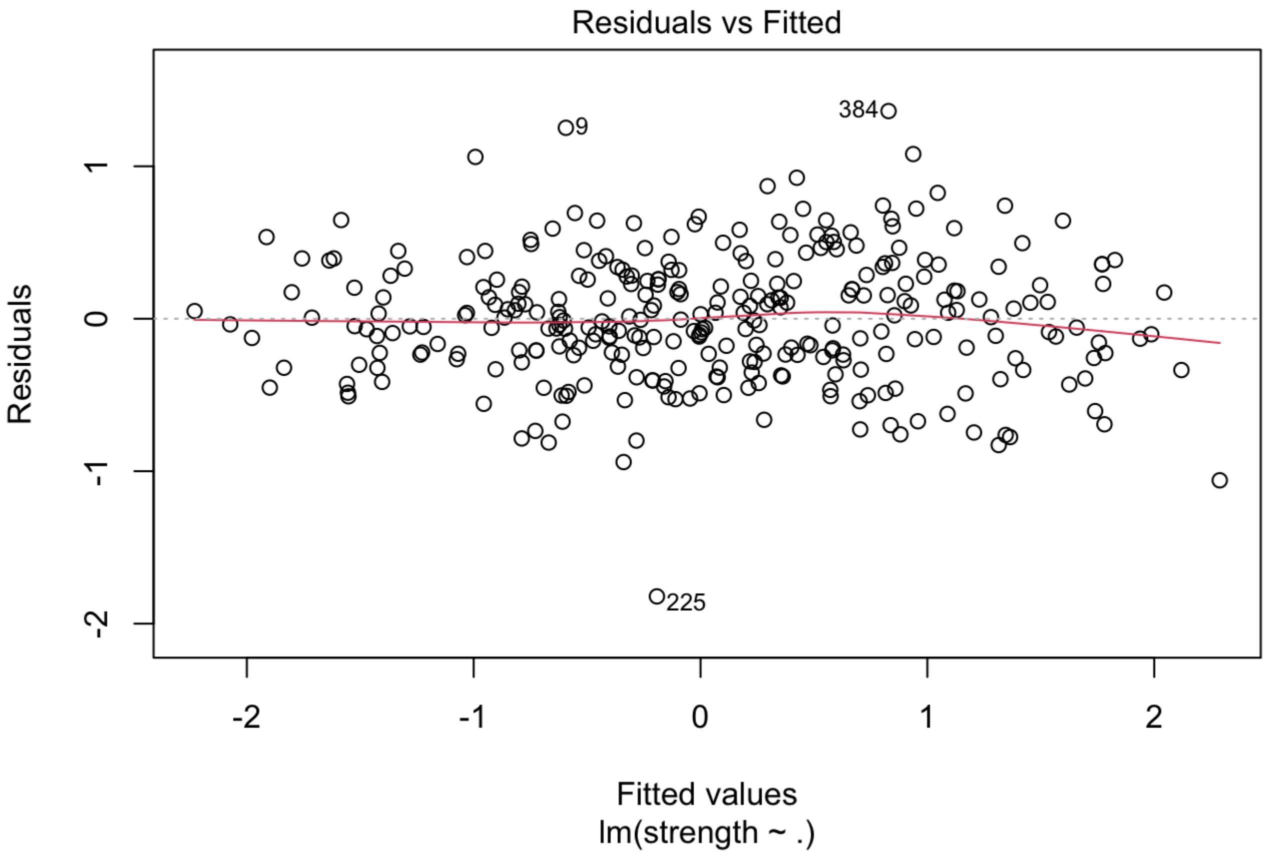

Residuals vs Fitted
Residuals
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
-2
-1
0
1
2
228
227
226
Fitted values
lm(strength ~ .)

