**All majors:**

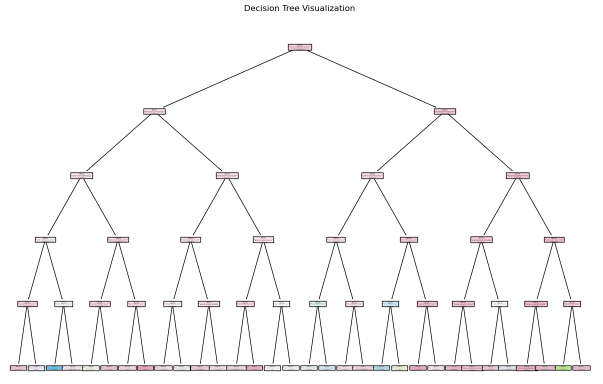
**Method 1: Decision Tree**

Steps:

* 1. Split the data into training (70%) and testing (30%) sets.
* 2. Trained a Decision Tree model with a maximum depth of 5.
* 3. Evaluated the model using the test set.

Results:

* - Accuracy: 0.1948.
* - Classification Report: Very low scores for all metrics like Precision and Recall.
* - Decision Tree Visualization: A tree diagram was generated to display nodes and branches based on input features.



**Method 2: Naive Bayes**

Steps:

* 1. Used 5-fold Cross-Validation.
* 2. Trained a Gaussian Naive Bayes model.

Results:

* - Accuracy per fold: [0.0475, 0.0369, 0.0299, 0.0202, 0.0418].
* - Mean Accuracy: 0.0352.

**Method 3: RandomForest**

Steps**:**

* The dataset was split into **80% training** and **20% testing**.

Results**:**

* Accuracy: 0.31
* Predicted: psychology

We see that the data is imbalanced:

A blue square with white text

Description automatically generated A blue square with white text

Description automatically generated

A graph of blue bars

Description automatically generated with medium confidence

**5 top\_majors**

**Majors**:

* Accounting (223 samples)
* Engineering (222 samples)
* Mass Communication (217 samples)
* Criminology (209 samples)
* Art (194 samples)

A graph of green rectangular bars

Description automatically generated with medium confidence

**Method 1: Decision Tree**

**Approach:**

* A Decision Tree Classifier was trained with a **maximum depth of 5**.
* The dataset was split into **70% training** and **30% testing**.
* The model's performance was evaluated on the test set.

**Results:**

* **Accuracy**: 0.265
* **Classification Report**: Low Precision, Recall, and F1-scores across all classes.
* **Confusion Matrix**: Significant misclassifications, showing poor separation of classes.

A diagram of a decision tree

Description automatically generated

**Method 2: Gaussian Naive Bayes**

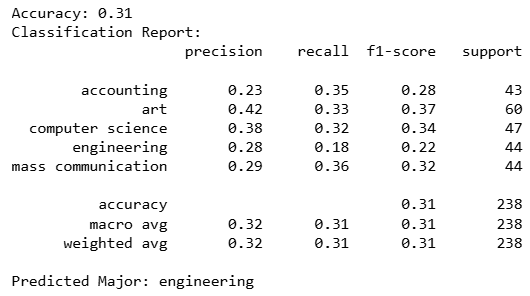
**Approach:**

* Gaussian Naive Bayes was evaluated using **5-fold Cross-Validation**.

**Results:**

* **Accuracy per fold**: [0.295, 0.286, 0.267, 0.422, 0.258]
* **Mean Accuracy**: 0.306

**Method 3: RandomForest**



**Approach:**

* The dataset was split into **80% training** and **20% testing**.

**Results:**

* **Accuracy** : 0.31
* **Predicted** : engineering

**Combined Results:**

| * **Method** | * **Accuracy per Fold** | * **Mean Accuracy** | * **Additional Observations** |
| --- | --- | --- | --- |
| * Decision Tree | * - | * **0.265** | * Poor classification; significant overlap in predictions. |
| * Gaussian Naive Bayes * RandomForest | * [0.295, 0.286, 0.267, 0.422, 0.258] * - | * **0.306** * **0.31** | * Poor classification * Poor classification |

**General Observations:**

* **Gaussian Naive Bayes** had a similar accuracy to Logistic Regression but exhibited higher variability across folds.
* **Decision Tree** was the poorest performer, possibly due to insufficient depth or limited separability in the dataset.

**All major with SMOT**

Techniques Used

* **SMOTE (Synthetic Minority Oversampling Technique):**  
  SMOTE was applied to address the issue of class imbalance in the dataset. Class imbalance occurs when one class significantly outnumbers the other(s), leading to biased predictions favoring the majority class. SMOTE works by generating synthetic samples for the minority class rather than duplicating existing samples.

Why SMOTE?

* + Preserves Variability: SMOTE generates new synthetic samples by interpolating between existing ones, introducing more variability in the minority class data. This avoids overfitting, which can occur with simple duplication of data.
  + Avoids Information Loss: Unlike undersampling, which reduces the size of the majority class, SMOTE retains all data points from the majority class, ensuring no information is lost.
  + Improves Model Generalization: By creating a balanced dataset with diverse samples, models are better able to generalize and perform well on unseen data.
* **Shuffle:**  
  After applying SMOTE, the dataset was shuffled to randomize the order of samples. This prevents models from learning patterns based on the order of the data rather than the underlying features.
* Models Used:  
  The following models were tested for classification:
  + Random Forest
  + Gradient Boosting
  + Decision Tree
  + Naive Bayes
  + K-Nearest Neighbors (KNN)
  + Artificial Neural Network (ANN)
* Data Splitting and Evaluation:  
  The dataset was divided into training and testing sets using train\_test\_split. Performance metrics included Accuracy, the Confusion Matrix, and the Classification Report for a comprehensive evaluation.

2. Results

* Model Comparisons:  
  The accuracy scores of the tested models were as follows:
  + Random Forest: 0.873
  + Gradient Boosting: 0.296
  + Decision Tree: 0.780
  + Naive Bayes: 0.091
  + KNN: 0.801
  + ANN: 0.647
* Best Model:  
  The Random Forest model achieved the highest accuracy (87.3%), making it the most effective model for this dataset.

3. Final Outcome

* Selected Model:  
  The Random Forest model was chosen due to its superior performance and robustness.
* Final Prediction:  
  Using the balanced dataset generated with SMOTE, the Random Forest model provided accurate predictions, achieving an overall accuracy of 87.3% on the testing data.

A screenshot of a computer

Description automatically generated

**A white paper with black text

Description automatically generated**

**A graph of a bar chart

Description automatically generated with medium confidence**

A graph of a curve

Description automatically generated with medium confidence  
  
"This chart illustrates the efficiency of different models in classifying data. Each curve represents a model, and the closer the curve is to the top-left corner, the better the model performs in accurately classifying data. The Area Under the Curve (AUC) provides a numerical summary of the performance: for example, the Random Forest model has an AUC of 0.99, indicating excellent performance, while the Naive Bayes model has an AUC of 0.67, which is comparatively lower. The purpose of this chart is to compare the models and identify the most accurate one."