### **What You’ve Achieved**

* Moved to PySpark for production-scale processing and modeling.
* Assembled all engineered features for each provider, handled nulls correctly.
* Verified class imbalance and feature distributions in Spark.
* Trained a tree-based model, identified key features, and achieved high baseline performance.

## **Documentation for Reports/Notebook**

Copy the below for your report or notebook summary:

# **Feature Profiling & Model Readiness Check**

## **PySpark Implementation Rationale**

* Initial engineering was done in pandas for flexibility;
* Switched to PySpark for scalability and efficient ML pipelines as data/model size grows;
* All engineered features were imported and validated in Spark DataFrames for consistency and production readiness.

## **Key Checks & Actions**

* **Class balance:** 9.3% fraud, 90.7% non-fraud (matches business reality)
* **Null handling:** All features filled appropriately (0 for engineered features)
* **Feature assembly:** Used Spark VectorAssembler for ML pipeline input
* **Model training:** RandomForest baseline—AUC = 0.954
* **Feature importances:** Most important features are claim volume, total/maximum reimbursement, total deductible, and high-diagnosis claims—matches business knowledge
* **No data leakage or errors in pipeline**

## **Next Steps**

* Fine-tune model parameters and validate on stratified splits
* Consider SHAP or Spark ML explainability for detailed feature impact
* Prepare for final model selection, threshold tuning, and business-ready results