initial_eda

June 13, 2024

[17]:
```
import os
import csv
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[18]:
```
data_path = '../data.csv'
```

[19]:
```
# load the data to understand its structure
data = pd.read_csv(data_path)
data.head(), data.info(), data.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 33 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   id                       569 non-null    int64
 1   diagnosis                569 non-null    object
 2   radius_mean              569 non-null    float64
 3   texture_mean             569 non-null    float64
 4   perimeter_mean           569 non-null    float64
 5   area_mean                569 non-null    float64
 6   smoothness_mean          569 non-null    float64
 7   compactness_mean         569 non-null    float64
 8   concavity_mean           569 non-null    float64
 9   concave points_mean      569 non-null    float64
 10  symmetry_mean            569 non-null    float64
 11  fractal_dimension_mean   569 non-null    float64
 12  radius_se                569 non-null    float64
 13  texture_se               569 non-null    float64
 14  perimeter_se             569 non-null    float64
 15  area_se                  569 non-null    float64
 16  smoothness_se            569 non-null    float64
 17  compactness_se           569 non-null    float64
 18  concavity_se             569 non-null    float64
 19  concave points_se        569 non-null    float64
```

```
 20  symmetry_se              569 non-null    float64
 21  fractal_dimension_se     569 non-null    float64
 22  radius_worst             569 non-null    float64
