# Concrete Compressive Strength.

## Group 4 Team Project.

Alejandro Avila, Anthony Ramirez, Benjamin B An, Chengze Liu, Vishu Kodikanti.

## Introduction.

Concrete is the most important material in civil engineering, and concrete compressive strength is one determining factor for concrete performance. The chemical compositions, physical formations, as well as the age of concrete, may impact its compressive strength. With gratitude to I-Cheng Yeh from UC Irvine Machine Learning Repository, we examine the dataset **Concrete Compressive Strength**.

We create two different models using the concrete compressive strength dataset: one utilizing linear regression and the other employing a random forest approach. Our goal is to determine the best model and identify the most significant predictors to aid in predicting concrete compressive strength.

## Dataset Description.

The dataset provided by the UC Irvine Machine Learning Repository will help us investigate the influence of different materials on concrete compression strength. This dataset consists of 1030 observations and 9 variables. Among these, 8 variables serve as predictors, while 1 variable serves as the response variable. Together, they offer a robust and comprehensive dataset that forms the foundation for our analysis. Each piece of data in this dataset tells us about a specific mix of materials, how they are assembled, and the conditions they are subjected to. The dataset helps to identify all the factors contributing to the strength of concrete. In the following sections, we will provide detailed information about these variables:

## Predictors:

- ***Cement***: is a continuous numerical value measured in kilograms per cubic meter (kg/m$^3$) and is the binder that sets,hardens and adheres to other materials to bind them together.
- ***Blast_Furnace_Slag***: is an integer numerical value measured in kilograms per cubic meter (kg/m$^3$) and is a byproduct of iron when added to concrete improves its properties suck as workability, strength and durability.
- ***Fly_Ash***: is a continuous numerical value measured in kilograms per cubic meter (kg/m$^3$) and is a byproduct of coal power plants that can be used to improve durability and workability of concrete mixes.
- ***Water***: is a numerical continuous value measured in kilograms per cubic meter (kg/m$^3$) and water is added to concrete that helps to create a paste that glues all aggregates together.

- ***Superplasticizer***: is a numerical continuous value measured in kilograms per cubic meter and is an admixture that transforms stiff, low-slump concrete into flowing, pourable and easily placed concrete.
- ***Coarse_Aggregate***: is a numerical continuous value measured in kilograms per cubic meter (kg/m$^3$) and is typically made up of crushed stone, gravel, or recycled concrete. It is used to provide bulk to the concrete mixture, which helps to reduce shrinkage and cracking.
- ***Fine_Aggregate***: is a numerical continuous value measured in kilograms per cubic meter (kg/m$^3$) and is a term used to describe small, inert materials, whether granular or crushed stone. These fine aggregates are essential components in concrete mixtures and play a crucial role in determining the properties of the resulting concrete.
- ***Age***: is an integer numerical value measured in days and this variable refers to the number of days since the concrete sample was created or cast.

### Response:

- ***Concrete_Strength***: The target variable in our data set is a continuous numerical value expressed in megapascals (MPa). It represents the compressive strength of materials, in this case concrete, and is widely utilized in material science and engineering for assessing the strength characteristics of substances.

## Linear Regression

### Importing the data set.

We read our CSV file and print the dataset summary with the following code:

```
#importing the data set and displaying the summary
strength<-read.csv("C:/Users/alex-/OneDrive/Desktop/Concrete.csv")
summary(strength)
```

```
     Cement      Blast_Furnace_Slag    Fly_Ash           Water      
 Min.   :102.0   Min.   :  0.0      Min.   :  0.00   Min.   :121.8  
 1st Qu.:192.4   1st Qu.:  0.0      1st Qu.:  0.00   1st Qu.:164.9  
 Median :272.9   Median : 22.0      Median :  0.00   Median :185.0  
 Mean   :281.2   Mean   : 73.9      Mean   : 54.19   Mean   :181.6  
 3rd Qu.:350.0   3rd Qu.:142.9      3rd Qu.:118.30   3rd Qu.:192.0  
 Max.   :540.0   Max.   :359.4      Max.   :200.10   Max.   :247.0  
 Superplasticizer Coarse_Aggregate Fine_Aggregate       Age        
 Min.   : 0.000   Min.   : 801.0   Min.   :594.0   Min.   :  1.00  
 1st Qu.: 0.000   1st Qu.: 932.0   1st Qu.:731.0   1st Qu.:  7.00  
 Median : 6.400   Median : 968.0   Median :779.5   Median : 28.00  
 Mean   : 6.205   Mean   : 972.9   Mean   :773.6   Mean   : 45.66  
 3rd Qu.:10.200   3rd Qu.:1029.4   3rd Qu.:824.0   3rd Qu.: 56.00  
 Max.   :32.200   Max.   :1145.0   Max.   :992.6   Max.   :365.00  
 Concrete_Strength
 Min.   : 2.33    
 1st Qu.:23.71    
 Median :34.45    
 Mean   :35.82    
 3rd Qu.:46.13    
 Max.   :82.60    
```

