**Support Vector Machines**

**Support Vector Machine (SVM) Training and Hyperparameters**

The Support Vector Machine (SVM) model was implemented to classify surgeon skill levels using nonlinear variability measures. The following hyperparameters were used for training:

* **Kernel**: Radial Basis Function (RBF)
* **C (Regularization Parameter)**: Default (1.0)
* **Gamma**: Auto-computed using scale (1 / (n\_features \* X.var()))
* **Probability**: Enabled for probabilistic output
* **Random State**: 42 (to ensure reproducibility)

**Feature Selection**

To enhance classification performance, **Recursive Feature Elimination (RFE)** was applied with the following settings:

* **Estimator**: Linear kernel SVM
* **Number of Features to Select**: 10
* **Step**: 1 (features eliminated iteratively one by one)
* **Classification Performance**
* The proposed SVM model achieved a classification accuracy of **0.59**, demonstrating moderate predictive performance. A detailed classification report revealed that the **Expert** class had a precision of **0.57**, recall of **0.69**, and F1-score of **0.62**, whereas the **Intermediate** class had a precision of **0.60**, recall of **0.57**, and F1-score of **0.59**. The **Novice** class exhibited a precision of **0.59**, recall of **0.50**, and F1-score of **0.54**. The macro-averaged F1-score was **0.58**, highlighting balanced performance across all classes.
* **Metrics Summary**
* The overall model metrics are summarized in Table X. Sensitivity and specificity values were **0.59** and **0.79**, respectively, while the Matthew's Correlation Coefficient (MCC) was **0.38**, indicating moderate correlation between predicted and true labels. The Area Under the Curve (AUC) value of **0.76** underscores the model's capability to distinguish between skill levels effectively.
* **Table X. Summary of Model Performance Metrics**

| **Metric** | **Score** |
| --- | --- |
| Accuracy | 0.59 |
| Sensitivity | 0.59 |
| Specificity | 0.79 |
| F1-Score | 0.58 |
| MCC | 0.38 |
| AUC | 0.76 |

#### Random Forest and XGBoost classifiers

#### Model Training and Hyperparameter Selection

The Random Forest and XGBoost classifiers were implemented to classify surgeon skill levels based on nonlinear variability measures of muscle activity during different tasks. The models were tuned using default parameters with the following specifications:

* **Random Forest**:
  + Number of estimators: **100**
  + Criterion: **gini**
  + Maximum depth: **None**
  + Minimum samples split: **2**
  + Minimum samples leaf: **1**
* **XGBoost**:
  + Booster: **gbtree**
  + Objective: **multi:softprob**
  + Evaluation metric: **logloss**
  + Learning rate (eta): **0.3**
  + Maximum depth: **6**
  + Number of estimators (trees): **100**
  + Subsample: **1.0**
  + Colsample by tree: **1.0**

Both models were trained using the SMOTE-resampled dataset to address class imbalance, ensuring fair performance evaluation across skill levels.

### Results

#### Hyperparameters and Model Performance

The selected hyperparameters for both Random Forest and XGBoost models ensured optimal performance on the classification task. Using the tuned parameters:

* **Random Forest** achieved an accuracy of **71.60%**, with a macro-averaged F1-score of **71.43%**.
* **XGBoost** outperformed Random Forest slightly, with an accuracy of **72.53%** and a macro-averaged F1-score of **72.40%**.

### Table Y. Hyperparameter Settings and Performance Metrics

| **Model** | **Hyperparameter** | **Value** | **Accuracy (%)** | **F1-Score (%)** |
| --- | --- | --- | --- | --- |
| Random Forest | Number of estimators | 100 | 71.60 | 71.43 |
|  | Criterion | gini |  |  |
|  | Max depth | None |  |  |
| XGBoost | Booster | gbtree | 72.53 | 72.40 |
|  | Evaluation metric | logloss |  |  |
|  | Learning rate (eta) | 0.3 |  |  |
|  | Maximum depth | 6 |  |  |

**Classification Performance**

The classification performance of Random Forest and XGBoost models was evaluated using accuracy, F1-score, and Matthews Correlation Coefficient (MCC). The metrics and detailed classification reports are presented below.

* **Random Forest**:
  + Accuracy: **71.60%**
  + F1-Score: **71.43%**
  + MCC: **57.57%**
  + The class-wise performance indicated that precision, recall, and F1-scores were highest for Class 0 (**Expert**), achieving an F1-score of **0.78**. Intermediate skill levels (Class 1) achieved an F1-score of **0.68**, while novice skill levels (Class 2) showed an F1-score of **0.69**.
* **XGBoost**:
  + Accuracy: **72.53%**
  + F1-Score: **72.40%**
  + MCC: **58.86%**
  + Class 0 (Expert) showed the highest F1-score (**0.78**) among the skill levels, while Class 1 (Intermediate) and Class 2 (Novice) had F1-scores of **0.69** and **0.71**, respectively.

