

```

from sklearn.linear_model import Perceptron
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
# Load wine quality dataset
wine_data = pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-red.csv', sep=';')

# Separate features and target
X = wine_data.iloc[:, :-1]
y = wine_data.iloc[:, -1]

# Split into train and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Initialize Perceptron Classifier
perceptron = Perceptron()

# Train Perceptron Classifier
perceptron.fit(X_train, y_train)

# Predict wine quality for test set
y_pred = perceptron.predict(X_test)

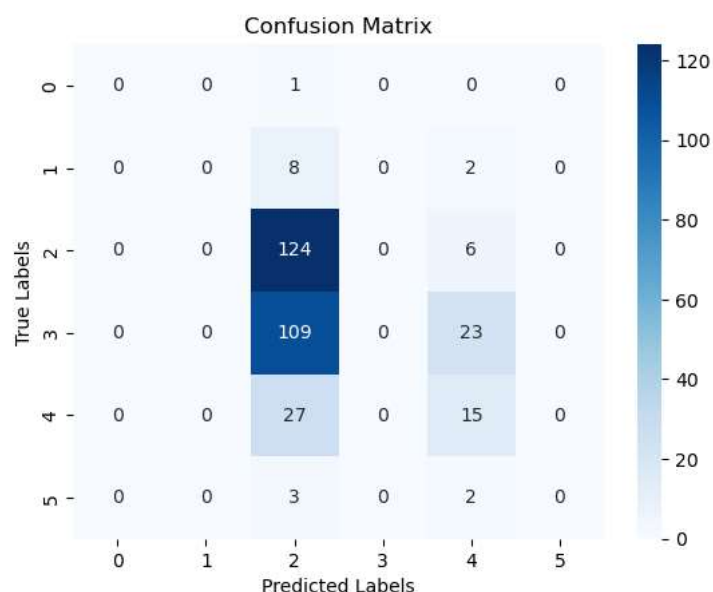
# Print accuracy
print(f"Perceptron Classifier accuracy: {accuracy_score(y_test, y_pred)}")

Perceptron Classifier accuracy: 0.434375

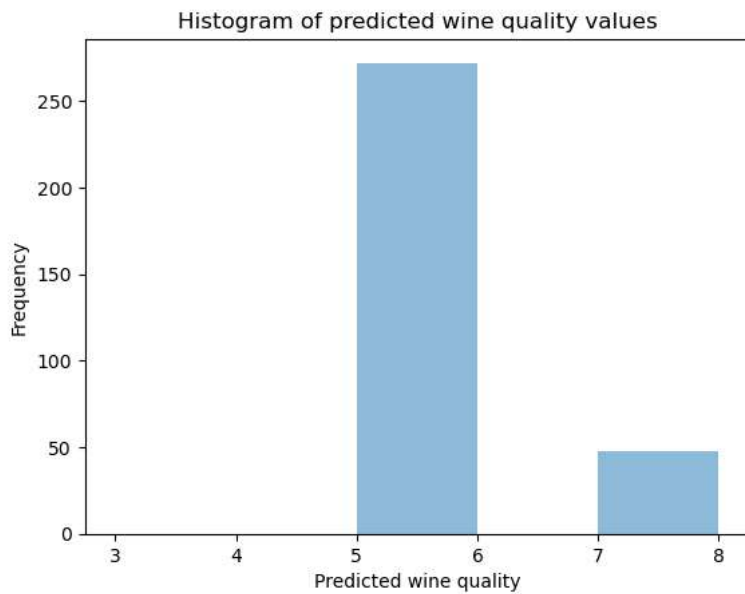
# Calculate confusion matrix
cm = confusion_matrix(y_test, y_pred)

# Plot confusion matrix
sns.heatmap(cm, annot=True, cmap="Blues", fmt="d")
plt.title("Confusion Matrix")
plt.xlabel("Predicted Labels")
plt.ylabel("True Labels")
plt.show()

```



```
# Plot histogram of predicted wine quality values  
plt.hist(y_pred, bins=range(3, 9), alpha=0.5)  
plt.xlabel("Predicted wine quality")  
plt.ylabel("Frequency")  
plt.title("Histogram of predicted wine quality values")  
plt.show()
```



[Colab paid products](#) - [Cancel contracts here](#)

● ×