

```

In [17]: #Import pandas package as pd
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

file_url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data'
data = pd.read_csv(file_url, sep= ',', header = None)
#data = data.iloc[:, :32]
#create the features columns
features = ['ID number', 'Diagnosis', 'mean radius', 'mean texture', 'mean perimeter', 'mean area', 'mean smoothness', 'mean compactness', 'mean concavity', 'mean concave points', 'mean symmetry', 'mean fractal dimension', 'SE radius', 'SE texture', 'SE perimeter', 'SE area', 'SE smoothness', 'SE compactness', 'SE concavity', 'SE concave points', 'SE symmetry', 'SE fractal dimension', 'Worst radius', 'Worst texture', 'Worst perimeter', 'Worst area', 'Worst smoothness', 'Worst compactness', 'Worst concavity', 'Worst concave points', 'Worst symmetry', 'Worst fractal dimension']

data.columns = features
data = data.set_index('ID number')
data.head()

```

Out[17]:

	Diagnosis	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	mean fractal dimension
ID number											
842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.25748	0.46015
842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.16341	0.17538
84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.20347	0.26434
84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.26109	0.41882
84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.18091	0.27383

5 rows × 12 columns

MEAN

```
In [19]: mean_value = data.loc[:,features[2:32]].mean()  
print (mean_value)
```

```
mean radius          14.127292  
mean texture         19.289649  
mean perimeter       91.969033  
mean area           654.889104  
mean smoothness      0.096360  
mean compactness     0.104341  
mean concavity       0.088799  
mean concave points  0.048919  
mean symmetry        0.181162  
mean fractal dimension 0.062798  
SE radius            0.405172  
SE texture           1.216853  
SE perimeter         2.866059  
SE area             40.337079  
SE smoothness        0.007041  
SE compactness       0.025478  
SE concavity         0.031894  
SE concave points    0.011796  
SE symmetry          0.020542  
SE fractal dimension 0.003795  
Worst radius         16.269190  
Worst texture        25.677223  
Worst perimeter     107.261213  
Worst area          880.583128  
Worst smoothness     0.132369  
Worst compactness    0.254265  
Worst concavity      0.272188  
Worst concave points 0.114606  
Worst symmetry       0.290076  
Worst fractal dimension 0.083946  
dtype: float64
```

Mode

```
In [20]: data.loc[:,features[2:32]].mode().iloc[0,:]
```

```
Out[20]: mean radius      12.340000
mean texture    14.930000
mean perimeter  82.610000
mean area      512.200000
mean smoothness 0.100700
mean compactness 0.114700
mean concavity  0.000000
mean concave points 0.000000
mean symmetry   0.160100
mean fractal dimension 0.056670
SE radius       0.220400
SE texture      0.856100
SE perimeter    1.778000
SE area         16.640000
SE smoothness   0.005080
SE compactness  0.011040
SE concavity    0.000000
SE concave points 0.000000
SE symmetry     0.013440
SE fractal dimension 0.001784
Worst radius    12.360000
Worst texture   17.700000
Worst perimeter 101.700000
Worst area      284.400000
Worst smoothness 0.121600
Worst compactness 0.148600
Worst concavity 0.000000
Worst concave points 0.000000
Worst symmetry  0.222600
Worst fractal dimension 0.074270
Name: 0, dtype: float64
```

skew

```
In [21]: data.loc[:, features[2:32]].skew()
```

```
Out[21]: mean radius      0.942380
         mean texture    0.650450
         mean perimeter  0.990650
         mean area       1.645732
         mean smoothness 0.456324
         mean compactness 1.190123
         mean concavity   1.401180
         mean concave points 1.171180
         mean symmetry    0.725609
         mean fractal dimension 1.304489
         SE radius        3.088612
         SE texture       1.646444
         SE perimeter     3.443615
         SE area          5.447186
         SE smoothness    2.314450
         SE compactness   1.902221
         SE concavity     5.110463
         SE concave points 1.444678
         SE symmetry      2.195133
         SE fractal dimension 3.923969
         Worst radius     1.103115
         Worst texture    0.498321
         Worst perimeter  1.128164
         Worst area       1.859373
         Worst smoothness 0.415426
         Worst compactness 1.473555
         Worst concavity  1.150237
         Worst concave points 0.492616
         Worst symmetry    1.433928
         Worst fractal dimension 1.662579
         dtype: float64
```

