Data Mining and Machine Learning Portfolio

SAKET ABHAYKUMAR KULKARNI
School of Computing
National College of Ireland
Dublin, Ireland
x23102381@student.ncirl.ie

***Abstract*—The performance of five different machine-learning techniques—Naive Bayes, K-means Clustering, Linear Regression, Decision Tree, and Logistic Regression—across three different datasets—taxi fares, student placement, and car prices—is carefully explored in this extensive study. This project follows the CRISP-DM methodology in data mining.**

Naive Bayes is a probabilistic classification technique used to predict outcomes in student placement. To find patterns in taxi fare data, K-means Clustering is utilized. By analysing the relationships between variables, Linear Regression provides insight into the factors that affect taxi prices. Because of their interpretability, decision tree algorithms give useful insights into the classification of the type pf fuel used by a car. The classification technique of logistic regression evaluates the probability of student placement success. Performance evaluation includes Sensitivity/Specificity, MAPE, F1 score, and Accuracy along with standard metrics. Deepness, clarity, originality, potential impact, technical proficiency, and reproducibility are among the evaluation criteria. This work presents important implications for researchers and practitioners equally by providing a thorough understanding of how various algorithms can be strategically applied to real-world datasets, in addition to providing deep insights into the machine learning performance.

I. INTRODUCTION

This portfolio is built to provide the analysis of various data mining and machine learning methods that will be implemented on different types of data sets. The objective of the project is to evaluate the performance of the machine learning algorithms to see what kind of accuracy is produced. The data sets picked are an Automobile pricing data set which includes information about the cars and their features along with a price set on the car. The next data set is a Student placement data set containing information about the students along with various grades and certificates they achieved and all of this leads to the student's placement in a company which is either a yes or a no. The third and final data set contains information about taxi fares and its respective variables that affect it like the distance or the duration of the taxi ride. All the data sets are diverse and provide a good amount of information to perform data mining and machine learning algorithms on them.

A. Machine Learning Algorithms

1) Naive Bayes: The statistical classification algorithm Naive Bayes relies on the assumption of feature independence. Calculates the probability of a student getting placed or not based on the variables affecting it.

2) K-Means Clustering: K-Means Clustering is an unsupervised learning technique that splits data into K clusters according to similarity. It is useful for data segmentation and pattern detection since it places each data point in the cluster whose mean is closest. Here, it is used to categorise different types of taxi rides and groups them into clusters.

3) Linear regression: Linear Regression is a supervised method used to find the relationship between an independent and a dependent variable. Here it is used to find the variable affecting taxi fares the most.

4) Logistic Regression: Logistic Regression is a classification algorithm used to classify the students who got placed and who didn't, using that model, it predicts the probability of getting placed or not on test data.

5) Decision Tree: A flexible approach for classification and regression applications is the decision tree. It creates a tree-like structure by recursively splitting data based on features and then traversing the branches to make decisions. The algorithm is used to predict the classification of fuel type for the automobile information provided.

The project revolves around the performance of machine learning algorithms, all the code is written using R programming language, one of the best programming languages for statistical analysis and data mining. The objective of the project is to get a decent variance in the accuracy of performance by the machine learning algorithms. Some algorithms don't perform optimally with certain data sets, it might not be of high accuracy. In that case, it is justifiable as to why that model did not perform and why it is not a good fit for the data. Analysis is done on the price of taxi fares, types of taxi rides, Placement status of students whether they get placed or not as there are a lot of factors affecting the placement and finally to categorise cars based on the fuel type of the car by considering all the variables.

The report contains a step-by-step procedure followed to perform all the machine learning algorithms to be able to mine the data thoroughly. The discussion of Methodologies will be explained after the brief on the data sets is given. Finally, the evaluation process takes place where each algorithm and its performance are evaluated individually to gain insights from the data as well as track the accuracy of the algorithm.

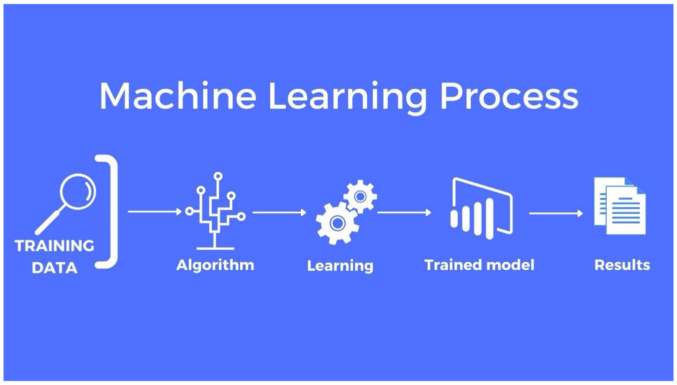


Figure 1 – Stages of Machine Learning

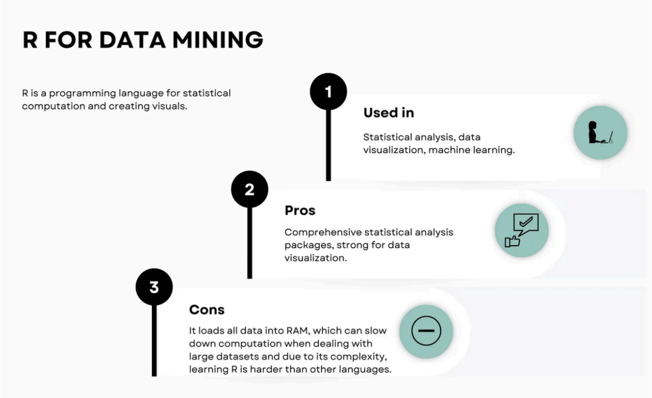
II. RELATED WORK

Data is essential in today's world of decision-making, and tools for analyzing data are like building bricks for those who work with data. These technologies, such as Python or Oracle BI, assist in making predictions and creating visual representations of information in addition to analysing data and using it to train computers. Many of these tools are useful even if you're not a programming specialist. Let's now discuss the operation of data mining. There are several steps to it. To use the information in the next phases, you must first gather it from various sources, such as organized or disorganized data. After that, you clean up this data to make it better, kind of like organizing a messy space. After cleaning up, you use mathematical and computer algorithms to identify patterns in the data. This process is similar to identifying shapes in clouds or determining what typically goes together. It is useful for tasks like grouping information, which is useful when making decisions about products or understanding connections. Next, you examine the patterns you have identified and attempt to understand them. This process is similar to solving a puzzle or understanding what is hidden in the data. Once you have done this, you ensure the information is credible and use it to make decisions or forecasts. For instance, a store might use this to forecast which products will sell well in the future and make plans accordingly.

