K-Means Clustering

```python
import numpy as np
np.random.seed(0)
import matplotlib.pyplot as plt
%matplotlib inline
%config InlineBackend.figure_format = 'retina'
from sklearn.datasets import make_blobs


def initialize_clusters(points, k):
    return points[np.random.randint
                    (points.shape[0], size=k)]


def get_distances(centroid, points):
    return np.linalg.norm(points - centroid, axis=1)


X, y = make_blobs(centers=3, n_samples=500,
                    random_state=1)


fig, ax = plt.subplots(figsize=(12,8))
ax.scatter(X[:, 0], X[:, 1], alpha=0.5)
ax.set_xlabel('$x_0$')
ax.set_ylabel('$x_1$')


k = 3
maxiter = 50
centroids = initialize_clusters(X, k)
classes = np.zeros(X.shape[0], dtype=np.float64)
distances = np.zeros([X.shape[0], k], dtype=np.float64)
```

```python
for i in range (maxiter):
    for i, c in enumerate (centroids):
        distances [:, i] = get_distances (c, x)
    classes = np. argmin ( distances, axis = 1)
    for c in range (k):
        centroids[c] = np. mean (x [classes == c], 0)


group_colors = ['skyblue', 'coral', 'lightgreen']
colors = [group_colors[j] for j in classes]


fig, ax = plt. subplots (figsize = (12, 8))
ax. scatter (x [:, 0], x [:, 1], color = colors, alpha = 0.5)
ax. scatter (centroids [:, 0], centroids [:, 1], colors =
            ['blue', 'darkred', 'green'], maker = 'o',
            lw = 2)
ax. set_xlabel ('$x_0$');
ax. set_ylabel ('$x_1$');


# Comparing my model with sklearn. KMeans.


from sklearn. cluster import KMeans
km = KMeans (n_clusters = 3, init = 'random', n_init = 10,
            max_iter = 50, tol = 1e-04, random state = 1)
y_km = km. fit_predict(x)


plt. figure (figsize = (12, 8))
plt. scatter (x [y_km == 0, 0], x [y_km == 0, 1],
            c = 'skyblue', marker = 'o', alpha = 0.5)
plt. scatter (x [y_km == 1, 0], x [y_km == 1, 1],
            c = 'coral', marker = 'o', alpha = 0.5)
plt. scatter (x [y_km == 2, 0], x [y_km == 2, 1],
            c = 'lightgreen', marker = 'o', alpha = 0.5)
```

```
plt.scatter (km.cluster_centers_[:,0],
            km.cluster_centers_[:,1],
         c = ['blue', 'darkred', 'green'],
         marker = 'o', alpha = 0.5)
plt.show()
```