Åbo Akademi- Mini Project 3
Human Activity Recognition Using Cellphone
Dimensionality Reduction + DBSCAN

Markku Pulli

7.3.2021

Abstract 3
Introduction 3
Data Observations and Data Processing 3
Methodology/Modeling 5
Results & Conclusions 7
    Test data verification, LDA+DBSCAN 8
References 9

Abstract

Goal of the project is to separate input data to six different clusters according to activity. Several different approaches were tested. Using only the sample data without any dimensionality reduction[1] (DR) algorithm or feature extraction, feature selection, principal component analysis (PCA), kernelPCA and linear discriminant analysis (LDA). After multiple rounds of parameter optimization on each algorithm only LDA shows best separation outcome and performance from all of these. Focus on this report is on non-dimensionality algorithm and LDA. DBSCAN algorithm is used for clustering on both cases. DBSCAN parameter tuning (2) is very time-consuming task. With short Python code range of different parameters were implemented and results analyzed. DBSCAN in the best case can create five clusters with silhouette score 0.805.

Introduction

Dataset – The experiments have been carried out with a group of 30 volunteers within an age bracket of 19-48 years. Each person performed six activities (WALKING, WALKING_UPSTAIRS, WALKING_DOWNSTAIRS, SITTING, STANDING, LAYING) wearing a smartphone (Samsung Galaxy S II) on the waist. Using its embedded accelerometer and gyroscope, we captured 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz. The experiments have been video-recorded to label the data manually. The obtained dataset has been randomly partitioned into two sets, where 70% of the volunteers was selected for generating the training data and 30% the test data.

Data Observations and Data Processing

Data is in shape of 7532*562. Number of samples is 13-fold comparing to features. 561 features will offer an opportunity to use dimensionality reduction algorithm. Also, data is almost evenly spread through different activities. See table 1.

Activity	Description	Samples	share %
1	Walking	1226	16.7%
2	Walking upstairs	1073	14.6%
3	Walking downstairs	986	13.4%
4	Sitting	1286	17.5%
5	Standing	1374	18.7%
6	Laying	1407	19.1%
Total		7352	100.0%

Table 1: Activities and sample counts.

According to data source the data is already scaled and normalized. This can be seen on table 2. First five feature (X1…X5) means and variance for each activity is shown this table.

Activity	X1		X2		X3		X4		X5	
	mean	variance	mean	variance	mean	variance	mean	variance	mean	variance
1	0.276259	0.002536	-0.017767	0.000436	-0.108892	0.001052	-0.31264	0.020869	-0.020279	0.031568
2	0.261925	0.00609	-0.026648	0.001372	-0.120423	0.003625	-0.221069	0.023843	-0.000344	0.040554
3	0.28817	0.009044	-0.016368	0.000732	-0.10586	0.002566	0.139856	0.049337	0.07919	0.05943
4	0.27345	0.001764	-0.012142	0.00105	-0.106576	0.002054	-0.98344	0.001042	-0.936213	0.016919
5	0.279298	0.000404	-0.016124	0.000319	-0.107327	0.001273	-0.985346	0.000523	-0.936012	0.006714
6	0.269187	0.01031	-0.018345	0.005405	-0.107178	0.008057	-0.959476	0.006196	-0.937598	0.021585

Table 2: Data mean and variance for each activity.

From Figure 1. it is visible data is mostly very dense and overlapping. Also, density varies depending on the activity. Without any use of DR algorithm, it will be challenging DBSCAN create clusters without labels.

