Project on Breast Cancer Prediction

In [102]:
```
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [103]:
```
df=pd.read_csv(r"C:\Users\rubin\Downloads\BreastCancerPrediction.csv")
df.head()
```

Out[103]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_m
0	842302	M	17.99	10.38	122.80	1001.0	0.1
1	842517	M	20.57	17.77	132.90	1326.0	0.0
2	84300903	M	19.69	21.25	130.00	1203.0	0.1
3	84348301	M	11.42	20.38	77.58	386.1	0.1
4	84358402	M	20.29	14.34	135.10	1297.0	0.1

5 rows × 33 columns

In [95]:
```
df.head()
```

Out[95]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_m
0	842302	M	17.99	10.38	122.80	1001.0	0.1
1	842517	M	20.57	17.77	132.90	1326.0	0.0
2	84300903	M	19.69	21.25	130.00	1203.0	0.1
3	84348301	M	11.42	20.38	77.58	386.1	0.1
4	84358402	M	20.29	14.34	135.10	1297.0	0.1

5 rows × 33 columns

In [96]:
```
df.tail()
```

Out[96]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_m
564	926424	M	21.56	22.39	142.00	1479.0	0.1
565	926682	M	20.13	28.25	131.20	1261.0	0.09
566	926954	M	16.60	28.08	108.30	858.1	0.08
567	927241	M	20.60	29.33	140.10	1265.0	0.1
568	92751	B	7.76	24.54	47.92	181.0	0.05

5 rows × 33 columns

In [97]:
```
df.fillna(method='ffill',inplace=True)
```

In [98]:
```
df.isnull().sum()
```

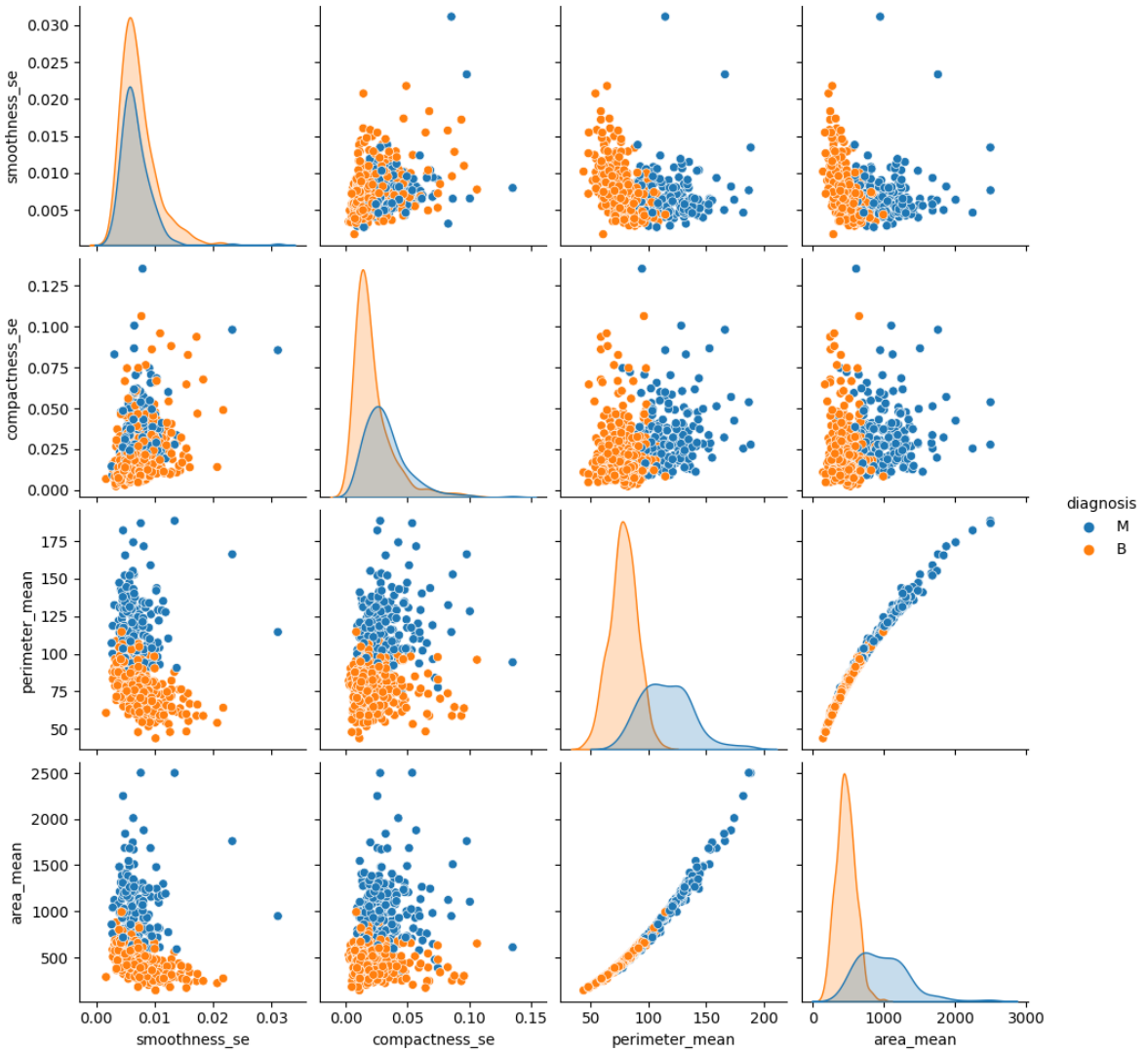
Out[98]:
```
id                         0
diagnosis                  0
radius_mean                0
texture_mean               0
perimeter_mean             0
area_mean                  0
smoothness_mean            0
compactness_mean           0
