PREDICTING POLITICAL PARTY AFFILIATION USING CLASSIFICATION MODELS

Project Report

Submitted to the Faculty of Engineering of

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA, KAKINADA

Impartial fulfillment of the requirements for the award of the Degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

B.V S S Sri Rama Varshith
(21481A0540)

B.Jaswanth
(21481A0520)

B.Divya Teja
(21481A0514)

D.Hema Sri
(21481A0559)

Under the Enviable and Esteemed Guidance of

Dr.G.Sridevi, M.Tech,Ph.D

Professor, Department of CSE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SESHADRI RAO GUDLAVALLERU ENGINEERING COLLEGE

(An Autonomous Institute with Permanent Affiliation to JNTUK, Kakinada)

SESHADRI RAO KNOWLEDGE VILLAGE GUDLAVALLERU – 521356

ANDHRA PRADESH

2023-24

SESHADRI RAO GUDLAVALLERU ENGINEERING COLLEGE

(An Autonomous Institute with Permanent Affiliation to JNTUK, Kakinada)

SESHADRI RAO KNOWLEDGE VILLAGE, GUDLAVALLERU

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project report entitled **"Prediction Of Political Party Affiliation Using Classification Models"** is a bonafide record of work carried out by B.V S S Sri Rama Varshith(21481A0540), B.Jaswanth(21481A0520), B.Divya Teja (21481A0514), D. Hema Sri(21481A0559) under the guidance and supervision of **Dr. G.Sridevi,** Professor, Computer Science and Engineering, in the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering of Jawaharlal Nehru Technological University Kakinada, Kakinada during the academic year 2023-24.

Project Guide

(Dr. G. Sridevi)

Head of the Department

(Dr. M. BABU RAO)

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people who made it possible and whose constant guidance and encouragements crown all the efforts with success.

We would like to express our deep sense of gratitude and sincere thanks to **Dr.G.Sridevi, Professor, Computer Science and Engineering** for her constant guidance, supervision and motivation in completing the project work.

We feel elated to express our floral gratitude and sincere thanks to **Dr. M. Babu Rao, Head of the Department, Computer Science and Engineering** for his encouragements all the way during analysis of the project. His annotations, insinuations and criticisms are the key behind the successful completion of the project work.

We would like to take this opportunity to thank our beloved principal **Dr.Burra Karuna Kumar** for providing a great support for us in completing our project and giving us the opportunity for doing project.

Our Special thanks to the faculty of our department and programmers of our computer lab. Finally, we thank our family members, non-teaching staff and our friends, who had directly or in directly helped and supported us in completing our project in time.

Team Members

B.V.S.S.Sri Rama Varshith(21481A0540)
B.Jaswanth(21481A0520)
B.Divya Teja(21481A0514)
D.Hema Sri(21481A0559)

INDEX

LIST OF TABLES:

LIST OF FIGURES:

ABSTRACT

This project aims to predict political party affiliation (Republican or Democrat) based on voting records using various classification models. The dataset comprises 16 features representing different voting issues, with categorical values (yes/no), and a target variable indicating party affiliation. The analysis was conducted using the Orange data mining tool, which facilitated data preprocessing, model building, and evaluation.

Initially, the dataset was explored to understand its structure and handle missing values by imputing them with the most frequent value. Multiple classification models, including k-Nearest Neighbors (kNN), Decision Tree, and Naive Bayes, were evaluated using a 75%-25% train-test split. Following this, feature selection was performed to identify the most influential features, which were then used to refine the models.

The results indicated that the Decision Tree model achieved the highest accuracy, particularly after preprocessing and feature selection. This project highlights the effectiveness of different classification models and the importance of preprocessing and feature selection in improving model performance. Future work could explore more sophisticated imputation techniques, additional classifiers, and hyper parameter tuning to further enhance prediction accuracy.

CHAPTER – 1
INTRODUCTION

1.1 Introduction

The ability to accurately predict political party affiliation based on voting records is a valuable tool for political analysts, researchers, and strategists. This project focuses on using data mining techniques to classify political party affiliation (Republican or Democrat) using a dataset of voting records. Each record consists of 16 features representing votes on various legislative issues, with values categorized as 'yes' or 'no'. The target variable is the party affiliation of the voter.

The primary objective of this project is to identify the best classification model for predicting party affiliation by leveraging the capabilities of the Orange data mining tool. Orange provides a comprehensive suite of tools for data visualization, preprocessing, model building, and evaluation, making it an ideal platform for this analysis.

The process begins with an initial exploration of the dataset to understand its structure and address any missing values. Following this, several classification models, including k-Nearest Neighbors (kNN), Decision Tree, and Naive Bayes, are trained and evaluated to identify the most accurate predictor. Preprocessing steps such as imputation of missing values and feature selection are employed to enhance model performance.

The models are assessed using various performance metrics, and the impact of preprocessing and feature selection on their accuracy is analyzed. The project aims to demonstrate how effective preprocessing and careful model selection can lead to significant improvements in predictive accuracy. The findings from this analysis provide insights into the strengths and weaknesses of different classification approaches and the importance of data preparation in machine learning tasks.

By the end of the project, the Decision Tree model emerged as the best-performing classifier, achieving the highest accuracy in predicting political party affiliation. The results underscore the potential of data mining techniques in political analysis and highlight areas for future research and enhancement.

Classification

- Classification may be defined as the process of predicting class or category fromobserved values or given data points. The categorized output can have the form such as “Black” or “White” or “spam” or “no spam”.
- Mathematically, classification is the task of approximating a mapping function (f) from input variables (X) to output variables (Y). It is basically belongs to the supervised machine learning in which targets are also provided along with the input data set.
- An example of classification problem can be the spam detection in emails. There can be only two categories of output, “spam” and “no spam”; hence this is a binary type classification.
- To implement this classification, we first need to train the classifier. For this example, “spam” and “no spam” emails would be used as the training data. After successfully train the classifier, it can be used to detect an unknown email.

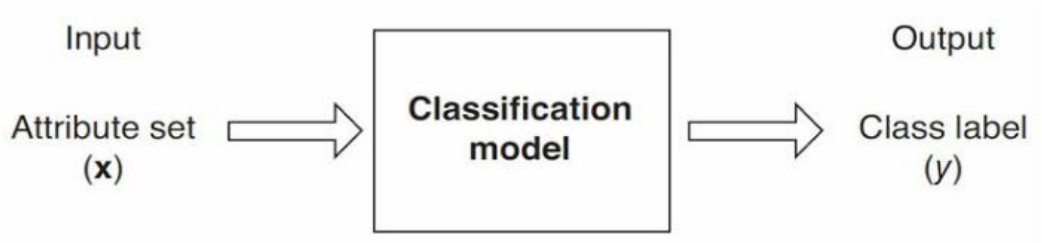


Fig 1.1.1 Classification Definition

Types of Learners in Classification:

We have two types of learners in respective to classification problems −

Lazy Learners:

As the name suggests, such kind of learners waits for the testing data to be appeared after storing the training data. Classification is done only after getting the testing data. They spend less time on training but more time on predicting. Examples of lazy learners are K-nearest neighbor and case-based reasoning.

Eager Learners:

As opposite to lazy learners, eager learners construct classification model without waiting for the testing data to be appeared after storing the training data. They spend more time on training but less time on predicting. Examples of eager learners are Decision Trees, Naïve Bayes and Artificial Neural Networks (ANN).

