# K-Means Clustering on Image Dataset

Parakh Gupta
Machine Learning Engineer Intern
AI-Tech Systems
ai-techsystems.com
Delhi,India
parakhgupta.98@gmail.com

*Abstract*— K-means clustering is a type of unsupervised learning, which is used when you have unlabeled data. The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K here K=10. The dataset is the fruits dataset that is taken from Kaggle. The dataset consists of large amount of fruit images. The images of the dataset are clustered into 10 clusters. As the dimensions of data increases, the difficulty to visualize it and perform computations on it also increases so to reduce the dimensions of the data, Principal Component Analysis (PCA) is used.

***Keywords— Machine Learning, Unsupervised Learning, PCA, K-Means***

## I. INTRODUCTION

Clustering is an interesting field of Unsupervised Machine learning where we classify datasets into set of similar groups. K-means clustering is one of the simplest and popular unsupervised machine learning algorithms. Typically, unsupervised algorithms make inferences from datasets using only input vectors without referring to known, or labelled, outcomes. The key assumption behind the clustering algorithm is that nearby points in the feature space, possess similar qualities and they can be clustered together.

K-means is the technique for clustering similar data in one cluster. K means is an unsupervised learning technique in which the data is unlabelled. In this technique data is clustered together based on the similarity between them. In this technique each datapoint is assigned to a particular cluster. The data point is assigned to that cluster to which its properties match significantly. Euclidean distance is used as the measure for assigning a data point to the cluster. K-means technique can be used to provide label to the unlabelled data by labelling the data point corresponding to the cluster number it belongs to.

We will be doing a clustering on images. Images are also same as datapoints in regular Machine Learning and can considered as similar issue. In this article we will be having a set of images of fruits. We will try to cluster them into 10 different clusters. For this purpose, we can extract image features from a pretrained keras model like VGG16. Once we have the vectors, we use PCA to reduce the dimensions of the data and then apply K-Means clustering over the datapoints. So, here are some the pictures in my dataset, having around 114 images of fruits.

## II. DATA PREPROCESSING

This dataset from Kaggle consist of 57,276 images of fruits and these images are divided into 114 classes of fruits and each class contain approx. 400 images. Each image shape is 100x100 pixels with 3 colour channels.

### *A. Using a pre-trained model in Keras to extract the feature of a given image*

Let's consider VGG as our first model for feature extraction. VGG is a convolutional neural network model for image recognition proposed by the Visual Geometry Group in the University of Oxford, where VGG16 refers to a VGG model with 16 weight layers, and VGG19 refers to a VGG model with 19 weight layers. Fig. 2 illustrates the architecture of VGG16: the input layer takes an image in the size of (224 x 224 x 3), and the output layer is a softmax prediction on 1000 classes. From the input layer to the last max pooling layer (labeled by 7 x 7 x 512) is regarded as **the**

feature extraction part of the model, while the rest of the network is regarded as the classification part of the model. We have used only feature extraction part of the model and got feature of images in shape (57276, 7, 7, 512) and flatten the result to get (57276, 25088).

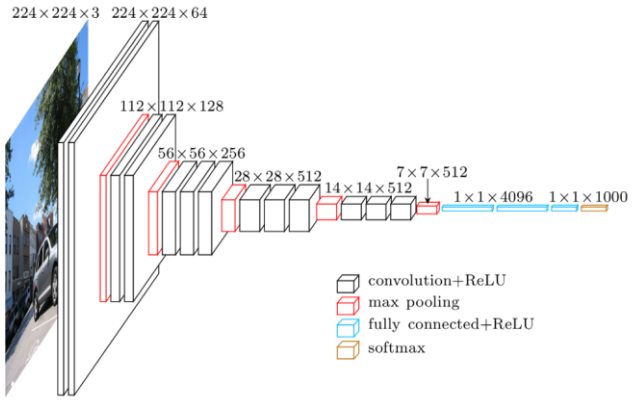


Fig. 2. VGG16 Architecture

### B. *Principal Component Analysis*

Principal Component Analysis, or PCA, is a dimensionality-reduction method that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information in the large set.

