

Assignment No :- 01

Title :- Compute Estimators of the main statistical measures like Mean, Variance, Standard Deviation, Covariance, Correlation and Standard error with respect to any example. Display graphically the distribution of samples.

```
In [1]: import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt
```

```
In [2]: from sklearn.datasets import load_breast_cancer  
cancer dataset = load_breast_cancer()
```

In [3]: cancer dataset


```
le aspirates. Cancer Letters 77 (1994) \n      163-171.',  
 'feature_names': array(['mean radius', 'mean texture', 'mean perimeter', 'me  
an area',  
     'mean smoothness', 'mean compactness', 'mean concavity',  
     'mean concave points', 'mean symmetry', 'mean fractal dimension',  
     'radius error', 'texture error', 'perimeter error', 'area error',  
     'smoothness error', 'compactness error', 'concavity error',  
     'concave points error', 'symmetry error',  
     'fractal dimension error', 'worst radius', 'worst texture',  
     'worst perimeter', 'worst area', 'worst smoothness',  
     'worst compactness', 'worst concavity', 'worst concave points',  
     'worst symmetry', 'worst fractal dimension'], dtype='<U23'),  
 'filename': '/home/ihack-pc/.local/lib/python3.8/site-packages/sklearn/datas  
ets/data/breast_cancer.csv'}
```

In [4]: `type(cancer dataset)`

```
Out[4]: sklearn.utils.Bunch
```

```
In [5]: cancer_dataset.keys()
```

```
Out[5]: dict_keys(['data', 'target', 'frame', 'target_names', 'DESCR', 'feature_name  
s', 'filename'])
```

```
In [6]: type(cancer_dataset['data'])
```

Out[6]: numpy.ndarray

```
In [7]: cancer_dataset['target']
```

```
In [8]: cancer dataset['target names']
```

```
Out[8]: array(['malignant', 'benign'], dtype='|<U9')
```

```
In [9]: print(cancer_dataset['DESCR'])
```

.. _breast_cancer_dataset:

Breast cancer wisconsin (diagnostic) dataset

Data Set Characteristics:

:Number of Instances: 569

:Number of Attributes: 30 numeric, predictive attributes and the class

:Attribute Information:

- radius (mean of distances from center to points on the perimeter)
- texture (standard deviation of gray-scale values)
- perimeter
- area
- smoothness (local variation in radius lengths)
- compactness ($\text{perimeter}^2 / \text{area} - 1.0$)
- concavity (severity of concave portions of the contour)
- concave points (number of concave portions of the contour)
- symmetry
- fractal dimension ("coastline approximation" - 1)

The mean, standard error, and "worst" or largest (mean of the three worst/largest values) of these features were computed for each image, resulting in 30 features. For instance, field 0 is Mean Radius, field

d

10 is Radius SE, field 20 is Worst Radius.

- class:

- WDBC-Malignant
- WDBC-Benign

:Summary Statistics:

	Min	Max
radius (mean):	6.981	28.11
texture (mean):	9.71	39.28
perimeter (mean):	43.79	188.5
area (mean):	143.5	2501.0
smoothness (mean):	0.053	0.163
compactness (mean):	0.019	0.345
concavity (mean):	0.0	0.427
concave points (mean):	0.0	0.201
symmetry (mean):	0.106	0.304
fractal dimension (mean):	0.05	0.097
radius (standard error):	0.112	2.873
texture (standard error):	0.36	4.885
perimeter (standard error):	0.757	21.98
area (standard error):	6.802	542.2
smoothness (standard error):	0.002	0.031
compactness (standard error):	0.002	0.135
concavity (standard error):	0.0	0.396
concave points (standard error):	0.0	0.053
symmetry (standard error):	0.008	0.079
fractal dimension (standard error):	0.001	0.03
radius (worst):	7.93	36.04

texture (worst):	12.02	49.54
perimeter (worst):	50.41	251.2
area (worst):	185.2	4254.0
smoothness (worst):	0.071	0.223
compactness (worst):	0.027	1.058
concavity (worst):	0.0	1.252
concave points (worst):	0.0	0.291
symmetry (worst):	0.156	0.664
fractal dimension (worst):	0.055	0.208

:Missing Attribute Values: None

:Class Distribution: 212 - Malignant, 357 - Benign

:Creator: Dr. William H. Wolberg, W. Nick Street, Olvi L. Mangasarian

:Donor: Nick Street

:Date: November, 1995

This is a copy of UCI ML Breast Cancer Wisconsin (Diagnostic) datasets.
<https://goo.gl/U2Uwz2>

Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image.