Classification Models:

Table 1.1.0: Supervised Learning Algorithms

Decision Tree	Highly interpretable classification or regression model that splits data-feature values into branches at decision nodes.(e.g., if a feature is a color, each possible color becomes a new branch) until a final decision output is made.	Classification
Naïve Bayes	The Bayesian method is a classification method that makes use of the Bayesian theorem. The theorem updates the prior knowledge of an event with the independent probability of each feature that can affect the event.	Regression and Classification
Support Vector Machine	SVM, is typically, used for the classification task. SVM algorithm finds a hyperplane that optimally divided the classes. It is best used with a non-linear solver.	Regression and classification
Random Forest	The algorithm is built upon a decision tree to improve the accuracy drastically. Random forest generates many times n simple decision trees and uses the 'majority vote' method to decide on which label to return. For the classification task, the final prediction will be the one with the most vote; while for the regression task, the average prediction of all the trees is the final prediction.	Regression and Classification
AdaBoost	Classification or regression technique that uses a multitude of models to come up with a decision but weights them based on their accuracy in predicting the outcome	Regression and Classification
Gradient-boosting trees	Gradient boosting trees is a state-of-the-art classification/regression technique. It is focusing on the error committed by the previous trees and tries to correct it.	Regression and Classification

1.2 Problem statement

How can we accurately predict political party affiliation based on congressional voting records using classification models?

Understanding voting behavior and predicting political party affiliation is essential for political analysts, policymakers, and researchers. The dataset comprises voting records on various issues, with votes categorized as 'yes' or 'no,' and the target variable is the political party affiliation, classified as either Republican or Democrat.

The primary challenge is to develop a robust classification model to accurately predict a congressperson's party affiliation based on their voting patterns. This involves handling the 5.6% of missing data uniformly distributed across all columns. We opted to impute missing values with the most frequent value for each categorical feature to maintain the dataset's integrity and completeness. Additionally, we employed feature selection techniques to identify and retain the most relevant features, ensuring that our models focus on the most informative aspects of the data.

We evaluated multiple classification algorithms, including k-Nearest Neighbors (kNN), Decision Trees, and Naive Bayes, to identify the most accurate model. By splitting the data into training and testing sets, we could assess each model's performance on unseen data, providing a realistic estimate of their generalization capability. The goal was not only to achieve high predictive accuracy but also to gain insights into the factors influencing political party affiliation based on voting records.

CHAPTER 2

PROPOSED METHOD

2.1 Methodology

The methodology for predicting political party affiliation using classification models involves several detailed steps. These steps include data loading and exploration, data preprocessing, data splitting, model training, evaluation, and final model selection. Each of these steps is critical in building an effective and accurate predictive model.

2.1.1 Procedure Explanation

1. Data Acquisition and Exploration:

Dataset Description: Obtain the voting dataset containing voting records on various legislative issues, with features such as handicapped-infants, water project cost sharing, and adoption of the budget resolution, along with the target variable 'Party' indicating political party affiliation (Republican or Democrat).

Data Exploration: Use the Orange data mining tool to load the dataset and explore its structure, summary statistics, and distributions of features.

2. Data Preprocessing:

Handling Missing Values: Implement an imputation strategy to address missing values in the dataset, using techniques such as imputing with the most frequent value for categorical features.

Data Cleaning: Address any inconsistencies or anomalies in the dataset, such as outliers or incorrect data entries.

Feature Encoding: If necessary, encode categorical features into numerical representations suitable for modeling.

3. Model Building and Evaluation:

Data Splitting: Divide the preprocessed dataset into training (75%) and testing (25%) sets using the Data Sampler widget in Orange.

Model Selection: Train and evaluate multiple classification models using the training dataset, including k-Nearest Neighbors (kNN), Decision Tree, and Naive Bayes classifiers.

Performance Evaluation: Assess the performance of each model using evaluation metrics such as accuracy, precision, recall, F1 score, and confusion matrices. Use the Test & Score widget to evaluate models on the test dataset and visualize their performance.

4. Feature Selection:

Ranking Features: Rank the features based on their importance using the Rank widget in Orange. Identify the most informative features that contribute to predicting party affiliation.

Feature Selection: Select the top-ranked features or apply additional feature selection techniques to refine the dataset and improve model performance.

5. Model Refinement and Optimization:

Hyperparameter Tuning: Fine-tune the parameters of the selected classification models using techniques such as grid search or random search to optimize their performance.

Ensemble Methods: Explore ensemble methods such as Random Forests or Gradient Boosting Machines to further enhance predictive accuracy.

6. Performance Comparison and Interpretation:

Model Comparison: Compare the performance of different classification models before and after preprocessing, feature selection, and hyperparameter tuning.

Interpretability: Interpret the decision-making process of the selected model, such as decision rules or feature importance rankings, to gain insights into the factors influencing party affiliation predictions.

7. Validation and Sensitivity Analysis:

Cross-Validation: Perform cross-validation to validate the generalization performance of the selected model and assess its robustness to variations in the dataset.

Sensitivity Analysis: Conduct sensitivity analysis to evaluate the model's stability and sensitivity to changes in input parameters or dataset characteristics.

8. Documentation and Reporting:

Document Workflow: Document the entire workflow, including data preprocessing steps, model building process, evaluation results, and interpretation of findings.

Report Generation: Generate a comprehensive report summarizing the methodology, results, and conclusions of the project, including visualizations, and performance metrics.

2.1.2 Block Diagram

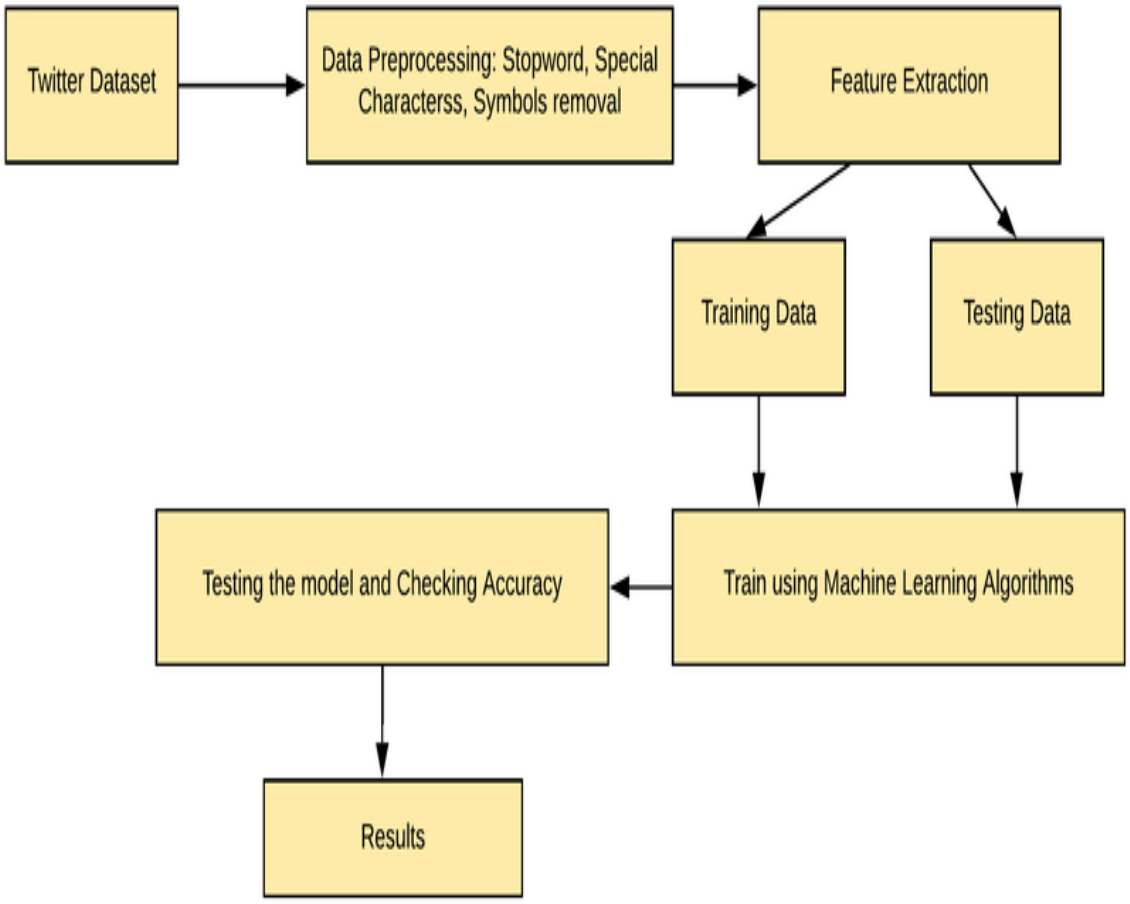


Fig 2.1.1 Block Diagram

2.2 Data Preparation:

2.2.1 Dataset Description:

Overview:

The dataset contains voting records on various legislative issues, along with the political party affiliation (Republican or Democrat) of the corresponding voters. Each instance represents a member of the United States Congress, and each feature represents the voting stance on a specific issue. The dataset aims to predict the political party affiliation of a member based on their voting patterns.