Standard Deviation

```
In [22]: data.loc[:, features[2:32]].std()
```

```
Out[22]: mean radius          3.524049
         mean texture         4.301036
         mean perimeter       24.298981
         mean area           351.914129
         mean smoothness      0.014064
         mean compactness     0.052813
         mean concavity       0.079720
         mean concave points  0.038803
         mean symmetry        0.027414
         mean fractal dimension 0.007060
         SE radius            0.277313
         SE texture           0.551648
         SE perimeter         2.021855
         SE area              45.491006
         SE smoothness        0.003003
         SE compactness       0.017908
         SE concavity         0.030186
         SE concave points    0.006170
         SE symmetry          0.008266
         SE fractal dimension 0.002646
         Worst radius         4.833242
         Worst texture        6.146258
         Worst perimeter      33.602542
         Worst area           569.356993
         Worst smoothness     0.022832
         Worst compactness    0.157336
         Worst concavity      0.208624
         Worst concave points 0.065732
         Worst symmetry        0.061867
         Worst fractal dimension 0.018061
         dtype: float64
```

Variance

```
In [23]: data.loc[:, features[2:32]].var()
```

```
Out[23]: mean radius          12.418920
mean texture          18.498909
mean perimeter        590.440480
mean area            123843.554318
mean smoothness        0.000198
mean compactness        0.002789
mean concavity          0.006355
mean concave points     0.001506
mean symmetry           0.000752
mean fractal dimension  0.000050
SE radius              0.076902
SE texture              0.304316
SE perimeter            4.087896
SE area                2069.431583
SE smoothness           0.000009
SE compactness           0.000321
SE concavity             0.000911
SE concave points        0.000038
SE symmetry              0.000068
SE fractal dimension     0.000007
Worst radius            23.360224
Worst texture            37.776483
Worst perimeter         1129.130847
Worst area              324167.385102
Worst smoothness         0.000521
Worst compactness        0.024755
Worst concavity           0.043524
Worst concave points     0.004321
Worst symmetry           0.003828
Worst fractal dimension  0.000326
dtype: float64
```

Correlations of features using PCC

```
In [ ]: correlations = data.drop('Diagnosis', 1).corr(method='pearson')
display(correlations)
```

```
In [25]: correlations_flat = correlations.where(np.triu(np.ones(correlations.shape)).astype(np.bool))
correlations_flat = correlations_flat.stack().reset_index()
correlations_flat.columns = ['row', 'col', 'value']
correlations_flat.loc[correlations_flat['row'] == correlations_flat['col'], 'value'] = np.nan
correlations_flat = correlations_flat.sort_values(by=['value'], ascending=False).dropna()
display(correlations_flat)
```

```
-----
NameError                                Traceback (most recent call last)
<ipython-input-25-3c5b5cd3a64c> in <module>
----> 1 correlations_flat = correlations.where(np.triu(np.ones(correlations.shape)).astype(np.bool))
      2 correlations_flat = correlations_flat.stack().reset_index()
      3 correlations_flat.columns = ['row', 'col', 'value']
      4 correlations_flat.loc[correlations_flat['row'] == correlations_flat['col'], 'value'] = np.nan
      5 correlations_flat = correlations_flat.sort_values(by=['value'], ascending=False).dropna()
```

NameError: name 'correlations' is not defined

```
In [ ]: plt.hist= data.hist(column = 'mean perimeter',by = 'Diagnosis')

plt.xlabel('perimeter')
plt.ylabel('count')
plt.show()
```

In []:

In []: