Defect Prediction Summary

Motivation

I picked CM1 and JM1 because they're simple NASA PROMISE benchmarks. CM1 is small (498 modules, about 10 % defects) so it's a tough, skewed problem. JM1 is much bigger (13 k modules, ~16 % defects) and lets me see how the same pipelines scale and whether extra data boosts performance.

Model and Feature Choices

I went with three models. Logistic Regression gives me a quick interpretable baseline. Random Forest can grab non-linear interactions that LR misses. Gaussian NB is a lightweight sanity check. I used `SelectKBest` with `k = 15` to drop noise while keeping most of the signal and speeding things up.

Hyperparameter Search

I tuned the Random Forest pipeline on JM1 with 40 random trials:

- SMOTE neighbors 3 or 5
- n_estimators 100 to 600
- max_depth None / 10 / 20
- min_samples_leaf 1 or 3

The search finished in about five minutes on my laptop.

Summary Metrics

acc	prec	rec	f1	roc_auc	pr_auc	set	model
0.750000	0.200000	0.500000	0.285714	0.682222	0.225887	CM1	LogReg
0.780000	0.166667	0.300000	0.214286	0.643889	0.179233	CM1	RF
0.750000	0.058824	0.100000	0.074074	0.536667	0.127429	CM1	GNB
0.723211	0.303299	0.567696	0.395368	0.723959	0.360287	JM1	LogReg
0.836047	0.481707	0.375297	0.421896	0.801787	0.438474	JM1	RF
0.825445	0.417355	0.239905	0.304676	0.705738	0.313388	JM1	GNB
0.772435	0.359813	0.548694	0.434619	0.764308	0.391687	JM1	RF_tuned

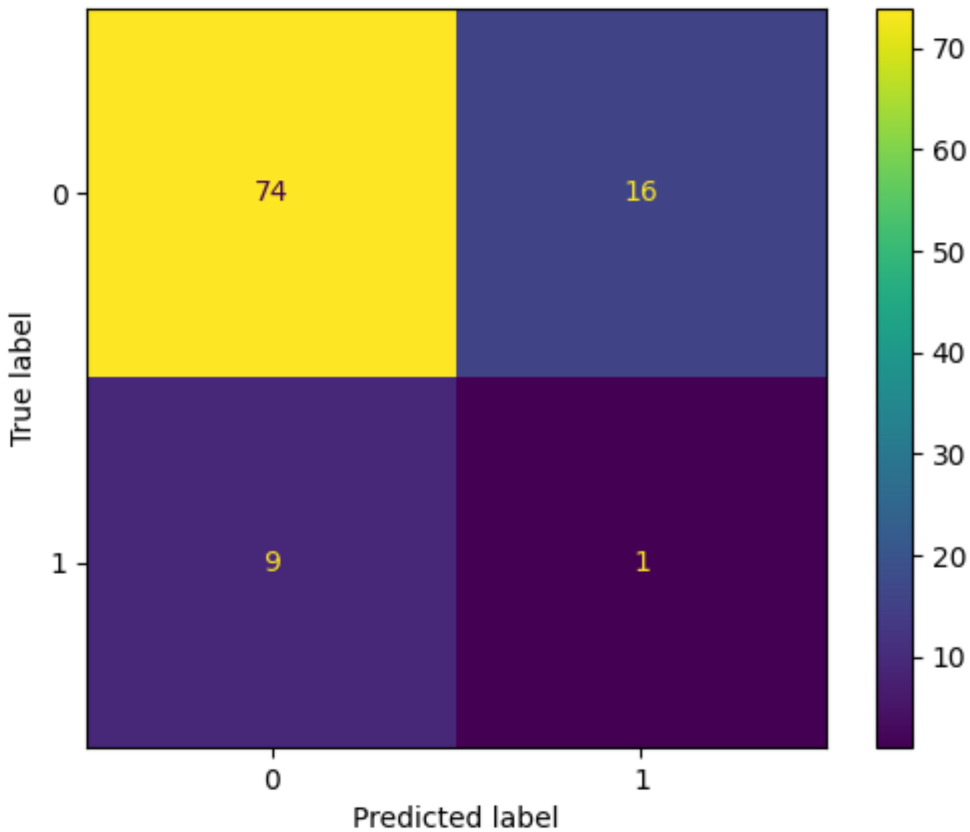
Related Work

Oueslati & Manita (2024) applied several classifiers—including Logistic Regression optimized via a Fractional Chaotic Grey Wolf Optimizer, Random Forest, and SVM—on CM1 and JM1. On CM1, their tuned Logistic Regression (LR-FCGWO) achieved *92.98% accuracy* and *F1-score 93.28%*, slightly outperforming Random Forest (92.43% acc., F1=92.72%). On JM1, Random Forest led with *91.70% accuracy* and *F1-score 91.72%*, while LR-FCGWO scored 87.88%/87.22% (acc./F1). These results provide strong baselines for comparison with our own metrics.

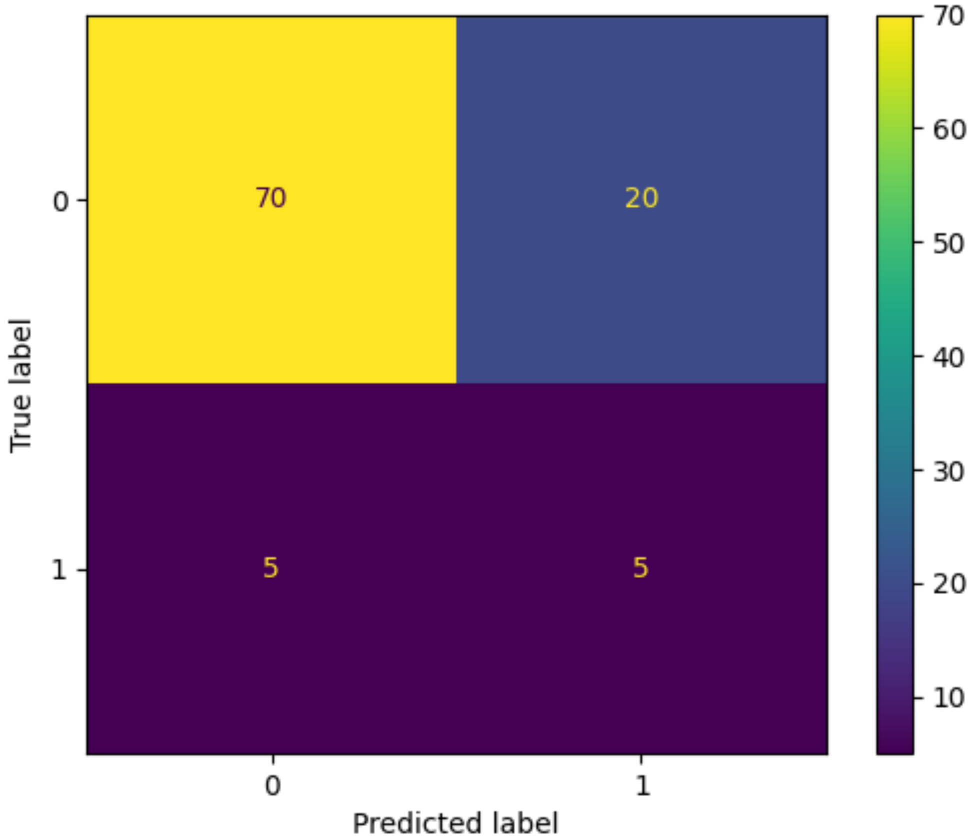
Reference: Oueslati, H. & Manita, M. (2024). *Enhanced Software Defect Prediction Using Fractional Chaotic Grey Wolf Optimizer*. Proceedings of the ENASE Conference.

