Earthquake Damage Prediction Analysis

Damage Analysis and prediction of damage caused to buildings during earthquakes

Team Members

- G V Datta Adithya, 19BCE7119
- Polu Venkata Sai Pavan Kalyan, 19BCE7414
- Karamsetty Kaushik, 19BCD7137
- Shiva Kumar Jalla, 19BCE7778

Abstract

→ Around the world, we face numerous natural calamities, and earthquakes are one of the most prominent calamities amongst them.

→ This project was developed as a means as estimating the damage that can be caused due to an incoming earthquake so that people residing in the danger zone can take steps to relocate and avoid the incoming disaster.

→ The data that we are utilizing in this project is collected on the basis of the damage that was suffered by the various buildings located in the calamity zone, and this project was made to analyze this data to develop a prediction model to estimate the damage that could be done to buildings for future earthquakes.

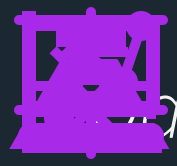
Problem Definition

→ The prediction of damage caused to buildings due to earthquakes.

→ Resolution of this problem statement allows for us to receive statistical analysis on how effective an earthquake is on a particular building and the probably of the building being unable to withstand the impact.

Data Processing Steps

- Data Cleaning
- Data Reduction
- Data Transformation
- Data Visualization
- Implementation of the Decision Tree Algorithm
- Implementation of the Naïve Bayes Algorithm



Data Preprocessing

Data Preprocessing has been done in three steps in this project, these specifically are, Data Cleaning, Data Reduction, and Data Transformation.

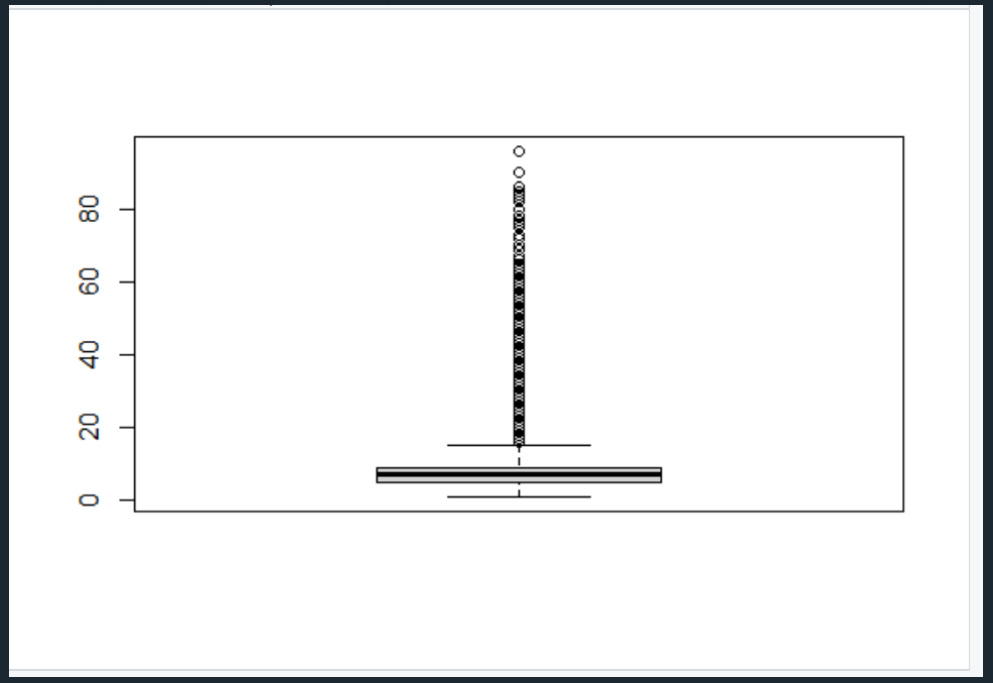
These steps have been done through the removal of NA values, which provides us with a cleaner dataset, with values for each section.

Next, we have detected and removed Outliers which allows for us to discard abnormal collections of data, in order to maintain relatively accurate data to work with.

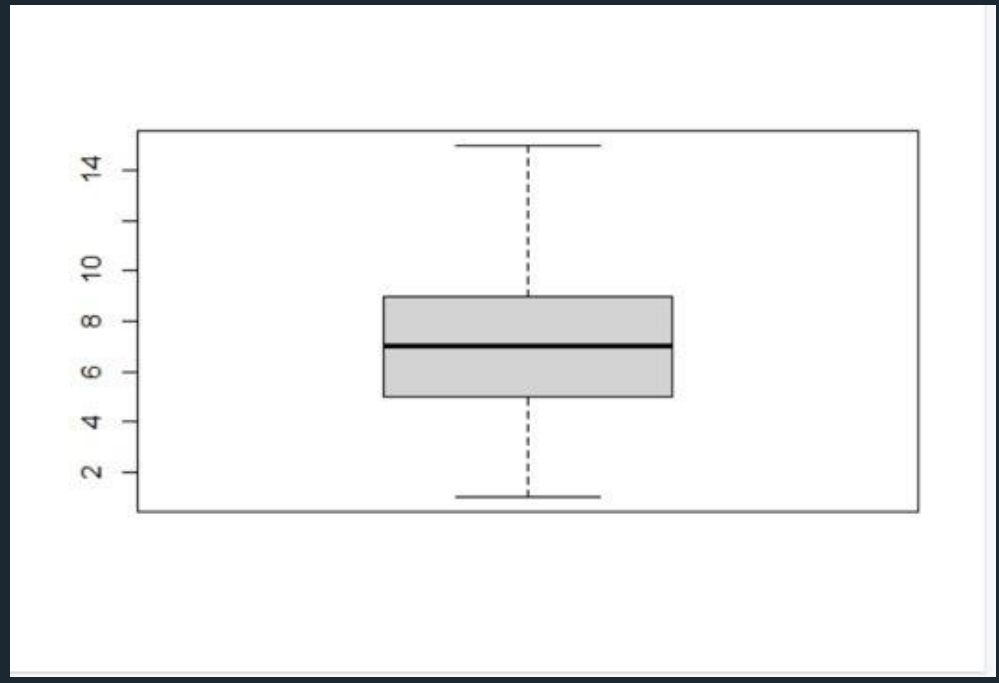
Finally, we perform checks for the various traits present for each building that was affected and attach them to the specific column.