Features:

1. **Handicapped Infants:** Indicates whether the member voted in favor of legislation regarding handicapped infants.
2. **Water Project Cost Sharing:** Indicates whether the member voted in favor of legislation regarding water project cost sharing.
3. **Adoption of the Budget Resolution:** Indicates whether the member voted in favor of adopting the budget resolution.
4. **Physician Fee Freeze:** Indicates whether the member voted in favor of legislation regarding physician fee freeze.
5. **El Salvador Aid:** Indicates whether the member voted in favor of legislation regarding aid to El Salvador.
6. **Religious Groups in Schools:** Indicates whether the member voted in favor of legislation regarding religious groups in schools.
7. **Anti-Satellite Test Ban:** Indicates whether the member voted in favor of legislation regarding anti-satellite test ban.
8. **Aid to Nicaraguan Contras:** Indicates whether the member voted in favor of legislation regarding aid to Nicaraguan contras.
9. **MX Missile:** Indicates whether the member voted in favor of legislation regarding MX missile.
10. **Immigration:** Indicates whether the member voted in favor of legislation regarding immigration.
11. **Synfuels Corporation Cutback:** Indicates whether the member voted in favor of legislation regarding synfuels corporation cutback.
12. **Education Spending:** Indicates whether the member voted in favor of legislation regarding education spending.
13. **Superfund Right to Sue:** Indicates whether the member voted in favor of legislation regarding superfund right to sue.
14. **Crime:** Indicates whether the member voted in favor of legislation regarding crime.
15. **Duty Free Exports:** Indicates whether the member voted in favor of legislation regarding duty-free exports.
16. **Export Administration Act South Africa:** Indicates whether the member voted in favor of legislation regarding the export administration act for South Africa.

Target Variable:

Party: Indicates the political party affiliation of the member, which can be either Republican or Democrat.

Data Format:

The values for all features are categorical, with options being either 'yes' or 'no'.

The target variable 'Party' is also categorical, with values indicating the political party affiliation.

Dataset Size:

The dataset consists of a total of N instances (members of Congress) and M features (voting issues).

2.2.2 Data Pre-Processing

To build a best classification model the main step is data preprocessing. The collected data needs to be preprocessed to ensure its quality. This involves handling missing values, dealing with outliers, and transforming the data into a format suitable for analysis. Data preprocessing also involves converting the data into numerical form, as most classification algorithms require numerical input. Below method is suitable for out dataset.

Data Transformation

Converting data into a format suitable for modeling, including normalizing or standardizing numerical features and encoding categorical variables.

Feature Selection/Feature Engineering

In this step we apply the technique for the classification model is feature selection or feature Engineering. Feature selection involves identifying the most relevant attributes in the dataset for classification. This can be done using various techniques, such as correlation analysis, information gain, and principal component analysis.

Feature Engineering is the process of creating new features or modifying existing features in a dataset to improve the performance of a machine learning model. Features are the input variables or attributes that are used by the model to make predictions. Effective feature engineering can enhance a model's ability to learn patterns in the data and make more

accurate predictions. For our dataset feature engineering is most suitable because it make more accurate predictions. This can be achieved by any one of the following method:

Frequency Encoding

Replace categorical values with their frequency or count in the dataset. This can help the model understand how common each category is**.**

Categorical Variable Transformations

For ordinal variables, create new features based on rank or category. For nominal variables, consider encoding using methods such as one-hot encoding

After Preprocessing the data again we apply below techniques

Model Training: To build a best classification model we need to train the model. Model training involves using the selected classification algorithm to learn the patterns in the data. The data is divided into a training set and a validation set. The model istrained using the training set, and its performance is evaluated on the validation set.

Model Evaluation: The last step in building a classification model is model evaluation. Model evaluation involves assessing the performance of the trained model on a test set. This is done to ensure that the model generalizes well.

After preprocessing the data, adding extra feature, training the data(using data sampler)and again evaluate the model then we get the best model by observing the evaluation metrics(accuracy, precision, recall)

CHAPTER 3

RESULTS

3.1 ORANGE tool description:

Orange is an open-source data visualization and analysis tool designed for users seeking intuitive yet powerful solutions in machine learning and data mining. Its hallmark feature is a visual programming interface, facilitating the construction of data analysis workflows through interconnected components (widgets). With this approach, users can perform various tasks seamlessly, including data preprocessing, exploratory data analysis, predictive modeling, and visualization. Orange offers an array of preprocessing techniques, allowing users to handle missing values, scale features, encode categorical variables, and select relevant features effortlessly. Moreover, its extensive collection of visualization tools enables users to explore datasets visually, uncovering relationships, distributions, and patterns. Through integration with machine learning algorithms and ensemble learning methods, Orange empowers users to train models for classification, regression, clustering, and association rule mining. Model evaluation tools further aid in assessing model performance, ensuring robust and reliable results.

3.2 SCREENSHOTS:

ORANGE Tool: **Demonstrate performing classification on data sets.**

STEP 1: Load your dataset using the File Widget.

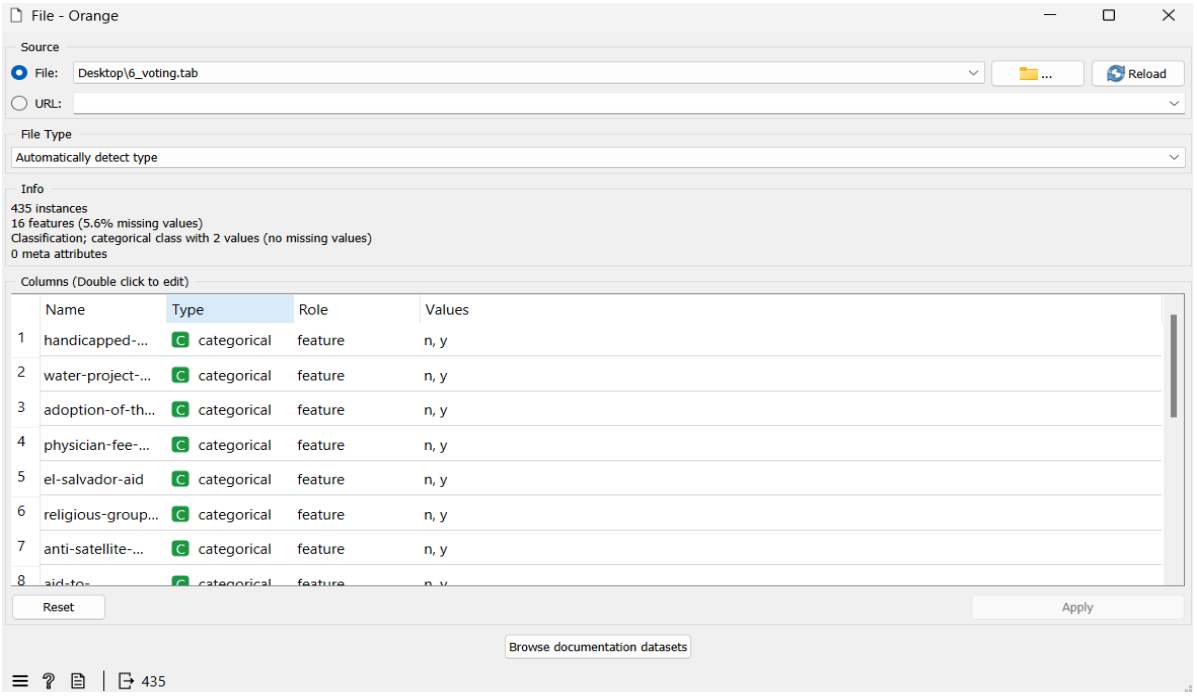


Fig 3.2.1 Load the Dataset

STEP 2:

Connect the File widget to the Data Info widget to get an overview of the dataset and to the Feature Statistics widget to obtain statistics on each feature and also to the data table widget.

