iris-project

December 10, 2023

IRIS DATASET

Importing libraries and dataset

[ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
df=pd.read_csv('/content/IRIS (2).csv')
df
```

[ ]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
..	…	…	…	…	…
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

[150 rows x 5 columns]

Data Preprocessing

[ ]:
```
df.head()
```

[ ]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

[ ]:
```
df.tail()
```

[ ]:

	sepal_length	sepal_width	petal_length	petal_width	species
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

[ ]:
```
df.dtypes
```

[ ]:
```
sepal_length    float64
sepal_width     float64
petal_length    float64
petal_width     float64
species          object
dtype: object
```

[ ]:
```
df.columns
```

[ ]:
```
Index(['sepal_length', 'sepal_width', 'petal_length', 'petal_width',
       'species'],
      dtype='object')
```

[ ]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   sepal_length  150 non-null    float64
 1   sepal_width   150 non-null    float64
 2   petal_length  150 non-null    float64
 3   petal_width   150 non-null    float64
 4   species       150 non-null    object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

[ ]:
```
df.describe()
```

[ ]:

	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000

```
max         7.900000      4.400000      6.900000      2.500000
```

[ ]:
```
df.isna().sum()
```

[ ]:
```
sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

[ ]:
```
df.duplicated().sum()
```

[ ]:
```
3
```

[ ]:
```
df.drop_duplicates(inplace=True)
```

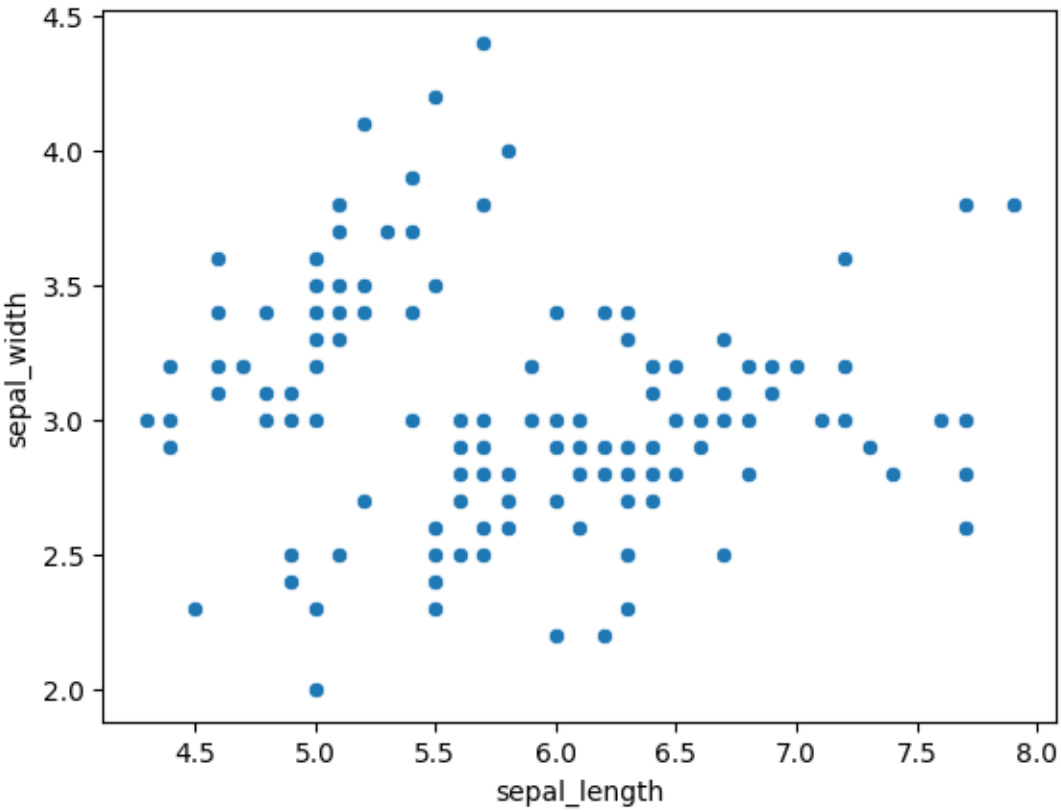
[ ]:
```
df.shape
```

[ ]:
```
(147, 5)
```

Data Visualization

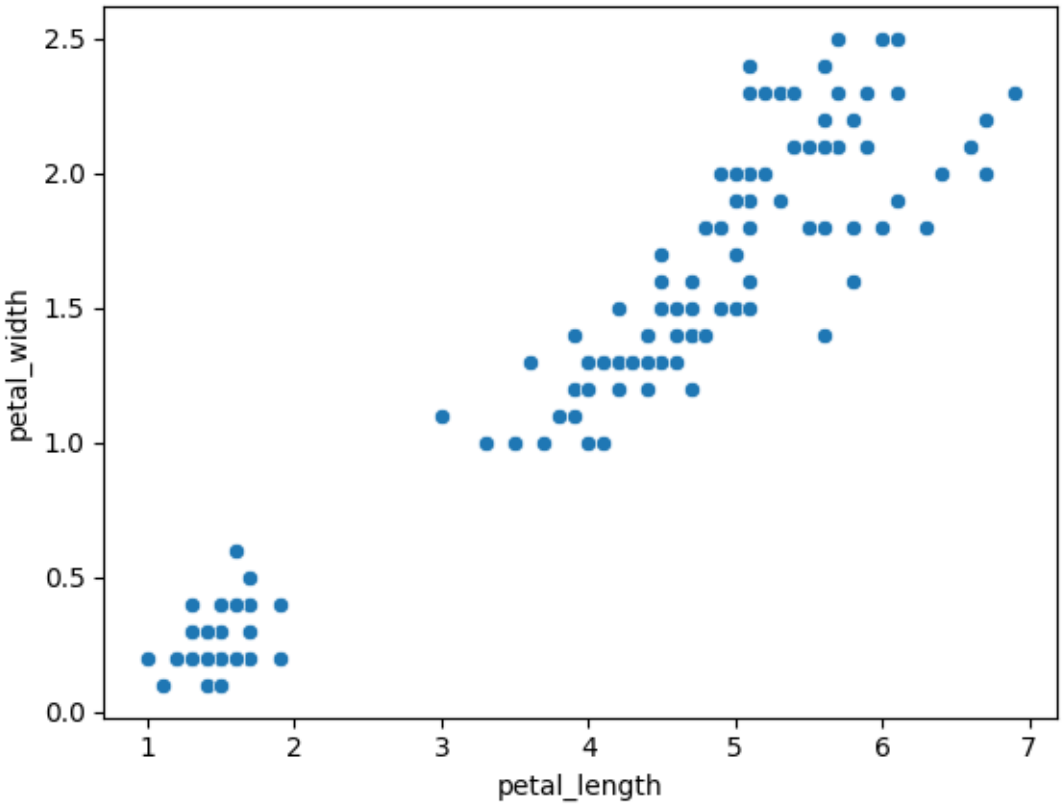
[ ]:
```
sns.scatterplot(x='sepal_length',y='sepal_width',data=df)
```

[ ]:
```
<Axes: xlabel='sepal_length', ylabel='sepal_width'>
```



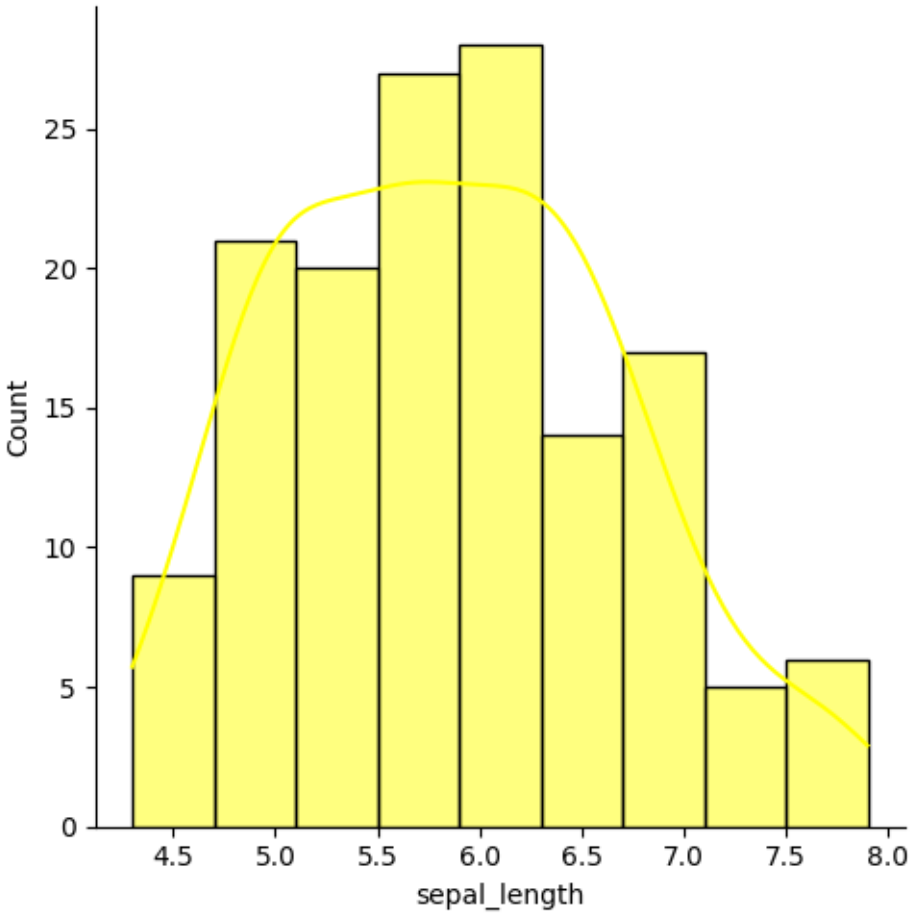
[ ]:
```
sns.scatterplot(x='petal_length',y='petal_width',data=df)
```