Data Preprocessing

Before removing outliers based on "PERCENTAGE"

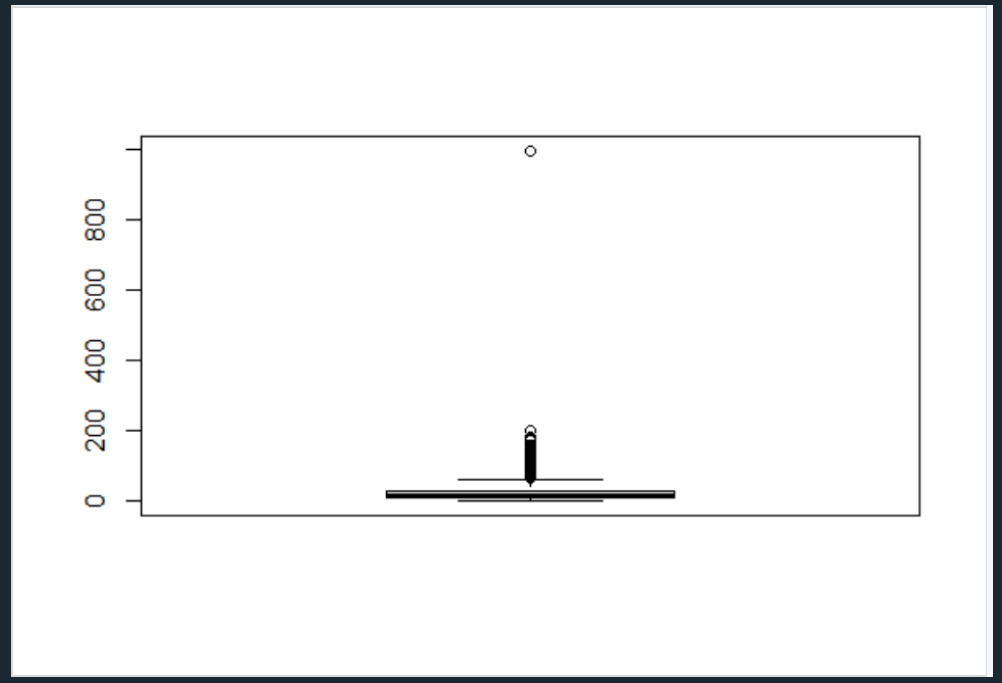


After removing outliers based on "PERCENTAGE"

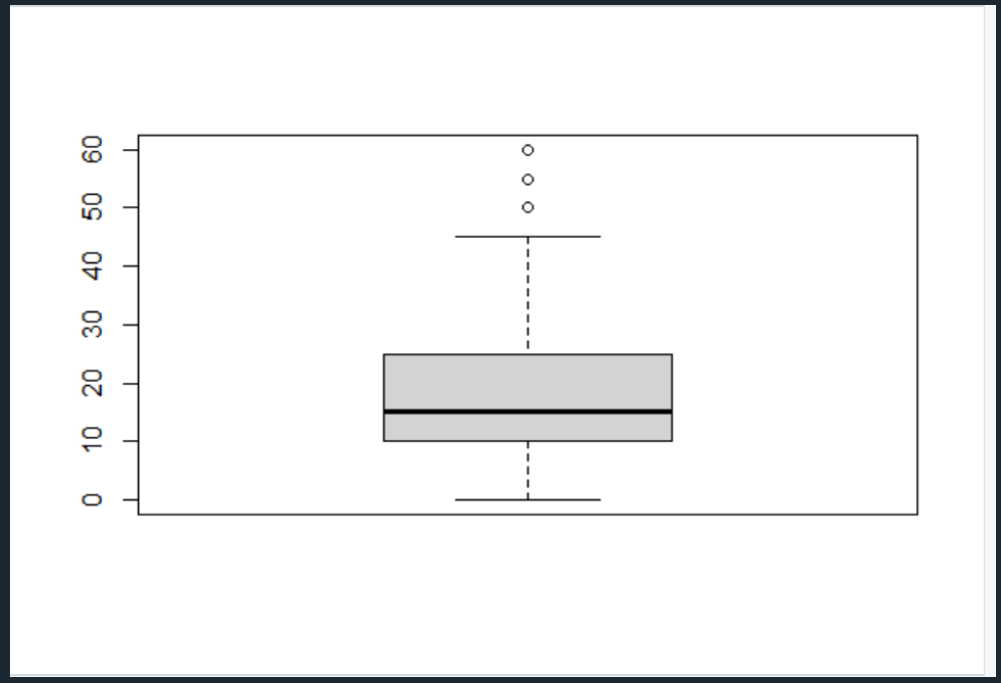


Data Preprocessing

Before Removing Outliers

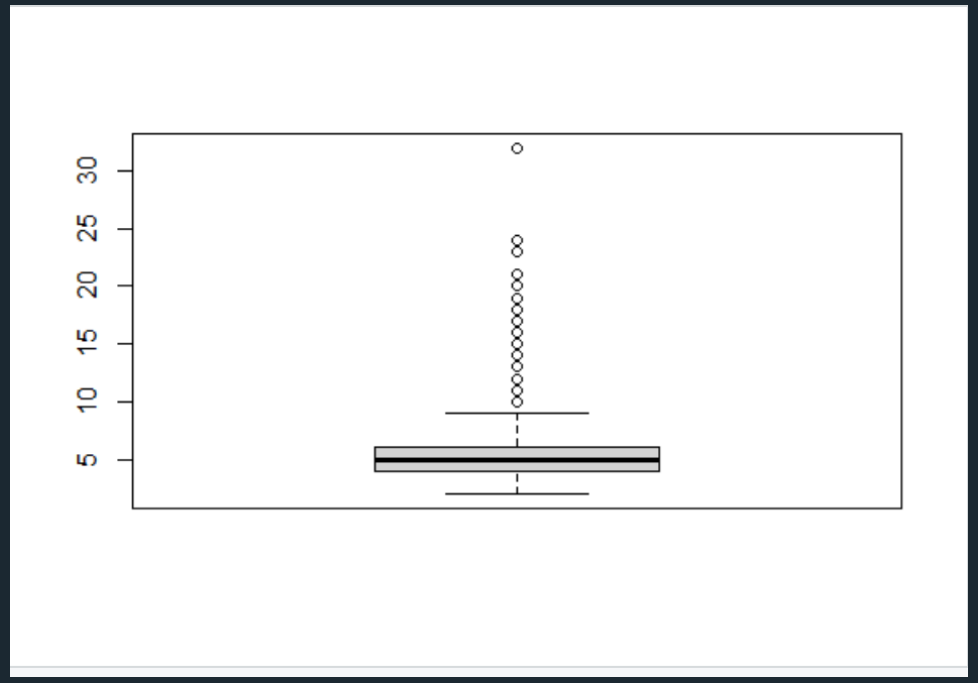


After removing outliers based on "AGE"

