Clustering Models

K-means method

Setting up for clustering analysis

In [1]:
```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt

import sklearn
from sklearn.preprocessing import scale
import sklearn.metrics as sm
from sklearn.metrics import confusion_matrix, classification_report
```

In [2]:
```
from sklearn.cluster import KMeans
from mpl_toolkits.mplot3d import Axes3D
from sklearn import datasets
```

In [3]:
```
%matplotlib inline
plt.figure(figsize=(7,4))
```

Out[3]:
```
<Figure size 504x288 with 0 Axes>
```

<Figure size 504x288 with 0 Axes>

In this lab we will work with iris dataset.

In [4]:
```
iris = datasets.load_iris()

```

In [6]:
```

X = scale(iris.data)
y = pd.DataFrame(iris.target)
variable_names = iris.feature_names
X[0:10]
```

Out[6]:
```
array([[-0.90068117,  1.01900435, -1.34022653, -1.3154443 ],
       [-1.14301691, -0.13197948, -1.34022653, -1.3154443 ],
       [-1.38535265,  0.32841405, -1.39706395, -1.3154443 ],
       [-1.50652052,  0.09821729, -1.2833891 , -1.3154443 ],
       [-1.02184904,  1.24920112, -1.34022653, -1.3154443 ],
       [-0.53717756,  1.93979142, -1.16971425, -1.05217993],
       [-1.50652052,  0.78880759, -1.34022653, -1.18381211],
       [-1.02184904,  0.78880759, -1.2833891 , -1.3154443 ],
       [-1.74885626, -0.36217625, -1.34022653, -1.3154443 ],
       [-1.14301691,  0.09821729, -1.2833891 , -1.44707648]])
```

In [7]:
```
1 y
```

Out[7]:

	0
0	0
1	0
2	0
3	0
4	0
...	...
145	2
146	2
147	2
148	2
149	2

150 rows × 1 columns

In [8]:
```
1 variable_names
```

Out[8]:
```
['sepal length (cm)',
 'sepal width (cm)',
 'petal length (cm)',
 'petal width (cm)']
```

This is what our X data looks like. Now we are going to cluster this data.

Building and running your model

In this section we will set the number of clusters and random number generator. Also, we need to fit our model to the data.

In [9]:
```
1 clustering = KMeans(n_clusters=3, random_state=5)
2
3 clustering.fit(X)
```

Out[9]:
```
KMeans(n_clusters=3, random_state=5)
```

Plotting your model outputs

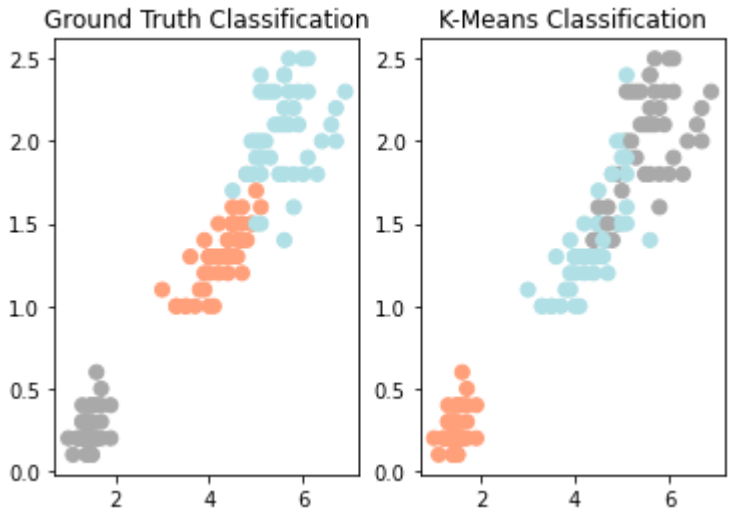
Here we will plot the model output.

In [10]:
```
iris_df = pd.DataFrame(iris.data)
iris_df.columns = ['Sepal_Length', 'Sepal_Width', 'Petal_Length', 'Petal_Wid
y.columns = ['Targets']
```

Now it is time to set the colour theme for our clusters. We set three colours.

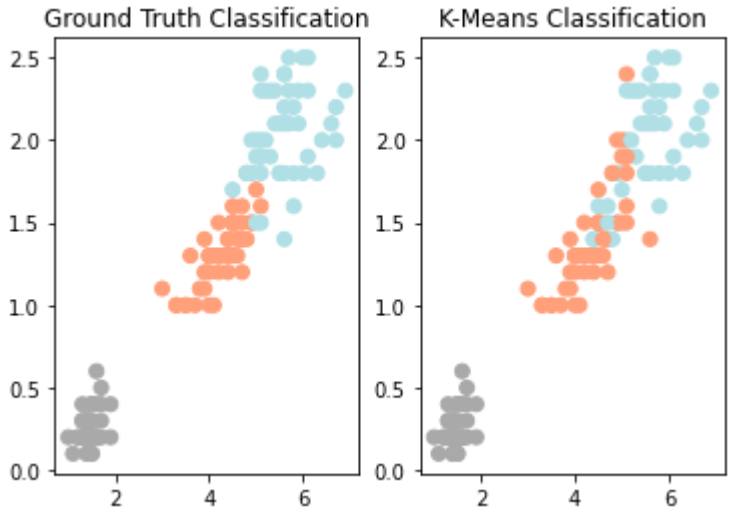
In [21]:
```
color_theme = np.array(['darkgray', 'lightsalmon', 'powderblue'])

plt.subplot(1,2,1)

plt.scatter(x=iris_df.Petal_Length, y=iris_df.Petal_Width, c=color_theme[iri
plt.title('Ground Truth Classification')

plt.subplot(1,2,2)

plt.scatter(x=iris_df.Petal_Length, y=iris_df.Petal_Width, c=color_theme[clu
plt.title('K-Means Classification')
```

Out[21]: Text(0.5, 1.0, 'K-Means Classification')



In [23]:
```
relabel = np.choose(clustering.labels_, [2, 0, 1 ]).astype(np.int64)

plt.subplot(1,2,1)

plt.scatter(x=iris_df.Petal_Length, y=iris_df.Petal_Width, c=color_theme[iri
plt.title('Ground Truth Classification')

plt.subplot(1,2,2)

plt.scatter(x=iris_df.Petal_Length, y=iris_df.Petal_Width, c=color_theme[rel
plt.title('K-Means Classification')
```

Out[23]: Text(0.5, 1.0, 'K-Means Classification')



Question 1: What is the difference between Code cells In [7] and In [8]? Why have we done relabeling?

Question 2: What is the difference between Ground Truth Classification and K-Means Classification in the above figures?

Evaluate your clustering results

In [60]:
```
print(classification_report(y, relabel))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        50
           1       0.74      0.78      0.76        50
           2       0.77      0.72      0.74        50

    accuracy                           0.83       150
   macro avg       0.83      0.83      0.83       150
weighted avg       0.83      0.83      0.83       150
```

Question 3: What do y and relabel represent in the print(classification_report(y, relabel)) command?

Question 4: What is precision and recall?