[ ]:
```
<Axes: xlabel='petal_length', ylabel='petal_width'>
```



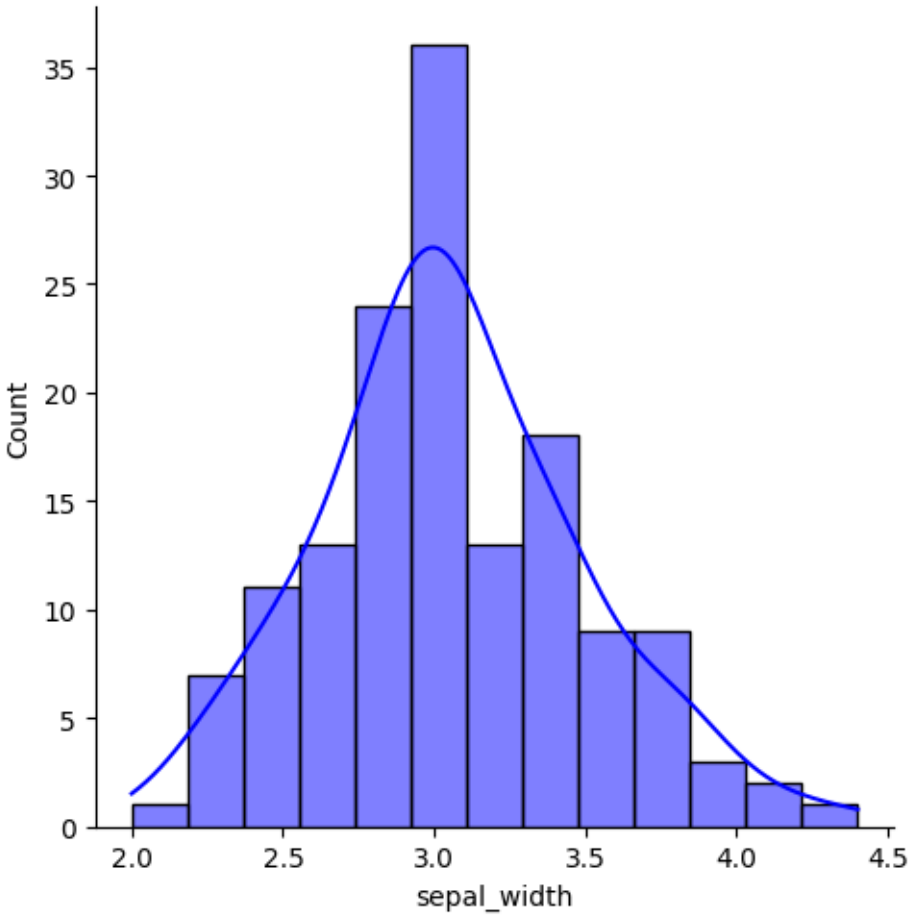
[ ]:
```
sns.displot(df['sepal_length'],kde=True,color='yellow')
```

[ ]:
```
<seaborn.axisgrid.FacetGrid at 0x7d9b57371d80>
```



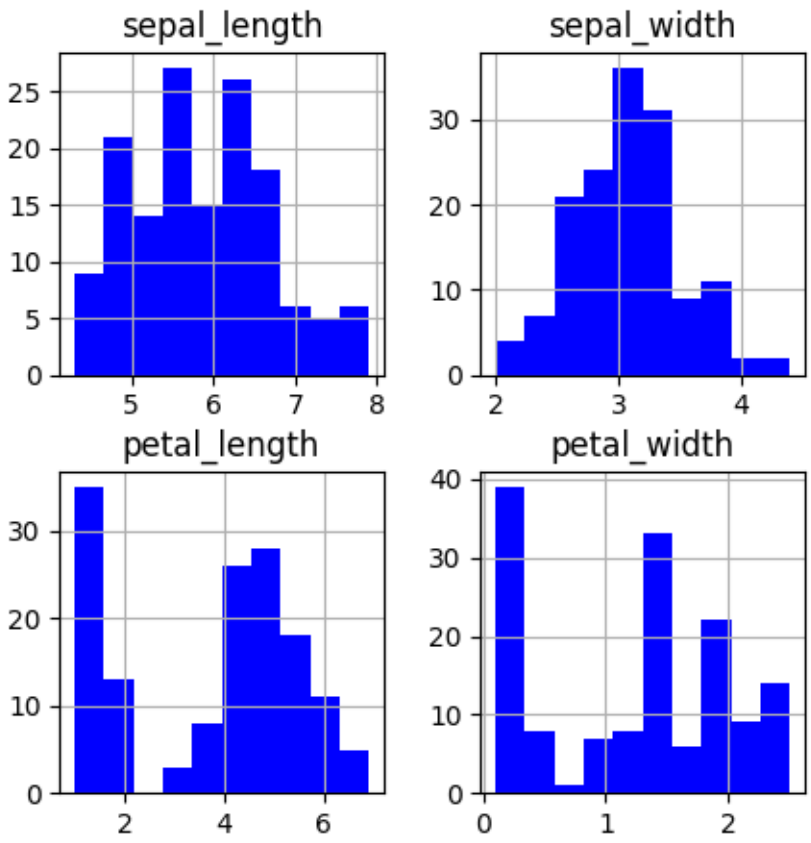
[ ]:
```
sns.displot(df['sepal_width'],kde=True,color='blue')
```

[ ]:
```
<seaborn.axisgrid.FacetGrid at 0x7d9b5503beb0>
```



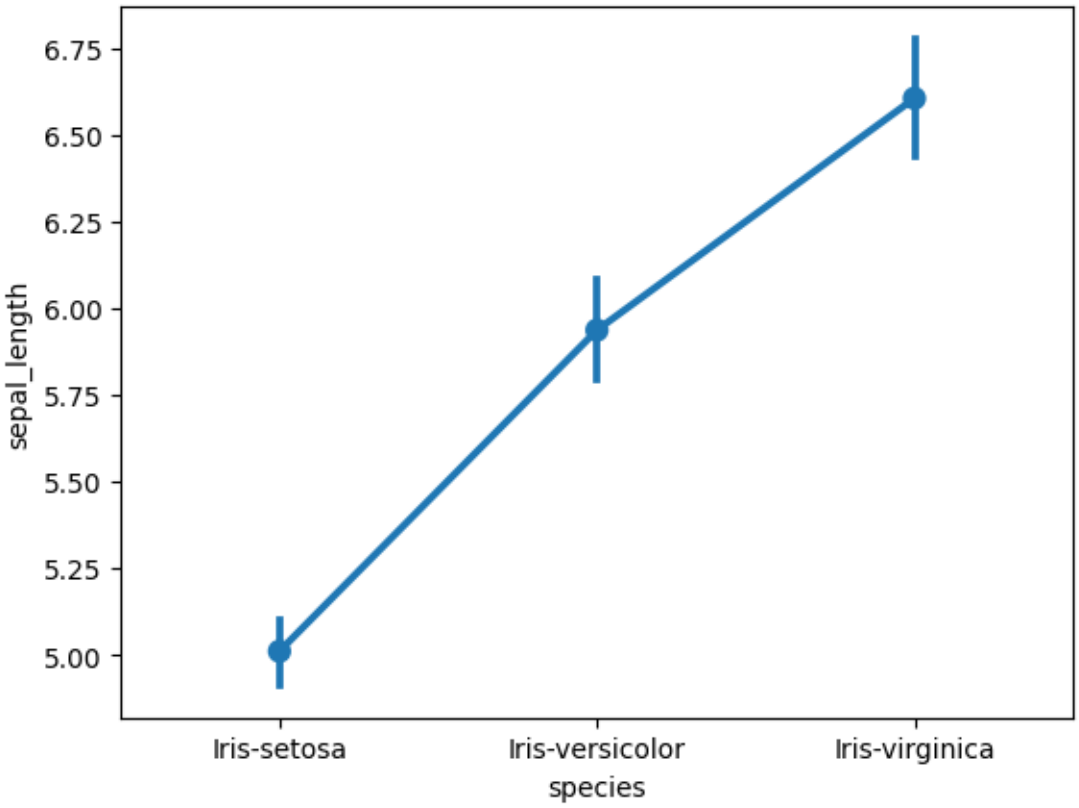
[ ]:
```
df.hist(figsize=(5,5),color='blue')
```

