

Rock Paper Scissors

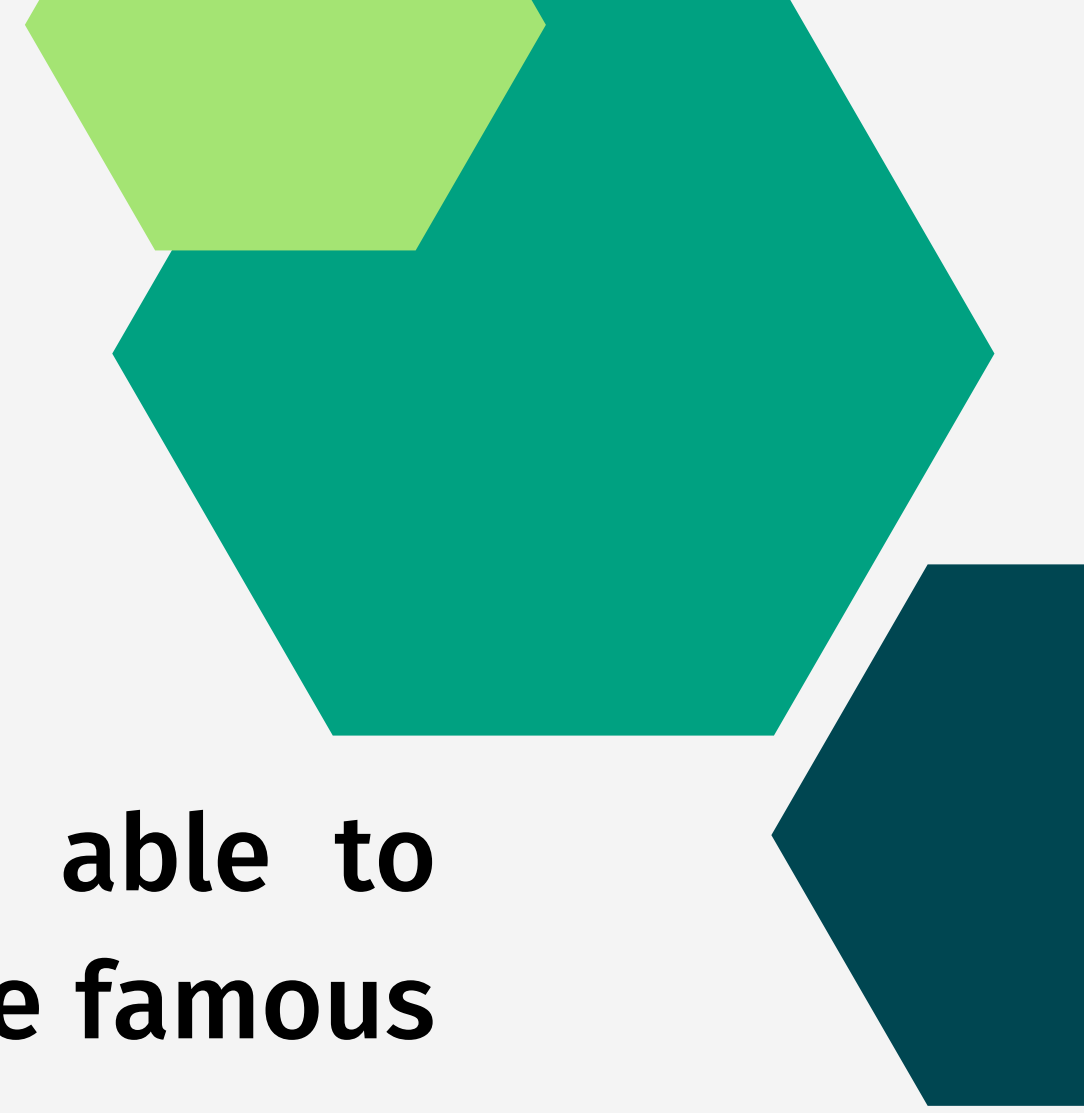
Gesture Recognition



Goal of the Project

The goal of this project is to develop a classifier able to correctly recognize the gesture among the three of the famous rock paper scissor game. The input is an image

It would be interesting to apply the techniques used in this project for a more noble purpose, for example the recognition of sign language gestures



Raw Dataset



Paper
712 Images



Scissors
750 Images



Rock
726 Images

All the hand in the images are pointing in the same direction, the background is green and the light is always the same; there isn't much variability