A. Tool used for Data Mining

The tool used for this project is R, R is a programming language for statistical computation and creating visuals. and it is used in the R studio which can be installed in any platform. The R programming language has become popular because it has a wide range of tools that can be used to manipulate data, do calculations, and create graphics and visuals. These facilities make the language useful for statistical applications and data analysis. Since R is open source, it can be used without restriction and has a strong package collection that increases its extensibility. Major corporations such as Pfizer, Bank of America, and Shell employ R for data analysis, demonstrating its practicality in the real world. Notably, R has packages for extraction, cleaning, visualization, and the implementation of intricate statistical models, so it may be used throughout the whole data mining process.

Important libraries such as caret make it easier to apply machine learning models, ggplot2 excels at data visualization, and dplyr helps with data cleaning and manipulation, and all of these libraries have been used for this project While R has many benefits, like a robust package repository and excellent plotting capabilities, it has memory management issues because it loads all data into RAM, which can slow down computation for large datasets which explains the delay in running the decision tree algorithm. Moreover, compared to Python, R has a steeper learning curve, especially for rookies. Even with all of its complexity, R is still a strong and popular language for data analysis.


R FOR DATA MINING

R is a programming language for statistical computation and creating visuals.

1 Used in

Statistical analysis, data visualization, machine learning.

2 Pros

Comprehensive statistical analysis packages, strong for data visualization.

3 Cons

It loads all data into RAM, which can slow down computation when dealing with large datasets and due to its complexity, learning R is harder than other languages.

Figure 2 – Pros and Cons of R programming language

III. DATA

This section is about the data sets picked for the machine learning algorithms.

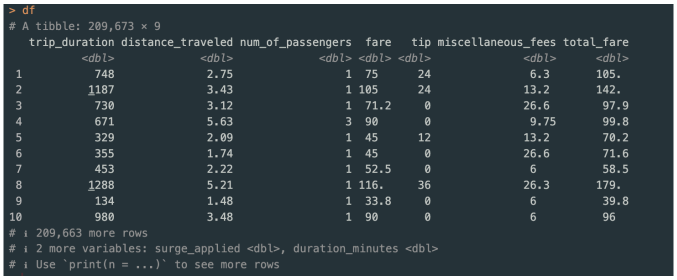
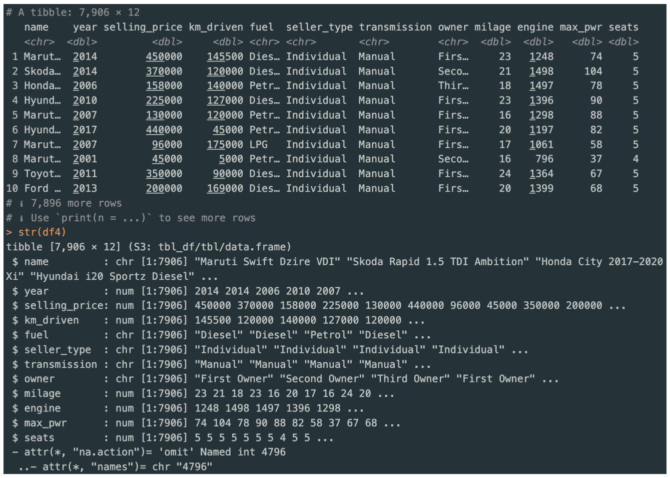
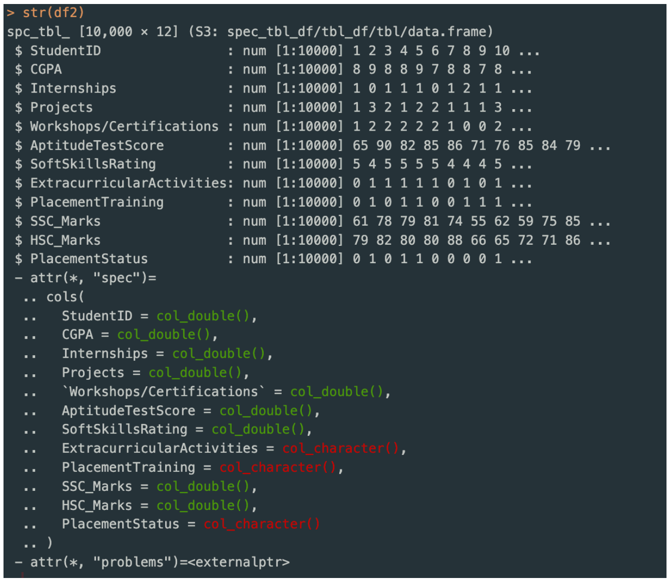

```
> df
# A tibble: 209,673 × 9
   trip_duration distance_traveled num_of_passengers  fare   tip miscellaneous_fees total_fare
           <dbl>             <dbl>             <dbl> <dbl> <dbl>              <dbl>      <dbl>
 1           748              2.75                 1  75      24               6.3       105.
 2          1187              3.43                 1 105      24              13.2       142.
 3           730              3.12                 1  71.2     0              26.6        97.9
 4           671              5.63                 3  90       0               9.75       99.8
 5           329              2.09                 1  45      12              13.2        70.2
 6           355              1.74                 1  45       0              26.6        71.6
 7           453              2.22                 1  52.5     0               6          58.5
 8          1288              5.21                 1 116.     36              26.3       179.
 9           134              1.48                 1  33.8     0               6          39.8
10           980              3.48                 1  90       0               6          96
# ℹ 209,663 more rows
# ℹ 2 more variables: surge_applied <dbl>, duration_minutes <dbl>
# ℹ Use `print(n = ...)` to see more rows
```