[ ]:
```
array([[<Axes: title={'center': 'sepal_length'}>,
        <Axes: title={'center': 'sepal_width'}>],
       [<Axes: title={'center': 'petal_length'}>,
        <Axes: title={'center': 'petal_width'}>]], dtype=object)
```



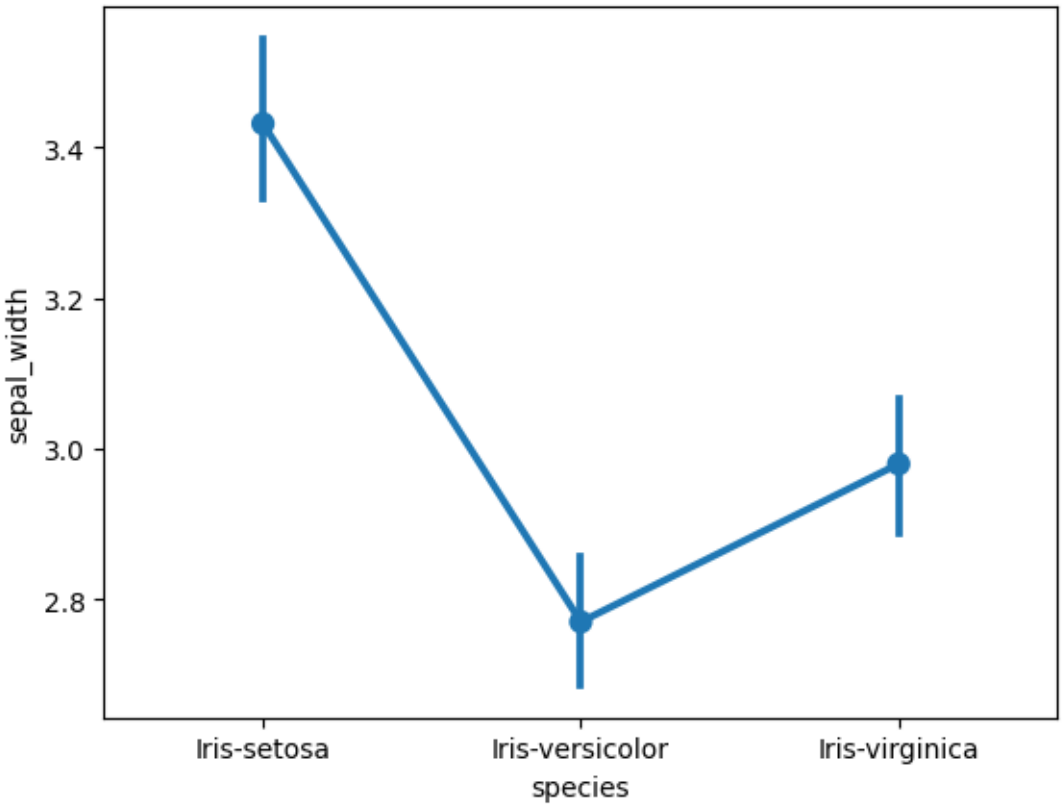
[ ]:
```
sns.pointplot(x='species',y='sepal_length',data=df)
```

[ ]:
```
<Axes: xlabel='species', ylabel='sepal_length'>
```



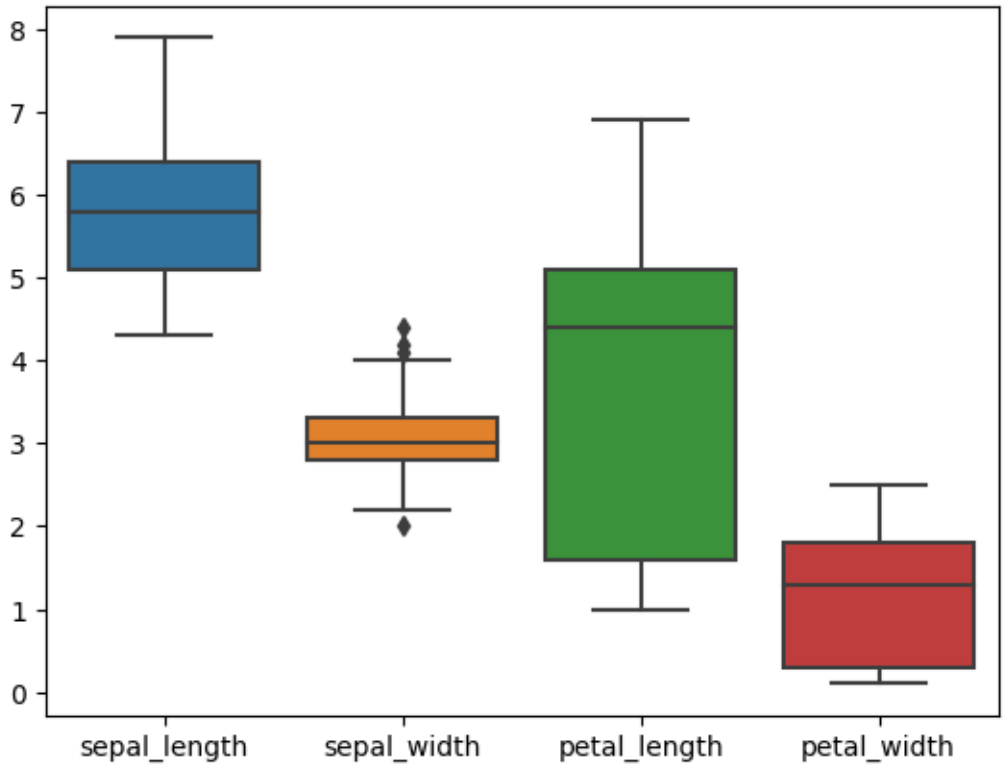
[ ]:
```
sns.pointplot(x='species',y='sepal_width',data=df)
```

[ ]:
```
<Axes: xlabel='species', ylabel='sepal_width'>
```



[ ]:
```
sns.boxplot(df)
```

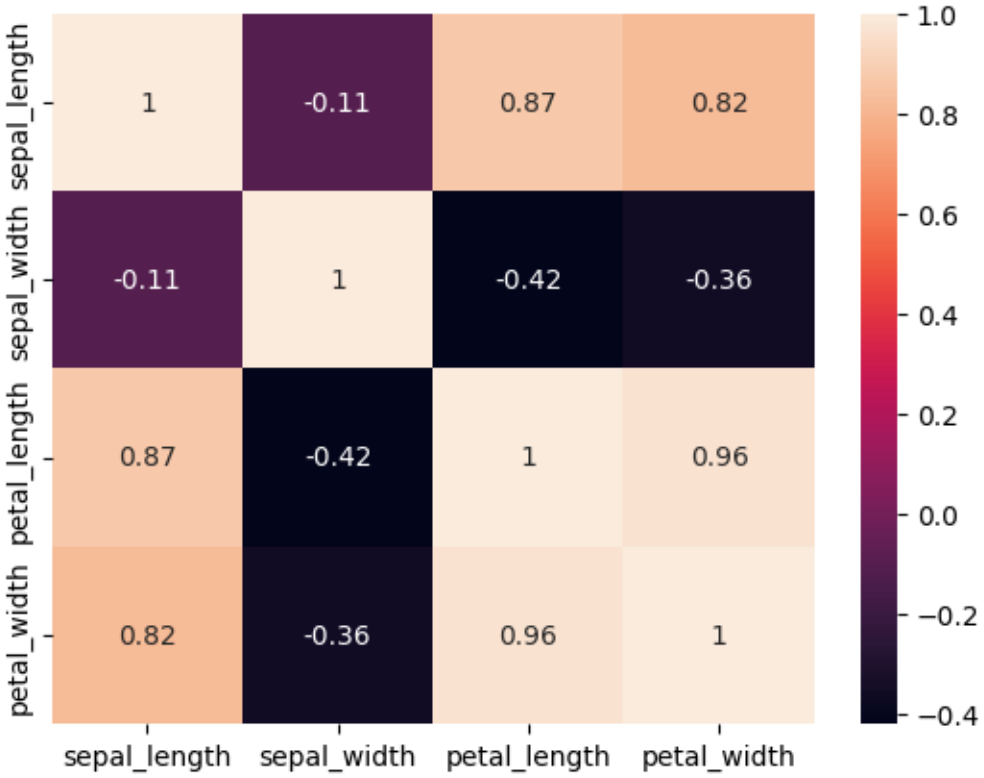
[ ]:
```
<Axes: >
```



[ ]:
```
sns.heatmap(df.corr(),annot=True)
```

```
<ipython-input-20-8df7bcac526d>:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  sns.heatmap(df.corr(),annot=True)
```