The summary shows all the predictors in our dataset: *Cement*, *Blast_furnace_slag*, *Fly_Ash*, *Water*, *Superplasticizer*, *Coarse_Aggregate*, *Fine_Aggregate*, *Age* and the target variable *Concrete_Strength*. We can interpret the data as follows:

- ***Cement***: The minimum amount of cement used is 102.0 (kg/m$^3$), and the maximum is 540.0 (kg/m$^3$). The mean of cement used is approximately 281.2 (kg/m$^3$).
- ***Blast_Furnace_Slag***: The minimum amount of Blast Furnace Slag used is 0.0 (kg/m$^3$), and the maximum is 359.4 (kg/m$^3$). The mean of Blast Furnace Slag used is approximately 73.9 (kg/m$^3$).
- ***Fly_Ash***: The minimum amount of Fly Ash used is 0.0 (kg/m$^3$), and the maximum is 200.1 (kg/m$^3$). The mean of Fly Ash used is approximately 54.19 (kg/m$^3$).
- ***Water***: The minimum amount of water used is 121.8 (kg/m$^3$), and the maximum is 247.0 (kg/m$^3$). The mean of water used is approximately 181.6 (kg/m$^3$).
- ***Superplasticizer***: The minimum amount of Superplasticizer used is 0.0 (kg/m$^3$), and the maximum is 32.2 (kg/m$^3$). The mean of Superplasticizer used is approximately 6.205 (kg/m$^3$).
- ***Coarse_Aggregate***: The minimum amount of Coarse Aggregate used is 801.0 (kg/m$^3$), and the maximum is 1145.0 (kg/m$^3$). The mean of Coarse Aggregate used is approximately 972.9 (kg/m$^3$).
- ***Fine_Aggregate***: The minimum amount of Fine Aggregate used is 594.0 (kg/m$^3$), and the maximum is 992.6 (kg/m$^3$). The mean of Fine Aggregate used is approximately 773.6 (kg/m$^3$).
- ***Age***: The minimum amount of days in Age is one day, and the maximum is 365 days. The mean of Age is approximately 45.66 days.

- ***Concrete_Strength***: The minimum amount of the concrete compressive strength is 2.33 MPa, and the maximum is 82.60 MPa. The mean of the concrete compressive strength is 35.82.

Then we check if we have any missing values in our dataset:

```
any(is.na(strength))
```

```
[1] FALSE
```

We get FALSE as a result showing that our dataset does not have any missing values.

### Linear Regression description.

Linear regression is a statistical technique used to understand and quantify the relationship between one or more predictor variables and a continuous outcome variable. In our analysis, we employ linear regression to investigate how changes in certain characteristics of concrete mixtures relate to the compressive strength of the resulting concrete.

The fundamental concept behind linear regression is to find the best-fitting line that represents the relationship between the predictor variables and the outcome variable. This line is characterized by an intercept term ($\beta_0$) and slope coefficients ($\beta_1$,$\beta_2$,..,$\beta_n$) associated with each predictor variable.

The equation of the linear regression model can be expressed as:

p(x)=$\beta_0 + \beta_1 \times x_1 + \beta_2 \times x_2 + ... + \beta_n \times x_n$

In this equation, p(x) represents the predicted outcome variable, while $x_1$,$x_2$,...,$x_n$ represent the predictor variables. The coefficients $\beta_0$,$\beta_1$,...,$\beta_n$ are estimated from the data to minimize the difference between the observed and predicted values of the outcome variable.