concavity_mean             0
concave points_mean        0
symmetry_mean              0
fractal_dimension_mean     0
radius_se                  0
texture_se                 0
perimeter_se               0
area_se                    0
smoothness_se              0
compactness_se             0
concavity_se               0
concave points_se          0
symmetry_se                0
fractal_dimension_se       0
radius_worst               0
texture_worst              0
perimeter_worst            0
area_worst                 0
smoothness_worst           0
compactness_worst          0
concavity_worst            0
concave points_worst       0
symmetry_worst             0
fractal_dimension_worst    0
Unnamed: 32                569
dtype: int64
```

In [99]:
```
df.columns
```

Out[99]:
```
Index(['id', 'diagnosis', 'radius_mean', 'texture_mean', 'perimeter_mean',
       'area_mean', 'smoothness_mean', 'compactness_mean', 'concavity_mean',
       'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
       'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
       'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
       'fractal_dimension_se', 'radius_worst', 'texture_worst',
       'perimeter_worst', 'area_worst', 'smoothness_worst',
       'compactness_worst', 'concavity_worst', 'concave points_worst',
       'symmetry_worst', 'fractal_dimension_worst', 'Unnamed: 32'],
      dtype='object')
```

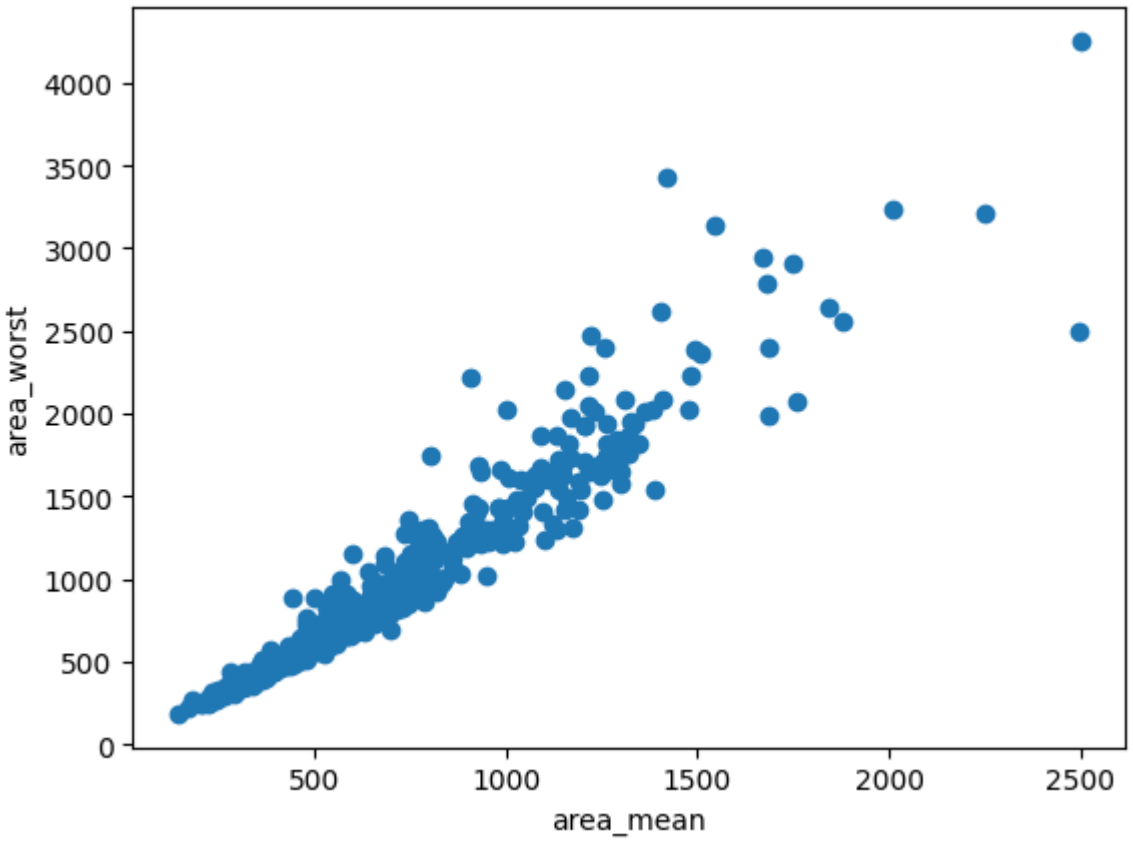
In [74]:
```
cols = ["diagnosis", "smoothness_se", "compactness_se", "perimeter_mean", "are

