K-Means Clustering

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 simple objective of K-Means clustering is group similar data points together and discover underlying patterns. To achieve this objective, K-means looks for a fixed number (*k*) of clusters in a dataset.
A cluster refers to a collection of data points aggregated together because of certain similarities.

You'll define a target number *k*, which refers to the number of centroids you need in the dataset. A centroid is the imaginary or real location representing the center of the cluster.
Every data point is allocated to each of the clusters through reducing the in-cluster sum of squares.
In other words, the K-means algorithm identifies *k* number of centroids, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible.

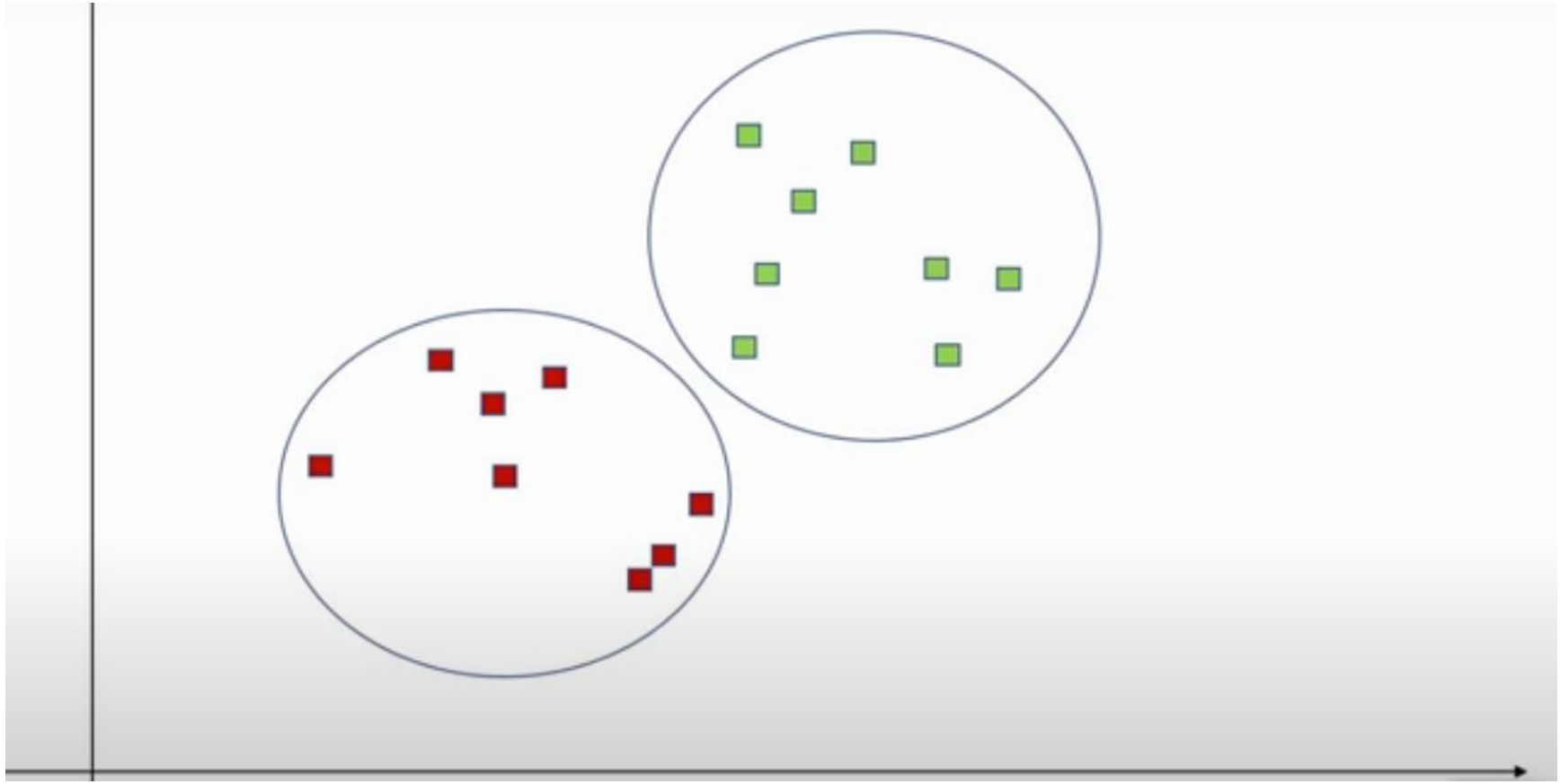
How K-Means works

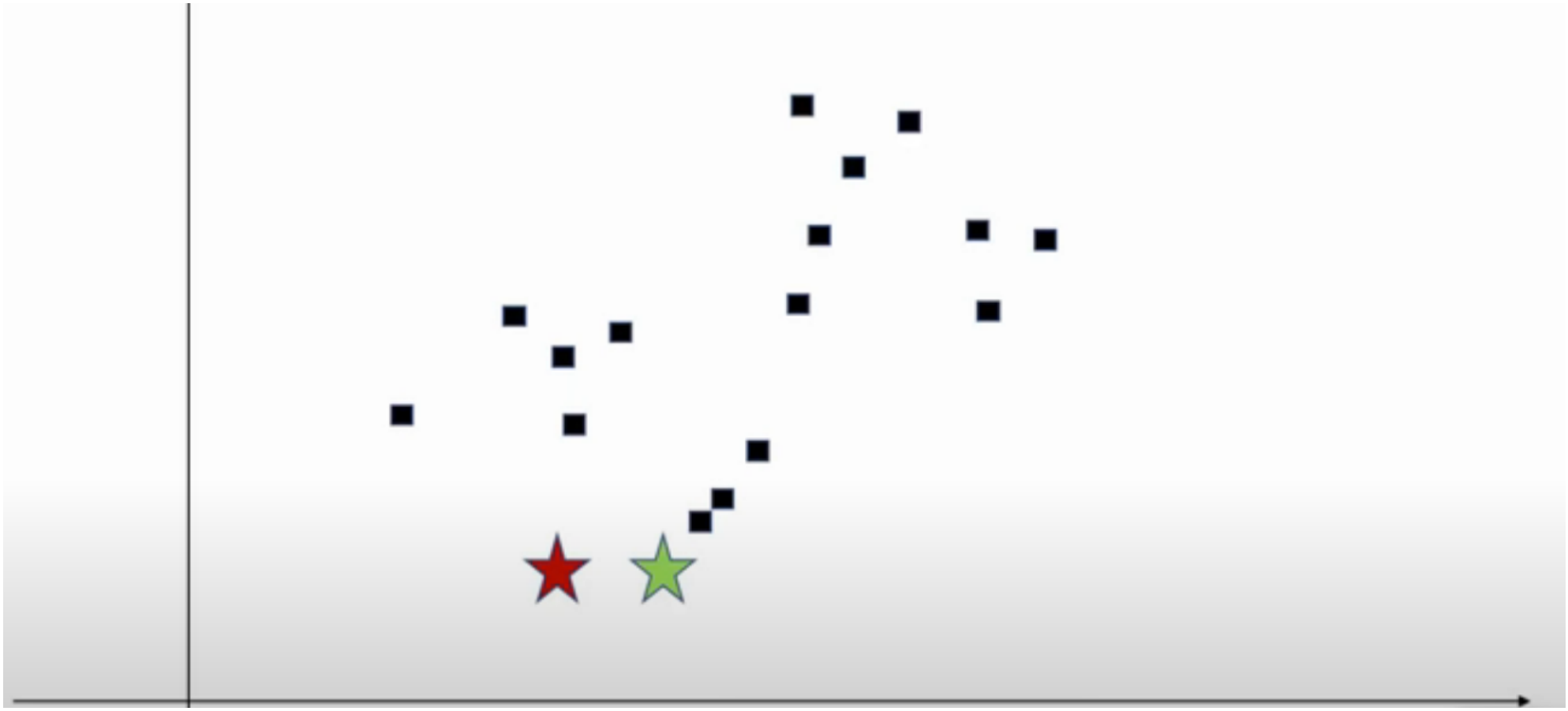
To process the learning data, the K-means algorithm in data mining starts with a first group of randomly selected centroids, which are used as the beginning points for every cluster, and then performs iterative (repetitive) calculations to optimize the positions of the centroids
It halts(paused) creating and optimizing clusters when either:

