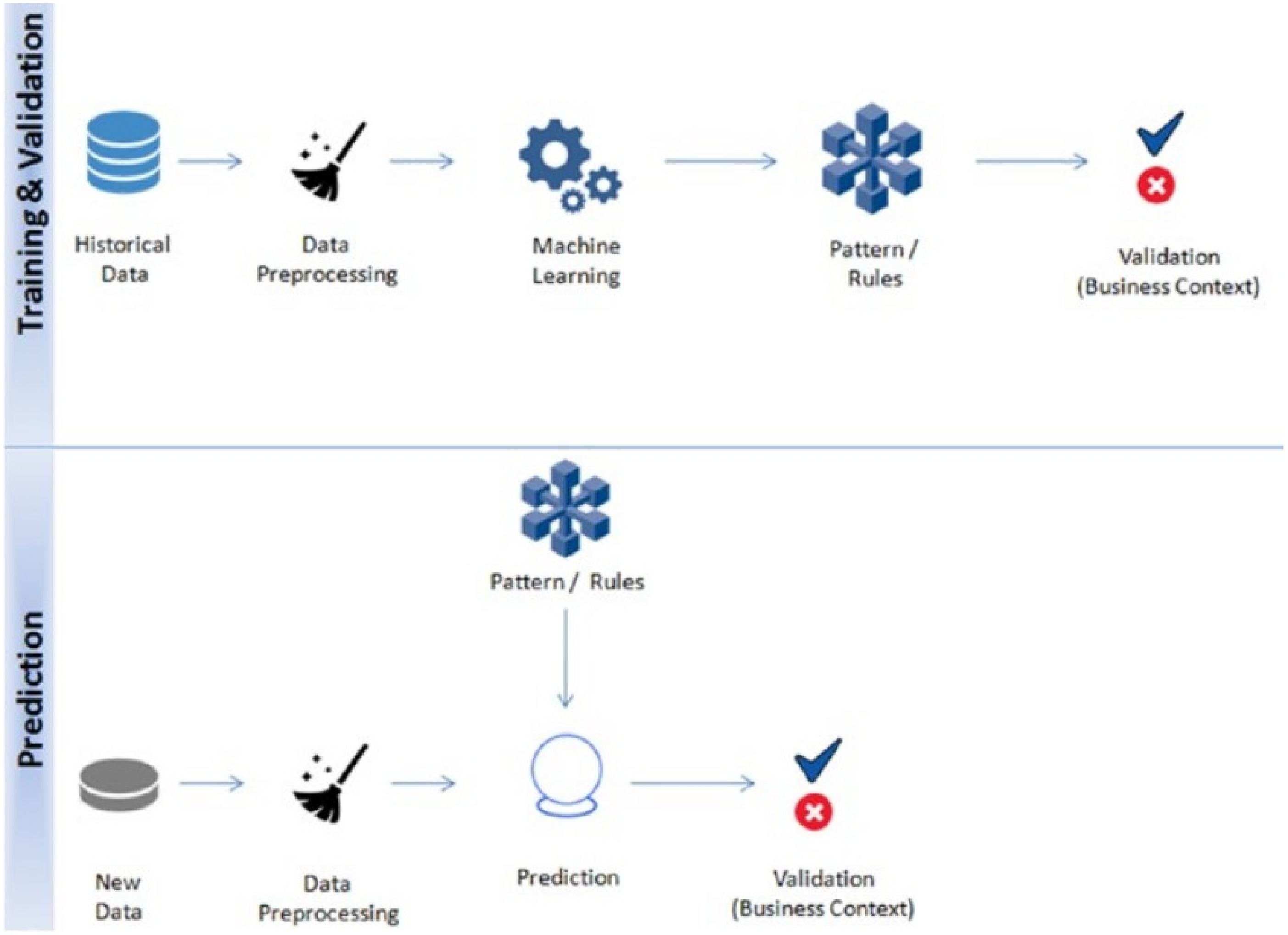
K-means Clustering

Tushar B. Kute,
http://tusharkute.com

Unsupervised learning flow



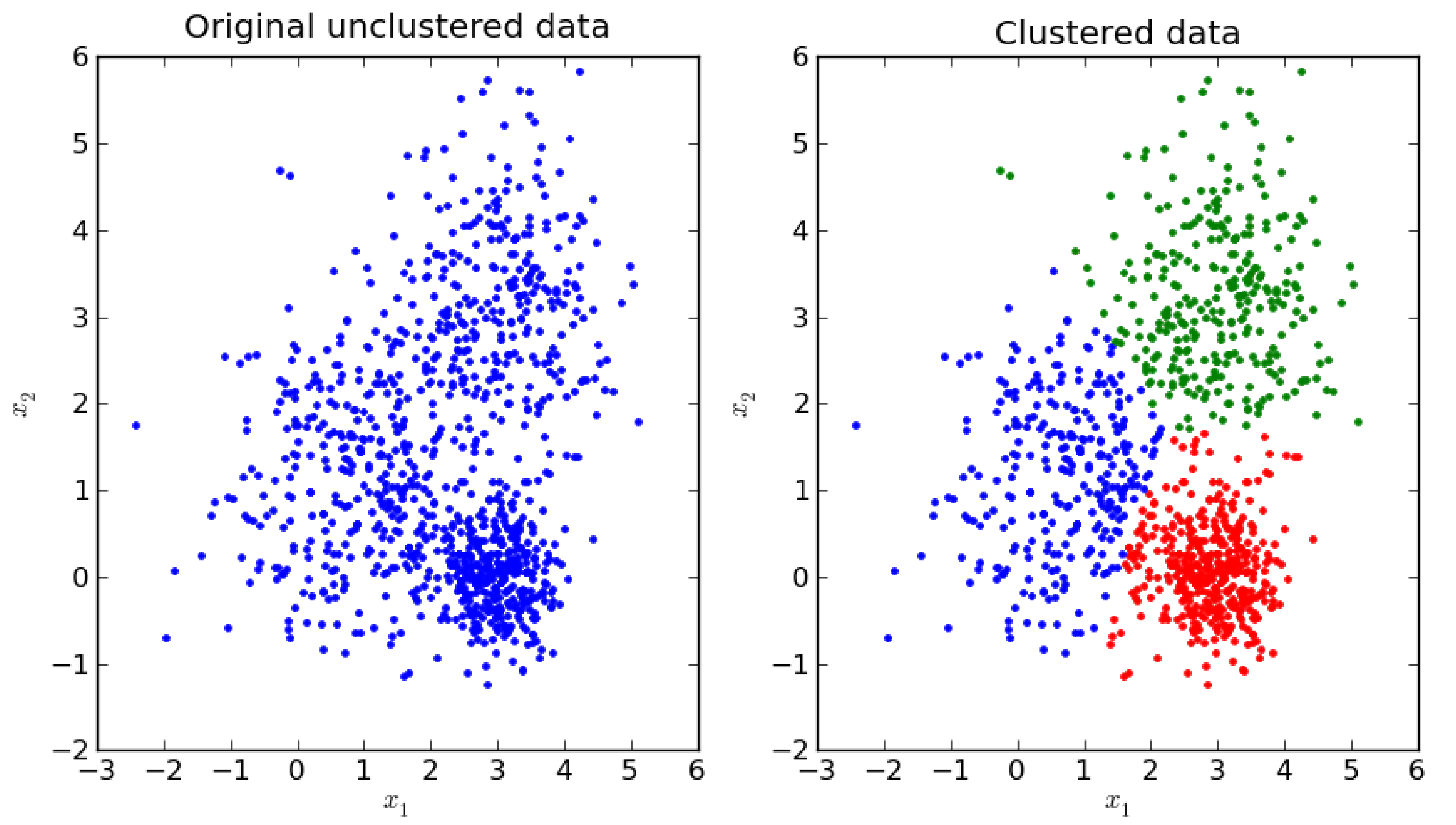
Clustering

- Clustering is an unsupervised learning problem.
- Key objective is to identify distinct groups (called clusters) based on some notion of similarity within a given dataset.
- Clustering analysis origins can be traced to the area of Anthropology and Psychology in the 193’s.
- The most popularly used clustering techniques are k-means (divisive) and hierarchical (agglomerative).

K-means clustering

- The key objective of a k-means algorithm is to organize data into clusters such that there is high intra-cluster similarity and low inter-cluster similarity. An item will only belong to one cluster, not several, that is, it generates a specific number of disjoint, non-hierarchical clusters.
- K-means uses the strategy of divide and concur, and it is a classic example for an expectation maximization (EM) algorithm. EM algorithms are made up of two steps:
  - The first step is known as expectation(E) and is used to find the expected point associated with a cluster; and
  - The second step is known as maximization(M) and is used to improve the estimation of the cluster using knowledge from the first step.
- The two steps are processed repeatedly until convergence is reached.

K-means clustering



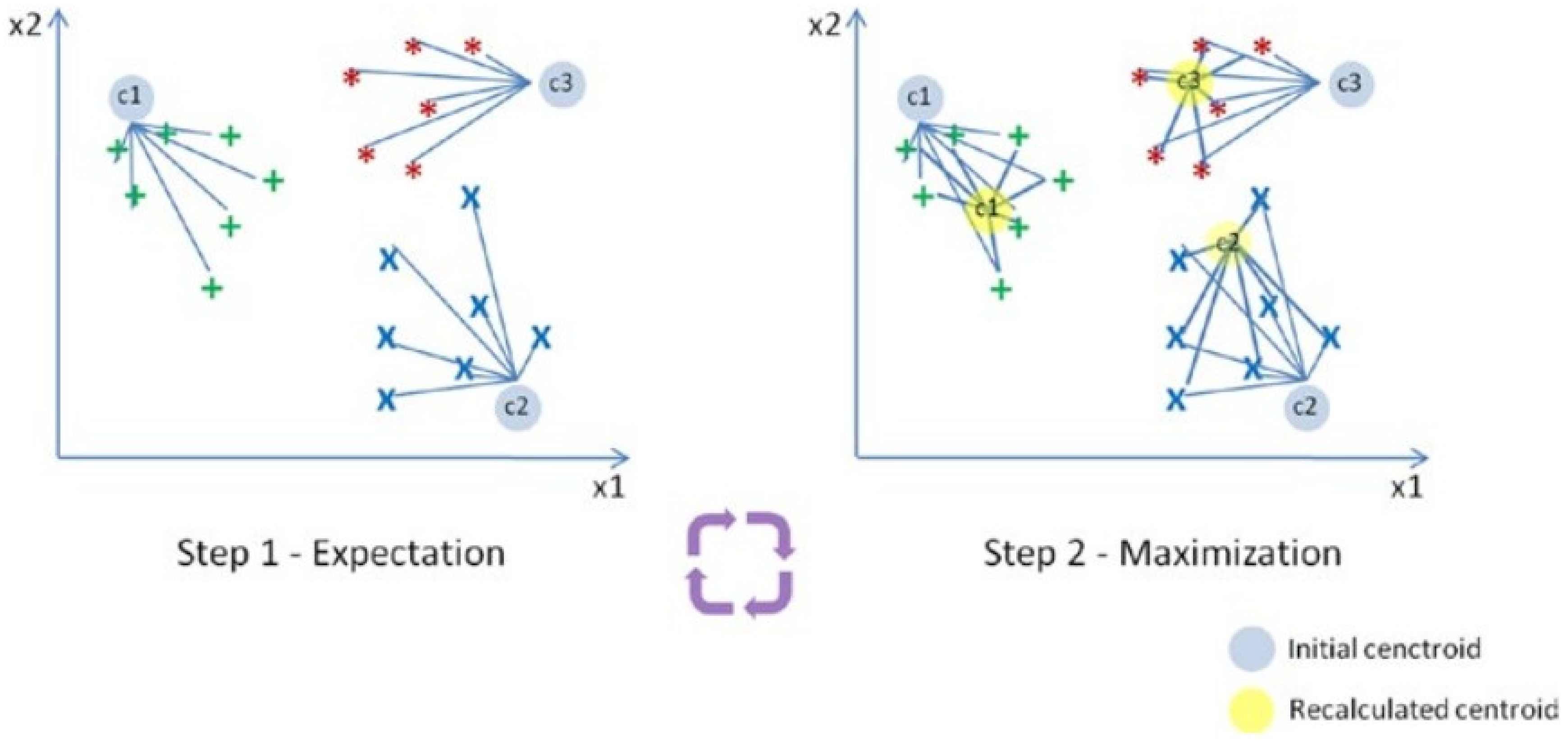
Generalized algorithm

- Our algorithm works as follows, assuming we have inputs x1, x2, x3,…,xn and value of K
  - Step 1 - Pick K random points as cluster centers called centroids.
  - Step 2 - Assign each xi to nearest cluster by calculating its distance to each centroid.
  - Step 3 - Find new cluster center by taking the average of the assigned points.
  - Step 4 - Repeat Step 2 and 3 until none of the cluster assignments change.

Algorithm work flow

- Step 1: In the first step k centroids (in above case k=3) is randomly picked (only in the first iteration) and all the points that are nearest to each centroid point are assigned to that specific cluster. Centroid is the arithmetic mean or average position of all the points.
- Step 2: Here the centroid point is recalculated using the average of the coordinates of all the points in that cluster. Then step one is repeated (assign nearest point) until the clusters converge.

Algorithm work flow



Limitations

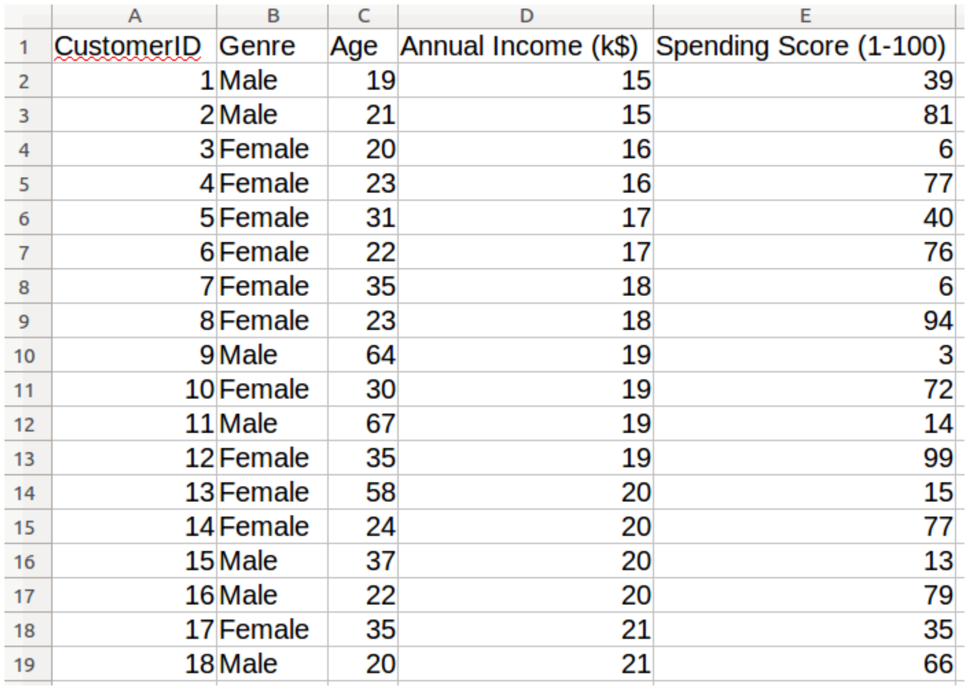
- K-means clustering needs the number of clusters to be specified.
- K-means has problems when clusters are of differing sized, densities, and non-globular shapes.
- Presence of outlier can skew the results.

Use cases

- K-Means is widely used for many applications.
  - Image Segmentation
  - Clustering Gene Segmentation Data
  - News Article Clustering
  - Clustering Languages
  - Species Clustering
  - Anomaly Detection

Example:

- Let’s assume the dataset of customers. This is information of Clients that subscribe to Membership card, maintains the Purchase history, score is Dependent on INCOME, and number times in week the show up in Mall, total expense in same mall



	A	B	C	D	E
1	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
2	1	Male	19	15	39
3	2	Male	21	15	81
4	3	Female	20	16	6
5	4	Female	23	16	77
6	5	Female	31	17	40
7	6	Female	22	17	76
8	7	Female	35	18	6
9	8	Female	23	18	94
10	9	Male	64	19	3
11	10	Female	30	19	72
12	11	Male	67	19	14
13	12	Female	35	19	99
14	13	Female	58	20	15
15	14	Female	24	20	77
16	15	Male	37	20	13
17	16	Male	22	20	79
18	17	Female	35	21	35
19	18	Male	20	21	66

Read the dataset

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Import Mall Dataset
dataset = pd.read_csv('Mall_Customers.csv')

# Get annual income and spending score
X = dataset.iloc[:, [3, 4]].values
```

How many clusters?

- There are two commonly used methods to determine the ideal number of clusters possible in K-means –
  - Elbow Method
  - Silhouette Method

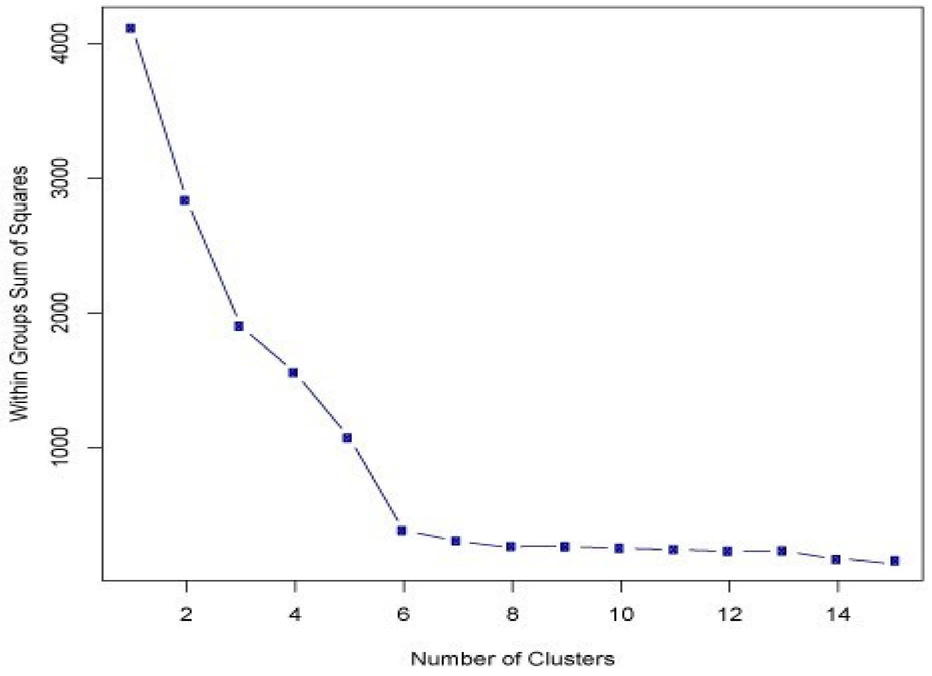
How many clusters?

- Elbow Method:
  - First of all, compute the sum of squared error (SSE) for some values of k (for example 2, 4, 6, 8, etc.). The SSE is defined as the sum of the squared distance between each member of the cluster and its centroid. Mathematically:

$$SSE = \sum_{i=1}^{K} \sum_{x \in c_i} dist(x, c_i)^2$$

  - If you plot k against the SSE, you will see that the error decreases as k gets larger; this is because when the number of clusters increases, they should be smaller, so distortion is also smaller. The idea of the elbow method is to choose the k at which the SSE decreases abruptly. This produces an "elbow effect" in the graph

Find no. of clusters



In this case, k=6 is the value that the Elbow method has selected.

Applying elbow method

```
from sklearn.cluster import KMeans
wcss = []
for i in range(1, 11):
    # Fit values into KM
    kmeans = KMeans(n_clusters=i)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
plt.plot(range(1, 11), wcss)
plt.title("ELBOW METHOD")
plt.xlabel("No. of Clus")
plt.ylabel("WCSS")
plt.plot()
plt.show()
```

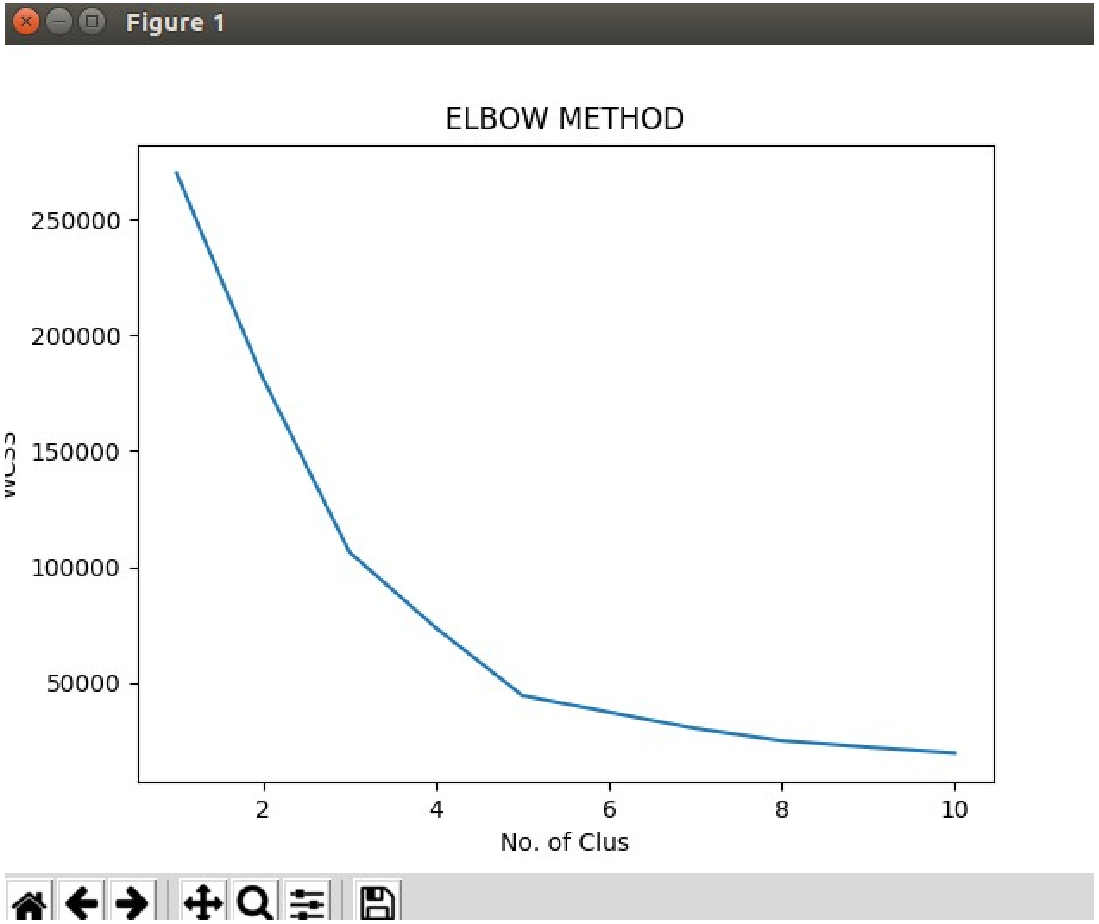
The Kmeans() function

- **`KMeans(n_clusters=8, init='k-means++', n_init=10, max_iter=300, tol=0.0001, precompute_distances='auto', verbose=0, random_state=None, copy_x=True, n_jobs=1, algorithm='auto')`**
  - n_clusters : int, optional, default: 8
    - The number of clusters to form as well as the number of centroids to generate.
  - random_state : int, RandomState instance or None, optional, default: None
    - If int, random_state is the seed used by the random number generator; If RandomState instance, random_state is the random number generator;

The Kmeans() function attributes

- cluster_centers_ : array, [n_clusters, n_features]
  - Coordinates of cluster centers
- labels_ : :
  - Labels of each point
- inertia_ : float
  - Sum of squared distances of samples to their closest cluster center.

Visualizing elbow



Got an elbow at 5

Silhouette Method

