MAKALAH PEMBELAJARAN MESIN

“Titanic_all”

Pengampu :

Dr.Juni Nurma Sari S.Kom, M.MT

Nama:

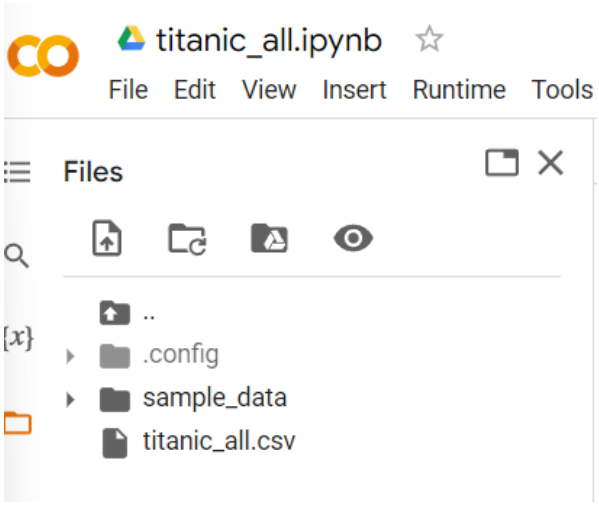
Wanda Trisnahayu (2055301143)

Kelas : 3 TI C

D4 - TEKNIK INFORMATIKA

TAHUN 2022

1. Upload data



2. Import library python yang kita butuhkan

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as scp
```

3. Read data pada titanic_all.csv

```
titanic = pd.read_csv("titanic_all.csv")
print("data :",titanic.shape)
titanic.info()
titanic.head()
```

```
data : (1309, 12)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1309 entries, 0 to 1308
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  1309 non-null   int64
 1   Survived     1309 non-null   int64
 2   Pclass       1309 non-null   int64
 3   Name         1309 non-null   object
 4   Sex          1309 non-null   object
 5   Age          1046 non-null   float64
 6   SibSp        1309 non-null   int64
 7   Parch        1309 non-null   int64
 8   Ticket       1309 non-null   object
 9   Fare         1308 non-null   float64
 10  Cabin        295 non-null    object
 11  Embarked     1307 non-null   object
dtypes: float64(2), int64(5), object(5)
memory usage: 122.8+ KB
```

memory usage: 122.8+ KB

	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	Cumings, 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

4. Basic summary

```
[ ] #BASIC SUMMARY
    titanic.describe() #analisa deskriptif
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	1309.000000	1309.000000	1309.000000	1046.000000	1309.000000	1309.000000	1308.000000
mean	655.000000	0.377387	2.294882	29.881138	0.498854	0.385027	33.295479
std	378.020061	0.484918	0.837836	14.413493	1.041658	0.865560	51.758668
min	1.000000	0.000000	1.000000	0.170000	0.000000	0.000000	0.000000
25%	328.000000	0.000000	2.000000	21.000000	0.000000	0.000000	7.895800
50%	655.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	982.000000	1.000000	3.000000	39.000000	1.000000	0.000000	31.275000
max	1309.000000	1.000000	3.000000	80.000000	8.000000	9.000000	512.329200

5. Display numeric dan categorical

```
display(titanic.describe(include=np.number).transpose()) #numeric
display(titanic.describe(include=np.object).transpose()) #categorical
```

	count	mean	std	min	25%	50%	75%	max
PassengerId	1309.0	655.000000	378.020061	1.00	328.0000	655.0000	982.000	1309.0000
Survived	1309.0	0.377387	0.484918	0.00	0.0000	0.0000	1.000	1.0000
Pclass	1309.0	2.294882	0.837836	1.00	2.0000	3.0000	3.000	3.0000
Age	1046.0	29.881138	14.413493	0.17	21.0000	28.0000	39.000	80.0000
SibSp	1309.0	0.498854	1.041658	0.00	0.0000	0.0000	1.000	8.0000
Parch	1309.0	0.385027	0.865560	0.00	0.0000	0.0000	0.000	9.0000
Fare	1308.0	33.295479	51.758668	0.00	7.8958	14.4542	31.275	512.3292

	count	unique	top	freq
Name	1309	1307	Connolly, Miss. Kate	2
Sex	1309	2	male	843
Ticket	1309	929	CA. 2343	11
Cabin	295	186	C23 C25 C27	6
Embarked	1307	3	S	914

