Data Mining Project: Clustering-Based Action Clips Classification

Team RR
Rudranil Naskar
2022MT11287
Raman Jakhar
2022MT11941

April 12, 2025

1 Non-competitive Part

1.1 Introduction

The objective of this project is to cluster the video dataset into 15 clusters, and then label new clips based on the cluster labeling done in training.

1.2 Data pre-processing

There were 2 kinds of videos in the dataset, some with 25 fps and others with 29 fps. Hence it was preferable to choose a frame rate less than or equal to 25 fps for frame extraction. **We chose 25 fps for feature extraction** as the I3D model used was pretrained using 25 fps videos. Only exception was Motion Boundary Histogram feature extraction, where we used only 1 fps, since it was computationally expensive.
All frames were **resized to 224x224** and **normalized to mean [0.485, 0.456, 0.406], std [0.229, 0.224, 0.225]** before for **ResNet50** feature extraction.

1.3 Feature Extraction and Normalization

We used the following methods:-

Optical Flow Magnitude

We used **Farneback method** to compute optical flow between frames and return the mean and variance of motion magnitudes for each video. We chose not to do normalization in this method as the magnitude carries useful information like speed of action.

HOG

We computes **HOG** features on sampled keyframes (every 10th frame) and then used their average. We did not do normalization as HOG already does built normalization.

ResNet50

We used **ResNet50** (torchvision) to extract **2048D features** from the last pooling layer per frame and return their average. For normalization, we **first standardized** the features for better convergence, **then applied L-2** normalization for more meaningful similarity measure.

I3D

We used the **Inception-v1 Inflated 3D ConvNet used for Kinetics CVPR paper**. The model was introduced in: *Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset Joao Carreira, Andrew Zisserman* `https://arxiv.org/pdf/1705.07750v1.pdf`. We extracted features from the logits layer of the Inception I3D Model. Unlike other feature extractors, this one takes in the video without preprocessing as it performs its own preprocessing in accordance with what is required by the I3D model. No normalization was done as I3D already returns soft-maxed features

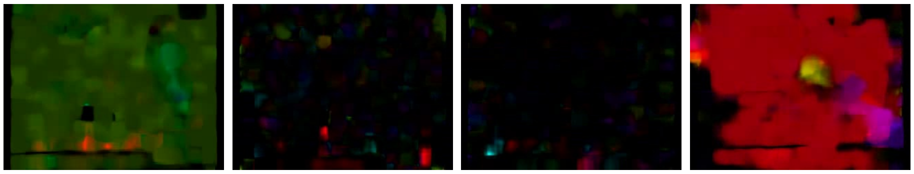
Motion Boundary Histogram

We used **motion boundary histograms** to extract features for each video. We applied **rootSIFT normalization** as it approximates the **Hellinger kernel** and reduces the dominance of large bin responses.

1.4 Exploratory Data Analysis

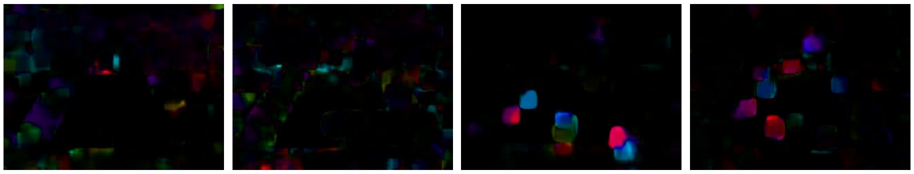
We first visualized Optical Flow Maps and Sample Frames of videos from each class.

Baseball Pitch

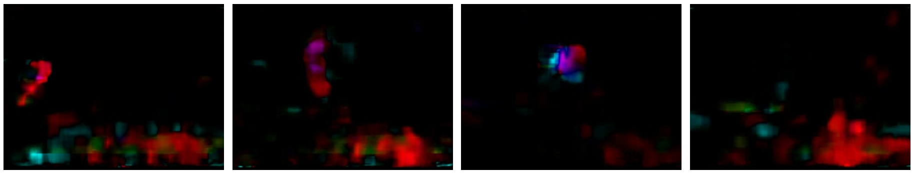


Bench Press

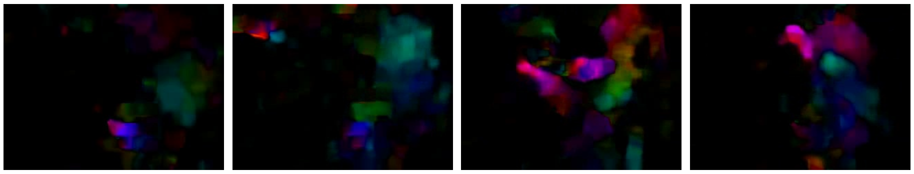
Billards



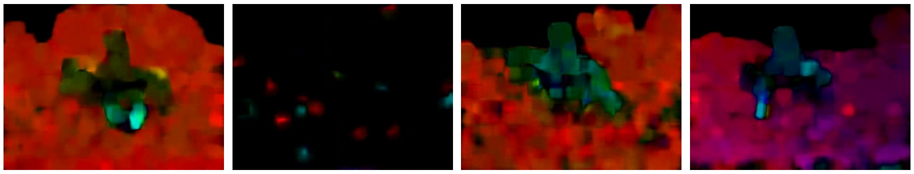
Driving



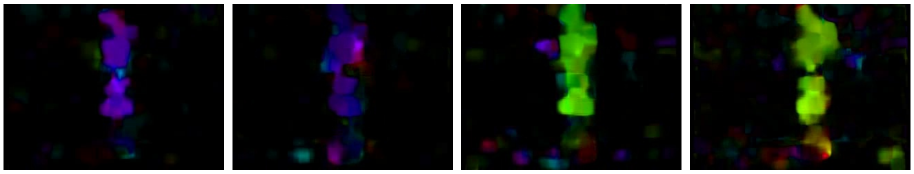
Drumming



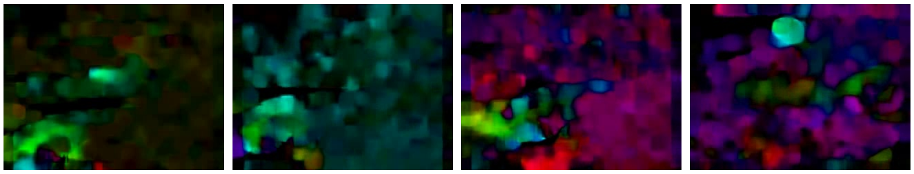
Horse Riding



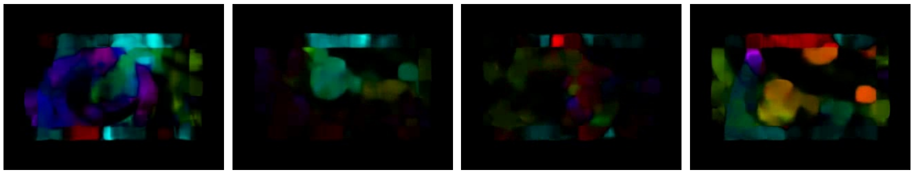
Jump Rope



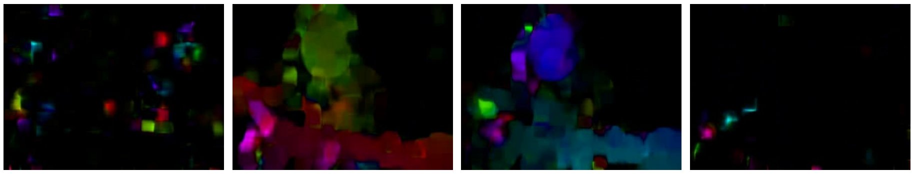
Kayaking



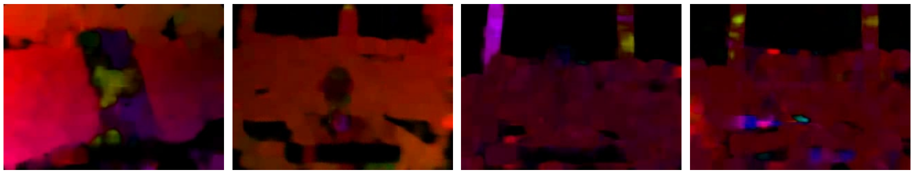
Mixing



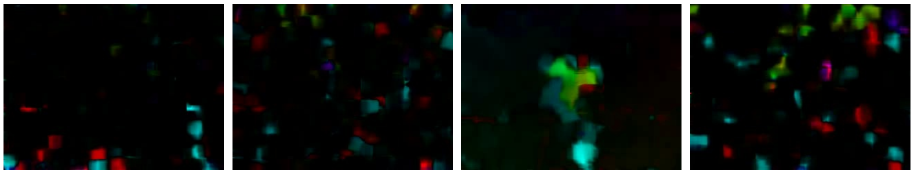
Playing Guitar



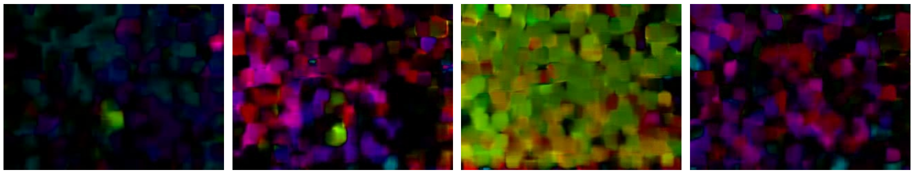
Pole Vault



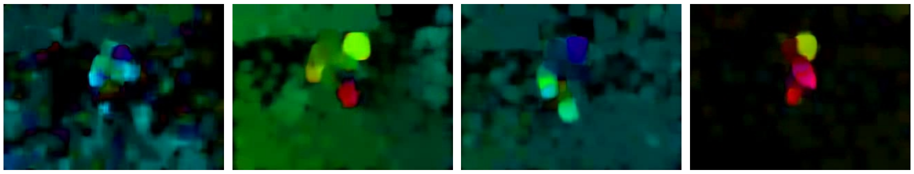
Punch



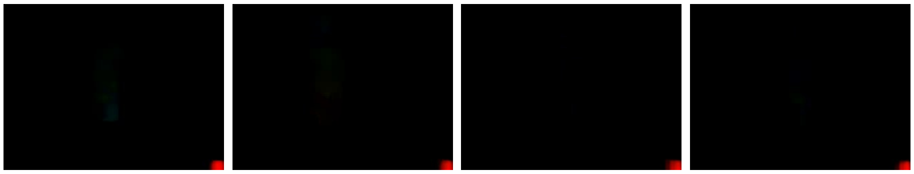
Rock Climbing Indoor



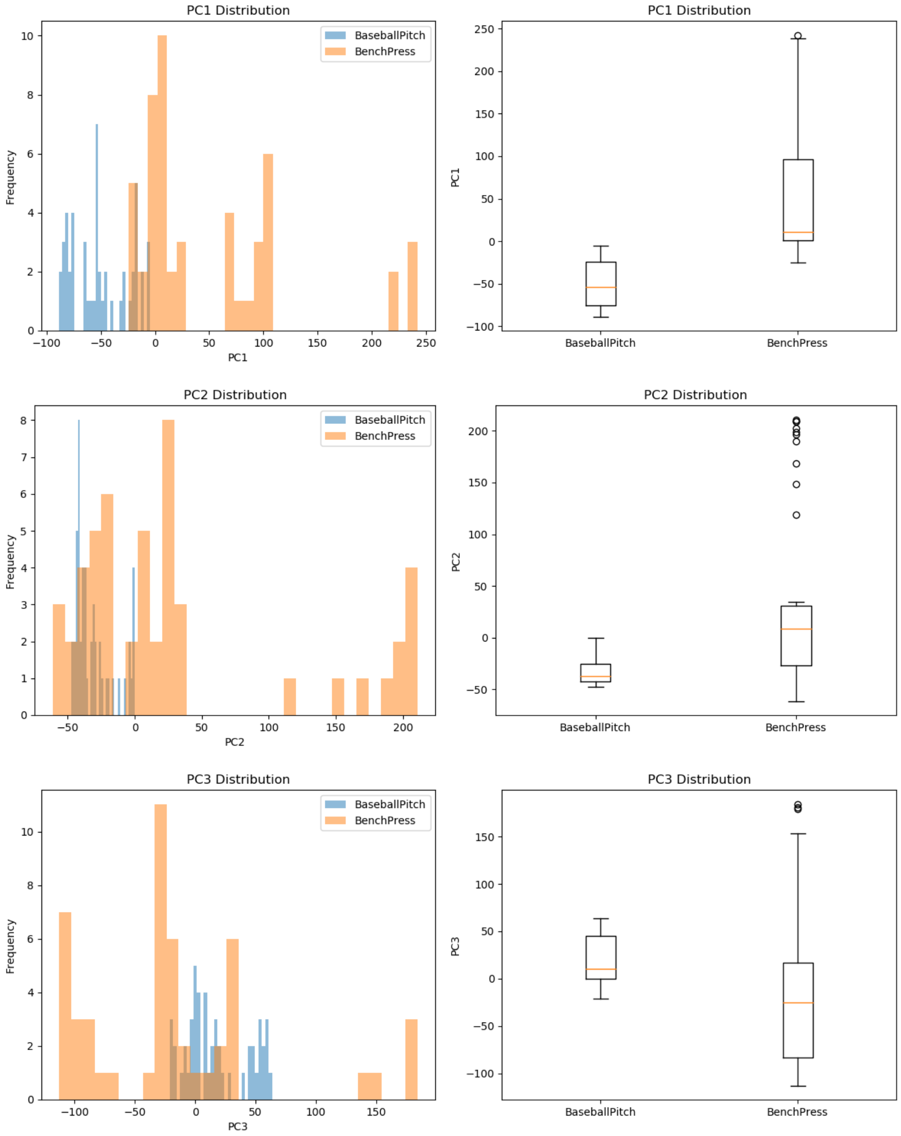
Socccer Juggling

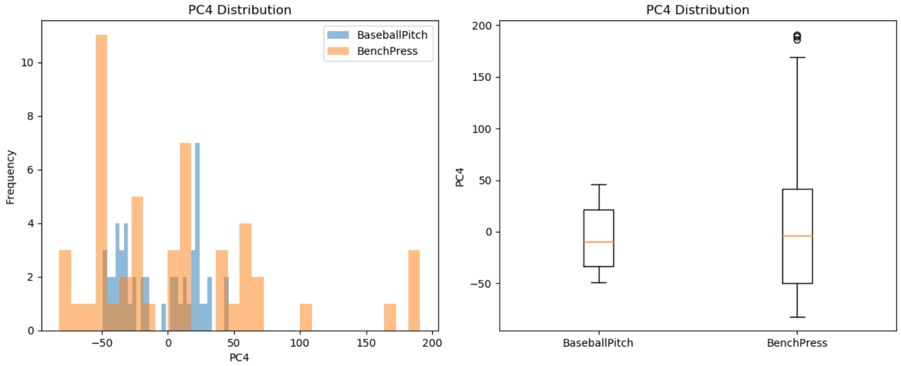


Tennis Swing

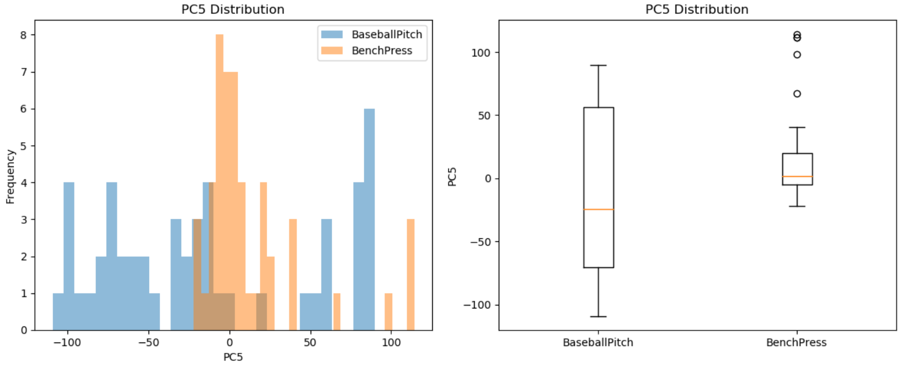


We then plotted Histograms and Boxplots of the top 10 components of 2 classes (BaseballPitch and BenchPress) obtained after PCA of :

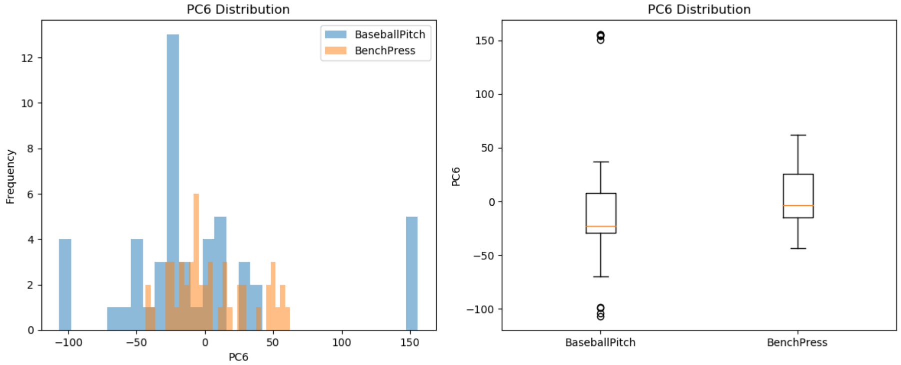



PC4 Distribution

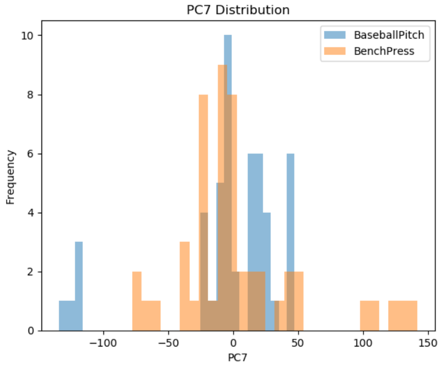
PC4 Distribution

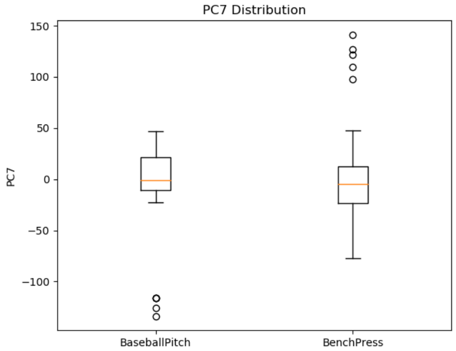

PC5 Distribution

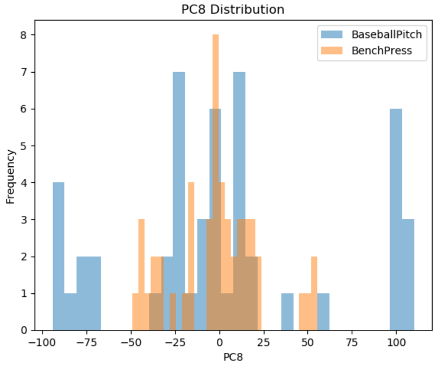
PC5 Distribution

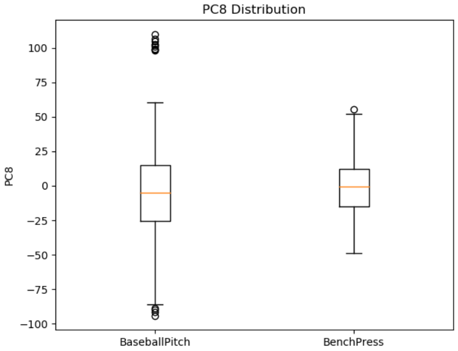

PC6 Distribution

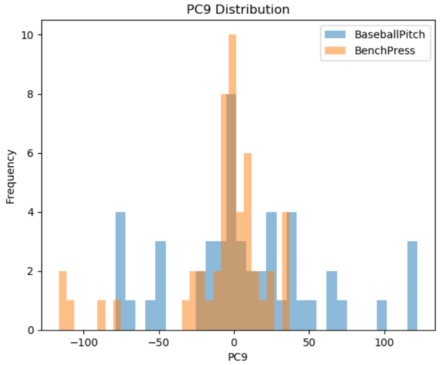
PC6 Distribution

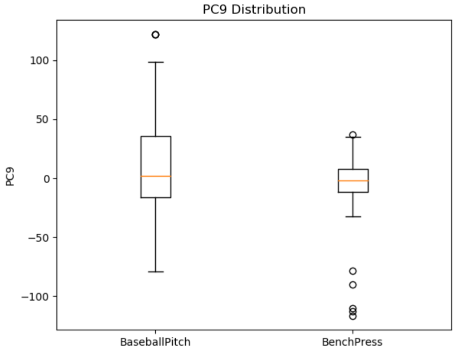

PC7 Distribution
BaseballPitch
BenchPress
Frequency
0
2
4
6
8
10
−100
−50
0
50
100
150
PC7

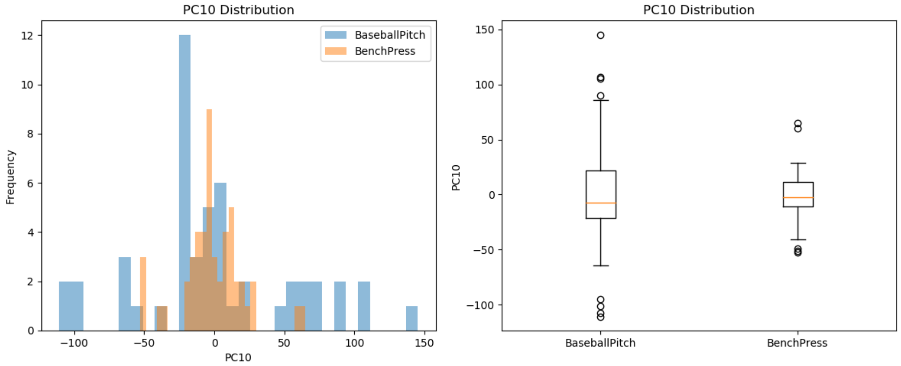

PC7 Distribution
PC7
150
100
50
0
−50
−100
BaseballPitch
BenchPress


PC8 Distribution
BaseballPitch
BenchPress
Frequency
0
1
2
3
4
5
6
7
8
−100
−75
−50
−25
0
25
50
75
100
PC8


PC8 Distribution
PC8
100
75
50
25
0
−25
−50
−75
−100
BaseballPitch
BenchPress


PC9 Distribution
BaseballPitch
BenchPress
Frequency
0
2
4
6
8
10
−100
−50
0
50
100
PC9


PC9 Distribution
PC9
100
50
0
−50
−100
BaseballPitch
BenchPress



ANOVA test:

We performed ANOVA Test to get F-statistic and p-value of the top 10 components.

Component	**F-statistic**	**p-value**
PC1	82.4258	1.2071e-14
PC2	26.0589	1.6267e-06
PC3	9.3104	0.00293
PC4	1.7553	0.18829
PC5	6.4123	0.01292
PC6	1.2085	0.27432
PC7	0.8327	0.36374
PC8	0.1006	0.75182
PC9	4.2076	0.04291
PC10	0.0293	0.86455

F-statistics and p-values for the first 10 principal components.

Key observations:

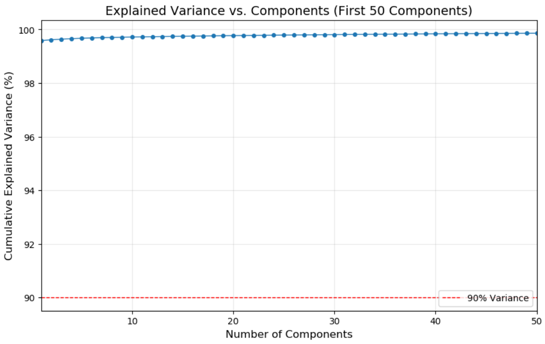
- Top discriminating component: PC1 (F=82.4, p=0.0000)
- Significant components at α=0.05 : 5 out of 10

Takeaways from EDA

PC1 alone captures significant differences. However, class separation requires multiple complementary features (not just 1 dominant component as 5/10 components show statistically significant differences (α=0.05)).

1.5 Dimensionality Reduction and Clustering:

We decided upon the ideal number of components to retain along with the parameters of the clustering algorithms by GridSearch. Dataset was split into 60 training, 20% validation and 20% test data. Below are the results for each method:



Explained Variance Plot

1.5.1 Without using Motion Boundary Histogram features

K Means

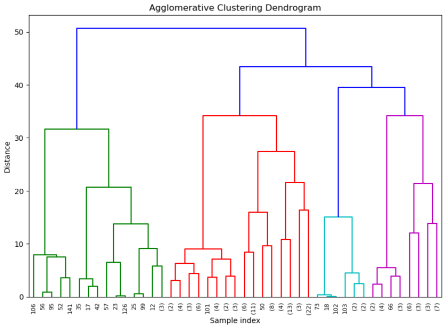
- Number of Components retained: 5
- Best Parameters: 10
- ARI:0.201

