Unsupervised Case Study

Data preparation

Combining data from different files:

	id	activity
0	1	WALKING
1	2	WALKING_UPSTAIRS
2	3	WALKING_DOWNSTAIRS
3	4	SITTING
4	5	STANDING

+

	0
0	1 tBodyAcc-mean()-X
1	2 tBodyAcc-mean()-Y
2	3 tBodyAcc-mean()-Z
3	4 tBodyAcc-std()-X
4	5 tBodyAcc-std()-Y

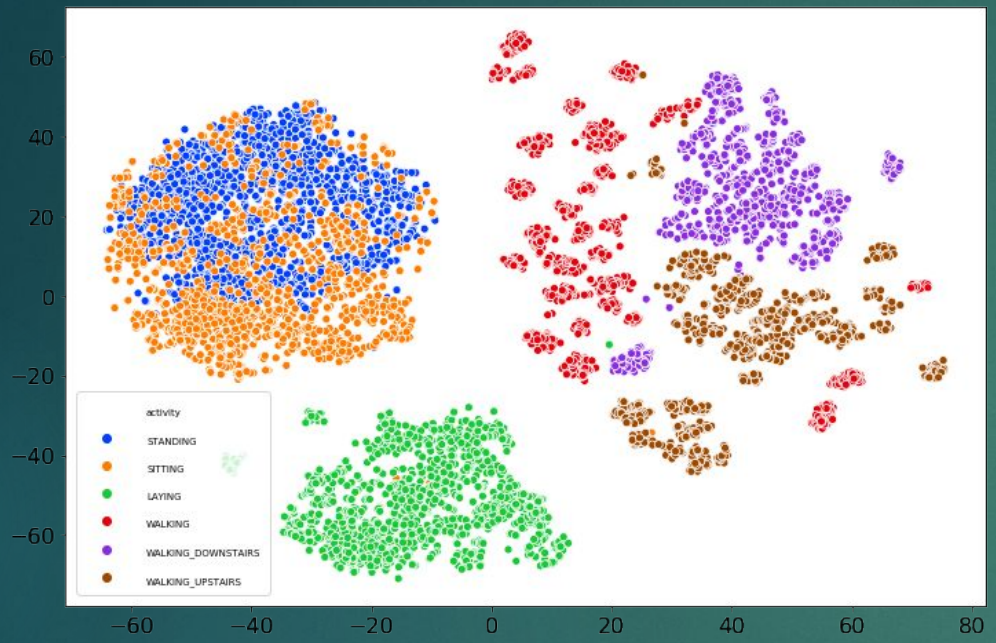
+

	id
0	5
1	5
2	5
3	5
4	5

=

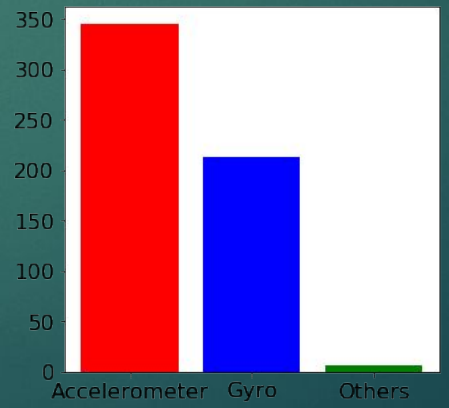
	1 tBodyAcc-mean()-X	2 tBodyAcc-mean()-Y	3 tBodyAcc-mean()-Z	4 tBodyAcc-std()-X	5 tBodyAcc-std()-Y	6 tBodyAcc-std()-Z	7 tBodyAcc-mad()-X	8 tBodyAcc-mad()-Y	9 tBodyAcc-mad()-Z	10 tBodyAcc-max()-X	...	555 angle(tBodyAccMean,gravity)
0	0.288585	-0.020294	-0.132905	-0.995279	-0.983111	-0.913526	-0.995112	-0.983185	-0.923527	-0.934724	...	-0.112754
1	0.278419	-0.016411	-0.123520	-0.998245	-0.975300	-0.960322	-0.998807	-0.974914	-0.957686	-0.943068	...	0.053477
2	0.279653	-0.019467	-0.113462	-0.995380	-0.967187	-0.978944	-0.996520	-0.963668	-0.977469	-0.938692	...	-0.118559
3	0.279174	-0.026201	-0.123283	-0.996091	-0.983403	-0.990675	-0.997099	-0.982750	-0.989302	-0.938692	...	-0.036788
4	0.276629	-0.016570	-0.115362	-0.998139	-0.980817	-0.990482	-0.998321	-0.979672	-0.990441	-0.942469	...	0.123320

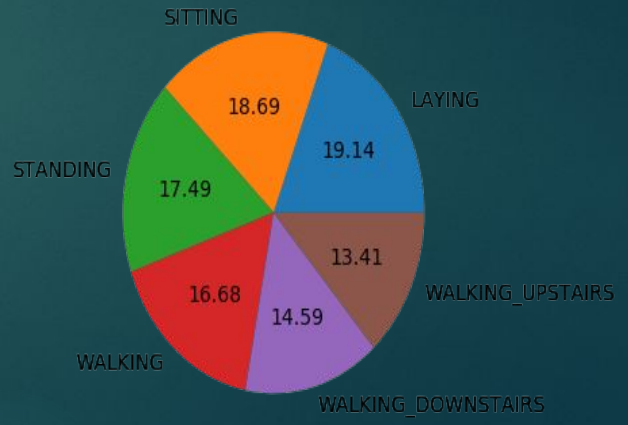
EDA



Tried PCA to with two components for visualization but wasn't of much help

Used The t-Distributed Stochastic Neighbour Embedding (t-SNE) which is a non-linear technique for dimensionality reduction that is suited for the visualization of high-dimensional datasets.





Algorithms Used

- K-Means
- SVM
- Hierarchical Clustering
- GMM
- Spectral Clustering

Since we knew the number of clusters beforehand, did not use LOF or DBSCAN

Dimensionality Reductions

- PCA
- LDA
- Varclus

K Means

1.Train and test instance size - Clustering approach, hence not splitting into test and train

2.Columns used and no. of models - All features being used. Since K means is not impacted by the presence of correlated variables and is also quite faster, I'm using the features as such with normalized values

3.Parameters used - Running 'k-means++' for initial cluster centers with 10 iterations for starting with different centroid seeds and maximum iterations up to 300 per run

4.Using metric adjusted score - We see that the K means adjusted rand score is 0.42 approximately, which means K means is performing poorly here

SVM

1.Train and test instance size - Splitting the train and test into 70-30 respectively

2.Columns used and no. of models - All features being used in first case where we have the maximum test accuracy, Using PCA and LDA for dimensionality reduction and in third case tried the Varclus approach to extract the most important variables and worked with them

3.Parameters used - SVC(C=1000, cache_size=200, class_weight=None, coef0=0.0,

decision_function_shape='ovr', degree=3, gamma=0.001, kernel='rbf',

max_iter=-1, probability=False, random_state=None, shrinking=True,

tol=0.001, verbose=False)

This was obtained from Grid Search Method.

4.Accuracy:0.982321 with all features,0.959655 with PCA, 0.981414 with LDA and 0.947 with Varclus

Hierarchical Clustering

1.Train and test instance size - Clustering approach, hence not splitting into test and train

2.Columns used and no. of models - in first case, tried the Varclus approach to extract the most important variables and worked with them which doesn't give satisfactory results. In second approach, Using PCA and LDA for dimensionality reduction. PCA as such doesn't improve the score but LDA on the other hand performs a lot better.

3.Parameters used - n_clusters=6, affinity='euclidean', linkage='ward'

4.Adjusted_rand_score:0.30 with Varclus,0.48 with PCA, 0.96 with LDA.

Gaussian Mixture Models

1.Train and test instance size - Clustering approach, hence not splitting into test and train

2.Columns used and no. of models - Using PCA and LDA for dimensionality reduction. PCA as such doesn't improve the score but LDA on the other hand performs a lot better.

3.Parameters used -n_components=6

4.Adjusted_rand_score:0.44 with PCA, 0.957 with LDA.