Project - Handwritten Image Recognition

The project aims to implement an unsupervised algorithm to group similar images together. The project implements the k-means algorithm. The project also has an implementation of PCA.

Dataset contain 654 images of size 28*28 pixels.

We know that these images belong to 4 different classes (0-3). We want to see with the K-means algorithm how many clusters are formed and what is the composition of these clusters. Thereby analyze whether these clusters have any resemblance with the actual digits that they represent.

Approach to solve this problem of handwritten digit recognition can be broadly divided into

1. Data Preprocessing.
    a. Read Image and Pixel Conversion
    b. Standardization of data
2. Feature extraction using PCA.
3. Kmeans Algorithm.
    a. Execution steps of algorithm.
4. Cluster Evaluation
    a. Elbow Method and Cluster formation analysis
    b. Silhouette Method.
5. Conclusion.
6. Attachments.
7. References.

1. Data Preprocessing

a. Read image and Pixel Conversion

Python's OS module is used to traverse through the directory and sub-directory to read the image files. Once the files are read from respective sub-folders they need to be converted into a format that can be written in the data frame. Therefore image files are first converted into numpy.ndarray of pixel values, a multidimensional array of the same size as that of the image (28*28). Thereafter multidimensional array is flattened into series, ready to be populated into the data frame

The data frame is this way is populated with rows having 784 columns with each row representing an image. The indexing happens in the order in which the files are read. As these images are being read, a corresponding list of the labels is populated. The label is nothing but the name of the sub-directory in which that image is present. The idea behind doing this exercise is to understand how well the clusters are formed concerning these labels i.e. to say differences between the images or variations of images and the corresponding number of clusters. In this project, however we will be only implementing unsupervised K-means algorithm on the pixel data.

2. Feature extraction using PCA

PCA is a dimensionality reduction algorithm, used for both supervised and unsupervised algorithms. Dimensionality reduction concept is transforming existing features to a meaningful reduced number of features. PCA achieves this by using a combination of the original feature that will give the largest variance, that is, it will cover the maximum variation in the data of the dataset. The first few sets of components explain the maximum variance, based on which the remaining can be discarded as they do not account for a lot of variability in the data.

PCA helps here to reduce the processing time which otherwise would be high to calculate the Euclidean distance between 2 data points.

PCA helps us to visualize the data in 2d or 3d space (by looking at the top 2 or 3 components) and can help us in this project to see the cluster formation

3. K-means algorithm

Clustering is the task of grouping similar images.

'k' in k-means refers to the number of clusters and *'means'* in the K-means refers to averaging of the data; that is, finding the centroid of the clusters.

The principle of the k-means algorithm lies in the initialization of centroid. The algorithm has different ways to initialize centroid.

a. Execution of algorithm

Step for execution are as below

1. Python's scikit has an optimal way of choosing the initial centroid. It uses the default (k-means++) to select initial cluster centers for k-mean clustering in a smart way to speed up convergence.
2. The distance of the data point is calculated with every centroid. A label is assigned to the data point based on the nearest distance. The Euclidean method is used to calculate distance.
3. Once labeling is done, recalculate centroid based on the new cluster assigned.
4. Repeat steps 2 and 3 until a given data point consistently converges to a given label as in the previous iteration.

Above steps are executed with range of k values (2-23) to evaluate number of cluster required for data points to converge in a given cluster homogeneously.

4. Cluster Evaluation

In unsupervised learning, we use clusters to group similar objects as a way of classifying the data points.

We know in this project the class or the label associated with every image. Therefore the project aims to find out how many clusters will be formed and the composition of each cluster.

A fundamental step for every unsupervised algorithm is to determine an optimal number of clusters. Elbow and Silhouette methods are one of the most primitive ways of determining the number of clusters.

Elbow method is based on the principle of the sum of the squared distances of the data points to their respective cluster centroids. This is also called as inertia. A graph is plotted with inertia values and k values.

Whenever we plot the k values against the inertia we generally should see downward trend with increasing k value. The reason for this is, as the k value increases, the clusters become more. As the number of clusters increase, the data points keep getting more concentrated in smaller clusters thereby increasing the chances of forming clusters that are more compact.

However, at a certain value of k we generally observe that the reduction in inertia is marginal which means the higher k doesn't help in drastically differentiating the data points. This value of k is chosen as the optimal number of clusters and is known as the elbow point.

The calculation approach of this method may or may not be reliable as it does not take into consideration the distance of data points from the cluster centroid it belongs and neighboring clusters i.e. how closely it is matched to data points within its cluster and neighboring clusters.

On the other hand, in Silhouette Method, Silhouette analysis measures how close each point in one cluster is to points in the neighboring cluster. The Silhouette Coefficient is calculated using (a) the mean intra-cluster distance and (b) the mean nearest-cluster distance for each sample. The Euclidean distance is used to calculate the distance between the data points.

For e.g. for k=3, consider cluster 1, cluster 2, cluster 3.

Take a data point ***i*** belonging to cluster 1. Calculate the distance of this point with all the data points belonging to the same cluster. Get the average distance to say **a**

For the same data point ***i*** of cluster 1, now calculate the distance of this data points with all data points of cluster 2, get an average distance to say **b**.

Same way for cluster 3, let say **c** is the average distance.

Now check whether cluster 2 or cluster 3 is at a minimum distance from data point **i** (cluster 1) ie whether **c>b or b>c .Whichever is minimum will be used in Silhouette formula to calculate the coefficient for cluster 1**

Silhouette coefficient= 1-(a/b)

The same process is followed to calculate the coefficient for cluster 2 and cluster 3. From this, cluster with the highest coefficient will be considered for k=3

The measure has a range of [-1, 1].

The measure of the Silhouette coefficient is in the range of [-1, 1]. As seen in the above example if the average distance of **a** is higher than the minimum of **b and c.** The Silhouette coefficient of the cluster 1 will be close to +1. This indicates the data points of cluster 1 are far away from neighboring clusters and very close to cluster it belongs to.

Similarly if the average distance of **a** is lower than a minimum of b and c, the Silhouette coefficient of cluster 1 will be close to -1 thereby indicating that data points of cluster 1 are more close to neighboring clusters and far away from the cluster it belongs to.

Similarly, if the average distance of **a and minimum of b and c results in 0** it indicates that the data point could belong to any cluster

For deeper analysis, we have automated the task to get a count of images for each class in a given cluster for a given k value. This was repeated for a range of PCA variance (0.96 -0.99). Details of this are provided in the excel file (k-elbow.xlsx) attached below. Refer sheets PCA-0.96, PCA-0.97, PCA-0.98; PCA-0.99.This analysis was to determine the best k value with maximum homogeneity.

a. Elbow Method and Cluster formation analysis

In this implementation, we have tried to find the optimal number of clusters using the elbow method. We ran the algorithm on different versions of the transformed data – ranging from 96% variation in the data to 99% variation in the data. This was followed by an in-depth analysis of how these clusters compared to the actual digits they were representing. Results are described below.

PCA - 0.96

As seen in the excel file attached (sheet =PCA – 0.96) algorithm was executed with a range of k-values (2-23). For each k-value, the count of images for a given class is populated.

Over here (attached file sheet-PCA-0.96) for 0.96 we observe that the elbow point could be either k-16 or k-18.

Also, the difference between the inertia of adjacent k values was calculated (sheet 0.96-Elbow), it was found that minimum distance was for k-18.

A further analysis was done. It was observed from the sheet (PCA-0.96, cell marked in green). For k-16 we see many clusters that have a majority of 1 digit. A similar thing is seen in k-18. However, in k-18 there is more spread of all digits

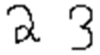
across the cluster indicating the variations of the images are well understood in comparison to k-16.

Similar kind of analysis was done with other k-values.

From above all observation k-18 is chosen as the optimal number of clusters and this is known elbow point ie a value higher than this will drastically in differentiating the data points.

The ideal outcome of this exercise was to have 4 clusters each representing one of the digits. However, we see that k-18 has the most number of clusters dominated by 1 digit. Therefore a high number of clusters can be due to variation in images.

The way digits are handwritten would resemble other digits in the dataset. Eg digit 2 with a lesser curve at the upper end would read like a 1 by the model. Similarly, digit 3 with extended curves towards the middle may also be read as 0 by the model. For e.g. for k-18 cluster 0, cluster 8 both are dominated by digit 3 but they still have few counts of digit 2, shows that digit would have been written in a way to resemble digit 2.

 Cluster 0 of k-18

 Cluster 1 of k-18.Here cluster is dominated by digit 2 but still has few counts of digit 1.

Therefore the observation indicates how the clusters are composed. How well the variations of the images in the data set are handled with an increase in the number of clusters in k-18.

However, on checking the n_iter values, it was observed that they are way below the default values of 300. This indicates that a lesser number of iterations were required to converge the data, had the variation been very high the n_iter value would have also been high.

PCA -0.97

As seen in the excel file attached (sheet =PCA – 0.97) algorithm was run with range of k-values (2-23) .For each k-values clusters count of images for a given class is populated.

Similar analysis as PCA 0.96 was done.

It was observed that elbow point could be k-12, k-16, k-18 has elbow points. However, there was a spike in elbow point for k-13 which can be due to an error.

The difference between the inertia of adjacent k values was calculated (sheet 0.97-Elbow), it was found that minimum distance was for k-18.

A further analysis was done. It was observed from the sheet (PCA-0.97, cell marked in green). For k-16 we see many clusters that have a majority of 1 digit. A similar thing is seen in k-18. However, in k-18 there is more spread of all digits across the cluster indicating the variations of the images are well understood in comparison to k-16.

Therefor k-18 was chosen for this PCA variance.

PCA – 0.98

As seen in the excel file attached (sheet =PCA – 0.98) algorithm was run with range of k-values (2-23) .For each k-values clusters count of images for a given class is populated.

The difference between the inertia of adjacent k values was calculated (sheet 0.98-Elbow), it was found that minimum distance was for k-18.

Similar analysis as PCA 0.96 was done and it was found that k-16 was good for this PCA variance.

PCA – 0.99

As seen in the excel file attached (sheet =PCA – 0.99) algorithm was run with range of k-values (2-23) .For each k-values clusters count of images for a given class is populated.

Similar analysis as PCA 0.96 was done and it was found that k-16 was good for this variance.

Observation

We have seen above that each PCA variance gives an optimal k value. However, we need to go ahead with the best PCA variance and respective k value. Based on the above observation we conclude PCA variance of 0.96 and the corresponding k-18 is the best and optimal choice for this project. The inertia

difference for this is also lowest when compared to other PCA variance. This shows the variation in images is well learned. This also signifies that k-18 allows the digits to spread well across all clusters thereby handling the variation in the data set.

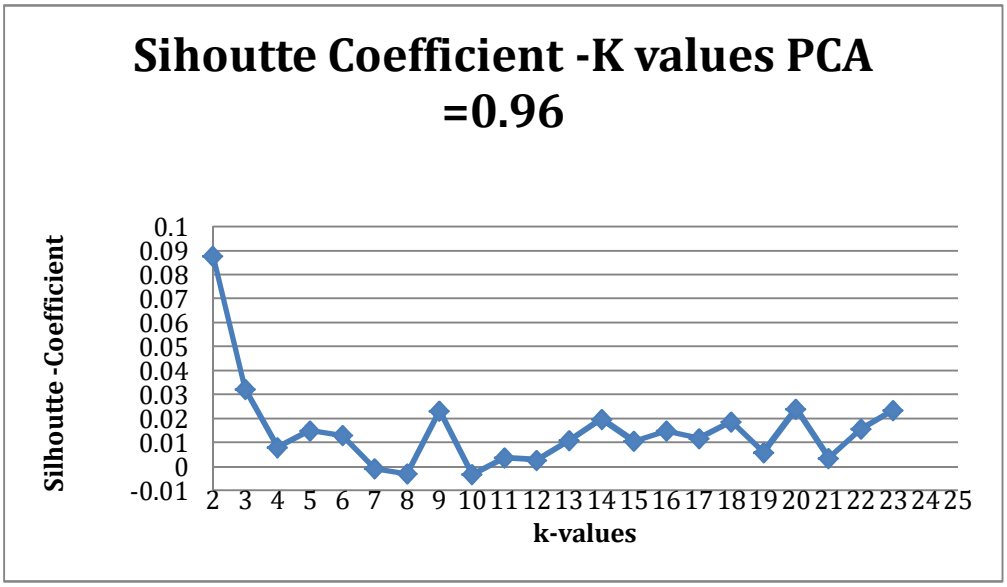
b. Silhouette Method and Cluster formation Analysis.

In this, we have implemented the Silhouette Coefficient method to find the best k value for the given data set. The silhouette shows which object lies well within their clusters. The silhouette value close to 1 implies within-cluster dissimilarity is smaller than between cluster dissimilarity. The silhouette value equal to 0 implies overlapping clusters. The silhouette value close to -1 implies the sample is misclassified. Following this approach below are the observations for k values for each of the PCA variance.

PCA - 0.96

A graph was plotted with k values and Silhouette coefficient (attached excel file, sheet-Silhouette-Coeff), we can see in the graph below k-20 is with 0.024 coefficient as 2nd highest coefficient after k-2.

Further analysis for each of the k values was done with the help of data in the excel sheet. Below is the observation.



K=2:- As we see the average Silhouette coefficient for the clusters with a k value of 2 is 0.87. This shows that within the cluster the dissimilarity is very small and

they appear to be well clustered. But as we know from ground truth that images belong to 4 classes (0-3). This also shows that the natural clusters are fused into 2 clusters. Keeping this into account we cannot go ahead with k-2 as at least 4 clusters will be required. .Therefore not going by the face values of the silhouette coefficient we further analyze other k-values.

K=3: As seen Silhouette coefficient has drastically gone down near to 0.032. On verifying the reason in the attached excel file (sheet PCA-0.96, k-2, and k-3 data), we can see that the clusters are not stable and not well converged. The same images are split across multiple clusters. This low coefficient can be the reason also of misclassification of images, an indication that within cluster dissimilarity and between cluster dissimilarity will be higher.

K=4:- The Silhouette coefficient is 0.008, a very low score, indicating that clusters are still not stable and not well converged. This shows that the algorithm still hasn't found the natural clusters. This can thereby lead to higher within and between cluster dissimilarity. More clusters will be required to form natural clusters.

K =5 and 6:- The Silhouette coefficient values are 0.014 and 0.013 respectively. Further analysis was done with the help of attached excel file(sheet PCA-0.96), we can see that both k values have a pure cluster dominated by only one type of digit and multiple clusters which has the majority of 1 digit. This can be an indication of the algorithm forming more natural clusters gradually if we decide to increase the value of k

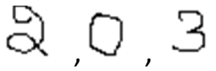
K =7 and 8:- The Silhouette coefficient values are -0008 and -0.0003. A negative value is an indication of the misclassification of images. Thereby signifying that the average distance to points in the clusters is very high compared to between the cluster distances. To correlate this we can see in the attached excel file (sheet PCA-0.96), k values 5 and 6 which had clusters dominated by majority one type of digit are now split into different clusters in k 7 and 8. This can be the reason for the high misclassification and negative coefficient.

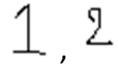
K values beyond k-9, we can see from the sheet(PCA-0.96), the clusters that were dominating with majority digit in k-9 have further split and spread across multiple clusters also with lower count values in each of the clusters. Despite this, we can see that all digits are very well classified across the cluster and dominated by one digit. This also implies that the dataset has a high variety of data. Since the digits are handwritten they would resemble another digit. Eg digit

2 with a lesser curve at the upper end would read like a 1 by the model. Similarly, digit 3 with extended curves towards the middle may also be read as 0 by the model. This can result in having more clusters to accommodate the variations.

Though above k -9 we see that clusters dominated with majority 1 digit are further split across multiple clusters with lower count but they are stabilized for k-20 i.e. we see that from the excel sheet (PCA-0.96) that the clusters are now dominated back with majority of 1 digit and with higher count values in each of the clusters.

For e.g.: in k-9 cluster 0 has (digit 0=2,2=13,3=56) but none are dominating the cluster with majority value. But cluster 0 in k-20 has a different composition (digit 0=25,2=5,3=9), here digit 0 is dominating the cluster also the count of other digits has decreased indicating the cluster is stabilizing with variation well spread.

 Cluster 0 of k-18.Here the way the digit 2 and 3 are written resemble digit 0.The extended curve of 2 at the top and the oval formation at the end would make the model read 2 as 0.Similary for 3,the upper extend curve moving towards middle portion of 3 would read like forming a 0.

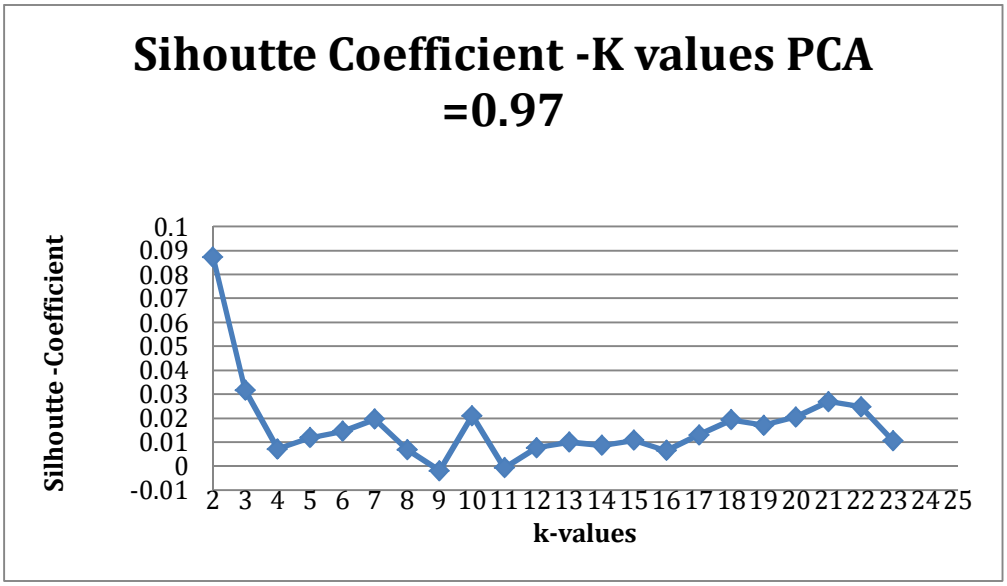
 Cluster 14 of k-20 .Here digit 1 dominates the cluster having few count of digit 2 also. The upper curve extends like a line for digit 2, this would read like a 1 by the model.

The above observations indicate that with the increase in the number of clusters the variations are well handled and spread. The same can be understood from the Silhouette coefficient's approach of how images closely match the images of the cluster it belongs to then the neighboring clusters. This is why k-20 has the highest coefficient and the intermediate k values have either a lower or negative coefficient.

Therefore k-20 is the best choice for this variance.

PCA - 0.97

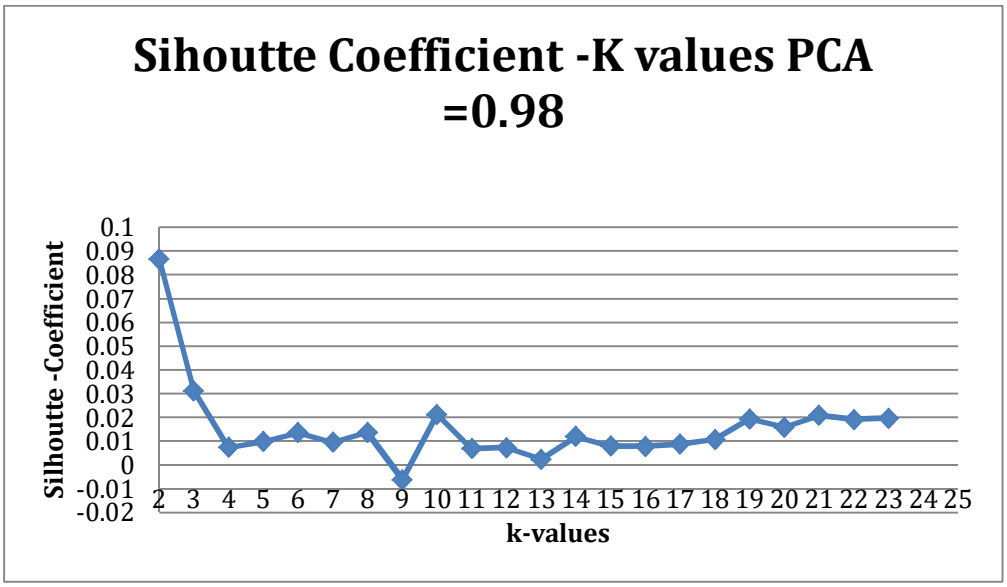
A graph was plotted with k values and Silhouette coefficient (attached excel file, sheet-Silhouette-Coeff), we can see in the graph below k-21 is with 0.027 coefficient as 2nd highest coefficient after k-2.



A similar analysis was done for this variance as PCA 0.96. A Silhouette coefficient of 0.27 for k-21 is the best choice for this variance.

PCA – 0.98

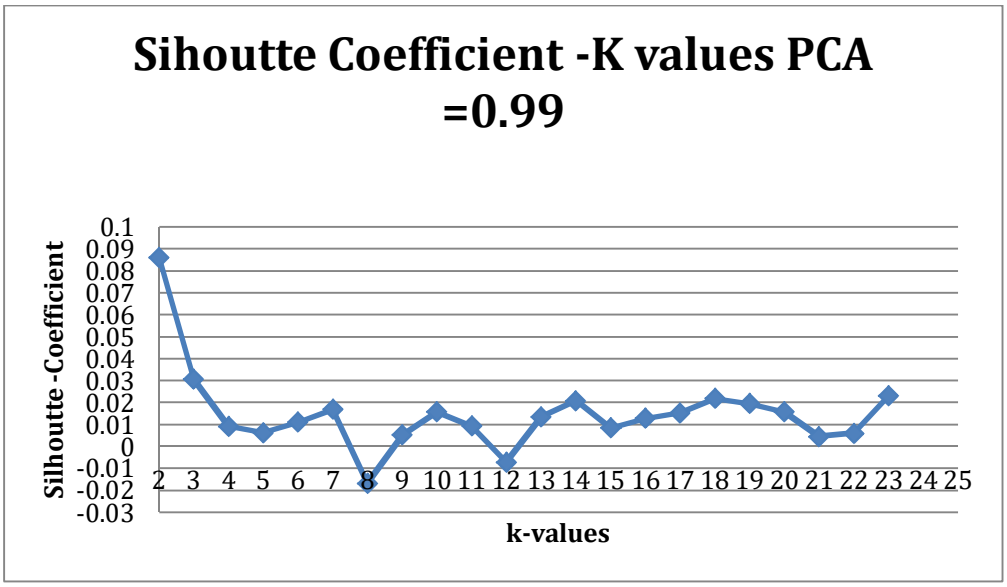
A graph was plotted with k values and Silhouette coefficient (attached excel file, sheet-Silhouette-Coeff), we can see in the graph below k-10 is with 0.021 coefficient as 2nd highest coefficient after k-2.



A similar analysis was done for this variance as PCA 0.96. A Silhouette coefficient of 0.21 for k-10 is the best choice for this variance.

PCA - 0.99

A graph was plotted with k values and Silhouette coefficient (attached excel file, sheet-Silhouette-Coeff), we can see in the graph below k-14 is with 0.020 coefficient as 2nd highest coefficient after k-2.



A similar analysis was done for this variance as PCA 0.96. A Silhouette coefficient of 0.20 for k-14 is the best choice for this variance.

Observation

As seen from the above observation each of the variances has its best choice of K value with the highest Silhouette coefficient. The best choice is with PCA 0.97 variance and k-21. The Silhouette coefficient for this k value is highest for this variance in comparison to other PCA variances. With k-21 and PCA 0.97 variance, the images are well classified among the clusters and dominated by 1 digit thereby able to handle the variation of images for each of the digits in the dataset.

5. Conclusion

We have seen above that Elbow and Silhouette methods differ in their best k values. Since the Elbow method may or may not be reliable when it comes to a number based on the calculation approach, therefore the project will go ahead with the Silhouette coefficient method. The project will consider PCA variance 0.97, with k-21 as an optimal value of k for the given dataset.

6. Attachment's

Handwritten-Kmean.xlsx

7. References

https://www.sciencedirect.com/science/article/pii/0377042787901257

https://stackoverflow.com/questions/23387275/how-do-you-manually-compute-for-silhouette-cohesion-and-separation-of-cluster