Project 2

Prameth Gaddale (pqg5273@psu.edu)

February 20, 2023

Abstract

This project primarily deals with feature subset selection for the classification problem through the use of the Taiji sequence dataset. The use of feature selection is to solve the issues of 'Curse of Dimensionality', computational efficiency, easier data collection, storage size, and interpretability through the strategy of dimensionality reduction. Implementations of the feature selection algorithms in this project include **Filtering** and **Wrapper** methods.

Contents

1 Introduction

The last project followed an approached where vanilla classification problem was implemented through the use of *Fisher Linear Discriminant Analysis*, which would not necessarily correspond to complexity involved by higher number of feature dimensions [1].

The approach previously taken in the *Project-1* didn't involve any data normalization or feature engineering steps which are essential pre-processing steps for the data to be involved with before being used for making crucial classification decisions/predictions.

Feature Selection which is the part of the family of *Feature Engineering* methods which involve the steps followed before fitting the training the final regression or classification model to improve the performance through the use of relevant features. Modern machine learning datasets contain thousands of features corresponding to the dataset used.

However, each of them correspond to either useful/useless category of features used for training the parameters of the machine learning model. For example, in the given dataset there are features that don't have any variance, or have a constant value for all the training examples. It would be wise to exclude the feature in that case to gain more leverage in overall computational efficiency, storage and algorithmic efficiency.

2 Approach

Feature Selection in a broad way consists of choosing a proper subset of features of all the given features based on some defined criterion. Feature Selection operations performed in this project involve the use of:

- Filter Method
- Wrapper Method

Both of the methods differ based upon the optimization criterion set by the objective function.

Filter Methods select the best features based on their discrimination potential through the use of a chosen metric. There is no actually iterative optimization taking place in this algorithms, however, the features are selected based of metric ranking.

On the other hand, *Wrapper Methods* select the best feature subset through a criteria set by solving a classification task on the subset iteratively through a search algorithm.

2.1 Data

The given dataset is PSU-TMM100 (Taiji) based on human-sequence forms, rendered using 3D motion capture devices from various crucial body parts and foot pressure sensors.

The 3D body joints from MoCAP captures 17 joints and the foot pressure data consists 1910 elements as shown in 1. The total number of features come out to be 1961 for each observation.

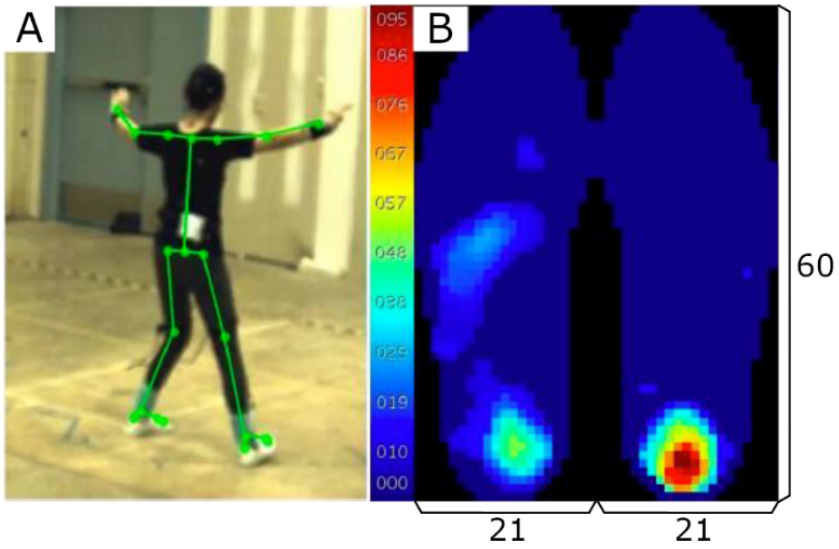

Figure 1: PSU-TMM100 Dataset (A). The Video Data with 17 joints. (B). The left-right foot pressure sensor heatmap.

Data Splitting Strategy

The training process involves the *Leave One Subject Out* strategy during classification to enable testing with pseudo-unseen data and resembles the prediction based off in a real scene setting.

2.2 Filter Method

Feature selection filtering is a technique used to select a relevant subset of features from a larger given set of features. In a way, reducing the use of irrelevant and redundant features in the given data. The reduced redundancy has help find gains in the efficiency of the overall classification process. The goal of feature selection is to improve the model performance and reduce the risk of over-fitting [2].

Filter methods usually evaluate each of the given feature independently and assign a corresponding score to that feature based on its relevance to the target vector. Features that score below a certain threshold are then removed from consideration, or can be sorted out by taking a set of features from thr top. The commonly used filter methods include:

- Variance Ratio
- Augmented Variance Ratio
- Minimum Redundancy Maximum Relevance

The metric implemented in this *Report* is the *Variance Ratio* which is defined as in (1)

$$VR(F) = \frac{Var(S_F)}{\frac{1}{C}\sum_{k=1,\ldots,C} Var_k(S_k)} \tag{1}$$

- $VR(F)$: Variance Ratio Score for the Feature F
- $Var(S_F)$: Within class variance
- $Var_k(S_F)$: Within class variance of the class k
- C: Total number of classes.

