

MACHINE LEARNING ASSIGNMENT-3

Name : venkata.k.l.spandana vegi

Id : 700744287

GITHUB LINK

https://github.com/spandanavegi/machine_learning.1/blob/main/ML_ASSIGNMENT_3.ipynb

VIDEO LINK:

https://drive.google.com/file/d/1_onlrvuEuTGawsj87EWb9XdzquQqjIA/view?usp=sharing

1)

Find the correlation between 'survived' (target column) and 'sex' column for the Titanic use case in class.

a. Do you think we should keep this feature?

Here after importing the required libraries we print the values using head function.

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cummings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Here after displaying the head function with values. We used correlation function and found the relation between the sex and survived

a) From the above case we can say that we have to keep this feature.

-0.5433513806577547

```
1 #from the above case we can say that we have to keep this feature.
```

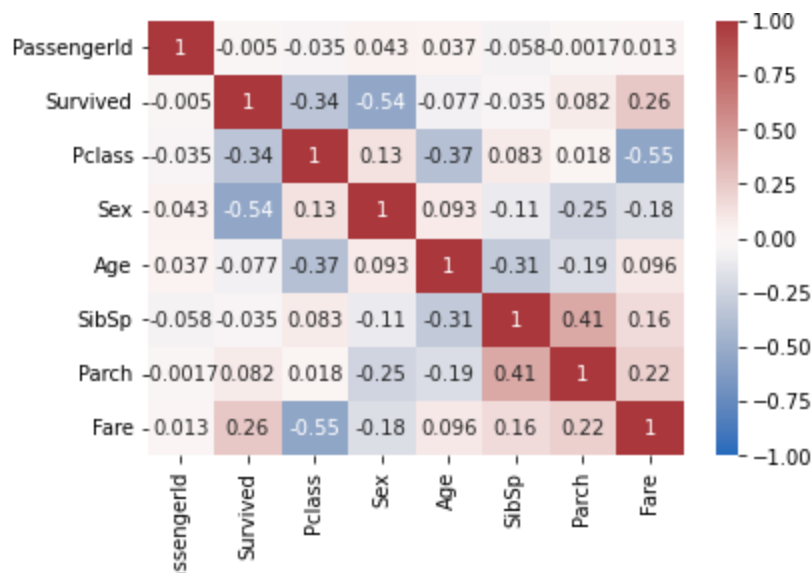
```
1 matrix = df.corr()  
2 print(matrix)
```

```
      PassengerId  Survived  Pclass    Sex    Age  SibSp  \  
PassengerId      1.000000 -0.005007 -0.035144  0.042939  0.036847 -0.057527  
Survived         -0.005007  1.000000 -0.338481 -0.543351 -0.077221 -0.035322  
Pclass           -0.035144 -0.338481  1.000000  0.131900 -0.369226  0.083081  
Sex              0.042939 -0.543351  0.131900  1.000000  0.093254 -0.114631  
Age              0.036847 -0.077221 -0.369226  0.093254  1.000000 -0.308247  
SibSp            -0.057527 -0.035322  0.083081 -0.114631 -0.308247  1.000000  
Parch            -0.001652  0.081629  0.018443 -0.245489 -0.189119  0.414838  
Fare             0.012658  0.257307 -0.549500 -0.182333  0.096067  0.159651  
  
      Parch    Fare  
PassengerId -0.001652  0.012658  
Survived     0.081629  0.257307  
Pclass       0.018443 -0.549500  
Sex          -0.245489 -0.182333  
Age          -0.189119  0.096067  
SibSp        0.414838  0.159651  
Parch        1.000000  0.216225  
Fare         0.216225  1.000000
```

Then we displayed the correlation between all the attributes.

	Age	Embarked	Fare	Parch	Pclass	Sex	SibSp	Survived	train
Age	1.000000	0.048993	0.178740	-0.150917	-0.408106	0.063645	-0.243699	-0.077221	-0.018528
Embarked	0.048993	1.000000	0.062017	-0.095975	0.037527	-0.121868	-0.073937	0.108669	-0.072726
Fare	0.178740	0.062017	1.000000	0.221539	-0.558629	-0.185523	0.160238	0.257307	-0.030831
Parch	-0.150917	-0.095975	0.221539	1.000000	0.018322	-0.213125	0.373587	0.081629	-0.005793
Pclass	-0.408106	0.037527	-0.558629	0.018322	1.000000	0.124617	0.060832	-0.338481	0.023988
Sex	0.063645	-0.121868	-0.185523	-0.213125	0.124617	1.000000	-0.109609	-0.543351	0.010928
SibSp	-0.243699	-0.073937	0.160238	0.373587	0.060832	-0.109609	1.000000	-0.035322	0.033867
Survived	-0.077221	0.108669	0.257307	0.081629	-0.338481	-0.543351	-0.035322	1.000000	nan
train	-0.018528	-0.072726	-0.030831	-0.005793	0.023988	0.010928	0.033867	nan	1.000000

Then using a heat map and plot show, we displayed the contents in the matrix form.