 23  texture_worst            569 non-null    float64
 24  perimeter_worst          569 non-null    float64
 25  area_worst               569 non-null    float64
 26  smoothness_worst         569 non-null    float64
 27  compactness_worst        569 non-null    float64
 28  concavity_worst          569 non-null    float64
 29  concave points_worst     569 non-null    float64
 30  symmetry_worst           569 non-null    float64
 31  fractal_dimension_worst  569 non-null    float64
 32  Unnamed: 32              0 non-null      float64
dtypes: float64(31), int64(1), object(1)
memory usage: 146.8+ KB
```

[19]: (

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

	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	\
0	0.11840	0.27760	0.3001	0.14710	
1	0.08474	0.07864	0.0869	0.07017	
2	0.10960	0.15990	0.1974	0.12790	
3	0.14250	0.28390	0.2414	0.10520	
4	0.10030	0.13280	0.1980	0.10430	

	…	texture_worst	perimeter_worst	area_worst	smoothness_worst	\
0	…	17.33	184.60	2019.0	0.1622	
1	…	23.41	158.80	1956.0	0.1238	
2	…	25.53	152.50	1709.0	0.1444	
3	…	26.50	98.87	567.7	0.2098	
4	…	16.67	152.20	1575.0	0.1374	

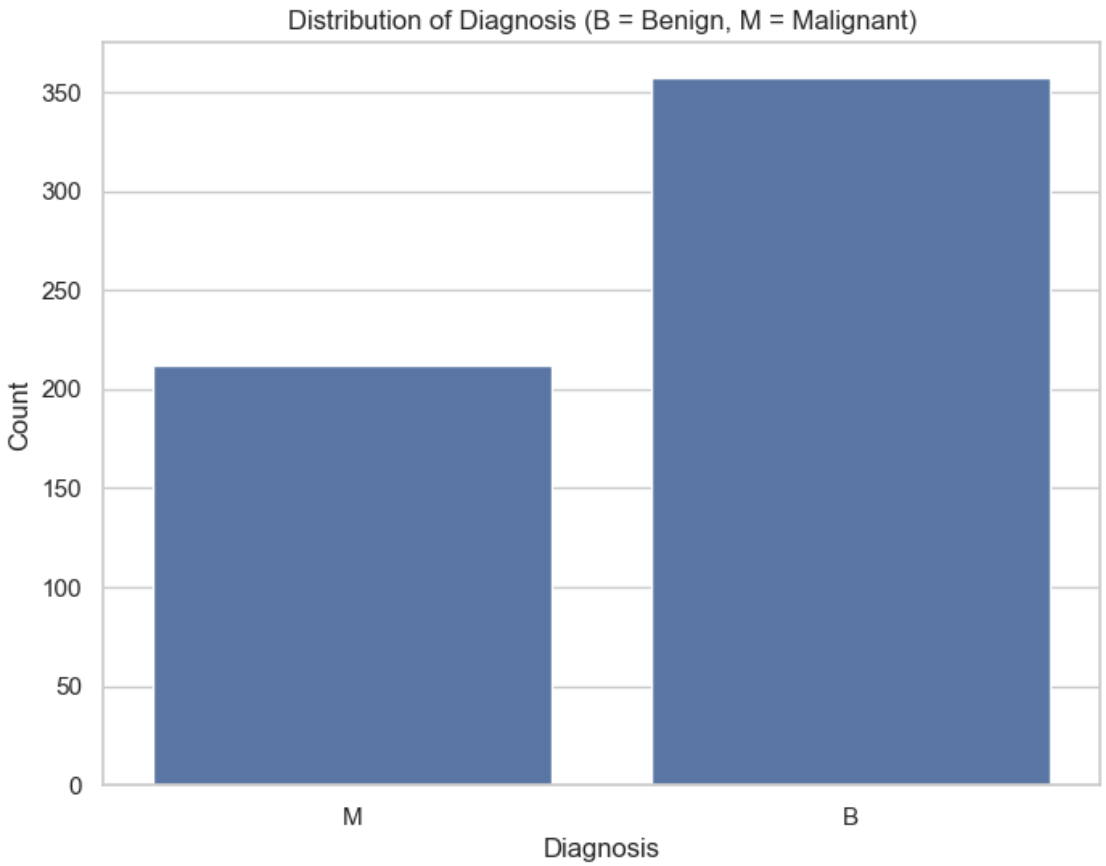
	compactness_worst	concavity_worst	concave points_worst	symmetry_worst	\
0	0.6656	0.7119	0.2654	0.4601	
1	0.1866	0.2416	0.1860	0.2750	
2	0.4245	0.4504	0.2430	0.3613	
3	0.8663	0.6869	0.2575	0.6638	
4	0.2050	0.4000	0.1625	0.2364	

	fractal_dimension_worst	Unnamed: 32
0	0.11890	NaN
1	0.08902	NaN

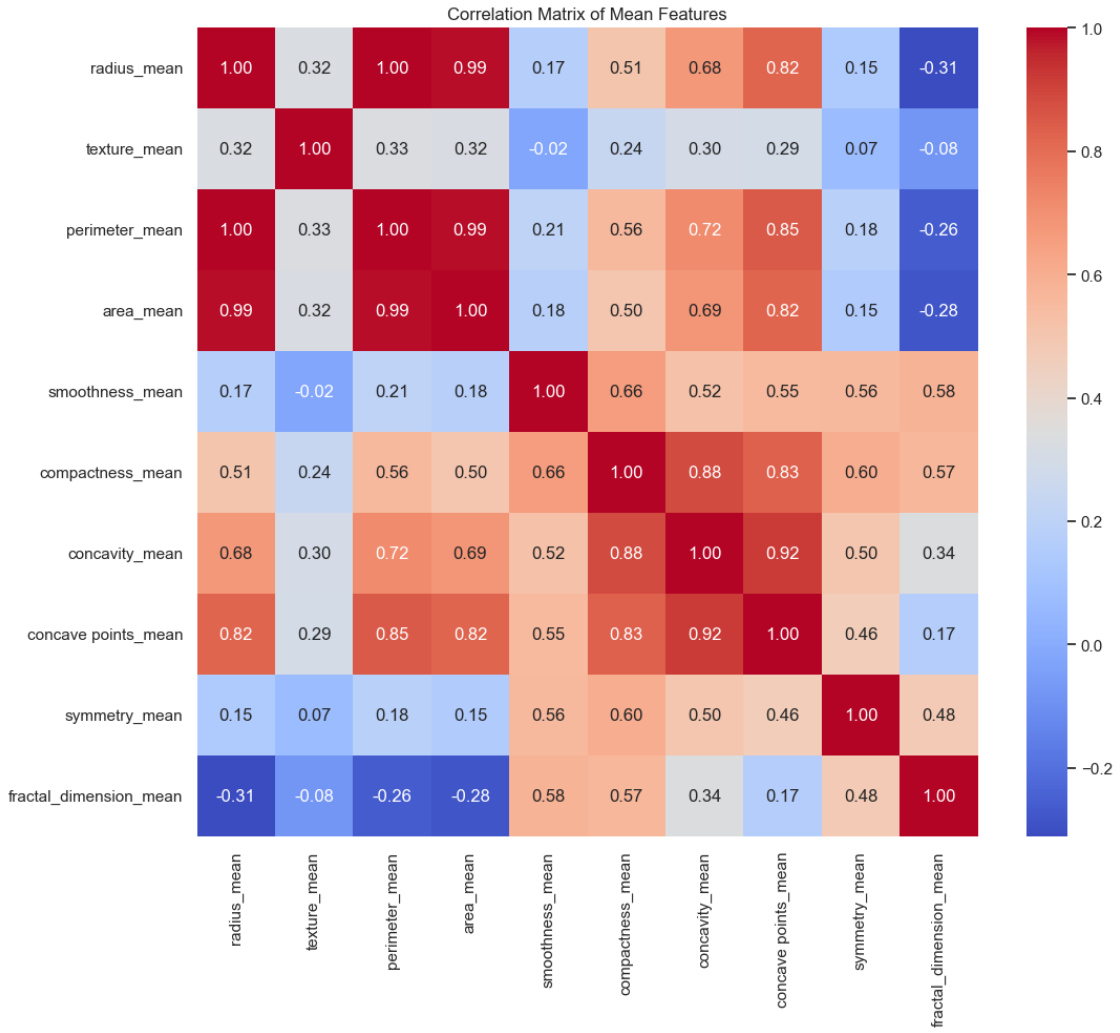
```
2                  0.08758          NaN
3                  0.17300          NaN
4                  0.07678          NaN