Filter approaches are generally easy to implement and computationally efficient, making them a popular choice for feature selection in large data sets. We represent the variance ratio for a particular feature to be the ratio of the inter class variance contained in that feature to the ratio of the mean of the intra class variances for all the classes. Therefore, a larger value of the variance ratio would indicate a more desirable feature with the highest variance ratio feature being the most desirable one.

Filter Method Algorithm

The algorithm of the Filter Method is given by,

- Consider all the given features, F in the Train set.
- Ensuring the filter feature count, $filter\_feature\_count$ to be less than total dimensions, start iterating over all the features.
- Compute the variance, $Var(F)$ of each feature, and compute the ratio with the help of per-class variance $Var_k(S_F)$.
- Receive the indices by sorting the variance scores computed from the previous step.
- Hence, we can get the top-100 features from the sorted list indices which represent the filtering method features.

2.3 Wrapper Method

Wrapper methods typically employ a search algorithm to select a subset of features based on their impact on model performance [3]. Unlike filter methods, which evaluate features independently, wrapper methods considers the interdependence between the features and the model.

Generally the wrapper methodology involves defining the target vector and a set of potential features (predictors). Upon that, we'd define a search algorithm procedure that iteratively adds or removes features to/from the model. Subsequently, a machine learning model is trained using a subset of features and the performance of the model is evaluated either through the use of training data or validation data. Consequently, we use the performance metric to guide the search algorithm in selecting the best subset of features. Upon finding a good enough set of features from the search procedure, we train a classification model using the selected subset of features.

Optimization Criterion

For this case, we consider the local classification rate we achieve while iterating towards our selected features using sequential forward selection. In our case, we have put a cap of 0.75 classification rate which must be achieved by that particular subset of features to be selected in the final selected index list. As the algorithm is greedy, it tends to add another feature in order to increase the number of final features in the set. However in parallel it won't let go of the temporary feature completely as it had already picked on its first run.

Classifier Used

In this case, the classifier used for training the wrapper method through the Sequential Forward Selection method is Linear Discriminant Analysis, as it perform better than KNN for reducing the over-fitting. Eventually, performing better in the test-set generalization process.

Wrapper Method Algorithm

The algorithm of the Wrapper method through the use of Sequential Forward Selection is given by,

- Initially, consider an empty set of the features S, and store the complete set of features F.
- While the size of S is smaller than a pre-defined constant value, for each feature from F, add it to S and train a classifer.
- If the performance on the train set or the validation set is greater than a pre-defined threshold, add that feature to the final set, and loop over all the subset in a ordered fashion to find relevant features.
- Return the best set of the features received.

3 Report

3.1 Model Pipeline

- The given data was first normalized with respect to the minimum and maximum values in the design matrix. Hence, the range is normalized to be [0,1].
- After the data normalization part, the number of features from the filter method is specified and the filtering method is run with the criterion of Variance Ratio.
- In our case, we pass all the 1961 features and receive the 100 selected features sorted based on the variance score and their associated corresponding variance scores.
- Upon receiving the filtered features, we implement the wrapper method with sequential forward selection algorithm for searching the relevant features based on the classification accuracy on the linear discriminant classifier.

- Then the features received from the wrapper method are used to train a K-Nearest Neighbor classifier with $k = 10$.

The schematic of the model pipeline is shown in .

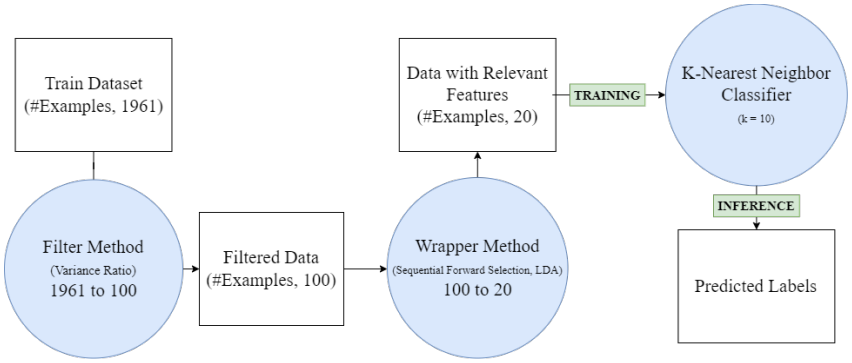

Figure 2: Model Pipeline

3.2 Classification Results

3.2.1 Baseline Model (No Filter, No Wrapper)

As a baseline measure for comparison purposes, a model without feature selection of filtering and wrapping methodology was implemented. The data used for the classification involved the use of all the 1961 features for train and test phases. The classification model used for training and inference was *K-Nearest Neighbors Classifier* from *Scikit-Learn* with $k = 10$.

Classification Rates

The classification model is not generalizing well over the test set as being trained on the wide variety of features. Its evident from the figures of the per class training and test set accuracies.