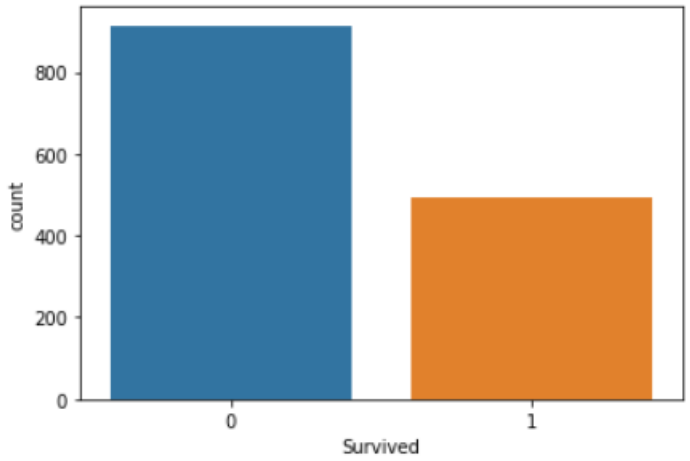
6. Visualisasi

```
[ ]  #VISUALISASI
     def countplot(column):
       return sns.countplot(x=column, data=titanic)
```

7. Countplot

```
[ ]  countplot('Pclass')
     countplot('Sex')
     countplot('Embarked')
     countplot('Survived')
```

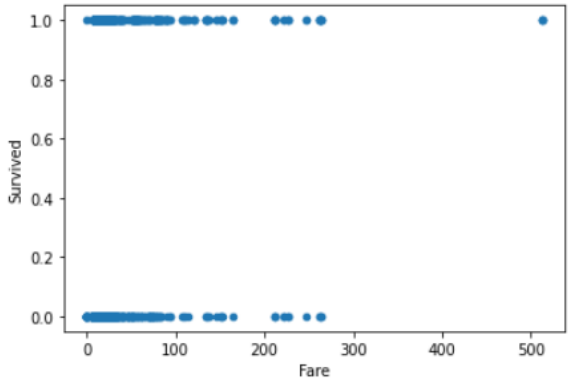
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe16ec42490>
```



8. Plot scatter

```
titanic.plot.scatter(x="Fare", y='Survived', figsize=(6,4))
```

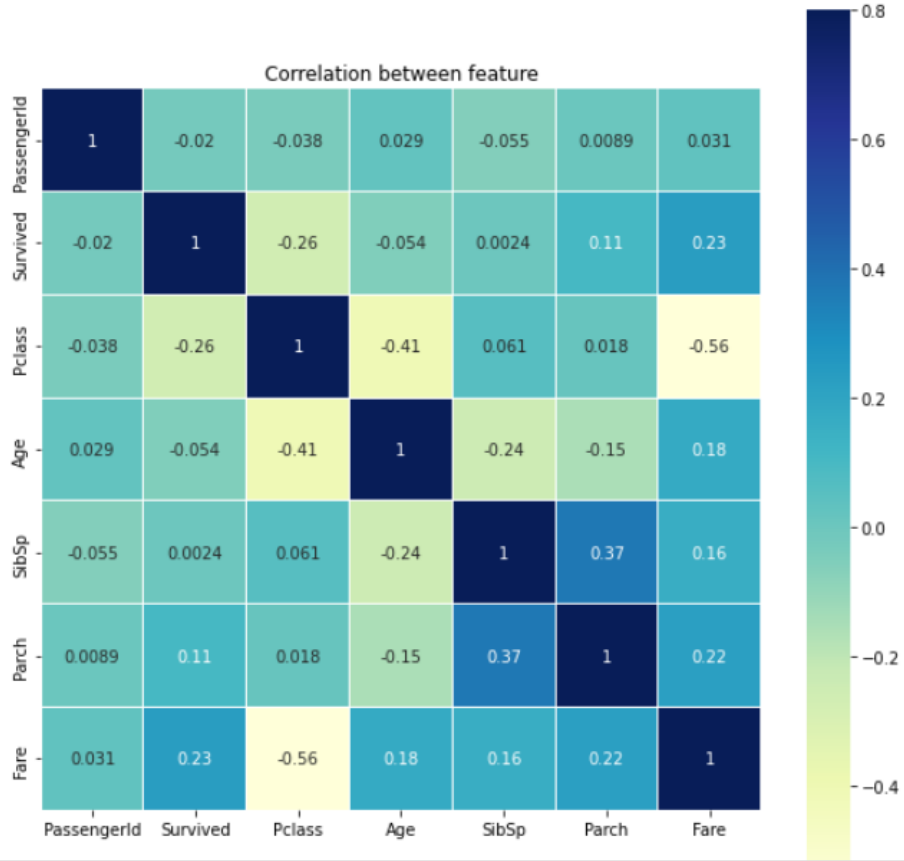
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe16eb93e50>
```



9. Korelasi

```
titanic.corr(method='pearson')
corr= titanic.corr() #survived
plt.figure(figsize=(10,10))
sns.heatmap(corr, vmax=.8, linewidths=0.01, square=True, annot=True,cmap='YlGnBu'
linecolor="white")
plt.title('Correlation between feature')
```

```
Text(0.5, 1.0, 'Correlation between feature')
```



✓ 0s completed at 8:33 AM

10. Compute_freq_chi2

```
def compute_freq_chi2(x,y):
  freqtab = pd.crosstab(x,y)
  print("Frequency Table")
  print("----------------")
  print(freqtab)
  print("----------------")
  chi2, pval, df, expected = scp.chi2_contingency(freqtab)

  print("Chisquare test statistic", chi2)
  print("p-value", pval)
  return
```

11. Compute_freq_chi2

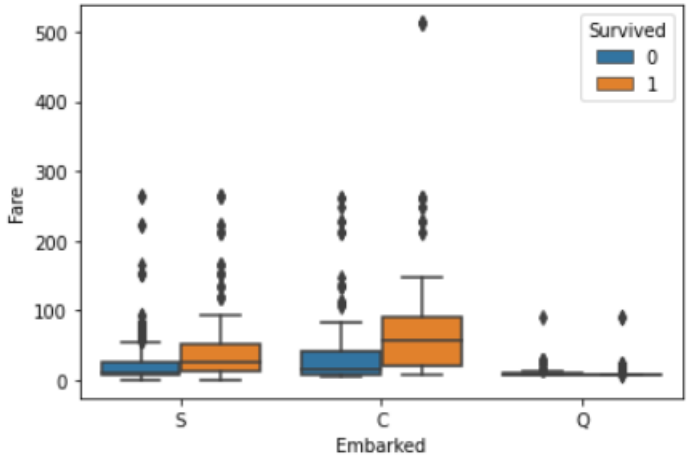
```
compute_freq_chi2(titanic.Survived, titanic.Pclass)
compute_freq_chi2(titanic.Survived, titanic.Embarked)
compute_freq_chi2(titanic.Survived, titanic.Sex)
```

```
Frequency Table
----------------
Pclass      1    2    3
Survived
0         137  160  518
1         186  117  191
----------------
Chisquare test statistic 91.72367559290264
p-value 1.2090852275863847e-20
Frequency Table
----------------
Embarked    C   Q    S
Survived
0         137  69  609
1         133  54  305
----------------
Chisquare test statistic 24.684434014740326
p-value 4.363583182075015e-06
Frequency Table
----------------
Sex       female  male
Survived
0             81   734
1            385   109
----------------
Chisquare test statistic 617.3133522952658
p-value 2.871410444001617e-136
```

12. Sns.bloxplot

```
[ ]  sns.boxplot(x="Embarked", y="Fare", hue="Survived", data=titanic)
```

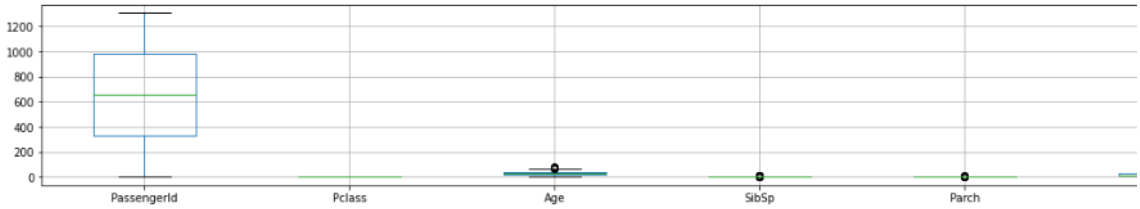
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fe16bd0e310>
```



13. Data.bloxpot