Separating plane described above was obtained using Multisurface Method-Tree (MSM-T) [K. P. Bennett, "Decision Tree Construction Via Linear Programming." Proceedings of the 4th Midwest Artificial Intelligence and Cognitive Science Society, pp. 97-101, 1992], a classification method which uses linear programming to construct a decision tree. Relevant features were selected using an exhaustive search in the space of 1-4 features and 1-3 separating planes.

The actual linear program used to obtain the separating plane in the 3-dimensional space is that described in: [K. P. Bennett and O. L. Mangasarian: "Robust Linear Programming Discrimination of Two Linearly Inseparable Sets", Optimization Methods and Software 1, 1992, 23-34].

This database is also available through the UW CS ftp server:

```
ftp ftp.cs.wisc.edu
cd math-prog/cpo-dataset/machine-learn/WDBC/
```

.. topic:: References

- W.N. Street, W.H. Wolberg and O.L. Mangasarian. Nuclear feature extraction for breast tumor diagnosis. IS&T/SPIE 1993 International Symposium on Electronic Imaging: Science and Technology, volume 1905, pages 861-870, San Jose, CA, 1993.
- O.L. Mangasarian, W.N. Street and W.H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. Operations Research, 43(4), pages 570-577, July-August 1995.
- W.H. Wolberg, W.N. Street, and O.L. Mangasarian. Machine learning techniques to diagnose breast cancer from fine-needle aspirates. Cancer Letters 77

(1994)
163-171.

In [10]: `print(cancer_dataset['feature_names'])`

```
['mean radius' 'mean texture' 'mean perimeter' 'mean area'
 'mean smoothness' 'mean compactness' 'mean concavity'
 'mean concave points' 'mean symmetry' 'mean fractal dimension'
 'radius error' 'texture error' 'perimeter error' 'area error'
 'smoothness error' 'compactness error' 'concavity error'
 'concave points error' 'symmetry error' 'fractal dimension error'
 'worst radius' 'worst texture' 'worst perimeter' 'worst area'
 'worst smoothness' 'worst compactness' 'worst concavity'
 'worst concave points' 'worst symmetry' 'worst fractal dimension']
```

In [11]: `print(cancer_dataset['filename'])`

```
/home/ihack-pc/.local/lib/python3.8/site-packages/scikit-learn/datasets/data/breast_cancer.csv
```

In [12]: `cancer_df = pd.DataFrame(np.c_[cancer_dataset['data'], cancer_dataset['target']])`

In [13]: `cancer_df.to_csv('breast_cancer_dataframe.csv')`

In [14]: `cancer_df.head()`

Out[14]:

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809

5 rows × 31 columns

In [15]: `cancer_df.tail()`

Out[15]:

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry
564	21.56	22.39	142.00	1479.0	0.11100	0.11590	0.24390	0.13890	0.1726
565	20.13	28.25	131.20	1261.0	0.09780	0.10340	0.14400	0.09791	0.1752
566	16.60	28.08	108.30	858.1	0.08455	0.10230	0.09251	0.05302	0.1590
567	20.60	29.33	140.10	1265.0	0.11780	0.27700	0.35140	0.15200	0.2397
568	7.76	24.54	47.92	181.0	0.05263	0.04362	0.00000	0.00000	0.1581