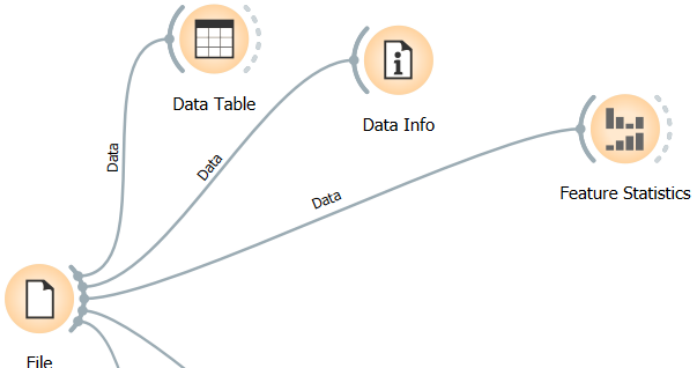


Fig 3.2.2 Flow of widgets for data analysis

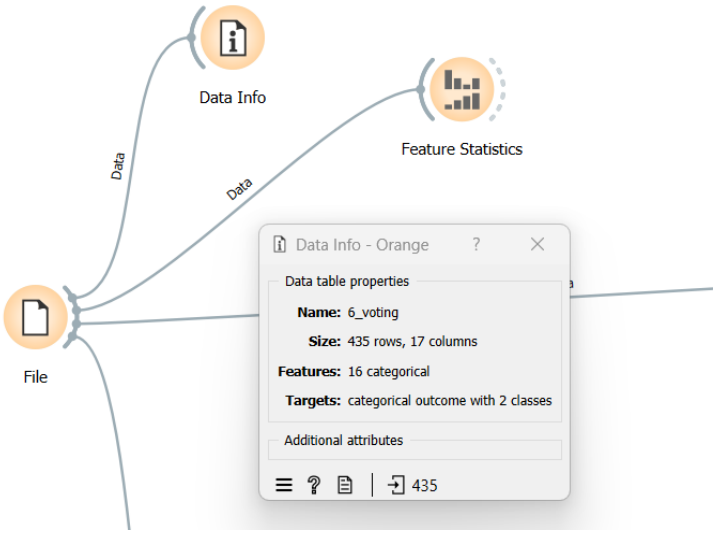


Fig 3.2.3 Data info of dataset

Fig 3.2.4 Feature Statistics of the dataset

STEP 3: The dataset contains 435 rows and 16 columns with one target value. The data contains some missing values. Before preprocessing we need to analyze metrics for different classification models. We use the training models like KNN, Tree, Naïve Bayes.

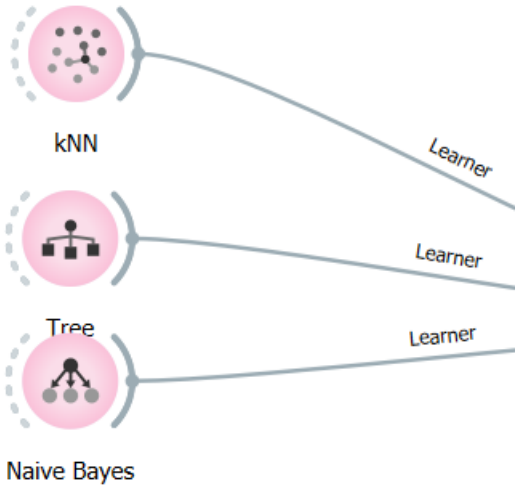


Fig 3.2.5 Classification models connections

STEP 4: Test and Score is used to evaluate the performance of our models.

- Give a connection from file widget to the Data sampler in which the training data is 75% and the testing data is 25%.
- Give a connection from datasampler to Test and Score .
    - Our classification models are also connected to test and score.

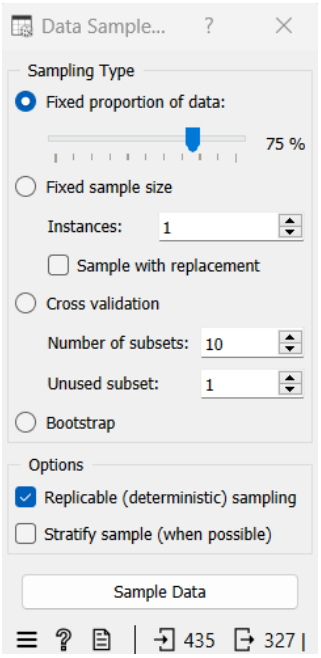


Fig 3.2.6 Data Sampler

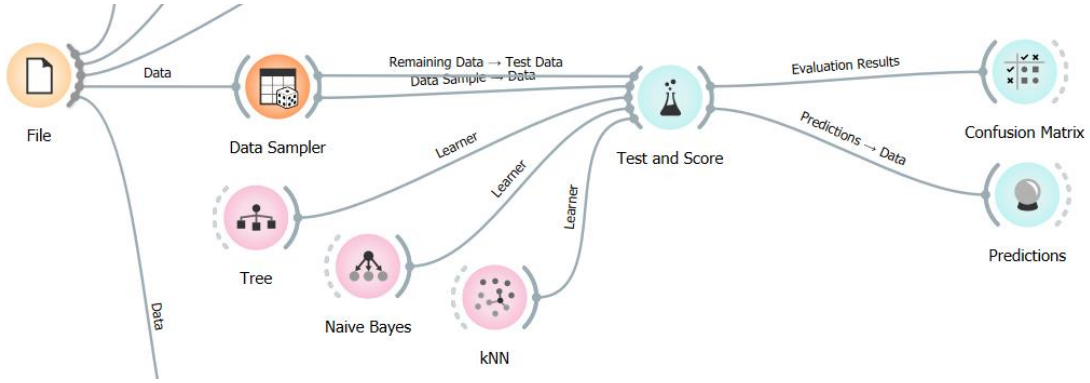


Fig 3.2.7 Flow of widgets before preprocessing

This is the evaluation metrics derived from the test and score **before performing preprocessing techniques.**

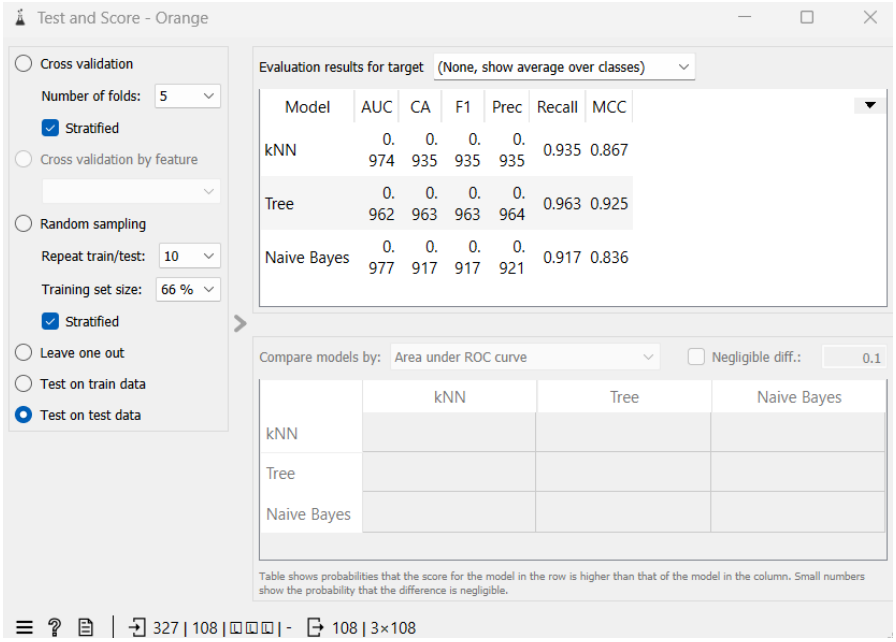


Fig 3.2.8 Test and Score before Preprocessing

Confusion matrix before preprocessing:

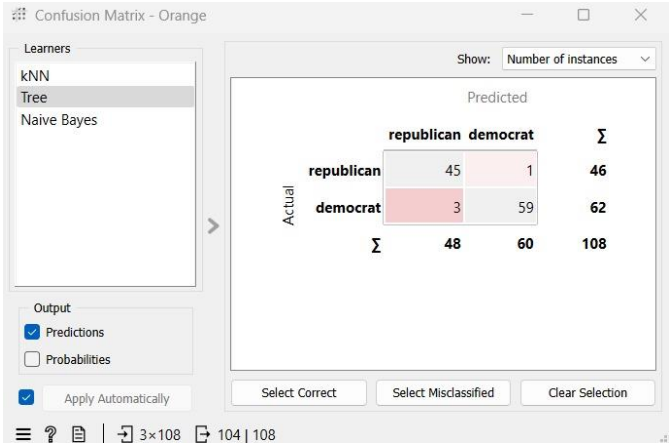


Fig 3.2.9 Confusion Matrix before preprocessing

STEP 5: Apply preprocessing to the data.

