K-Means Clustering

## Importing the libraries

```
In [0]:  import numpy as np
         import matplotlib.pyplot as plt
         import pandas as pd
```

Importing the dataset

```
In [0]:  dataset = pd.read_csv('Mall_Customers.csv')
         X = dataset.iloc[:, [3, 4]].values
```

Using the elbow method to find the optimal number of clusters

```
In [3]: from sklearn.cluster import KMeans
        wcss = []
        for i in range(1, 11):
            kmeans = KMeans(n_clusters = i, init = 'k-means++', random_state = 42
        )
            kmeans.fit(X)
            wcss.append(kmeans.inertia_)
        plt.plot(range(1, 11), wcss)
        plt.title('The Elbow Method')
        plt.xlabel('Number of clusters')
        plt.ylabel('WCSS')
        plt.show()
```
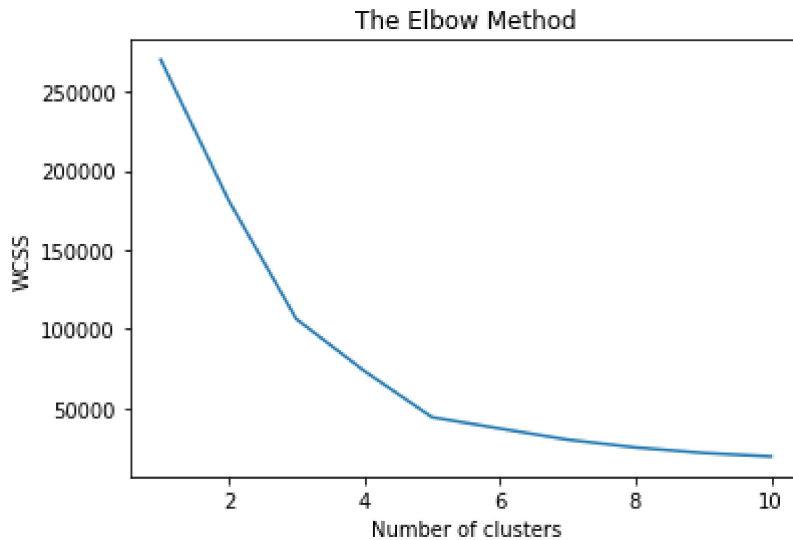

Training the K-Means model on the dataset

```
In [0]: kmeans = KMeans(n_clusters = 5, init = 'k-means++', random_state = 42)
        y_kmeans = kmeans.fit_predict(X)
```

Visualising the clusters

```python
In [5]: plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'red',
        label = 'Cluster 1')
        plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'blue'
        , label = 'Cluster 2')
        plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'gree
        n', label = 'Cluster 3')
        plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'cyan'
        , label = 'Cluster 4')
        plt.scatter(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], s = 100, c = 'magen
        ta', label = 'Cluster 5')
        plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1],
        s = 300, c = 'yellow', label = 'Centroids')
        plt.title('Clusters of customers')
        plt.xlabel('Annual Income (k$)')
        plt.ylabel('Spending Score (1-100)')
        plt.legend()
        plt.show()
```