Confusion Matrices

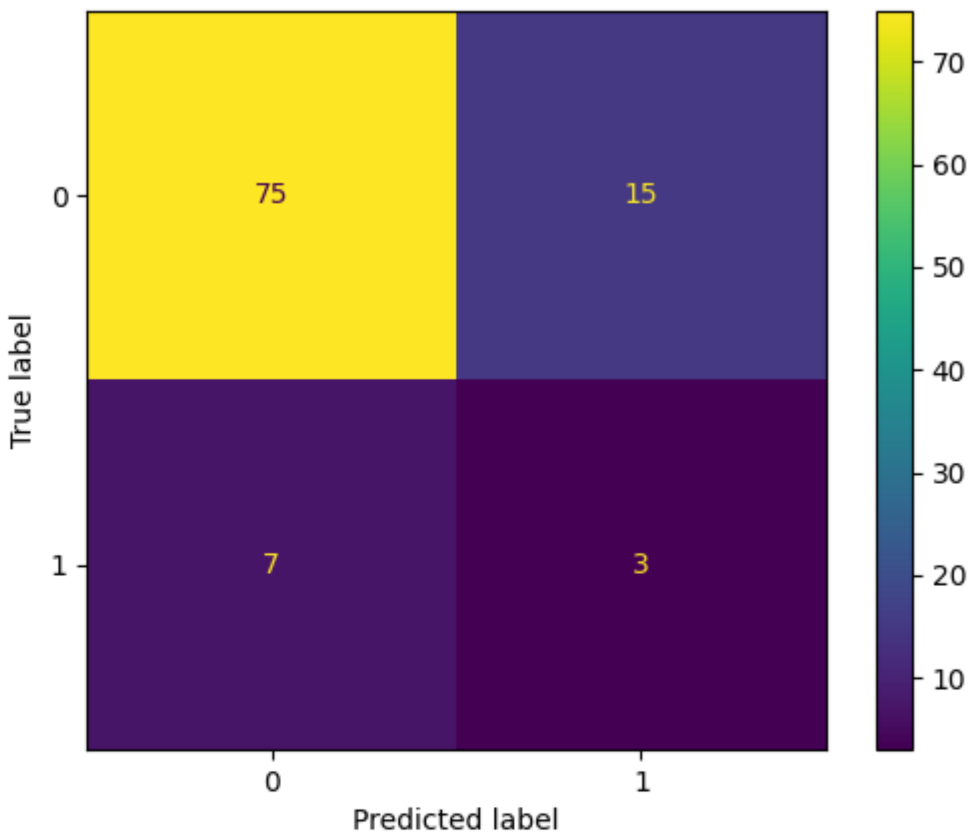
CM1_GNB_confusion.png



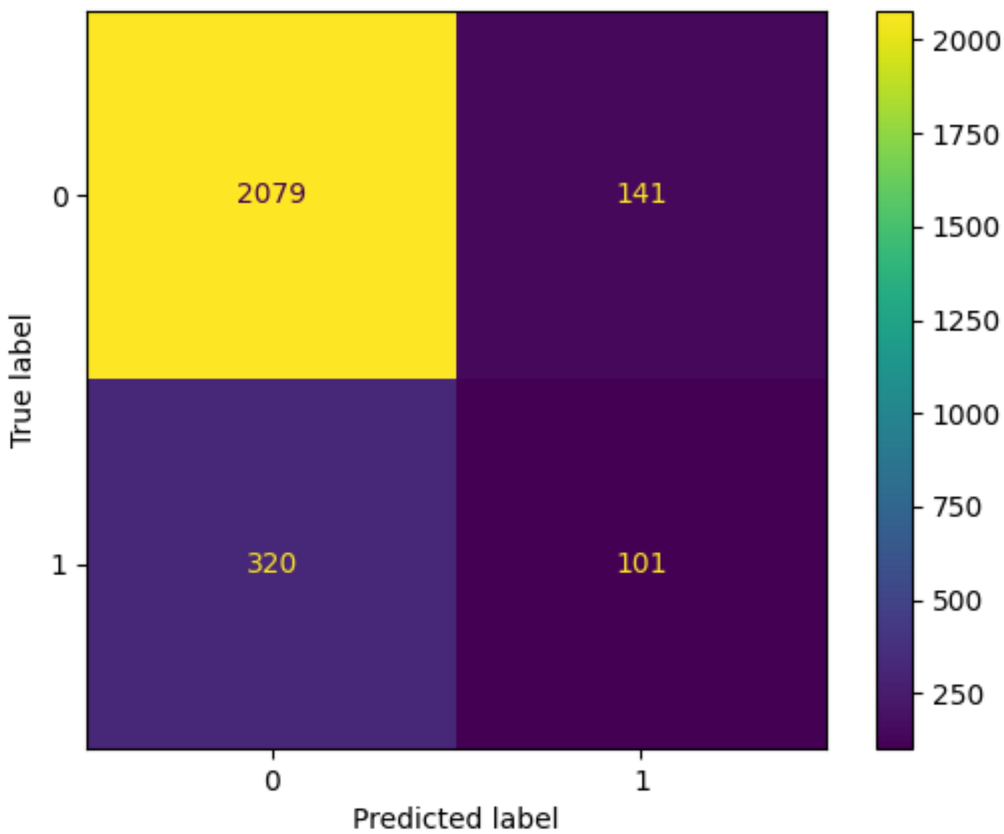
CM1_LogReg_confusion.png



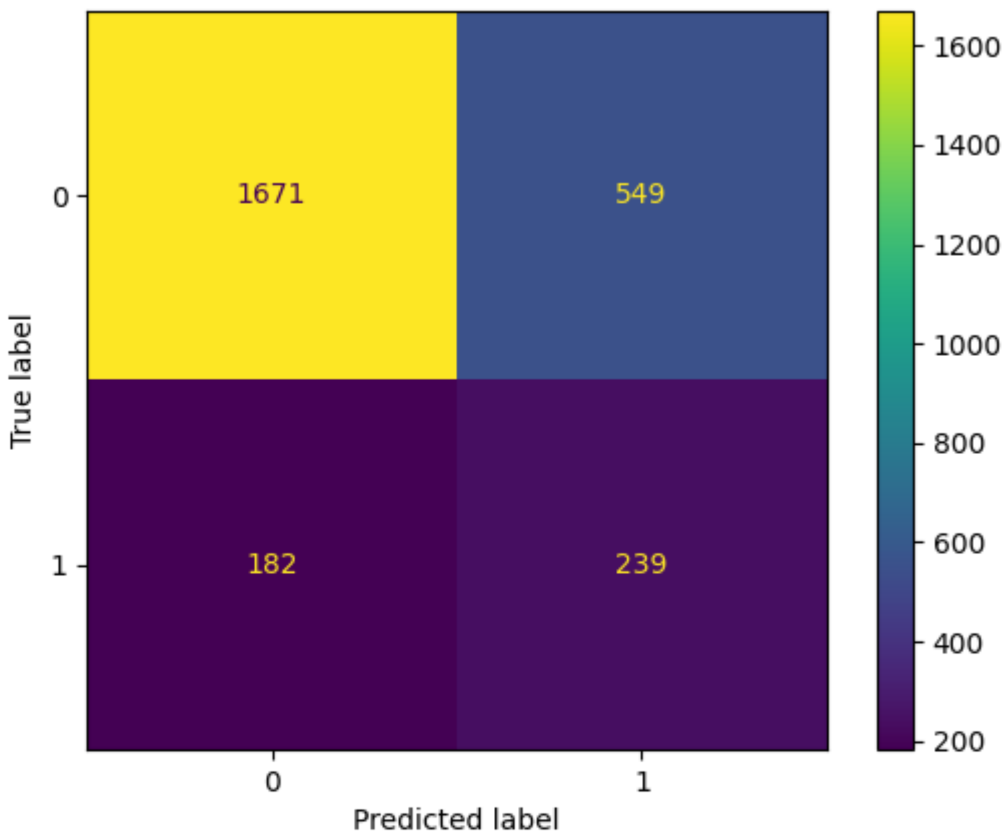
