Name: - L Prathyusha

TASK 2 - Data science and Business Analytics Internship

In [19]:

```
import pandas as pd
import matplotlib.pyplot as plt
import warnings
import seaborn as sns
```

In [5]:

```
data = pd.read_csv('C:/Users/Prathyu Lachireddy/Desktop/Iris - Iris.csv')
data
```

Out[5]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...	...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [6]:

```
data.head()
```

Out[6]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

In [7]:

```
data.tail()
```

Out[7]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

In [8]:

```
data.nunique()
```

Out[8]:

```
Id               150
SepalLengthCm     35
SepalWidthCm      23
PetalLengthCm     43
PetalWidthCm      22
Species            3
dtype: int64
```

In [11]:

```
data.shape
```

Out[11]:

```
(150, 6)
```

In [12]:

```
data.columns
```

Out[12]:

```
Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthC
m',
       'Species'],
      dtype='object')
```

In [13]:

```
len(data)
```

Out[13]:

```
150
```

In [14]:

```
data.dtypes
```

Out[14]:

```
Id                int64
SepalLengthCm    float64
SepalWidthCm     float64
PetalLengthCm    float64
PetalWidthCm     float64
Species           object
dtype: object
```

In [25]:

```
data.corr()
```

Out[25]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
Id	1.000000	0.716676	-0.397729	0.882747	0.899759
SepalLengthCm	0.716676	1.000000	-0.109369	0.871754	0.817954
SepalWidthCm	-0.397729	-0.109369	1.000000	-0.420516	-0.356544
PetalLengthCm	0.882747	0.871754	-0.420516	1.000000	0.962757
PetalWidthCm	0.899759	0.817954	-0.356544	0.962757	1.000000