Then by using the naives bayes theorem we calculated the accuracy.

	precision	recall	f1-score	support
0.0	0.79	0.80	0.80	85
1.0	0.70	0.69	0.70	58
accuracy			0.76	143
macro avg	0.75	0.74	0.75	143
weighted avg	0.75	0.76	0.75	143

```
[[68 17]
 [18 40]]
accuracy is 0.7552447552447552
```

2)

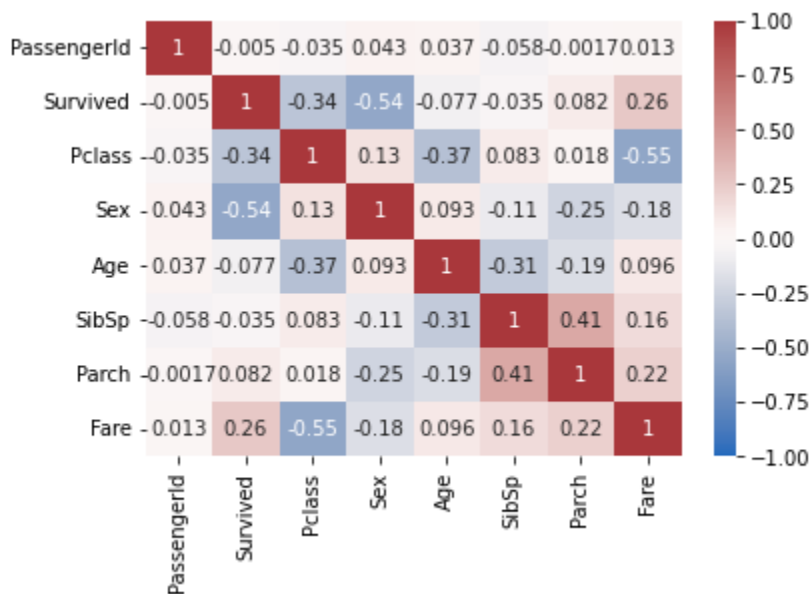
Here we have imported glass dataset and displayed the values using head function

	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type
0	1.52101	13.64	4.49	1.10	71.78	0.06	8.75	0.0	0.0	1
1	1.51761	13.89	3.60	1.36	72.73	0.48	7.83	0.0	0.0	1
2	1.51618	13.53	3.55	1.54	72.99	0.39	7.78	0.0	0.0	1
3	1.51766	13.21	3.69	1.29	72.61	0.57	8.22	0.0	0.0	1
4	1.51742	13.27	3.62	1.24	73.08	0.55	8.07	0.0	0.0	1

Then we have found correlation between all the attributes.

	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type
RI	1.000000	-0.191885	-0.122274	-0.407326	-0.542052	-0.289833	0.810403	-0.000386	0.143010	-0.164237
Na	-0.191885	1.000000	-0.273732	0.156794	-0.069809	-0.266087	-0.275442	0.326603	-0.241346	0.502898
Mg	-0.122274	-0.273732	1.000000	-0.481799	-0.165927	0.005396	-0.443750	-0.492262	0.083060	-0.744993
Al	-0.407326	0.156794	-0.481799	1.000000	-0.005524	0.325958	-0.259592	0.479404	-0.074402	0.598829
Si	-0.542052	-0.069809	-0.165927	-0.005524	1.000000	-0.193331	-0.208732	-0.102151	-0.094201	0.151565
K	-0.289833	-0.266087	0.005396	0.325958	-0.193331	1.000000	-0.317836	-0.042618	-0.007719	-0.010054
Ca	0.810403	-0.275442	-0.443750	-0.259592	-0.208732	-0.317836	1.000000	-0.112841	0.124968	0.000952
Ba	-0.000386	0.326603	-0.492262	0.479404	-0.102151	-0.042618	-0.112841	1.000000	-0.058692	0.575161
Fe	0.143010	-0.241346	0.083060	-0.074402	-0.094201	-0.007719	0.124968	-0.058692	1.000000	-0.188278
Type	-0.164237	0.502898	-0.744993	0.598829	0.151565	-0.010054	0.000952	0.575161	-0.188278	1.000000

Then using a heat map and plot show, we displaced the contents in the matrix form.



Then using naives bayes theorem we got an accuracy of 83.7%

	precision	recall	f1-score	support
1	0.90	0.95	0.92	19
2	0.92	0.92	0.92	12
3	1.00	0.50	0.67	6
5	0.00	0.00	0.00	1
6	1.00	1.00	1.00	1
7	0.75	0.75	0.75	4
accuracy			0.84	43
macro avg	0.76	0.69	0.71	43
weighted avg	0.89	0.84	0.85	43

```
[[18  1  0  0  0  0]
 [ 1 11  0  0  0  0]
 [ 1  0  3  2  0  0]
 [ 0  0  0  0  0  1]
 [ 0  0  0  0  1  0]
 [ 0  0  0  1  0  3]]
accuracy is 0.8372093023255814
```

Then using support vector machine learning algorithm we got an accuracy of 74.4%

Which algorithm got better accuracy? Can you justify why?

Naive Bayes algorithm got better accuracy. Naive Bayes requires a small amount of training data,

It tends to perform well for problems like spam detection and text classification.

SVM algorithms typically don't account for easily interpretable probabilities and also SVM is more expensive than naive bayes.