**Table X. Performance Metrics for Random Forest and XGBoost Models**

| **Metric** | **Random Forest** | **XGBoost** |
| --- | --- | --- |
| Accuracy (%) | 71.60 | 72.53 |
| F1-Score (%) | 71.43 | 72.40 |
| MCC (%) | 57.57 | 58.86 |

**Detailed Classification Reports**

* **Random Forest**:

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
| --- | --- | --- | --- | --- |
| Expert (0) | 0.74 | 0.82 | 0.78 | 108 |
| Intermediate (1) | 0.67 | 0.69 | 0.68 | 108 |
| Novice (2) | 0.74 | 0.64 | 0.69 | 108 |

* **XGBoost**:

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
| --- | --- | --- | --- | --- |
| Expert (0) | 0.75 | 0.81 | 0.78 | 108 |
| Intermediate (1) | 0.71 | 0.67 | 0.69 | 108 |
| Novice (2) | 0.72 | 0.69 | 0.71 | 108 |

**Naïve Bayes**

Hyperparameter Optimization

Hyperparameter tuning was performed to ensure optimal performance of the Naive Bayes classifier. While Naive Bayes inherently has fewer hyperparameters compared to other algorithms, the following configurations were explored:

- \*\*Smoothing Parameter (alpha):\*\* The Laplace smoothing parameter was kept at its default value of 1 to handle potential zero probabilities in categorical data.

- \*\*Feature Scaling:\*\* Features were normalized using the StandardScaler to standardize the input feature space, ensuring compatibility with the algorithm's assumptions.

No additional parameter tuning was required as Gaussian Naive Bayes assumes a normal distribution of continuous features.

#### Naive Bayes Algorithm

The Gaussian Naive Bayes algorithm was utilized for classification. This algorithm operates on the principle of Bayes' theorem, assuming feature independence given the target class. Key steps include:

1. \*\*Data Preprocessing:\*\*

- Features were scaled to have zero mean and unit variance.

- Missing values were imputed using column-wise mean values.

- The dataset was split into training (70%) and testing (30%) subsets, stratified by class distribution.

2. \*\*Model Training:\*\*

- The Gaussian Naive Bayes classifier computes class-conditional probabilities using the Gaussian probability density function.

- The model calculates the posterior probability for each class given the input features, assigning the class with the highest probability to each instance.

3. \*\*Model Evaluation:\*\*

- Predictions were generated on the test set, and evaluation metrics were computed, including accuracy, sensitivity, specificity, F1-score, Matthews correlation coefficient (MCC), and area under the ROC curve (AUC).

- Class probabilities for each instance were extracted to generate ROC curves and assess the classifier's discriminative performance across skill levels.

4. \*\*Feature Importance and Task Analysis:\*\*

- Although Naive Bayes does not directly provide feature importance, the class-wise means (μ) and variances (σ²) were analyzed to identify features contributing to skill-level classification.

- Tasks and muscle groups were aggregated to determine their relevance in distinguishing skill levels, supporting interpretability and actionable insights for surgeon training.

This methodology ensured that the results were robust, interpretable, and aligned with the study's objectives of assessing skill-level classification using nonlinear variability measures.

Classification Performance

The Naive Bayes algorithm achieved a classification accuracy of \*\*54.7%\*\* on the test dataset, demonstrating its ability to differentiate between skill levels of surgeons. The detailed classification metrics are presented below:

- \*\*Expert:\*\*

- Precision: 0.58

- Recall (Sensitivity): 0.26

- F1-Score: 0.35

- Support: 43

- \*\*Intermediate:\*\*

- Precision: 0.53

- Recall (Sensitivity): 0.67

- F1-Score: 0.59

- Support: 108

- \*\*Novice:\*\*

- Precision: 0.56

- Recall (Sensitivity): 0.54

- F1-Score: 0.55

- Support: 94

- \*\*Overall Accuracy:\*\* 0.55

#### Aggregated Metrics

A summary of the key evaluation metrics is as follows:

| Metric | Score |

|----------------|-----------|

| Accuracy | 0.546939 |

| Sensitivity | 0.488345 |

| Specificity | 0.745214 |

| F1-Score | 0.499594 |

| MCC | 0.251210 |

| AUC | 0.650108 |

These results indicate that the classifier shows moderate performance, with higher specificity (74.5%) compared to sensitivity (48.8%). The area under the ROC curve (AUC) value of 0.65 suggests some ability to distinguish between classes but highlights room for improvement in model optimization. Further exploration of features, tasks, and muscle groups may enhance classification performance.