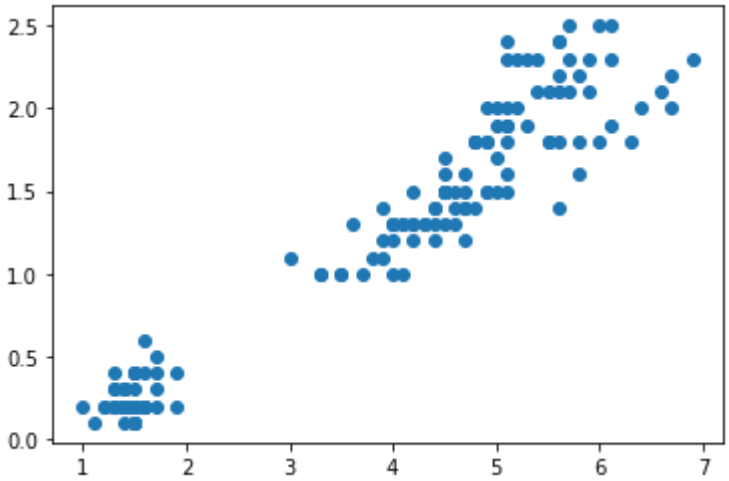
Petal Length vs Petal Width

In [24]:

```
plt.scatter(data['PetalLengthCm'],data['PetalWidthCm'])
```

Out[24]:

```
<matplotlib.collections.PathCollection at 0x2151c2c99a0>
```



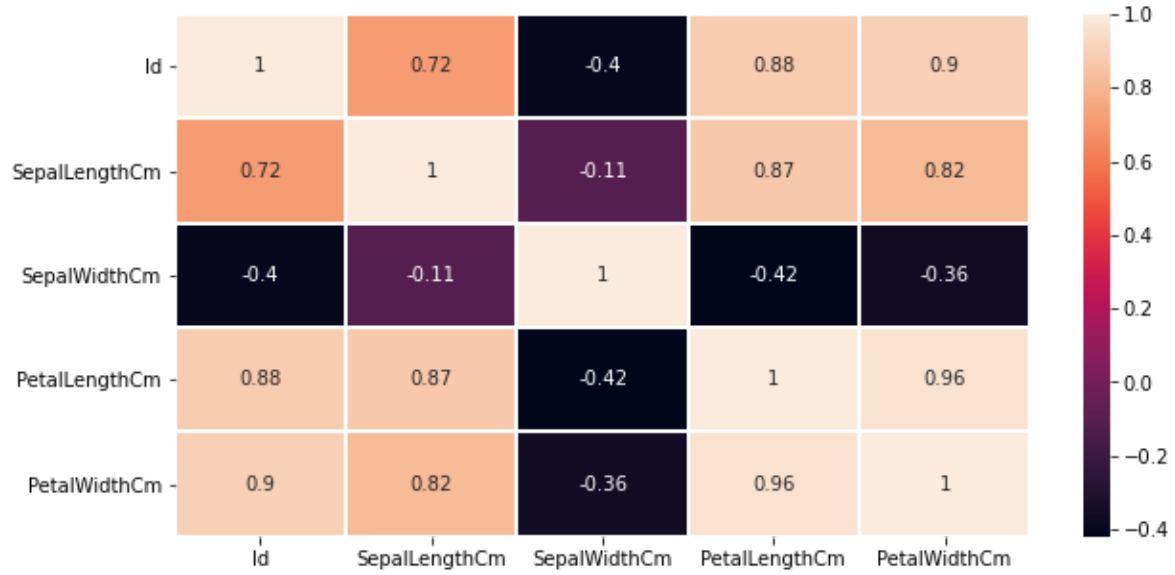
DATA VISUALISATION

In [31]:

```
Heat=plt.figure(figsize=(10,5))
sns.heatmap(data.corr(),linewidths=1,annot=True)
```

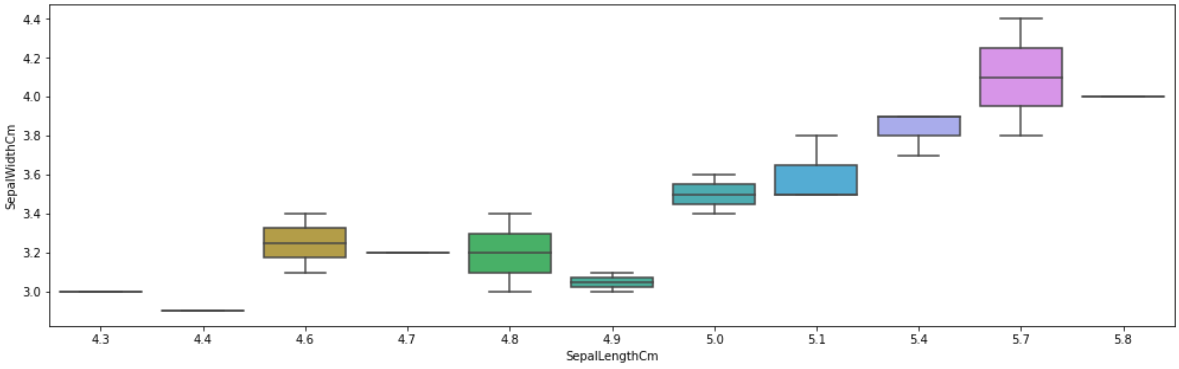
Out[31]:

<AxesSubplot:>



In [29]:

```
P=data.head(20)
fig_dims = (17, 5)
fig, ax = plt.subplots(figsize=fig_dims)
fig=sns.boxplot(data=P, x='SepalLengthCm',y='SepalWidthCm')
plt.show()
```

