DATA ANALYTICS 1
ASSIGNMENT 4

Section 1: Classification

3a. When one or more classes have far fewer samples than others, the model becomes biased toward the majority classes, since minimizing overall error can be achieved by simply predicting the dominant classes.
This results in:

- Poor recall and precision for minority classes
- Skewed decision boundaries
- Misleading high overall accuracy despite poor performance on rare categories.

Strategies to Mitigate the Impact

1. Resampling Techniques: Oversampling minority classes (e.g., *SMOTE*, *ADASYN*). Undersampling majority classes (random or cluster-based). Balances class distribution to give equal importance during training.
2. Class Weighting / Cost-Sensitive Learning: Assign higher misclassification cost. Forces the model to pay more attention to underrepresented classes.
3. Data Augmentation: Generate synthetic or transformed samples (especially effective for images or text).
4. Evaluation Metrics: Use macro-averaged F1-score, balanced accuracy, or per-class recall instead of overall accuracy.
5. Ensemble and Cross-Validation: Use stratified cross-validation and ensembles (bagging/boosting) to stabilize learning from limited samples.

Class imbalance distorts learning toward majority classes. Balancing data (resampling), adjusting class weights, and using proper evaluation metrics are the most effective mitigation strategies, particularly when the dataset is small.

3b. Both Random Forest and SVM models rely on hyperparameters that control their complexity. Improper tuning can make them overfit (memorize training data) or underfit (fail to learn patterns).

Random Forest

- Too many deep trees (large max_depth, small min_samples_leaf): Trees capture noise and overfit training data.
- Too few trees or shallow trees (small n_estimators, low max_depth): Model becomes too simple which leads to underfitting.
- Low feature randomness (max_features too high): Trees become correlated leads to less generalization which leads to overfitting.
- High randomness (max_features too low): Trees become weak which leads to underfitting.

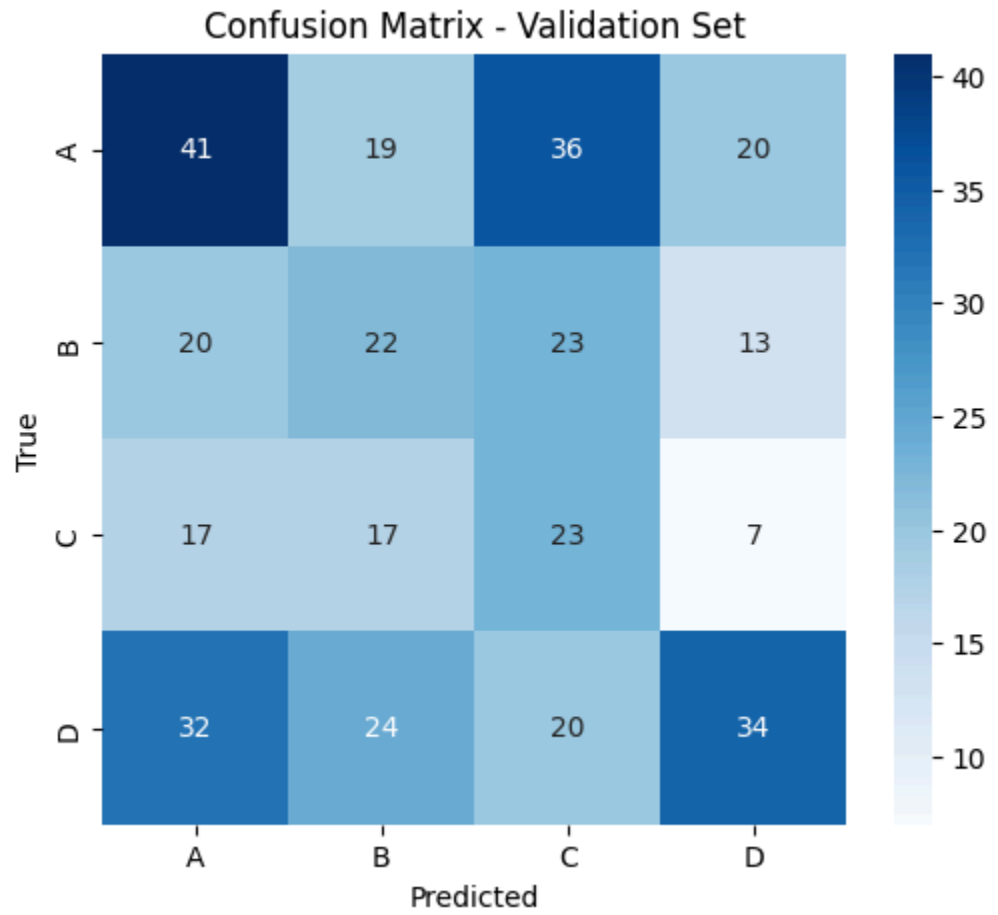
Support Vector Machine (SVM)