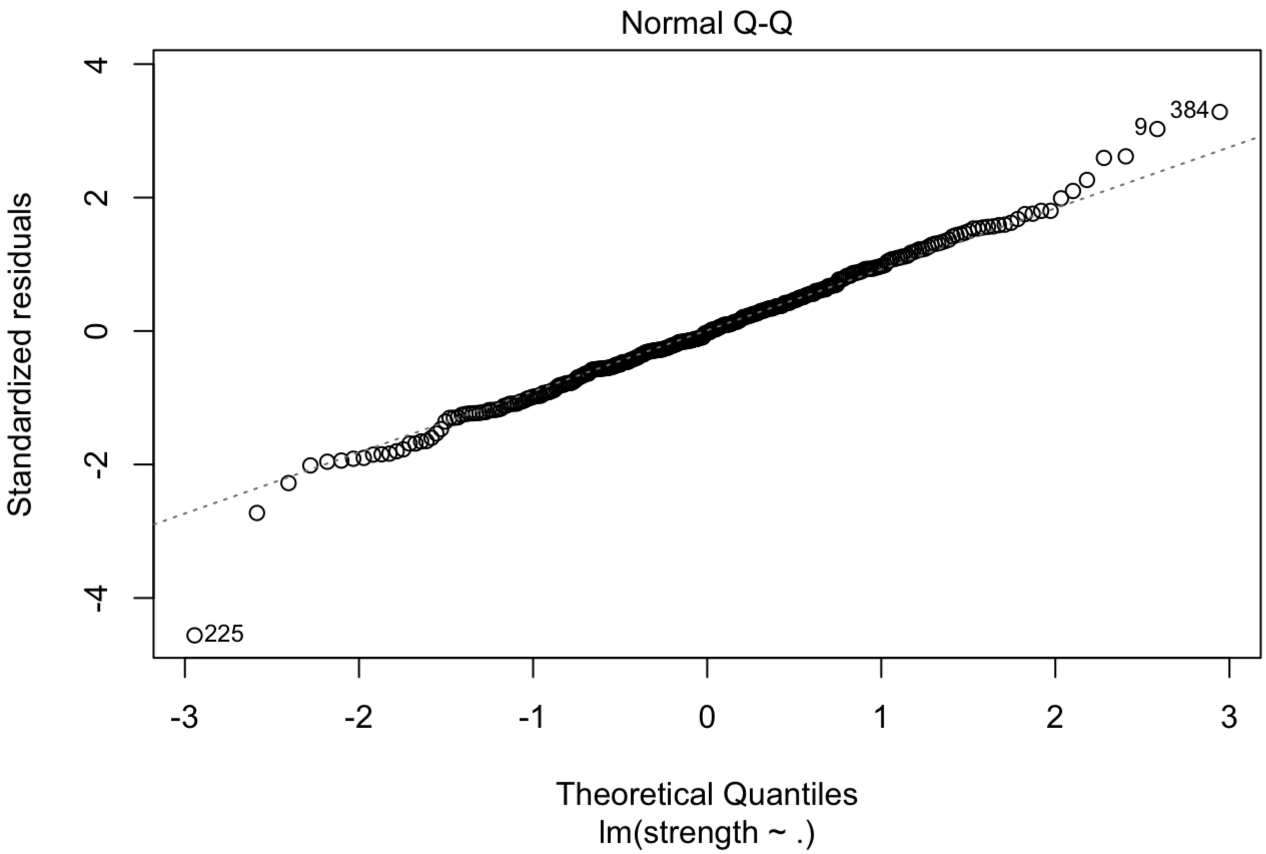

Normal Q-Q
Standardized residuals
3
2
1
0
-1
-2
-3
-4
-3
-2
-1
0
1
2
3
228
227
226
Theoretical Quantiles
lm(strength ~ .)

