Work Done:

The primary objective of this project is to predict hospital appointment no-shows through machine learning. The initial phase involved data loading and cleaning, addressing datatype issues, and removing anomalous entries. EDA, powered by Seaborn and Matplotlib, provided crucial insights into feature distributions, correlations, and class imbalances.

Decision Tree Modeling:

The core of the analysis centered around a single-layer decision tree. This model exhibited a 73% accuracy in predicting appointment no-shows. The classification report highlighted precision, recall, and F1-score metrics, offering a comprehensive view of the model's performance. The confusion matrix detailed true positives, true negatives, false positives, and false negatives.

Naive Bayes Modeling:

In addition to the decision tree, a Gaussian Naive Bayes model was employed. Despite its simplicity, Naive Bayes achieved a respectable accuracy of 77.92%. The model's classification report illustrated precision, recall, and accuracy, providing valuable insights into its predictive capabilities.

Conflicts:

Imbalanced Classes:

Issue: The dataset exhibits a significant class imbalance between appointment no-shows and appointments kept.

Impact: Imbalanced classes can lead to biased models, where the algorithm may prioritize the majority class, resulting in reduced performance in predicting the minority class.

Mitigation: Techniques such as oversampling the minority class, undersampling the majority class, or using synthetic data generation methods is applied to address the imbalance issue.

Limited Model Exploration:

Issue: Only a single-layer decision tree and Naive Bayes models were explored.

Impact: The performance of these models might not represent the best achievable accuracy, and other models may offer better predictive capabilities.

Mitigation: I have to explore a variety of machine learning models, including ensemble methods (e.g., Random Forest, XGBoost) and linear models (e.g., Logistic Regression), to identify the most suitable algorithm for the dataset.

Next Steps:

Ensemble Methods:

Explore ensemble methods, such as combining the decision tree with other models, to potentially improve overall accuracy.

Feature Importance Analysis:

Conduct a thorough analysis of feature importance within the decision tree. Understanding which features contribute the most can guide further model refinement.

Model Comparison:

Extend the analysis by training and evaluating additional models, such as Random Forest and Logistic Regression, to identify the most effective approach.

Web App Development:

Move towards deploying the decision tree and Naive Bayes models in a user-friendly web app. This step would facilitate broader accessibility and application.