```python
import seaborn as sns
import pandas as pd
import numpy as np
```

```python
df=sns.load_dataset('iris')
```

```python
df.head()
```

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

```python
df['species'].unique()
```

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

```python
df.isnull().sum()
```

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

```python
df=df[df['species']!='setosa']
```

```python
df.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
50	7.0	3.2	4.7	1.4	versicolor
51	6.4	3.2	4.5	1.5	versicolor
52	6.9	3.1	4.9	1.5	versicolor
53	5.5	2.3	4.0	1.3	versicolor
54	6.5	2.8	4.6	1.5	versicolor

```python
df['species']=df['species'].map({'versicolor':0,'virginica':1})
```

```python
df.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
50	7.0	3.2	4.7	1.4	0
51	6.4	3.2	4.5	1.5	0
52	6.9	3.1	4.9	1.5	0
53	5.5	2.3	4.0	1.3	0
54	6.5	2.8	4.6	1.5	0

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

```python
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.25,random_state=42)
```

```python
from sklearn.linear_model import LogisticRegression
```

```python
lm=LogisticRegression()
```

```python
from sklearn.model_selection import GridSearchCV
parameter={'penalty':['11','12','elasticnet'],'c':[1,2,3,4,5,6,10,20]}
```

```python
lm.fit(X_train,y_train)
```

```
▼ LogisticRegression
LogisticRegression()
```

```python
pred=lm.predict(X_test)
```

```python
from sklearn.metrics import confusion_matrix, classification_report
```

```python
confusion_matrix(y_test,pred)
```

```
array([[13,  1],
       [ 1, 10]])
```

```python
from sklearn.metrics import accuracy_score,classification_report
```

```python
score=accuracy_score(pred,y_test)
```

```python
print(score)
```

```
0.92
```

```python
print(classification_report(pred,y_test))
```

```
              precision    recall  f1-score   support

           0       0.93      0.93      0.93        14
           1       0.91      0.91      0.91        11

    accuracy                           0.92        25
   macro avg       0.92      0.92      0.92        25
weighted avg       0.92      0.92      0.92        25
```