Figure 3 – Taxi fares dataset information and structure which is used for making the k-means and linear regression models.



```
# A tibble: 7,906 × 12
   name    year selling_price km_driven fuel  seller_type transmission owner milage engine max_pwr seats
   <chr>  <dbl>         <dbl>     <dbl> <chr> <chr>       <chr>        <chr>  <dbl>  <dbl>   <dbl> <dbl>
 1 Marut…  2014        450000    145500 Dies… Individual  Manual       Firs…     23   1248      74     5
 2 Skoda…  2014        370000    120000 Dies… Individual  Manual       Seco…     21   1498     104     5
 3 Honda…  2006        158000    140000 Petr… Individual  Manual       Thir…     18   1497      78     5
 4 Hyund…  2010        225000    127000 Dies… Individual  Manual       Firs…     23   1396      90     5
 5 Marut…  2007        130000    120000 Petr… Individual  Manual       Firs…     16   1298      88     5
 6 Hyund…  2017        440000     45000 Petr… Individual  Manual       Firs…     20   1197      82     5
 7 Marut…  2007         96000    175000 LPG   Individual  Manual       Firs…     17   1061      58     5
 8 Marut…  2001         45000      5000 Petr… Individual  Manual       Seco…     16    796      37     4
 9 Toyot…  2011        350000     90000 Dies… Individual  Manual       Firs…     24   1364      67     5
10 Ford …  2013        200000    169000 Dies… Individual  Manual       Firs…     20   1399      68     5
# ℹ 7,896 more rows
# ℹ Use `print(n = ...)` to see more rows
> str(df4)
tibble [7,906 × 12] (S3: tbl_df/tbl/data.frame)
 $ name         : chr [1:7906] "Maruti Swift Dzire VDI" "Skoda Rapid 1.5 TDI Ambition" "Honda City 2017-2020 
Xi" "Hyundai i20 Sportz Diesel" ...
 $ year         : num [1:7906] 2014 2014 2006 2010 2007 ...
 $ selling_price: num [1:7906] 450000 370000 158000 225000 130000 440000 96000 45000 350000 200000 ...
 $ km_driven    : num [1:7906] 145500 120000 140000 127000 120000 ...
 $ fuel         : chr [1:7906] "Diesel" "Diesel" "Petrol" "Diesel" ...
 $ seller_type  : chr [1:7906] "Individual" "Individual" "Individual" "Individual" ...
 $ transmission : chr [1:7906] "Manual" "Manual" "Manual" "Manual" ...
 $ owner        : chr [1:7906] "First Owner" "Second Owner" "Third Owner" "First Owner" ...
 $ milage       : num [1:7906] 23 21 18 23 16 20 17 16 24 20 ...
 $ engine       : num [1:7906] 1248 1498 1497 1396 1298 ...
 $ max_pwr      : num [1:7906] 74 104 78 90 88 82 58 37 67 68 ...
 $ seats        : num [1:7906] 5 5 5 5 5 5 5 4 5 5 ...
 - attr(*, "na.action")= 'omit' Named int 4796
  ..- attr(*, "names")= chr "4796"
```

Figure 4 – Automobile prices data used for decision tree.



```
> str(df2)
spc_tbl_ [10,000 × 12] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ StudentID                : num [1:10000] 1 2 3 4 5 6 7 8 9 10 ...
 $ CGPA                     : num [1:10000] 8 9 8 8 9 7 8 8 7 8 ...
 $ Internships              : num [1:10000] 1 0 1 1 1 0 1 2 1 1 ...
 $ Projects                 : num [1:10000] 1 3 2 1 2 2 1 1 1 3 ...
 $ Workshops/Certifications : num [1:10000] 1 2 2 2 2 2 1 0 0 2 ...
 $ AptitudeTestScore        : num [1:10000] 65 90 82 85 86 71 76 85 84 79 ...
 $ SoftSkillsRating         : num [1:10000] 5 4 5 5 5 5 4 4 4 5 ...
 $ ExtracurricularActivities: num [1:10000] 0 1 1 1 1 1 0 1 0 1 ...
 $ PlacementTraining        : num [1:10000] 0 1 0 1 1 0 0 1 1 1 ...
 $ SSC_Marks                : num [1:10000] 61 78 79 81 74 55 62 59 75 85 ...
 $ HSC_Marks                : num [1:10000] 79 82 80 80 88 66 65 72 71 86 ...
 $ PlacementStatus          : num [1:10000] 0 1 0 1 1 0 0 0 0 1 ...
 - attr(*, "spec")=
  .. cols(
  ..   StudentID = col_double(),
  ..   CGPA = col_double(),
  ..   Internships = col_double(),
  ..   Projects = col_double(),
  ..   `Workshops/Certifications` = col_double(),
  ..   AptitudeTestScore = col_double(),
  ..   SoftSkillsRating = col_double(),
  ..   ExtracurricularActivities = col_character(),
  ..   PlacementTraining = col_character(),
  ..   SSC_Marks = col_double(),
  ..   HSC_Marks = col_double(),
  ..   PlacementStatus = col_character()
  .. )
 - attr(*, "problems")=<externalptr>
```

Figure 5 – Student Placement dataset used for Naïve Bayes and Logistic Regression.

IV. METHODOLOGIES

The methodology followed for this project is CRISP-DM which is an industry Standard Process for Data Mining. CRISP-DM [1] is a widely used framework for data mining that outlines a structured approach to planning, executing, and evaluating data mining projects. It provides a step-by-step process that can be adapted to various business domains and data mining techniques, making it a valuable tool for both beginners and experienced practitioners. It constitutes 6 stages in total which are Business Problem Understanding, Data Understanding, Data preparation, Modelling, Evaluation, and Deployment. These stages will be discussed in the below sub-sections.

A. Business Problem Understanding

Keeping the objective in mind, this first phase focuses on understanding the project's goals, objectives, and requirements from a business aspect. This understanding is then translated into a definition of the data mining problem and a preliminary plan that outlines a precise set of tasks and intended results. Each data set is observed and analysed to get an in-depth understanding of the components of the data to find the problem first after which the solution will be focused on.

B. Data Understanding

Data Understanding is a very crucial point during this process, whatever data is being handled has to be scanned and understood thoroughly to conduct analysis. This stage involves processes like pre-processing, cleaning, and checking for missing values. The preliminary data analysis is done here with some basic feature descriptions as well as exploratory data analysis.

C. Data Preparation

All the important tasks that are involved in creating the data to be ready for modelling are done in the preparation phase. The initial raw data is extracted, cleaned and processed well to make sure the variables are all formatted to fit into the model. It includes processes like cleaning, feature engineering, scaling and even transformation of the data which in this case has been performed on the data sets chosen.