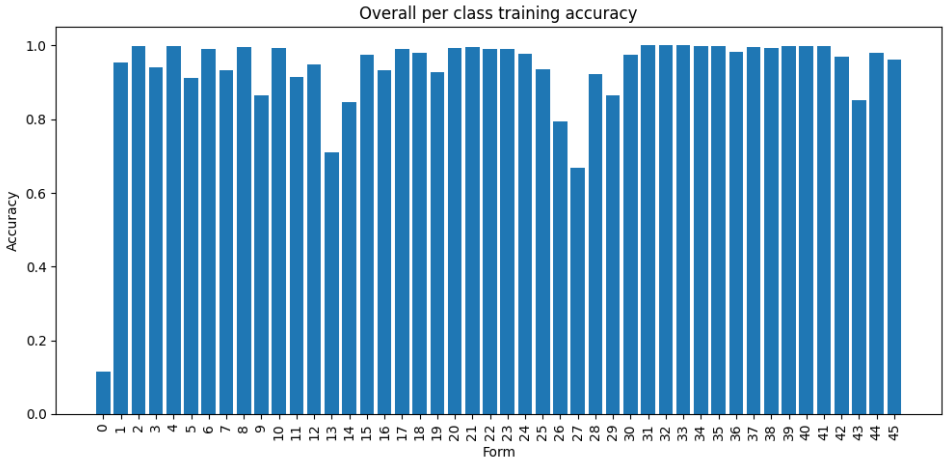

Figure 3: Training Set Confusion Matrix: No Filter, No Wrapper

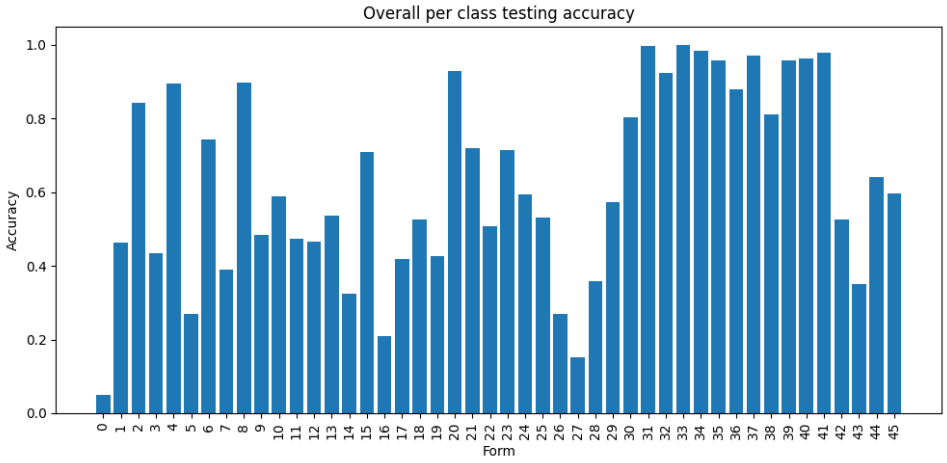

Figure 4: Test Set Confusion Matrix: No Filter, No Wrapper

Category	Train	Test
Subject 1	0.927	0.617
Subject 2	0.929	0.585
Subject 3	0.930	0.624
Subject 4	0.928	0.610
Subject 5	0.928	0.640
Subject 6	0.926	0.657
Subject 7	0.927	0.669
Subject 8	0.929	0.665
Subject 9	0.929	0.683
Subject 10	0.929	0.543
Overall	0.928	0.623

Table 1: Subject-wise accuracy rates for no filter with no wrapper configuration.

The table and the subject-wise accuracies represent the classic case of over-fitting, as the test set accuracy is far below the training set accuracy.