- The centroids have stabilized — there is no change in their values because the clustering has been successful.
- The defined number of iterations has been achieved.

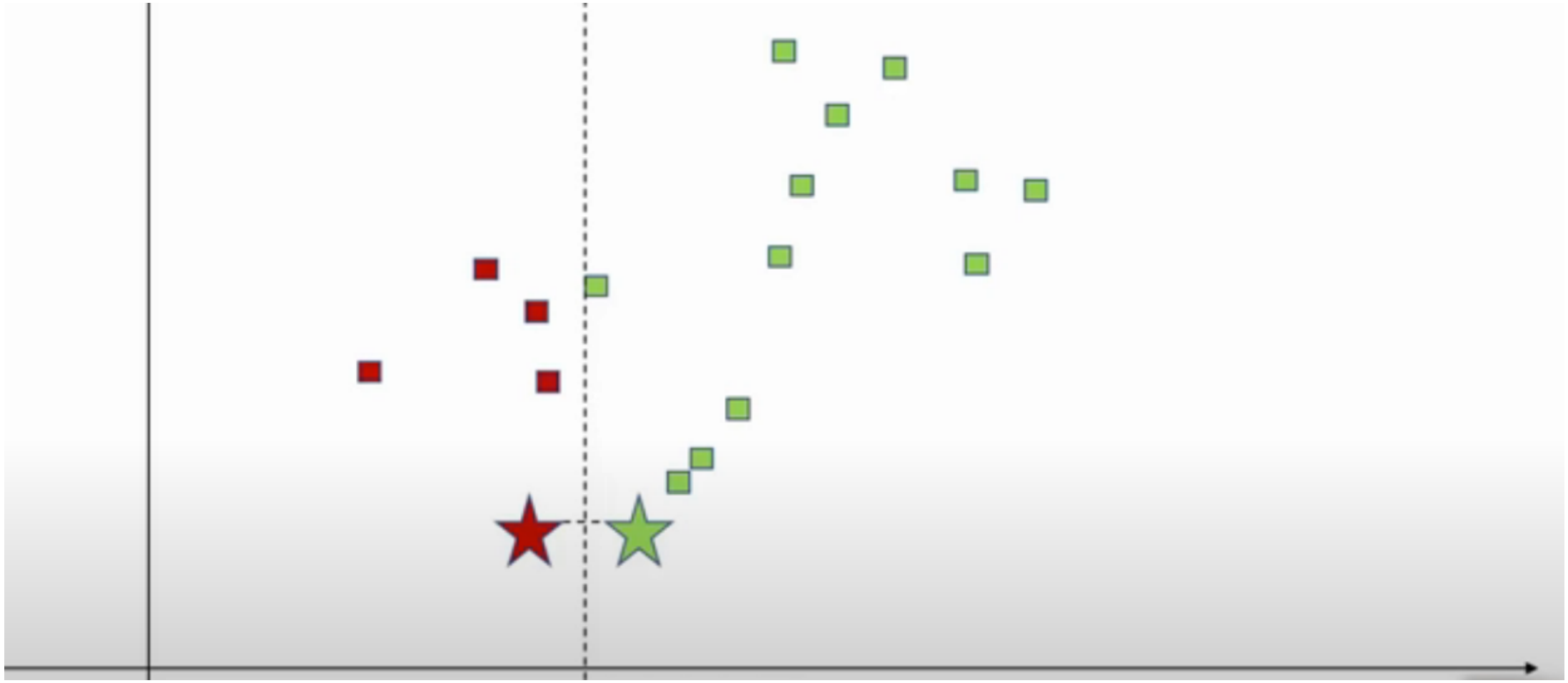
A basic data where visually the data can be seen as in 2 clusters but we can do this same with the help of K-Means clustering. In K-Means clustering K simple means an independent numerical option that is opened for a data scientist.

But there is a mathematical way to get the accurate number of K with respect to the data.
We will be seeing it in upcoming slides

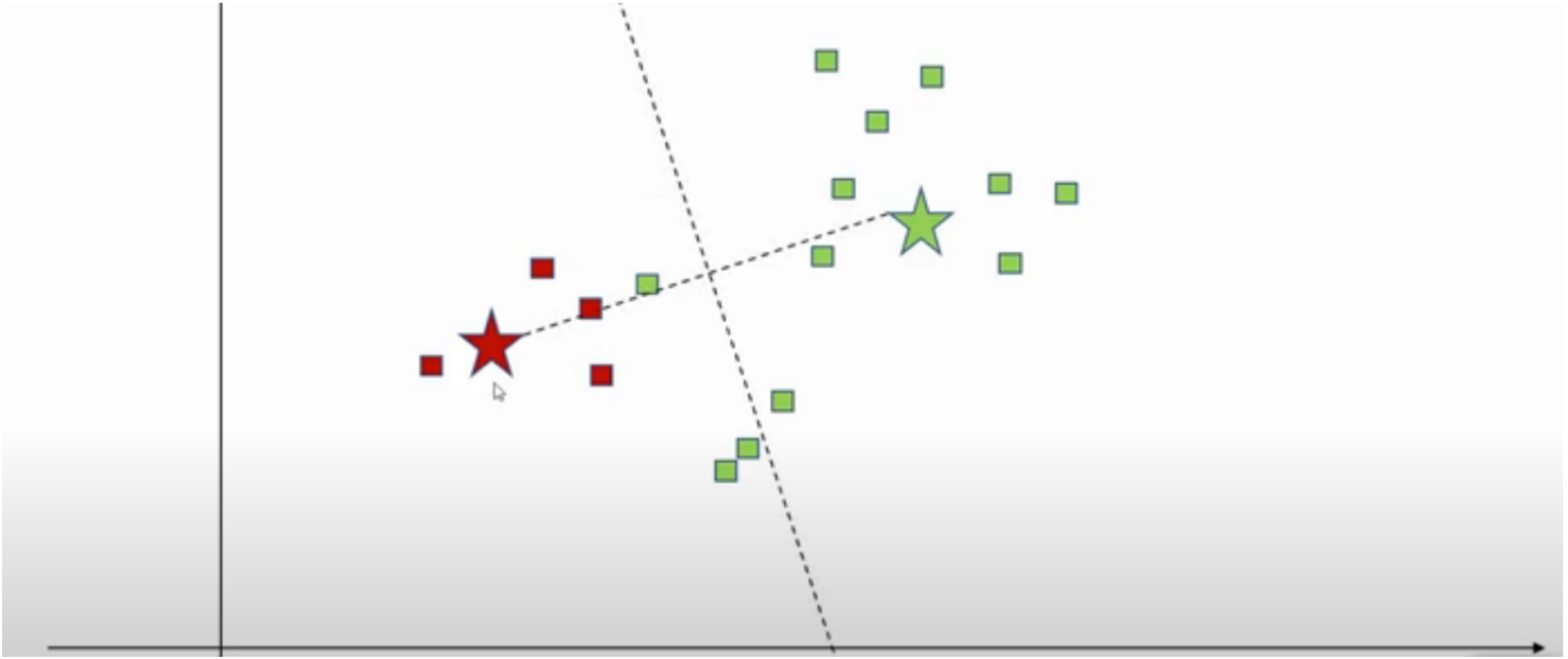




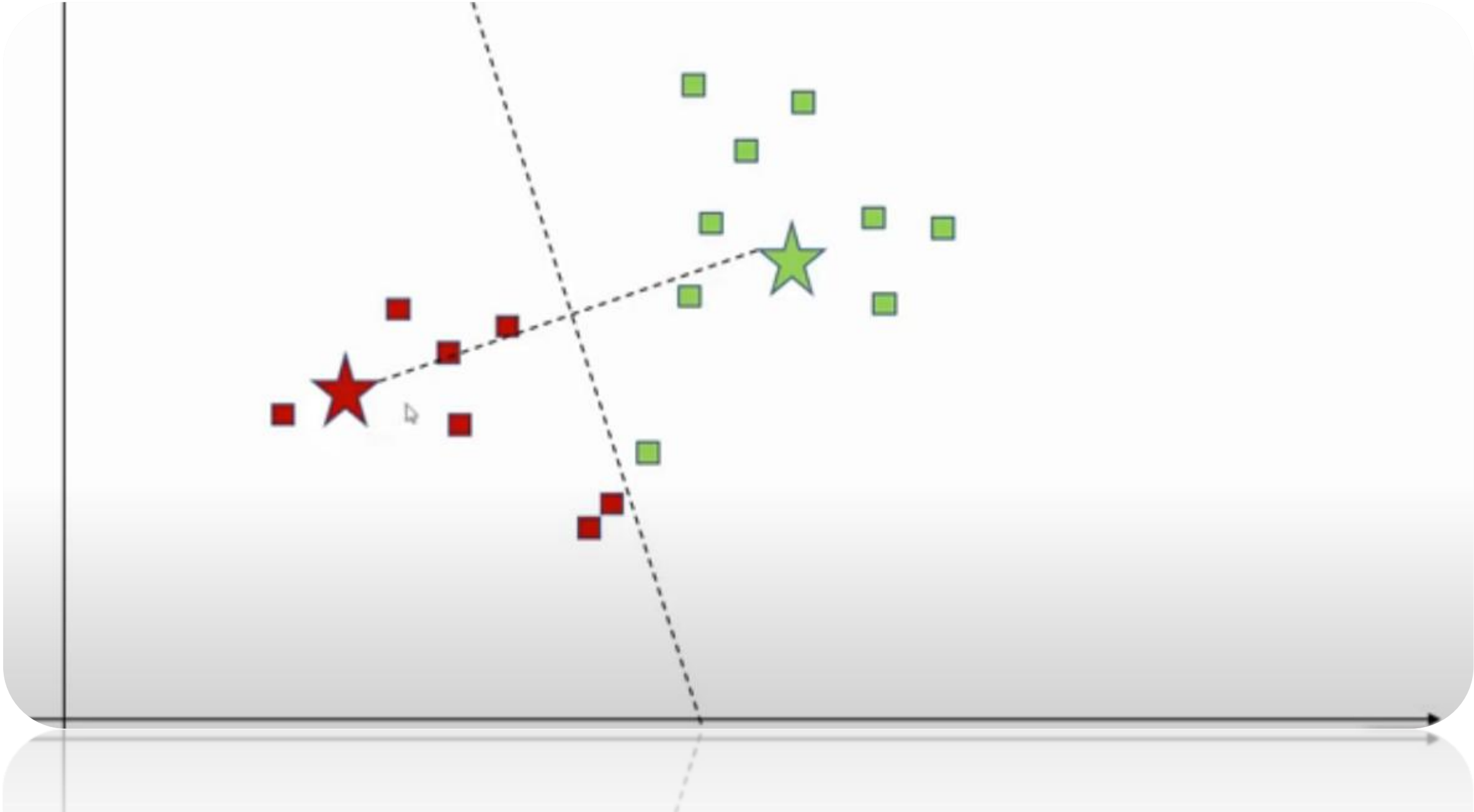
With first iteration, we are starting with 2 K's and they can be plotted anywhere on the graph at random places

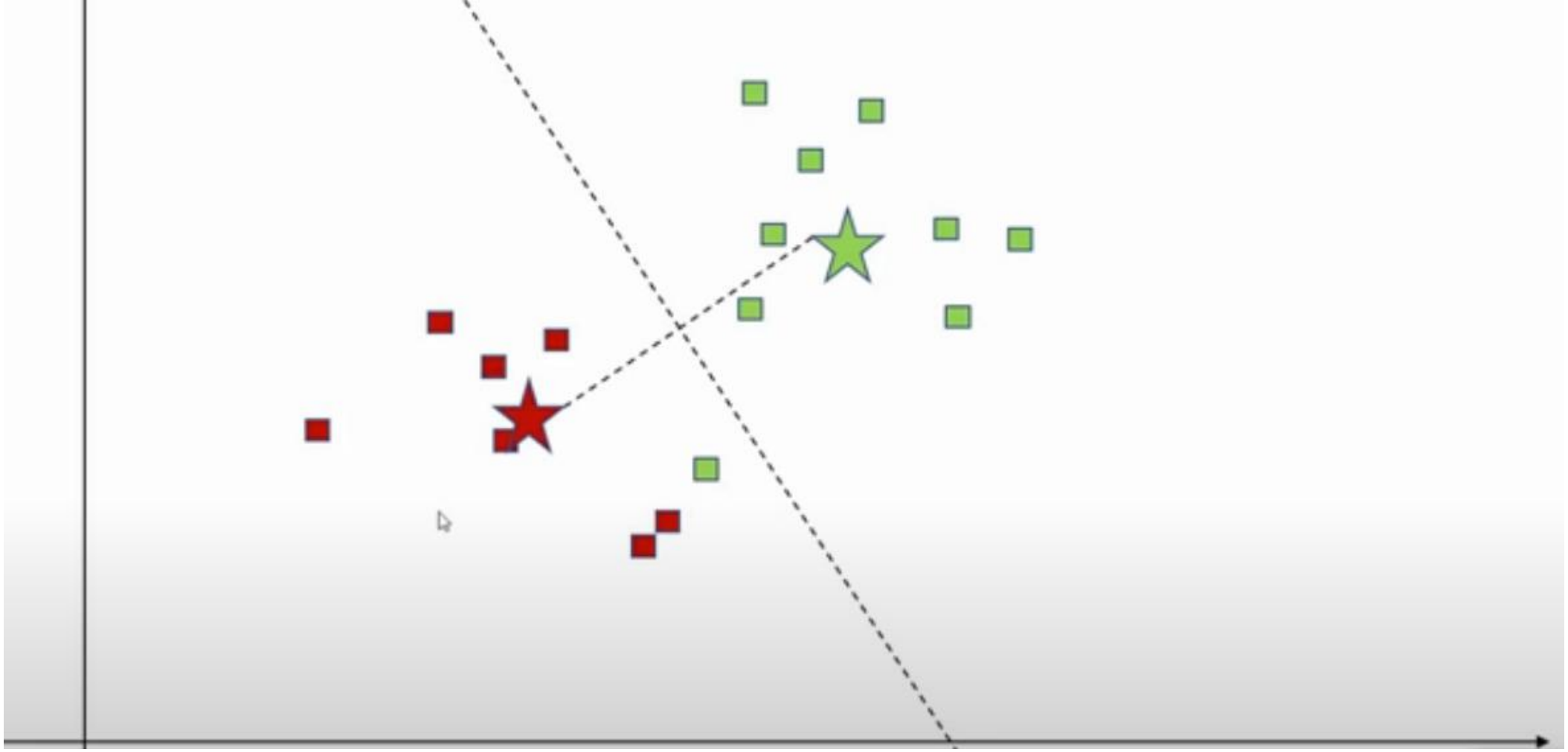


From the above image, the points are left side of the perpendicular line are labelled with red color whereas the points which are right hand side of the perpendicular are labelled as green color. This is the first iteration of K-Means cluster with K=2.