[ ]: <Axes: >



Input and Output Separation

[ ]:
```
x=df.iloc[:,:-1].values
x
```

[ ]:
```
array([[5.1, 3.5, 1.4, 0.2],
       [4.9, 3. , 1.4, 0.2],
       [4.7, 3.2, 1.3, 0.2],
       [4.6, 3.1, 1.5, 0.2],
       [5. , 3.6, 1.4, 0.2],
       [5.4, 3.9, 1.7, 0.4],
       [4.6, 3.4, 1.4, 0.3],
       [5. , 3.4, 1.5, 0.2],
       [4.4, 2.9, 1.4, 0.2],
       [4.9, 3.1, 1.5, 0.1],
       [5.4, 3.7, 1.5, 0.2],
       [4.8, 3.4, 1.6, 0.2],
       [4.8, 3. , 1.4, 0.1],
       [4.3, 3. , 1.1, 0.1],
       [5.8, 4. , 1.2, 0.2],
       [5.7, 4.4, 1.5, 0.4],
       [5.4, 3.9, 1.3, 0.4],
```

```
[5.1, 3.5, 1.4, 0.3],
[5.7, 3.8, 1.7, 0.3],
[5.1, 3.8, 1.5, 0.3],
[5.4, 3.4, 1.7, 0.2],
[5.1, 3.7, 1.5, 0.4],
[4.6, 3.6, 1. , 0.2],
[5.1, 3.3, 1.7, 0.5],
[4.8, 3.4, 1.9, 0.2],
[5. , 3. , 1.6, 0.2],
[5. , 3.4, 1.6, 0.4],
[5.2, 3.5, 1.5, 0.2],
[5.2, 3.4, 1.4, 0.2],
[4.7, 3.2, 1.6, 0.2],
[4.8, 3.1, 1.6, 0.2],
[5.4, 3.4, 1.5, 0.4],
[5.2, 4.1, 1.5, 0.1],
[5.5, 4.2, 1.4, 0.2],
[5. , 3.2, 1.2, 0.2],
[5.5, 3.5, 1.3, 0.2],
[4.4, 3. , 1.3, 0.2],
[5.1, 3.4, 1.5, 0.2],
[5. , 3.5, 1.3, 0.3],
[4.5, 2.3, 1.3, 0.3],
[4.4, 3.2, 1.3, 0.2],
[5. , 3.5, 1.6, 0.6],
[5.1, 3.8, 1.9, 0.4],
[4.8, 3. , 1.4, 0.3],
[5.1, 3.8, 1.6, 0.2],
[4.6, 3.2, 1.4, 0.2],
[5.3, 3.7, 1.5, 0.2],
[5. , 3.3, 1.4, 0.2],
[7. , 3.2, 4.7, 1.4],
[6.4, 3.2, 4.5, 1.5],
[6.9, 3.1, 4.9, 1.5],
[5.5, 2.3, 4. , 1.3],
[6.5, 2.8, 4.6, 1.5],
[5.7, 2.8, 4.5, 1.3],
[6.3, 3.3, 4.7, 1.6],
[4.9, 2.4, 3.3, 1. ],
[6.6, 2.9, 4.6, 1.3],
[5.2, 2.7, 3.9, 1.4],
[5. , 2. , 3.5, 1. ],
[5.9, 3. , 4.2, 1.5],
[6. , 2.2, 4. , 1. ],
[6.1, 2.9, 4.7, 1.4],
[5.6, 2.9, 3.6, 1.3],
[6.7, 3.1, 4.4, 1.4],
```

```
[5.6, 3. , 4.5, 1.5],
[5.8, 2.7, 4.1, 1. ],
[6.2, 2.2, 4.5, 1.5],
[5.6, 2.5, 3.9, 1.1],
[5.9, 3.2, 4.8, 1.8],
[6.1, 2.8, 4. , 1.3],
[6.3, 2.5, 4.9, 1.5],
[6.1, 2.8, 4.7, 1.2],
[6.4, 2.9, 4.3, 1.3],
[6.6, 3. , 4.4, 1.4],
[6.8, 2.8, 4.8, 1.4],
[6.7, 3. , 5. , 1.7],
[6. , 2.9, 4.5, 1.5],
[5.7, 2.6, 3.5, 1. ],
[5.5, 2.4, 3.8, 1.1],
[5.5, 2.4, 3.7, 1. ],
[5.8, 2.7, 3.9, 1.2],
[6. , 2.7, 5.1, 1.6],
[5.4, 3. , 4.5, 1.5],
[6. , 3.4, 4.5, 1.6],
[6.7, 3.1, 4.7, 1.5],
[6.3, 2.3, 4.4, 1.3],
[5.6, 3. , 4.1, 1.3],
[5.5, 2.5, 4. , 1.3],
[5.5, 2.6, 4.4, 1.2],
[6.1, 3. , 4.6, 1.4],
[5.8, 2.6, 4. , 1.2],
[5. , 2.3, 3.3, 1. ],
[5.6, 2.7, 4.2, 1.3],
[5.7, 3. , 4.2, 1.2],
[5.7, 2.9, 4.2, 1.3],
[6.2, 2.9, 4.3, 1.3],
[5.1, 2.5, 3. , 1.1],
[5.7, 2.8, 4.1, 1.3],
[6.3, 3.3, 6. , 2.5],
[5.8, 2.7, 5.1, 1.9],
[7.1, 3. , 5.9, 2.1],
[6.3, 2.9, 5.6, 1.8],
[6.5, 3. , 5.8, 2.2],
[7.6, 3. , 6.6, 2.1],
[4.9, 2.5, 4.5, 1.7],
[7.3, 2.9, 6.3, 1.8],
[6.7, 2.5, 5.8, 1.8],
[7.2, 3.6, 6.1, 2.5],
[6.5, 3.2, 5.1, 2. ],
[6.4, 2.7, 5.3, 1.9],
[6.8, 3. , 5.5, 2.1],
```

```
       [5.7, 2.5, 5. , 2. ],
       [5.8, 2.8, 5.1, 2.4],
       [6.4, 3.2, 5.3, 2.3],
       [6.5, 3. , 5.5, 1.8],
       [7.7, 3.8, 6.7, 2.2],
       [7.7, 2.6, 6.9, 2.3],
       [6. , 2.2, 5. , 1.5],
       [6.9, 3.2, 5.7, 2.3],
       [5.6, 2.8, 4.9, 2. ],
       [7.7, 2.8, 6.7, 2. ],
       [6.3, 2.7, 4.9, 1.8],
       [6.7, 3.3, 5.7, 2.1],
       [7.2, 3.2, 6. , 1.8],
       [6.2, 2.8, 4.8, 1.8],
       [6.1, 3. , 4.9, 1.8],
       [6.4, 2.8, 5.6, 2.1],
       [7.2, 3. , 5.8, 1.6],
       [7.4, 2.8, 6.1, 1.9],
       [7.9, 3.8, 6.4, 2. ],
       [6.4, 2.8, 5.6, 2.2],
       [6.3, 2.8, 5.1, 1.5],
       [6.1, 2.6, 5.6, 1.4],
       [7.7, 3. , 6.1, 2.3],
       [6.3, 3.4, 5.6, 2.4],
       [6.4, 3.1, 5.5, 1.8],
       [6. , 3. , 4.8, 1.8],
       [6.9, 3.1, 5.4, 2.1],
       [6.7, 3.1, 5.6, 2.4],
       [6.9, 3.1, 5.1, 2.3],
       [6.8, 3.2, 5.9, 2.3],
       [6.7, 3.3, 5.7, 2.5],
       [6.7, 3. , 5.2, 2.3],
       [6.3, 2.5, 5. , 1.9],
       [6.5, 3. , 5.2, 2. ],
       [6.2, 3.4, 5.4, 2.3],
       [5.9, 3. , 5.1, 1.8]])
```

[ ]:
```
y=df.iloc[:,-1].values
y
```

[ ]:
```
array(['Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
```

```
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-virginica'], dtype=object)
```

Training and testing data

Splitting the data into training and testing data

[ ]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=42,test_size=0.
 ↪30)
```

[ ]:
```
x_train
```