- Silhouette analysis is a way to measure how close each point in a cluster is to the points in its neighboring clusters. Its a way to find out the optimum value for k during k-means clustering.
- Silhouette values lies in the range of [-1, 1]. A value of +1 indicates that the sample is far away from its neighboring cluster and very close to the cluster its assigned.
- Similarly, value of -1 indicates that the point is close to its neighboring cluster than to the cluster its assigned.
- A value of 0 means its at the boundary of the distance between the two cluster. Value of +1 is idea and -1 is least preferred. Hence, higher the value better is the cluster configuration.

Silhouette Method

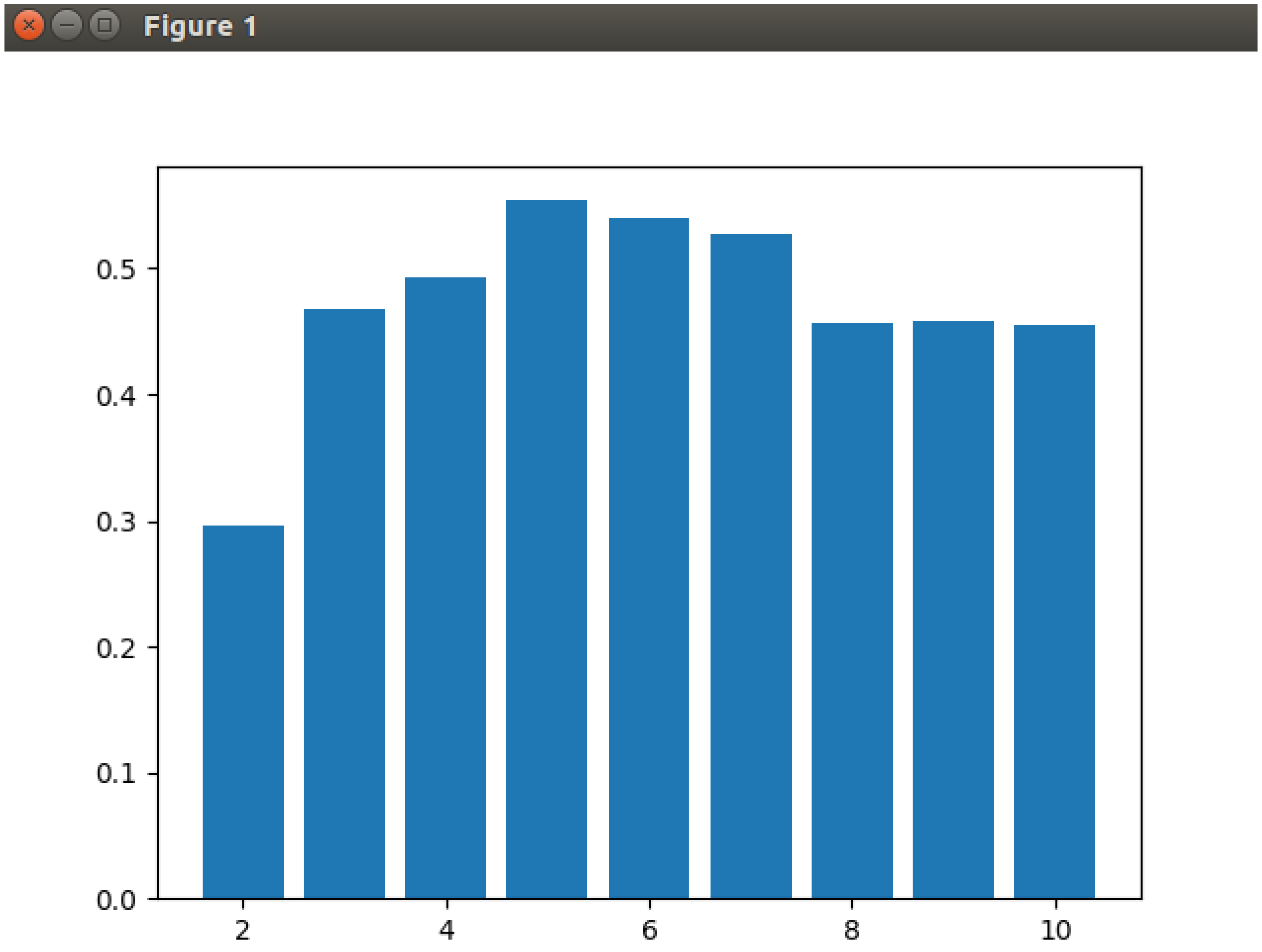
- In the data, lets define a(i) to be the mean distance of point (i) w.r.t to all the other points in the cluster its assigned (A). We can interpret a(i) as how well the point is assigned to the cluster. Smaller the value better the assignment.
- Similarly, lets define b(i) to be the mean distance of point(i) w.r.t. to other points to its closet neighboring cluster (B). The cluster (B) is the cluster to which point (i) is not assigned to but its distance is closest amongst all other cluster.
- Thus, the silhouette s(i) can be calculated as:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

Applying Silhouette Method