[5 rows x 33 columns],
None,
                 id  radius_mean  texture_mean  perimeter_mean    area_mean  \
count  5.690000e+02   569.000000    569.000000      569.000000   569.000000
mean   3.037183e+07    14.127292     19.289649       91.969033   654.889104
std    1.250206e+08     3.524049      4.301036       24.298981   351.914129
min    8.670000e+03     6.981000      9.710000       43.790000   143.500000
25%    8.692180e+05    11.700000     16.170000       75.170000   420.300000
50%    9.060240e+05    13.370000     18.840000       86.240000   551.100000
75%    8.813129e+06    15.780000     21.800000      104.100000   782.700000
max    9.113205e+08    28.110000     39.280000      188.500000  2501.000000

       smoothness_mean  compactness_mean  concavity_mean  concave points_mean
\
count       569.000000        569.000000      569.000000           569.000000
mean          0.096360          0.104341        0.088799             0.048919
std           0.014064          0.052813        0.079720             0.038803
min           0.052630          0.019380        0.000000             0.000000
25%           0.086370          0.064920        0.029560             0.020310
50%           0.095870          0.092630        0.061540             0.033500
75%           0.105300          0.130400        0.130700             0.074000
max           0.163400          0.345400        0.426800             0.201200

       symmetry_mean  …  texture_worst  perimeter_worst   area_worst  \
count     569.000000  …     569.000000       569.000000   569.000000
mean        0.181162  …      25.677223       107.261213   880.583128
std         0.027414  …       6.146258        33.602542   569.356993
min         0.106000  …      12.020000        50.410000   185.200000
25%         0.161900  …      21.080000        84.110000   515.300000
50%         0.179200  …      25.410000        97.660000   686.500000
75%         0.195700  …      29.720000       125.400000  1084.000000
max         0.304000  …      49.540000       251.200000  4254.000000

       smoothness_worst  compactness_worst  concavity_worst  \
count        569.000000         569.000000       569.000000
mean           0.132369           0.254265         0.272188
std            0.022832           0.157336         0.208624
min            0.071170           0.027290         0.000000
25%            0.116600           0.147200         0.114500
50%            0.131300           0.211900         0.226700
75%            0.146000           0.339100         0.382900
max            0.222600           1.058000         1.252000
```

```
       concave points_worst  symmetry_worst  fractal_dimension_worst  \
count            569.000000      569.000000               569.000000
mean               0.114606        0.290076                 0.083946
std                0.065732        0.061867                 0.018061
min                0.000000        0.156500                 0.055040
25%                0.064930        0.250400                 0.071460
50%                0.099930        0.282200                 0.080040
75%                0.161400        0.317900                 0.092080
max                0.291000        0.663800                 0.207500

       Unnamed: 32
count          0.0
mean           NaN
std            NaN
min            NaN
25%            NaN
50%            NaN
75%            NaN
max            NaN