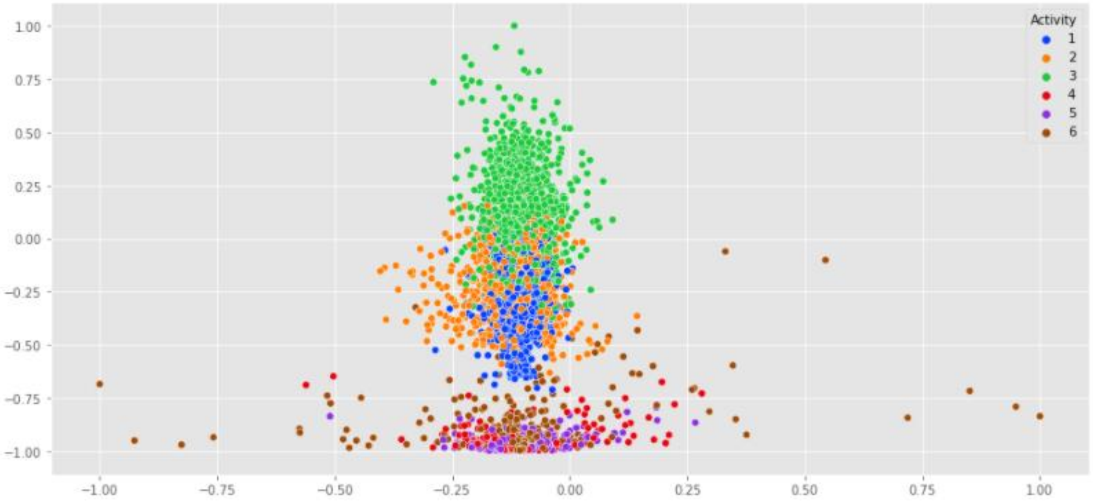


Figure 1: Feature sample data for each activity.

LDA n_component selection was done with help of variance ratio. After analysis four components are selected. This covers 99% of the variance.

Variance of each component 1,2,3,4 = [0.73276802, 0.17514607, 0.05460691, 0.02842483]

Transformed data is shown in figure 2. Separation between clusters is well created. Only cluster 4 and 5 have overlap. Activities 4 & 5 are sitting and standing. Both are mostly still, minimum movement, activities so sensor reading is in same range and separation is challenging.

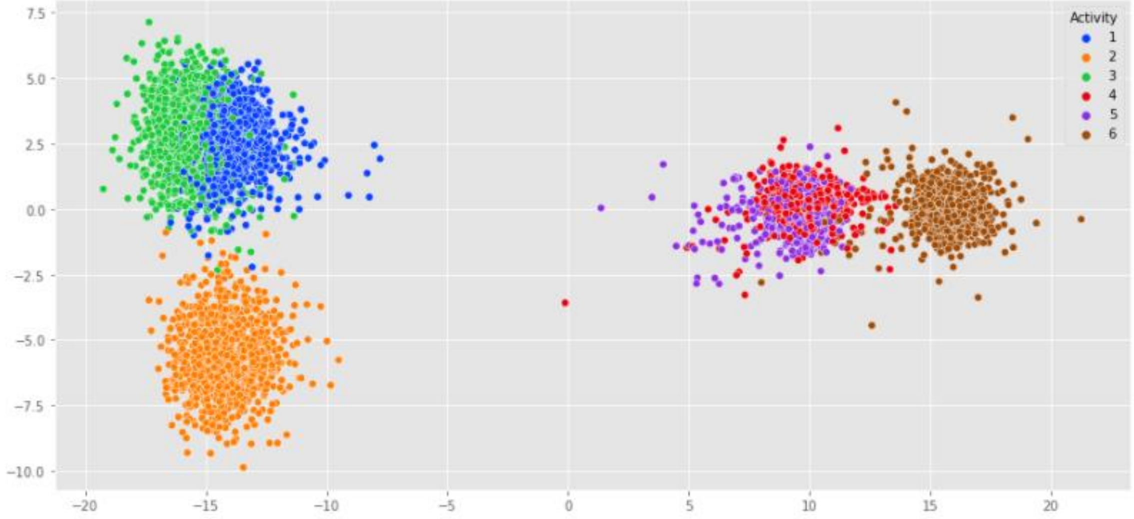


Figure 2: Feature sample data for each activity after LDA is completed.

Link to the dataset:
https://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones

Methodology/Modeling

Nearest neighbor algorithm is used to find initial set of DBSCAN parameters, see figure 3. Parameters are eps and min_samples. Eps is the distance between samples and min_sample is minimum required number of samples to create a cluster.

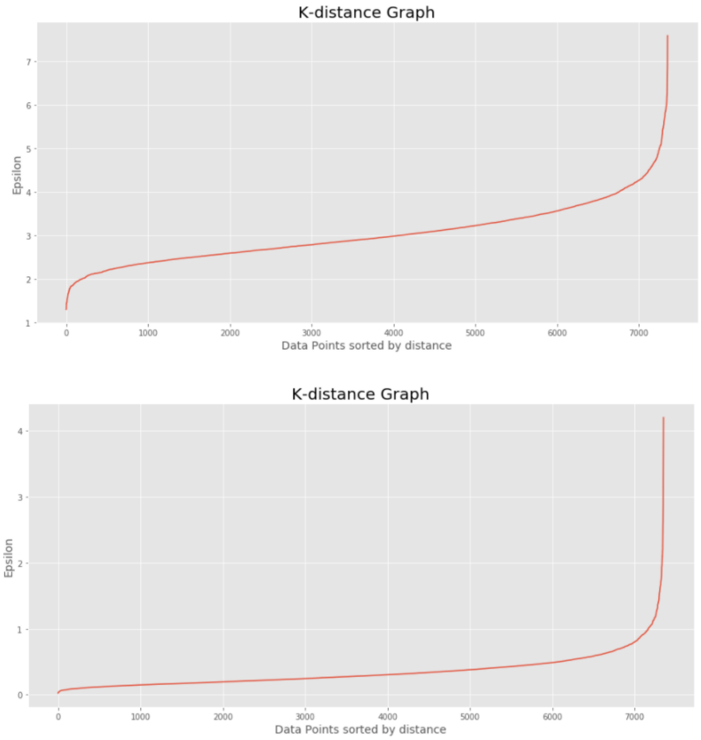
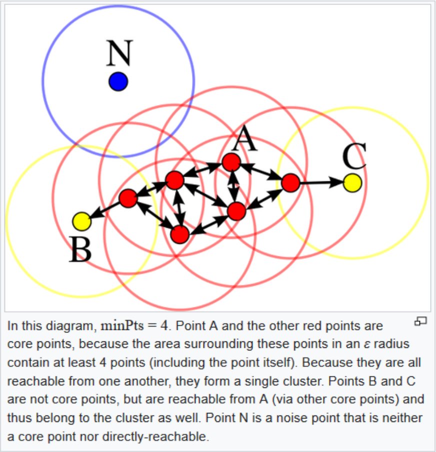


Figure 3: K distance curve. Top curve without dimensionality reduction.

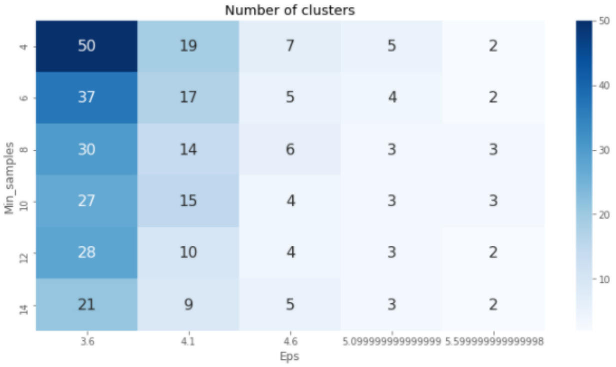
Figure 4 shows details of how algorithm works. Basically >=min_sample points within distance eps create a cluster. Same methodology is used around core points until no new points are reachable.



In this diagram, $\mathrm{minPts} = 4$. Point A and the other red points are core points, because the area surrounding these points in an ε radius contain at least 4 points (including the point itself). Because they are all reachable from one another, they form a single cluster. Points B and C are not core points, but are reachable from A (via other core points) and thus belong to the cluster as well. Point N is a noise point that is neither a core point nor directly-reachable.

Figure 4: DBSCAN methodology.[Wikipedia]

Initial set of parameters was set to eps=[0.9…1.8] step=0.1 and min_samples=[4…16] step=2. Table 3 shows the cluster count DBSCAN has created for each parameter pair. Cluster count varies from 5 to 12. Sample data cluster count is 6.



