CROSS-TESTING

Purpose of Cross-Testing

Cross-testing is performed to evaluate a model's robustness and generalization when applied to data distributions different from those used during training. Unlike standard evaluation, cross-testing helps assess how well a model handles distribution shifts, which are common in real-world scenarios.

Cross-Testing of Saved Models

After identifying the best-performing Logistic Regression models from both datasets, cross-testing was conducted.

- **Model_A: Logistic Regression trained on the balanced dataset**
- **Model_B: Logistic Regression trained on the imbalanced dataset**

Each model was evaluated on the dataset it was not trained on, using its corresponding saved TF-IDF vectorizer to maintain feature consistency.

```
import joblib
model_A = joblib.load("Model_A.pkl")
tfidf_A = joblib.load("tfidf_A.pkl")
model_B = joblib.load("Model_B.pkl")
tfidf_B = joblib.load("tfidf_B.pkl")
```

Cross-Testing Setup

Model	Trained On	Tested On
Model_A	Balanced dataset	Imbalanced dataset
Model_B	Imbalanced dataset	Balanced dataset

This setup ensures a fair comparison without retraining and prevents feature mismatch.

Cross-Testing Results

Model_A → Imbalanced Data

- Accuracy: ~0.53
- Strong performance on extreme ratings (1 and 5)
- Moderate performance on middle ratings (2, 3, and 4)

<u>Observation</u>:
Model_A demonstrates good generalization to real-world imbalanced data, indicating effective class-neutral learning.

evaluate(model_A, X_imb_A, y_test_imbalanced_loaded,

"Model_A → Imbalanced")

```
===== Model_A → Imbalanced =====
Accuracy: 0.5307593462635024

Classification Report:
              precision    recall  f1-score   support

           1       0.50      0.80      0.62      4982
           2       0.44      0.45      0.44      7473
           3       0.52      0.45      0.49     12449
           4       0.59      0.45      0.51     14933
           5       0.56      0.68      0.62      9969

    accuracy                           0.53     49806
   macro avg       0.52      0.57      0.53     49806
weighted avg       0.54      0.53      0.52     49806


Confusion Matrix:
 [[3981  547  233   86  135]
 [1715 3335 1531  514  378]
 [1261 2464 5623 2196  905]
 [ 585 1008 2820 6672 3848]
 [ 403  279  527 1936 6824]]
```

<u>Model_B → Balanced Data</u>

- **Accuracy: ~0.58**
- **Improved performance across all rating classes**
- **More stable precision and recall compared to Model_A**

<u>Observation</u>:
Model_B benefits from learning natural class distributions and adapts effectively when evaluated on balanced data.

evaluate(model_B, X_bal_B, y_test_balanced_loaded,

"Model_B → Balanced")

```
===== Model_B → Balanced =====
Accuracy: 0.5771394134190438

Classification Report:
              precision    recall  f1-score   support

           1       0.66      0.77      0.71      4978
           2       0.53      0.45      0.49      4978
           3       0.49      0.47      0.48      4978
           4       0.52      0.49      0.51      4978
           5       0.65      0.70      0.67      4978

    accuracy                           0.58     24890
   macro avg       0.57      0.58      0.57     24890
weighted avg       0.57      0.58      0.57     24890


Confusion Matrix:
 [[3829  649  259  104  137]
 [1096 2236 1078  337  231]
 [ 495  907 2337  909  330]
 [ 175  267  865 2459 1212]
 [ 182  128  230  934 3504]]
```

Key Insights from Cross-Testing

- **Training on balanced data improves fairness but slightly reduces real-world accuracy.**
- **Training on imbalanced data improves overall accuracy but may introduce class bias.**
- **Logistic Regression shows stable performance across both testing scenarios.**
- **Data distribution has a significant impact on model behavior and outcomes.**

Conclusion

Cross-testing confirms that data distribution plays a critical role in model performance. Model_A emphasizes fairness across classes, while Model_B reflects realistic usage patterns. Evaluating both models provides a comprehensive understanding of robustness and reliability**.**