When we translate this general formula into our model we get the following linear regression model that we can use to predict our outcome:

Strength = $\beta_0$ +$\beta_1$ × Cement + $\beta_2$ × Blast Furnace Slag + $\beta_3$ × Fly Ash+ $\beta_4$ × Water +$\beta_5$ × Superplasticizer+ $\beta_6$ × Coarse Aggregate+$\beta_7$ × Fine Aggregate+ $\beta_8$ × Age +$\epsilon$

Then we use the following code to create our linear model and to print a summary of our model:

#Creating the linear regression model and printing summary.

```
strength.lm = lm(Concrete_Strength ~ .,data=strength)
summary(strength.lm)
```

```
Call:
lm(formula = Concrete_Strength ~ ., data = strength)

Residuals:
    Min      1Q  Median      3Q     Max
-28.654  -6.302   0.703   6.569  34.450

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)        -23.331214  26.585504  -0.878 0.380372
Cement               0.119804   0.008489  14.113  < 2e-16 ***
Blast_Furnace_Slag   0.103866   0.010136  10.247  < 2e-16 ***
Fly_Ash              0.087934   0.012583   6.988 5.02e-12 ***
Water               -0.149918   0.040177  -3.731 0.000201 ***
Superplasticizer     0.292225   0.093424   3.128 0.001810 **
Coarse_Aggregate     0.018086   0.009392   1.926 0.054425 .
Fine_Aggregate       0.020190   0.010702   1.887 0.059491 .
Age                  0.114222   0.005427  21.046  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.4 on 1021 degrees of freedom
Multiple R-squared:  0.6155,    Adjusted R-squared:  0.6125
F-statistic: 204.3 on 8 and 1021 DF,  p-value: < 2.2e-16
```

From the above summary of our linear regression we can observe that the predictors: Cement, Blast_furnace_slag, Fly_Ash, Water, Superplasticizer and Age are statistically significant in predicting the concrete compressive strength because the p-value is less than the significance level of 0.05, indicating strong evidence against the null hypothesis that the coefficient for these predictors is zero. This means that changes in the amounts of these materials, along with the age of the concrete, have a significant impact on its compressive strength.

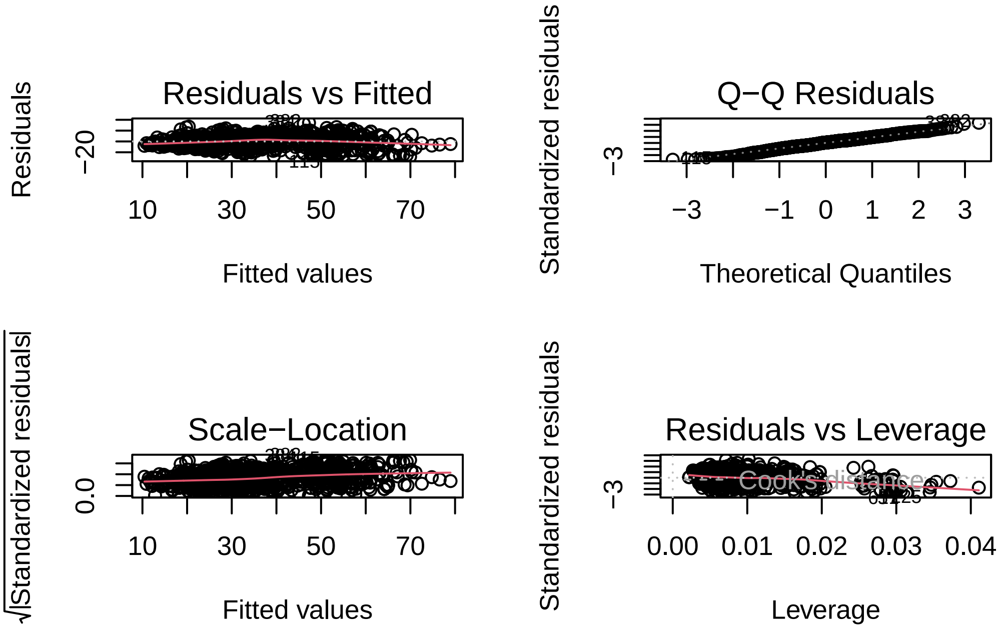
Additionally, the coefficients for these predictors are positive, suggesting that increases in their amounts lead to higher concrete compressive strength Conversely, the coefficient for Water is negative, indicating that higher water content in the concrete mixture is associated with lower compressive strength. The model's overall performance is reasonably good, with an adjusted R-squared value of 0.6125, indicating that approximately 61.25% of the variability in concrete compressive strength is explained by the predictors included in the model.