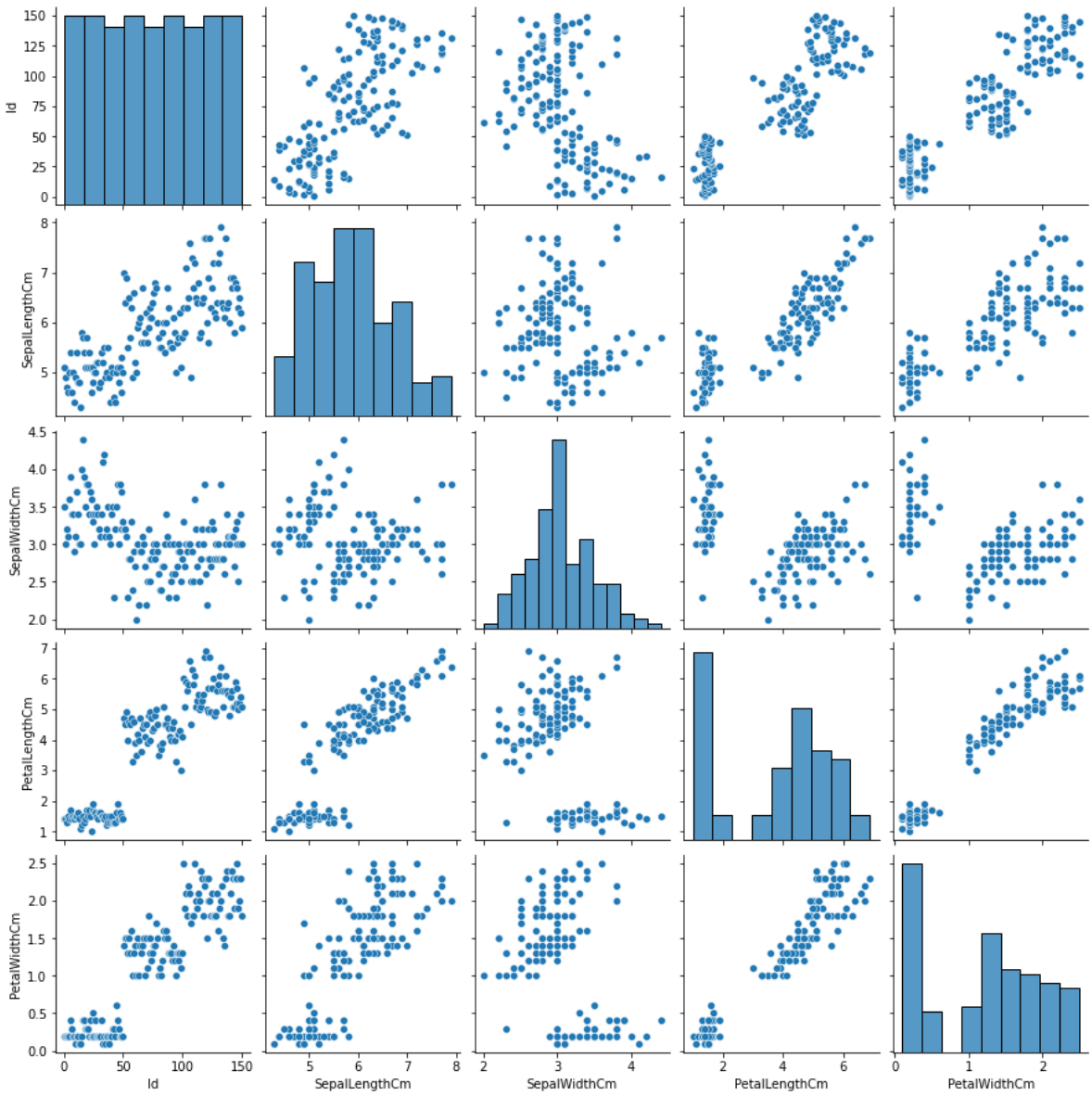


In [27]:

```
sns.pairplot(data)
```

Out[27]:

```
<seaborn.axisgrid.PairGrid at 0x2151c0fca60>
```

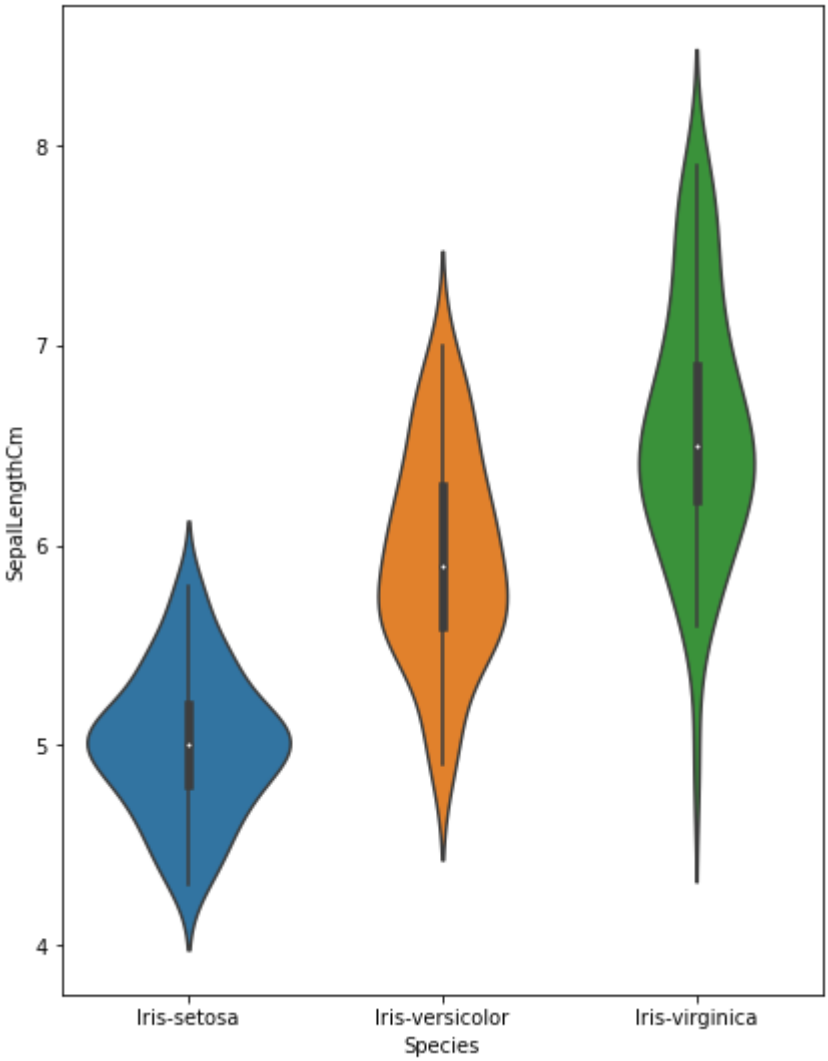


In [32]:

```
plt.figure(figsize=(15,20))
plt.subplot(2,2,1)
sns.violinplot(x='Species',y='SepalLengthCm',data=data)
```

Out[32]:

```
<AxesSubplot:xlabel='Species', ylabel='SepalLengthCm'>
```



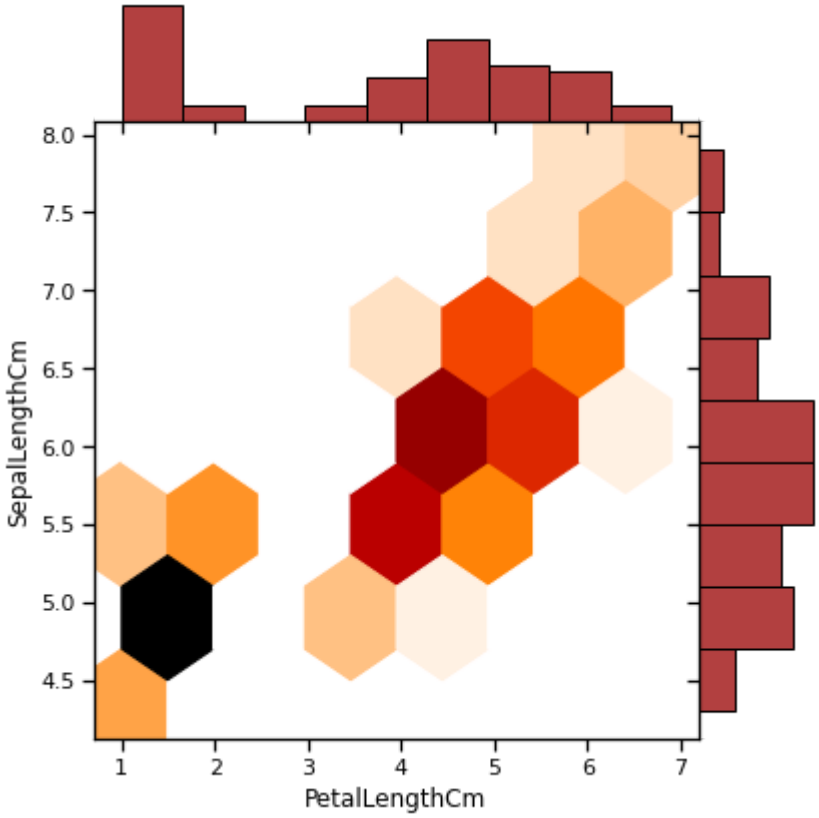
In [34]:

```
cmap=plt.cm.gist_heat_r
sns.jointplot(data['PetalLengthCm'],data['SepalLengthCm'],kind="hex",space=0, color=cmap(.6
```

```
C:\Users\Prathyu Lachireddy\anaconda3\lib\site-packages\seaborn\_decorators.
py:36: FutureWarning: Pass the following variables as keyword args: x, y. Fr
om version 0.12, the only valid positional argument will be `data`, and pass
ing other arguments without an explicit keyword will result in an error or m
isinterpretation.
  warnings.warn(
```

Out[34]:

```
<seaborn.axisgrid.JointGrid at 0x2151d6c73a0>
```



Optimum number of clusters

In [38]:

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
data['Species']=le.fit_transform(data['Species'])
data['Species'].value_counts()
```

Out[38]:

```
2    50
1    50
0    50
Name: Species, dtype: int64
```

In [46]:

```
from sklearn.cluster import KMeans
features=data.iloc[:,[0,1,2,3]].values
wcss=[]
for i in range(1,11):
    kmeans = KMeans(n_clusters = i, init = 'k-means++',random_state=42)
    kmeans.fit(data)
    wcss.append(kmeans.inertia_)

wcss
```

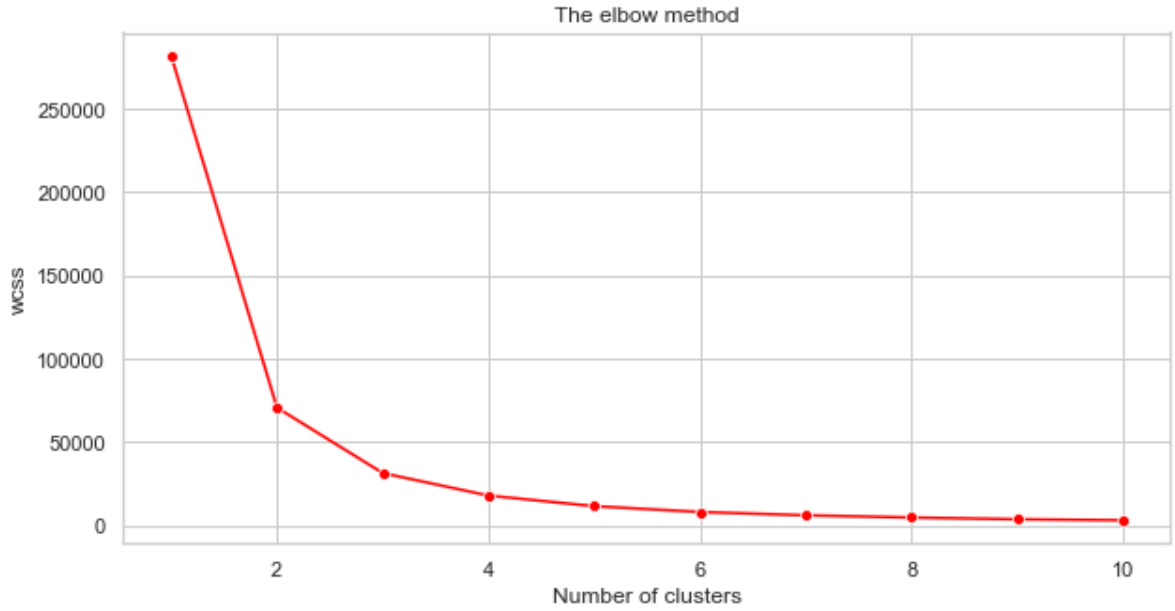
Out[46]:

```
[282018.32439999987,
 70649.63226666668,
 31326.886800000007,
 17800.405256045524,
 11454.835972191317,
 7943.981827777778,
 5939.900176265764,
 4602.8868015873,
 3565.483296568627,
 2969.778712184875]
```

In [47]:

```
plt.figure(figsize=(10,5))
sns.set(style='whitegrid')
sns.lineplot(range(1,11),wcss,marker='o',color='red')
plt.title('The elbow method')
plt.xlabel('Number of clusters')
plt.ylabel('wcss')
plt.show()
```

```
C:\Users\Prathyu Lachireddy\anaconda3\lib\site-packages\seaborn\_decorators.
py:36: FutureWarning: Pass the following variables as keyword args: x, y. Fr
om version 0.12, the only valid positional argument will be `data`, and pass
ing other arguments without an explicit keyword will result in an error or m
isinterpretation.
  warnings.warn(
```



In [49]:

```
kmeans=KMeans(n_clusters=3,init='k-means++',random_state=5)
y_means=kmeans.fit_predict(features)
y_means
```

Out[49]:

```
array([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, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
```

In [53]:

```
fig=plt.figure(figsize=(10,7))
plt.title('Clusters with centroids')
plt.scatter(features[y_means ==0,0], features[y_means==0,1], s=100,c='seagreen')
plt.scatter(features[y_means ==1,0], features[y_means==1,1], s=100,c='purple')
plt.scatter(features[y_means ==2,0], features[y_means==2,1], s=100,c='yellow')
```

Out[53]:

```
<matplotlib.collections.PathCollection at 0x2151f7a0100>
```