```
silh = []
for i in range(2, 11):
    # Fit values into KM
    kmeans = KMeans(n_clusters=i)
    labels = kmeans.fit_predict(X)
    silhouette_avg = silhouette_score(X, labels)
    silh.append(silhouette_avg)
    print 'Cluster',i,'Score:',silhouette_avg

plt.bar(range(2,11),silh)
plt.show()
print 'Ideal values of clusters:',
    silh.index(max(silh)) + 2
```

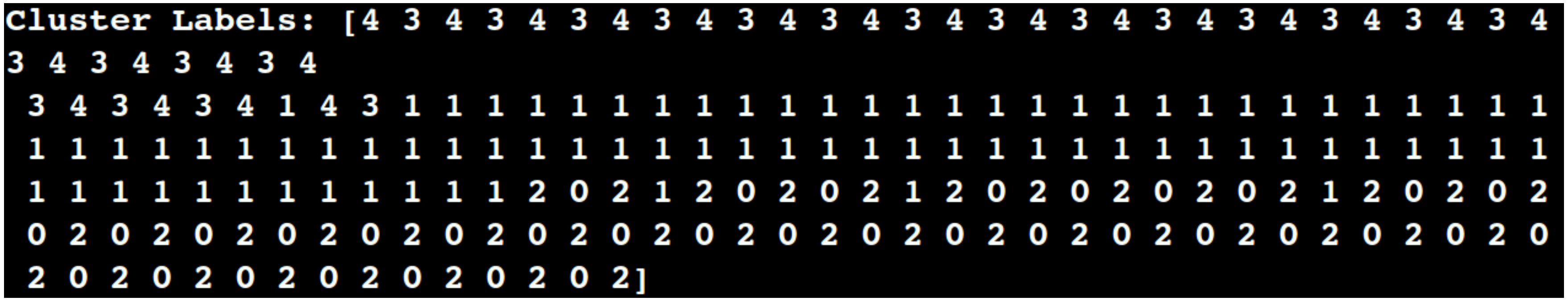
Find optimal value of k



Finding and visualizing

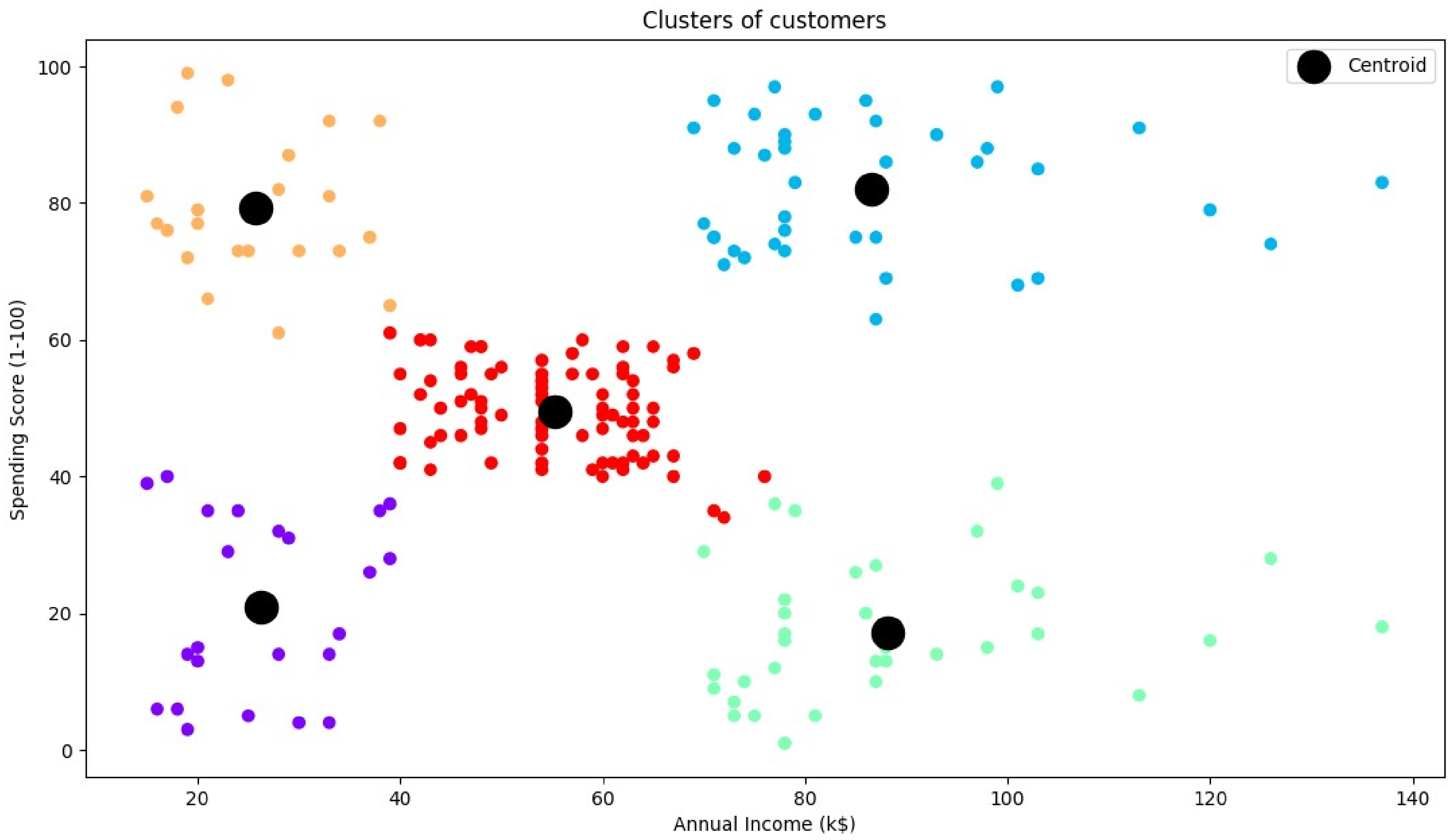
```
kmeans = KMeans(n_clusters=5)
y_kmeans = kmeans.fit_predict(X)

# Print cluster labels
print 'Cluster Labels:', y_kmeans

# Visualising the Clusters
plt.scatter(X[:,0], X[:,1], c=y_kmeans, cmap='rainbow')

# Cluster center visualization
plt.scatter(kmeans.cluster_centers_[:, 0],
kmeans.cluster_centers_[:, 1], s=300, c = 'black',
label='Centroid')
plt.title('Clusters of customers')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```

Cluster labels



```
Cluster Labels: [4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4 3 4
3 4 3 4 3 4 3 4
 3 4 3 4 3 4 1 4 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 2 0 2 1 2 0 2 0 2 1 2 0 2 0 2 0 2 0 2 0 2 1 2 0 2 0 2
 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0
 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2]
```

Visualizing clusters



Thank you

@mitu_skillologies

/mITuSkillologies

@mitu_group

/company/mitu-skillologies

Web Resources
http://mitu.co.in
http://tusharkute.com

contact@mitu.co.in

tushar@tusharkute.com