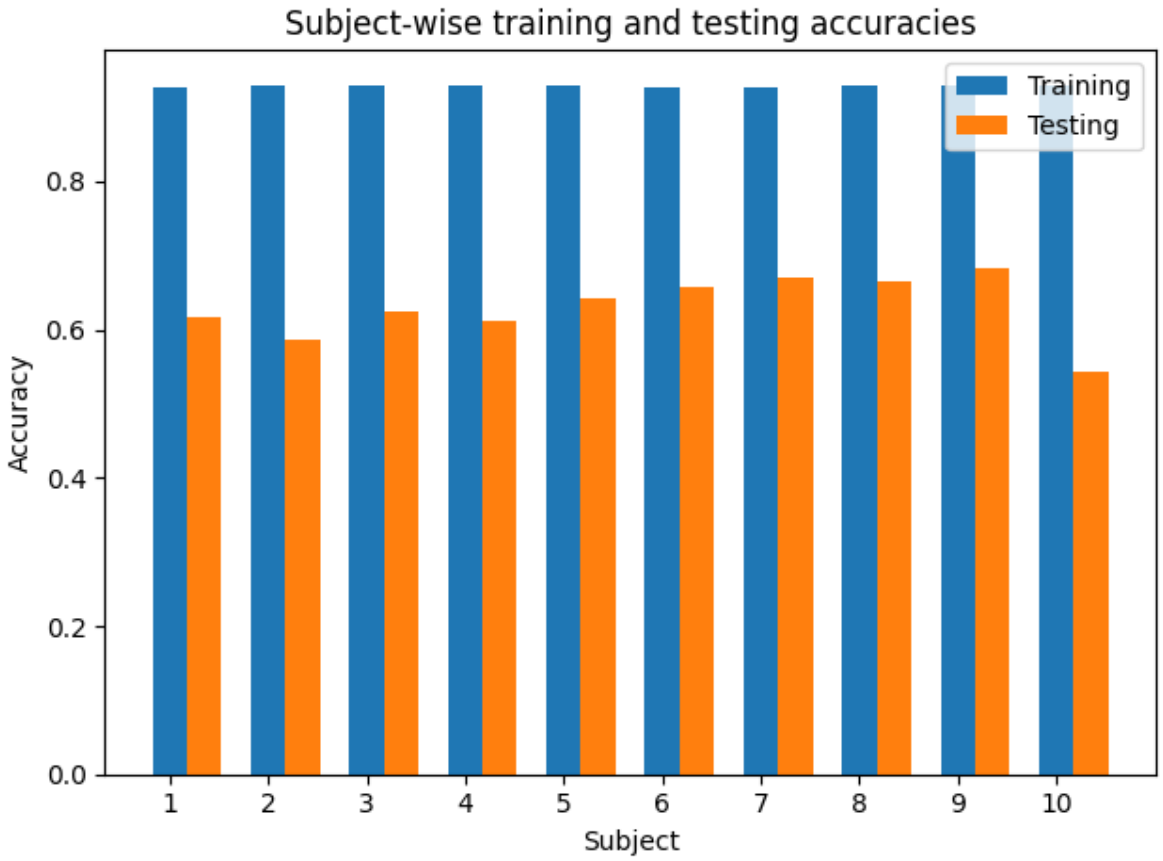

Figure 5: Subject-wise training and testing accuracies.

Confusion Matrix

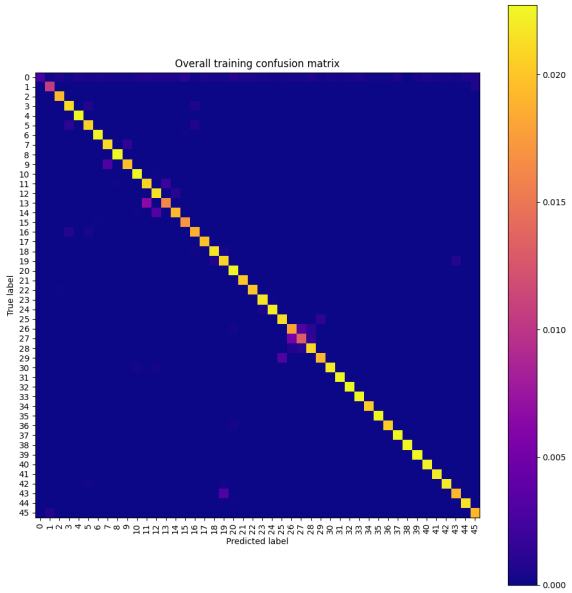

Figure 6: Training Set Confusion Matrix: No Filter, No Wrapper

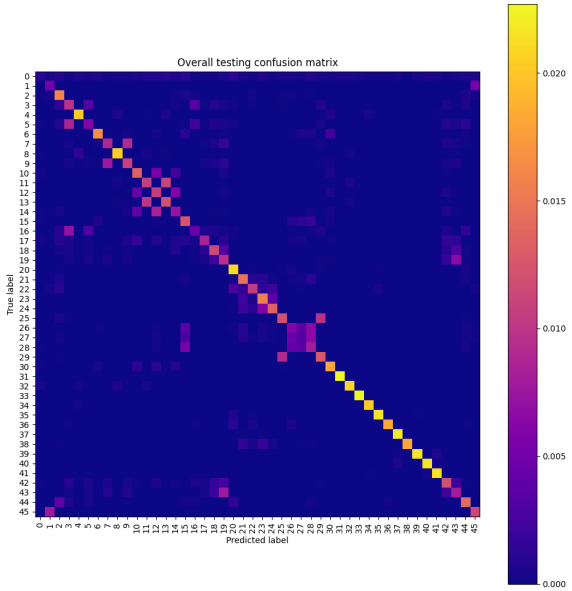

Figure 7: Test Set Confusion Matrix: No Filter, No Wrapper

In this case, its observed that there is a high degree of mis-classification rate in the features close to each other, as indicated by the darker shades in near-diagonal elements.

That would give us the reasoning behind the poses which might not be all that different from each other. They may have a number of joints positioned in similar ways without substantial change in the feet movement being likely explanations to be the cause of this over-fitting behavior.

This behavior justifies the performance of the feature selection methods because we now recognize the need to select those joints or the associated feet pixels which can potentially vary significantly, or be discriminative enough for even these same poses. Hopefully, adding the filter would help curb the over-fitting nature of the KNN Classifier employed.

3.2.2 Filter, No Wrapper

The filtering method with Variance Ratio was implemented and the top 100 features were considered. The processed training data was used to fit the *K-Nearest Neighbors Classifier* from *Scikit-Learn* with $k = 10$.

Classification Rates

Let us look at the classification rates for the features both for training and test data followed by the the training and testing classification rates per subject as shown in the following figures.

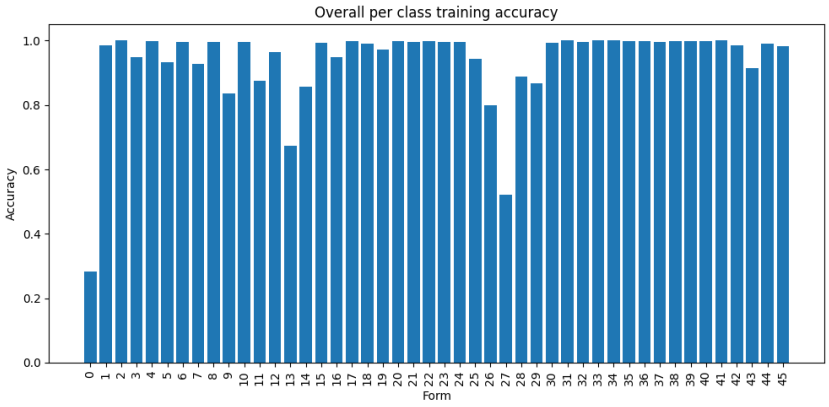

Figure 8: Per Class Training Accuracy per Form

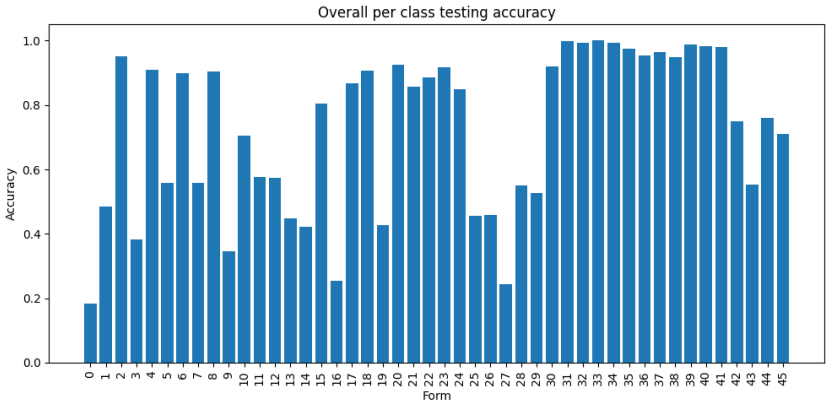

Figure 9: Per Class Testing Accuracy per Form

Clearly, there is a minor improvement observed in the test classification rates. This comes at the expense of training classification rates which would always increase as long as we gave it more features, associating this problem with a tradeoff with features and training accuracy. However, the over-fitting problem has ameliorated at a certain extent with nearly all features reaching more than 40% test accuracy with some noticeable outliers.

Category	Train	Test
Subject 1	0.924	0.664
Subject 2	0.934	0.666
Subject 3	0.935	0.722
Subject 4	0.935	0.776
Subject 5	0.937	0.774
Subject 6	0.935	0.767
Subject 7	0.937	0.782
Subject 8	0.935	0.779
Subject 9	0.929	0.642
Subject 10	0.938	0.685
Overall	0.934	0.726

Table 2: Subject-wise accuracy rates for filter with no wrapper.

Its evident from 2 that the model is overfitting the training data distribution as the training accuracy rate is substantially higher compared to the testing accuracy rate.

Figure 10: Subject-wise training and testing accuracies.

Looking at the subject-wise training and test accuracies for each individual subjects, we find that nearly all subjects reach more than 60% test accuracy. Hence, which is much better than the no filter method, however there is constant steady behavior expected with the training accuracy.

Confusion Matrix

Looking at the confusion matrices, its evident that the over-fitting issue hasn't been solved yet. The training set confusion matrix diagonal elements are bright compared to the rest which is the desirable feature.
However, the testing set confusion matrix hasn't been set yet. Hopefully introducing a wrapper method suitable to the data would be helpful in fixing the issue.