- Regularization parameter (C): Large C, tries to perfectly fit training data, overfitting. Small C, allows more margin violations, underfitting.
- Kernel and gamma (in RBF kernel): High gamma, decision boundary too complex, overfitting. Low gamma, boundary too smooth, underfitting.

High model complexity (deep trees, large C, high gamma) causes overfitting, while too much regularization (shallow trees, small C, low gamma) causes underfitting. Balancing these hyperparameters is essential for optimal generalization.

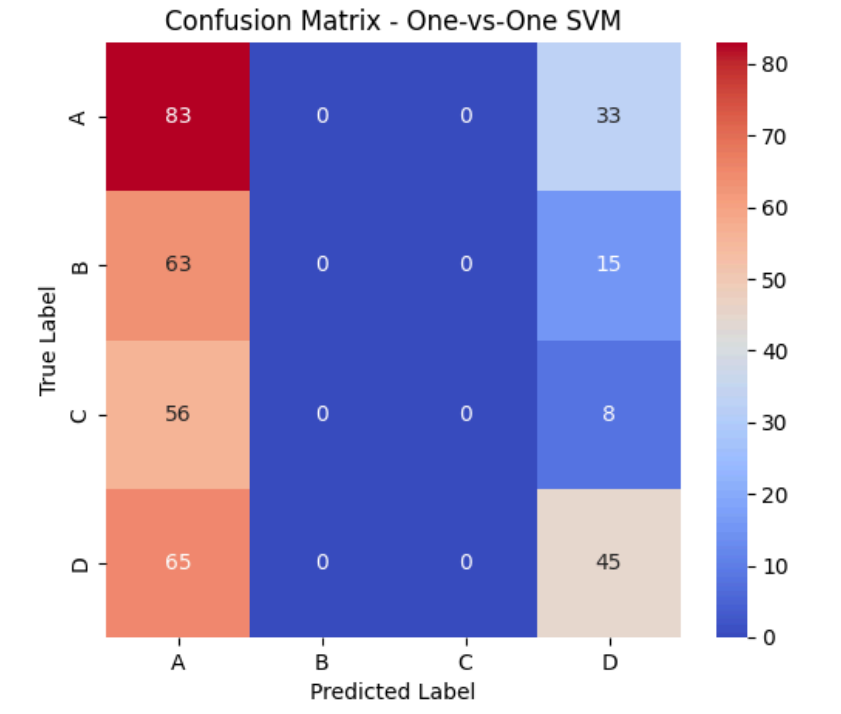
3c.
RANDOM FOREST RESULTS

```
Final Random Forest Validation Accuracy: 32.61%

Classification Report on Validation Set:
              precision    recall  f1-score   support

           A       0.37      0.35      0.36       116
           B       0.27      0.28      0.28        78
           C       0.23      0.36      0.28        64
           D       0.46      0.31      0.37       110

    accuracy                           0.33       368
   macro avg       0.33      0.33      0.32       368
weighted avg       0.35      0.33      0.33       368
```