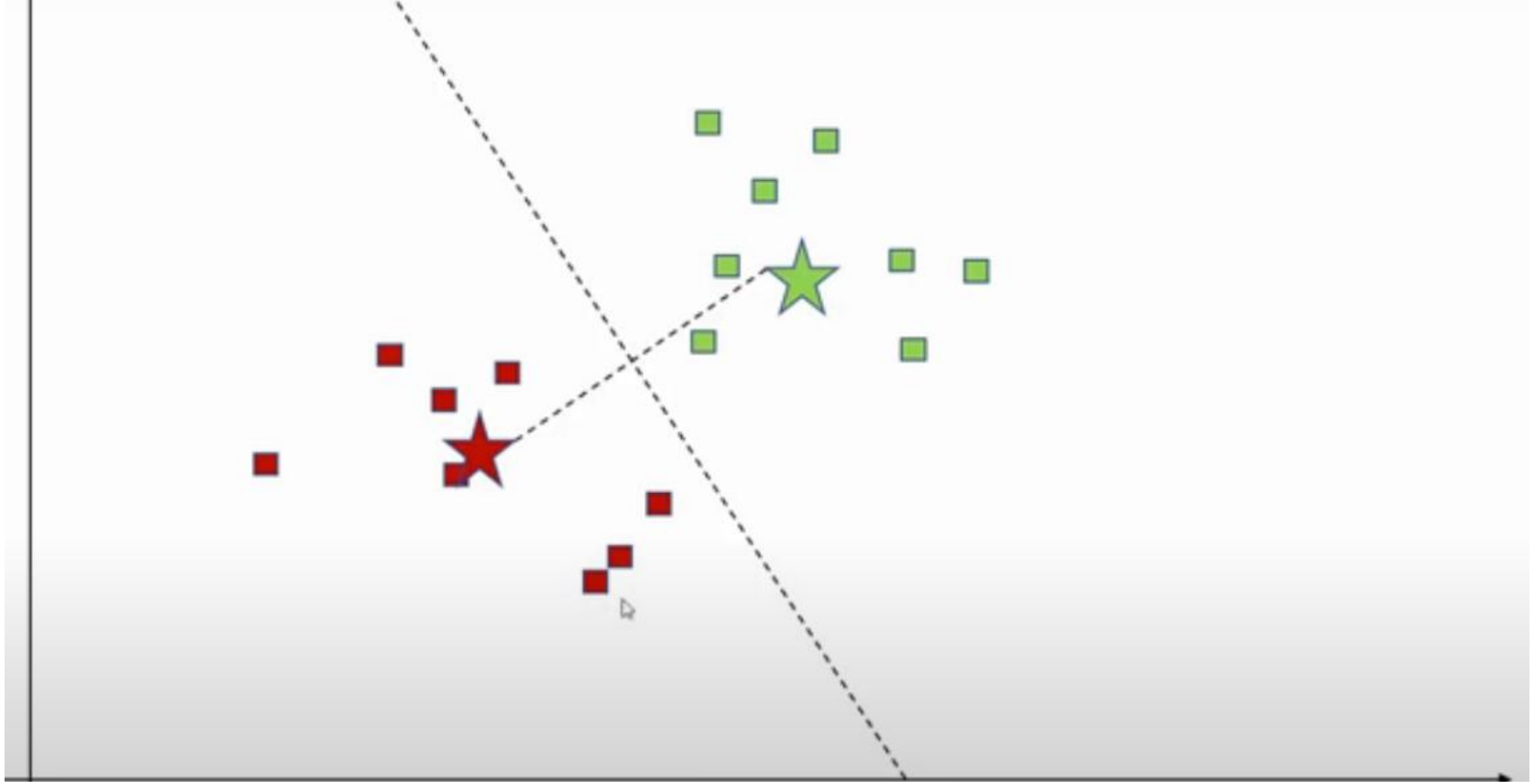


Now in 2nd iteration the centroids points are moved to their new location. And just like before a perpendicular line is being drawn to separate the centroids. Now the points that are left side of the line will be turned into red and the points which are on the right hand side will remained green.

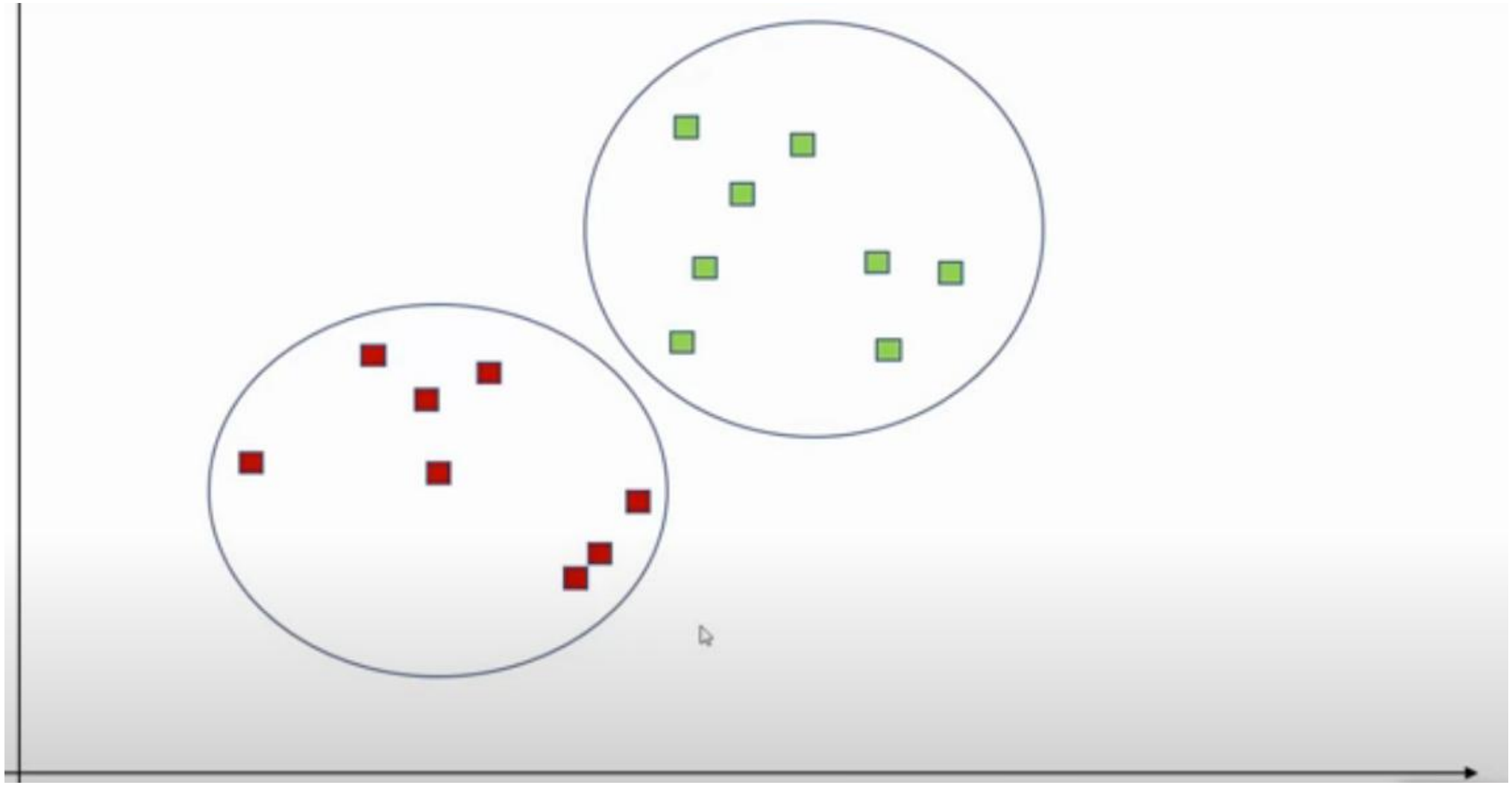




Again the algorithm will iterate for 3rd time and in here the centroids have changed their location and to separate the centroids the perpendicular line is being drawn, points that are on the left hand side will be turned to red and point that are on right hand side will be turned to green likewise