The dataset is not fully balanced but is very close to it

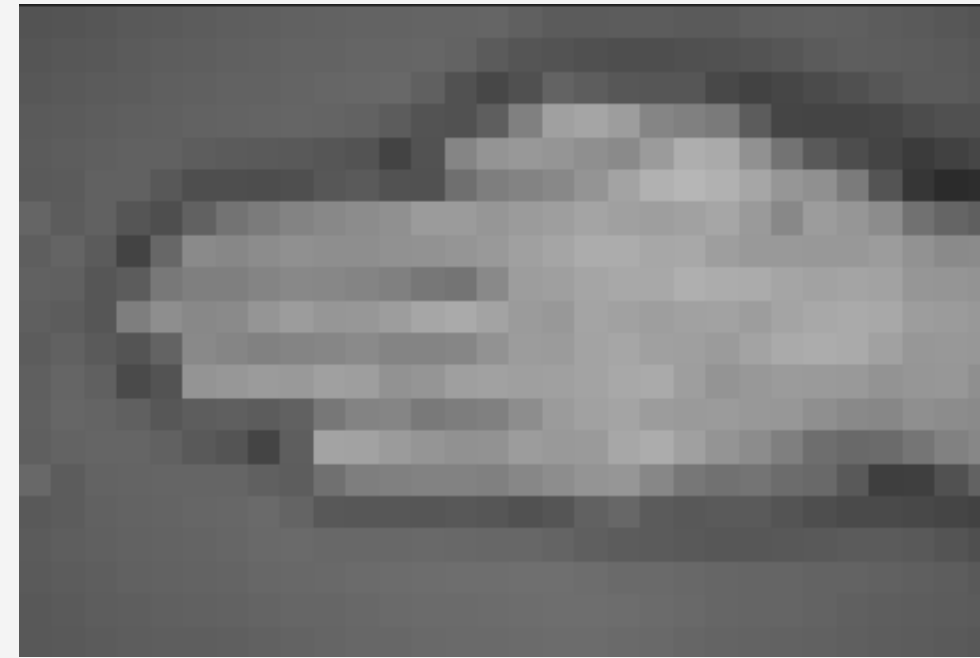
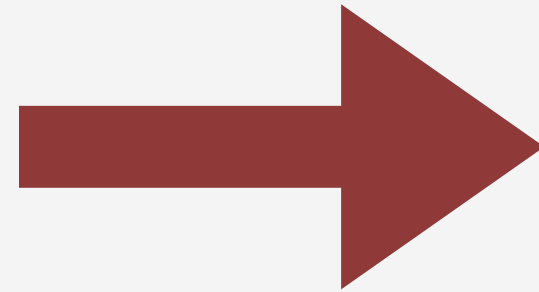
Source: www.kaggle.com/drgfreeman/rockpaperscissors

Data Transformation

- Pixel Values as Features



200x300 pixels



20x30 pixels

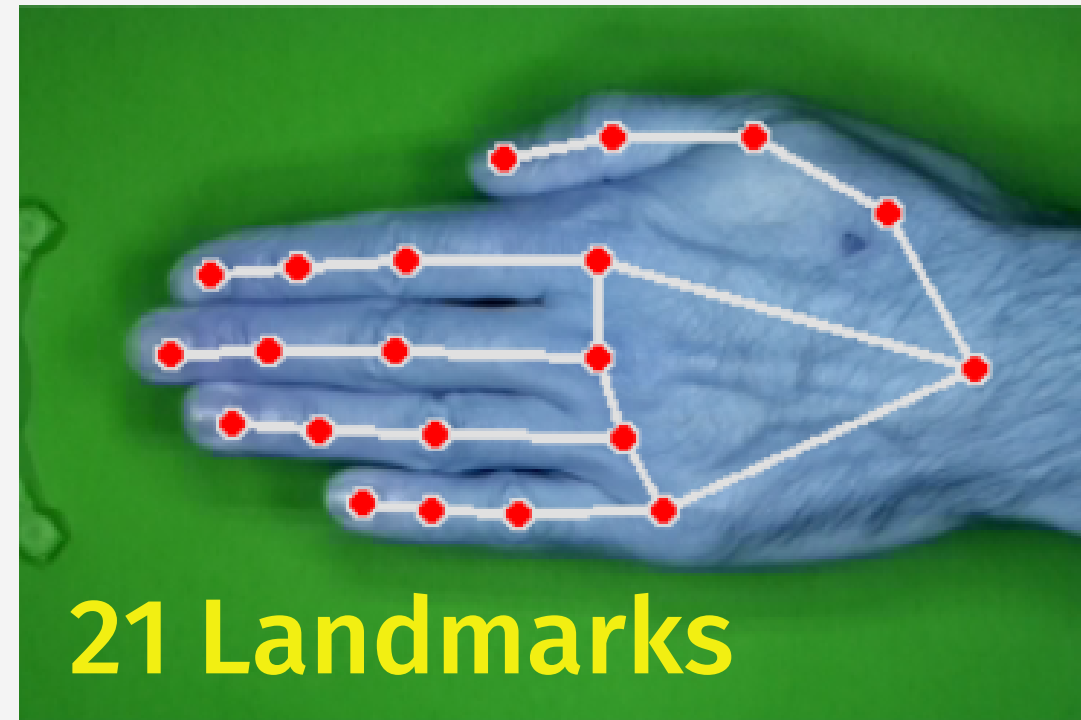
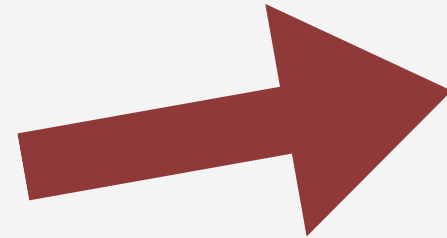
Records: 2188