D. Modelling

In this phase, The modelling techniques chosen for the project are applied according to the data sets prepared, and the process starts by fitting the data into the model finding an optimal solution and evaluating performance. Typically, there are several techniques for the same data mining problem type. Some techniques have specific requirements for the format of data it needs. Sometimes it is necessary to back a step into the data preparation phase and make some more changes accordingly until the data is ready.

E. Evaluation

One of the most important stages in the course of data analysis is the evaluation phase. The machine learning models are all developed and ready to run on the data. In this stage, the data has to be completely processed and ready for fitting. The data must be split into a training set and a testing set where the model trains the first set and the second set that is untouched will be used to test the trained model. Even if a model is working well with the training data, for it to meet its goals, it must be tested and evaluated on data that has not been tested. At this point, appropriate evaluation measures are assessed and thoroughly examined. Finally, the accuracy of the model is checked and evaluated.

F. Deployment

The project usually doesn't finish with the production of the model. The knowledge or insight gained from the modelling process must be provided so that the model's end users may make use of it, even if the model's primary goal is to evaluate and understand the data better. Customers, business leaders, and employees might all be considered end users. It often involves using live models in an organization's decision-making processes, such as a product recommender system, real-time personalized website, or lead scoring for marketing. The deployment phase might be as straightforward as creating a dashboard or as complicated as putting in place a repeatable data mining procedure throughout the entire company, depending on the requirements.

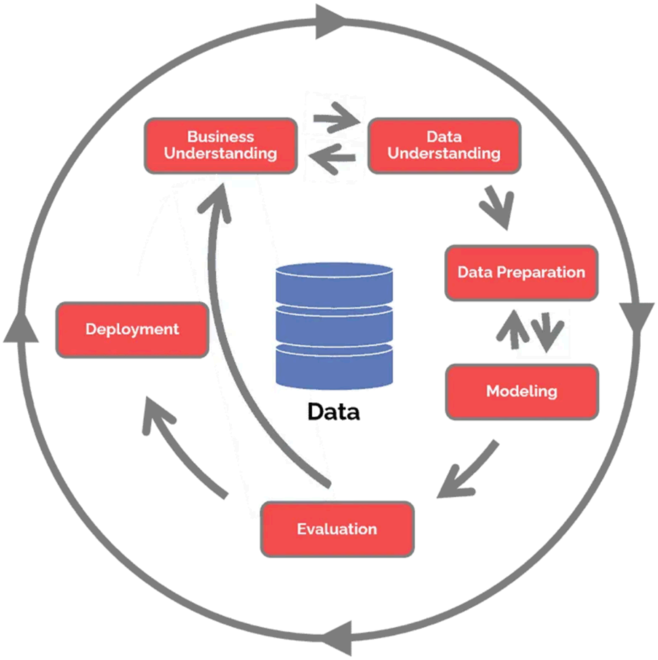


Figure 6 – Stages of CRISP-DM

V. MODEL EVALUATION AND RESULTS

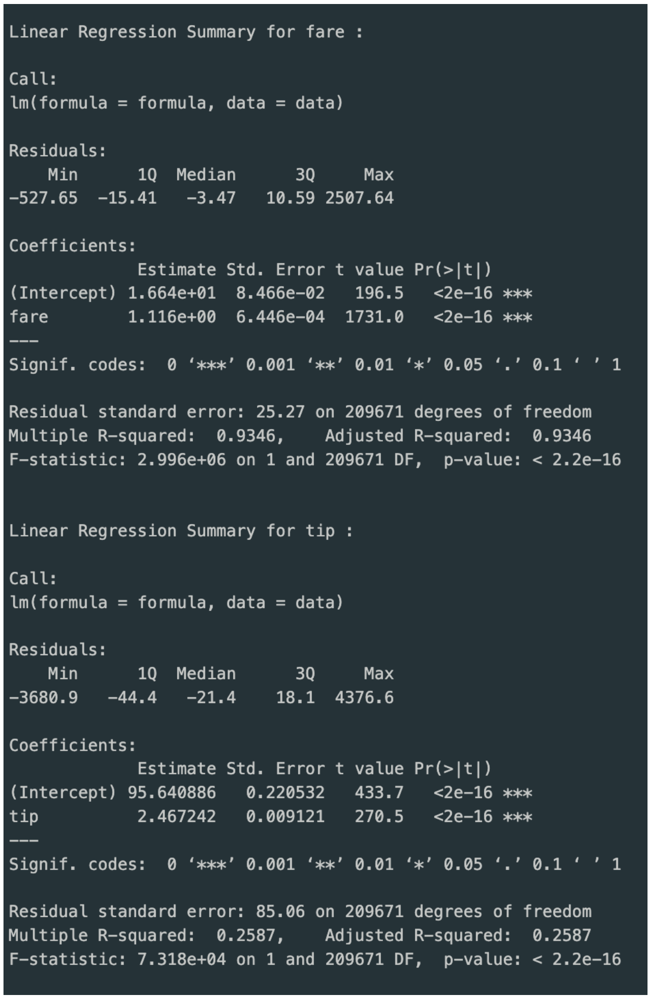
A. Model 1 - Linear Regression

This analysis reviews a taxi fare dataset using the CRISPDM approach. The dataset, which is displayed as a table, includes information on the following: total fare, surge application, number of passengers, fare, tip, and other costs and includes duration in minutes. The linear Regression analysis is performed to find insights on the variables that affect the total fare price of the taxi and how significant each variable is in terms of affecting the total fare price. Linear regression compares each variable with the target variable and gives a result where the significance is determined.

Removal of missing values and numeric variable normalizing are steps in the initial data-cleaning process. The distribution of each variable is then revealed via a summary of statistics. A summary of the dataset's central tendencies, dispersions, and possible outliers is given by these statistics.

The regression summary that is produced displays coefficients, standard errors, t-values, and p-values, providing information about the importance and effect of each variable on the total taxi fare.

After dividing the dataset into training and testing sets, the training data is used to build a linear regression model. Each independent variable's coefficients and related data are displayed in the model summary. Using the testing set, the model's predictive performance is evaluated, and the Mean Absolute Error (MAE) is computed. The model is said to be effective, which is proved by the Mean Absolute Error (MAE) of 0.4171716 which would be around 41 per cent, which shows a relatively low error in forecasting total fares.



```
Linear Regression Summary for fare :

Call:
lm(formula = formula, data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-527.65  -15.41   -3.47   10.59 2507.64 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1.664e+01  8.466e-02   196.5   <2e-16 ***
fare        1.116e+00  6.446e-04  1731.0   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 25.27 on 209671 degrees of freedom
Multiple R-squared:  0.9346,	Adjusted R-squared:  0.9346 
F-statistic: 2.996e+06 on 1 and 209671 DF,  p-value: < 2.2e-16


Linear Regression Summary for tip :

Call:
lm(formula = formula, data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-3680.9   -44.4   -21.4    18.1  4376.6 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 95.640886   0.220532   433.7   <2e-16 ***
tip          2.467242   0.009121   270.5   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 85.06 on 209671 degrees of freedom
Multiple R-squared:  0.2587,	Adjusted R-squared:  0.2587 
F-statistic: 7.318e+04 on 1 and 209671 DF,  p-value: < 2.2e-16
```

Figure 7 – Linear regression Results (partial)

B. Model 2 - K-Means Clustering

To improve the prediction of taxi fares, a K-Means clustering technique was applied to a large dataset that included features like the number of passengers, the distance travelled, the fare, the tip, other taxes, the total fare, the application of surge, and the length. The principal objective was to classify similar taxi rides into separate clusters to enable a more thorough understanding of the underlying trends and variances in the data. The ideal number of clusters was carefully determined by applying the elbow approach. Through evaluation of the within-cluster sum of squares (WSS) over a variety of possible cluster counts, a critical point was found when the rate of WSS reduction decreased, indicating an ideal trade-off between model complexity and explanatory power. The elbow point in this case indicated that five clusters would be best suited for covering the underlying structure of the data.

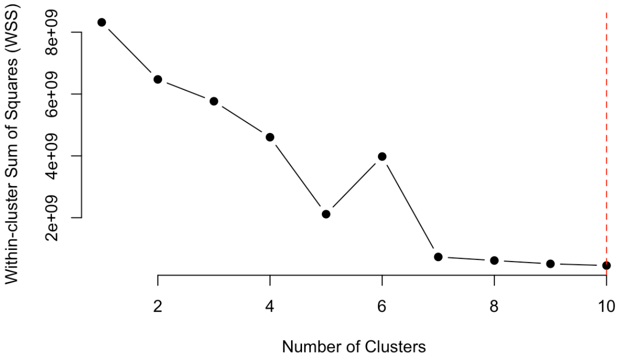
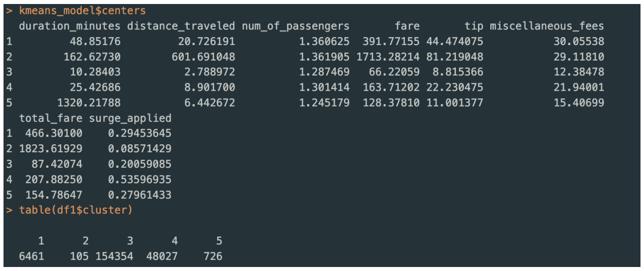


Figure 8 – Elbow point method to determine the number of clusters.

Based on feature similarity, the resulting K-Means model categorized each taxi ride into one of these five clusters. The average features of the rides within each category are represented by the centroids of these clusters. Interestingly, every cluster had unique features that allowed for a detailed examination of the variables affecting the cost of a cab ride. It was clear from a detailed analysis of the clusters that rides in the same group had comparable time, space, and financial characteristics. For example, some clusters showed higher fares and longer travel distances, while other clusters showed cheaper fares and shorter travel distances. The cluster profiles were significantly altered by the addition of surge pricing and the effects of additional charges.



```
> kmeans_model$centers
  duration_minutes distance_traveled num_of_passengers       fare       tip miscellaneous_fees
1         48.85176         20.726191          1.360625  391.77155 44.474075           30.05538
2        162.62730        601.691048          1.361905 1713.28214 81.219048           29.11810
3         10.28403          2.788972          1.287469   66.22059  8.815366           12.38478
4         25.42686          8.901700          1.301414  163.71202 22.230475           21.94001
5       1320.21788          6.442672          1.245179  128.37810 11.001377           15.40699
  total_fare surge_applied
1  466.30100    0.29453645
2 1823.61929    0.08571429
3   87.42074    0.20059085
4  207.88250    0.53596935
5  154.78647    0.27961433
> table(df1$cluster)

     1      2      3      4      5 
  6461    105 154354  48027    726 
```

Figure 9 – Clusters created by the K-means algorithm show clearly that cluster 3 is the majority in categories of taxi rides.

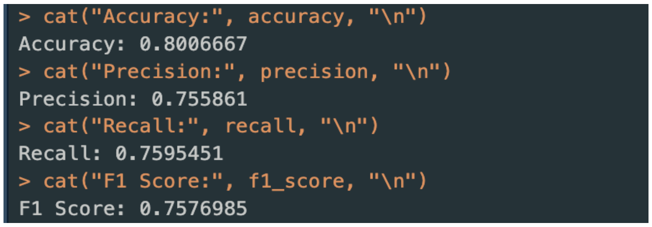
Five clusters are obtained from the taxi fare dataset using the K-Means clustering algorithm; these clusters each reveal different patterns in the ride characteristics. Cluster 1 is defined as taking an average of 48.85 minutes, travelling 20.73 kilometres, and costing 466.30 rupees in total. With an average time of 162.63 minutes and a significant distance covered of 601.69 kilometres, Cluster 2 denotes long-duration journeys, which carry a higher total charge of 1823.62 rupees. On the other hand, Cluster 3 illustrates the typical short-term, moderate-cost journeys, which take an average of 10.28 minutes to complete, covering 2.79 kilometres, and cost 87.42 rupees in total. Cluster 4 features ride with a medium duration of 25.43 minutes and an 8.90-kilometre distance covered, costing 207.88 rupees in total. Last but not least, Cluster 5 is notable for its lengthy rides, which average 1320.22 minutes and cover 6.44 kilometres for a moderate total charge of 154.79 rupees. Out of all of these clusters, Cluster 3 is the most popular; it represents typical short-haul rides at modest prices. These observations have the potential to greatly assist all parties involved in the taxi sector by directing choices about resource allocation, differentiation of services, and fare efficiency. The clustering results enable well-informed decision-making for improving urban transportation services by offering a thorough understanding of various customer categories and riding patterns.