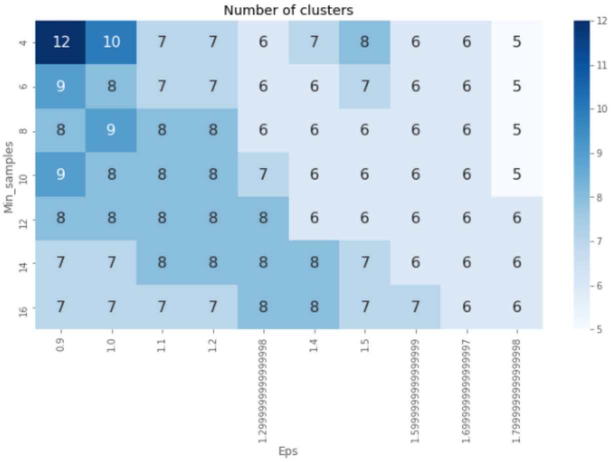
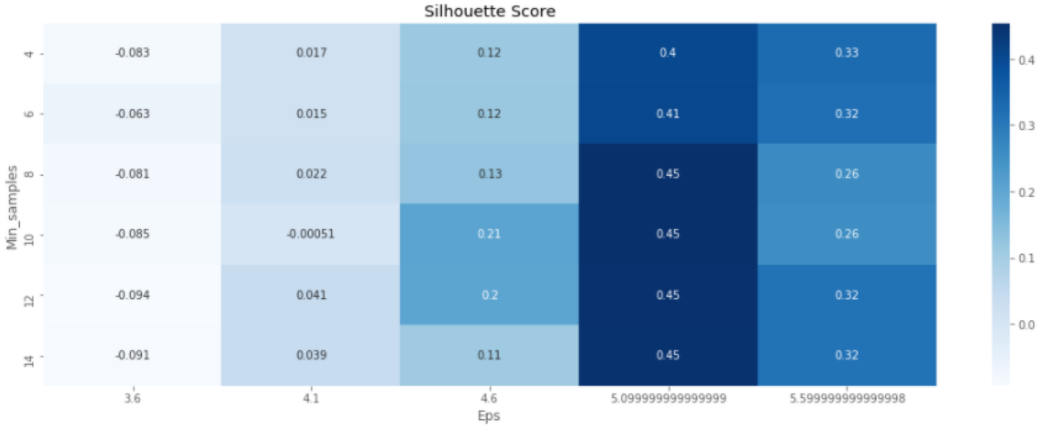
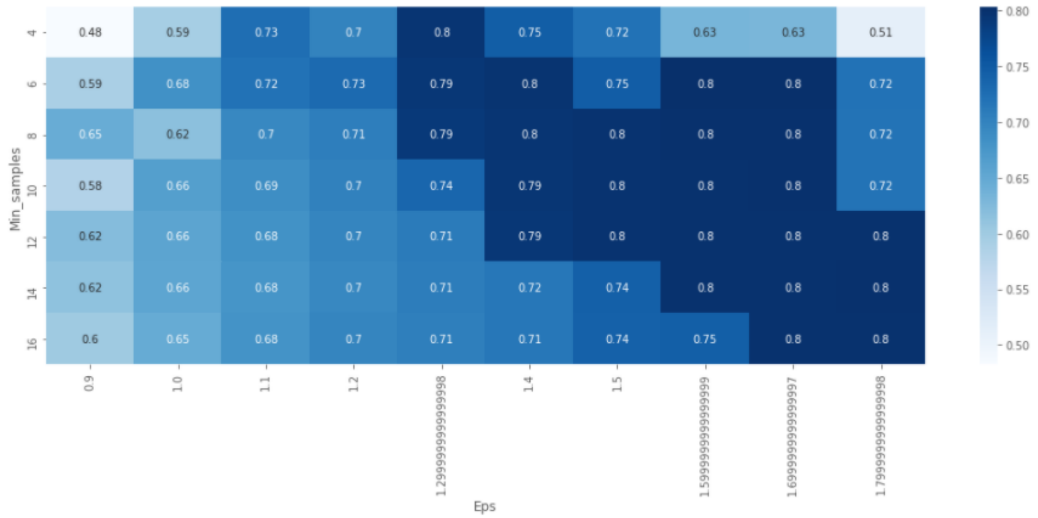


Table 3: Tables include cluster count for each parameter pair. Top table w/o dimensionality reduction.

Table 4 includes silhouette[2] score for same. Range is {-1…1}. Score close to '1' is most optimal. Top table without DR has highest score 0.45. This is with three separated clusters. With LDA score goes up to 0.8 while cluster count is 6.