```
[ ]: array([[5.7, 3. , 4.2, 1.2],
            [6.8, 3.2, 5.9, 2.3],
            [6.5, 3.2, 5.1, 2. ],
            [5.1, 3.5, 1.4, 0.2],
            [6.6, 3. , 4.4, 1.4],
            [5.8, 2.7, 4.1, 1. ],
            [5.2, 3.4, 1.4, 0.2],
            [4.4, 3.2, 1.3, 0.2],
            [6. , 2.2, 4. , 1. ],
            [4.8, 3.4, 1.9, 0.2],
            [5. , 3. , 1.6, 0.2],
            [5.1, 3.3, 1.7, 0.5],
            [6. , 2.2, 5. , 1.5],
            [6.7, 3.1, 4.7, 1.5],
            [6.7, 3. , 5.2, 2.3],
            [5.1, 3.8, 1.6, 0.2],
            [5.7, 4.4, 1.5, 0.4],
            [6.3, 2.9, 5.6, 1.8],
            [4.5, 2.3, 1.3, 0.3],
            [5.9, 3.2, 4.8, 1.8],
            [6.5, 3. , 5.5, 1.8],
            [5. , 3.3, 1.4, 0.2],
            [5.7, 2.9, 4.2, 1.3],
            [6.8, 3. , 5.5, 2.1],
            [5.5, 4.2, 1.4, 0.2],
            [5.6, 3. , 4.1, 1.3],
            [6.3, 3.3, 6. , 2.5],
            [5.6, 2.9, 3.6, 1.3],
            [6.4, 2.8, 5.6, 2.1],
            [5.5, 2.4, 3.8, 1.1],
            [5.7, 2.8, 4.5, 1.3],
            [5.4, 3.9, 1.7, 0.4],
            [7.7, 2.8, 6.7, 2. ],
            [5.7, 2.8, 4.1, 1.3],
            [6.4, 3.2, 4.5, 1.5],
            [5.5, 3.5, 1.3, 0.2],
            [5.8, 2.7, 3.9, 1.2],
            [5.7, 2.6, 3.5, 1. ],
            [5. , 3.2, 1.2, 0.2],
            [5.7, 2.5, 5. , 2. ],
            [5. , 3.4, 1.5, 0.2],
            [4.8, 3. , 1.4, 0.3],
            [6.3, 2.5, 4.9, 1.5],
            [6.2, 2.9, 4.3, 1.3],
            [6. , 3.4, 4.5, 1.6],
            [6.3, 2.8, 5.1, 1.5],
            [6.1, 2.6, 5.6, 1.4],
```

```
[6.1, 3. , 4.6, 1.4],
[4.4, 2.9, 1.4, 0.2],
[4.3, 3. , 1.1, 0.1],
[6. , 3. , 4.8, 1.8],
[6.7, 3.3, 5.7, 2.1],
[4.6, 3.1, 1.5, 0.2],
[5.1, 3.5, 1.4, 0.3],
[5. , 3.5, 1.3, 0.3],
[6.4, 2.9, 4.3, 1.3],
[7.4, 2.8, 6.1, 1.9],
[4.6, 3.4, 1.4, 0.3],
[6.4, 2.7, 5.3, 1.9],
[7.1, 3. , 5.9, 2.1],
[4.7, 3.2, 1.3, 0.2],
[6.7, 3.1, 4.4, 1.4],
[6.3, 3.3, 4.7, 1.6],
[7.2, 3.2, 6. , 1.8],
[6.9, 3.1, 4.9, 1.5],
[5.8, 2.8, 5.1, 2.4],
[5.3, 3.7, 1.5, 0.2],
[6.4, 3.2, 5.3, 2.3],
[6.1, 2.9, 4.7, 1.4],
[6.5, 3. , 5.2, 2. ],
[5.5, 2.4, 3.7, 1. ],
[5.9, 3. , 4.2, 1.5],
[5. , 2.3, 3.3, 1. ],
[5. , 3.5, 1.6, 0.6],
[5. , 2. , 3.5, 1. ],
[5.8, 2.6, 4. , 1.2],
[7. , 3.2, 4.7, 1.4],
[5.5, 2.6, 4.4, 1.2],
[7.2, 3.6, 6.1, 2.5],
[5.1, 3.7, 1.5, 0.4],
[5.2, 2.7, 3.9, 1.4],
[6.7, 3. , 5. , 1.7],
[5.2, 4.1, 1.5, 0.1],
[6.7, 3.3, 5.7, 2.5],
[5.1, 3.4, 1.5, 0.2],
[6.9, 3.1, 5.4, 2.1],
[4.9, 3. , 1.4, 0.2],
[6.5, 2.8, 4.6, 1.5],
[6.4, 2.8, 5.6, 2.2],
[7.6, 3. , 6.6, 2.1],
[5.8, 2.7, 5.1, 1.9],
[7.7, 2.6, 6.9, 2.3],
[5.5, 2.5, 4. , 1.3],
[6.8, 2.8, 4.8, 1.4],
```

```
       [6.3, 2.7, 4.9, 1.8],
       [5.9, 3. , 5.1, 1.8],
       [5.4, 3.4, 1.7, 0.2],
       [6.1, 2.8, 4.7, 1.2],
       [6.7, 2.5, 5.8, 1.8],
       [5.8, 4. , 1.2, 0.2],
       [5.6, 2.7, 4.2, 1.3],
       [6.5, 3. , 5.8, 2.2]])
```

[ ]:
```
x_test
```

[ ]:
```
array([[6.1, 3. , 4.9, 1.8],
       [5.5, 2.3, 4. , 1.3],
       [6.7, 3.1, 5.6, 2.4],
       [5.1, 3.8, 1.5, 0.3],
       [4.9, 2.5, 4.5, 1.7],
       [4.8, 3. , 1.4, 0.1],
       [6. , 2.9, 4.5, 1.5],
       [5.4, 3.4, 1.5, 0.4],
       [6. , 2.7, 5.1, 1.6],
       [4.9, 3.1, 1.5, 0.1],
       [5. , 3.4, 1.6, 0.4],
       [5.1, 2.5, 3. , 1.1],
       [6.3, 2.5, 5. , 1.9],
       [5.6, 2.5, 3.9, 1.1],
       [6.3, 3.4, 5.6, 2.4],
       [6.2, 2.2, 4.5, 1.5],
       [5.7, 3.8, 1.7, 0.3],
       [6.1, 2.8, 4. , 1.3],
       [6.2, 2.8, 4.8, 1.8],
       [4.8, 3.1, 1.6, 0.2],
       [4.7, 3.2, 1.6, 0.2],
       [7.3, 2.9, 6.3, 1.8],
       [4.4, 3. , 1.3, 0.2],
       [6.9, 3.2, 5.7, 2.3],
       [4.9, 2.4, 3.3, 1. ],
       [4.6, 3.6, 1. , 0.2],
       [5.6, 3. , 4.5, 1.5],
       [7.7, 3. , 6.1, 2.3],
       [5.4, 3. , 4.5, 1.5],
       [4.8, 3.4, 1.6, 0.2],
       [6.4, 3.1, 5.5, 1.8],
       [4.6, 3.2, 1.4, 0.2],
       [5.6, 2.8, 4.9, 2. ],
       [5.2, 3.5, 1.5, 0.2],
       [5. , 3.6, 1.4, 0.2],
       [7.2, 3. , 5.8, 1.6],
```

```
       [7.9, 3.8, 6.4, 2. ],
       [6.2, 3.4, 5.4, 2.3],
       [5.1, 3.8, 1.9, 0.4],
       [5.4, 3.9, 1.3, 0.4],
       [5.4, 3.7, 1.5, 0.2],
       [7.7, 3.8, 6.7, 2.2],
       [6.3, 2.3, 4.4, 1.3],
       [6.9, 3.1, 5.1, 2.3],
       [6.6, 2.9, 4.6, 1.3]])
```

[ ]:
```
y_train
```

[ ]:
```
array(['Iris-versicolor', 'Iris-virginica', 'Iris-virginica',
       'Iris-setosa', 'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',
       'Iris-setosa', 'Iris-versicolor', 'Iris-setosa', 'Iris-setosa',
       'Iris-setosa', 'Iris-virginica', 'Iris-versicolor',
       'Iris-virginica', 'Iris-setosa', 'Iris-setosa', 'Iris-virginica',
       'Iris-setosa', 'Iris-versicolor', 'Iris-virginica', 'Iris-setosa',
       'Iris-versicolor', 'Iris-virginica', 'Iris-setosa',
       'Iris-versicolor', 'Iris-virginica', 'Iris-versicolor',
       'Iris-virginica', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-setosa', 'Iris-virginica', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-setosa', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-setosa', 'Iris-virginica', 'Iris-setosa',
       'Iris-setosa', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-virginica', 'Iris-virginica',
       'Iris-versicolor', 'Iris-setosa', 'Iris-setosa', 'Iris-virginica',
       'Iris-virginica', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-versicolor', 'Iris-virginica', 'Iris-setosa',
       'Iris-virginica', 'Iris-virginica', 'Iris-setosa',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-virginica',
       'Iris-versicolor', 'Iris-virginica', 'Iris-setosa',
       'Iris-virginica', 'Iris-versicolor', 'Iris-virginica',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-setosa', 'Iris-versicolor', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-versicolor', 'Iris-virginica',
       'Iris-setosa', 'Iris-versicolor', 'Iris-versicolor', 'Iris-setosa',
       'Iris-virginica', 'Iris-setosa', 'Iris-virginica', 'Iris-setosa',
       'Iris-versicolor', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-virginica', 'Iris-versicolor',
       'Iris-versicolor', 'Iris-virginica', 'Iris-virginica',
       'Iris-setosa', 'Iris-versicolor', 'Iris-virginica', 'Iris-setosa',
       'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