5 rows × 31 columns

In [16]: `cancer_df.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 31 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   mean radius      569 non-null    float64
 1   mean texture     569 non-null    float64
 2   mean perimeter   569 non-null    float64
 3   mean area        569 non-null    float64
 4   mean smoothness  569 non-null    float64
 5   mean compactness 569 non-null    float64
 6   mean concavity   569 non-null    float64
 7   mean concave points 569 non-null    float64
 8   mean symmetry    569 non-null    float64
 9   mean fractal dimension 569 non-null    float64
 10  radius error    569 non-null    float64
 11  texture error   569 non-null    float64
 12  perimeter error 569 non-null    float64
 13  area error      569 non-null    float64
 14  smoothness error 569 non-null    float64
 15  compactness error 569 non-null    float64
 16  concavity error 569 non-null    float64
 17  concave points error 569 non-null    float64
 18  symmetry error   569 non-null    float64
 19  fractal dimension error 569 non-null    float64
 20  worst radius     569 non-null    float64
 21  worst texture    569 non-null    float64
 22  worst perimeter   569 non-null    float64
 23  worst area        569 non-null    float64
 24  worst smoothness  569 non-null    float64
 25  worst compactness 569 non-null    float64
 26  worst concavity   569 non-null    float64
 27  worst concave points 569 non-null    float64
 28  worst symmetry    569 non-null    float64
 29  worst fractal dimension 569 non-null    float64
 30  target           569 non-null    float64
dtypes: float64(31)
memory usage: 137.9 KB
```

Finding Means Value & Describing Dataset

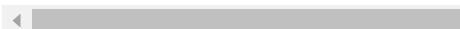
In [17]: `cancer_df.describe()`

Out[17]:

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity
count	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000	569.000000
mean	14.127292	19.289649	91.969033	654.889104	0.096360	0.104341	0.088799
std	3.524049	4.301036	24.298981	351.914129	0.014064	0.052813	0.079720
min	6.981000	9.710000	43.790000	143.500000	0.052630	0.019380	0.000000
25%	11.700000	16.170000	75.170000	420.300000	0.086370	0.064920	0.029560
50%	13.370000	18.840000	86.240000	551.100000	0.095870	0.092630	0.061540
75%	15.780000	21.800000	104.100000	782.700000	0.105300	0.130400	0.130700

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity
max	28.110000	39.280000	188.500000	2501.000000	0.163400	0.345400	0.426800

8 rows × 31 columns



In [18]: `cancer_df.isnull().sum()`

```
Out[18]: mean radius          0
          mean texture         0
          mean perimeter        0
          mean area             0
          mean smoothness        0
          mean compactness       0
          mean concavity         0
          mean concave points    0
          mean symmetry          0
          mean fractal dimension 0
          radius error           0
          texture error          0
          perimeter error        0
          area error              0
          smoothness error        0
          compactness error       0
          concavity error         0
          concave points error    0
          symmetry error          0
          fractal dimension error 0
          worst radius            0
          worst texture           0
          worst perimeter          0
          worst area               0
          worst smoothness         0
          worst compactness        0
          worst concavity          0
          worst concave points     0
          worst symmetry           0
          worst fractal dimension 0
          target                  0
          dtype: int64
```

Finding the variance

In [32]: `cancer_df.var()`

```
Out[32]: 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
          radius error            0.076902
          texture error           0.304316
```

perimeter error	4.087896
area error	2069.431583
smoothness error	0.000009
compactness error	0.000321
concavity error	0.000911
concave points error	0.000038
symmetry error	0.000068
fractal dimension error	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
target	0.234177
dtype:	float64

Standard Deviation

In [33]:

```
cancer_df.std()
```

Out[33]:

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
radius error	0.277313
texture error	0.551648
perimeter error	2.021855
area error	45.491006
smoothness error	0.003003
compactness error	0.017908
concavity error	0.030186
concave points error	0.006170
symmetry error	0.008266
fractal dimension error	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
target	0.483918
dtype:	float64

Finding Covariance

In [34]: cancer_df.cov()

Out[34]:

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness
mean radius	12.418920	4.907582	85.447142	1224.483409	0.008454	0.094197
mean texture	4.907582	18.498909	34.439759	485.993787	-0.001415	0.053767
mean perimeter	85.447142	34.439759	590.440480	8435.772345	0.070836	0.714714
mean area	1224.483409	485.993787	8435.772345	123843.554318	0.876178	9.264931
mean smoothness	0.008454	-0.001415	0.070836	0.876178	0.000198	0.000490
mean compactness	0.094197	0.053767	0.714714	9.264931	0.000490	0.002789
mean concavity	0.190128	0.103692	1.387234	19.244924	0.000585	0.003718
mean concave points	0.112475	0.048977	0.802360	11.241958	0.000302	0.001703
mean symmetry	0.014273	0.008419	0.121922	1.459596	0.000215	0.000873
mean fractal dimension	-0.007754	-0.002321	-0.044859	-0.703426	0.000058	0.000211
radius error	0.663650	0.329037	4.661401	71.490945	0.001176	0.007286
texture error	-0.189189	0.916695	-1.162988	-12.867168	0.000531	0.001346
perimeter error	4.803550	2.449449	34.053028	517.009995	0.008420	0.058612
area error	117.968162	50.840865	823.492755	12808.517580	0.157742	1.094708
smoothness error	-0.002355	0.000085	-0.014788	-0.176221	0.000014	0.000021
compactness error	0.013001	0.014787	0.109111	1.339725	0.000080	0.000699
concavity error	0.020659	0.018604	0.167296	2.205952	0.000105	0.000910
concave points error	0.008180	0.004348	0.061055	0.808460	0.000033	0.000209
symmetry error	-0.003039	0.000325	-0.016396	-0.210896	0.000023	0.000100
fractal dimension error	-0.000398	0.000620	-0.000355	-0.018519	0.000011	0.000071
worst radius	16.513749	7.329267	113.858063	1637.521341	0.014487	0.136643
worst texture	6.433100	24.110148	45.258113	621.824934	0.003118	0.080544
worst perimeter	114.288570	51.745933	792.328208	11341.789807	0.112879	1.047413

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness
worst area	1888.227223	841.283832	13026.148359	192192.557633	1.655299	15.323436
worst smoothness	0.009625	0.007611	0.083526	0.992514	0.000259	0.000682
worst compactness	0.229249	0.188010	1.742478	21.616602	0.001045	0.007194
worst concavity	0.387386	0.270110	2.858506	37.634415	0.001276	0.008994
worst concave points	0.172393	0.083491	1.231848	16.701789	0.000465	0.002831
worst symmetry	0.035746	0.027942	0.284300	3.125809	0.000343	0.001667
worst fractal dimension	0.000450	0.009260	0.022391	0.023756	0.000127	0.000656
target	-1.244954	-0.864145	-8.732438	-120.738222	-0.002440	-0.015246

31 rows × 31 columns



Finding Correlation

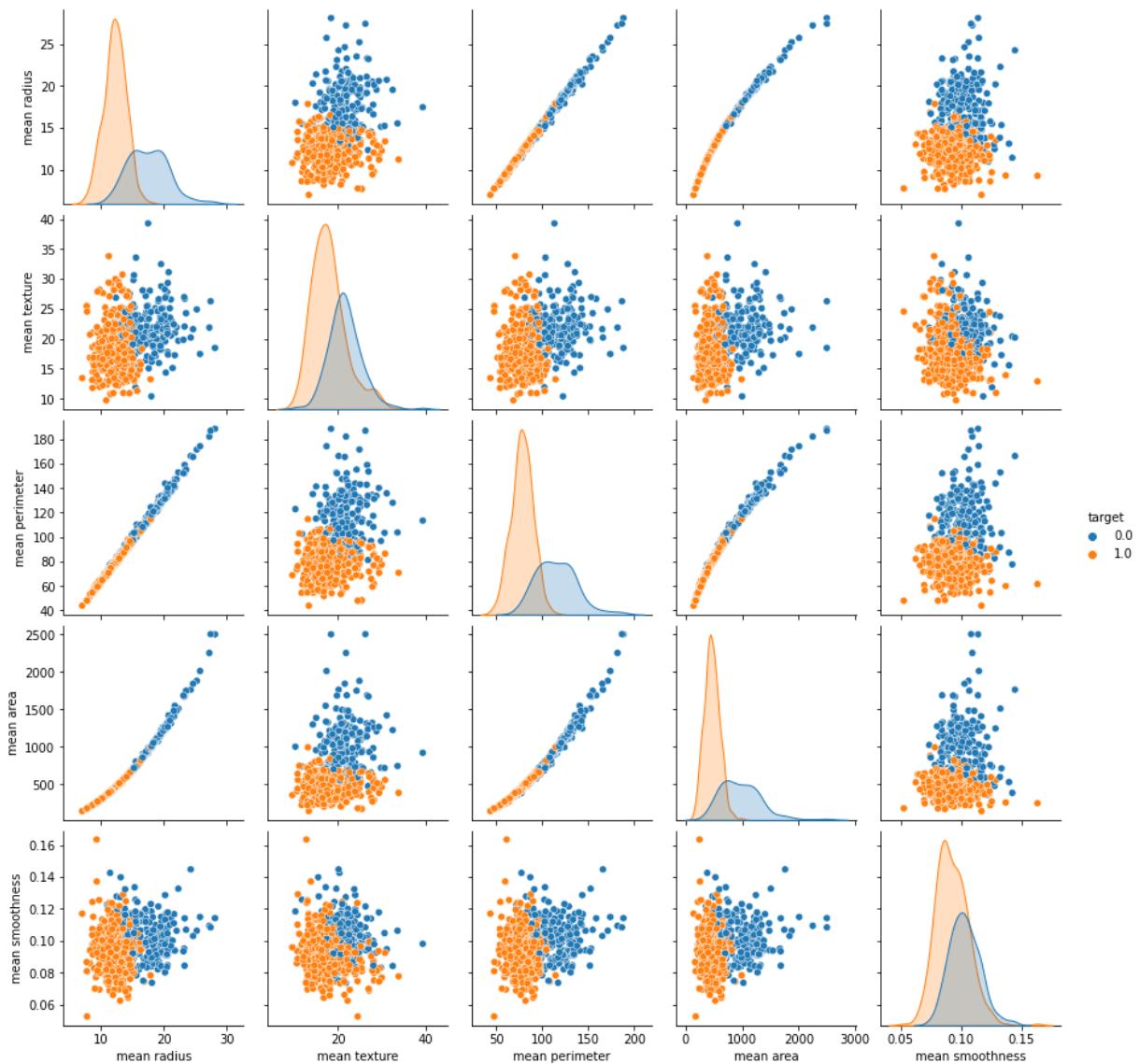
In [19]: `import seaborn as sns`

In [20]: `sns.pairplot(cancer_df, hue = 'target')`

Out[20]: <seaborn.axisgrid.PairGrid at 0x7f6d8ce53df0>



```
In [21]: sns.pairplot(cancer_df, hue = 'target', vars = ['mean radius', 'mean texture',  
Out[21]: <seaborn.axisgrid.PairGrid at 0x7f6d5ffc1490>
```



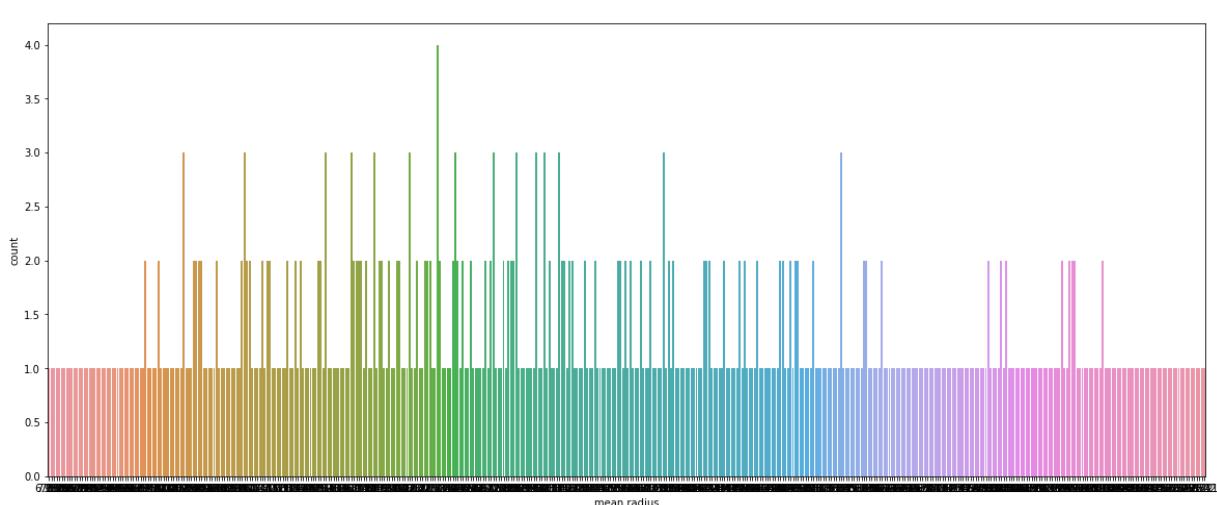
In [23]:

```
plt.figure(figsize = (20,8))
sns.countplot(cancer_df['mean radius'])
```

/home/ihack-pc/.local/lib/python3.8/site-packages/seaborn/_decorators.py:36:
FutureWarning: Pass the following variable as a keyword arg: x. From version
0.12, the only valid positional argument will be `data`, and passing other ar-
guments without an explicit keyword will result in an error or misinterpretation.

```
warnings.warn(
```

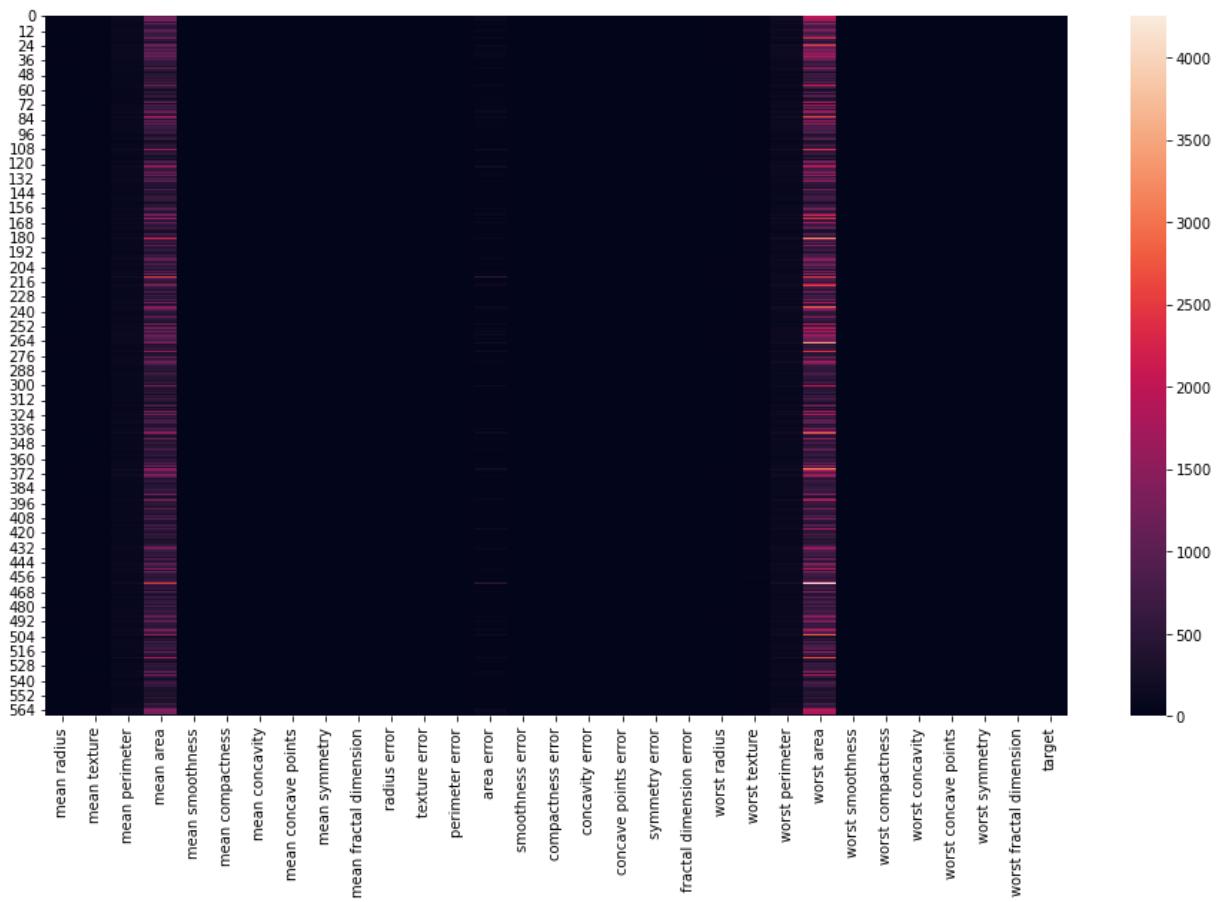
Out[23]:



In [24]:

```
plt.figure(figsize=(16,9))
sns.heatmap(cancer_df)
```

Out[24]: <AxesSubplot:>



In [25]:

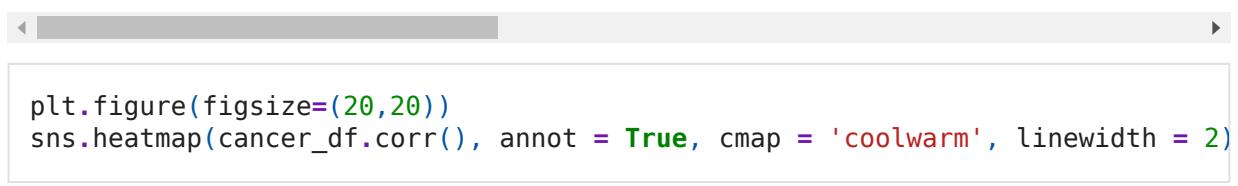
```
cancer_df.corr()
```

Out[25]:

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	target
mean radius	1.000000	0.323782	0.997855	0.987357	0.170581	0.506124	0.676764	0
mean texture	0.323782	1.000000	0.329533	0.321086	-0.023389	0.236702	0.302418	0
mean perimeter	0.997855	0.329533	1.000000	0.986507	0.207278	0.556936	0.716136	0
mean area	0.987357	0.321086	0.986507	1.000000	0.177028	0.498502	0.685983	0
mean smoothness	0.170581	-0.023389	0.207278	0.177028	1.000000	0.659123	0.521984	0
mean compactness	0.506124	0.236702	0.556936	0.498502	0.659123	1.000000	0.883121	0
mean concavity	0.676764	0.302418	0.716136	0.685983	0.521984	0.883121	1.000000	0
mean concave points	0.822529	0.293464	0.850977	0.823269	0.553695	0.831135	0.921391	1
mean symmetry	0.147741	0.071401	0.183027	0.151293	0.557775	0.602641	0.500667	0
mean fractal dimension	-0.311631	-0.076437	-0.261477	-0.283110	0.584792	0.565369	0.336783	0

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points
radius error	0.679090	0.275869	0.691765	0.732562	0.301467	0.497473	0.631925	0.000000
texture error	-0.097317	0.386358	-0.086761	-0.066280	0.068406	0.046205	0.076218	0.000000
perimeter error	0.674172	0.281673	0.693135	0.726628	0.296092	0.548905	0.660391	0.000000
area error	0.735864	0.259845	0.744983	0.800086	0.246552	0.455653	0.617427	0.000000
smoothness error	-0.222600	0.006614	-0.202694	-0.166777	0.332375	0.135299	0.098564	0.000000
compactness error	0.206000	0.191975	0.250744	0.212583	0.318943	0.738722	0.670279	0.000000
concavity error	0.194204	0.143293	0.228082	0.207660	0.248396	0.570517	0.691270	0.000000
concave points error	0.376169	0.163851	0.407217	0.372320	0.380676	0.642262	0.683260	0.000000
symmetry error	-0.104321	0.009127	-0.081629	-0.072497	0.200774	0.229977	0.178009	0.000000