CM1_RF_confusion.png



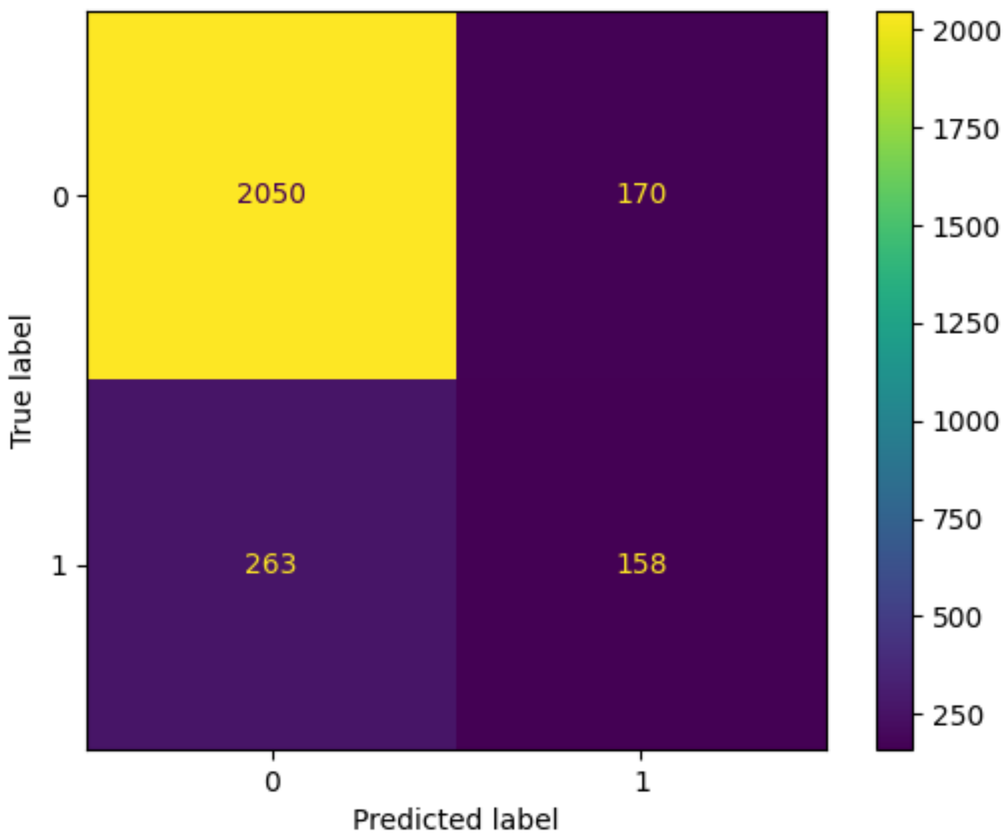
JM1_GNB_confusion.png



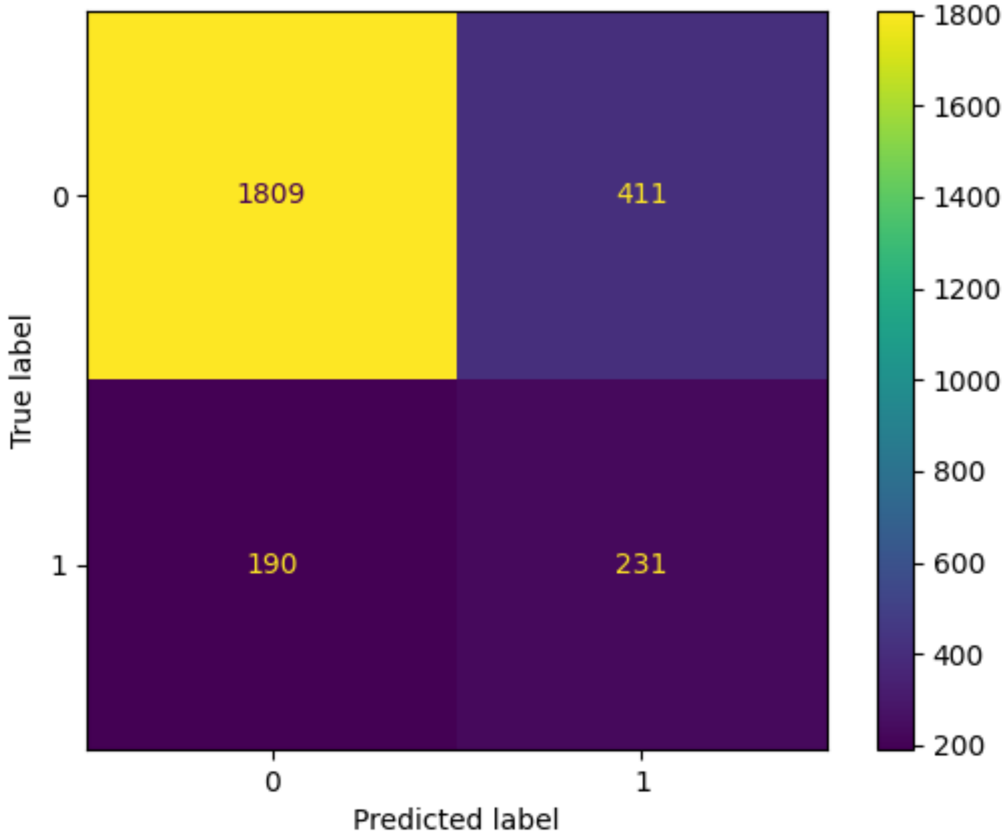
JM1_LogReg_confusion.png



JM1_RF_confusion.png

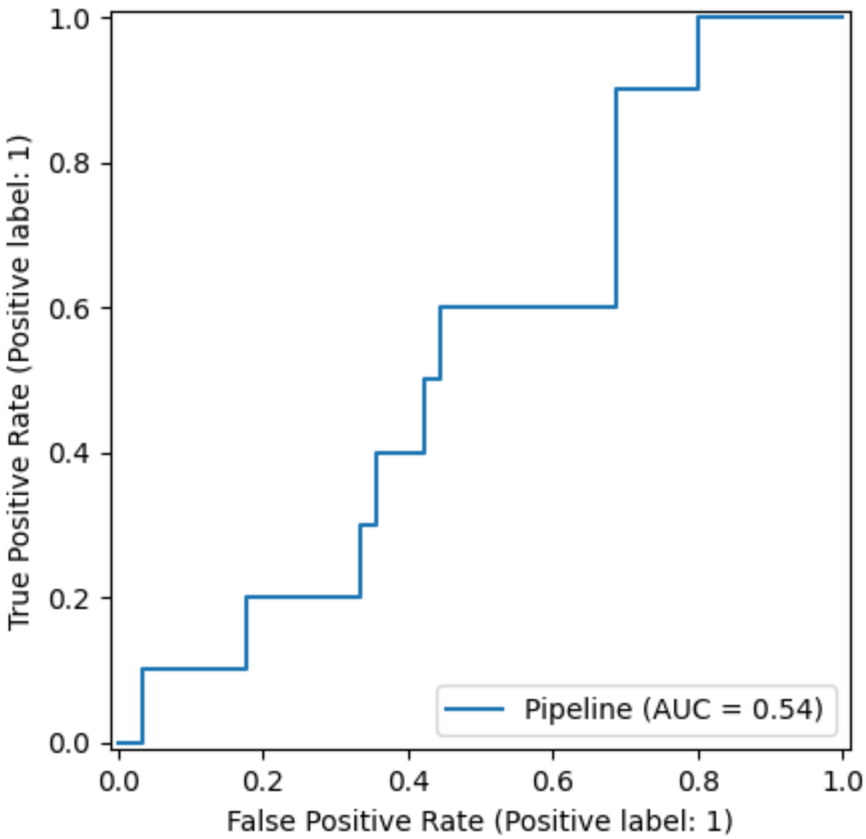