Spectral

- Number of Components retained: 5
- Best Parameters: nearest_neighbors
- ARI:0.183

Agglomerative

- Number of Components retained: 5
- Best Parameters: ward
- ARI:0.188



Dendogram for Agglomerative Clustering without using MBH

1.5.2 Using all features

K Means

- Number of Components retained: 5
- Best Parameters: n_init = 10
- ARI:-0.0063

Spectral

- Number of Components retained: 5
- Best Parameters: affinity = nearest_neighbors
- ARI:005

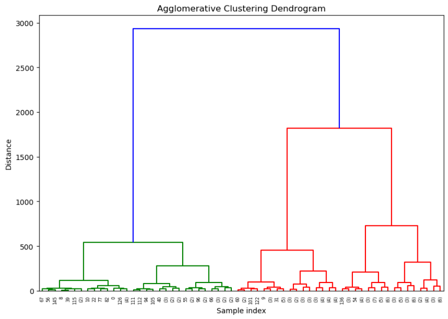
Agglomerative

- Number of Components retained: 5
- Best Parameters: linkage = ward
- ARI:0.004

1.6 Conclusion

We see that the ARI obtained by using all features was exceptionally low. Removing MBH features increased this significantly in all clustering algorithms. This is because of the MBH features were extracted at an exceptionally low fps due to computational constraints, and hence added more noise than information in the feature set.

However, very simpler approaches gave us much greater ARI (as reported below in the competitive part) than any clustering implementation. This can



Dendogram for Agglomerative Clustering using all features

be attributed to the fact perhaps features like Optical Flow and MBH add more noise than information after averaging through all the frames, as well as clustering is not a very suitable approach for a task like action classification.

2 Competitive Part

We made 2 models:

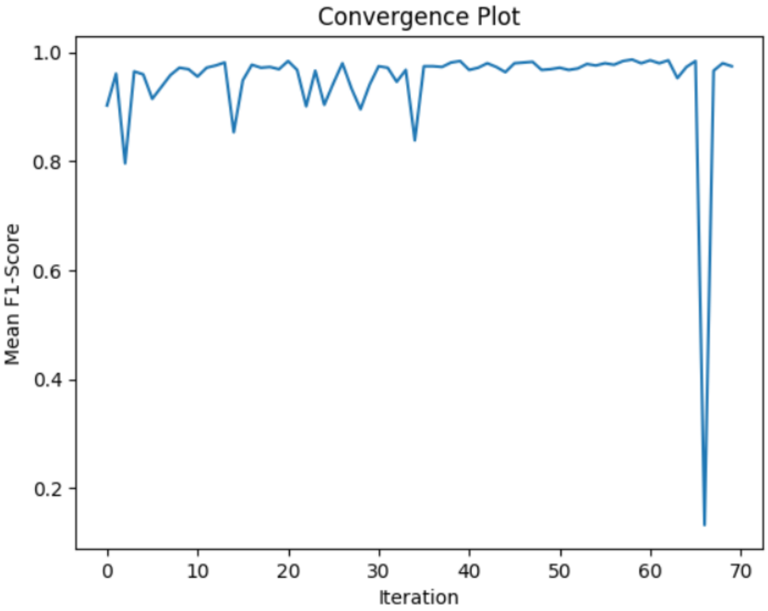
2.1 Model 1

The **ARI** was 1.00

We first extracted features using the **I3D model**. We then dropped the features which had a **range of less than 0.8**. Then we passed the remaining **24 features** through a Random Forest Classifier. To optimize the classifier's performance, we applied **Bayesian Optimization**, which yielded the following best parameters:

- Criterion: entropy
- Max Depth: 20
- Max Features: sqrt
- Min Samples per Leaf: 1
- Min Samples per Split: 2
- Number of Estimators: 300

Below is the convergence curve illustrating the optimization process:



2.2 Model 2

The **ARI** was 1.00

For this model, we extracted 2048D features using **ResNet50** and passed them through a **Random Forest Classifier**. The model was trained with the following parameters:

- Criterion: gini
- Max Depth: 20
- Max Features: auto
- Min Samples per Leaf: 6
- Min Samples per Split: 2
- Number of Estimators: 296
- Random State: 42