

In [8]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as py
import seaborn as sns
```

In [9]:

```
d=pd.read_csv(r"C:\Users\user\Downloads\8_BreastCancerPrediction - 8_BreastCancerPredic
d
```

Out[9]:

	<b>id</b>	<b>diagnosis</b>	<b>radius_mean</b>	<b>texture_mean</b>	<b>perimeter_mean</b>	<b>area_mean</b>	<b>smoothness_mean</b>	<b>compactness_mean</b>	<b>concavity_mean</b>	<b>concave_points_mean</b>	<b>symmetry_mean</b>	<b>fractal_dimension_mean</b>	<b>radius_se</b>	<b>texture_se</b>	<b>perimeter_se</b>	<b>area_se</b>	<b>smoothness_se</b>	<b>compactness_se</b>	<b>concavity_se</b>	<b>concave_points_se</b>	<b>symmetry_se</b>	<b>fractal_dimension_se</b>	<b>radius_worst</b>	<b>texture_worst</b>	<b>perimeter_worst</b>	<b>area_worst</b>	<b>smoothness_worst</b>	<b>compactness_worst</b>	<b>concavity_worst</b>	<b>concave_points_worst</b>	<b>symmetry_worst</b>	<b>fractal_dimension_worst</b>
<b>0</b>	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	
<b>1</b>	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		
<b>2</b>	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		
<b>3</b>	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000			
<b>4</b>	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000			
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...			
<b>564</b>	926424	M	21.56	22.39	142.00	1479.0	0.11100	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		
<b>565</b>	926682	M	20.13	28.25	131.20	1261.0	0.09780	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		
<b>566</b>	926954	M	16.60	28.08	108.30	858.1	0.08455	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		
<b>567</b>	927241	M	20.60	29.33	140.10	1265.0	0.11780	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000			
<b>568</b>	92751	B	7.76	24.54	47.92	181.0	0.05263	0.12030	0.08520	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000		

569 rows × 32 columns

In [10]:

```
d.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 32 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
```

```
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
dtypes: float64(30), int64(1), object(1)
memory usage: 142.4+ KB
```

In [11]:

```
d.head()
```

Out[11]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	cor
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	

5 rows × 32 columns



In [12]:

```
d.describe()
```

Out[12]:

	id	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	com
count	5.690000e+02	569.000000	569.000000	569.000000	569.000000	569.000000	
mean	3.037183e+07	14.127292	19.289649	91.969033	654.889104	0.096360	
std	1.250206e+08	3.524049	4.301036	24.298981	351.914129	0.014064	
min	8.670000e+03	6.981000	9.710000	43.790000	143.500000	0.052630	
25%	8.692180e+05	11.700000	16.170000	75.170000	420.300000	0.086370	
50%	9.060240e+05	13.370000	18.840000	86.240000	551.100000	0.095870	
75%	8.813129e+06	15.780000	21.800000	104.100000	782.700000	0.105300	
max	9.113205e+08	28.110000	39.280000	188.500000	2501.000000	0.163400	

8 rows × 31 columns



In [13]:

#### d. columns

```
Out[13]: 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'],
      dtype='object')
```

In [14]:

d.index

Out[14]: RangeIndex(start=0, stop=569, step=1)

In [15]:

```
d=d.head(100)
```

d

Out[15]:

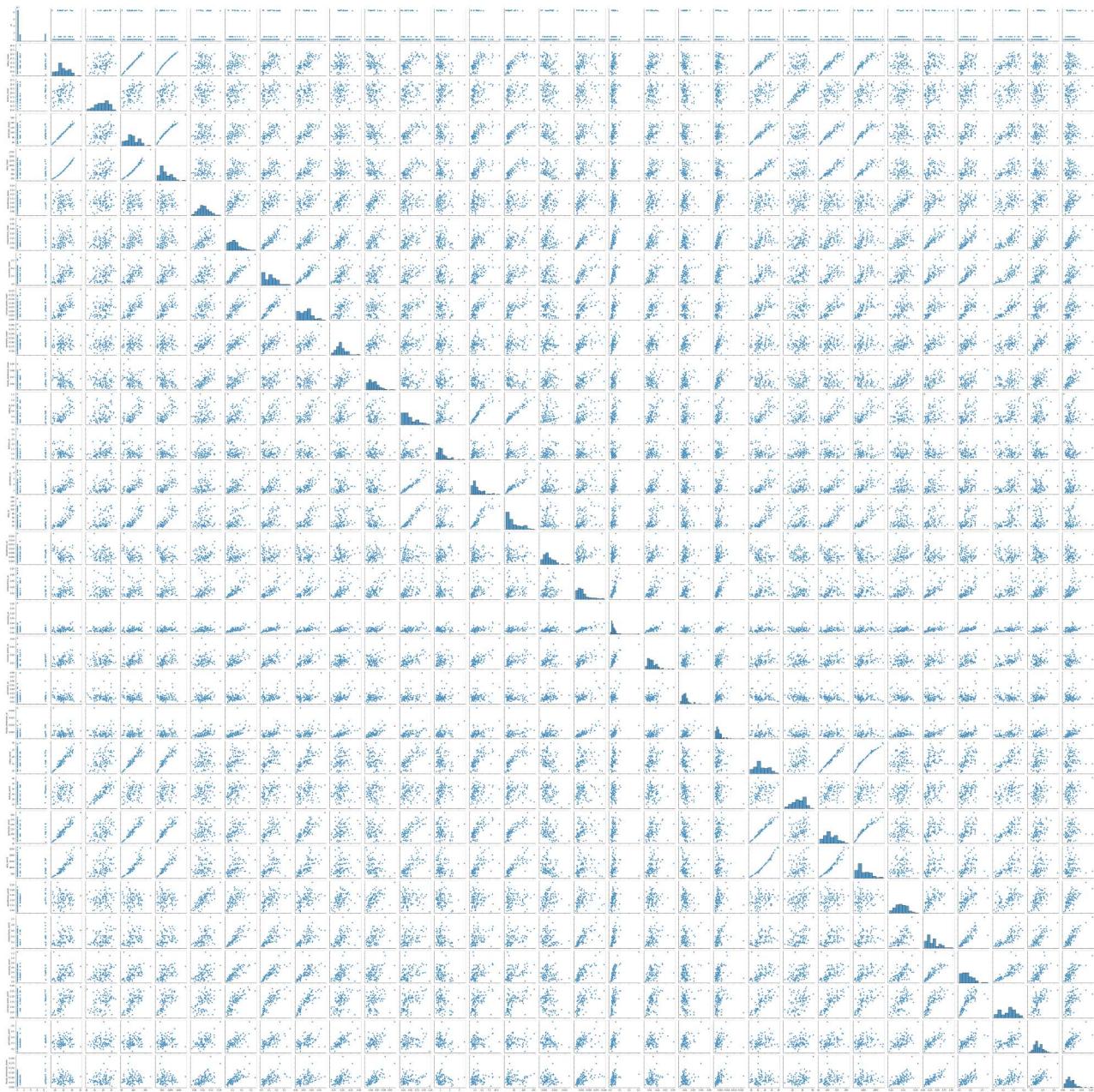
<b>id</b>	<b>diagnosis</b>	<b>radius_mean</b>	<b>texture_mean</b>	<b>perimeter_mean</b>	<b>area_mean</b>	<b>smoothness_mean</b>	<b>compactness_mean</b>
302	M	17.990	10.38	122.80	1001.0	0.11840	0.09078
517	M	20.570	17.77	132.90	1326.0	0.08474	0.10450
903	M	19.690	21.25	130.00	1203.0	0.10960	0.10240
301	M	11.420	20.38	77.58	386.1	0.14250	0.08983
402	M	20.290	14.34	135.10	1297.0	0.10030	0.09752
...	...	...	...	...	...	...	...
208	M	20.260	23.03	132.40	1264.0	0.09078	0.10450
211	B	12.180	17.84	77.79	451.1	0.10450	0.10240
261	B	9.787	19.94	62.11	294.5	0.10240	0.09752
485	B	11.600	12.84	74.34	412.6	0.08983	0.09078
548	M	14.420	19.77	94.48	642.5	0.09752	0.09078

100 rows × 32 columns

In [16]:

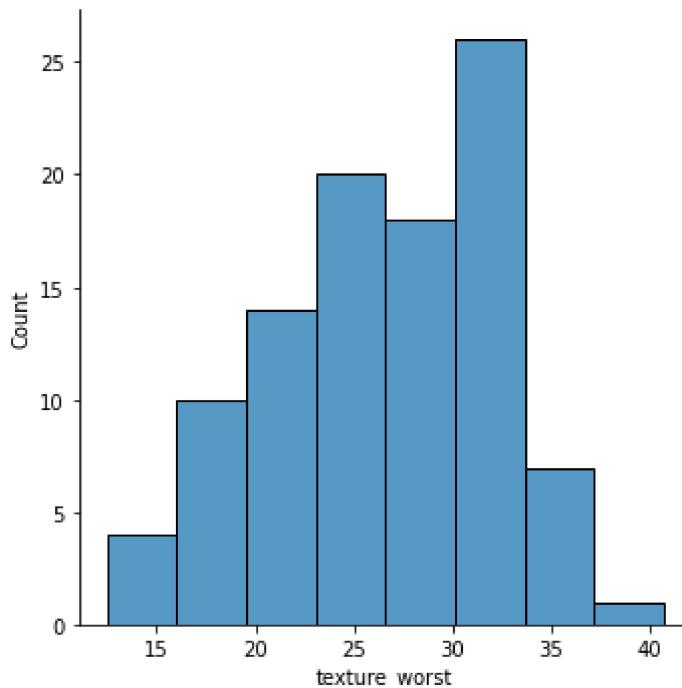
```
sns.pairplot(d)
```