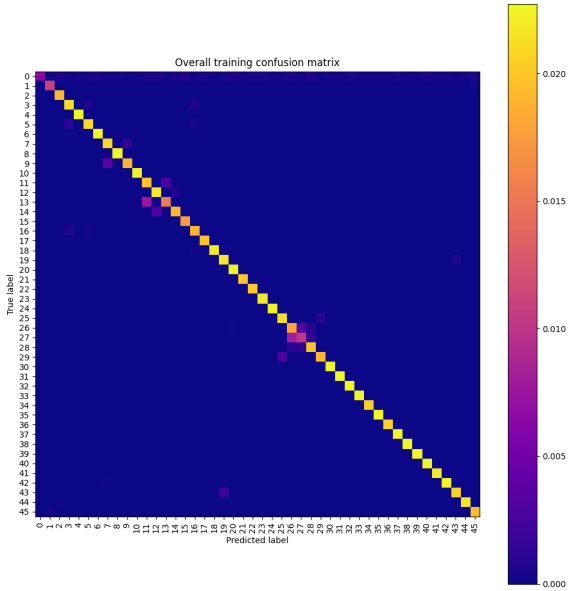

Figure 11: Training Set Confusion Matrix: Filter, No Wrapper

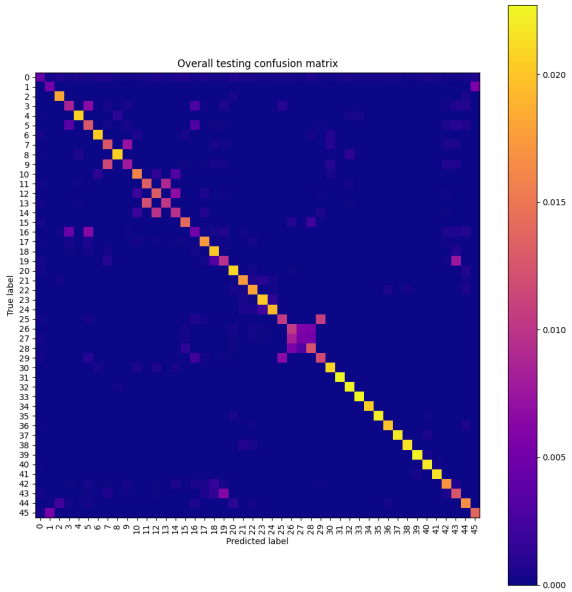

Figure 12: Test Set Confusion Matrix: Filter, No Wrapper

3.2.3 Filter, Wrapper

The filtering method with Variance Ratio as the measure was implemented. The top 100 features passed through the wrapper method with Sequential Forward Selection. The processed training data was used to fit the *K-Nearest Neighbors Classifier* from *Scikit-Learn* with $k = 10$.

Classification Rates

We observe a negligible change in the performance when compared between the filter and filter wrapper method. Its also evident after looking at the subject-wise training and test set accuracies.

