**Assignment-15-1**

**Comparative Report: XGBoost Model Performance**

This report presents a comparative evaluation of two Jupyter notebooks applying the XGBoost algorithm on different datasets. The notebooks are:

* XGBoost\_Assignment\_15\_1A.ipynb (Loan Dataset)
* XGBoost\_Cancer\_Prediction3.ipynb (Cancer Dataset)

Both notebooks implement the XGBoost classifier with identical hyperparameters. The models were evaluated using two metrics: Accuracy and ROC-AUC score. Below is a comparative summary of their performance.

# Model Performance Comparison

|  |  |  |
| --- | --- | --- |
| Notebook | Accuracy (%) | ROC-AUC (%) |
| XGBoost\_Assignment\_15\_1A | 96.49 | 96.48 |
| XGBoost\_Cancer\_Prediction3 | 96.49 | 96.48 |

# Observations

Both notebooks achieved identical performance, suggesting similar data quality and model effectiveness.

The cancer notebook is more modular and includes assertion-based validation, which is beneficial in educational or automated testing contexts.

The loan notebook lacked early stopping and validation parameters initially, but it used consistent model settings.

# Conclusion

XGBoost proved effective across both use cases, yielding high accuracy and AUC scores. Best practices such as early stopping, evaluation metric tuning, and modular code structure (as seen in the cancer notebook) can enhance the reliability of model training pipelines.

There were some errors in both notebooks

Original code:

#clf.fit(X\_train, y\_train, early\_stopping\_rounds=20, eval\_metric="auc", eval\_set=eval\_set, verbose=False)

Corrected code:

clf.fit(X\_train, y\_train, eval\_set=eval\_set, verbose=False)

It was a version difference.

Links for the notebooks in Colab:

Link para [XGBoost\_Assignment\_15\_1A](https://colab.research.google.com/drive/1Ikg3T86cgj6WjjPxtah1GzjxZvivnVC8?usp=sharing)

Link para [XGBoost\_Cancer\_Prediction3](https://colab.research.google.com/drive/1LjazMFu7FaIUO1ibGJaSPlWsUZOLKiik?usp=sharing)