Reducing the number of variables of a data set naturally comes at the expense of accuracy, but the trick in dimensionality reduction is to trade a little accuracy for simplicity. Because smaller data sets are easier to explore and visualize and make analyzing data much easier and faster for machine learning algorithms without extraneous variables to process.

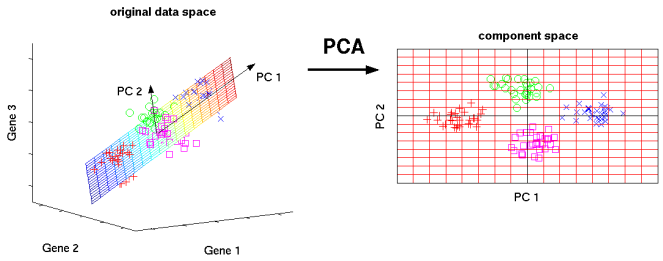


Fig. 3. PCA

The overall goal of PCA is to reduce the number of d dimensions (features) in a dataset by projecting it onto a k dimensional subspace where k < d. The approach used to complete PCA can be summarized as follows:

- Standardize the data.
- Use the standardized data to generate a covariance matrix (or perform Singular Vector Decomposition).
- Obtain eigenvectors (principal components) and eigenvalues from the covariance matrix. Each eigenvector will have a corresponding eigenvalue.
- Sort the eigenvalues in descending order.
- Select the *k* eigenvectors with the largest eigenvalues, where *k* is the number of dimensions used in the new feature space (k ≤ d).
- Construct a new matrix with the selected *k* eigenvectors.

## III. K-MEANS CLUSTERING ALGORITHM

Clustering is a method to divide a set of data into a specific number of groups. It's one of the popular method is k-means clustering. In k-means clustering, it partitions a collection of data into a k number group of data. It classifies a given set of data into k number of disjoint cluster. K-means algorithm consists of two separate phases. In the first phase it calculates the k centroid and in the second phase it takes each point to the cluster which has nearest centroid from the respective data point. There are different methods to define the distance of the nearest centroid and one of the most used methods is Euclidean distance. Once the grouping is done it recalculate the new centroid of each cluster and based on that centroid, a new Euclidean distance is calculated between each center and each data point and assigns the points in the cluster which have minimum Euclidean distance. Each cluster in the partition is defined by its member objects and by its centroid. The centroid for each cluster is the point to which the sum of distances from all the objects in that cluster is minimized. So K-means is an iterative algorithm in which it minimizes the sum of distances from each object to its cluster centroid, over all clusters.

### A. *Initialization*

The first thing k-means does, is randomly choose K examples (data points) from the dataset as initial centroids and that's simply because it does not know yet where the center of each cluster is.

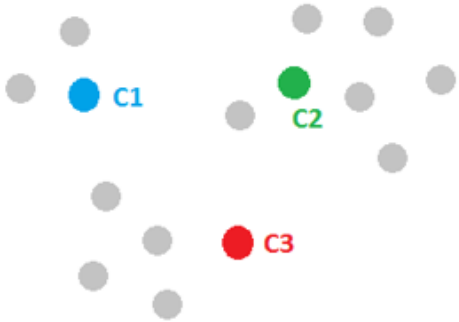


Fig. 4. k-means Step1 Initialization

### B. *Assign observations to the closest cluster center*

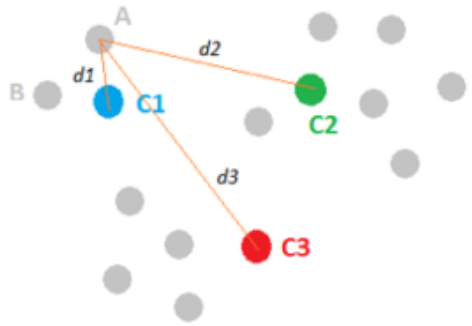


Fig. 5. Assign cluster

Then, all the data points that are the closest to a centroid will create a cluster. If we're using the Euclidean distance between data points and every centroid, a straight line is drawn between two centroids, then a boundary line divides this line into two clusters.

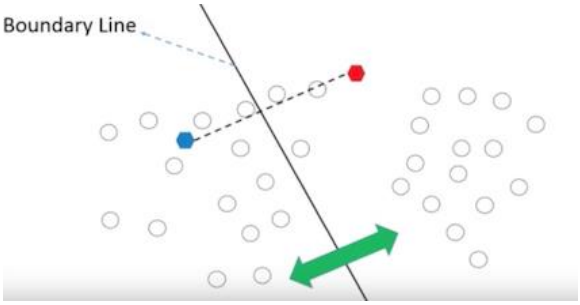


Fig. 6. Boundary line between clusters

### C. *Move the centroid*

Now, we have new clusters, that need centers. A centroid's new value is going to be the mean of all the examples in a cluster. Next, we need to update the cluster centers based on the points assigned to them. For instance, we can find the center mass of the blue cluster by summing over all the blue points and dividing by the total number of points, which is four here. And the resulted center mass C1', represented by a blue diamond, is our new center for the blue cluster. Similarly, we can find the new centers C2' and C3' for the green and red clusters.

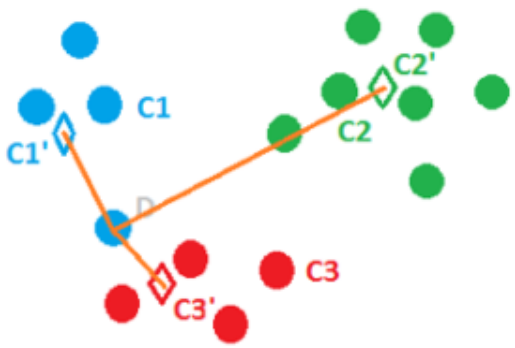


Fig. 7. Step3 Move the centroid

### D. *Repeat step 2 and step 3 until convergence*

We'll keep repeating step 2 and 3 until the centroids stop moving, in other words, K-means algorithm is converged.

## IV. Conclusion

Dimension is reduced to two using PCA and then k-means is applied to the reduced dataset. All the images are clustered in the 10 clusters on the basis of their shape and color. Each cluster contains images of fruit of specific shape only. Graph is plotted using two reduced features named PC1 and PC2. Black dot represents the centroid.

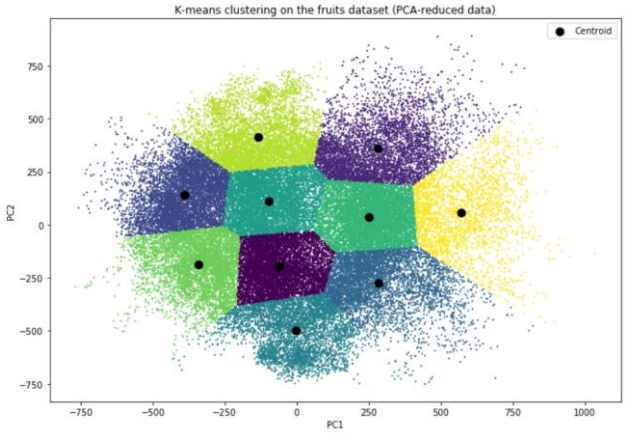


Fig. 8. Graph plot of k-means cluster

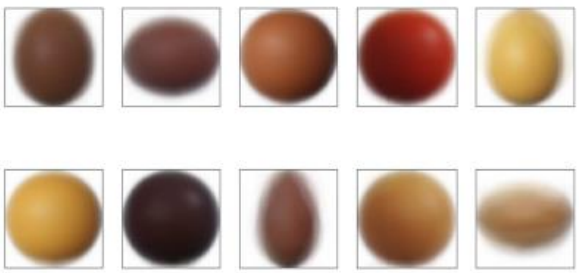

Fig. 9. Cluster Properties

## References


[1] Kaggle dataset :- https://www.kaggle.com/moltean/fruits

[2] https://jakevdp.github.io/PythonDataScienceHandbook/05.09-principal-component-analysis.html

[3] http://setosa.io/ev/principal-component-analysis/

[4] https://en.wikipedia.org/wiki/Principal_component_analysis

[5] https://medium.com/@franky07724_57962/using-keras-pre-trained-models-for-feature-extraction-in-image-clustering-a142c6cdf5b1

[6] https://towardsdatascience.com/image-clustering-using-transfer-learning-df5862779571