Min_samples \ Eps	0.9	1.0	1.1	1.2	1.2999999999999998	1.4	1.5	1.5999999999999999	1.6999999999999997	1.7999999999999998
4	0.48	0.59	0.73	0.7	0.8	0.75	0.72	0.63	0.63	0.51
6	0.59	0.68	0.72	0.73	0.79	0.8	0.75	0.8	0.8	0.72
8	0.65	0.62	0.7	0.71	0.79	0.8	0.8	0.8	0.8	0.72
10	0.58	0.66	0.69	0.7	0.74	0.79	0.8	0.8	0.8	0.72
12	0.62	0.66	0.68	0.7	0.71	0.79	0.8	0.8	0.8	0.8
14	0.62	0.66	0.68	0.7	0.71	0.72	0.74	0.8	0.8	0.8
16	0.6	0.65	0.68	0.7	0.71	0.71	0.74	0.75	0.8	0.8

Table 4: Tables include silhouette score for each parameter set. Top table is w/o dimensionality reduction.

From the table 4 following parameter sets are selected. Highest Silhouette score will produce best results.

Non-DR: eps=5.1 and min_samples=10 is chosen, silhouette score is 0.45.

With LDA: eps=1.6 and min_samples=10 is chosen, silhouette score is 0.8.

As seen non-DR case highest silhouette score can separate only 3 clusters and the score=0.45 is still extremely low. We can make conclusion non-DR case is not optimal to use in this situation.

DBSCAN Computation times	DBSCAN Parameter tuning	Final DBSCAN fit [s]
Without dimensionality reduction	4145s, 30 sets	138
LDA	226s, 70 sets	3.2

Table 5: Model performance comparison.

Results & Conclusions

Below graphs have results of each exercise. These are plots with highest silhouette score. Even when trying other parameters on non-DR results are not any better. With LDA results show significant improvement even there are two clusters merged.

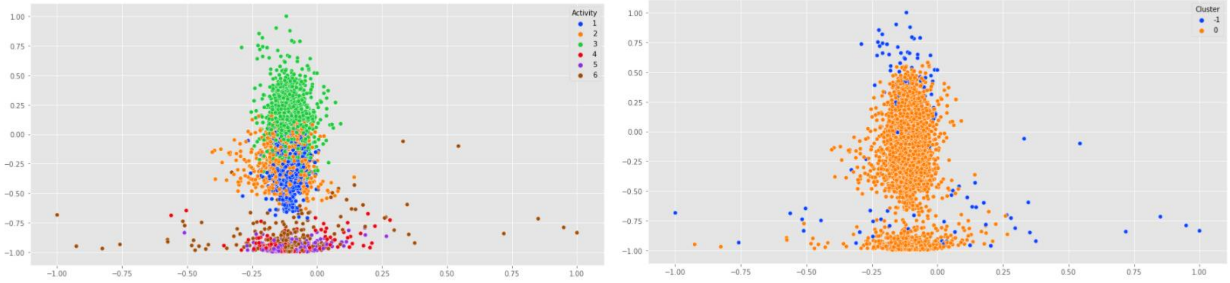


Figure 5: DBSCAN only. Left is with original cluster allocation. Right is cluster allocation with DBSCAN clusters.

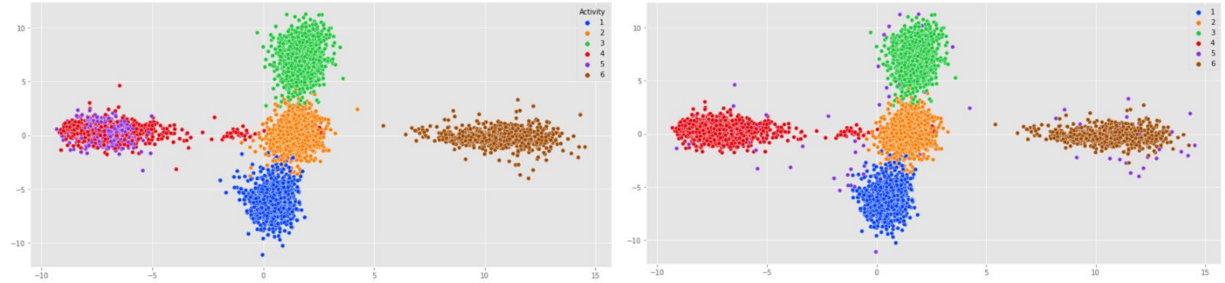


Figure 6: LDA & DBSCAN. Left is with original cluster allocation. Right is cluster allocation with DBSCAN clusters.

Performance KPI	non-DR	LDA
Estimated number of clusters:	4	5
Estimated number of noise points:	357	79
Homogeneity:	0.022	0.850
Completeness:	0.170	0.963
V-measure:	0.040	0.903
Adjusted Rand Index:	0.006	0.788
Adjusted Mutual Information:	0.038	0.903
Silhouette Coefficient:	-0.202	0.801

Table 6: DBSCAN performance comparison, non-DR vs LDA.

When comparing result between two exercises it is clearly seen when data feature count is high dimensionality reduction algorithm has high impact on model fit performance. Computation time is fraction of the non-DR case and accuracy is much better. This is especially important when implementing in a production environment.

Test data verification, LDA+DBSCAN

Using the test data same performance is seen as during the training. Cluster separation is clear only activities 4 and 5 are merged.

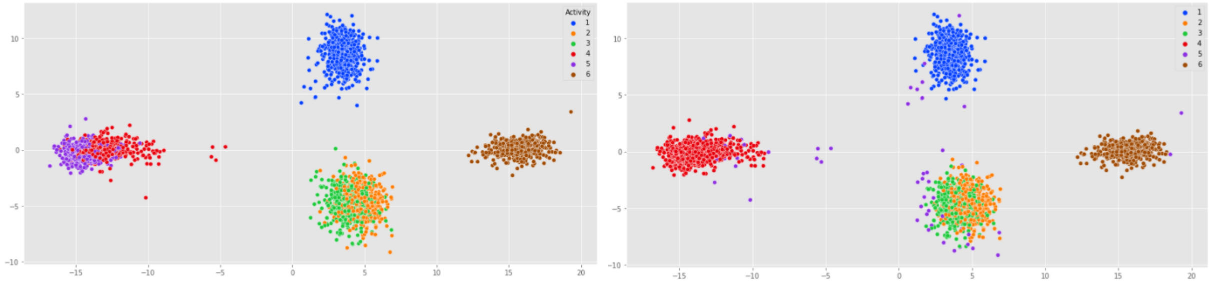


Figure 7: LDA & DBSCAN. Left is with original cluster allocation. Right is cluster allocation with DBSCAN clusters.

Estimated number of clusters: 5
Estimated number of noise points: 61
Homogeneity: 0.849
Completeness: 0.940
V-measure: 0.892
Adjusted Rand Index: 0.794
Adjusted Mutual Information: 0.892
Silhouette Coefficient: 0.825

Table 7: Test data model performance.

Next steps

As a final test LDA+kPCA+DBSCAN was tested but results were not showing any improved performance comparing to LDA only. To further improve model performance more studies are required to separate clusters 4 and 5. It is possible DBSCAN is not the optimal algorithm to use in this case. Different models need to be tested to come conclusion.

References

[1] Dimensionality reduction, or dimension reduction, is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation retains some meaningful properties of the original data, ideally close to its intrinsic dimension. Working in high-dimensional spaces can be undesirable for many reasons; raw data are often sparse as a consequence of the curse of dimensionality, and analyzing the data is usually computationally intractable. Dimensionality reduction is common in fields that deal with large numbers of observations and/or large numbers of variables, such as signal processing, speech recognition, neuroinformatics, and bioinformatics.

[2] The Silhouette Coefficient is calculated using the mean intra-cluster distance (a) and the mean nearest-cluster distance (b) for each sample. The Silhouette Coefficient for a sample is (b - a) / max(a, b). To clarify, b is the distance between a sample and the nearest cluster that the sample is not a part of. Note that Silhouette Coefficient is only defined if number of labels is 2 <= n_labels <= n_samples - 1. The best value is 1 and the worst value is -1. Values near 0 indicate overlapping clusters. Negative values generally indicate that a sample has been assigned to the wrong cluster, as a different cluster is more similar. Source: scikit-learn.