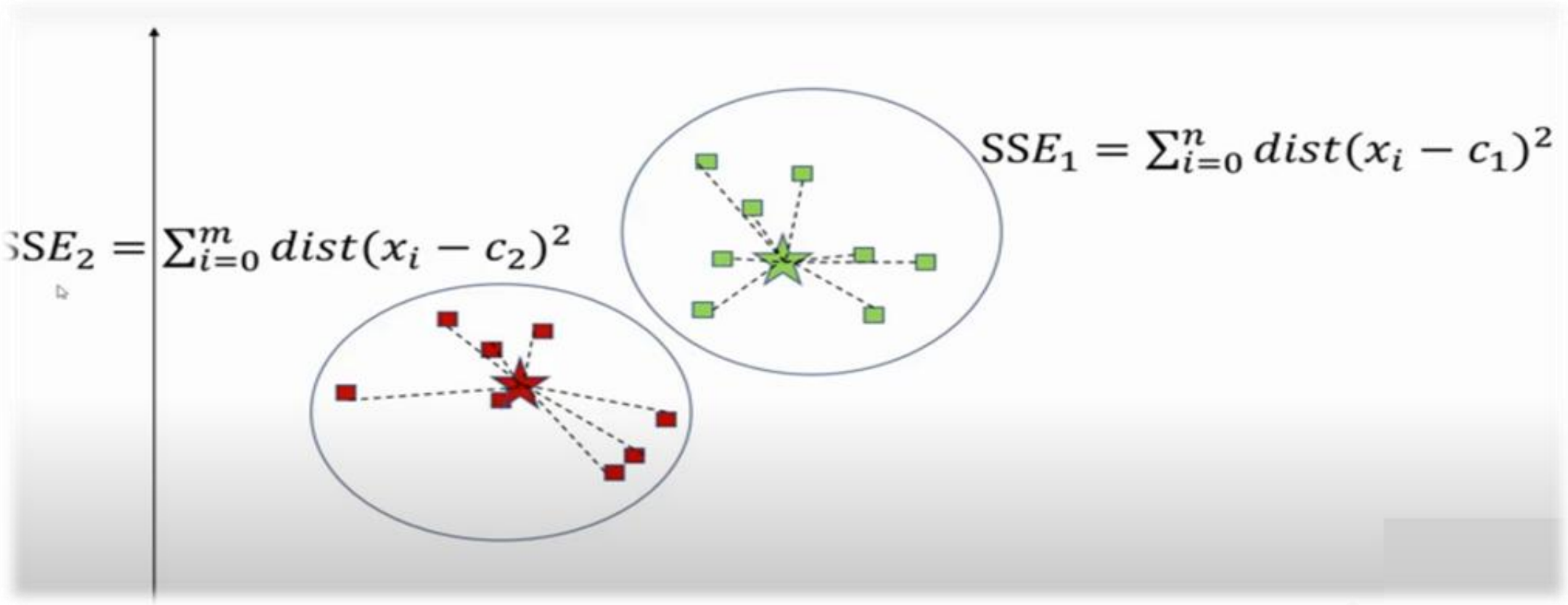


This is the last time the algorithm is iterating last time, it totally depends on data-to-data. Why last time? Because there are no more points that is responsible for shifting new centroids values. And right now the centroids have selected their own data-points. If there will be 3K's the algorithm will work similarly until the centroids won't move further.

