EVALUATION OF THE ALGORITHM

Since true labels are missing, we can only use the model itself to evaluate performance according to http://scikit-learn.org/stable/modules/clustering.html#clustering-performance-evaluation

First dataset was the evaluation dataset of Turkish students

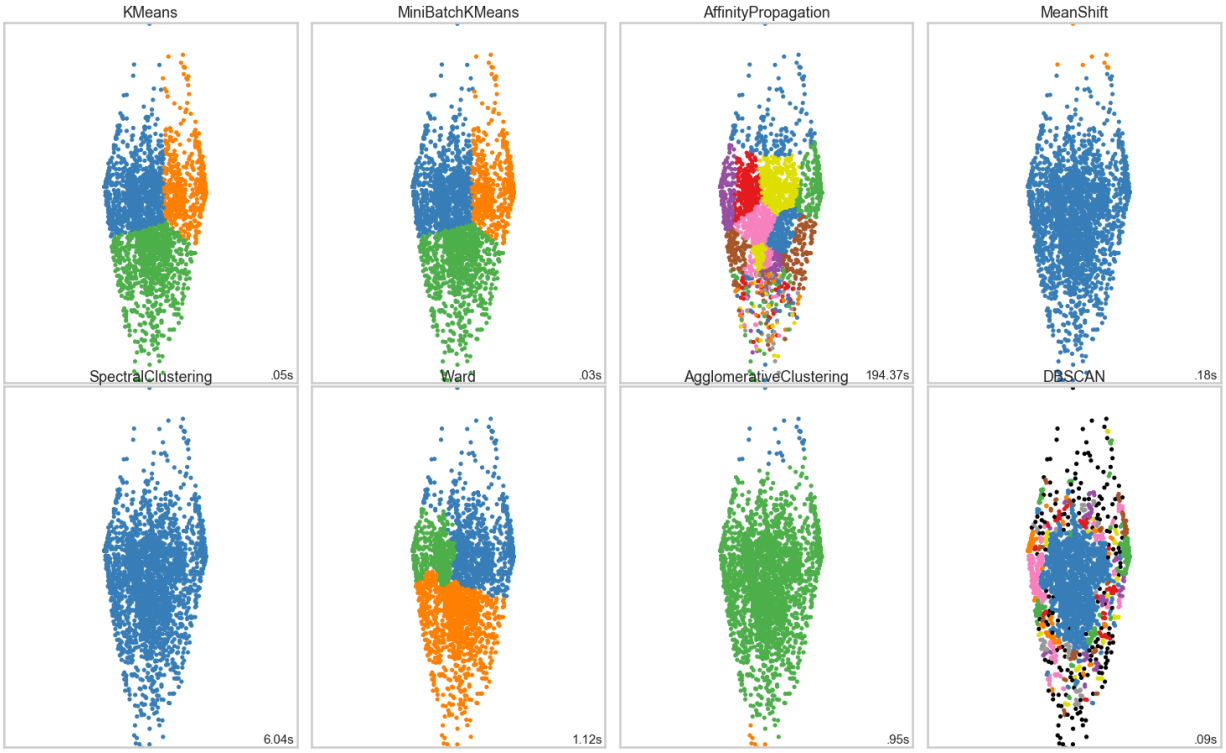
Second data was the set of power consumption with 19 73788 records I wasn't able to process the dataset for clustering due to memory error however I was able to run some of the algorithms on it.

SILHOUETTE SCORE		
	DATASET 1 (DS4)	**DATASET 2(DS5)**
KMEANS	0.573242	
BATCHKMEANS	0.573243	
MEANSHIFT	0.668646	
AFFINITY PROPAGATION	0.369718	
AGGLOMERATIVE CLUSTERING	0.488551	
SPECTRAL CLUSTERING	0497212	
DBSCAN	0.532586	

The range of silhouette score is between -1 and 1 where -1 shows improper clusters. Greater then value of silhouette score the better-defined cluster we get, here except for affinity propagation all of the algorithms are performing well. In addition, affinity propagation is slower as compared to other algorithms and results in many clusters.
Mean Shift performs really well and I think it's the best algorithm for Dataset 1 as its silhouette score is the highest also it is relatively faster than other algorithms it also does not require any number of clusters to begin.

Time to converge		
	DATASET 1 (DS4)	**DATASET 2(DS5)**
KMEANS	0.05s	0.57s
BATCHKMEANS	0.03s	
MEANSHIFT	0.18s	
AFFINITY PROPAGATION	194.37s	
AGGLOMERATIVE CLUSTERING	0.95s	
SPECTRAL CLUSTERING	6.04s	
DBSCAN	0.09s	

FOR DATA SET 1



COMPARISION OF CLUSTERING ALGORITHMS:

KMEANS: It is a fast algorithm and is scalable for large datasets, however it requires number of clusters initially which is a problem because initial number of cluster assignment manually can result in biased results. It also starts with a random choice of centroids which results in different clustering results each time.

MiniBatch KMeans is same as kmeans except that it converges way faster than KMeans however it has the same drawbacks.

Mean Shift is a very desirable technique as it doesnot require number of clusters initially also it moves towards denser regions which is a desirable quality. It can be a problem selecting the radius of the sliding window.

DBSCAN: It is a very desirable algorithm which doesnot requires intial cluster number also it is the only algorithm which figures out outliers and labels them. It performs well with different sizes of clusters. However it doesn't perform well with cluster of different densities which can be a very tricky thing as values of epsilon and neighbours are same for all density clusters in a given dataset.

The Best Algorithm so far is mean shift as it gives better results, its fast enough and it has only one problem to tackle that is off radius which can be iteratively decided.