In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data
names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'Class'
dataset = pd.read_csv(url, names=names)
```

In [3]:
```
dataset.head(3)
```

Out[3]:

	sepal-length	sepal-width	petal-length	petal-width	Class
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

In [4]:
```
x = dataset.drop('Class', 1)
y = dataset['Class']
```

```
C:\Users\ambat\AppData\Local\Temp\ipykernel_5004\1449409176.py:1: FutureWarni
ng: In a future version of pandas all arguments of DataFrame.drop except for
the argument 'labels' will be keyword-only.
  x = dataset.drop('Class', 1)
```

In [5]:
```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
y = le.fit_transform(y)
```

In [6]:
```
# Splitting the dataset into the Training set and Test set
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2, rando
```

In [7]:
```
from sklearn.decomposition import TruncatedSVD
svd = TruncatedSVD(n_components=2, n_iter=8)
X_train = svd.fit_transform(X_train)
X_test = svd.transform(X_test)
```

In [8]:
```
explained_variance = svd.explained_variance_ratio_
```

In [9]:
```
explained_variance
```

Out[9]:
```
array([0.54666515, 0.43243966])
```

In [10]:
```
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression

classifier = LogisticRegression()
classifier.fit(X_train, y_train)

# Predicting the Test set results
y_pred = classifier.predict(X_test)
```

In [11]:
```
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score

cm = confusion_matrix(y_test, y_pred)
print(cm)
print('Accuracy is ',accuracy_score(y_test, y_pred))
```

```
[[11  0  0]
 [ 0 13  0]
 [ 0  0  6]]
Accuracy is  1.0
```