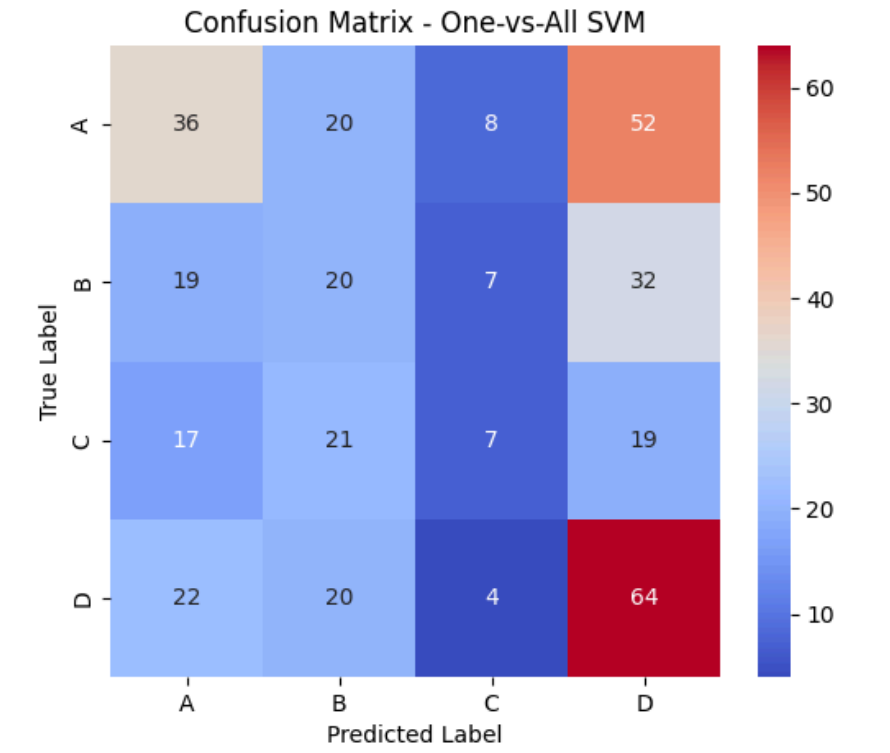
OVO CLASSIFIER RESULTS

```
Best Validation Accuracy: 35.87%  (C=10, γ=0.001)

============================================================
TRAINING FINAL MODEL
============================================================
Training 6 classifiers...

🔹 Test Accuracy: 34.78%
Precision (macro): 18.91%
Recall (macro): 28.12%
F1-score (macro): 21.50%

Detailed Classification Report:
              precision    recall  f1-score   support

           A       0.31      0.72      0.43       116
           B       0.00      0.00      0.00        78
           C       0.00      0.00      0.00        64
           D       0.45      0.41      0.43       110

    accuracy                           0.35       368
   macro avg       0.19      0.28      0.21       368
weighted avg       0.23      0.35      0.26       368


============================================================
SUMMARY
============================================================
OVO Accuracy: 0.3478
```



OVA CLASSIFIER RESULTS

```
Best Validation Accuracy: 35.33%  (C=100, γ=0.001)
Training 4 classifiers (One-vs-All)...

Evaluation Metrics:
Accuracy:  34.51%
Precision: 32.06%
Recall:    31.45%
F1-score:  30.30%

Detailed Classification Report:
              precision    recall  f1-score   support

           A       0.38      0.31      0.34       116
           B       0.25      0.26      0.25        78
           C       0.27      0.11      0.16        64
           D       0.38      0.58      0.46       110

    accuracy                           0.35       368
   macro avg       0.32      0.31      0.30       368
weighted avg       0.33      0.35      0.33       368
```



3d.

Metric	OVO	OVA
Validation Accuracy	35.87%	35.33%
Test Accuracy	34.78%	34.51%
Precision	18.91%	32.06%
Recall	28.12%	31.45%
F1 score	21.5%	30.3%

Analysis from Confusion-Matrix

1. One-vs-One (OVO): Strong bias toward predicting class A, most other classes misclassified as A. Classes B and C almost never correctly identified (0 recall). Indicates severe imbalance in decision boundaries; some binary models dominate.

2. One-vs-All (OVA): Predictions are more evenly distributed across all classes. Class D shows the best detection (recall ≈ 0.58). Still moderate confusion between classes A & D, but noticeably better than OVO. The heatmap shows all rows containing non-zero correct counts, less class collapse.

Interpretation

OVO trains K(K-1)/2 pairwise classifiers, each focuses on two classes only. On small, possibly imbalanced datasets, many binary pairs get too few samples leading to unstable decision boundaries.
OVA trains one classifier per class vs all others. Each model sees the full dataset, giving it more context and robustness despite class imbalance.

Conclusion

For this dataset, the One-vs-All (OVA) classifier performs better overall:

- Higher precision, recall, and F1-score across all classes.
- More balanced class predictions in the confusion matrix.

- Slightly simpler and more stable under limited data.

Hence, OVA > OVO in both interpretability and practical accuracy for this customer-segmentation dataset. The One-vs-All classifier achieved higher macro-averaged precision (32 %), recall (31 %) and F1-score (30 %) compared to One-vs-One (19 %, 28 %, 21 %). The OVA confusion matrix shows better class balance and fewer misclassifications, indicating that the OVA approach generalizes better for this dataset. Therefore, the One-vs-All SVM is the preferable model for the given customer segmentation problem.