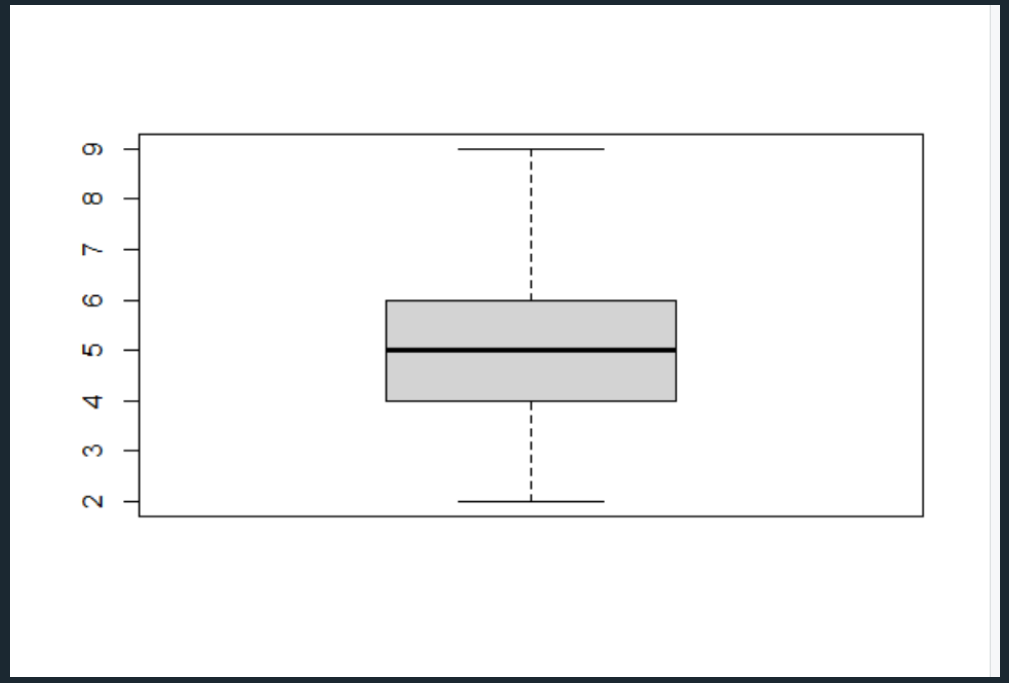


Data Preprocessing

Before removing outliers based on "HEIGHT_PERCENTAGE"



After removing outliers based on "HEIGHT_PERCENTAGE"



Data Visualization

After performing the pre-processing, the data was visualized using the ggplot2 library, for different measures.

Bar plot for the individual damage grade performance.

For the continuous column age and continuous area percentage, a dot plot was retrieved.

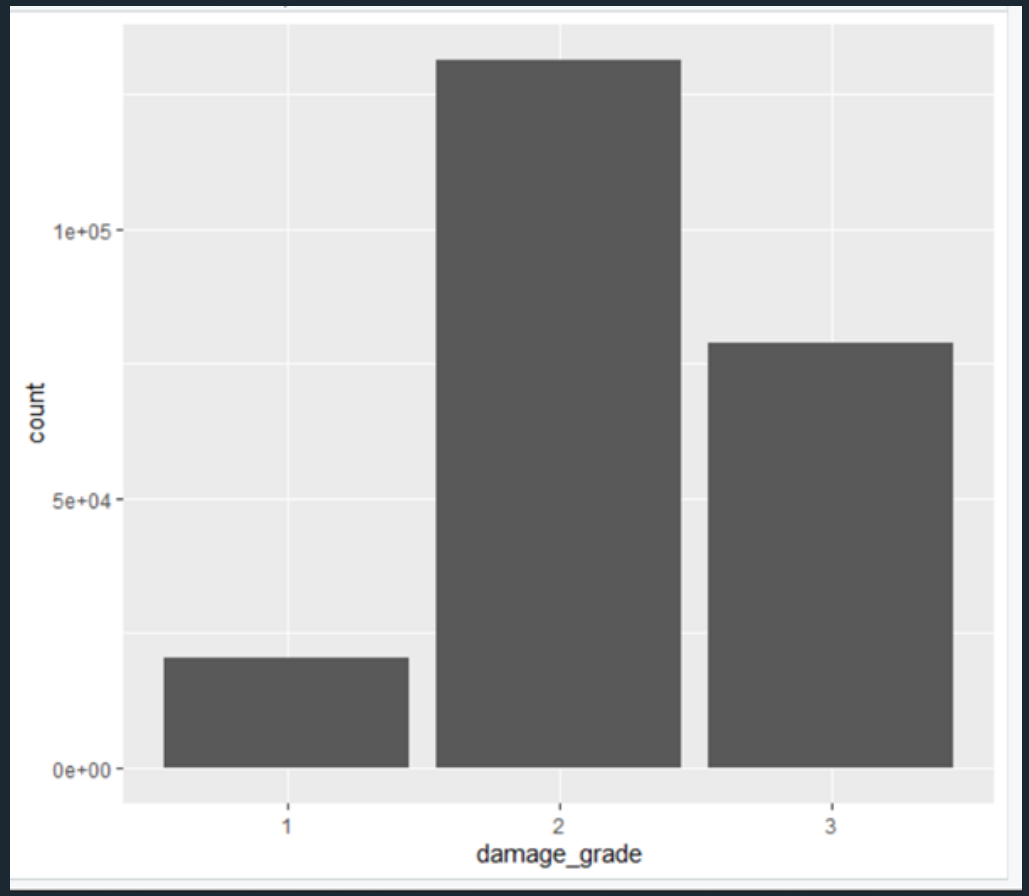
For the distribution of each attribute, a Q-Q plot was retrieved.