Features: 600



Data Transformation

- Landmarks as Features



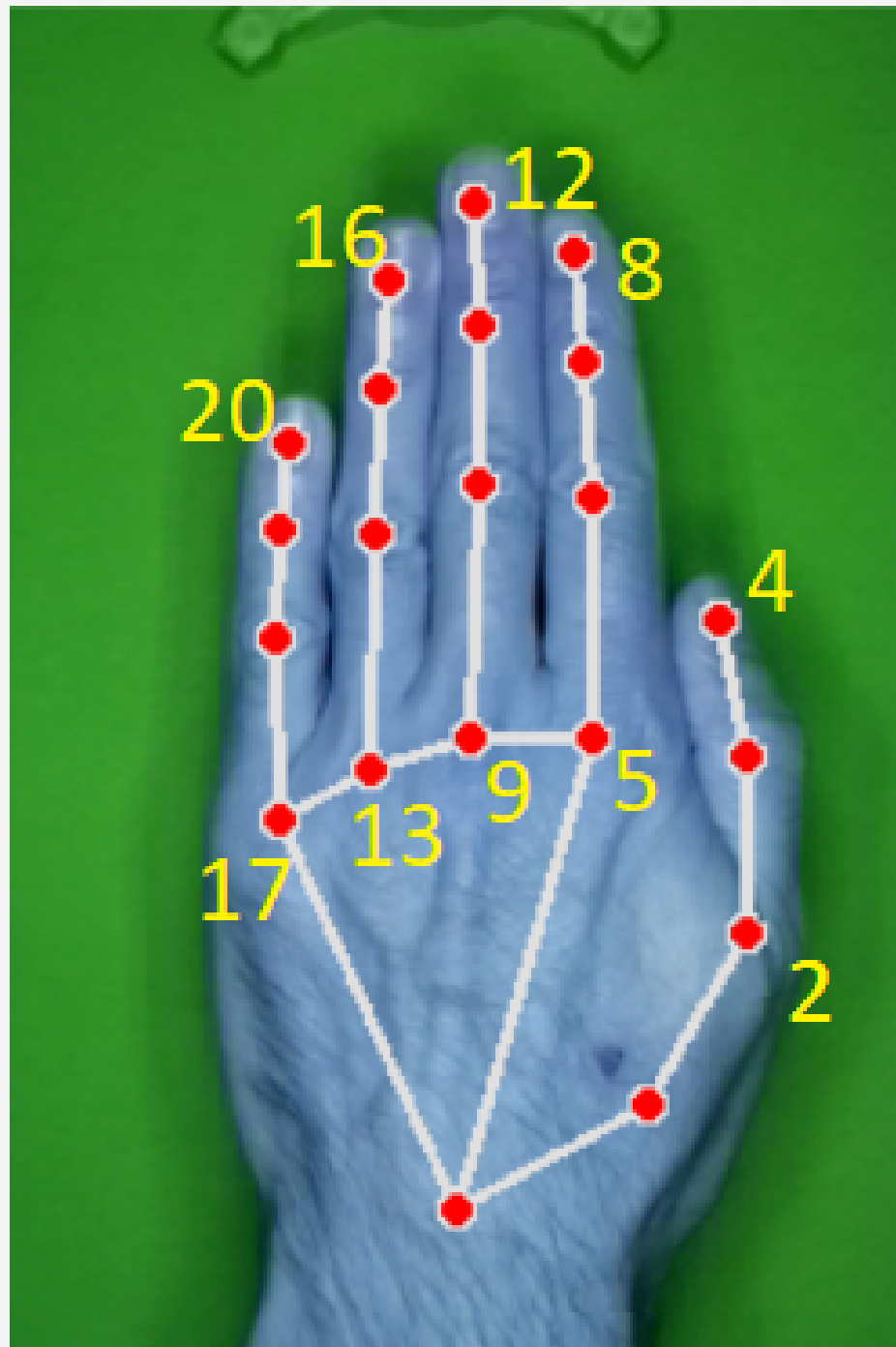
At each Landmark are associated 3 values that is the coordinates of the Landmark itself. Since all hand are placed at the same distance from the camera I discarded the third coordinate which indicates the depth.

Records: 2091

Features: 42

Data Transformation

- Distances as Features



Euclidian Distance between the landmarks:

(4, 2) (5, 8) (9, 12) (13, 16) (17, 20)

Records: 2091

Features: 5

Data Augmentation

The Dataset created with the distances is unaffected by the fact that all the hand in the images point in the same direction

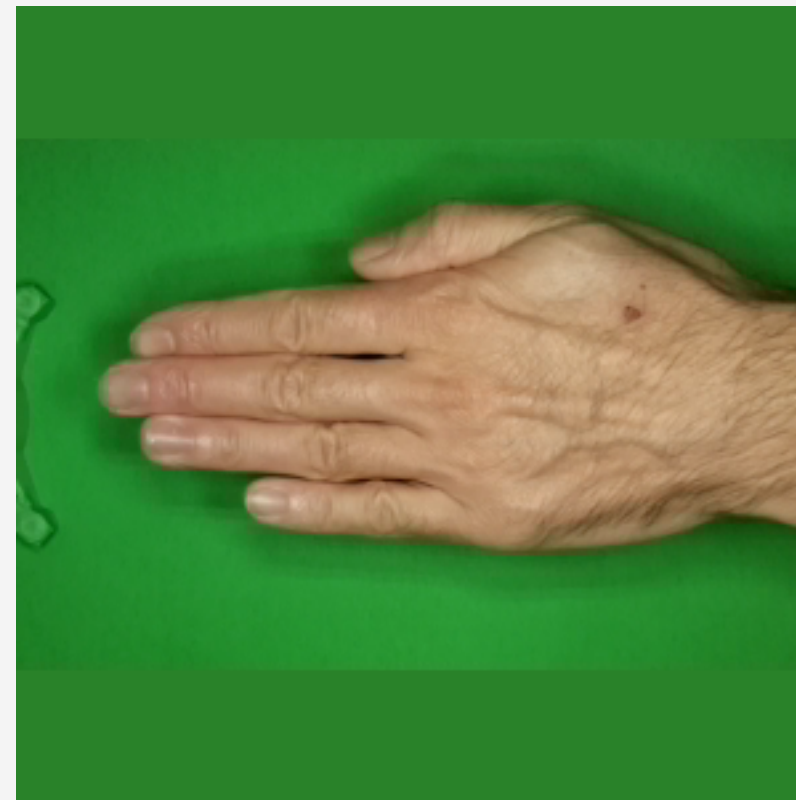
Instead, a Classifier trained on ones of the other two probably will not be able to correctly classify an image where the hand is directed in a different way