As seen, clusters 2 and 5 show an odd type of taxi ride which can be considered as outlier and hasn't been removed in this case to maintain the size of the data set, but clusters 1, 2 and 4 show good range of taxi rides and more realistic to present to the stakeholders showing that these category of rides are most common.

C. Model 3 - Logistic Regression

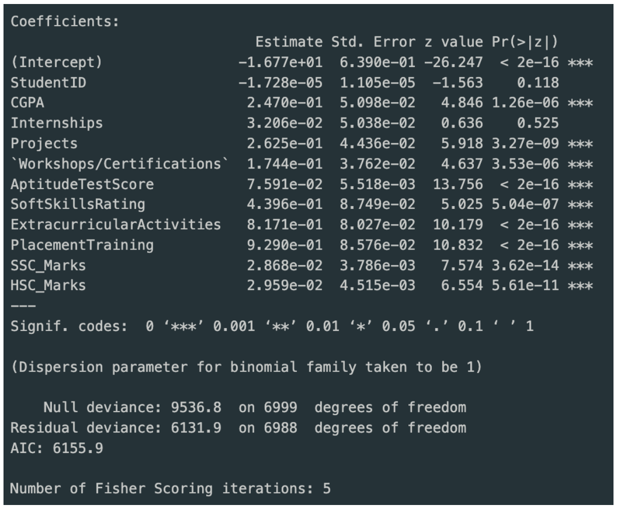
The logistic regression model is run on the Student's Placement data, Here the prediction is made whether the student will get placed or not based on the variables or elements that affect the placement status of a student. For insightful and in-depth analysis, the dataset which included extracurricular and academic characteristics went through several preparatory stages. The continuous variables that were found to have an impact on placement outcomes, such as CGPA, projects, internships, and aptitude test results, were kept in place. The model's smooth integration of categorical variables, like extracurricular activities and placement training, was made possible by their binary encoding. The dataset was split using a 70-30 split ratio to create training and testing sets, ensuring strong model performance. Training the logistic regression model on the well-chosen training data fits the binary character of the placement status variable. To improve model interpretability without sacrificing prediction accuracy, the SoftSkillsRating and CGPA were rounded.

The testing dataset was used to assess the resulting logistic regression model, which produced a confusion matrix that carefully outlines the true positive, true negative, false positive, and false negative results. This matrix is a critical instrument for assessing the predictive power of the model, providing information on its accuracy, recall, and overall efficacy in determining students' placement status. This well planned logistic regression procedure captures the combination of statistical precision, predictive analytics, and data-driven judgment, providing a significant contribution to the larger conversation about student placement prediction in the academic and professional domains.



```
> cat("Accuracy:", accuracy, "\n")
Accuracy: 0.8006667
> cat("Precision:", precision, "\n")
Precision: 0.755861
> cat("Recall:", recall, "\n")
Recall: 0.7595451
> cat("F1 Score:", f1_score, "\n")
F1 Score: 0.7576985
```

Figure 10 – Evaluation Methods of Logistic Regression



```
Coefficients:
                             Estimate Std. Error z value Pr(>|z|)    
(Intercept)                -1.677e+01  6.390e-01 -26.247  < 2e-16 ***
StudentID                  -1.728e-05  1.105e-05  -1.563    0.118    
CGPA                        2.470e-01  5.098e-02   4.846 1.26e-06 ***
Internships                 3.206e-02  5.038e-02   0.636    0.525    
Projects                    2.625e-01  4.436e-02   5.918 3.27e-09 ***
`Workshops/Certifications`  1.744e-01  3.762e-02   4.637 3.53e-06 ***
AptitudeTestScore           7.591e-02  5.518e-03  13.756  < 2e-16 ***
SoftSkillsRating            4.396e-01  8.749e-02   5.025 5.04e-07 ***
ExtracurricularActivities   8.171e-01  8.027e-02  10.179  < 2e-16 ***
PlacementTraining           9.290e-01  8.576e-02  10.832  < 2e-16 ***
SSC_Marks                   2.868e-02  3.786e-03   7.574 3.62e-14 ***
HSC_Marks                   2.959e-02  4.515e-03   6.554 5.61e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9536.8  on 6999  degrees of freedom
Residual deviance: 6131.9  on 6988  degrees of freedom
AIC: 6155.9

Number of Fisher Scoring iterations: 5
```

Figure 11 – Logistic Regression analysis results

The logistic regression model provides important insights into the roles of different factors and is intended to predict the placement status of students. Projects, CGPA, Aptitude Test Score, and Soft Skills Rating were found to be statistically significant predictors of placement outcomes among the variables. Interestingly, the positive coefficient for CGPA shows that placement chances are higher when CGPA rises. In the same way, involvement in projects, improved soft skills, and higher aptitude test scores all improve placement chances. Conversely, the negative coefficients for internships, workshops/certifications point to more complex relationships that need to be looked into. When the logistic regression model is used with the placement data, it performs quite well. The model forecasts the overall result with an accuracy of 80.07. With a precision of 75.59, the model is roughly three-quarters accurate when forecasting positive situations, such as successful placements. The significance of the model's recall rate of 75.95 lies in its ability to accurately identify real positive cases. At 75.77, the F1 Score—which balances memory and precision—indicates a well-rounded performance. Taken as a whole, these metrics indicate that the logistic regression model demonstrates an adequate capacity for precise forecasting, especially about the positive outcome of student placements, offering a solid basis for decision-making.

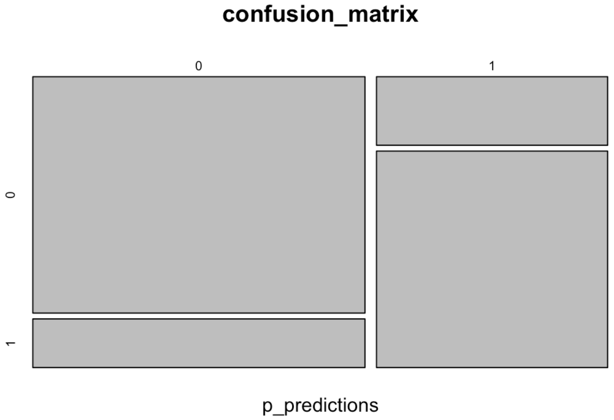
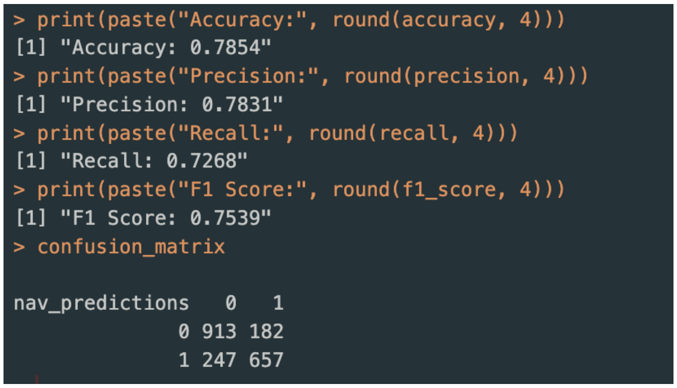


Figure 12 - Confusion Matric Provided by the Logistic Regression Algorithm shows a high prediction of not placed.

D. Model 4 - Naive Bayes

The Model predicts students' placement status based on a variety of variables using the Naive Bayes classification algorithm. Because it operates under the notion of feature independence, Naive Bayes is a probabilistic algorithm that works best with datasets that support this idea. The binary format is used during the preprocessing stage to convert categorical variables like "PlacementStatus," "ExtracurricularActivities," and "PlacementTraining." To assess model performance on test data, the dataset is divided into training and testing sets by usual machine learning methods.



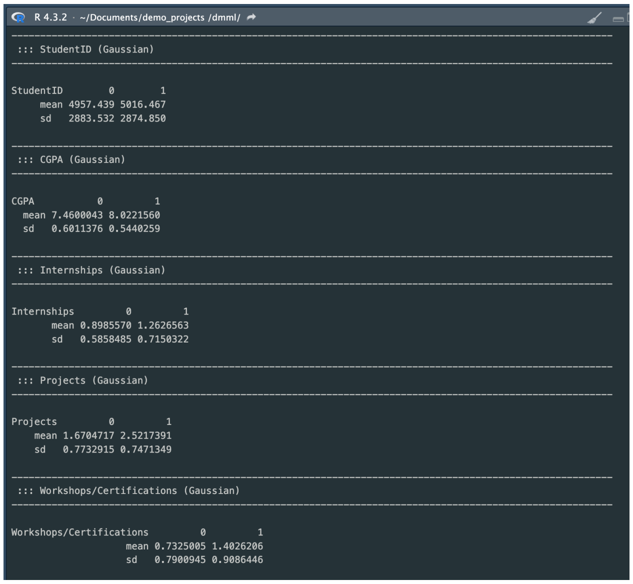
```
> print(paste("Accuracy:", round(accuracy, 4)))
[1] "Accuracy: 0.7854"
> print(paste("Precision:", round(precision, 4)))
[1] "Precision: 0.7831"
> print(paste("Recall:", round(recall, 4)))
[1] "Recall: 0.7268"
> print(paste("F1 Score:", round(f1_score, 4)))
[1] "F1 Score: 0.7539"
> confusion_matrix

nav_predictions   0   1
              0 913 182
              1 247 657
```

Figure 13 – Evaluation Methods used to check solving power of Naïve Bayes

The naive Bayes function in R is used to train the Naive Bayes model is the main focus of the project. The model is applied to the testing set when the training phase is over, producing predictions for the students' ”PlacementStatus.” The ensuing calculation of a confusion matrix and other performance metrics, such as accuracy, precision, recall, and F1 score, offers an evaluation of the predictive power of the Naive Bayes model. The model uses scatter plots as a kind of visualization in addition to model evaluation. The Naive Bayes model's classification of instances based on attributes like CGPA, Soft Skills Rating, Aptitude Test Score, and others is shown in these charts. Visualizations improve the interpretability of the model and help in a more sophisticated understanding of its decisions.

The out provided by the Naive Bayes algorithm gives insightful information about the variables affecting the placement status of the student and a priori probabilities, which is important for understanding the model's predictive power. Priori probabilities results tell that, In the training data, the placement status of 0 which is the student not being placed occurs with a higher frequency of 58.03 per cent compared to placement status 1 which is the student getting placed, with a frequency of 41.97 per cent.



```
------------------------------------------------------------------------------------------
 ::: StudentID (Gaussian)
------------------------------------------------------------------------------------------

StudentID        0        1
     mean 4957.439 5016.467
     sd   2883.532 2874.850

------------------------------------------------------------------------------------------
 ::: CGPA (Gaussian)
------------------------------------------------------------------------------------------

CGPA           0         1
  mean 7.4600043 8.0221560
  sd   0.6011376 0.5440259

------------------------------------------------------------------------------------------
 ::: Internships (Gaussian)
------------------------------------------------------------------------------------------

Internships         0         1
       mean 0.8985570 1.2626563
       sd   0.5858485 0.7150322

------------------------------------------------------------------------------------------
 ::: Projects (Gaussian)
------------------------------------------------------------------------------------------

Projects         0         1
    mean 1.6704717 2.5217391
    sd   0.7732915 0.7471349

------------------------------------------------------------------------------------------
 ::: Workshops/Certifications (Gaussian)
------------------------------------------------------------------------------------------

Workshops/Certifications         0         1
                    mean 0.7325005 1.4026206
                    sd   0.7900945 0.9086446
```

Figure 14 – Naive Bayes Model results show the Probabilities based on the variables.

The mean and standard deviation values for every feature, classified by placement status (0 and 1), are provided by the Gaussian Naive Bayes output. These data provide information about the feature distribution of students who were placed (1) and those who were not placed (0). According to the CGPA feature, students who received a placement (1) have a significantly lower standard deviation of 0.54 and a higher mean of about 8.02 compared to those who were not placed (0), which is roughly 7.46 with a standard deviation of 0.60.