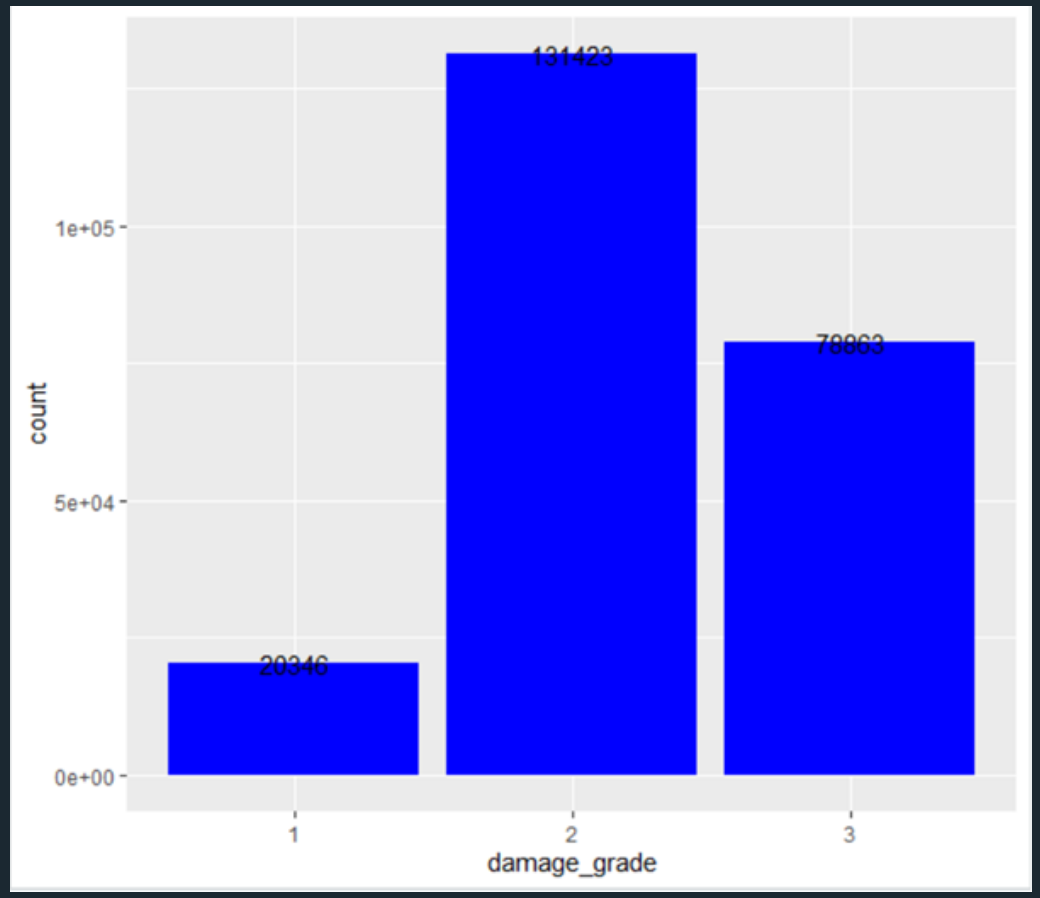
For the LAN surface condition, a density plot was retrieved.

For taking a measure on age, area_percentage, and height_percentage on damage_grade, we have set up violin plots.

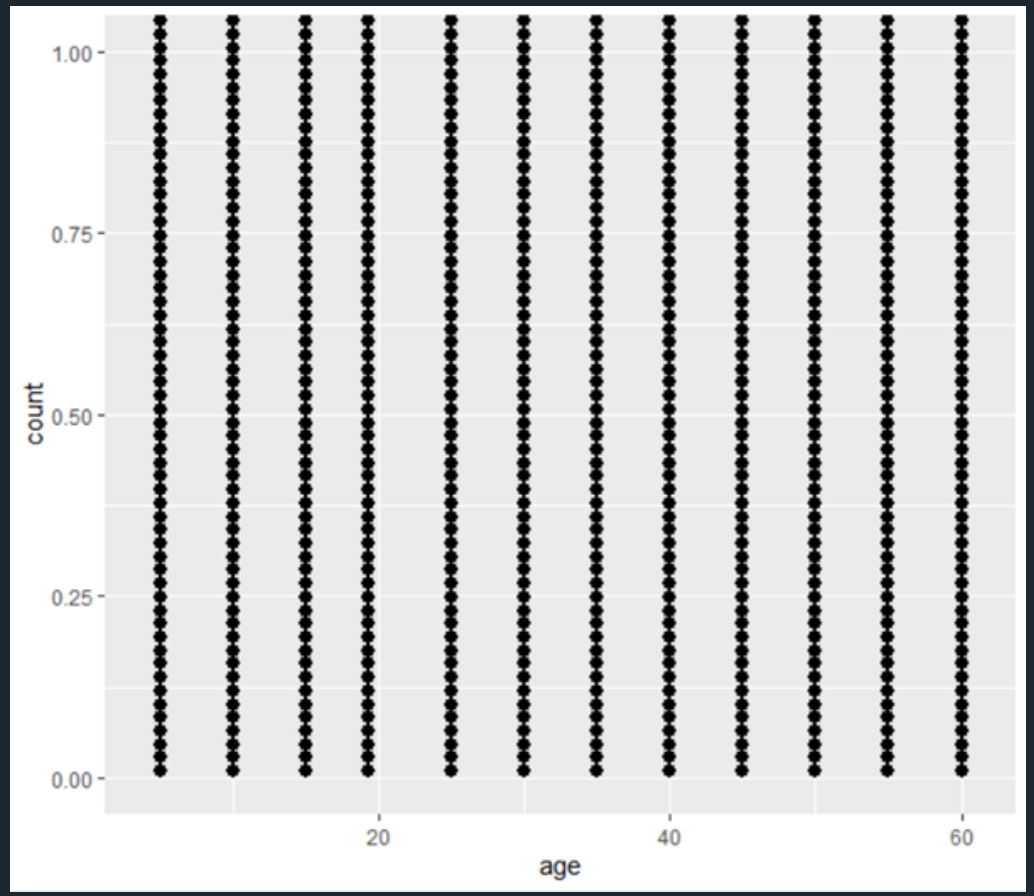
Jitterplots have been retrieved for the land surface condition and the damage grade.

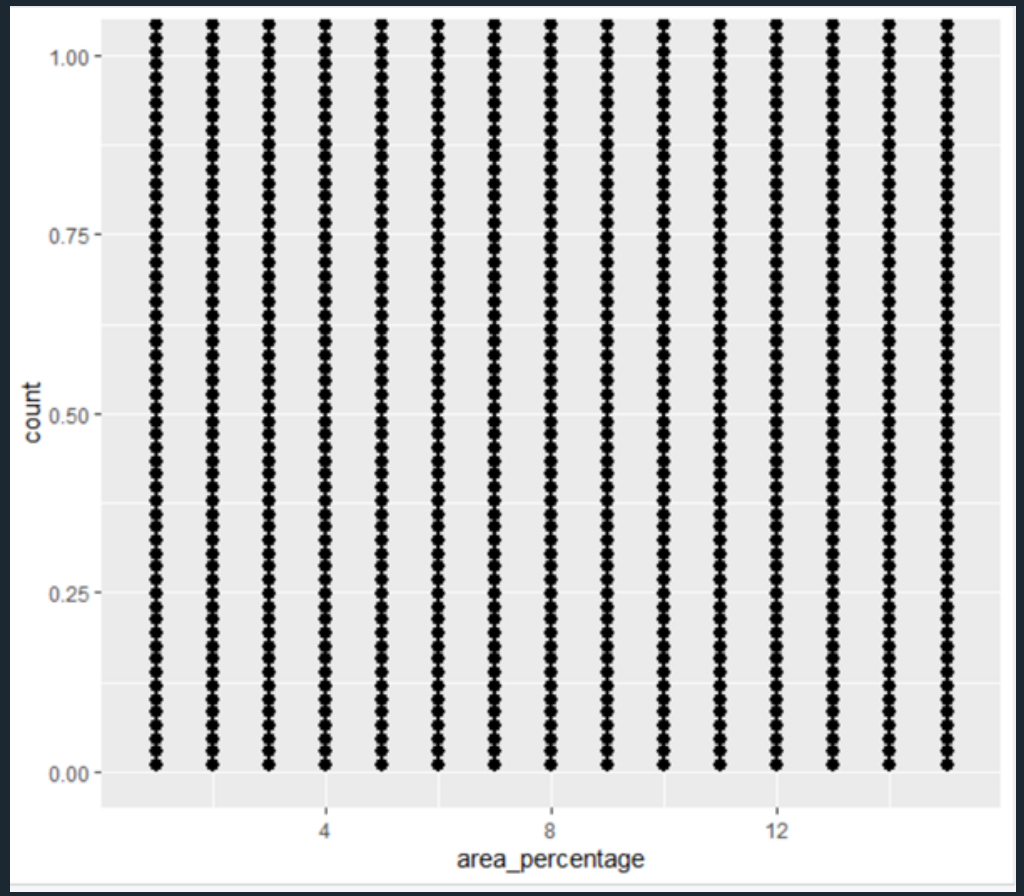
Damage Grade Performance





Continuous column age and continuous area percentage





Machine Learning Prediction Algorithms

- According to the analytics that we have gathered, we then proceed to work on ML models that can predict the results.
- Finally, apply the suitable algorithms that are required for the Data Set and Predict the Result.
- The two algorithms that we are going to utilize for the prediction model are the Decision Tree Algorithm and the Naïve Bayes Algorithm.

Testing predictions and providing the accuracy report

```
> testPred <- predict(tree, newdata = validate)
> tab1 <- table(testPred, validate$damage)
> print(tab1)

testPred     1     2     3
       1   213   124     3
       2  3760 24597 13838
       3    52  1550  1828
> accuracy1 <- 1 - (sum(diag(tab)) / sum(tab))
> print('Accuracy:')
[1] "Accuracy:"
> print(accuracy1 * 100)
[1] 42.34541
```

Decision Tree Algorithm

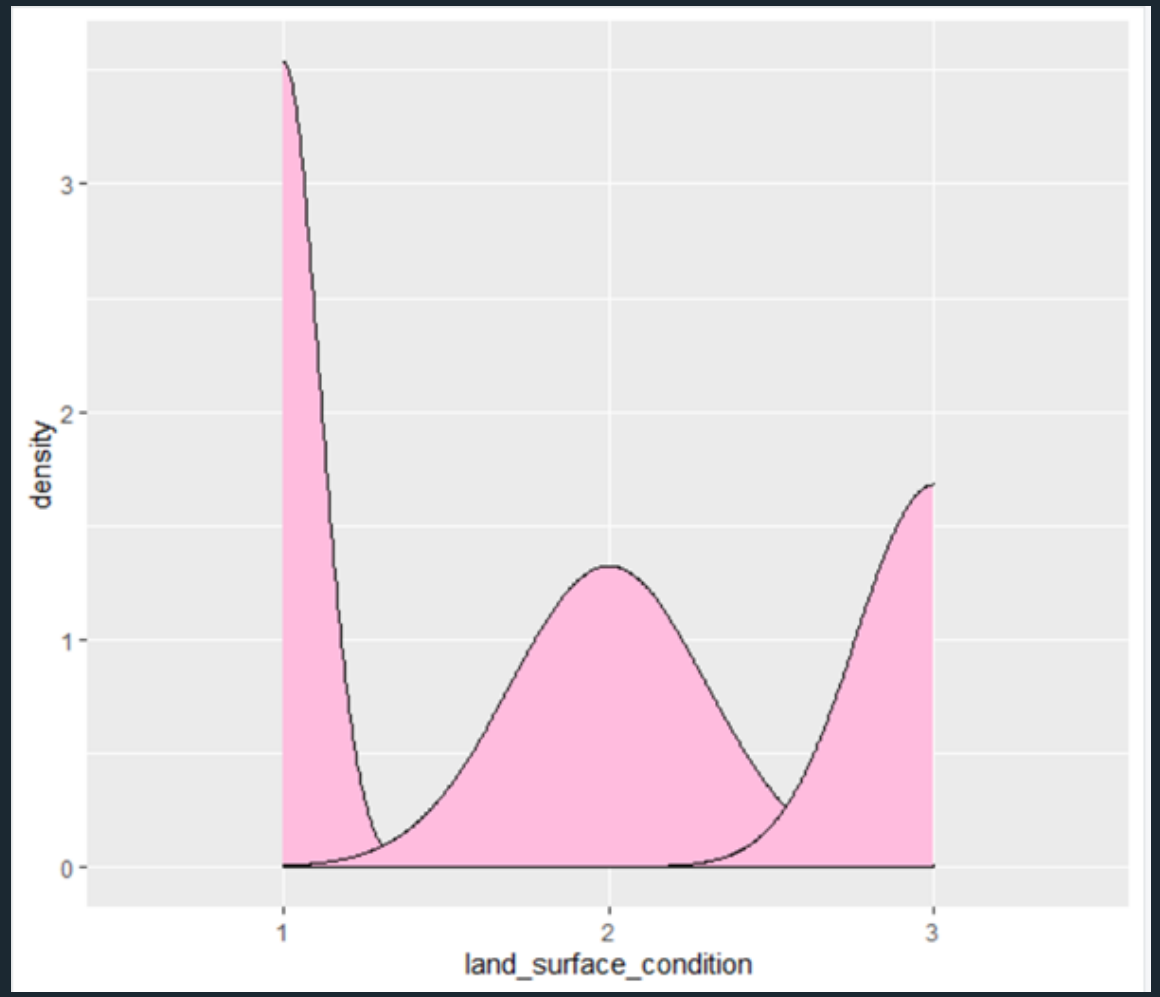
The Decision Tree Algorithm is a general, predictive modelling tool with applications spanning several different areas.

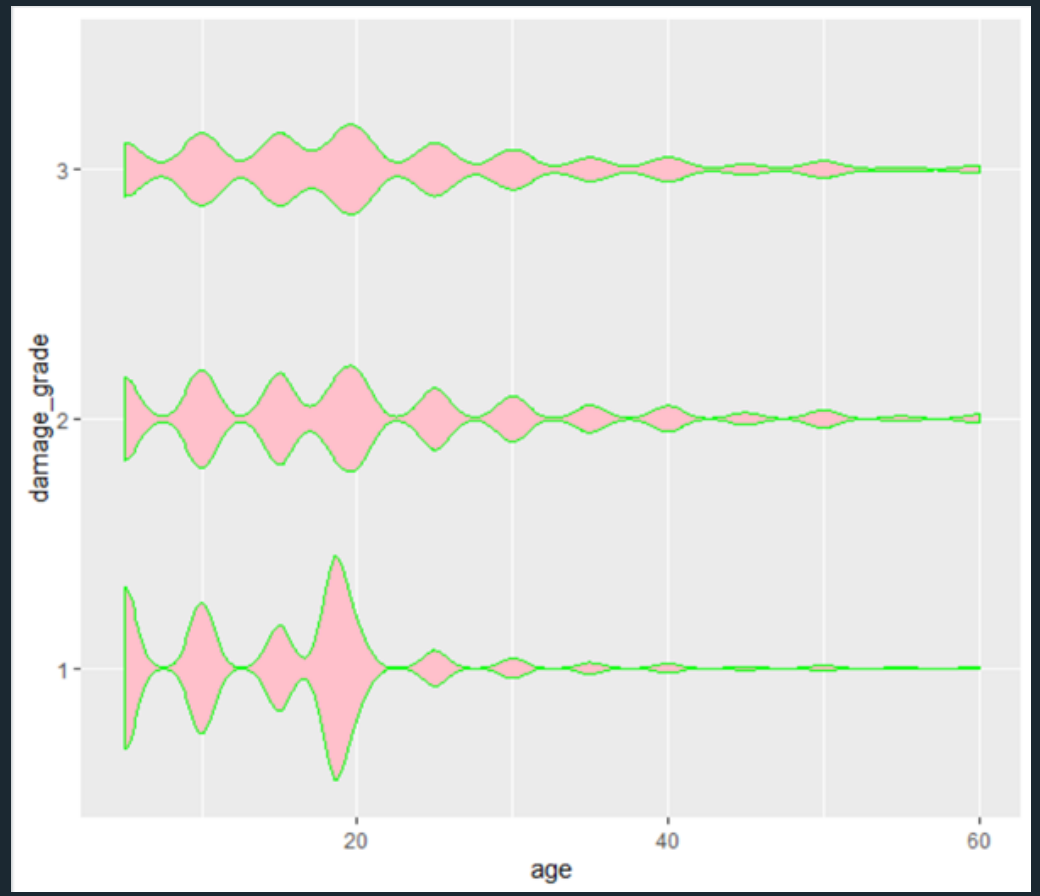
In general, decision trees are constructed via an algorithmic approach that identifies ways to split a data set based on various conditions.

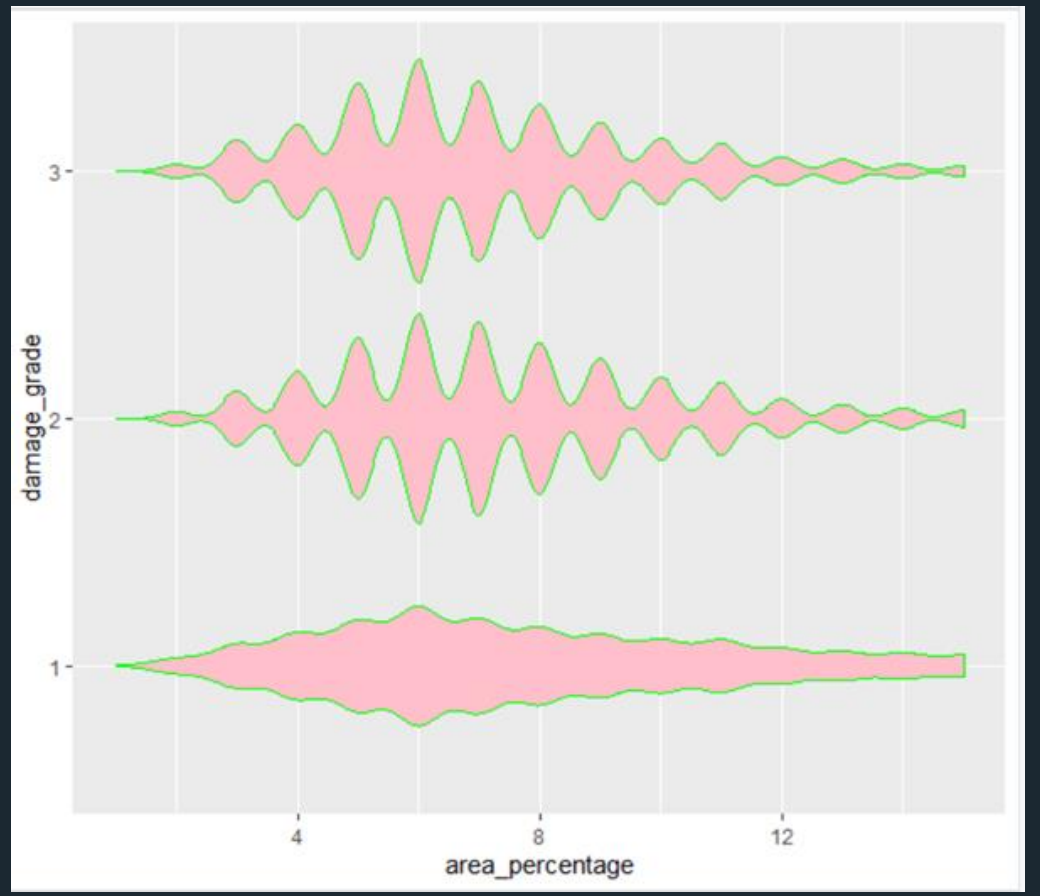
It is one of the most widely used and practical methods for supervised learning.

In this analysis, we have used the randomforest package to perform predictions.

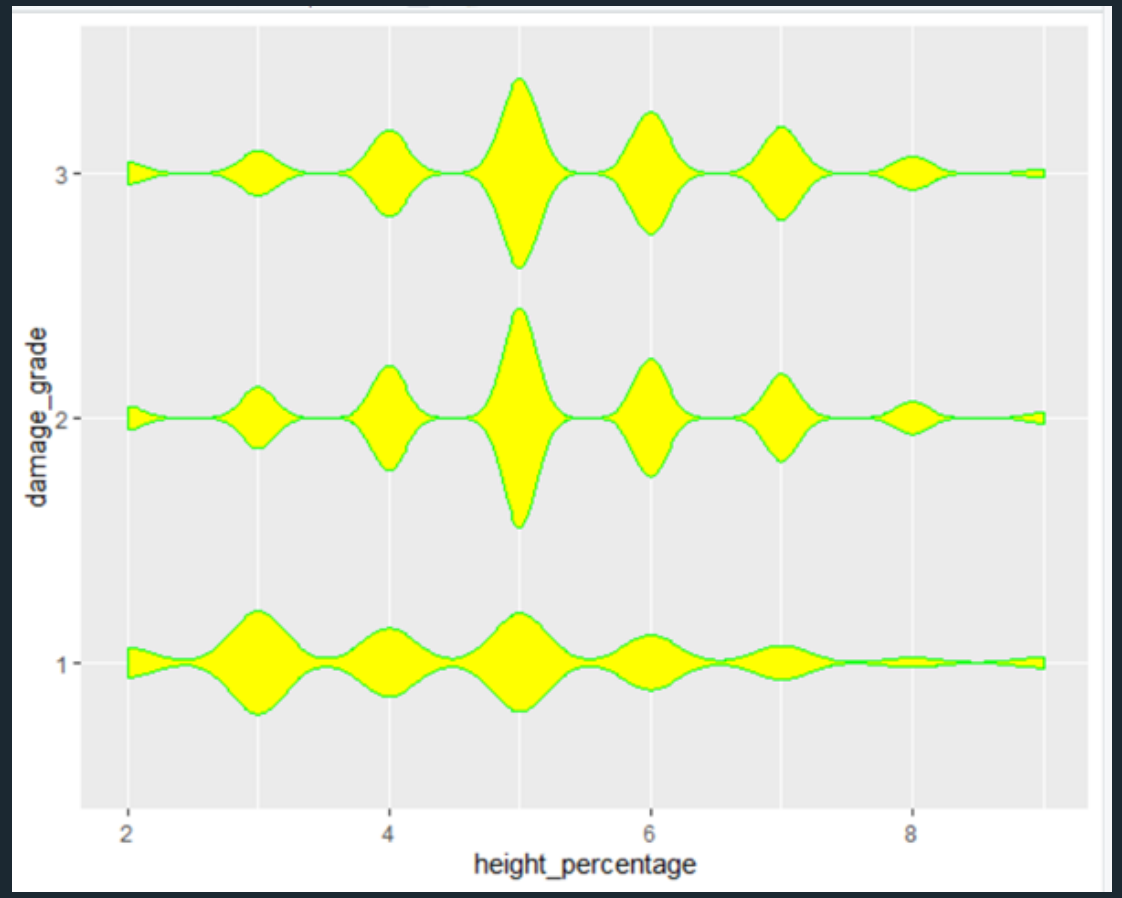
Distribution of each attribute

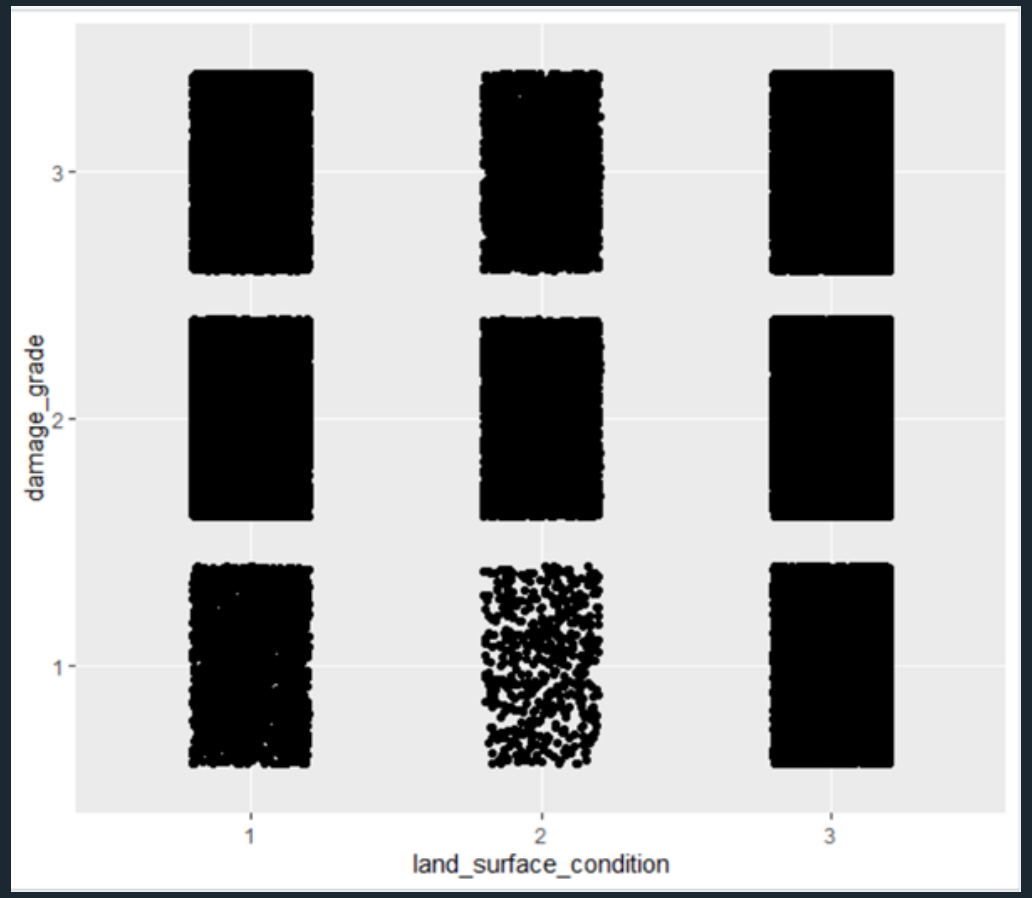


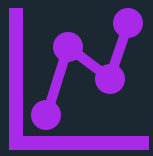

damage_grade
3
2
1
20
40
60
age


damage_grade
3
2
1
4
8
12
area_percentage

Jitter plots







Naïve Bayes Algorithm

- The Naive Bayes Algorithm is a one of the popular classification machine learning algorithms that helps to classify the data based upon the conditional probability values computation.
- It implements the Bayes theorem for the computation and used class levels represented as feature values or vectors of predictors for classification.
- Naive Bayes Algorithm is a fast algorithm for classification problems.

Naive Bayes Classifier for Discrete Predictors

```
> y_pred <- predict(classifier_cl, newdata = test_cl)
> y_pred
  [1] 3 3 3 3 3 3 3 3 3 3 3 3 1 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
 [31] 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
 [61] 3 3 3 3 3 3 1 3 3 3 1 3 3 3 3 2 3 3 3 3 1 1 3 1 3 3 3 3 1 3
 [91] 3 3 3 3 3 1 1 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 1 3 1 3 3 3 3 3
[121] 3 3 3 3 3 1 3 3 3 2 3 3 3 3 3 3 3 3 3 1 2 3 3 3 1 3 1 3 3 3
[151] 1 3 3 3 3 3 3 3 3 3 3 3 3 1 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 1
[181] 3 3 2 1 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 2 3 3 3 3
[211] 3 3 1 3 3 3 3 3 3 1 3 3 3 1 3 3 3 3 1 3 3 3 3 3 3 3 3 1 3 3
[241] 3 3 3 3 3 3 3 1 3 1 3 3 3 3 3 1 3 3 3 3 3 1 3 3 3 3 3 3 3 3
[271] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3
[301] 3 1 1 3 3 3 3 1 1 3 3 1 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 1 3 1
[331] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 3 3 1 3 3
[361] 3 3 3 3 1 3 3 3 3 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[391] 3 3 3 3 3 3 3 3 1 3 3 3 3 3 1 1 3 3 3 3 1 3 1 1 3 1 3 3 1 3 1 3 3
[421] 3 3 3 3 2 3 3 1 3 3 1 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 1 1 3 3
[451] 3 1 3 3 1 3 3 3 3 3 3 3 2 3 3 3 1 3 3 3 1 3 3 3 2 3 3 3 3 3 3 1
[481] 1 3 3 3 3 3 3 2 3 2 3 3 3 1 1 2 3 3 3 3 1 3 3 3 2 3 3 3 1 3 3
[511] 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 1 1 3 3 3 3 3 3 3 3 2
[541] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 2 3 3 3 3 3 3 3 3 3
[571] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 1 3 3 3 3 3 1 3 3 3
[601] 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[631] 3 3 3 1 1 3 3 3 3 3 3 3 3 3 2 1 3 3 1 3 3 3 3 3 3 1 2 3 3 3 3 3
[661] 3 1 1 3 3 3 3 3 1 1 3 1 3 1 3 3 1 3 3 3 3 3 3 3 1 1 3 2 3 3 3 3
[691] 3 3 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[721] 3 3 3 3 3 3 3 1 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 3 3 3
[751] 3 3 3 3 2 3 3 3 3 3 1 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[781] 3 3 1 3 3 3 3 3 3 3 1 1 3 3 3 3 3 2 3 3 3 3 1 3 3 3 3 3 3 3 3 3
[811] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 1 3 3 3 3 3 1 3 3
[841] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 1 3 3 3 3 3 3 1
[871] 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 2 3 3 3 3 3 3
```

```
  3 716.8311 407.9166

   geo_level_3_id
Y       [,1]     [,2]
  1 6299.175 3749.124
  2 6233.227 3662.836
  3 6325.302 3603.476

   count_floors_pre_eq
Y       [,1]      [,2]
  1 1.697619 0.6077631
  2 2.053708 0.5946307
  3 2.153827 0.6298165

   age
Y       [,1]      [,2]
  1 15.48892  9.005845
```

```
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
         1          2          3
0.08845063 0.56960841 0.34194096

Conditional probabilities:
   building_id
Y       [,1]     [,2]
  1 528510.9 303352.6
  2 524985.9 304760.8
  3 527450.2 304163.2

   geo_level_1_id
Y       [,1]     [,2]
```

Finding Accuracy

```
> tab1 <- table(y_pred, test_cl$damage_grade)
> print(tab1)

y_pred     1     2     3
     1  2661  4449   927
     2   147  1297   459
     3  3258 33716 22272
>
> accuracy1 <- 1 - (sum(diag(tab1)) / sum(tab1))
> print('Accuracy:')
[1] "Accuracy:"
> print(accuracy1 * 100)
[1] 62.08771
>
> length(y_pred)
[1] 69186
```

Conclusion

- → Going through the various observations that were provided in the dataset, it became very evident that there was a pattern of repetition in the details for a few particular buildings.
- → Studying and parsing through the datasets, the patterns grew more evident and as a result, we decided that it would be a good idea to work on forming a prediction model using effective prediction algorithms such as the Decision Tree Algorithm and the Naive Bayes Algorithm.

- → The models were developed to find the accuracy from test datasets.
- → Overall, the prediction model was a success and this project can be used as a means to estimate the impact of any future earthquake, to assess the damage that could be caused, and take precautionary measures further ahead of time.

References

→ This project is currently being hosted at, https://github.com/dat-adi/earthquake-analysis

→ Dataset for the earthquake analysis was taken from Kaggle, https://www.kaggle.com/mullerismail/richters-predictor-modeling-earthquake-damage/activity

→ The link to the competition conducted for the analysis of this data, https://www.drivendata.org/competitions/57/nepal-earthquake/

→ An article on how to predict the damage to a building in Python, https://medium.com/swlh/predicting-damage-to-building-due-to-earthquake-using-data-science-e85a62adc0c0

Thank you!