[ ]:
```
y_test
```

```
[ ]: array(['Iris-virginica', 'Iris-versicolor', 'Iris-virginica',
       'Iris-setosa', 'Iris-virginica', 'Iris-setosa', 'Iris-versicolor',
       'Iris-setosa', 'Iris-versicolor', 'Iris-setosa', 'Iris-setosa',
       'Iris-versicolor', 'Iris-virginica', 'Iris-versicolor',
       'Iris-virginica', 'Iris-versicolor', 'Iris-setosa',
       'Iris-versicolor', 'Iris-virginica', 'Iris-setosa', 'Iris-setosa',
       'Iris-virginica', 'Iris-setosa', 'Iris-virginica',
       'Iris-versicolor', 'Iris-setosa', 'Iris-versicolor',
       'Iris-virginica', 'Iris-versicolor', 'Iris-setosa',
       'Iris-virginica', 'Iris-setosa', 'Iris-virginica', 'Iris-setosa',
       'Iris-setosa', 'Iris-virginica', 'Iris-virginica',
       'Iris-virginica', 'Iris-setosa', 'Iris-setosa', 'Iris-setosa',
       'Iris-virginica', 'Iris-versicolor', 'Iris-virginica',
       'Iris-versicolor'], dtype=object)
```

Normalization using Standard Scaler

```
[ ]: from sklearn.preprocessing import StandardScaler
     scaler=StandardScaler()
     scaler.fit(x_train)
     x_train=scaler.transform(x_train)
     x_test=scaler.transform(x_test)
```

```
[ ]: x_train
```

```
[ ]: array([[-0.23690478, -0.08738704,  0.20877397, -0.03330265],
       [ 1.16208031,  0.38174338,  1.19788343,  1.46132027],
       [ 0.78053892,  0.38174338,  0.73242015,  1.05369584],
       [-0.99998756,  1.085439  , -1.42034749, -1.39205076],
       [ 0.90771938, -0.08738704,  0.32513979,  0.23844697],
       [-0.10972432, -0.79108266,  0.15059106, -0.30505227],
       [-0.8728071 ,  0.85087379, -1.42034749, -1.39205076],
       [-1.8902508 ,  0.38174338, -1.4785304 , -1.39205076],
       [ 0.14463661, -1.9639087 ,  0.09240815, -0.30505227],
       [-1.38152895,  0.85087379, -1.12943294, -1.39205076],
       [-1.12716803, -0.08738704, -1.30398167, -1.39205076],
       [-0.99998756,  0.61630858, -1.24579876, -0.98442633],
       [ 0.14463661, -1.9639087 ,  0.67423724,  0.37432178],
       [ 1.03489985,  0.14717817,  0.49968851,  0.37432178],
       [ 1.03489985, -0.08738704,  0.79060306,  1.46132027],
       [-0.99998756,  1.78913462, -1.30398167, -1.39205076],
       [-0.23690478,  3.19652586, -1.36216458, -1.12030114],
       [ 0.52617799, -0.32195225,  1.0233347 ,  0.78194622],
       [-1.76307034, -1.72934349, -1.4785304 , -1.25617595],
       [ 0.01745614,  0.38174338,  0.55787142,  0.78194622],
       [ 0.78053892, -0.08738704,  0.96515179,  0.78194622],
       [-1.12716803,  0.61630858, -1.42034749, -1.39205076],
```

```
[-0.23690478, -0.32195225,  0.20877397,  0.10257216],
[ 1.16208031, -0.08738704,  0.96515179,  1.18957065],
[-0.49126571,  2.72739545, -1.42034749, -1.39205076],
[-0.36408525, -0.08738704,  0.15059106,  0.10257216],
[ 0.52617799,  0.61630858,  1.25606633,  1.7330699 ],
[-0.36408525, -0.32195225, -0.14032349,  0.10257216],
[ 0.65335846, -0.55651745,  1.0233347 ,  1.18957065],
[-0.49126571, -1.49477828, -0.02395767, -0.16917746],
[-0.23690478, -0.55651745,  0.3833227 ,  0.10257216],
[-0.61844617,  2.02369983, -1.24579876, -1.12030114],
[ 2.30670448, -0.55651745,  1.6633467 ,  1.05369584],
[-0.23690478, -0.55651745,  0.15059106,  0.10257216],
[ 0.65335846,  0.38174338,  0.3833227 ,  0.37432178],
[-0.49126571,  1.085439  , -1.4785304 , -1.39205076],
[-0.10972432, -0.79108266,  0.03422524, -0.03330265],
[-0.23690478, -1.02564787, -0.1985064 , -0.30505227],
[-1.12716803,  0.38174338, -1.53671331, -1.39205076],
[-0.23690478, -1.26021307,  0.67423724,  1.05369584],
[-1.12716803,  0.85087379, -1.36216458, -1.39205076],
[-1.38152895, -0.08738704, -1.42034749, -1.25617595],
[ 0.52617799, -1.26021307,  0.61605433,  0.37432178],
[ 0.39899753, -0.32195225,  0.26695688,  0.10257216],
[ 0.14463661,  0.85087379,  0.3833227 ,  0.5101966 ],
[ 0.52617799, -0.55651745,  0.73242015,  0.37432178],
[ 0.27181707, -1.02564787,  1.0233347 ,  0.23844697],
[ 0.27181707, -0.08738704,  0.44150561,  0.23844697],
[-1.8902508 , -0.32195225, -1.42034749, -1.39205076],
[-2.01743127, -0.08738704, -1.59489622, -1.52792557],
[ 0.14463661, -0.08738704,  0.55787142,  0.78194622],
[ 1.03489985,  0.61630858,  1.08151761,  1.18957065],
[-1.63588988,  0.14717817, -1.36216458, -1.39205076],
[-0.99998756,  1.085439  , -1.42034749, -1.25617595],
[-1.12716803,  1.085439  , -1.4785304 , -1.25617595],
[ 0.65335846, -0.32195225,  0.26695688,  0.10257216],
[ 1.92516309, -0.55651745,  1.31424924,  0.91782103],
[-1.63588988,  0.85087379, -1.42034749, -1.25617595],
[ 0.65335846, -0.79108266,  0.84878597,  0.91782103],
[ 1.5436217 , -0.08738704,  1.19788343,  1.18957065],
[-1.50870941,  0.38174338, -1.4785304 , -1.39205076],
[ 1.03489985,  0.14717817,  0.32513979,  0.23844697],
[ 0.52617799,  0.61630858,  0.49968851,  0.5101966 ],
[ 1.67080216,  0.38174338,  1.25606633,  0.78194622],
[ 1.28926077,  0.14717817,  0.61605433,  0.37432178],
[-0.10972432, -0.55651745,  0.73242015,  1.59719509],
[-0.74562664,  1.55456941, -1.36216458, -1.39205076],
[ 0.65335846,  0.38174338,  0.84878597,  1.46132027],
[ 0.27181707, -0.32195225,  0.49968851,  0.23844697],
```

```
       [ 0.78053892, -0.08738704,  0.79060306,  1.05369584],
       [-0.49126571, -1.49477828, -0.08214058, -0.30505227],
       [ 0.01745614, -0.08738704,  0.20877397,  0.37432178],
       [-1.12716803, -1.72934349, -0.31487221, -0.30505227],
       [-1.12716803,  1.085439  , -1.30398167, -0.84855152],
       [-1.12716803, -2.43303911, -0.1985064 , -0.30505227],
       [-0.10972432, -1.02564787,  0.09240815, -0.03330265],
       [ 1.41644124,  0.38174338,  0.49968851,  0.23844697],
       [-0.49126571, -1.02564787,  0.32513979, -0.03330265],
       [ 1.67080216,  1.32000421,  1.31424924,  1.7330699 ],
       [-0.99998756,  1.55456941, -1.36216458, -1.12030114],
       [-0.8728071 , -0.79108266,  0.03422524,  0.23844697],
       [ 1.03489985, -0.08738704,  0.67423724,  0.64607141],
       [-0.8728071 ,  2.49283024, -1.36216458, -1.52792557],
       [ 1.03489985,  0.61630858,  1.08151761,  1.7330699 ],
       [-0.99998756,  0.85087379, -1.36216458, -1.39205076],
       [ 1.28926077,  0.14717817,  0.90696888,  1.18957065],
       [-1.25434849, -0.08738704, -1.42034749, -1.39205076],
       [ 0.78053892, -0.55651745,  0.44150561,  0.37432178],
       [ 0.65335846, -0.55651745,  1.0233347 ,  1.32544546],
       [ 2.17952401, -0.08738704,  1.60516379,  1.18957065],
       [-0.10972432, -0.79108266,  0.73242015,  0.91782103],
       [ 2.30670448, -1.02564787,  1.77971252,  1.46132027],
       [-0.49126571, -1.26021307,  0.09240815,  0.10257216],
       [ 1.16208031, -0.55651745,  0.55787142,  0.23844697],
       [ 0.52617799, -0.79108266,  0.61605433,  0.78194622],
       [ 0.01745614, -0.08738704,  0.73242015,  0.78194622],
       [-0.61844617,  0.85087379, -1.24579876, -1.39205076],
       [ 0.27181707, -0.55651745,  0.49968851, -0.03330265],
       [ 1.03489985, -1.26021307,  1.13970052,  0.78194622],
       [-0.10972432,  2.25826503, -1.53671331, -1.39205076],
       [-0.36408525, -0.79108266,  0.20877397,  0.10257216],
       [ 0.78053892, -0.08738704,  1.13970052,  1.32544546]])
```

[ ]:
```
x_test
```

[ ]:
```
array([[ 0.27181707, -0.08738704,  0.61605433,  0.78194622],
       [-0.49126571, -1.72934349,  0.09240815,  0.10257216],
       [ 1.03489985,  0.14717817,  1.0233347 ,  1.59719509],
       [-0.99998756,  1.78913462, -1.36216458, -1.25617595],
       [-1.25434849, -1.26021307,  0.3833227 ,  0.64607141],
       [-1.38152895, -0.08738704, -1.42034749, -1.52792557],
       [ 0.14463661, -0.32195225,  0.3833227 ,  0.37432178],
       [-0.61844617,  0.85087379, -1.36216458, -1.12030114],
       [ 0.14463661, -0.79108266,  0.73242015,  0.5101966 ],
       [-1.25434849,  0.14717817, -1.36216458, -1.52792557],
       [-1.12716803,  0.85087379, -1.30398167, -1.12030114],
```

```
[-0.99998756, -1.26021307, -0.48942094, -0.16917746],
[ 0.52617799, -1.26021307,  0.67423724,  0.91782103],
[-0.36408525, -1.26021307,  0.03422524, -0.16917746],
[ 0.52617799,  0.85087379,  1.0233347 ,  1.59719509],
[ 0.39899753, -1.9639087 ,  0.3833227 ,  0.37432178],
[-0.23690478,  1.78913462, -1.24579876, -1.25617595],
[ 0.27181707, -0.55651745,  0.09240815,  0.10257216],
[ 0.39899753, -0.55651745,  0.55787142,  0.78194622],
[-1.38152895,  0.14717817, -1.30398167, -1.39205076],
[-1.50870941,  0.38174338, -1.30398167, -1.39205076],
[ 1.79798262, -0.32195225,  1.43061506,  0.78194622],
[-1.8902508 , -0.08738704, -1.4785304 , -1.39205076],
[ 1.28926077,  0.38174338,  1.08151761,  1.46132027],
[-1.25434849, -1.49477828, -0.31487221, -0.30505227],
[-1.63588988,  1.32000421, -1.65307913, -1.39205076],
[-0.36408525, -0.08738704,  0.3833227 ,  0.37432178],
[ 2.30670448, -0.08738704,  1.31424924,  1.46132027],
[-0.61844617, -0.08738704,  0.3833227 ,  0.37432178],
[-1.38152895,  0.85087379, -1.30398167, -1.39205076],
[ 0.65335846,  0.14717817,  0.96515179,  0.78194622],
[-1.63588988,  0.38174338, -1.42034749, -1.39205076],
[-0.36408525, -0.55651745,  0.61605433,  1.05369584],
[-0.8728071 ,  1.085439  , -1.36216458, -1.39205076],
[-1.12716803,  1.32000421, -1.42034749, -1.39205076],
[ 1.67080216, -0.08738704,  1.13970052,  0.5101966 ],
[ 2.5610654 ,  1.78913462,  1.48879797,  1.05369584],
[ 0.39899753,  0.85087379,  0.90696888,  1.46132027],
[-0.99998756,  1.78913462, -1.12943294, -1.12030114],
[-0.61844617,  2.02369983, -1.4785304 , -1.12030114],
[-0.61844617,  1.55456941, -1.36216458, -1.39205076],
[ 2.30670448,  1.78913462,  1.6633467 ,  1.32544546],
[ 0.52617799, -1.72934349,  0.32513979,  0.10257216],
[ 1.28926077,  0.14717817,  0.73242015,  1.46132027],
[ 0.90771938, -0.32195225,  0.44150561,  0.10257216]])
```

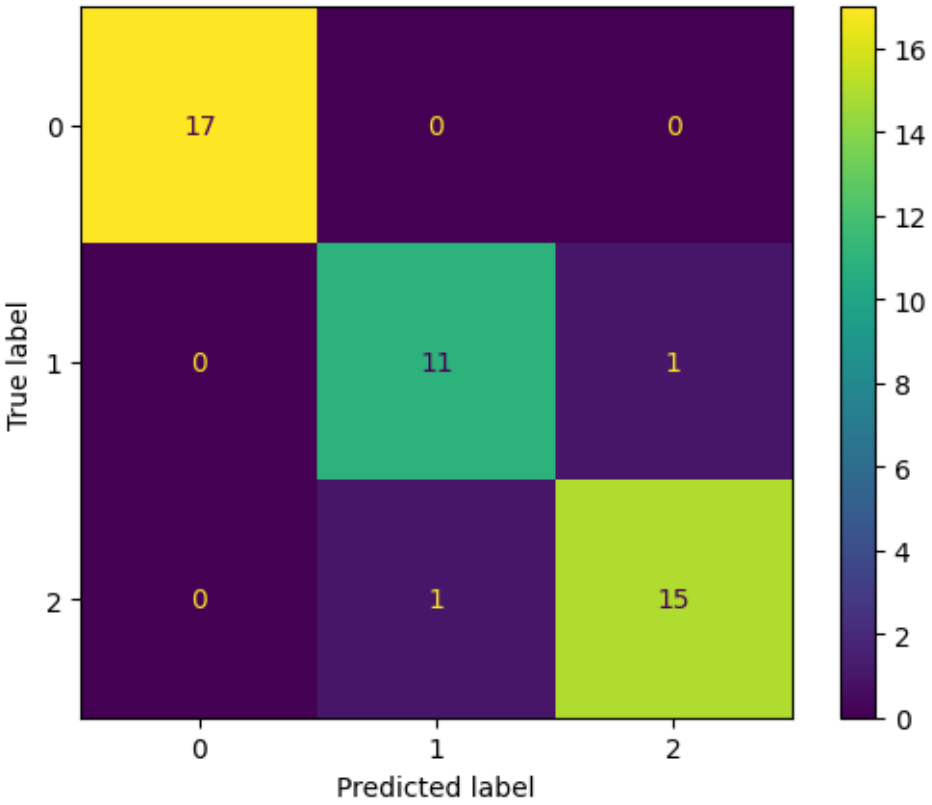
Model Creation

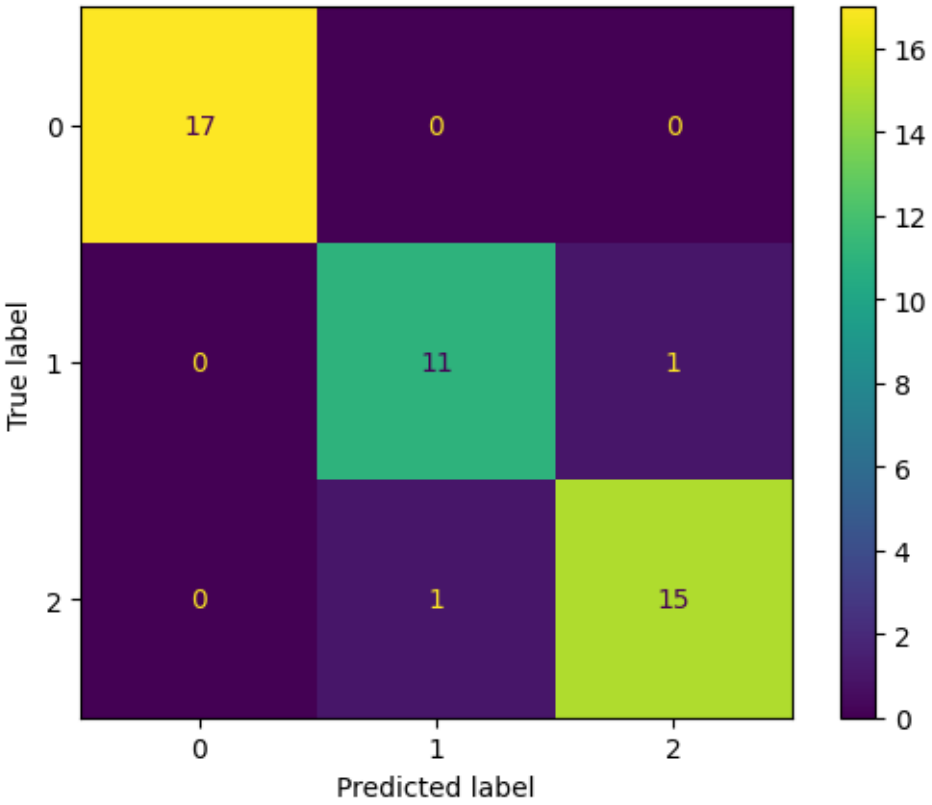
[ ]:
```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
knn=KNeighborsClassifier(n_neighbors=7)
svmm=SVC()
dec=DecisionTreeClassifier(criterion='entropy')
model=RandomForestClassifier(n_estimators=10,criterion='entropy')
lst_model=[knn,svmm,dec,model]
```