```
Out[16]: <seaborn.axisgrid.PairGrid at 0x1540ca477c0>
```



```
In [17]: sns.displot(d['texture_worst'])
```

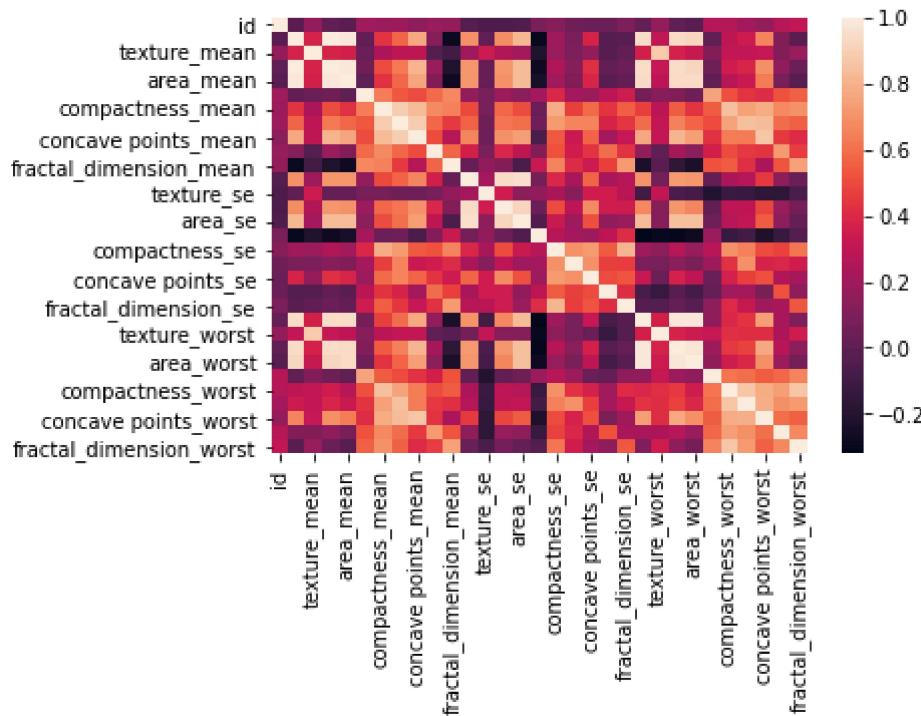
```
Out[17]: <seaborn.axisgrid.FacetGrid at 0x1542b18a520>
```



In [23]:

```
d1=d[['id','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']]
sns.heatmap(d1.corr())
```

Out[23]: <AxesSubplot:>



In [25]:

```
x=d1[['id','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',
       'perimeter_worst', 'area_worst', 'smoothness_worst',
       'compactness_worst', 'concavity_worst', 'concave points_worst',
       'symmetry_worst', 'fractal_dimension_worst']]
y=d1['texture_worst']
```

In [26]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3)
```

In [27]:

```
from sklearn.linear_model import LinearRegression
```

In [28]:

```
lr=LinearRegression()
lr.fit(x_train,y_train)
```

Out[28]: LinearRegression()

In [29]:

```
print(lr.intercept_)
```

15.57918838889606

In [30]:

```
coeff =pd.DataFrame(lr.coef_,x.columns,columns=["Co-efficient"])
coeff
```

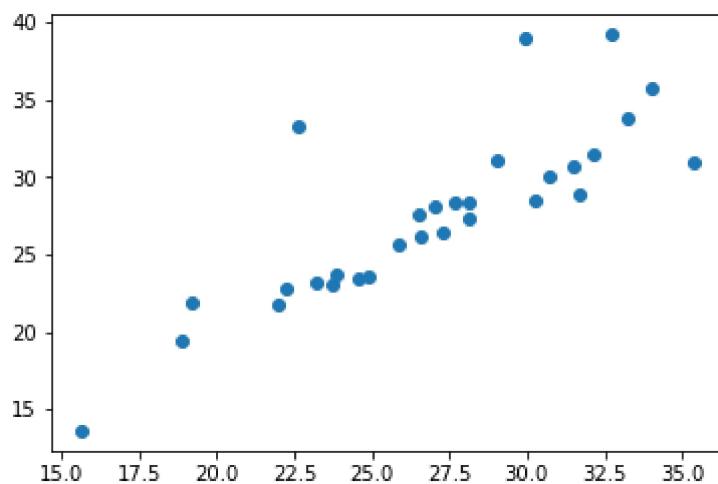
Out[30]:

	Co-efficient
<b>id</b>	-4.483739e-09
<b>radius_mean</b>	-3.394658e+00
<b>texture_mean</b>	1.024749e+00
<b>perimeter_mean</b>	4.687125e-01
<b>area_mean</b>	-4.733478e-03
<b>smoothness_mean</b>	-5.076405e+01
<b>compactness_mean</b>	2.216295e+01
<b>concavity_mean</b>	-3.241473e+01
<b>concave points_mean</b>	5.948617e+01
<b>symmetry_mean</b>	-1.119621e+01
<b>fractal_dimension_mean</b>	-2.385983e+02
<b>radius_se</b>	-6.417448e+00
<b>texture_se</b>	4.900662e+00

	Co-efficient
<b>perimeter_se</b>	7.397472e-01
<b>area_se</b>	-3.284273e-02
<b>smoothness_se</b>	-1.021574e+02
<b>compactness_se</b>	-1.941156e+01
<b>concavity_se</b>	1.066401e+02
<b>concave points_se</b>	-3.137977e+02
<b>symmetry_se</b>	-1.382181e+02
<b>fractal_dimension_se</b>	2.390458e+02
<b>radius_worst</b>	1.648708e-01
<b>perimeter_worst</b>	-4.117118e-02
<b>area_worst</b>	5.504161e-03
<b>smoothness_worst</b>	1.857311e+01
<b>compactness_worst</b>	-6.694474e+00
<b>concavity_worst</b>	-5.608466e+00
<b>concave points_worst</b>	1.833893e+01
<b>symmetry_worst</b>	2.399556e+01
<b>fractal_dimension_worst</b>	7.000838e+01

```
In [31]: prediction = lr.predict(x_test)
py.scatter(y_test,prediction)
```

```
Out[31]: <matplotlib.collections.PathCollection at 0x1543aab0970>
```



```
In [32]: print(lr.score(x_test,y_test))
```

```
0.5576128016072552
```