sns.pairplot(df[cols], hue="diagnosis")
plt.show()
```



In [75]:
```
plt.scatter(df["area_mean"],df["area_worst"])
plt.xlabel("area_mean")
plt.ylabel("area_worst")
```

Out[75]:
```
Text(0, 0.5, 'area_worst')
```



In [76]:
```
from sklearn.cluster import KMeans
```

In [77]:
```
km=KMeans()
km
```

Out[77]:
```
KMeans()
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [78]:
```
y_predicted=km.fit_predict(df[["area_mean","area_worst"]])
y_predicted
```

```
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
```

Out[78]:
```
array([0, 0, 5, 1, 5, 1, 5, 6, 1, 1, 2, 2, 2, 6, 1, 6, 6, 2, 0, 1, 1, 4,
       6, 3, 0, 2, 6, 2, 2, 2, 5, 6, 2, 5, 2, 2, 6, 1, 6, 1, 6, 4, 5, 6,
       1, 5, 4, 1, 1, 1, 1, 1, 1, 2, 6, 1, 0, 6, 1, 4, 4, 4, 6, 4, 6, 6,
       4, 4, 4, 1, 5, 4, 5, 6, 1, 2, 1, 5, 5, 1, 1, 1, 3, 2, 1, 5, 6, 5,
       1, 6, 6, 6, 6, 1, 6, 5, 1, 4, 1, 6, 6, 4, 1, 4, 4, 6, 1, 1, 0, 1,
       4, 1, 1, 4, 4, 1, 4, 6, 2, 2, 1, 5, 0, 6, 1, 1, 6, 5, 6, 5, 1, 2,
       2, 6, 5, 1, 1, 4, 6, 4, 4, 2, 4, 1, 4, 1, 1, 6, 6, 1, 1, 4, 4, 4,
       1, 1, 2, 2, 1, 4, 1, 5, 0, 1, 0, 6, 4, 2, 5, 6, 1, 6, 6, 4, 4, 4,
       4, 6, 1, 1, 3, 0, 2, 4, 6, 4, 2, 1, 1, 1, 6, 1, 4, 1, 6, 1, 6, 2,
       5, 6, 1, 2, 0, 6, 1, 6, 4, 2, 1, 6, 5, 1, 3, 2, 6, 6, 1, 4, 0, 0,
       1, 1, 4, 6, 1, 6, 4, 6, 1, 1, 2, 4, 4, 0, 4, 1, 3, 5, 6, 2, 1, 1,
       4, 1, 5, 4, 1, 1, 4, 4, 0, 1, 0, 2, 0, 6, 0, 6, 2, 6, 5, 2, 2, 6,
       2, 3, 4, 1, 1, 4, 1, 4, 0, 4, 2, 4, 4, 2, 1, 1, 5, 1, 5, 6, 1, 1,
       1, 1, 4, 4, 6, 6, 1, 1, 1, 1, 4, 1, 6, 4, 0, 1, 5, 4, 4, 4, 1, 4,
       1, 1, 4, 6, 1, 4, 4, 1, 1, 5, 4, 1, 4, 5, 1, 0, 1, 1, 6, 1, 2, 6,
       6, 1, 4, 4, 1, 2, 1, 5, 4, 3, 6, 4, 4, 5, 1, 4, 1, 6, 4, 1, 1, 6,
       3, 6, 4, 1, 1, 1, 4, 4, 1, 1, 1, 6, 1, 5, 5, 1, 3, 0, 2, 6, 5, 0,
       1, 6, 4, 1, 1, 4, 4, 4, 1, 1, 1, 6, 1, 6, 4, 2, 4, 4, 2, 0, 1, 1,
       1, 1, 4, 1, 2, 1, 1, 1, 1, 4, 6, 1, 2, 1, 1, 4, 4, 6, 6, 1, 4, 5,
       1, 4, 1, 6, 4, 1, 4, 4, 4, 4, 4, 1, 6, 1, 5, 5, 6, 6, 1, 6, 6, 1,
       4, 2, 1, 4, 2, 1, 2, 6, 6, 0, 1, 5, 1, 6, 1, 1, 1, 1, 1, 4, 5, 7,
       6, 4, 1, 1, 1, 4, 2, 1, 4, 1, 6, 1, 4, 1, 6, 1, 4, 6, 1, 6, 1, 1,
       6, 1, 6, 5, 1, 2, 1, 2, 2, 1, 1, 6, 1, 1, 5, 5, 6, 6, 1, 3, 4, 4,
       1, 4, 6, 6, 4, 6, 6, 6, 6, 4, 5, 5, 1, 1, 4, 3, 4, 1, 4, 4, 1, 1,
       1, 1, 1, 1, 6, 5, 4, 5, 6, 4, 4, 4, 4, 6, 6, 1, 1, 1, 4, 4, 4, 4,
       4, 4, 1, 4, 1, 4, 4, 4, 6, 4, 1, 4, 6, 5, 0, 5, 2, 5, 4])
```

In [79]:
```
df["cluster"]=y_predicted
df.head()
```

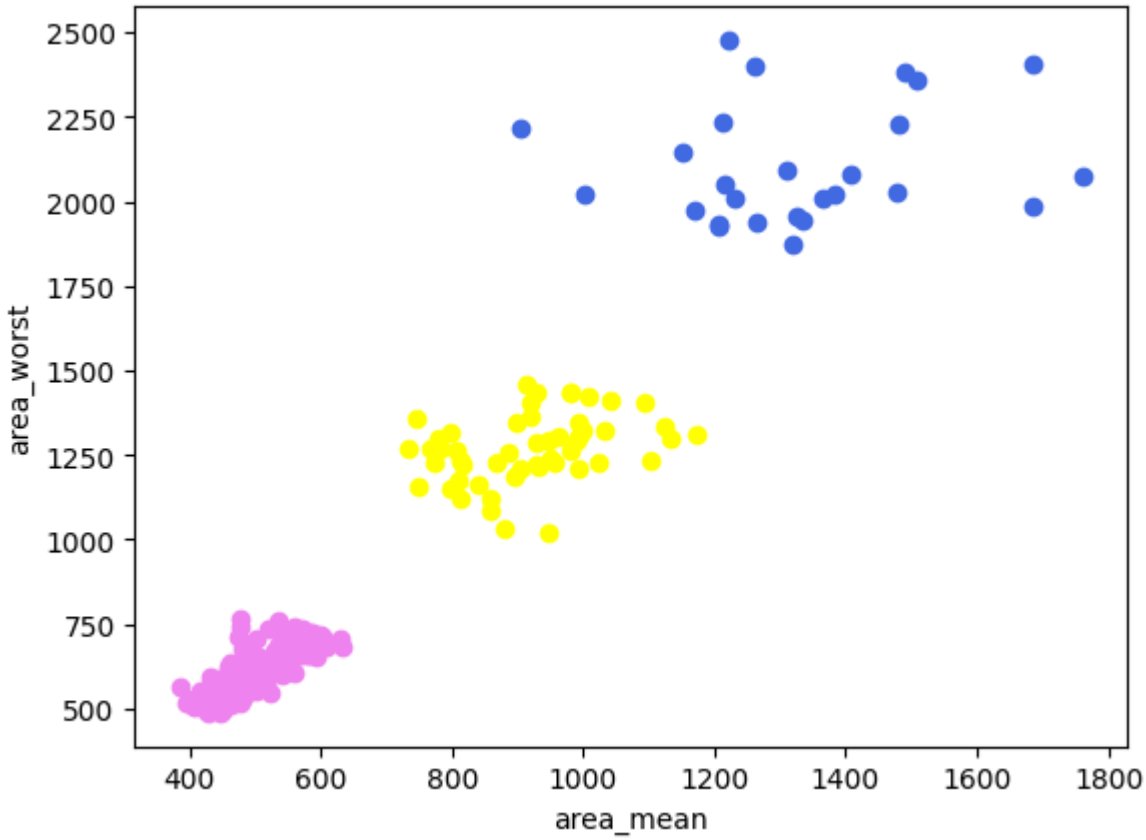
Out[79]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_m
0	842302	M	17.99	10.38	122.80	1001.0	0.1
1	842517	M	20.57	17.77	132.90	1326.0	0.0
2	84300903	M	19.69	21.25	130.00	1203.0	0.1
3	84348301	M	11.42	20.38	77.58	386.1	0.1
4	84358402	M	20.29	14.34	135.10	1297.0	0.1