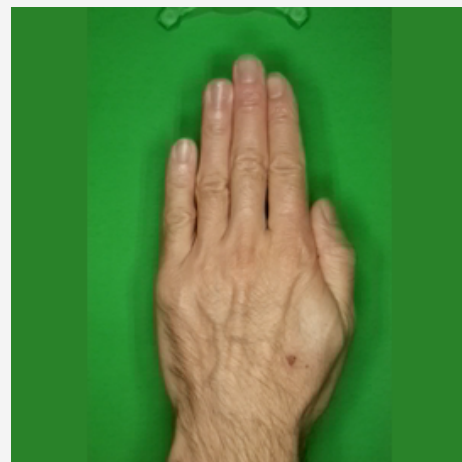
Data Augmentation



→
Padding



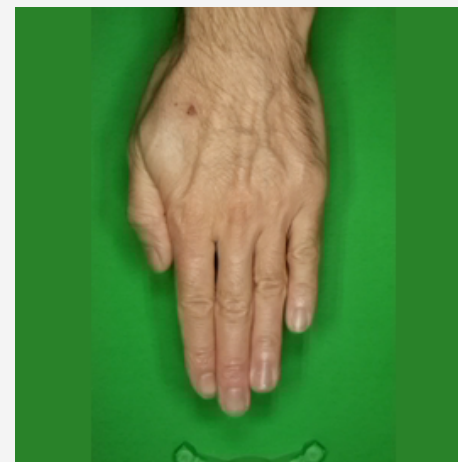
300x300



Rotation
→



Rotation
→



Pixels

Records: 2188
Features: 600



Records: 8752
Features: 625

Landmarks

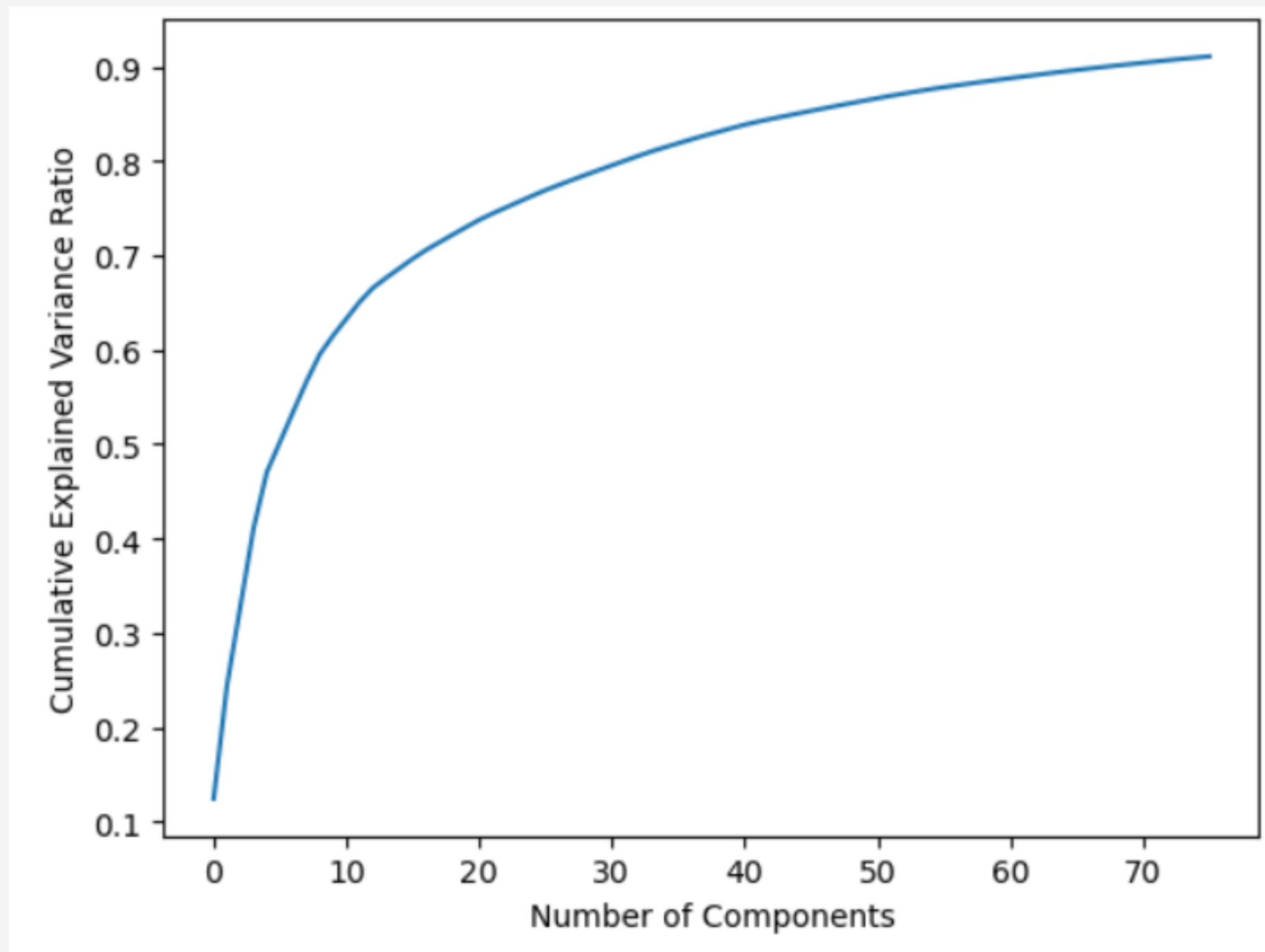
Records: 2091
Features: 42



Records: 4219
Features: 42

PCA

In the Dataset that have the pixel values as Features the Number of Features is too high. I decide to performe Principal Component Analysis in order to reduce it