[8 rows x 32 columns])
```

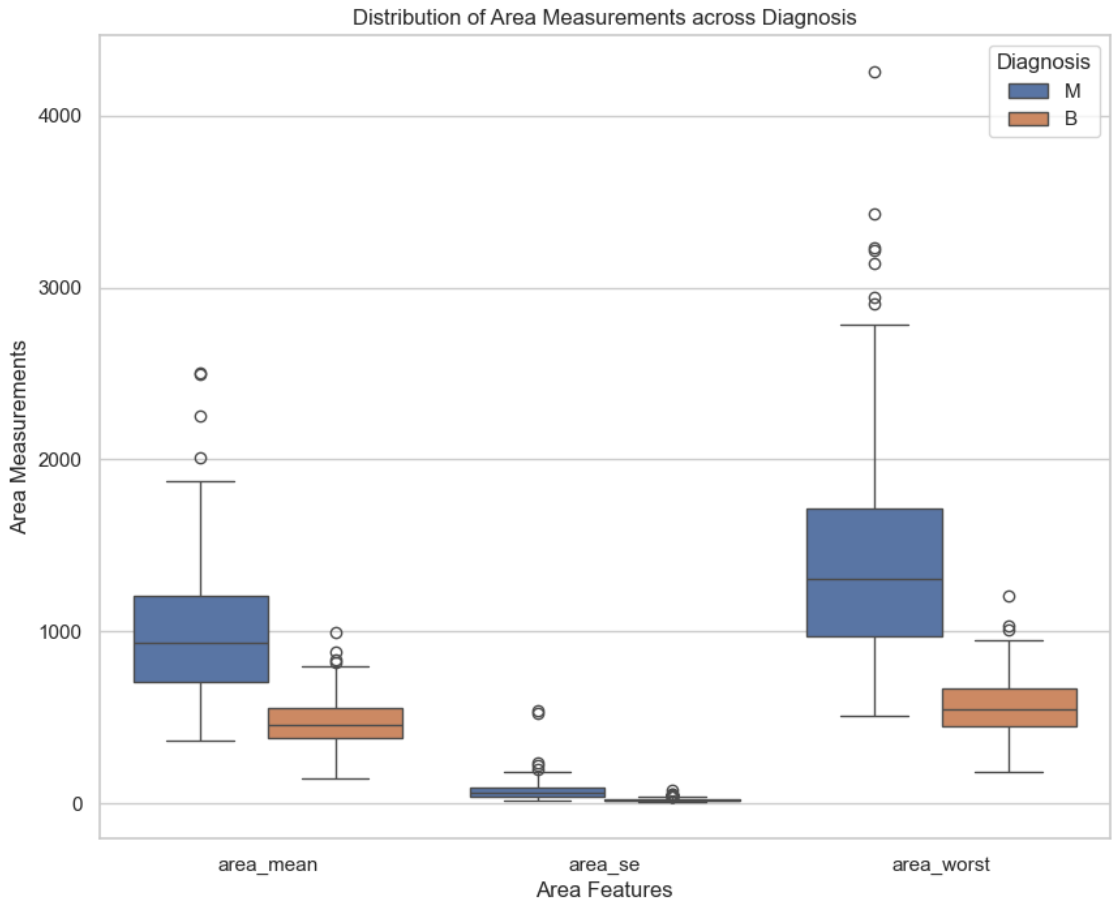
[20]:
```
# set the aesthetic style of the plots
sns.set(style="whitegrid")

# distribution of Diagnosis
plt.figure(figsize=(8, 6))
ax = sns.countplot(x='diagnosis', data=data)
ax.set_title('Distribution of Diagnosis (B = Benign, M = Malignant)')
ax.set_xlabel('Diagnosis')
ax.set_ylabel('Count')
plt.show()
```



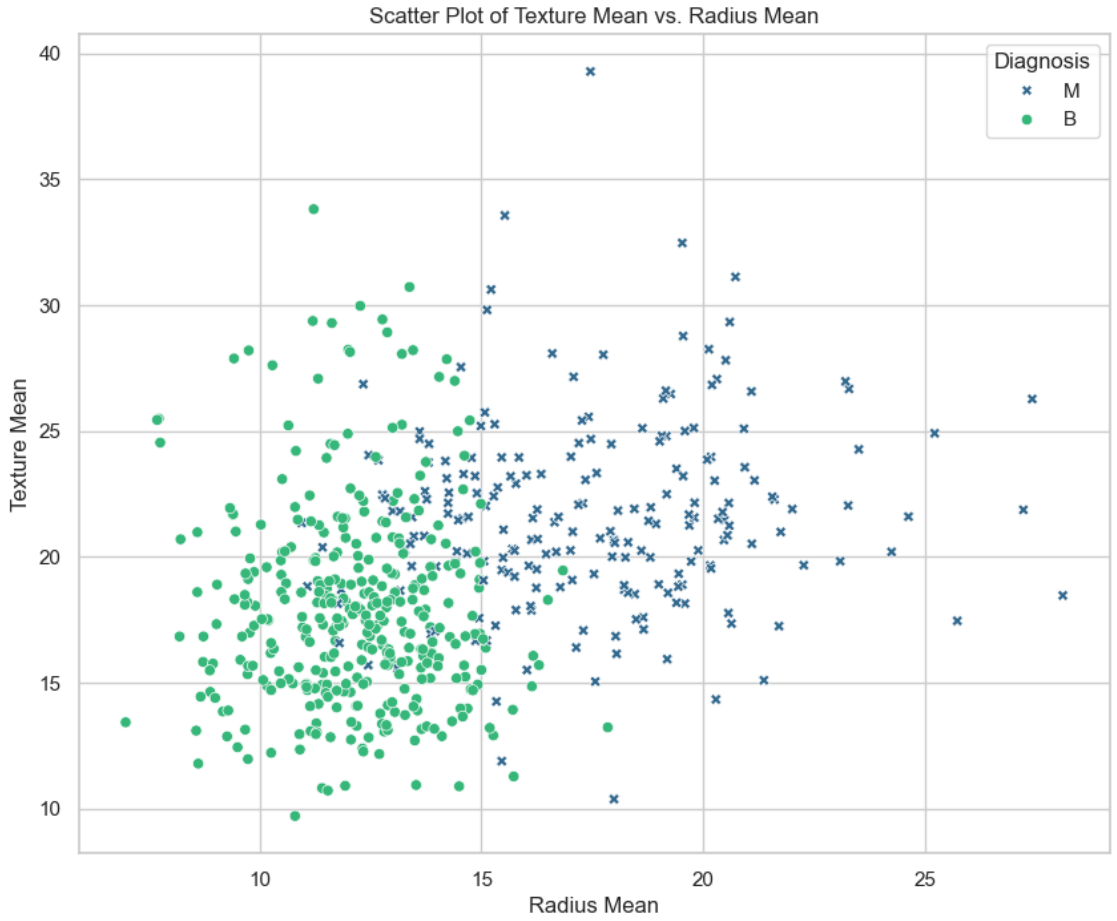
[21]:
```
mean_features = [col for col in data.columns if 'mean' in col and data[col].
 ↪dtype != 'object']
correlation_matrix = data[mean_features].corr()

plt.figure(figsize=(12, 10))
ax = sns.heatmap(correlation_matrix, annot=True, fmt=".2f", cmap='coolwarm')
ax.set_title('Correlation Matrix of Mean Features')
plt.show()
```



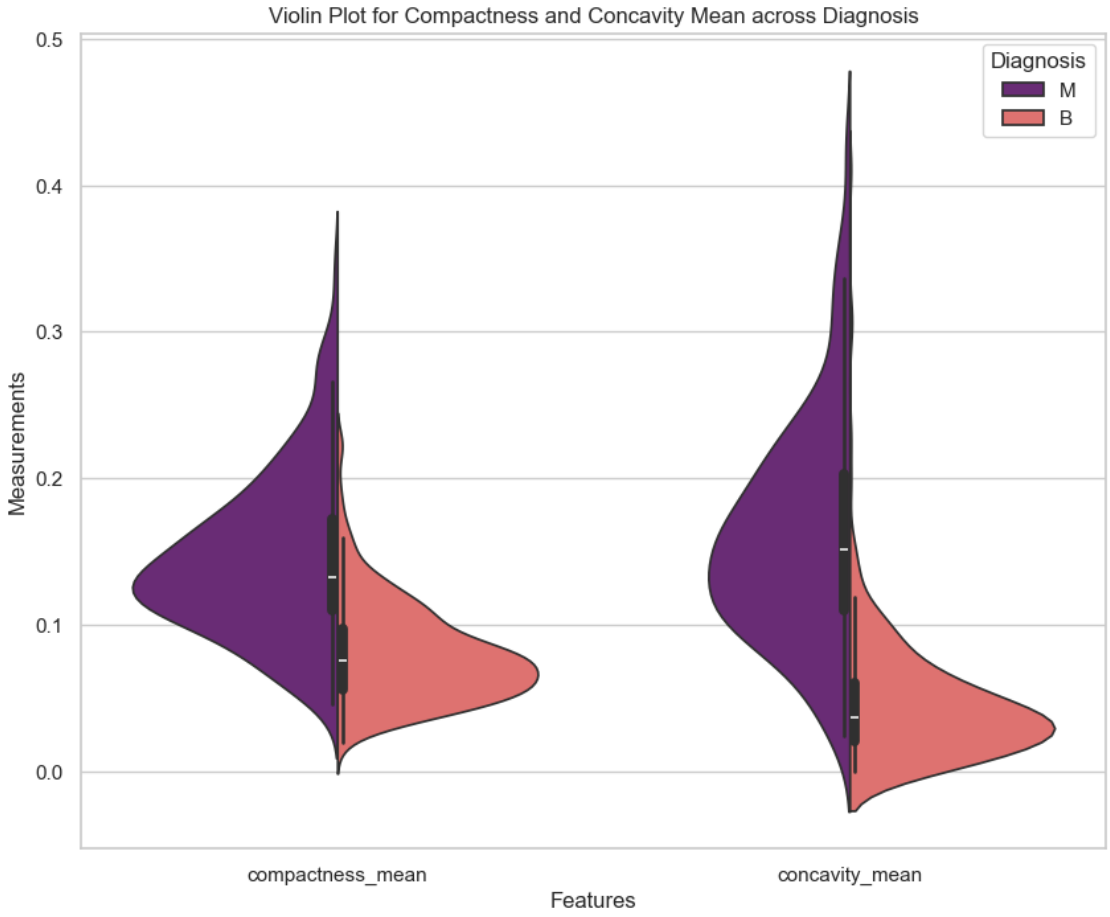
[22]:
```
# boxplot for a measurements
area_features = ['area_mean', 'area_se', 'area_worst']
melted_area_data = pd.melt(data, id_vars=['diagnosis'],␣
 ↪value_vars=area_features, var_name='Area Features', value_name='Value')

plt.figure(figsize=(10, 8))
ax = sns.boxplot(x='Area Features', y='Value', hue='diagnosis',␣
 ↪data=melted_area_data)
ax.set_title('Distribution of Area Measurements across Diagnosis')
ax.set_xlabel('Area Features')
ax.set_ylabel('Area Measurements')
plt.legend(title='Diagnosis')
plt.show()
```



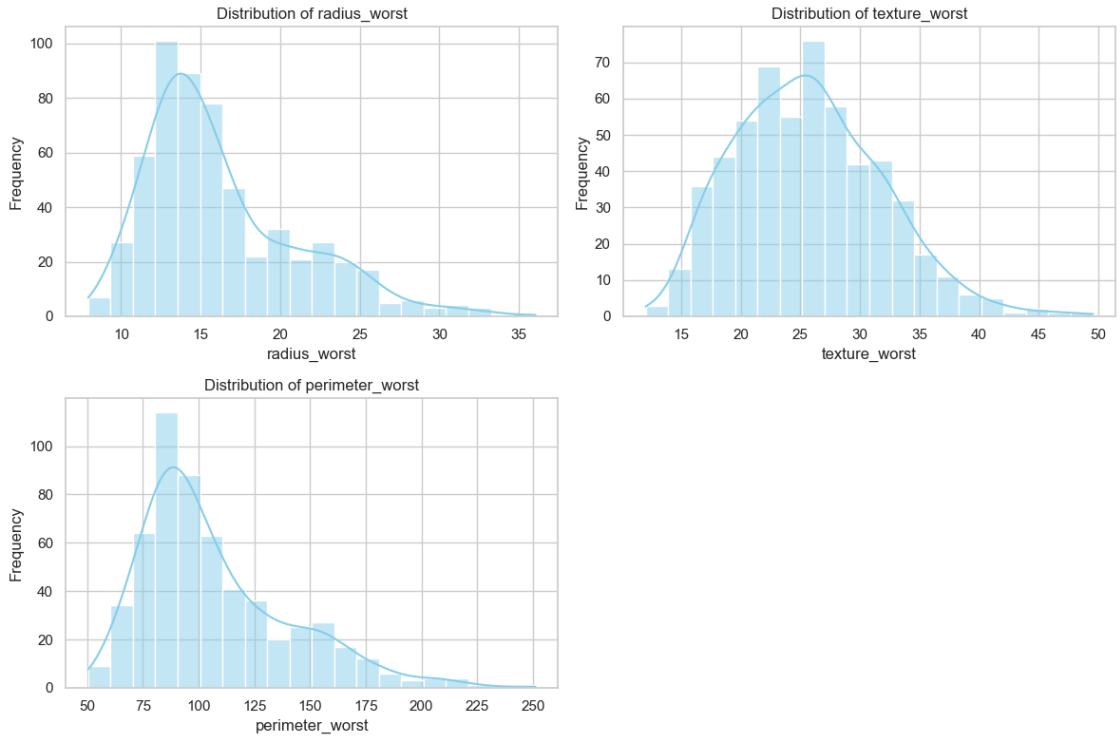
[23]:
```
# scatter plot for texture vs. radius
plt.figure(figsize=(10, 8))
ax = sns.scatterplot(x='radius_mean', y='texture_mean', hue='diagnosis',␣
 ↪data=data, style='diagnosis', markers={'B':'o', 'M':'X'}, palette='viridis')
ax.set_title('Scatter Plot of Texture Mean vs. Radius Mean')
ax.set_xlabel('Radius Mean')
ax.set_ylabel('Texture Mean')
plt.legend(title='Diagnosis')
plt.show()
```



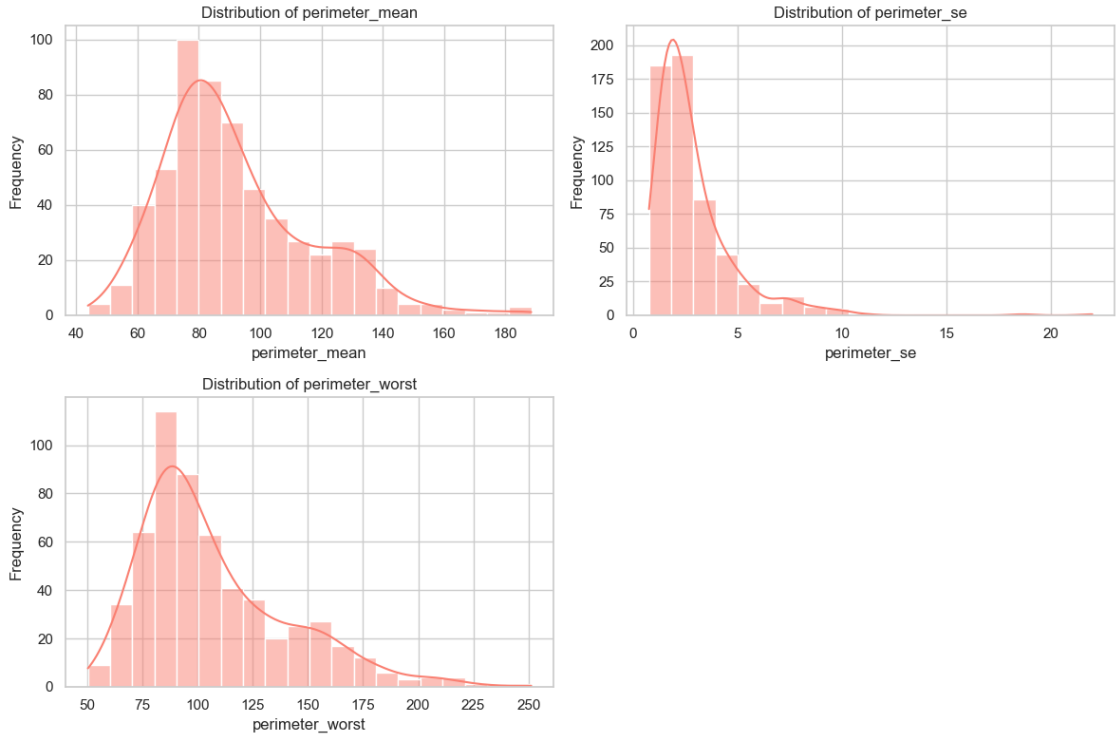
[24]:
```
# violin Plot for Compactness and Concavity
compactness_concavity_features = ['compactness_mean', 'concavity_mean']
melted_cc_data = pd.melt(data, id_vars=['diagnosis'],␣
 ↪value_vars=compactness_concavity_features, var_name='Features',␣
 ↪value_name='Value')

plt.figure(figsize=(10, 8))
ax = sns.violinplot(x='Features', y='Value', hue='diagnosis',␣
 ↪data=melted_cc_data, split=True, palette='magma')
ax.set_title('Violin Plot for Compactness and Concavity Mean across Diagnosis')
ax.set_xlabel('Features')
ax.set_ylabel('Measurements')
plt.legend(title='Diagnosis')
plt.show()
```



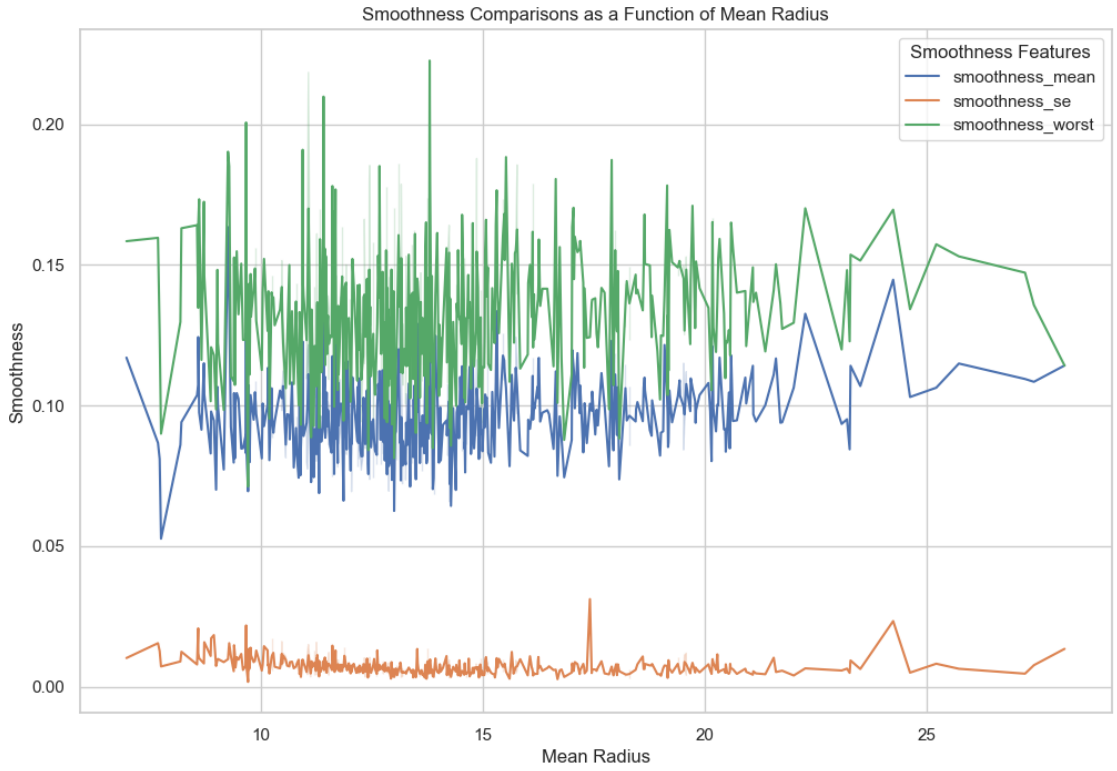
[25]:
```
# histograms of Worst Stage Features
worst_features = ['radius_worst', 'texture_worst', 'perimeter_worst']

plt.figure(figsize=(12, 8))
for i, feature in enumerate(worst_features, 1):
    plt.subplot(2, 2, i)
    sns.histplot(data[feature], bins=20, kde=True, color='skyblue')
    plt.title(f'Distribution of {feature}')
    plt.xlabel(feature)
    plt.ylabel('Frequency')

plt.tight_layout()
plt.show()
```



[26]:
```
# histogram of perimeter features
perimeter_features = ['perimeter_mean', 'perimeter_se', 'perimeter_worst']

plt.figure(figsize=(12, 8))
for i, feature in enumerate(perimeter_features, 1):
    plt.subplot(2, 2, i)
    sns.histplot(data[feature], bins=20, kde=True, color='salmon')
    plt.title(f'Distribution of {feature}')
    plt.xlabel(feature)
    plt.ylabel('Frequency')

plt.tight_layout()
plt.show()
```



[27]:
```
# line plots of smoothness comps
smoothness_features = ['smoothness_mean', 'smoothness_se', 'smoothness_worst']

plt.figure(figsize=(12, 8))
for feature in smoothness_features:
    sns.lineplot(data=data, x='radius_mean', y=feature, label=feature)

plt.title('Smoothness Comparisons as a Function of Mean Radius')
plt.xlabel('Mean Radius')
plt.ylabel('Smoothness')
plt.legend(title='Smoothness Features')
plt.show()
```


Smoothness Comparisons as a Function of Mean Radius
Smoothness Features
smoothness_mean
smoothness_se
smoothness_worst
Smoothness
0.20
0.15
0.10
0.05
0.00
10
15
20
25
Mean Radius

[ ]: