**Problem Statement and Goal**

\* Predicting flight delays as a multi-class classification problem

\* Switched to three classes instead of earlier 4 classes, removing granularity of whether a flight had a specific arrival delay or a departure delay.

\* Class 0: On-time flights

\* Class 1: Either arrival or departure delayed

\* Class 2: Both arrival and departure delayed

**Evaluation Metrics**

\* Primary: F1 Score

\* Secondary: PR AUC and ROC AUC for overall performance

**Key Objectives**

\* Correctly predict delays without sacrificing accuracy of on-time predictions

\* Balance between: a) High recall for delays - identifying delays correctly b) High precision for delays - not misclassifying on-time flights as delayed.

**Methodology Rationale**

\* Experimented with F2 scores

\* Switched back to focus on F1 Score to: a) Decrease false positives for delayed flights, especially Class 2 b) Improve accuracy of on-time flight predictions - Class1 c) Increase precision for Class 2 and Class 1

**Data Sampling Strategy**

\* Applied SMOTE with SMOTETomek

\* Sampling ratios (applied this change after our 1:1):

\* Class 0: 1x (baseline)

\* Class 1: 1.25x

\* Class 2: 2.50x

\*  Before SMOTE: Class 0 (77.75%), Class 2 (14.61%), Class 1 (7.64%)

\* After SMOTE: Class 1 (33.67%), Class 0 (33.24%), Class 2 (33.09%)

**Model Evaluations (F1 focus on Class 2)**

\* Ensemble Classifiers showed better performance

\* Evaluated all ensemble Classifiers by adjusting threshold for Class 2 to optimize on F1 score. Also applied class weights from suggestions. Captured results of a few good ones below.

**Random Forest Classifier:**

\* Best class weight multipliers applied to balanced class weights:

{0: 0.4, 1: 1.0, 2: 4.0}

\* This leads to classifier class\_weights of:

{0: 0.401118794671553, 1: 1.9799698690504237, 2: 4.029525121704785}

\* Out-of-Bag Score: 0.8987

\* Validation/Development F1 Score: 0.7162

\* Validation/Development Accuracy: 0.7626

\* F1 Scores per Class with weights:

Class 0: F1 Score = 0.8049, Class 1: F1 Score = 0.0250, Class 2: F1 Score = 0.3547 (class2 precision = 0.27, recall = 0.51)

\* Previously, F1 Scores per Class with NO weights:

Class 0 = 0.87, Class 1 and Class 2 = 0.01 and 0.04, Class 3 = 0.25

(class3 precision = 0.40, recall = 0.18)

\* Macro-Averaged F1 Score: 0.3949

\* Weighted-Averaged F1 Score: 0.6795

\* Test Accuracy: 0.6787

**XG Boost:**

\* Class weight multipliers applied to balanced class weights: {0: 0.5, 1: 1.0, 2: 4.0}

\* Validation F1 Score: 0.6991

\* Validation Accuracy: 0.7034

\* F1 Scores per Class:

Class 0: F1 Score = 0.8044 Class 1: F1 Score = 0.0439 Class 2: F1 Score = 0.3653 (class2 precision = 0.28, recall = 0.53)

\* Previously, F1 Scores per class:

Class 0: 0.88, Class 1: 0.02, Class 2: 0.00, Class 3: 0.18 (class3 precision = 0.52, recall = 0.11)

\* Macro-Averaged F1 Score: 0.4045

\* Weighted-Averaged F1 Score: 0.6822

\* Test Accuracy: 0.6779

**Stacking Classifier (New addition from feedback):**

\* Base models: Random Forest and XG Boost

\* Meta-classifier: Logistic Regression with PCA (80% variance retained)

\* Validation F1 Score: 0.7277

\* Validation Accuracy: 0.7627

\* F1 Scores per Class:

Class 0: F1 Score = 0.82 Class 1: F1 Score = 0.0595 Class 2: F1 Score = 0.3731 (class 2 precision = 0.30, recall = 0.49)

\* Macro-Averaged F1 Score: 0.4185

\* Weighted-Averaged F1 Score: 0.6988

\* Test Accuracy: 0.7015

**Voting Classifier - Ensemble of XG Boost and Random Forest**

\* Soft voting: (4,1) where XG Boost is given more weight since it showed higher recall for class2

\* Comparable performance to Stacking Classifier above

**Hybrid Ensemble of Stacking and Voting Classifier - created as a custom hybrid classifier**

\* Comparable performance to both Stacking and Voting Classifiers from above