fractal dimension error	-0.042641	0.054458	-0.005523	-0.019887	0.283607	0.507318	0.449301	0.000000
worst radius	0.969539	0.352573	0.969476	0.962746	0.213120	0.535315	0.688236	0.000000
worst texture	0.297008	0.912045	0.303038	0.287489	0.036072	0.248133	0.299879	0.000000
worst perimeter	0.965137	0.358040	0.970387	0.959120	0.238853	0.590210	0.729565	0.000000
worst area	0.941082	0.343546	0.941550	0.959213	0.206718	0.509604	0.675987	0.000000
worst smoothness	0.119616	0.077503	0.150549	0.123523	0.805324	0.565541	0.448822	0.000000
worst compactness	0.413463	0.277830	0.455774	0.390410	0.472468	0.865809	0.754968	0.000000
worst concavity	0.526911	0.301025	0.563879	0.512606	0.434926	0.816275	0.884103	0.000000
worst concave points	0.744214	0.295316	0.771241	0.722017	0.503053	0.815573	0.861323	0.000000
worst symmetry	0.163953	0.105008	0.189115	0.143570	0.394309	0.510223	0.409464	0.000000
worst fractal dimension	0.007066	0.119205	0.051019	0.003738	0.499316	0.687382	0.514930	0.000000
target	-0.730029	-0.415185	-0.742636	-0.708984	-0.358560	-0.596534	-0.696360	-0.000000

31 rows × 31 columns

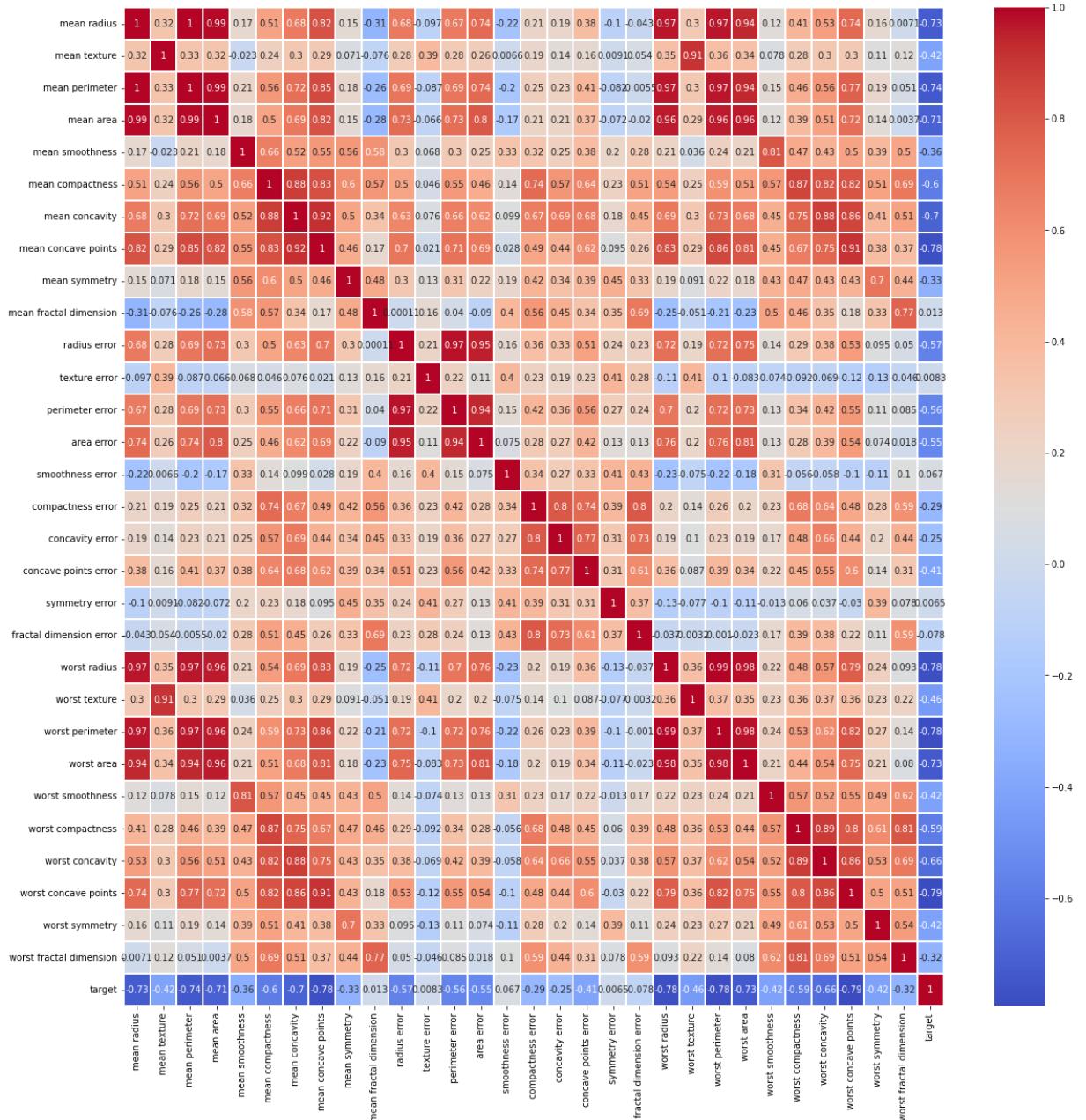


In [26]:

```
plt.figure(figsize=(20,20))
sns.heatmap(cancer_df.corr(), annot = True, cmap = 'coolwarm', linewidth = 2)
```

Out[26]:

<AxesSubplot:>



In [27]:

```
cancer_df2 = cancer_df.drop(['target'], axis =1)
print("The Shape of 'Cancer_df2' is : ",cancer_df2.shape)
```

The Shape of 'Cancer_df2' is : (569, 30)

In [28]:

```
plt.figure(figsize=(16,5))
ax = sns.barplot(cancer_df2.corrwith(cancer_df.target).index, cancer_df2.corr
ax.tick_params(labelrotation = 90)
```

```
/home/ihack-pc/.local/lib/python3.8/site-packages/seaborn/_decorators.py:36:
FutureWarning: Pass the following variables as keyword args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.
warnings.warn(
```