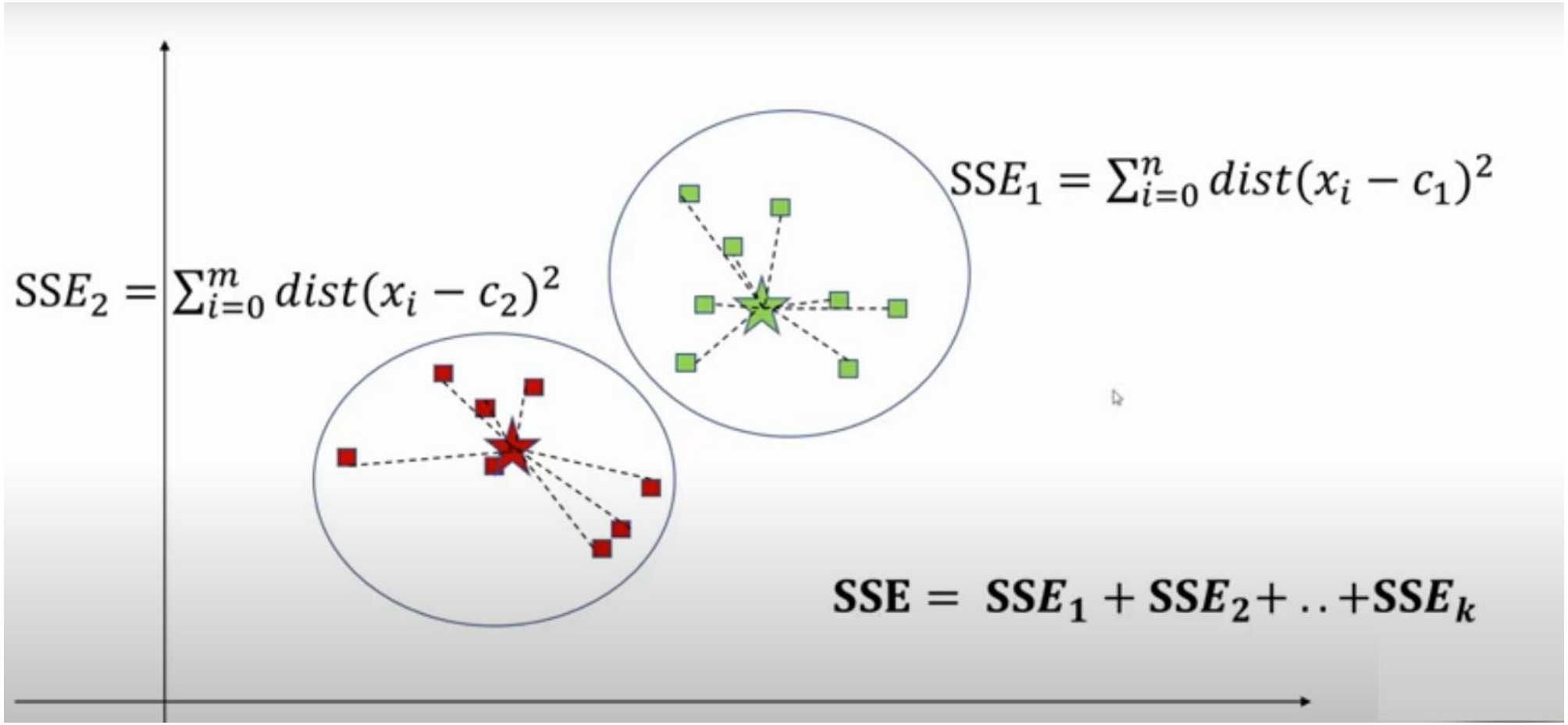


Final cluster after the iteration of K-Means algorithm

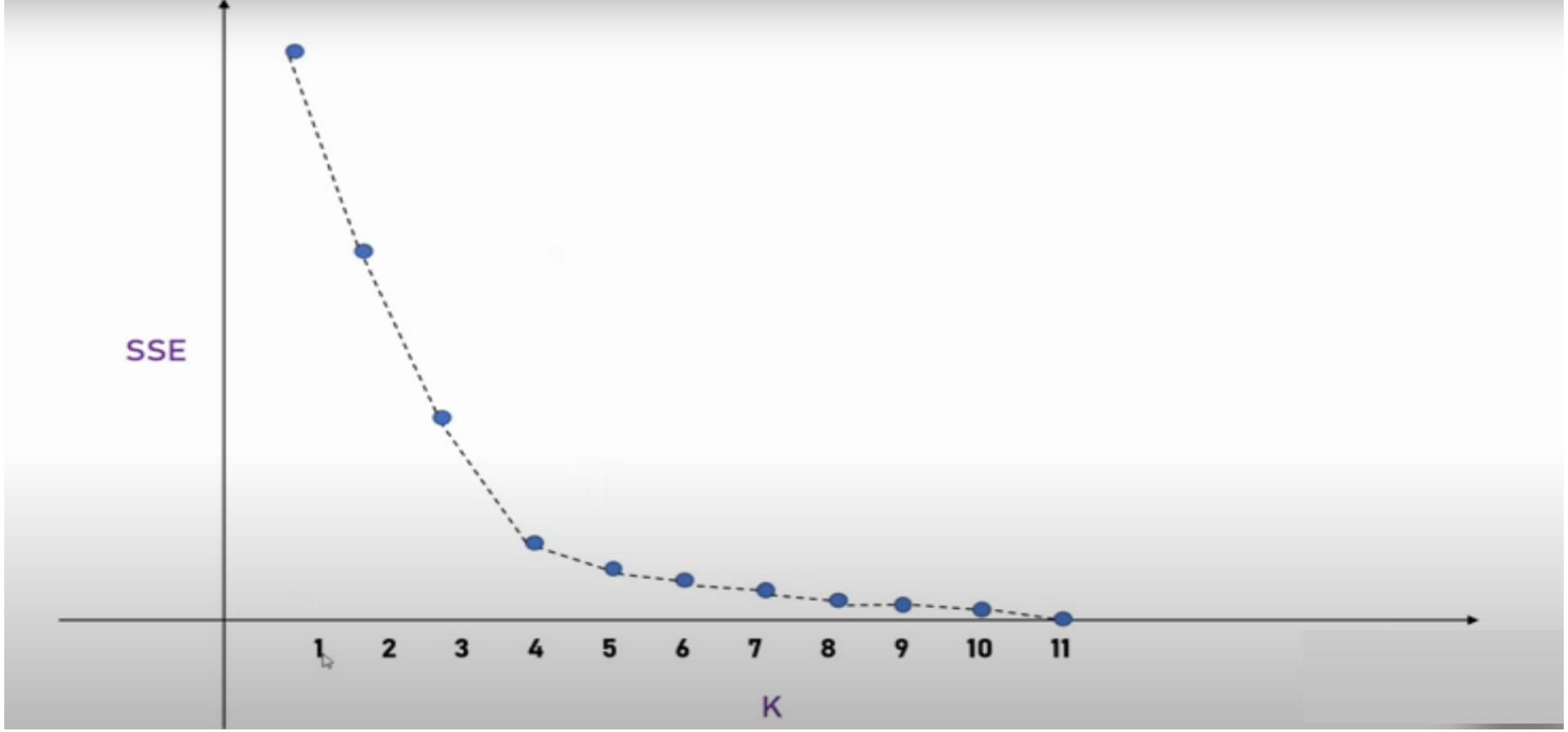
Determining Number of K's with Elbow Method



Initially it all starts with calculation SSE(Sum of Squared Errors) for each centroids with respective to their closest data points. As you can see in the image 2 separate SSE are being calculated with respect to each centroid value.

