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

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, confusion_matrix, precision_score, recall_score

url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"

column_names = ["Sepal_Length", "Sepal_Width", "Petal_Length", "Petal_Width", "Class"]
iris = pd.read_csv(url, names=column_names)

print(iris.head())

label_encoder = LabelEncoder()
iris["Class"] = label_encoder.fit_transform(iris["Class"])

print(iris.isnull().sum())

X = iris.drop(columns=["Class"])
Y = iris["Class"]

X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2, random_state=42)

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
```

```
X_test_scaled = scaler.transform(X_test)

naive_bayes = GaussianNB()
naive_bayes.fit(X_train_scaled, y_train)



y_pred_train = naive_bayes.predict(X_train_scaled)
y_pred_test = naive_bayes.predict(X_test_scaled)

train_accuracy = accuracy_score(y_train, y_pred_train)
test_accuracy = accuracy_score(y_test, y_pred_test)
precision = precision_score(y_test, y_pred_test, average="micro")
recall = recall_score(y_test, y_pred_test, average="micro")

cm = confusion_matrix(y_test, y_pred_test)

print("\nTraining Accuracy:", train_accuracy)
print("Testing Accuracy:", test_accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("\nConfusion Matrix:\n", cm)

plt.figure(figsize=(5, 4))
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.title("Confusion Matrix")
plt.show()
```