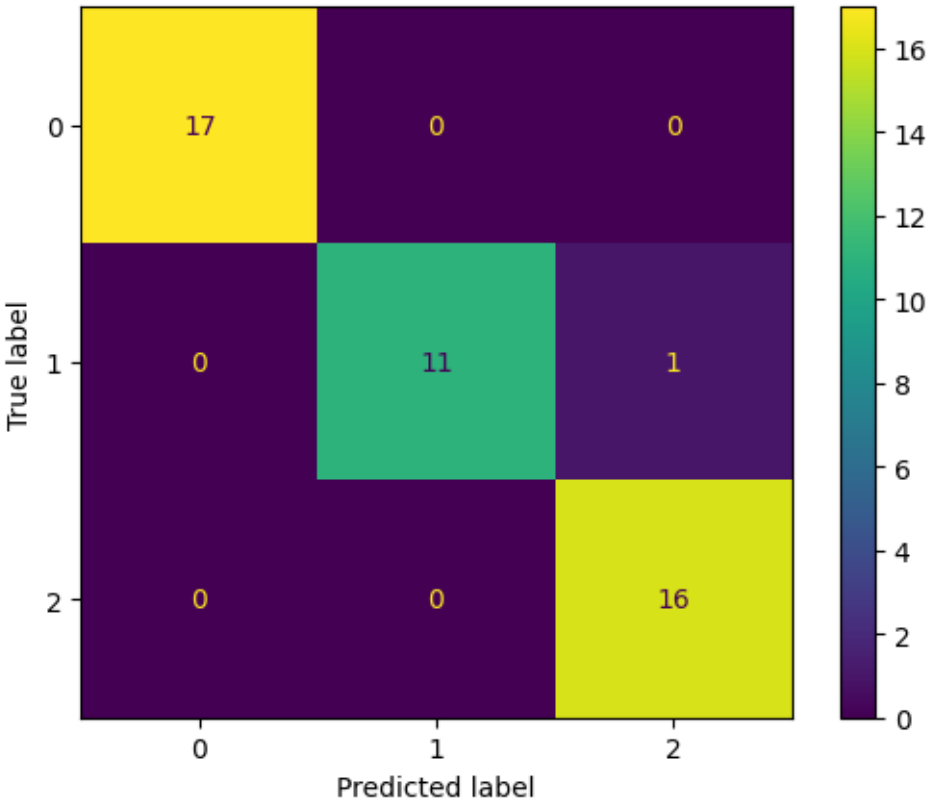
[ ]:
```
from sklearn.metrics import␣
 ↪confusion_matrix,accuracy_score,classification_report,ConfusionMatrixDisplay
for i in lst_model:
  print(i)
  i.fit(x_train,y_train)
  y_pred=i.predict(x_test)
  y_pred
  print(accuracy_score(y_test,y_pred))
  print("****************")
  print(classification_report(y_test,y_pred))
  print("****************")
  print(confusion_matrix(y_test,y_pred))
  print("****************")
  result=confusion_matrix(y_test,y_pred)
  labels=[0,1,2]
  cmd=ConfusionMatrixDisplay(result,display_labels=labels)
  cmd.plot()
```

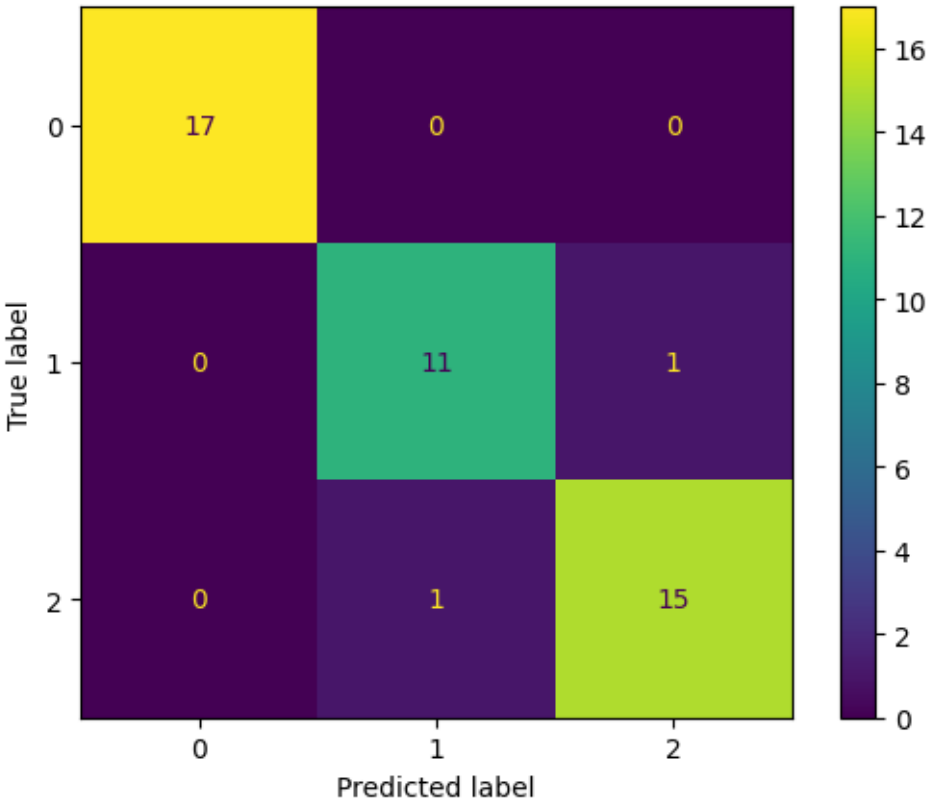
```
KNeighborsClassifier(n_neighbors=7)
0.9555555555555556
***************
                precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        17
Iris-versicolor       0.92      0.92      0.92        12
 Iris-virginica       0.94      0.94      0.94        16

       accuracy                           0.96        45
      macro avg       0.95      0.95      0.95        45
   weighted avg       0.96      0.96      0.96        45

***************
[[17  0  0]
 [ 0 11  1]
 [ 0  1 15]]
***************
SVC()
0.9555555555555556
***************
                precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        17
Iris-versicolor       0.92      0.92      0.92        12
 Iris-virginica       0.94      0.94      0.94        16

       accuracy                           0.96        45
      macro avg       0.95      0.95      0.95        45
```

```
   weighted avg       0.96      0.96      0.96        45

***************
[[17  0  0]
 [ 0 11  1]
 [ 0  1 15]]
***************
DecisionTreeClassifier(criterion='entropy')
0.9777777777777777
***************
                precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        17
Iris-versicolor       1.00      0.92      0.96        12
 Iris-virginica       0.94      1.00      0.97        16

       accuracy                           0.98        45
      macro avg       0.98      0.97      0.98        45
   weighted avg       0.98      0.98      0.98        45

***************
[[17  0  0]
 [ 0 11  1]
 [ 0  0 16]]
***************
RandomForestClassifier(criterion='entropy', n_estimators=10)
0.9555555555555556
***************
                precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        17
Iris-versicolor       0.92      0.92      0.92        12
 Iris-virginica       0.94      0.94      0.94        16

       accuracy                           0.96        45
      macro avg       0.95      0.95      0.95        45
   weighted avg       0.96      0.96      0.96        45

***************
[[17  0  0]
 [ 0 11  1]
 [ 0  1 15]]
***************
```


17
0
0
0
11
1
0
1
15
0
1
2
True label
Predicted label
16
14
12
10
8
6
4
2
0


17
0
0
0
11
1
0
1
15
0
1
2
True label
Predicted label
16
14
12
10
8
6
4
2
0


17
0
0
0
11
1
0
0
16
0
1
2
True label
Predicted label
16
14
12
10
8
6
4
2
0


17
0
0
0
11
1
0
1
15
0
1
2
True label
Predicted label
16
14
12
10
8
6
4
2
0