Report on Iris Dataset Classification with Imbalance Handling

To analyze the performance of various machine learning models on the Iris dataset, focusing on the impact of class imbalance and different imbalance handling techniques, I followed these steps:

1. **Data Loading and Preprocessing:** Loads the Iris dataset, handles missing values, converts data types, removes duplicates, and identifies and handles outliers.
2. **Exploratory Data Analysis:** Visualizes data distribution and correlations between features using box plots, pair plots, and heatmaps.
3. **Feature Selection:** Selects two features ('sepal\_width' and 'petal\_width') for classification due to their relatively lower correlation with other features, reducing redundancy and potential overfitting.
4. **Dataset Imbalance Creation:** Generates three datasets with varying levels of imbalance: slightly imbalanced (50:30:20), moderately imbalanced (70:20:10), and severely imbalanced (85:10:5).
5. **Model Training and Evaluation:** Trains and evaluates different models (Dummy Classifier, Logistic Regression, Decision Tree, Random Forest, and SVM) on each imbalanced dataset using various imbalance handling techniques. These techniques include SMOTE, Random Oversampling, Borderline-SMOTE, Random Undersampling, Tomek Links, Cluster Centroids, NearMiss, and using the original data (no balancing).
6. **Performance Comparison:** Compares model performance across different imbalance handling techniques using metrics like accuracy, precision, recall, and F1-score.

The following are my achieved results of various machine learning models and imbalance handling techniques applied to three different versions of the Iris dataset with varying levels of imbalance.

Results and Analysis

1. **Slightly Imbalanced Dataset:**



**Summary for Slightly Imbalanced Dataset**

* **Best Models:** SVM and Decision Tree.
* **Recommended Techniques:** Random Oversampling and SMOTE (k=3).
* **Top Results:**
* SVM achieved 93.33% accuracy with Random Oversampling.
* Decision Tree achieved 90% accuracy with Random Oversampling and SMOTE.
* Logistic Regression performed well with 83.33% to 90% accuracy, depending on the technique used.

1. **Moderately Imbalanced Dataset:**



**Summary for Moderately Imbalanced Dataset**

* **Best Models:** Logistic Regression and SVM.
* **Recommended Techniques:** SMOTE (k=3) and Random Oversampling.
* **Top Results:**
* Logistic Regression achieved 95.45% accuracy with SMOTE and Random Oversampling.
* SVM performed consistently with 90-91% accuracy across multiple techniques.
* Random Forest achieved up to 86% accuracy but was less consistent on minority class predictions.

1. **Severely Imbalanced Dataset:**



**Summary for Slightly Imbalanced Dataset**

* **Best Models:** SVM and Decision Tree
* **Recommended Techniques:** Random Oversampling and SMOTE (k=3)
* **Top Results:**
* SVM achieved 93.33% accuracy with Random Oversampling.
* Decision Tree achieved 90% accuracy with both Random Oversampling and SMOTE.
* Logistic Regression delivered consistent performance, ranging from 83.33% to 90% accuracy, depending on the imbalance handling method.