JM1_RF_tuned_confusion.png

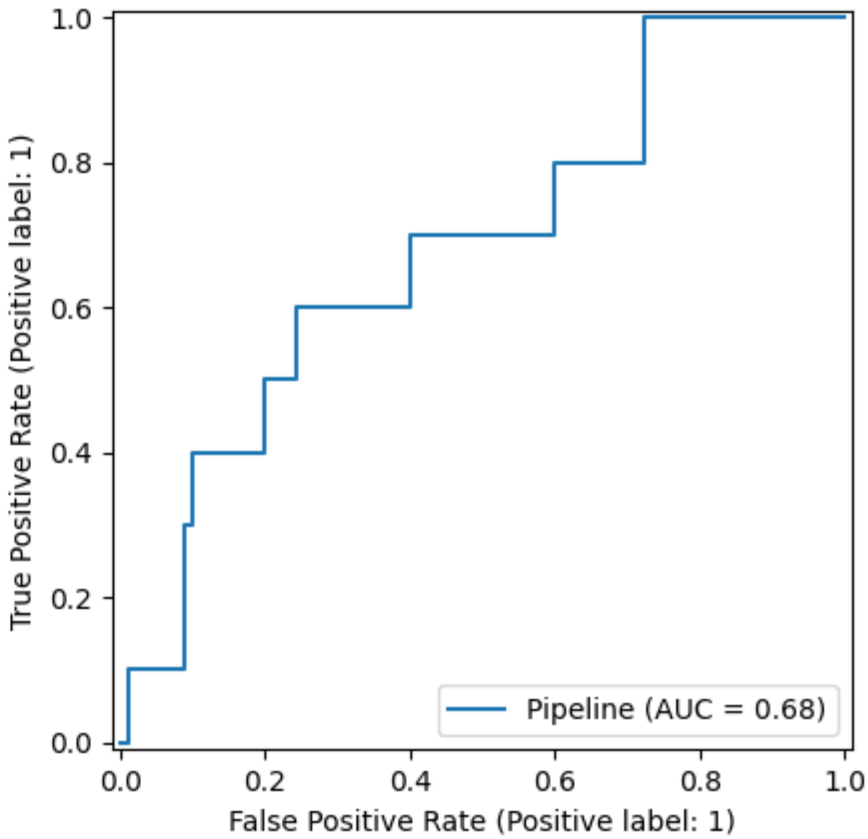


ROC Curves

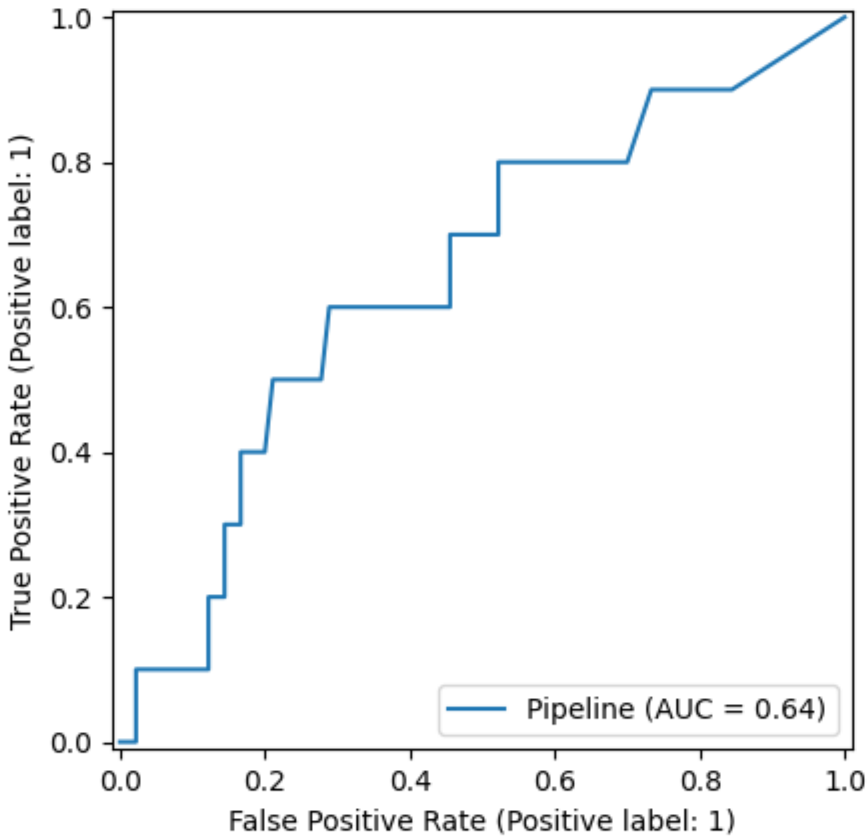
CM1_GNB_roc.png



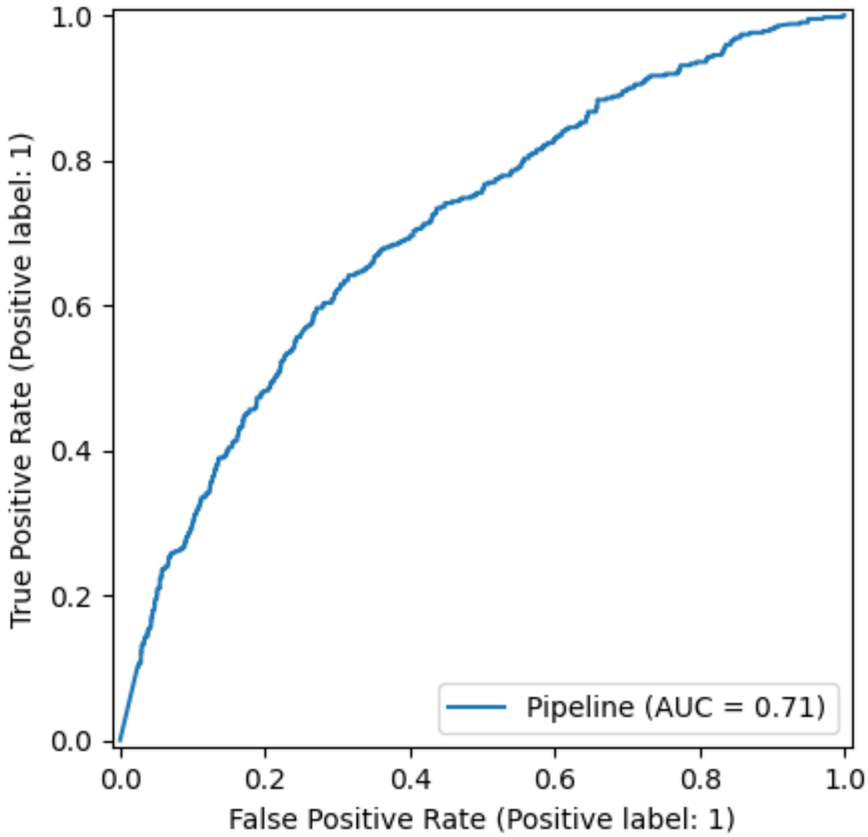