625 Features



70 Features

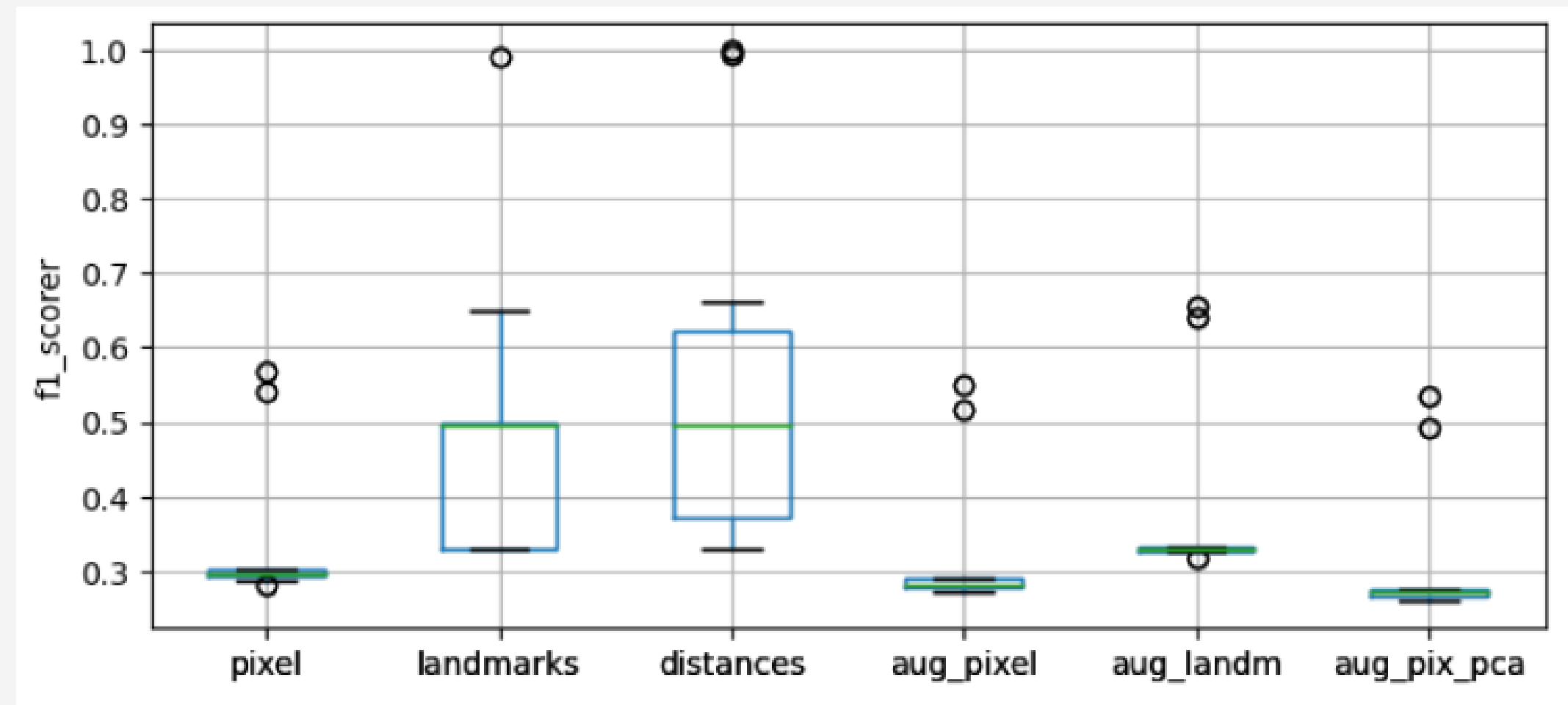
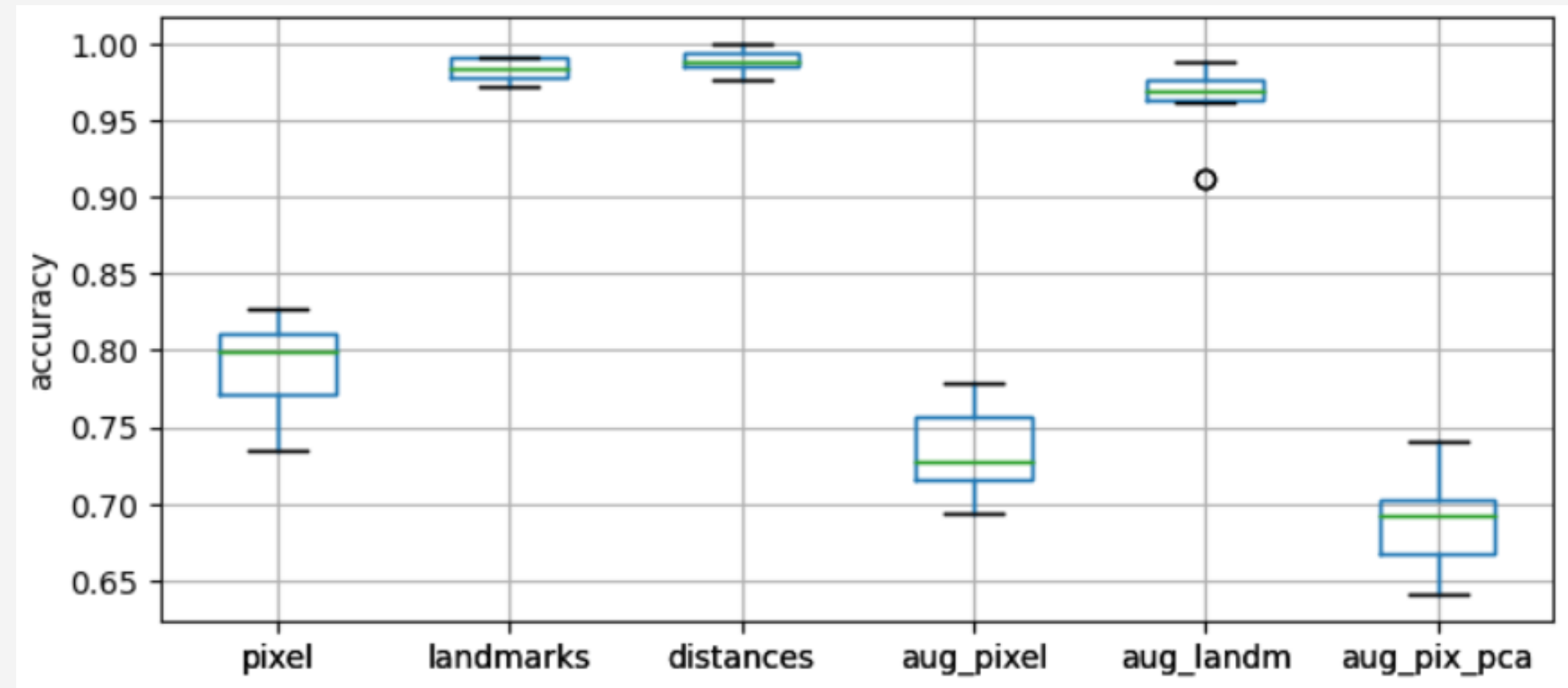


Classification

Decision Tree (criterion: gini)

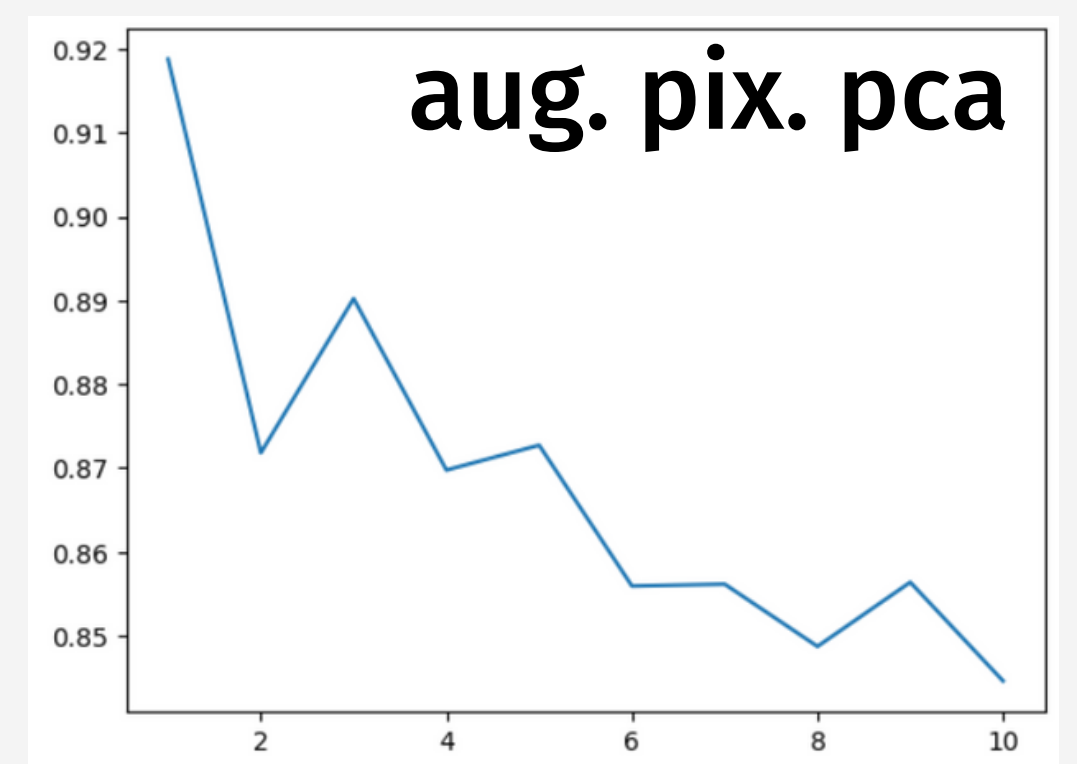
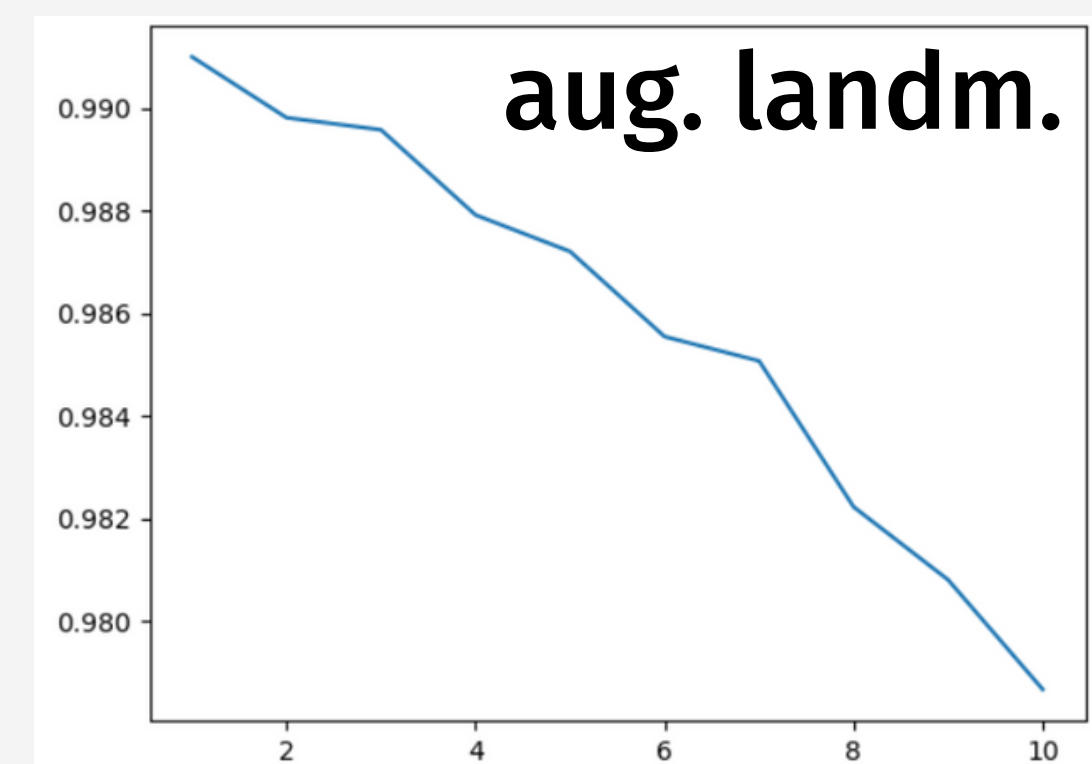