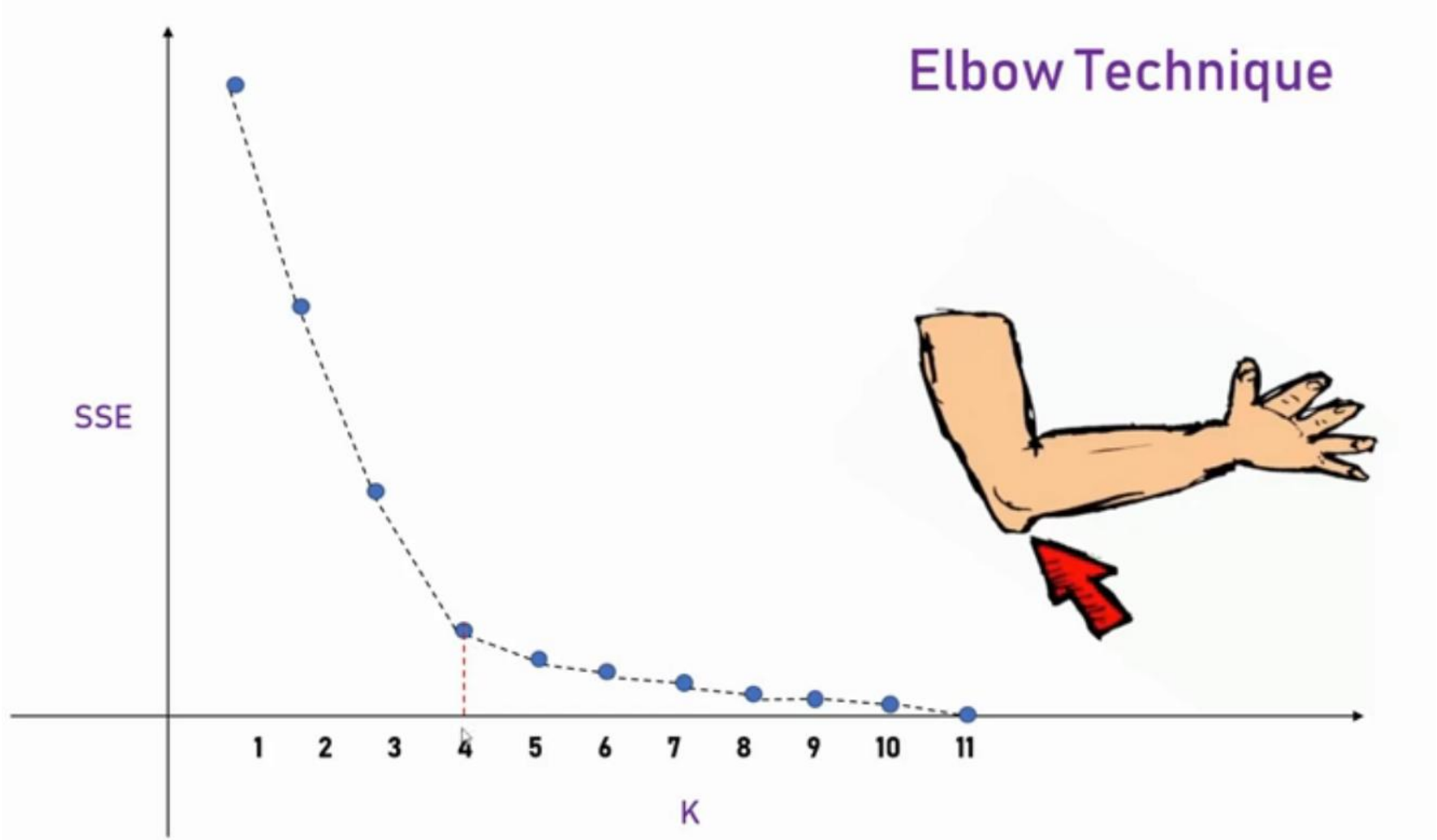


Then finally the SSE's will be added to get the final overview



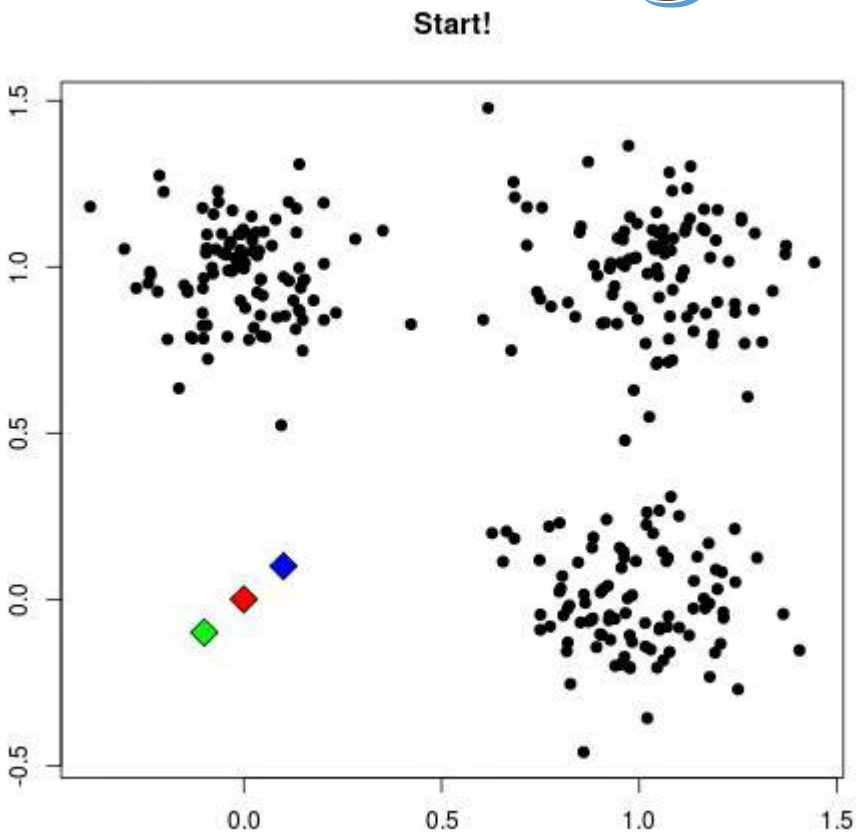
After adding multiple values from multiple k values we will plot the points and the above image is the reflection of the points that will get plotted.

As the number of K's get's increased the value of SSE's will also get decreased to the point where the points are not dropping much



This is the perfect representation of elbow method plot where after 4th K the SSE is not dropping so much hence a Data Scientist with the basic understanding of algorithms and K values will go with either 4th point or 5th point. The plot has drastically reduced the time complexity of the data scientist.

Quick Demonstration of K-Means Clustering



Metrics to check the Accuracy of Clusters

Silhouette score is used to evaluate the quality of clusters created using clustering algorithms such as K-Means in terms of how well samples are clustered with other samples that are similar to each other. The Silhouette score is calculated for each sample of different clusters. To calculate the Silhouette score for each observation/data point, the following distances need to be found out for each observations belonging to all the clusters:

- Mean distance between the observation and all other data points in the same cluster. This distance can also be called a **mean intra-cluster distance**. The mean distance is denoted by a
- Mean distance between the observation and all other data points of the next nearest cluster. This distance can also be called a **mean nearest-cluster distance**. The mean distance is denoted by b

The value of the Silhouette score varies from -1 to 1. If the score is 1, the cluster is dense and well-separated than other clusters. A value near 0 represents overlapping clusters with samples very close to the decision boundary of the neighboring clusters. A negative score [-1, 0] indicates that the samples might have got assigned to the wrong clusters.

Advantages and Disadvantages of K-Means

Advantages

1. Simple: It is easy to implement k-means and identify unknown groups of data from complex data sets. The results are presented in an easy and simple manner.

2. Flexible: K-means algorithm can easily adjust to the changes. If there are any problems, adjusting the cluster segment will allow changes to easily occur on the algorithm.

3. Suitable in a large dataset: K-means is suitable for a large number of datasets and it's computed much faster than the smaller dataset. It can also produce higher clusters.

4. Efficient: The algorithm used is good at segmenting the large data set. Its efficiency depends on the shape of the clusters. K-means work well in hyper-spherical clusters.

5. Easy to interpret: The results are easy to interpret. It generates cluster descriptions in a form minimized to ease understanding of the data.

Disadvantages

1. No-optimal set of clusters: K-means doesn't allow development of an optimal set of clusters and for effective results, you should decide on the clusters before.

2. Lacks consistency: K-means clustering gives varying results on different runs of an algorithm. A random choice of cluster patterns yields different clustering results resulting in inconsistency.

3. Sensitivity to scale: Changing or rescaling the dataset either through normalization or standardization will completely change the final results.

4. Handle numerical data: K-means algorithm can be performed in numerical data only.

5. Specify K-values: For K-means clustering to be effective, you have to specify the number of clusters (K) at the beginning of the algorithm.

6. Prediction issues: It is difficult to predict the k-values or the number of clusters. It is also difficult to compare the quality of the produced clusters.