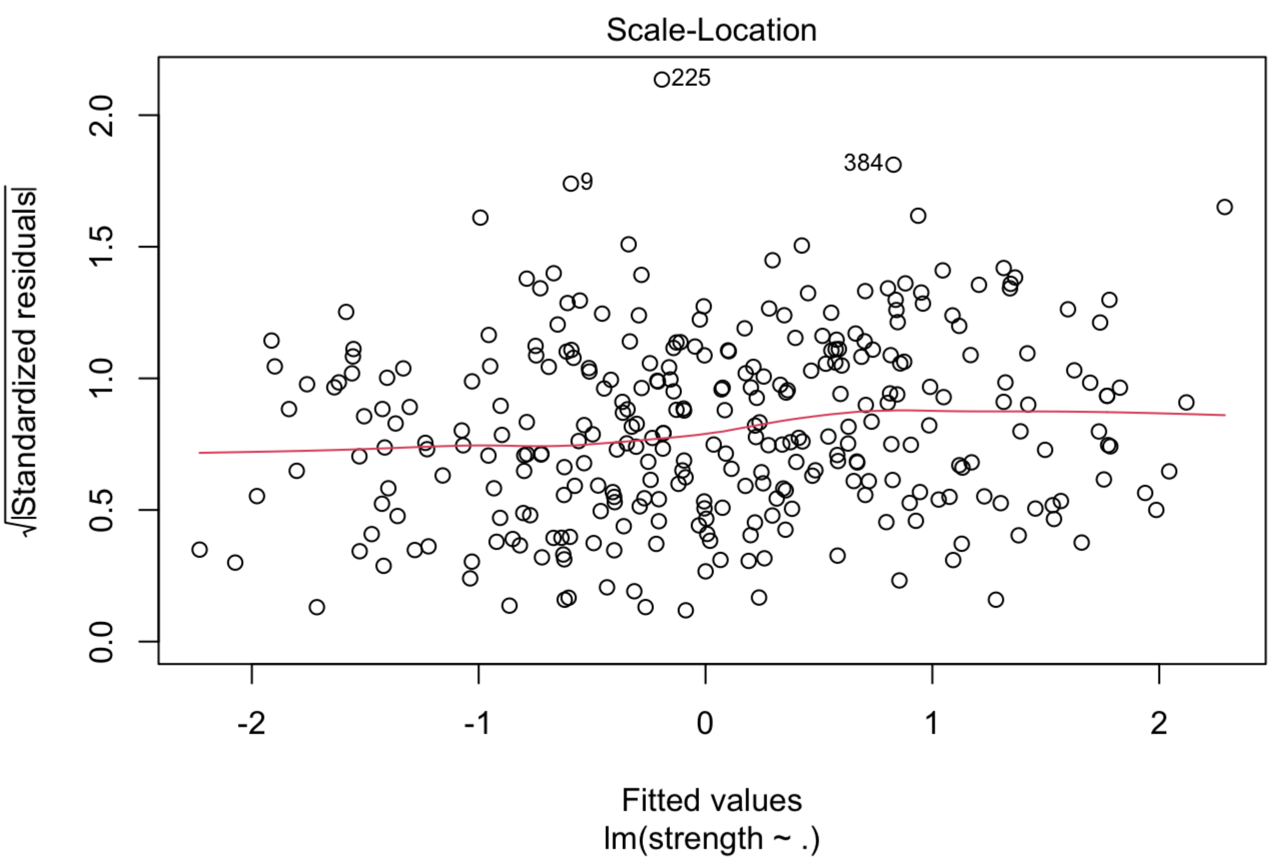

Scale-Location
226
227
228
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
-2
-1
0
1
2
Fitted values
lm(strength ~ .)