While these p-values are slightly above 0.05 for Coarse_Aggregate (0.0544525) and Fine_Aggregate (0.059491), they are still relatively low, indicating that there is some evidence to reject the null hypothesis that the coefficients for these variables are zero.In practical terms, this suggests that there may still be a meaningful relationship between Coarse_Aggregate, Fine_Aggregate, and concrete compressive strength, but their significance is weaker compared to other predictors in the model.Therefore, while Coarse_Aggregate and Fine_Aggregate may not be as strongly associated with concrete compressive strength as other variables in the model, We think they need to be still considered becaue they are relatively close to the significant p-value of 0.05.

### Linear regression plots.

We will now create plots for the linear model we have constructed. We will use the following code to generate the Residuals vs Fitted Plot, the Q-Q Residuals Plot, the Scale-Location Plot, and the Residuals vs Leverage Plot. These visual representations will help us better understand our data:

```r
par(mfrow = c(2, 2))
plot(strength.lm)
```



From the Residuals vs Fitted plot, we can visually inspect the relationship between the predicted values and the residuals. This helps in identifying patterns or trends in the residuals, which can indicate potential issues with the model assumptions or data. In our plot, we observe randomness where our points are scattered around zero. This indicates that the model captures all the information in the data, and there are no systematic errors. We don't observe signs of heteroscedasticity, as the fitted values are not correlated with the residuals; they appear to be independent. The plot exhibits a somewhat linear pattern, and the residuals may occasionally become negative as fitted values increase.

From the Q-Q plot, we can determine if the residuals of the regression model are normally distributed. If the points in this plot fall roughly along a straight diagonal line, then we can assume the residuals are normally distributed. In our plot, we observe that the points fall approximately along a straight line, indicating that the data follows the theoretical distribution closely. This suggests a good fit between the data and the chosen distribution, supporting the assumption of normality for the residuals. Additionally, the straight line alignment indicates that the variability of the residuals is consistent across different quantiles, further validating the model's assumptions.

From the Scale-Location plot, we can determine whether the residuals of a regression model exhibit constant variance across the range of fitted values. In out plot we observe the residuals are approximately constant across the range of fitted values this indicates that the assumption of constant variance (homoscedasticity) is met, and the model's residuals have consistent variability across different levels of the predictor variables.

From the Residual vs Leverage plot, we can determine influential observations; if any points in this plot fall outside of Cook's distance (the dashed lines), then it is considered an influential observation. In our plot, there are no values outside of the dashed lines. This means there aren't any overly influential points in our dataset.

## Training testing.

In this section, we will train the linear regression model to evaluate its accuracy. We will begin by creating a randomized 80:20 split (training:testing) in our dataset, then scale the numeric variables, calculate the mean squared error (MSE), and perform 10-fold cross-validation. Finally, we will determine the average MSE based on these 10 iterations.

```r
MSE <- rep(0, 10)

for (i in 1:10) {
  set.seed(i)
  sample <- sample(1:nrow(strength), 0.8 * nrow(strength))
  train_data <- strength[sample, ]
  test_data <- strength[-sample, ]

  numeric_cols <- sapply(train_data, is.numeric)
  train_data_scaled <- as.data.frame(scale(train_data[, numeric_cols]))
  test_data_scaled <- as.data.frame(scale(test_data[, numeric_cols]))

  # Fit linear regression model
  strength.lm <- lm(Concrete_Strength ~ ., data = train_data_scaled)

  # Make predictions
  yhat <- predict(strength.lm, newdata = test_data_scaled)

  # Calculate MSE
  MSE[i] <- mean((yhat - test_data_scaled$Concrete_Strength)^2)
}

cat("MSE Values:", MSE, "\n")
```

```
MSE Values: 0.3886544 0.4033706 0.3762411 0.3489594 0.3674301 0.3593702 0.3920841 0.3991185 0
```

```
cat("Average Test MSE:", mean(MSE), "\n")
```

```
Average Test MSE: 0.3762034
```

The average test Mean Squared Error (MSE) obtained across all 10 training and testing iterations is 0.3762034. This indicates that, on average, the squared difference between the predicted concrete strength values and the actual values in the test set is 0.376. The Mean Squared Error (MSE) serves as a quantitative measure of how well the model's predictions align with the true values.

By averaging the squared differences between each predicted value and its corresponding actual value, the MSE offers insight into the predictive accuracy of the model. A lower MSE suggests that the model's predictions are closer to the true values, indicating higher predictive accuracy and better performance overall. Given the relatively low MSE value obtained in this analysis, it suggests that the model performs well in predicting concrete strength, making it a good model for this particular dataset.

## Random Forest.

In this section, we will employ a Random Forest model to determine if it provides a better fit for predicting concrete compressive strength. Random forest is a widely used machine learning algorithm, trademarked by Leo Breiman and Adele Cutler, that aggregates the outputs of multiple decision trees to produce a single result. Its ease of use and flexibility have contributed to its widespread adoption, as it effectively addresses both classification and regression tasks.

The random forest algorithm extends the bagging method by incorporating both bagging and feature randomness to construct an uncorrelated forest of decision trees. Feature randomness, also referred to as feature bagging or 'the random subspace method,' generates a random subset of features, thereby ensuring low correlation among decision trees. This aspect represents a fundamental difference between decision trees and random forests. While decision trees evaluate all possible feature splits, random forests only consider a subset of these features.

We start by creating the model with the library called Random Forest:

```r
# Data Modeling:  Random Forest.
library(randomForest)
```

```
Warning: package 'randomForest' was built under R version 4.3.3

randomForest 4.7-1.1

Type rfNews() to see new features/changes/bug fixes.
```

Then separate the data set to training and test into an (80/20) split to keep consistent with the split we did in the linear regression model:

```r
# Split into train and test 80/20

set.seed(200)
rf_train_subset = sample(1:nrow(strength), nrow(strength)*0.8)
test.sample = strength[-rf_train_subset,]
rf.strength = randomForest(Concrete_Strength ~ . , data = strength, subset = rf_train_subs
```

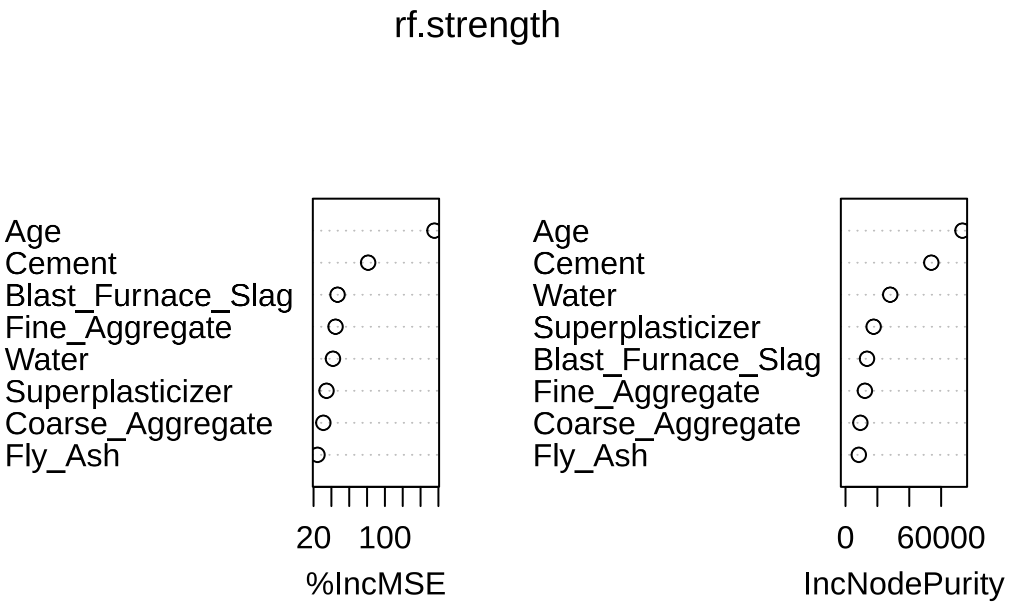
In this section, we are implementing a Random Forest algorithm to build a predictive model for concrete compressive strength. Firstly, we set the seed to ensure reproducibility of our results. Then, we create a training subset by randomly selecting 80% of the observations from the dataset "strength". The remaining 20% of the data are retained as a test sample for evaluating the performance of our model.

Next, we apply the Random Forest algorithm to the training subset, specifying the response variable "Concrete_Strength" and using all other variables as predictors. We set the number of predictors sampled at each split (mtry) to be half the total number of predictors, a common practice to promote model diversity and reduce overfitting. Finally, we extract the variable importance measures to assess the contribution of each predictor variable to the model's predictive performance.

### The Variable Importance Plot.

We then proceed to write the code to display the importance plot of concrete compressive strength:

```
varImpPlot(rf.strength)
```



The plot generated by the varImpPlot (Variable Importance Plot) function provides valuable insights into the most influential predictors identified by the Random Forest model. Upon examination of the plot, it is evident that the variables Age and Cement emerge as the most important predictors of concrete compressive strength.

This finding underscores the significant impact of these variables on the strength of concrete. Age likely represents the curing time of the concrete, which affects its overall strength as it continues to harden over time. Cement, on the other hand, serves as a fundamental component of concrete, influencing its binding properties and structural integrity. By identifying Age and Cement as the primary predictors, the plot highlights key factors that should be carefully considered in the formulation and design of concrete mixes to achieve optimal compressive strength.

## Calculating the test Mean Squared Error.

```
yhat_test = predict(rf.strength, newdata = test.sample)
(rf.MSE = mean(test.sample$Concrete_Strength - yhat_test)^2)
```

```
[1] 0.3327317
```

```
cat("MSE Value:",rf.MSE, "\n")
```

```
MSE Value: 0.3327317
```

The test Mean Squared Error (MSE) value of 0.3327317 obtained from the Random Forest model indicates a somewhat better fit compared to the previous MSE of 0.3762034. A lower MSE suggests that the model's predictions are closer to the actual values in the dataset, reflecting improved accuracy. While the difference in MSE values may appear small, even slight improvements in predictive performance can have significant implications, particularly in applications where precision is crucial. Therefore, the decrease in MSE from 0.3762034 to 0.3327317 suggests that the Random Forest model offers a slightly better predictive performance, contributing to a more reliable and accurate model for predicting concrete compressive strength.

## Random Forest cross-validation.

In this section, we are conducting a cross-validation procedure to assess the predictive performance of a Random Forest model for predicting concrete compressive strength. The process involves iterating through 10 random splits of the dataset into training and testing sets. Within each iteration, we set a random seed to ensure reproducibility and randomly sample 80% of the observations for training while retaining the remaining 20% for testing.

To prepare the data for modeling, we scale the numeric variables to have zero mean and unit variance using min-max scaling. Subsequently, we fit a Random Forest model to the scaled training data, specifying the response variable "Concrete_Strength" and using all other variables as predictors. After making predictions on the scaled test data, we compute the Mean Squared Error (MSE) as a measure of prediction accuracy. This process is repeated for each iteration, resulting in a vector of MSE values. Finally, we calculate the average test MSE across all iterations to evaluate the overall performance of the Random Forest model in predicting concrete compressive strength.

```r
MSE_vector = rep(0,10)
for (i in 1:10) {
  set.seed(i)
  sample <- sample(1:nrow(strength), 0.8 * nrow(strength))
  train_data <- strength[sample, ]
  test_data <- strength[-sample, ]

  #scaling numeric variables to have zero mean and unit variance (min-max scaling)
  numeric_cols <- sapply(train_data, is.numeric)
  train_data_scaled <- as.data.frame(scale(train_data[, numeric_cols]))
  test_data_scaled <- as.data.frame(scale(test_data[, numeric_cols]))

  strength.rf = randomForest(Concrete_Strength ~.,
                             data = train_data_scaled,
                             mtry = (ncol(strength)-1) / 2, importance = TRUE)

  yhat.rf = predict(strength.rf, newdata = test_data_scaled)
  MSE_vector[i] = mean((yhat.rf - test_data_scaled$Concrete_Strength)^2)
}

cat("MSE Values:", MSE_vector, "\n")
```

```
MSE Values: 0.1447107 0.1831991 0.09900091 0.1162509 0.1158998 0.09216117 0.09068783 0.15072
```

```r
cat("Average Test MSE:", mean(MSE_vector), "\n")
```

```
Average Test MSE: 0.1238416
```

The results of the cross-validation procedure reveal a range of Mean Squared Error (MSE) values across the 10 iterations. The MSE values range from 0.09089059 to 0.1911249, indicating variability in the predictive performance of the Random Forest model across different training and testing splits of the dataset.

Despite this variability, the average test MSE, computed as 0.1268866, provides an overall assessment of the model's predictive accuracy. This relatively low average test MSE suggests that the Random Forest model generally performs well in predicting concrete compressive strength.Notably, when compared to the MSE values obtained from a linear regression model under the same test conditions, the Random Forest model yielded lower average test MSE of 0.1268866. This difference highlights the Random Forest's superior predictive accuracy between the two models.

### Show variable importance

The last step we want to take is to display the variable importance using the following code:

```
importance(rf.strength)
```

```
                    %IncMSE IncNodePurity
Cement             81.20282     53826.228
Blast_Furnace_Slag  46.91444     13476.060
Fly_Ash            24.42712      8350.622
Water              41.75121     28051.190
Superplasticizer   34.53409     17647.932
Coarse_Aggregate   30.97528      9315.913
Fine_Aggregate     44.66182     12153.586
Age               155.52853     73363.592
```

From the table above,Cement seems to be the most influential variable, with an importance score of 81.20282. This suggests that the amount of cement in the concrete mixture has a substantial impact on its compressive strength, highlighting the pivotal role of cement as the binder material in concrete. Following Cement, Blast Furnace Slag and Fly Ash also exhibit notable importance scores of 46.91444 and 24.42712, respectively.

These supplementary materials contribute to enhancing the properties of concrete, such as workability and durability, further underscoring their significance in determining concrete strength. Additionally, Water and Superplasticizer demonstrate moderate importance scores of 41.75121 and 34.53409, emphasizing the crucial role of water-cement ratio and chemical admixtures in optimizing concrete performance. Moreover, Coarse Aggregate and Fine Aggregate exhibit importance scores of 30.97528 and 44.66182, respectively, highlighting their contributions to the overall structure and stability of concrete mixes. Finally, Age emerges as a critical factor, with an importance score of 155.52853, indicating that the curing duration significantly influences concrete strength, with longer curing periods generally associated with higher compressive strengths.

# Conclusion

## What we think about the models and our conclusion.

In conclusion, our exploration of linear regression and random forest models for predicting concrete compressive strength has provided valuable insights into their respective strengths and limitations. The linear regression model offers simplicity and interpretability, making it suitable for understanding the linear relationships between predictor variables and the response variable. By analyzing the coefficients, we can discern the quantitative impact of each predictor on concrete strength. However, the linear regression model assumes a linear relationship between variables, which may not capture the complex, nonlinear interactions present in the data.

On the other hand, the random forest model offers flexibility and robustness, capable of capturing nonlinear relationships and interactions between variables. By aggregating the predictions of multiple decision trees, the random forest model can handle complex data structures and provide accurate predictions. Additionally, the variable importance analysis provided by the random forest model allows us to identify the most influential predictors in determining concrete compressive strength.

Through 10 iterations cross-validation analysis, we calculated the average Mean Squared Error (MSE) for both the Linear Regression and Random Forest models. The average MSE for Linear Regression was found to be 0.3762034, while for Random Forest, it was 0.1258866. Comparing these results, it is evident that the Random Forest model consistently exhibited lower MSE values across multiple iterations. This suggests that the Random Forest model is more accurate in predicting concrete compressive strength compared to Linear Regression. The lower average MSE of the Random Forest model indicates that it provides better predictions on average and is more robust in handling the complexities and non-linearities present in the data. Therefore, based on these findings, we conclude that the Random Forest model appears to be the most accurate among the models to predict the concrete compressive strength between this two models.