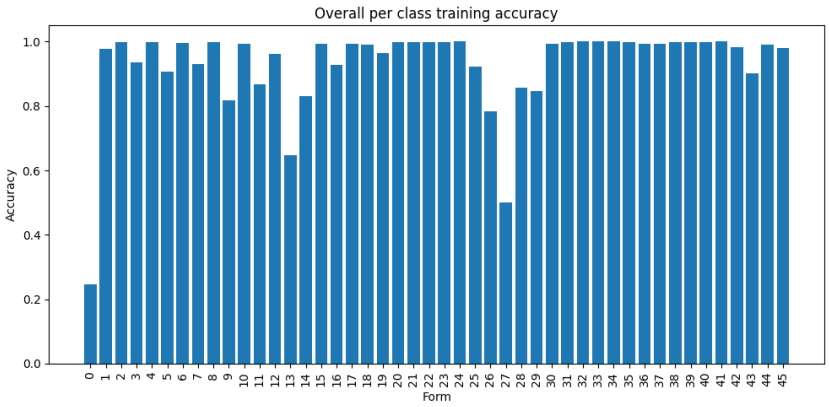

Figure 13: Per Class Training Accuracy per Form

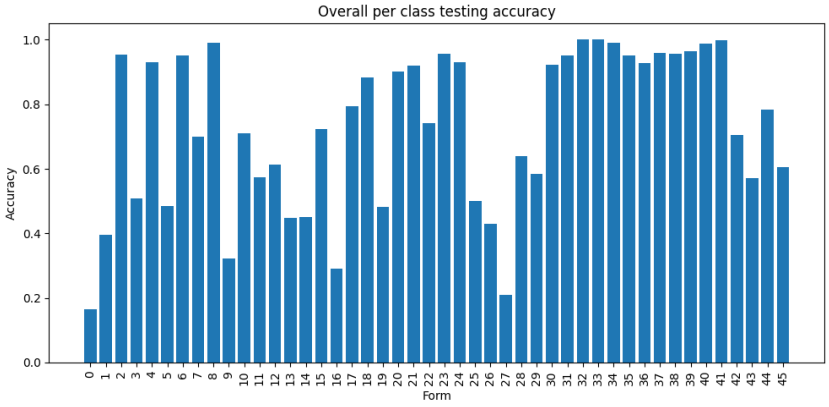

Figure 14: Per Class Testing Accuracy per Form

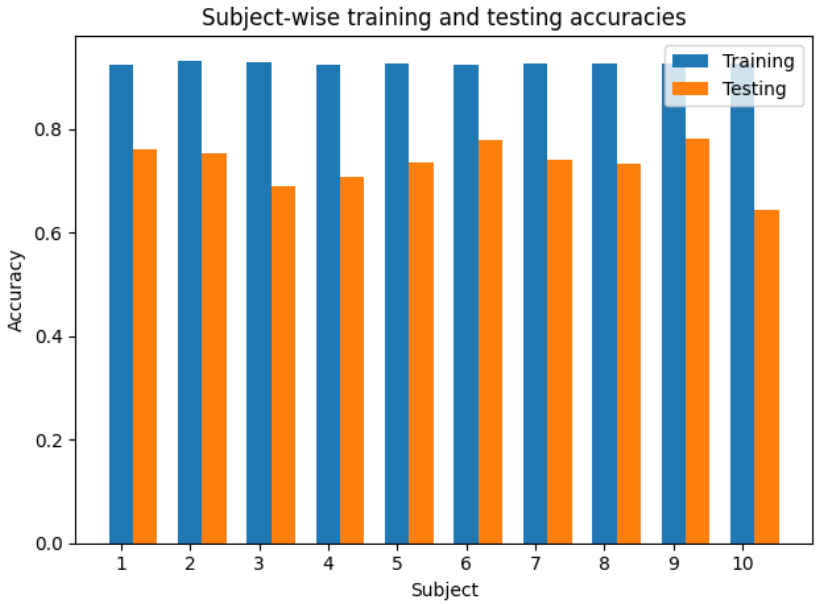

Figure 15: Subject-wise training and testing accuracies.

Confusion Matrix

The confusion matrices appear pretty much the same too as in the filter without wrapper method, like all other graphs. However, the contrast in the traning set has improved indicating that the training set has been overfitted due to the use of KNN classifier.

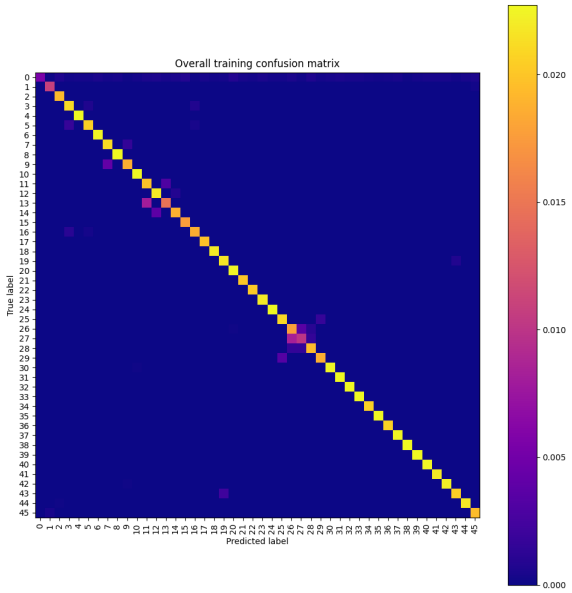

Figure 16: Training Set Confusion Matrix: Filter and Wrapper

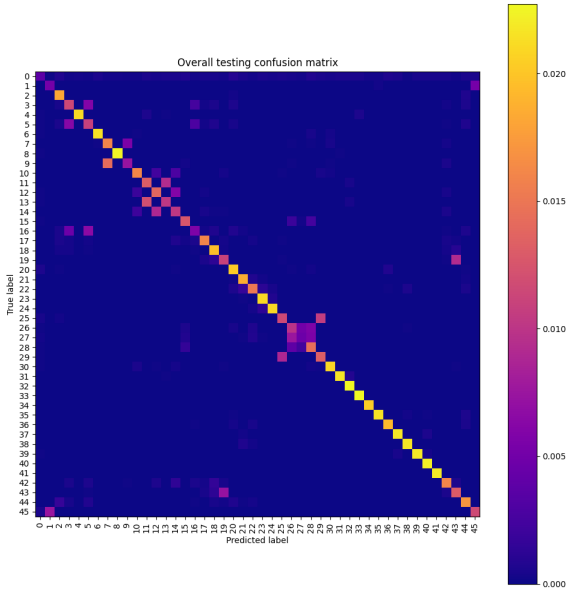

Figure 17: Test Set Confusion Matrix: Filter and Wrapper

Most Discriminative Features

Here is the plot representing the most discriminaive features recieved by the feature-wrapper method of solving the classification problem. Both the MOCAP Joint 3 and MOCAP Joint 8 have been selected 8 times, indicating that they are the features with most variance embedded in them, exploited by the filter method implementation.

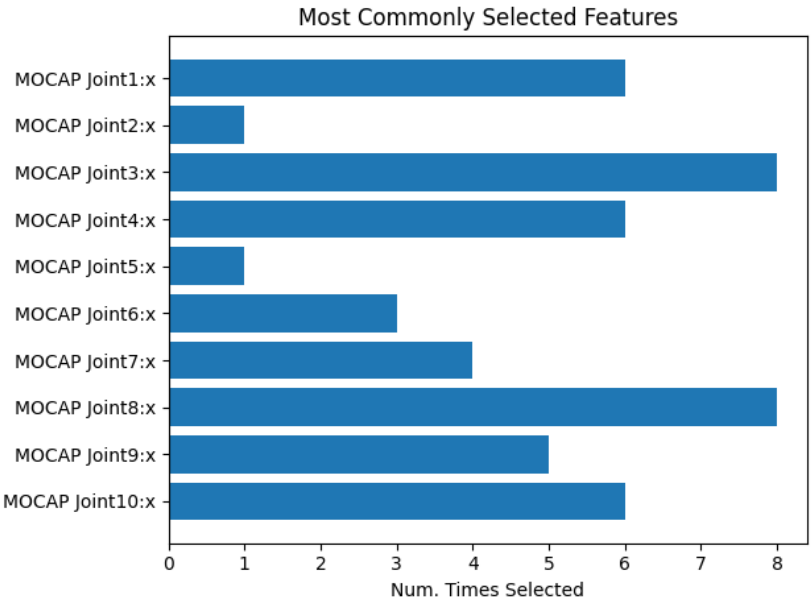

Figure 18: Most Discriminative Features: Filter

Most Commonly Selected Features

Here is the plot representing the most commonly selected features recieved by the feature-wrapper method of solving the classification problem. In this case, MOCAP Joint 7 and MOCAP Joint 10 have received greatest scores compared to the rest. Comparing the figures 17 and 18 we get that MOCAP Joint 7 has most varaince and gives great classification rates in the temporary feature vectors while training the wrapper method.