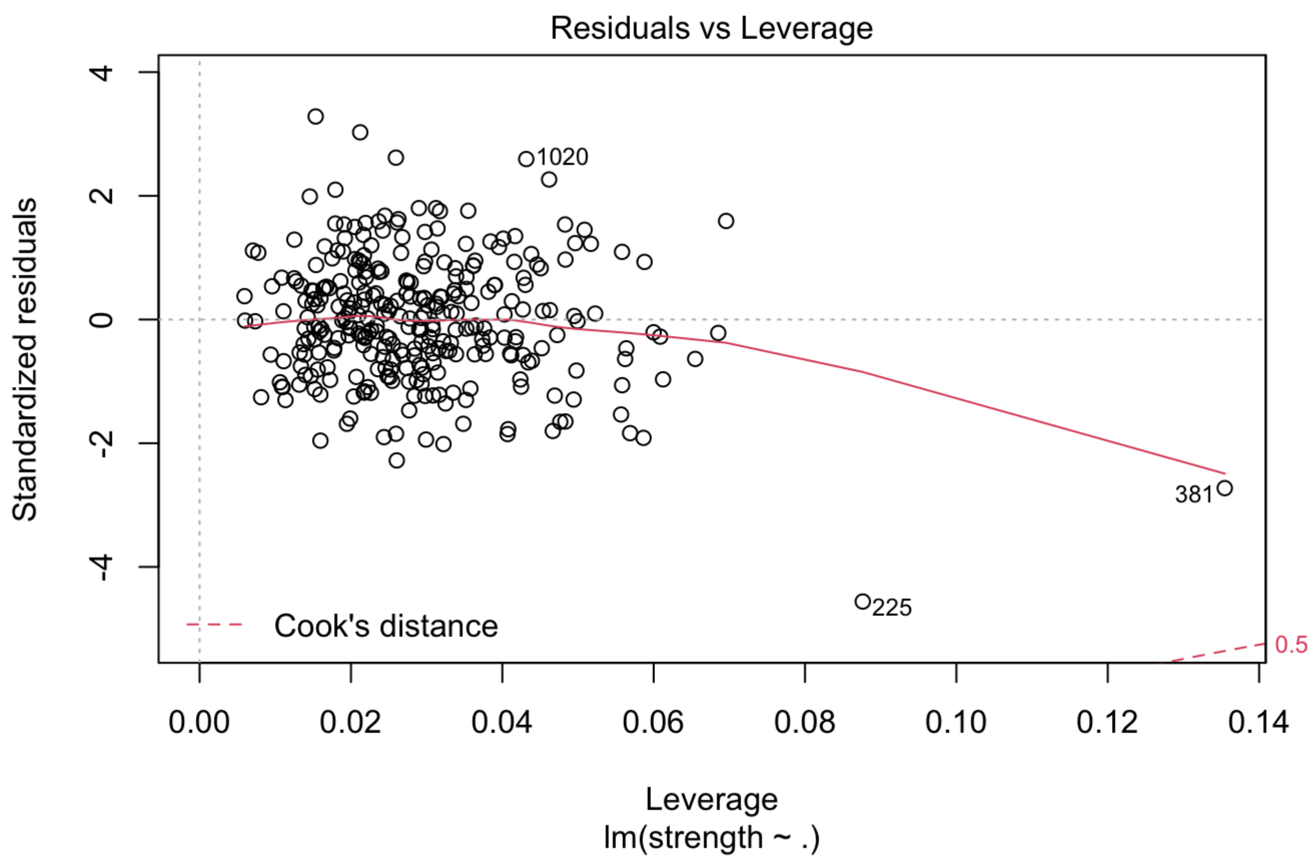

Residuals vs Leverage
Standardized residuals
2
0
-2
-4
229
227
226
Cook's distance
0.00
0.01
0.02
0.03
0.04
Leverage
lm(strength ~ .)

```
plot(test_mod)
```




Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(strength ~ .)
225
9
384
4
2
0
-2
-4
-3
-2
-1
0
1
2
3


Scale-Location
√|Standardized residuals|
Fitted values
lm(strength ~ .)
225
384
9
0.0
0.5
1.0
1.5
2.0
-2
-1
0
1
2



Based on the residuals versus leverage plots, there are no outliers in either the training or testing data. I think that deskewing the variables helped to handle the potential outliers.

Modeling

```
cv_5 <- trainControl(method = "repeatedcv",
                     number = 10,
                     repeats = 5)

aic_mod <- train(strength ~ .,
                 data = dat_train,
                 method = "lmStepAIC",
                 trControl = cv_5,
                 trace = 0)
ridge_mod <- train(strength ~ .,
                   data = dat_train,
                   trControl = cv_5,
                   method = "ridge")
glm_mod <- train(strength ~ .,
                   data = dat_train,
                   trControl = cv_5,
                   method = "glm")
```

I used AIC, ridge regression, and glm to model the data with cross-validation.

Results

```
aic_mod$results[2]
```

```
##        RMSE
## 1 0.4476188
```

```
aic_mod$results[2] - (2 * aic_mod$results[5])
```

```
##        RMSE
## 1 0.3587628
```

```
aic_mod$results[2] + (2 * aic_mod$results[5])
```

```
##        RMSE
## 1 0.5364749
```

The first value is the expected RMSE for the AIC model. The second and third values are the interval for the expected RMSE.

```
ridge_mod$results[1,2]
```

```
## [1] 0.4471155
```

```
ridge_mod$results[2,2] - (2 * ridge_mod$results[2,5])
```

```
## [1] 0.3514003
```

```
ridge_mod$results[2,2] + (2 * ridge_mod$results[2,5])
```

```
## [1] 0.5428292
```

The first value is the expected RMSE for the ridge model. The second and third values are the interval for the expected RMSE.

```
glm_mod$results[2]
```

```
##        RMSE
## 1 0.4480668
```

```
glm_mod$results[2] - (2 * glm_mod$results[5])
```

```
##        RMSE
## 1 0.3846392
```

```
glm_mod$results[2] + (2 * glm_mod$results[5])
```

```
##        RMSE
## 1 0.5114944
```

The first value is the expected RMSE for the generalized linear model. The second and third values are the interval for the expected RMSE.

`Conclusions`

Based on the results from the models, all three models that I used gave relatively the same expected RMSE. The ridge regression had the smallest expected RMSE with the AIC model just behind and then the glm. The generalized linear model had the smallest interval for the possible RMSE values for the future cement predictions. The other two models had larger intervals than the glm. All three models gave expected RMSE values that were very small. This is good because it shows that they do a good job in modeling the data. Since the models did have very similar expected RMSE values, I think that I would choose the generalized linear model as the best model for this data. Overall, all three models proved to be good representations of the cement data.