Connect the File widget to the Preprocess widget.

In the Preprocess widget, apply the following technique:

- Impute missing values.

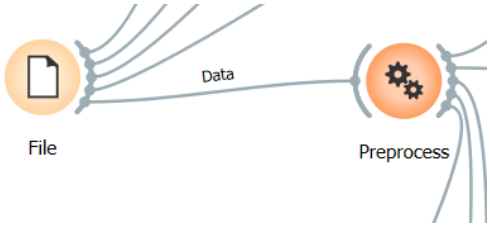


Fig 3.2.10 Connecting preprocessor to file widget

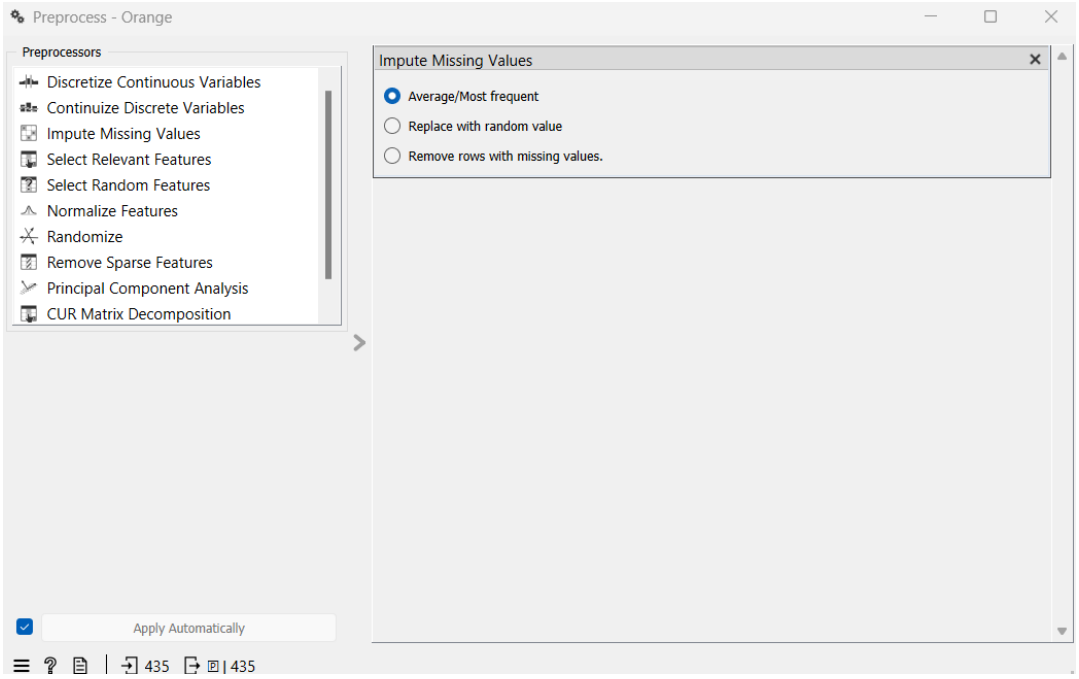


Fig 3.2.11 Applying preprocessing techniques

STEP 6:

Connect a Data table ,Data Sampler and the Rank widget from the Pre Processor widget. In the Rank widget we can identify the best ranked features in the given dataset.

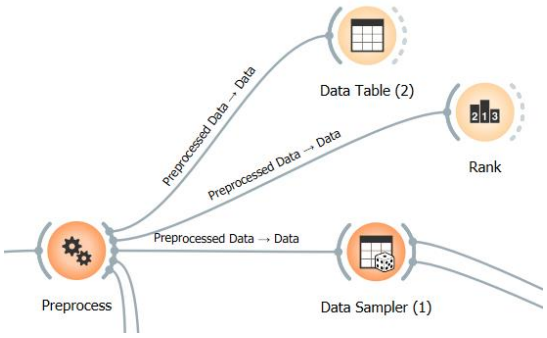
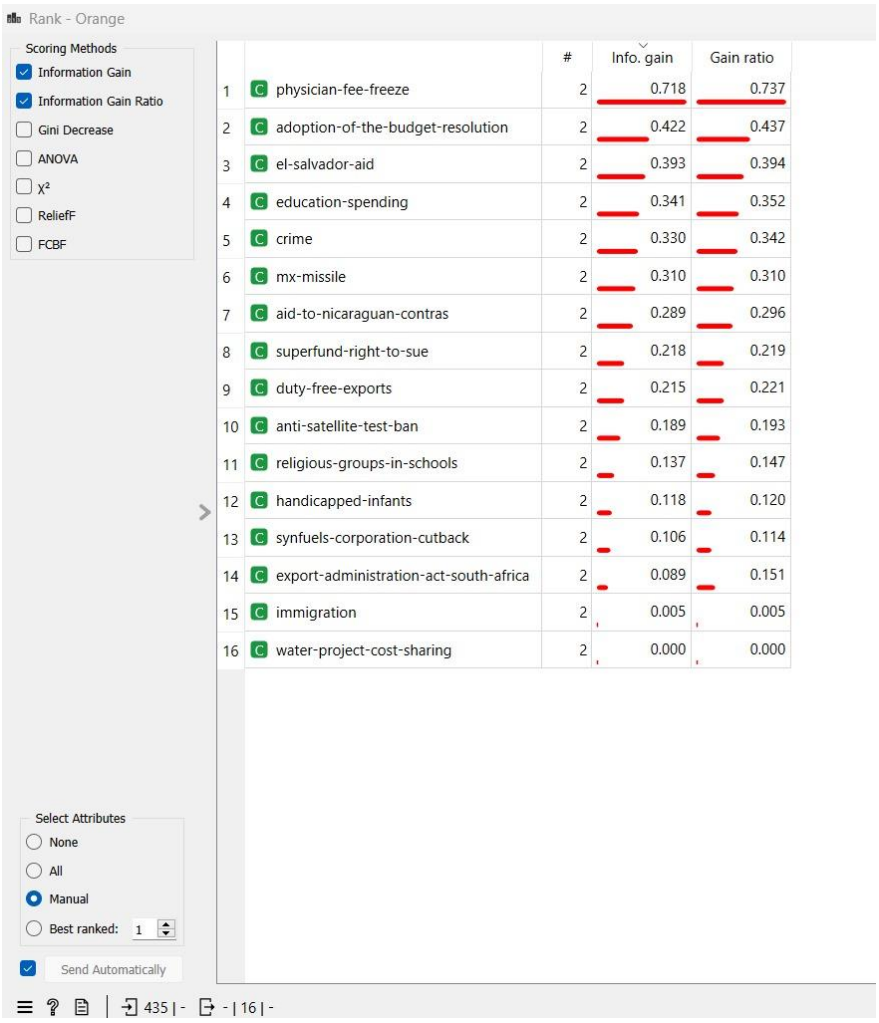


Fig 3.2.12 Connecting rank widget to preprocessor



		#	Info. gain	Gain ratio
1	physician-fee-freeze	2	0.718	0.737
2	adoption-of-the-budget-resolution	2	0.422	0.437
3	el-salvador-aid	2	0.393	0.394
4	education-spending	2	0.341	0.352
5	crime	2	0.330	0.342
6	mx-missile	2	0.310	0.310
7	aid-to-nicaraguan-contras	2	0.289	0.296
8	superfund-right-to-sue	2	0.218	0.219
9	duty-free-exports	2	0.215	0.221
10	anti-satellite-test-ban	2	0.189	0.193
11	religious-groups-in-schools	2	0.137	0.147
12	handicapped-infants	2	0.118	0.120
13	synfuels-corporation-cutback	2	0.106	0.114
14	export-administration-act-south-africa	2	0.089	0.151
15	immigration	2	0.005	0.005
16	water-project-cost-sharing	2	0.000	0.000

Fig 3.2.13 Rank of Features

STEP 7:

- Connect the Data Sampler widget to the Test and Score widget.
- Connect the Test and Score widget to multiple models for training and evaluation.

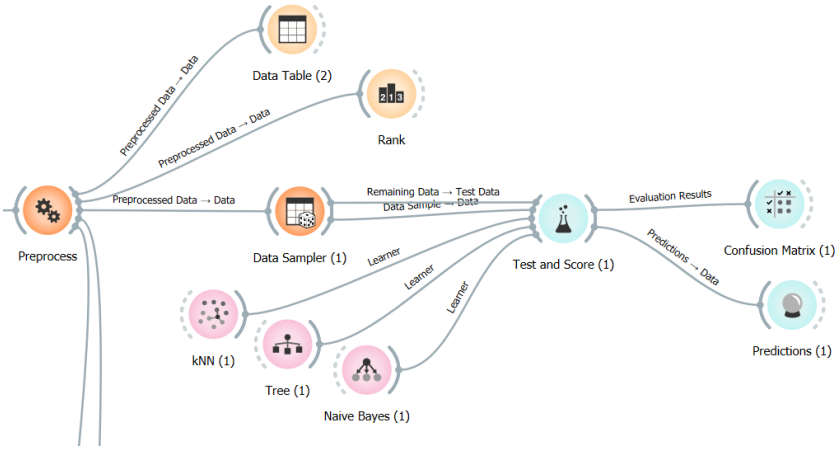


Fig 3.2.14 Flow of widgets after preprocessing

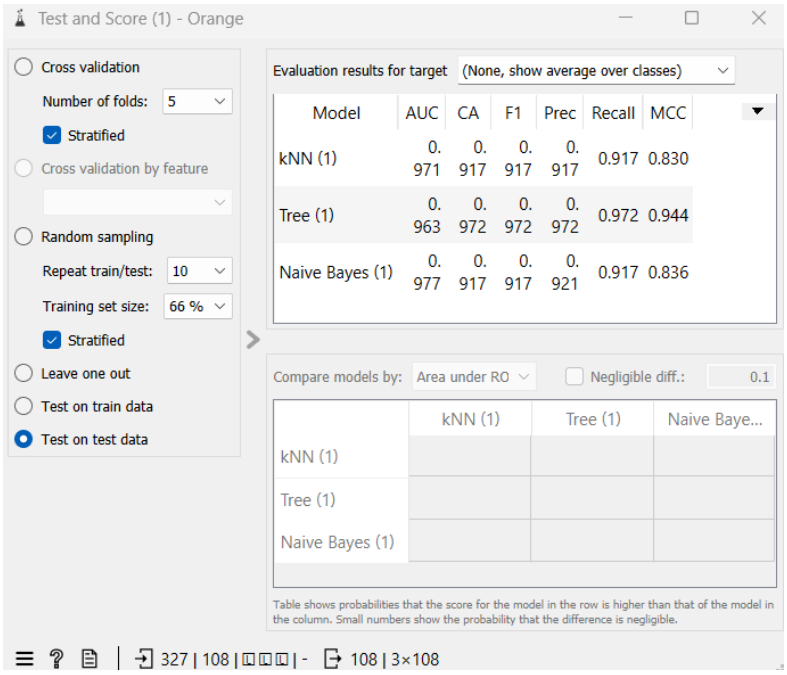


Fig 3.2.15 Test and Score after preprocessing

Connect the Test and Score widget to the Confusion Matrix widget to evaluate model performance.

Confusion Matrix:

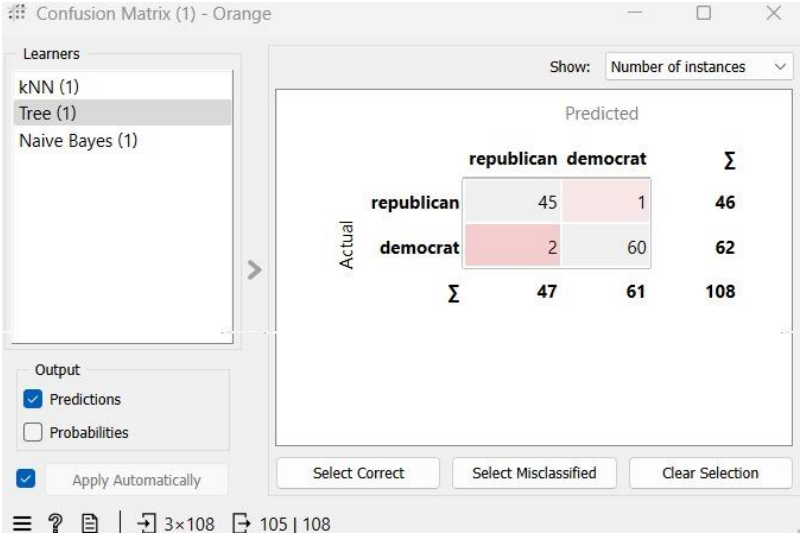


Fig 3.2.16 Confusion Matrix after preprocessing

STEP 8:

Now we will select the columns which have the best rank in the Rank widget by using the Select Column widget which is connect to the pre processor. In the Select Column widget we have selected the top 15 ranked features and ignored one feature which is in the last position in the Rank Widget.

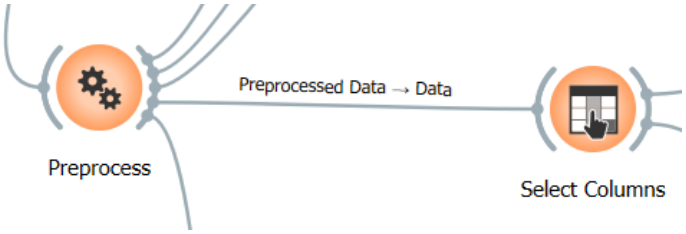


Fig 3.2.17 Connecting Select Column widget to preprocessor

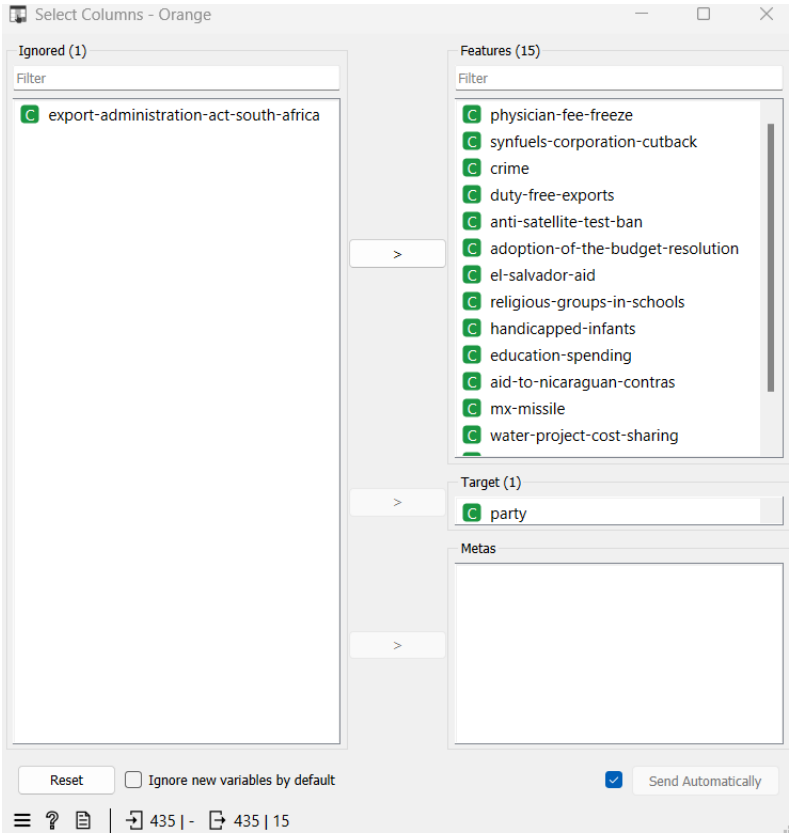


Fig 3.2.18 Selecting the columns

STEP 9:

- Connect a Data table and the Data sampler to the Select Column widget.
- Data sampler is then connected to the test score which is connected multiple models for training and evaluation.

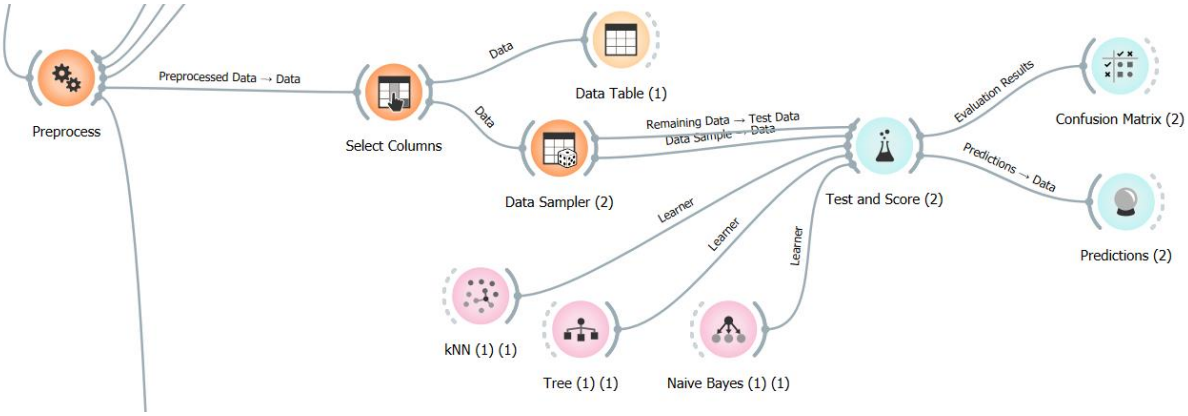


Fig 3.2.19 Flow of widget after selecting columns

This is the evaluation metrics derived from the test and score.

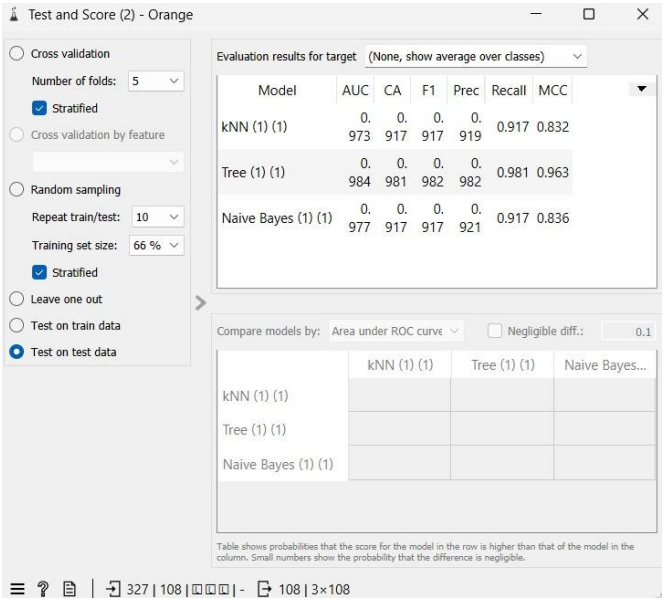


Fig 3.2.20 Test and Score after selecting top 15 columns

Confusion matrix after selecting top 15 features:

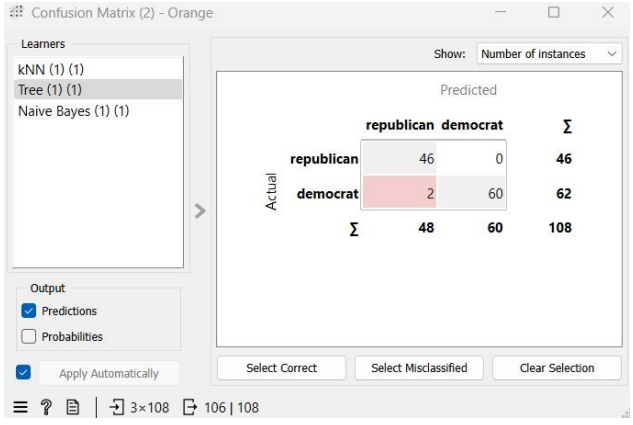


Fig 3.2.21 Confusion Matrix after selecting top 15 columns

Complete Flow:

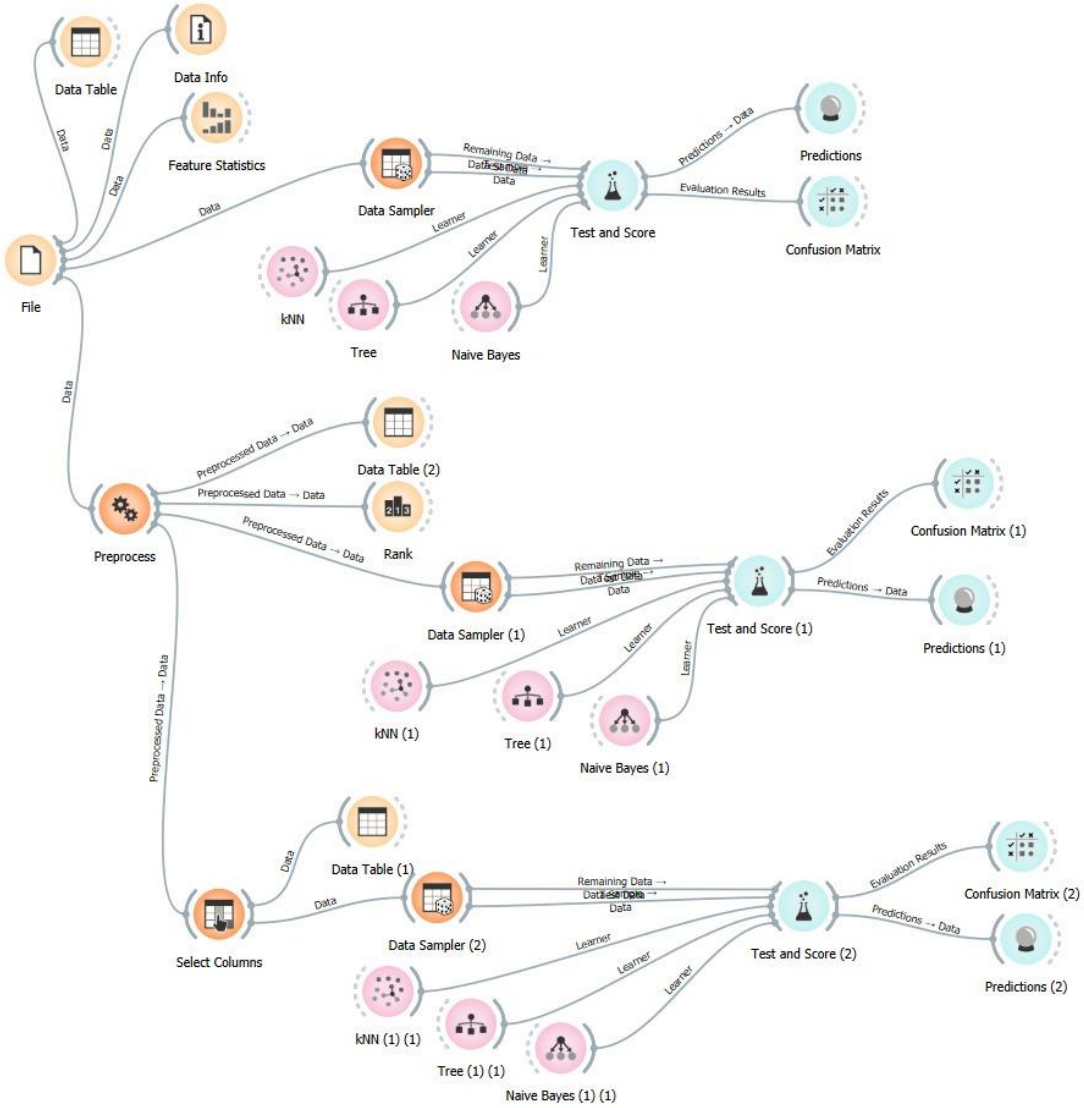


Fig 3.2.22 Complete Work Flow

Analysis:

Before Preprocessing:

Initially, we loaded the dataset containing congressional voting records into the Orange tool. Using the Data Info, Data Table, and Feature Statistics widgets, we conducted an initial evaluation of the dataset. We identified that the dataset contained 5.6% missing data uniformly distributed across all columns, with the "water project cost sharing" feature having the highest percentage of missing values. Despite the presence of missing values, we proceeded to split the data into training (75%) and testing (25%) sets and evaluated the

performance of various classification models including k-Nearest Neighbors (kNN), Decision Trees, and Naive Bayes. Among these models, the Decision Tree showed the highest accuracy at 96.3%, outperforming both kNN and Naive Bayes.

After Preprocessing

To address the issue of missing data, we employed the Preprocess widget to impute missing values using the most frequent value (mode) for each categorical feature. Post-imputation, we re-split the data into training and testing sets and re-evaluated the classification models. We observed an increase in the accuracy of the Decision Tree model to 97.2%, while the accuracy of the kNN model decreased, and the Naive Bayes model's accuracy remained unchanged. This indicated that the Decision Tree model benefitted the most from the imputed data, further solidifying its position as the most effective model for our dataset.

After Selecting Top 15 Features

To further refine our approach, we utilized the Rank widget to assess the importance of each feature based on various criteria like information gain. We identified the top 15 features that were most relevant for predicting political party affiliation and excluded the feature "water project cost sharing" due to its low information gain. Using the Select Columns widget, we filtered the dataset to include only these top-ranked features. Re-evaluating the classification models on this refined dataset, we observed that the accuracy of the Decision Tree model further increased, reinforcing its robustness and efficiency in prediction. Meanwhile, the accuracies of the kNN and Naive Bayes models remained unchanged. This step demonstrated the effectiveness of feature selection in enhancing model performance by focusing on the most informative data points and eliminating noise.

Comparison Table:

Model	Before Preprocessing	After Preprocessing	After selecting 15 Features
KNN	0.935	0.917	0.17
Tree	0.963	0.972	0.981
Naïve Bayes	0.917	0.917	0.917

Table 3.2.0 Comparison Table

CHAPTER 4
CONCLUSION AND FUTURE SCOPE

4.1 Conclusion:

This project aimed to predict political party affiliation based on voting records using classification models in the Orange data mining tool. After thorough exploration and preprocessing of the dataset, we identified the Decision Tree model as the most accurate predictor, consistently outperforming other classifiers. Through feature selection techniques, we pinpointed key voting issues that significantly influence party affiliation predictions.

The Decision Tree model's interpretability allowed us to understand the underlying decision-making process and gain insights into voting behavior. By refining the model and optimizing hyperparameters, we achieved even higher predictive accuracy, demonstrating the effectiveness of data-driven approaches in political analysis.

Our findings contribute to the understanding of political dynamics and highlight the importance of leveraging data mining techniques in political science research. Moving forward, future studies could explore more sophisticated modeling approaches and incorporate additional datasets to further enhance predictive accuracy and deepen our understanding of political behavior.

In conclusion, this project underscores the value of data-driven methodologies in elucidating complex political phenomena and provides valuable insights for researchers, policymakers, and political analysts seeking to understand and predict political party affiliation based on voting records.

4.2 Future Scope:

This project aimed to predict political party affiliation based on voting records using classification models in the Orange data mining tool. After thorough exploration and preprocessing of the dataset, the Decision Tree model emerged as the most accurate predictor, consistently outperforming other classifiers. Through feature selection techniques, key voting issues influencing party affiliation predictions were identified, contributing to a deeper understanding of political behavior. The interpretability of the Decision Tree model allowed for insights into the underlying decision-making process and voting behavior. By refining the model and optimizing hyperparameters, higher predictive accuracy was achieved, demonstrating the effectiveness of data-driven approaches in political analysis. These findings underscore the value of leveraging data mining techniques in political science research and highlight the importance of understanding political dynamics through empirical analysis.

Moving forward, future studies could explore more sophisticated modeling approaches and incorporate additional datasets to further enhance predictive accuracy and deepen the understanding of political behavior. Integration of external data sources, temporal analysis of voting patterns, and exploration of ensemble methods are potential avenues for future research. Additionally, ethical considerations regarding fairness, accountability, and privacy in predictive modeling applications in politics should be prioritized to ensure responsible and equitable use of predictive models. Collaboration with domain experts, political scientists, and policymakers could facilitate interdisciplinary research and enable the development of more informed and contextually relevant predictive models. Ultimately, the future scope of this project lies in advancing the capabilities and applicability of predictive modeling in political analysis, fostering interdisciplinary collaboration, and addressing ethical considerations to contribute to informed decision-making in political processes.

REFERENCES

[1] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32. DOI: 10.1023/A:1010933404324

[2] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). Springer.

[3] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning: with applications in R. Springer.

[4] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

[5] Platt, J. C. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In Advances in large margin classifiers (pp. 61-74). MIT Press.

[6] Rish, I. (2001). An empirical study of the naive Bayes classifier. In IJCAI 2001 workshop on empirical methods in artificial intelligence (Vol. 3, No. 22, pp. 41-46).

[7] Vapnik, V. (1995). The nature of statistical learning theory. Springer.

[8] Zhang, H. (2004). The optimality of naive Bayes. In Proceedings of the seventeenth international Florida artificial intelligence research society conference (pp. 562-567). AAAI Press.

SESHADRI RAO GUDLAVALLERU ENGINEERING COLLEGE

(An Autonomous Institute with Permanent Affiliation to JNTUK, Kakinada)
Seshadri Rao Knowledge Village, Gudlavalleru

Department of Computer Science and Engineering

Program Outcomes (POs)

Engineering Graduates will be able to:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions., component, or software to meet the desired needs.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

PSO1 : Design, develop, test and maintain reliable software systems and intelligent systems.

PSO2 : Design and develop web sites, web apps and mobile apps.

PROJECT PROFORMA

Classification of Project	**Application**	**Product**	**Research**	**Review**
	√			

Note: Tick Appropriate category

Data Mining Outcomes	
Course Outcome (CO1)	Describe fundamentals, and functionalities of data mining system and data preprocessing techniques.
Course Outcome (CO2)	Illustrate the major concepts and operations of multi dimensional data models.
Course Outcome (CO3)	Analyze the performance of association rule mining algorithms for finding frequent item sets from the large databases.
Course Outcome (CO4)	Apply classification algorithms to solve classification problems.
Course Outcome (CO5)	Use clustering methods to create clusters for the given data set.

Mapping Table

CS3509:DATAMINING															
Course Outcomes	Program Outcomes and Program Specific Outcome														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		SO1	SO2
CO1	1	1										1			
CO2	1											1			
CO3	2	3	2									2		1	
CO4	2	2	3	2								2		2	
CO5	1	2	3	1								2		1	

Note: Map each Data Mining out comes with Pos and PSOs with either 1 or 2 or 3 based on level of mapping as follows:

1-Slightly(Low) mapped 2-Moderately(Medium) mapped 3-Substantially(High) mapped