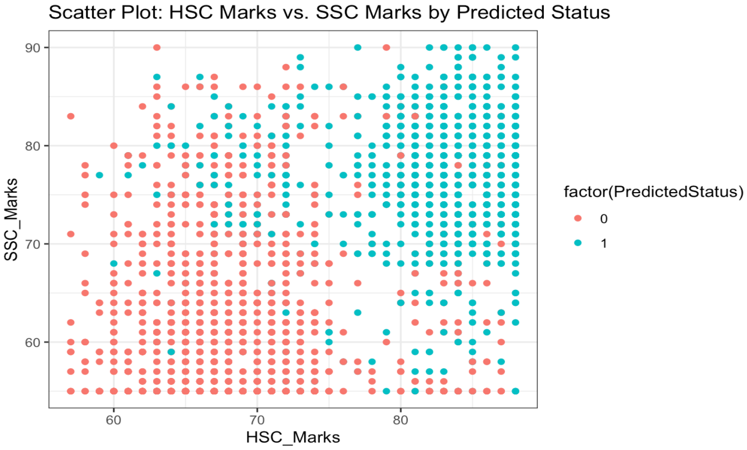


Figure 15 – Plots to show the predictions of Naïve Bayes by comparing two of the variables.

E. Model 5 - Decision Tree

The Decision tree code analyses an automobile dataset in-depth using the tidy verse and related tools. The script loads the data, examines its structure, and then focuses on cleaning it up by converting and rounding certain numerical columns and getting rid of duplicates and missing values. To depict the distribution of car owners and fuel kinds, bar charts are created as part of exploratory data analysis. The next step in data transformation is to delete unnecessary columns like "seller_type," "owner," and "transmission" and convert the "fuel" column to a factor. The rpart package is used to train a decision tree model that predicts fuel types once the dataset has been divided into training and testing sets. However, concerns regarding possible spatial limitations and the appropriateness of decision trees for this dataset are recognized. Given the complexity of the dataset and the algorithm's propensity to overfit noisy data, decision trees might not be the best fit. For improved model generalization, it is advised to investigate different machine learning algorithms and deal with any possible class imbalances in the target variable. In spite of these difficulties, the code provides a basis for comprehending the dataset and can be used as a platform for advanced modelling techniques.

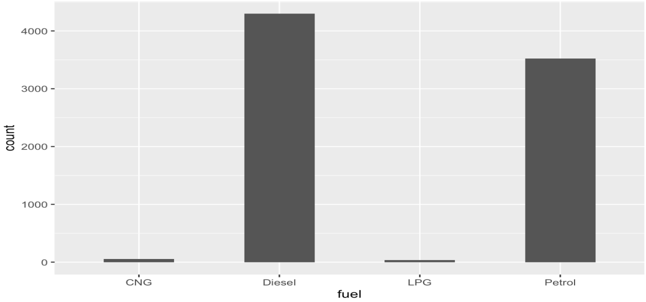


Figure 16 – Histogram of fuel type.

The Decision tree algorithm failed to run after a point due to which the results were not recorded, this was due to R not being able to handle the memory as it uses the RAM of the system.

VI. CONCLUSION

In this Project of studying the performance of different models, three different datasets including taxi fares, student placements, and automobile prices were strategically subjected to five machine-learning techniques: Naive Bayes, K-means Clustering, Linear Regression, Decision Tree, and Logistic Regression. The research sought to provide an in-depth evaluation of these algorithms' real-world performance using the CRISP-DM approach. Based on relevant criteria, Naive Bayes showed efficacy in predicting student placement outcomes, and K-means Clustering identified complex patterns in the taxi fare data. While decision trees offered insightful information about categorizing different car fuel kinds, linear regression illuminated important aspects impacting taxi fares. The power of logistic regression to determine the likelihood of a student being placed successfully was demonstrated. Recall, Precision, Accuracy, and F1 score were among the evaluation metrics that were carefully used together with standard measures to thoroughly evaluate algorithmic performance. The research, carried out using the R programming language, explored the subtleties of various algorithms' performance in addition to demonstrating how strategically to apply them. The report employed the CRISP-DM framework to guide its way through six distinct stages: Business Problem Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. This approach allowed for a thorough examination of a wide range of datasets and improved comprehension of algorithmic behaviour. Logistic Regression, K-means clustering, Linear Regression, and Naive Bayes, revealed how well they worked with particular problems in selected datasets.

VII. FUTURE WORK

There is always room for improvement, as in the first project and analysis of these models, there was a good response and performance of the model. whereas the performance could have been enhanced by doing a little more processing and tuning the data to fit the model, there were some cases of overfitting while performing the decision tree, such cases must be well read and fixed in the future, the data before training and testing must be analysed manually by the user to see any kind of pattern that might hinder the process while modelling. The main goal in the future is to deploy models that are industry-ready and can be used with any kind of data that is fit for the model. The Decision tree algorithm needs to be fixed and made sure it is suitable for running.

REFERENCES

[1] N. Rosidi, “Indispensable Data Mining Tools You Need as a Data Scientist,” Medium, Jul. 05, 2023. https://medium.com/@nathanrosidi/indispensable-data-mining-toolsyou-need-as-a-data-scientist-cb1923df3d51 (accessed Jan. 02, 2024).

[2] B. Lantz, Machine Learning with R: Expert Techniques for Predictive Modeling. Birmingham: Packt Publishing, Limited, . A, 2019.ˆ

[3] R Core Team, “R: The R Project for Statistical Computing,” Rproject.org, 2019. https://www.r-project.org

[4] J. Han, M. Kamber, and J. Pei, ”Data mining: Concepts and techniques,” Amsterdam: Elsevier, 2011.

[5] I. H. Witten, E. Frank, and M. A. Hall, ”Data mining: Practical machine learning tools and techniques,” Burlington: Morgan Kaufmann, 2016.

[6] A. K. Kuyucu, “ML Tutorial 7 — Naive Bayes Classifier for Text Data,” *Medium*, Dec. 17, 2023. https://medium.com/gitconnected/ml-tutorial-7-naive-bayes-classifier-for-text-data-d123f0cae75d (accessed on Jan. 05, 2024)

[7] K. Dissanayake, “Machine Learning Algorithms(7)- Naive Bayes' Algorithm and K-Nearest Neighbors Algorithm,” Medium, Nov. 20, 2023. https://medium.com/towardsdev/machine-learning-algorithms-7-naive-bayes-algorithm-and-k-nearest-neighbors-algorithm-80b154dc0f13 (accessed Jan. 05, 2024).