5 rows × 34 columns

In [80]:
```
df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]
plt.scatter(df1["area_mean"],df1["area_worst"],color="royalblue")
plt.scatter(df2["area_mean"],df2["area_worst"],color="violet")
plt.scatter(df3["area_mean"],df3["area_worst"],color="yellow")
plt.xlabel("area_mean")
plt.ylabel("area_worst")
```

Out[80]: Text(0, 0.5, 'area_worst')



In [81]:
```
from sklearn.preprocessing import MinMaxScaler
```

In [82]:
```
Scaler=MinMaxScaler()
```

In [83]:
```
Scaler.fit(df[["area_mean"]])
df["area_mean"]=Scaler.transform(df[["area_mean"]])
df.head()
```

Out[83]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_m
0	842302	M	17.99	10.38	122.80	0.363733	0.1
1	842517	M	20.57	17.77	132.90	0.501591	0.0
2	84300903	M	19.69	21.25	130.00	0.449417	0.1
3	84348301	M	11.42	20.38	77.58	0.102906	0.1
4	84358402	M	20.29	14.34	135.10	0.489290	0.1

5 rows × 34 columns

In [84]:
```
Scaler.fit(df[["area_worst"]])
df["area_worst"]=Scaler.transform(df[["area_worst"]])
df.head()
```

Out[84]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_m
0	842302	M	17.99	10.38	122.80	0.363733	0.1
1	842517	M	20.57	17.77	132.90	0.501591	0.0
2	84300903	M	19.69	21.25	130.00	0.449417	0.1
3	84348301	M	11.42	20.38	77.58	0.102906	0.1
4	84358402	M	20.29	14.34	135.10	0.489290	0.1

5 rows × 34 columns

In [85]:
```
km=KMeans()
km
```

Out[85]:
```
KMeans()
```
In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [86]:
```
y_predicted=km.fit_predict(df[["area_mean","area_worst"]])
y_predicted
```

```
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
```

Out[86]:
```
array([4, 4, 4, 0, 4, 0, 1, 5, 5, 0, 3, 3, 1, 3, 5, 5, 3, 3, 4, 5, 0, 6,
       3, 2, 4, 1, 5, 1, 3, 1, 1, 5, 1, 4, 3, 3, 5, 0, 5, 5, 5, 0, 1, 5,
       5, 1, 6, 5, 0, 5, 0, 5, 0, 1, 3, 0, 4, 3, 0, 6, 6, 6, 5, 6, 5, 5,
       6, 0, 6, 0, 4, 6, 1, 5, 0, 3, 5, 1, 4, 0, 0, 0, 2, 1, 0, 1, 5, 1,
       0, 5, 5, 5, 5, 5, 3, 4, 0, 6, 0, 5, 5, 6, 0, 6, 6, 5, 0, 0, 2, 0,
       6, 0, 5, 6, 6, 0, 6, 3, 3, 1, 0, 1, 2, 5, 5, 5, 5, 1, 5, 4, 0, 3,
       3, 3, 1, 0, 0, 0, 3, 0, 6, 3, 0, 0, 6, 0, 0, 5, 5, 5, 0, 6, 6, 0,
       5, 0, 1, 3, 0, 0, 0, 1, 4, 0, 2, 5, 6, 3, 1, 5, 0, 5, 3, 6, 6, 6,
       6, 3, 0, 0, 7, 4, 3, 0, 3, 6, 1, 0, 0, 0, 5, 0, 6, 5, 5, 0, 5, 1,
       4, 3, 0, 1, 2, 3, 0, 3, 6, 3, 0, 3, 4, 0, 7, 3, 5, 5, 0, 6, 4, 4,
       5, 5, 6, 3, 5, 5, 6, 5, 0, 5, 3, 0, 0, 4, 6, 5, 2, 4, 5, 1, 5, 0,
       0, 5, 1, 6, 0, 0, 6, 0, 4, 0, 4, 1, 4, 5, 4, 3, 3, 3, 4, 1, 1, 3,
       1, 2, 6, 5, 0, 6, 5, 0, 2, 6, 1, 0, 0, 1, 5, 5, 4, 0, 1, 3, 0, 0,
       0, 0, 0, 0, 5, 5, 0, 0, 0, 5, 6, 0, 5, 6, 4, 0, 4, 6, 0, 0, 5, 6,
       5, 5, 0, 5, 0, 0, 6, 0, 0, 1, 6, 0, 6, 4, 0, 4, 0, 0, 5, 0, 3, 3,
       3, 0, 0, 0, 0, 1, 0, 4, 6, 2, 5, 6, 0, 1, 0, 6, 0, 5, 0, 0, 0, 3,
       7, 3, 0, 0, 0, 5, 6, 6, 0, 5, 0, 3, 5, 4, 4, 0, 2, 2, 3, 5, 4, 4,
       5, 3, 6, 5, 5, 0, 0, 0, 0, 0, 5, 5, 0, 5, 0, 1, 6, 6, 3, 4, 0, 5,
       5, 0, 0, 0, 1, 0, 0, 0, 0, 0, 3, 0, 1, 0, 0, 0, 6, 5, 3, 0, 6, 1,
       0, 0, 0, 5, 0, 5, 6, 6, 6, 0, 0, 0, 5, 0, 1, 1, 5, 5, 0, 5, 5, 5,
       0, 1, 5, 6, 1, 0, 1, 5, 5, 4, 0, 1, 0, 5, 0, 5, 0, 5, 0, 6, 1, 7,
       5, 0, 5, 5, 5, 6, 1, 0, 6, 0, 5, 0, 6, 0, 5, 5, 0, 3, 0, 5, 5, 5,
       3, 0, 5, 4, 0, 3, 0, 1, 1, 0, 5, 5, 0, 0, 1, 4, 5, 5, 0, 2, 6, 6,
       0, 6, 3, 3, 0, 5, 5, 5, 3, 0, 1, 4, 0, 0, 6, 2, 0, 5, 6, 6, 5, 0,
       5, 0, 0, 0, 5, 4, 6, 4, 5, 0, 6, 6, 0, 5, 5, 5, 5, 5, 6, 6, 6, 0,
       6, 0, 0, 6, 0, 6, 6, 6, 5, 0, 5, 0, 3, 4, 4, 4, 3, 4, 6])
```

In [87]:
```
df["New cluster"]=y_predicted
df.head()
```

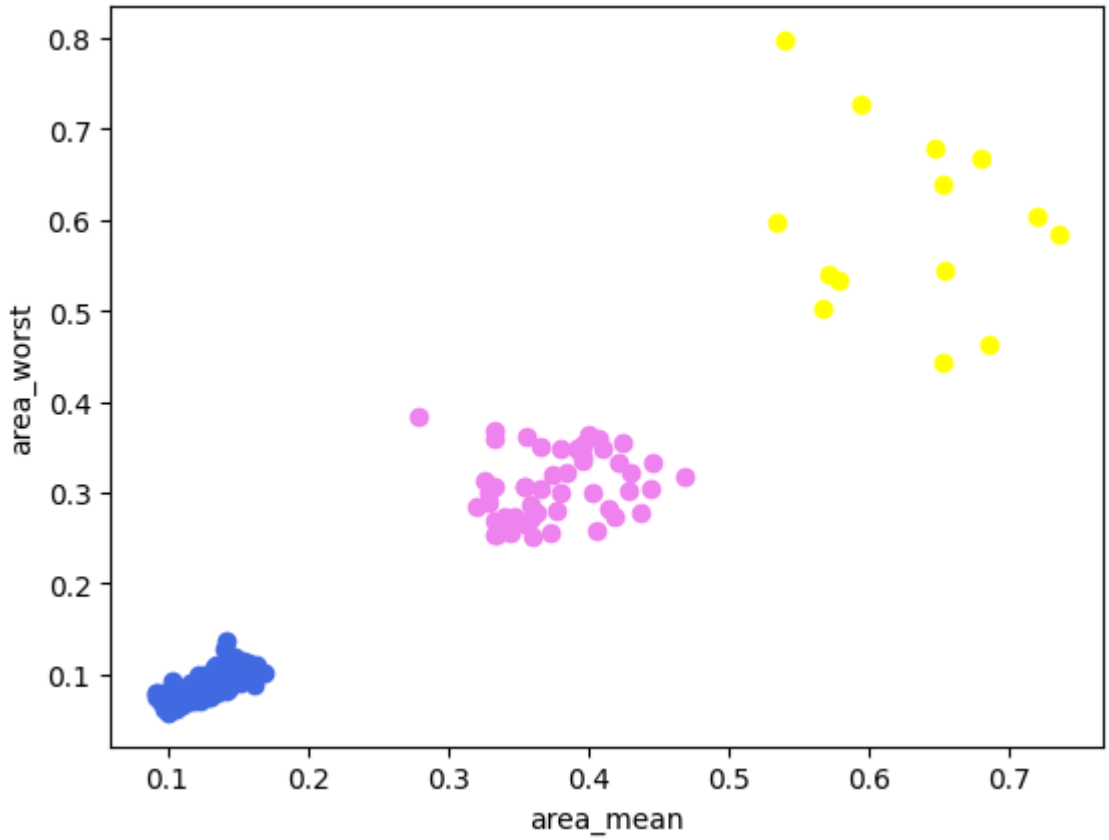
Out[87]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_m
0	842302	M	17.99	10.38	122.80	0.363733	0.1
1	842517	M	20.57	17.77	132.90	0.501591	0.0
2	84300903	M	19.69	21.25	130.00	0.449417	0.1
3	84348301	M	11.42	20.38	77.58	0.102906	0.1
4	84358402	M	20.29	14.34	135.10	0.489290	0.1