CM1_LogReg_roc.png



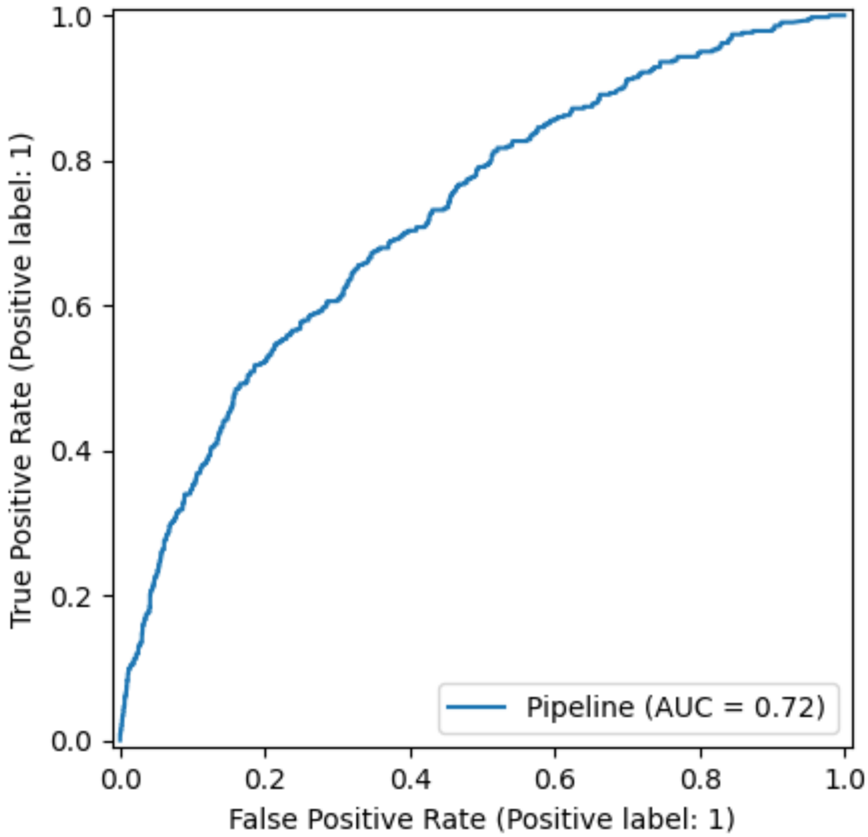
CM1_RF_roc.png



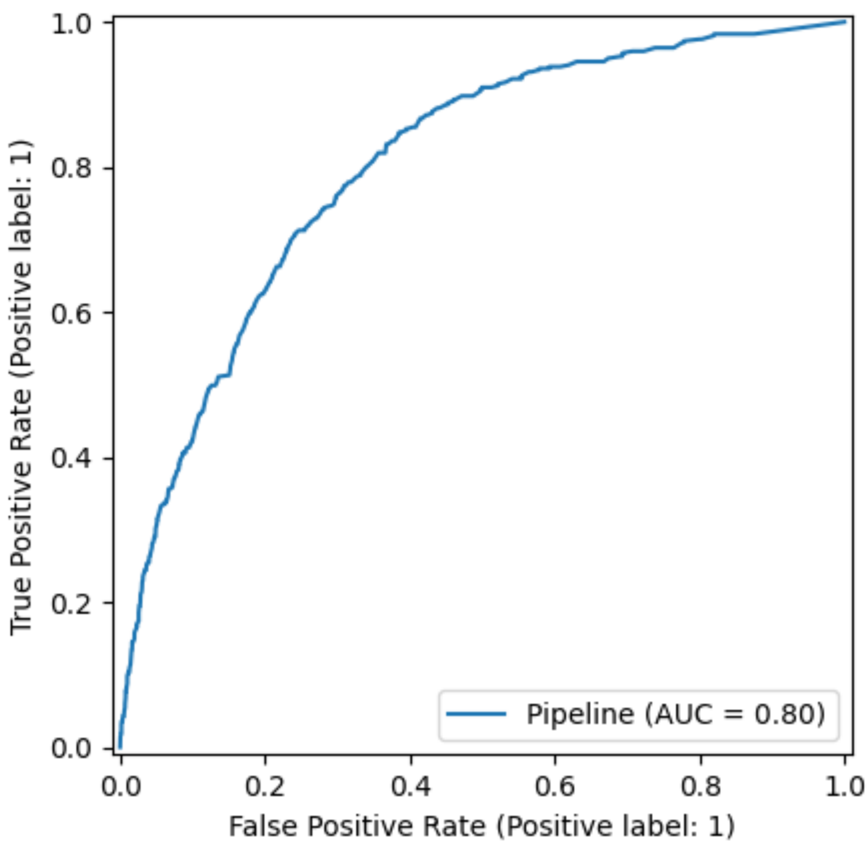
JM1_GNB_roc.png



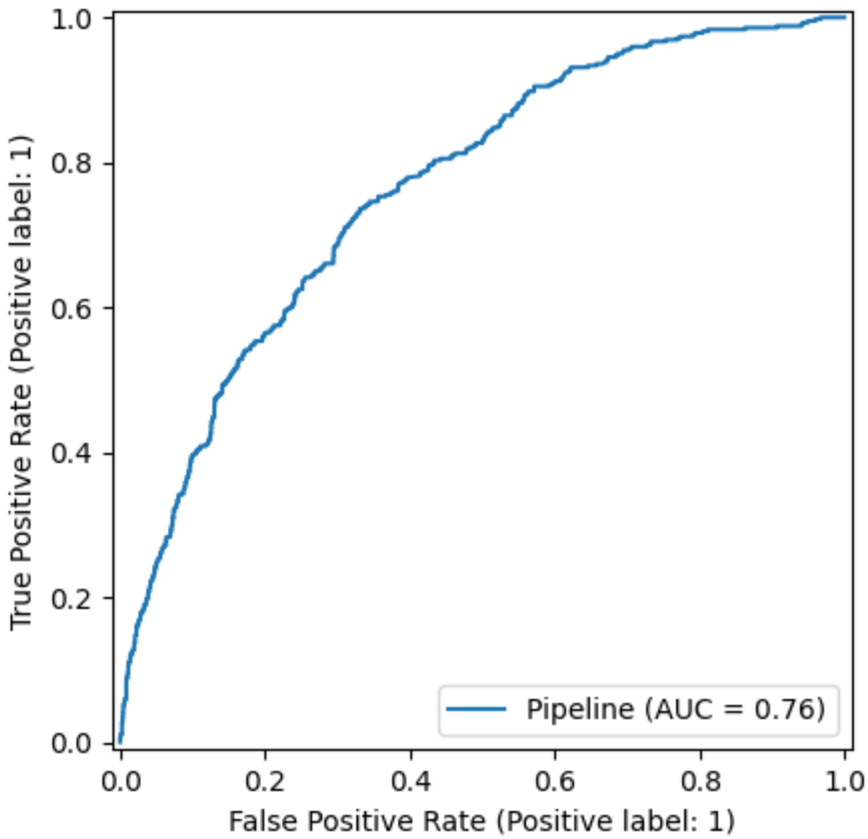
JM1_LogReg_roc.png



JM1_RF_roc.png

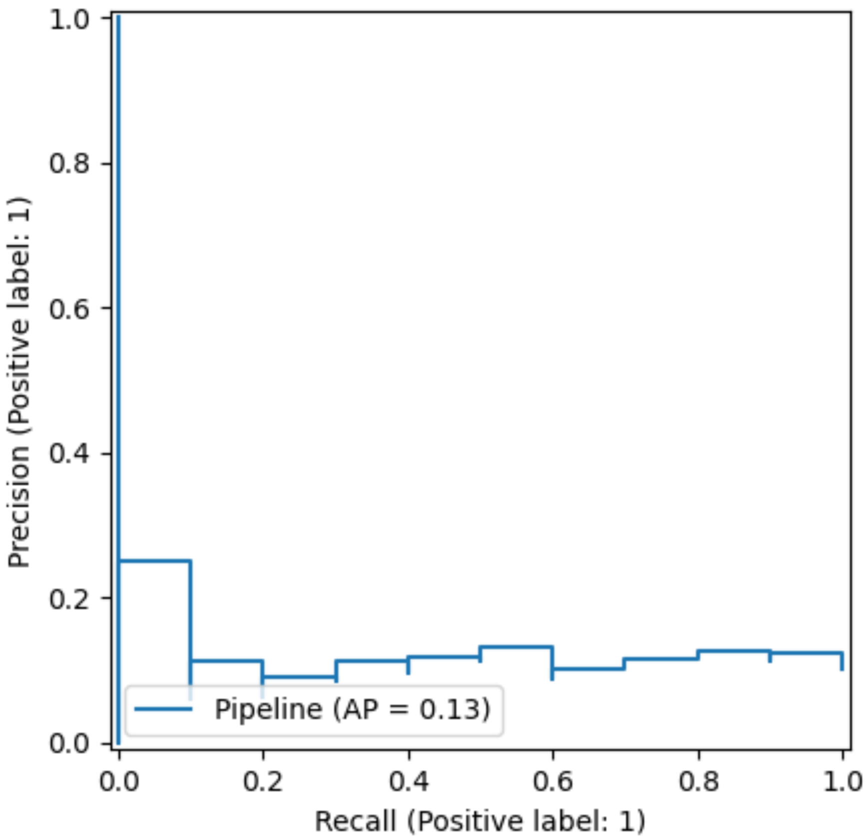


JM1_RF_tuned_roc.png

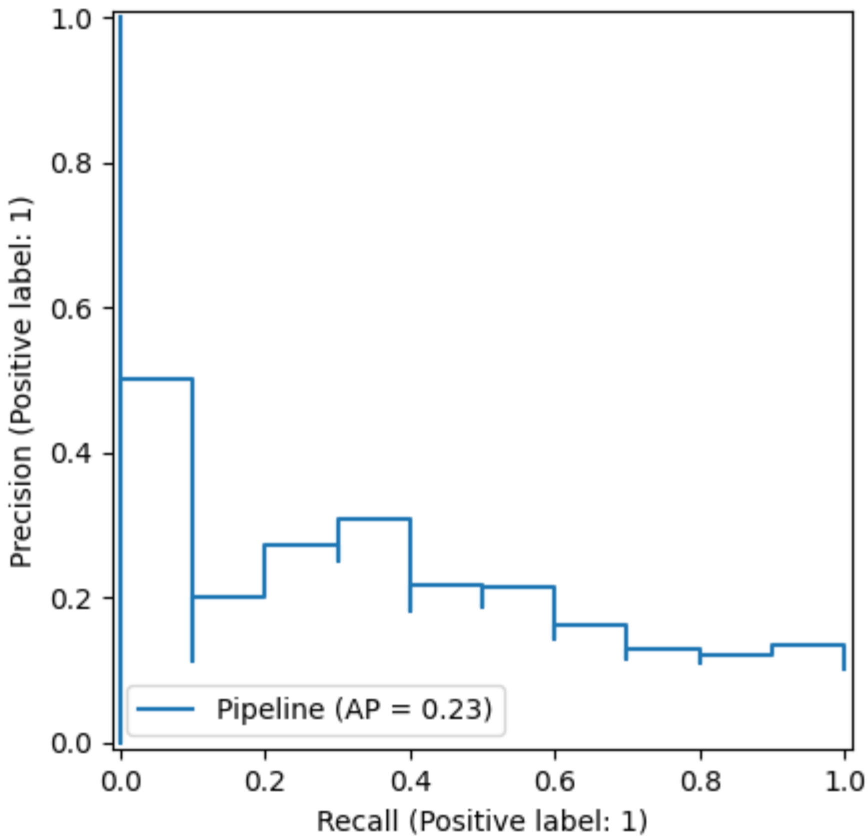


Precision-Recall Curves

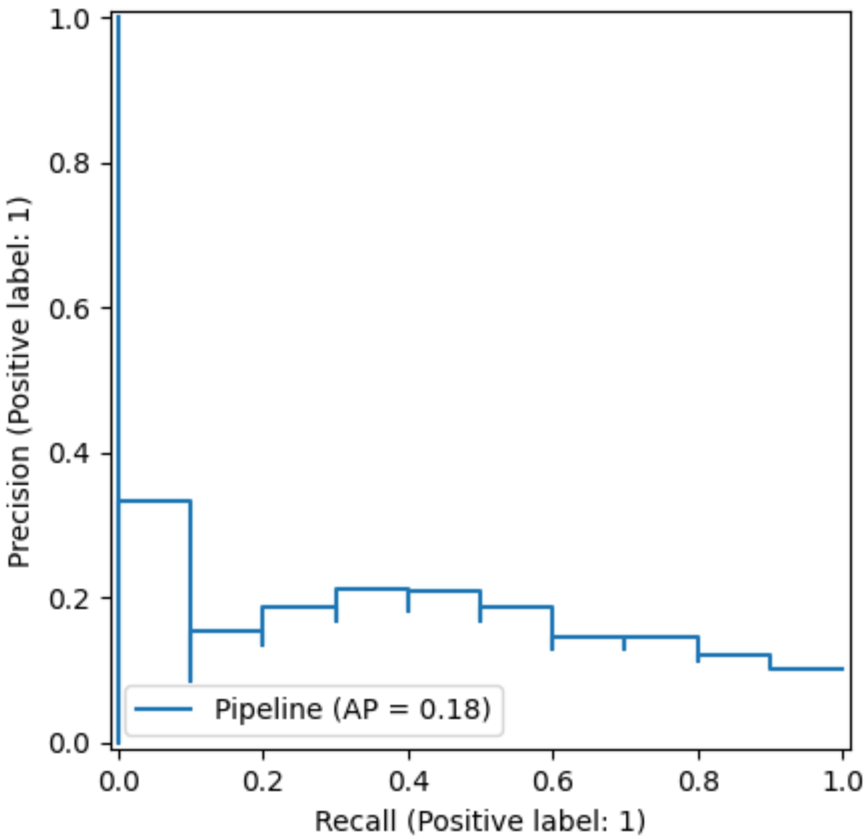
CM1_GNB_pr.png