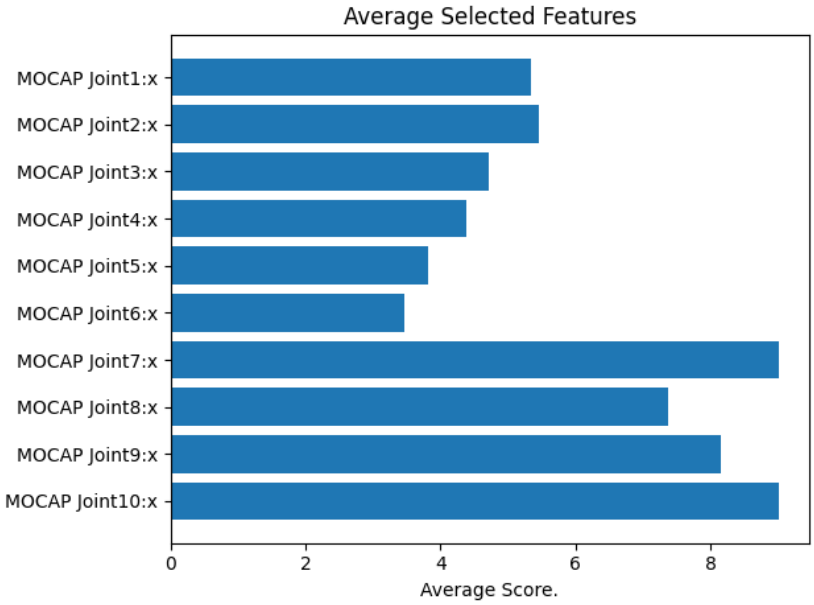

Figure 19: Most Commonly Selected Features: Wrapper

3.3 Question: Dataset Size Sufficiency

The sufficiency of a dataset size for classification problems in general depends on several factors such as the complexity of the problem, the total number of features, the amount of noise in the given dataset. Its also dependent on the desired level of accuracy being targeted. In general, larger the dataset is more reliable feature selection results are expected as they contain more information that can help distinguish between relevant and irrelevant features.

However, it is important to note that simply increasing the size of the dataset does not necessarily guarantee better feature selection results. In parallel, its important to consider the quality of the data and the distribution of the samples in the dataset. For example, in the case of the dataset having a high degree of class-imbalance or unknown/missing values, then the feature selection results may not be reliable.

In addition, the number of features available in the given dataset can also affect the sufficiency of the dataset size for feature selection. If the dataset contains a large number of features compared to the number of samples, then the feature selection process may be less reliable, as the algorithm may struggle to identify relevant features due to constraints set by the **curse of dimensionality** [1].

In an application point of view, incorrectly classifying patients based on clinical data can lead to misdiagnosis and mistreatment. For example, mis-classifying a patient as

having a certain condition when they do not can lead to unnecessary treatment and potentially harmful interventions [4].

In summary, while a larger dataset size can be beneficial for classification in general, it is important to also consider other factors such as the overall quality of the data and the number of features, in order to obtain reliable and accurate prediction results.

4 Conclusion

- To improve the performance of the classification on the Taiji dataset, filter method and wrapper method was implemented.
- Top 100 features were considered from the filter based on the quantitative relevance captured by the variance ratio estimates.
- In the wrapper method, it was observed that the total number of features selected for each of the test subjects was in the range of 15-20. Which is about less than 25% of the features received by the Filter method.
- The sequential forward selection algorithms characterizes most of the features to be redundant to receive greater performance on the training set which is not necessarily perfect.
- However, it's important to note that this comes by using a lot less features which can be crucial as in some cases we are just using 20 features to achieve performance on par with 100 features.
- Performance improvements of the wrapper method are not that crucial enough to greatly improve the working on the Taiji dataset.
- Using ensemble of machine learning classification algorithms for training seems to give better results than plain filter-wrapper methodolgy with the KNN classifer in the end for predictions.

References

[1] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer Sciences Media, 2006. 2, 16

[2] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001. 3

[3] U. Stanczyk and L. C. Jain, *Feature Selection for Data and Pattern Recognition*. Springer Publishing Company, Incorporated, 2014. 4

[4] K. S. Ladha and M. Eikermann, "Codifying healthcare – big data and the issue of misclassification," *BMC Anesthesiology*, vol. 15, no. 1, Dec. 2015. [Online]. Available: https://doi.org/10.1186/s12871-015-0165-y 17