5 rows × 35 columns

In [88]:
```
df1=df[df["New cluster"]==0]
df2=df[df["New cluster"]==1]
df3=df[df["New cluster"]==2]
plt.scatter(df1["area_mean"],df1["area_worst"],color="royalblue")
plt.scatter(df2["area_mean"],df2["area_worst"],color="violet")
plt.scatter(df3["area_mean"],df3["area_worst"],color="yellow")
plt.xlabel("area_mean")
plt.ylabel("area_worst")
```

Out[88]:
```
Text(0, 0.5, 'area_worst')
```

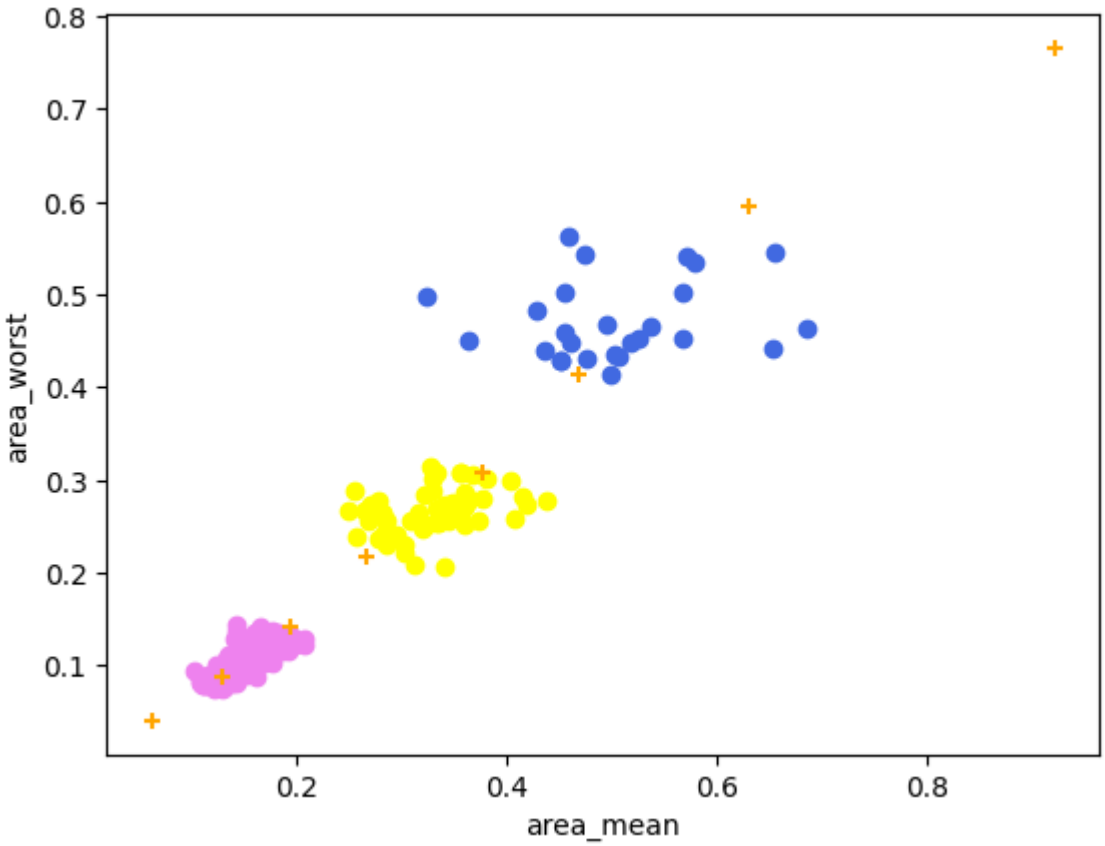


In [89]:
```
km.cluster_centers_
```

Out[89]:
```
array([[0.12862535, 0.08770433],
       [0.3759587 , 0.30821011],
       [0.62996516, 0.5945277 ],
       [0.26689363, 0.21800166],
       [0.4679017 , 0.41445186],
       [0.19468074, 0.14149153],
       [0.06191901, 0.0402961 ],
       [0.92110286, 0.76571716]])
```