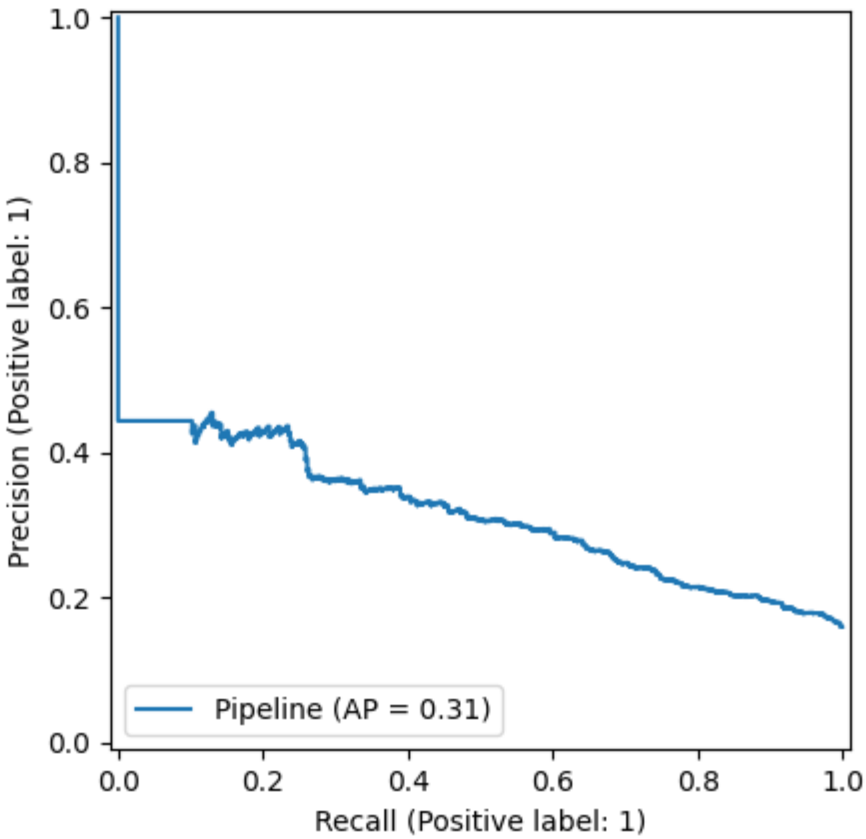
CM1_LogReg_pr.png


1.0
0.8
0.6
0.4
0.2
0.0
Precision (Positive label: 1)
Pipeline (AP = 0.23)
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Positive label: 1)

