Summary of Approach to the Motion signal classification competition

- **Selected Models:**
  Five different models were chosen for this problem: XGBoost, Random Forest, Support Vector Machines (SVM), Multi-Layer Perceptron (MLP) and Logistic Regression.

- **Machine learning approach used:**
  The approach used is supervised learning, where labelled data were utilized to train the selected models. The hyperparameters for the models were also tuned using "*GridSearchCV*", which explored different combinations of hyperparameters to find the best-performing model configurations.

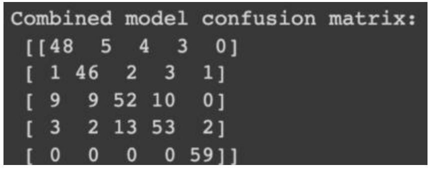
- **Data Splitting:**
  The data was split into training and validation sets in an 80-20 ratio using "*train_test_split*" function with a random state of 42.

- **Feature Engineering:**
  The given extracted features were preprocessed using a KNN imputer with five neighbours to fill in the missing values. Next, the data was standardized using a "*StandardScaler*". Then, feature selection was performed using the "*SelectKBest*" method with 2000 top features. Principal Component Analysis (PCA) was used to further reduce the dimensionality of the data to 100 components.

- **Model Combination:**
  The models were combined using a soft voting ensemble approach, *"VotingClassifier,"* with equal weights assigned to all models.

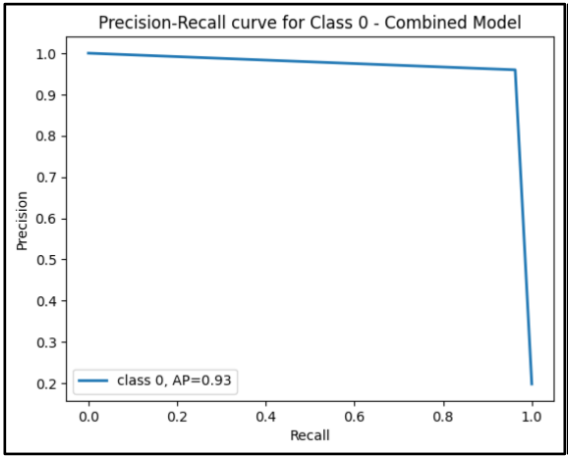
- **Model Assessment:**
  The performance of each individual model was first assessed using accuracy scores and confusion matrices. The Precision-Recall curves were also plotted for each class and each model. The model with the least accuracy (Logistic Regression) was dropped. The combined model's performance was then thoroughly assessed using a 5-fold cross-validation approach, which entailed partitioning the training dataset into five equal-sized subsets. During the cross-validation process, the model was iteratively trained on four subsets and validated on the remaining subset. The mean and standard deviation of the accuracy scores were obtained from these iterations.

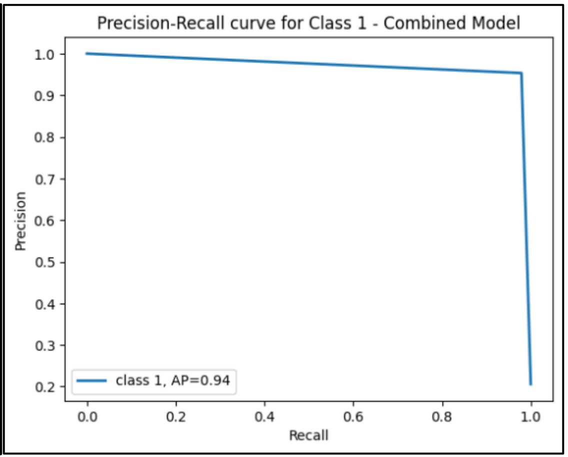
- **Confusion Matrix of my model using the validation/test set:**

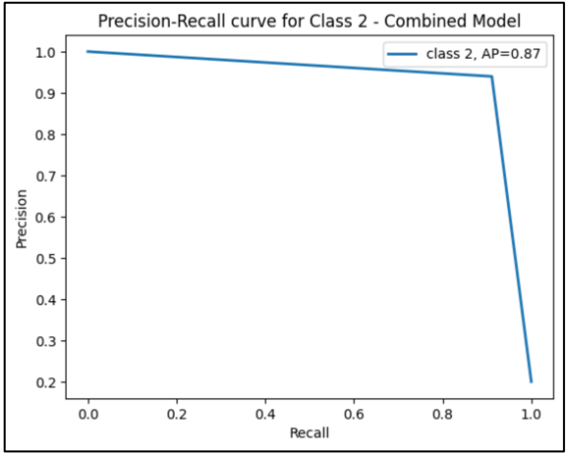


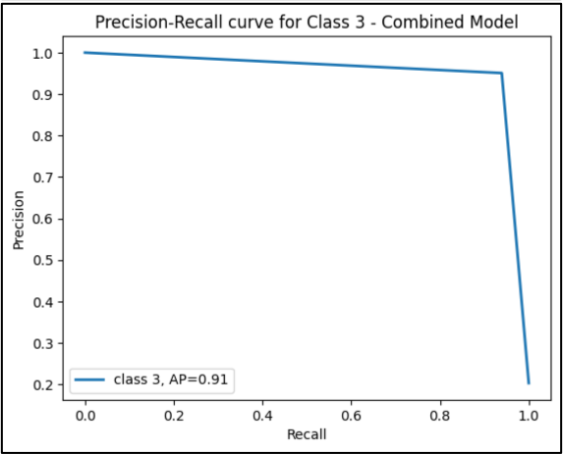
```
Combined model confusion matrix:
 [[48  5  4  3  0]
 [ 1 46  2  3  1]
 [ 9  9 52 10  0]
 [ 3  2 13 53  2]
 [ 0  0  0  0 59]]
```

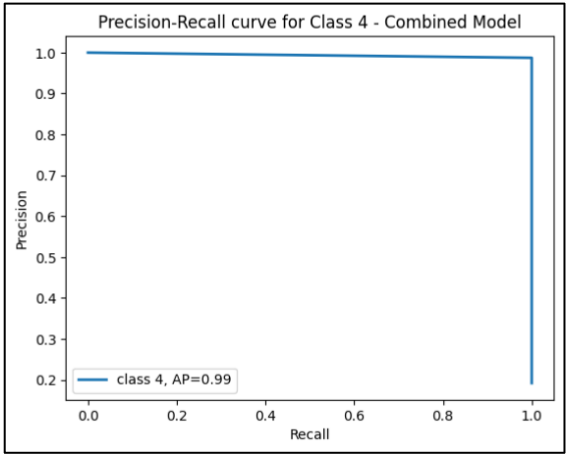
- **PR-Curve of my model for each class:**


Precision-Recall curve for Class 0 - Combined Model
Precision
Recall
class 0, AP=0.93


Precision-Recall curve for Class 1 - Combined Model
Precision
Recall
class 1, AP=0.94


Precision-Recall curve for Class 2 - Combined Model
Precision
Recall
class 2, AP=0.87


Precision-Recall curve for Class 3 - Combined Model
Precision
Recall
class 3, AP=0.91


Precision-Recall curve for Class 4 - Combined Model
Precision
Recall
class 4, AP=0.99