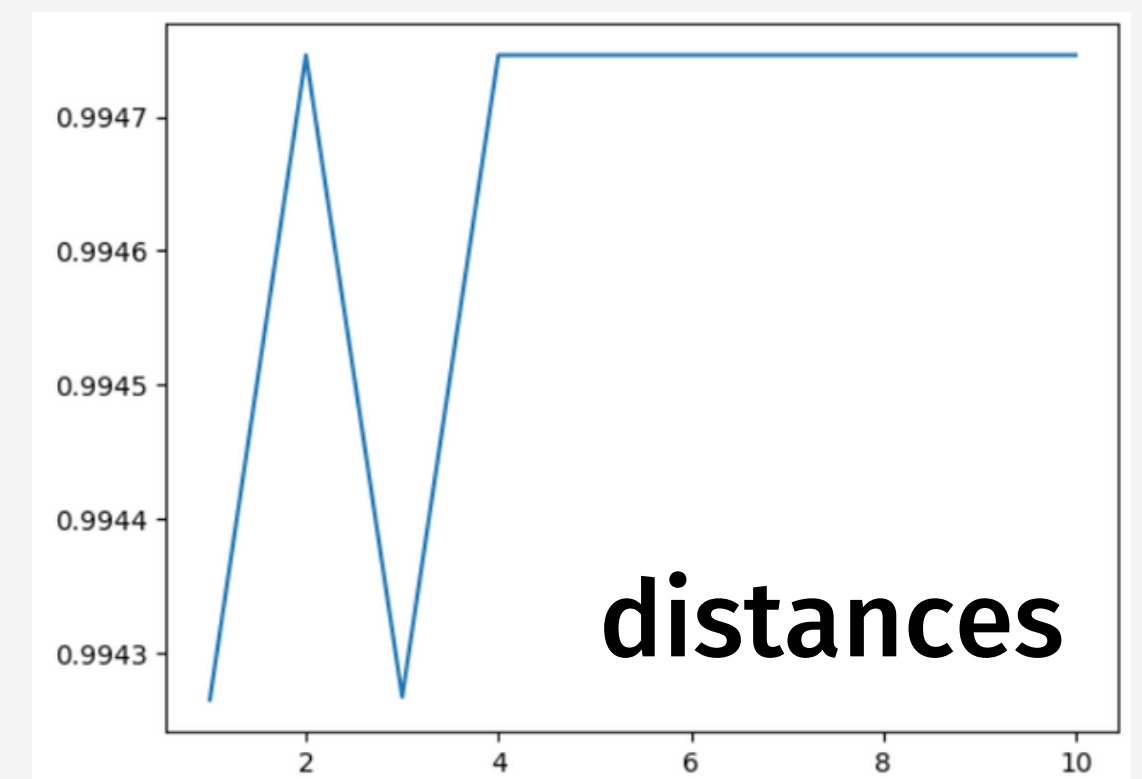
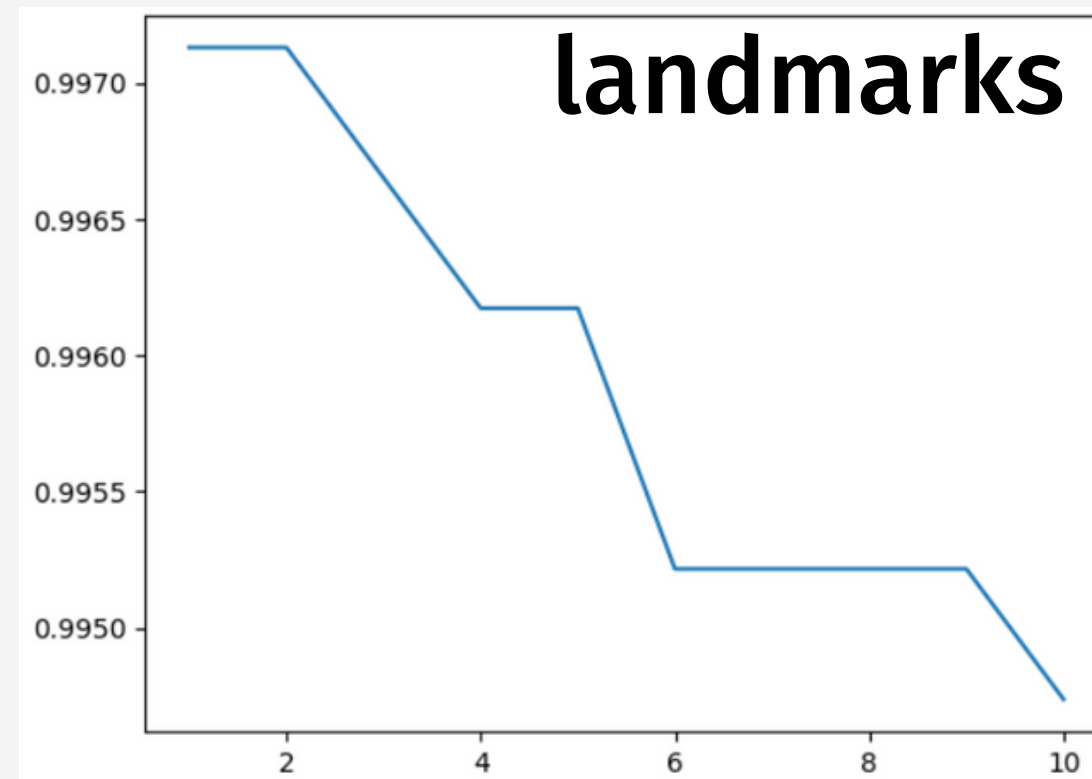
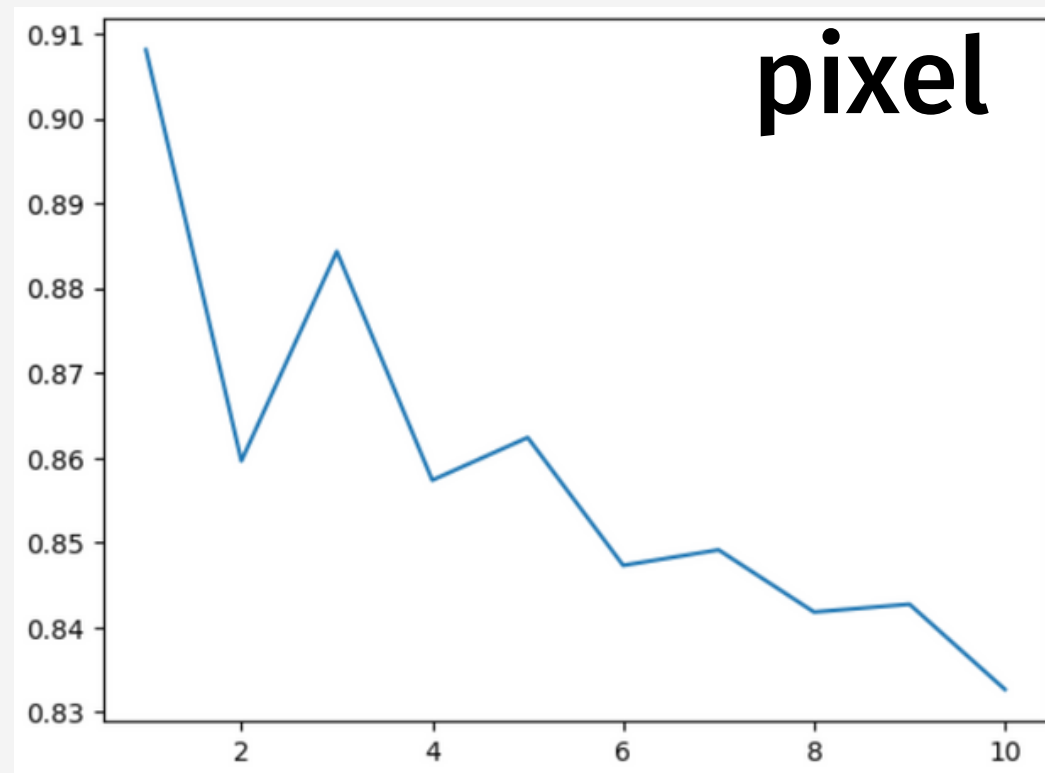
Confusion Matrix:

	paper	rock	scissors
	683	5	4
	5	680	1
scissors	6	1	706
	paper	rock	scissors



Classification

K-Nearest Neighbor ($k = 1 \div 10$)



Classification

Pixel

Classifier
Decision Tree (gini)
Decision Tree (entropy)
K-Nearest Neighbor
Naive Bayesian
Random Forest (50)
Random Forest (100)

Accuracy	F1	Score time
0.791	0.346	0.0197
0.797	0.348	0.0205
0.908	0.394	0.1922
0.763	0.335	0.0410
0.902	0.377	0.0420
0.914	0.380	0.0555

Landmarks

Accuracy	F1	Score time
0.983	0.494	0.0107
0.985	0.528	0.0086
0.997	0.798	0.0484
0.944	0.531	0.0116
0.994	0.664	0.0253
0.993	0.698	0.0402

Classification

Distances

Aug. Pixel

Classifier
Decision Tree (gini)
Decision Tree (entropy)
K-Nearest Neighbor
Naive Bayesian
Random Forest (50)
Random Forest (100)

Accuracy	F1	Score time
0.989	0.563	0.0096
0.989	0.578	0.0226
0.994	0.714	0.0375
0.991	0.564	0.0109
0.992	0.647	0.0239
0.992	0.664	0.0389

Accuracy	F1	Score time
0.732	0.331	0.0344
0.735	0.332	0.0356
0.905	0.394	2.4051
0.649	0.299	0.0986
0.894	0.375	0.0792
0.903	0.377	0.1212

Classification

Aug. Landmarks

Classifier
Decision Tree (gini)
Decision Tree (entropy)
K-Nearest Neighbor
Naive Bayesian
Random Forest (50)
Random Forest (100)

Accuracy	F1	Score time
0.966	0.391	0.0104
0.968	0.408	0.0096
0.990	0.614	0.1582
0.793	0.348	0.0141
0.985	0.530	0.0281
0.985	0.630	0.0408

Aug. Pixel PCA

Accuracy	F1	Score time
0.687	0.318	0.0624
0.681	0.314	0.0543
0.918	0.381	0.5829
0.693	0.316	0.0680
0.857	0.364	0.0850
0.865	0.366	0.1196