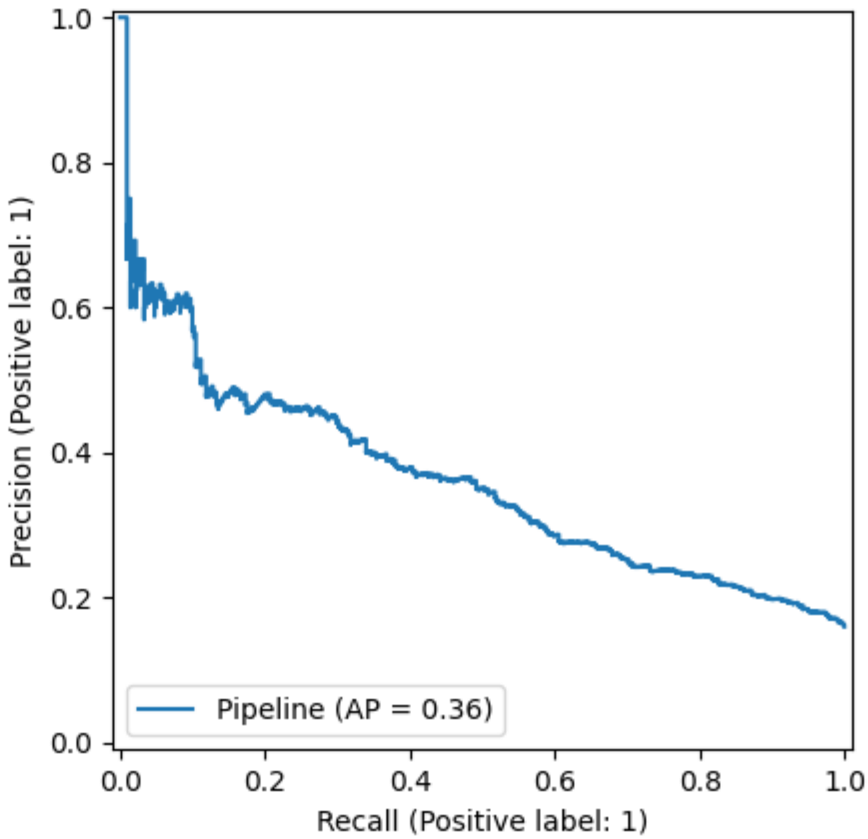
CM1_RF_pr.png



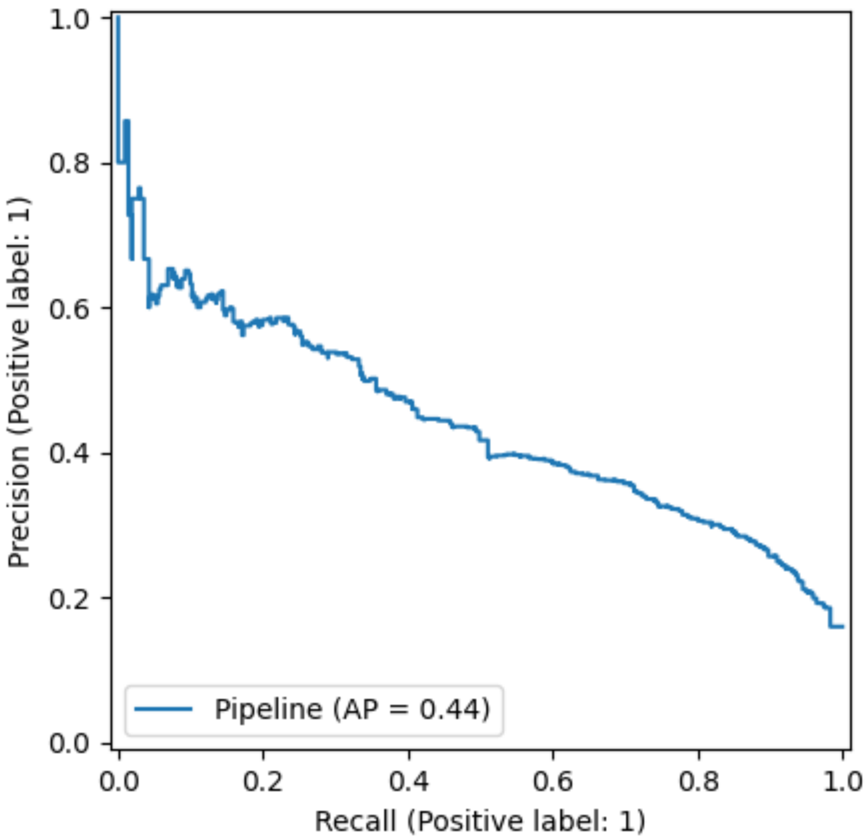
JM1_GNB_pr.png



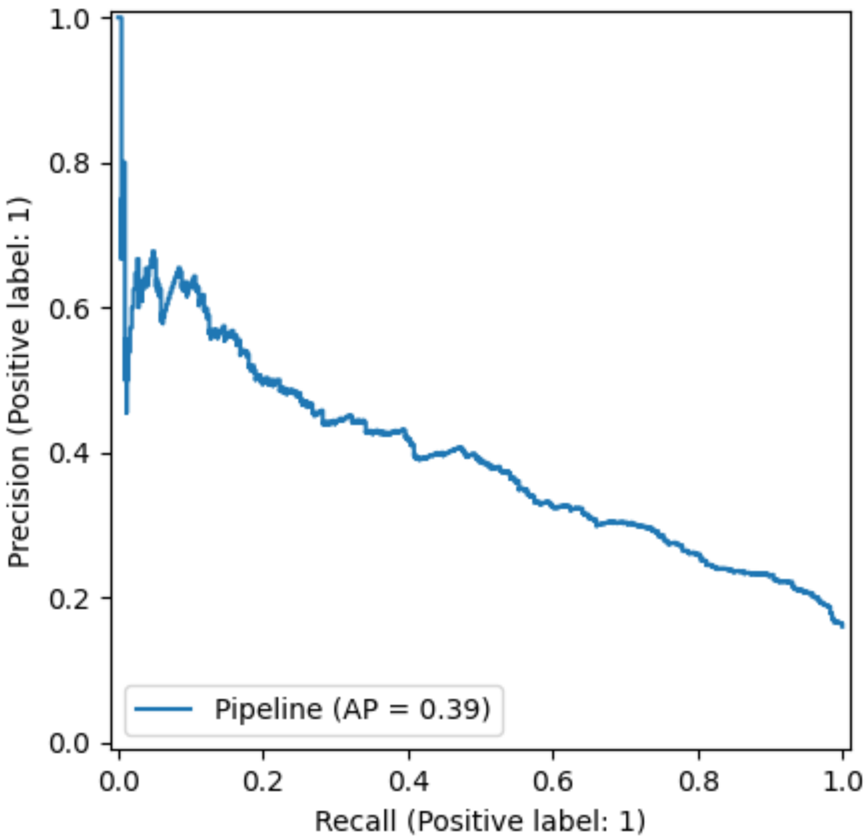
JM1_LogReg_pr.png


Pipeline (AP = 0.36)
Precision (Positive label: 1)
Recall (Positive label: 1)
0.0
0.2
0.4
0.6
0.8
1.0

JM1_RF_pr.png



JM1_RF_tuned_pr.png


1.0
0.8
0.6
0.4
0.2
0.0
Precision (Positive label: 1)
Pipeline (AP = 0.39)
0.0
0.2
0.4
0.6
0.8
1.0
Recall (Positive label: 1)