```
[ ]  data2=titanic.drop(['Survived'], axis=1)
     data2.boxplot(figsize=(20,3))
```

```
/usr/local/lib/python3.7/dist-packages/matplotlib/cbook/__init__.py:1376: VisibleDeprecationWarning: Cr
  X = np.atleast_1d(X.T if isinstance(X, np.ndarray) else np.asarray(X))
<matplotlib.axes._subplots.AxesSubplot at 0x7fe16bc1b310>
```



14. Plot

```
col = (['Survived', 'Pclass', 'Embarked', 'Fare', 'Age', 'SibSp'])
sns.pairplot(titanic[col], kind='scatter', diag_kind='kde')
plt.show()
```

15. Data cleaning

```
[ ]  #DATA CLEANING

     titanic.duplicated(keep=False).sum()
```

```
0
```

16. Cek null

```
[ ]  #CEK NULL

     def cek_null(df) :
       col_na = df.isnull().sum().sort_values(ascending=False)
       percent = col_na / len(df)
       missing_data = pd.concat([col_na, percent], axis=1, keys=['Total', 'Percent'])
       print(missing_data[missing_data['Total'] > 0])
```

17. Cek null

```
[ ]  cek_null(titanic)
     titanic['Cabin'].str.split(" ",expand=True).count().rename(lambda x : x+1)
```

```
              Total   Percent
Cabin          1014  0.774637
Age             263  0.200917
Embarked          2  0.001528
Fare              1  0.000764
1    295
2     41
3     15
4      5
dtype: int64
```

18. Data survived

```
[ ]  (titanic
     .groupby([titanic.Cabin.str[:1],'Survived'])
     .Survived
     .count()
     .unstack())
```

Survived	0	1
Cabin		
A	12.0	10.0
B	21.0	44.0
C	40.0	54.0
D	16.0	30.0
E	13.0	28.0
F	10.0	11.0
G	2.0	3.0
T	1.0	NaN

19. Data survived

```
[ ] (titanic
    .groupby([titanic.Cabin.str[:1],'Survived'])
    .Fare
    .mean()
    .unstack())
```

Survived	0	1
Cabin		
A	37.977425	45.164580
B	81.865871	141.720836
C	102.199170	112.169137
D	43.420563	58.120287
E	59.980446	52.050150
F	10.480840	24.987118
G	10.462500	16.700000
T	35.500000	NaN

20. Mengisi null

```
[ ] #MENGISI DATA NULL

    titanic['Cabin'] = titanic['Cabin'].fillna('U')
    (titanic.groupby([titanic.Cabin.str[:1],'Survived'])
      .Survived
      .count()
      .unstack())
    cek_null(titanic)
```

```
            Total   Percent
Age           263  0.200917
Embarked        2  0.001528
Fare            1  0.000764
```

21. Menghapus data

```
#MENGHAPUS DATA

titanic_cleaned = titanic.drop(['Name','Ticket', 'Cabin'],axis=1)
titanic_cleaned.head()

cek_null(titanic)
```

```
           Total   Percent
Age          263  0.200917
Embarked       2  0.001528
Fare           1  0.000764
```

22. Mengisi data null

```
#MENGISI DATA NULL

titanic_cleaned['Age'] = titanic_cleaned['Age'].fillna('median')
cek_null(titanic_cleaned)
```

```
           Total   Percent
Embarked       2  0.001528
Fare           1  0.000764
```

23. Mengisi data null

```
#MENGISI DATA NULL

titanic_cleaned['Embarked'] = titanic_cleaned['Embarked'].fillna('C')
cek_null(titanic_cleaned)

titanic_cleaned[titanic_cleaned['Fare'].isnull()]
```

```
Empty DataFrame
Columns: [Total, Percent]
Index: []
```

PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked

24. Menghapus data record

```
#MENGHAPUS DATA RECORD

titanic_cleaned.dropna(inplace=True)
titanic_cleaned.head()
```

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	1	0	3	male	22.0	1	0	7.2500	S
1	2	1	1	female	38.0	1	0	71.2833	C
2	3	1	3	female	26.0	0	0	7.9250	S
3	4	1	1	female	35.0	1	0	53.1000	S
4	5	0	3	male	35.0	0	0	8.0500	S