In [90]:
```
df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]
plt.scatter(df1["area_mean"],df1["area_worst"],color="royalblue")
plt.scatter(df2["area_mean"],df2["area_worst"],color="violet")
plt.scatter(df3["area_mean"],df3["area_worst"],color="yellow")
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color="orange",m
plt.xlabel("area_mean")
plt.ylabel("area_worst")
```

Out[90]:
```
Text(0, 0.5, 'area_worst')
```



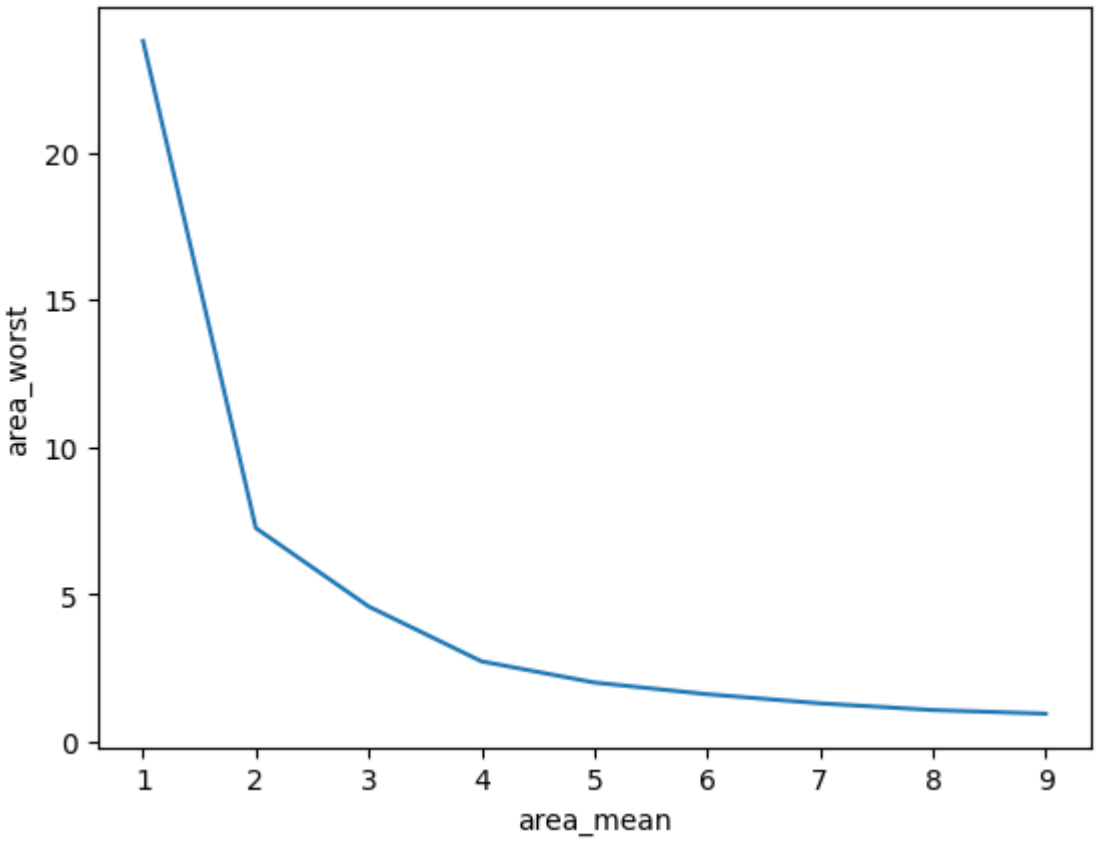
In [91]:
```
k_rng=range(1,10)
sse=[]
for k in k_rng:
    km=KMeans(n_clusters=k)
    km.fit(df[["area_mean","area_worst"]])
    sse.append(km.inertia_)
sse
```

```
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to supp
ress the warning
  warnings.warn(
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to supp
ress the warning
  warnings.warn(
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to supp
ress the warning
  warnings.warn(
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to supp
ress the warning
  warnings.warn(
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to supp
ress the warning
  warnings.warn(
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to supp
ress the warning
  warnings.warn(
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to supp
ress the warning
  warnings.warn(
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to supp
ress the warning
  warnings.warn(
C:\Users\rubin\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` will
change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to supp
ress the warning
  warnings.warn(
```

Out[91]:
```
[23.778690666252164,
 7.245561269117197,
 4.58540152843407,
 2.7231942409326817,
 2.004438108506093,
 1.6083909296655703,
 1.2982826386788158,
 1.0706268479634247,
 0.9476249403025575]
```

In [92]:
```
plt.plot(k_rng,sse)
plt.xlabel("area_mean")
plt.ylabel("area_worst")
```

Out[92]:
```
Text(0, 0.5, 'area_worst')
```



Conclusion:

For the given dataset we have done KMeans cluster model and the data is categorized into groups.

In [ ]: