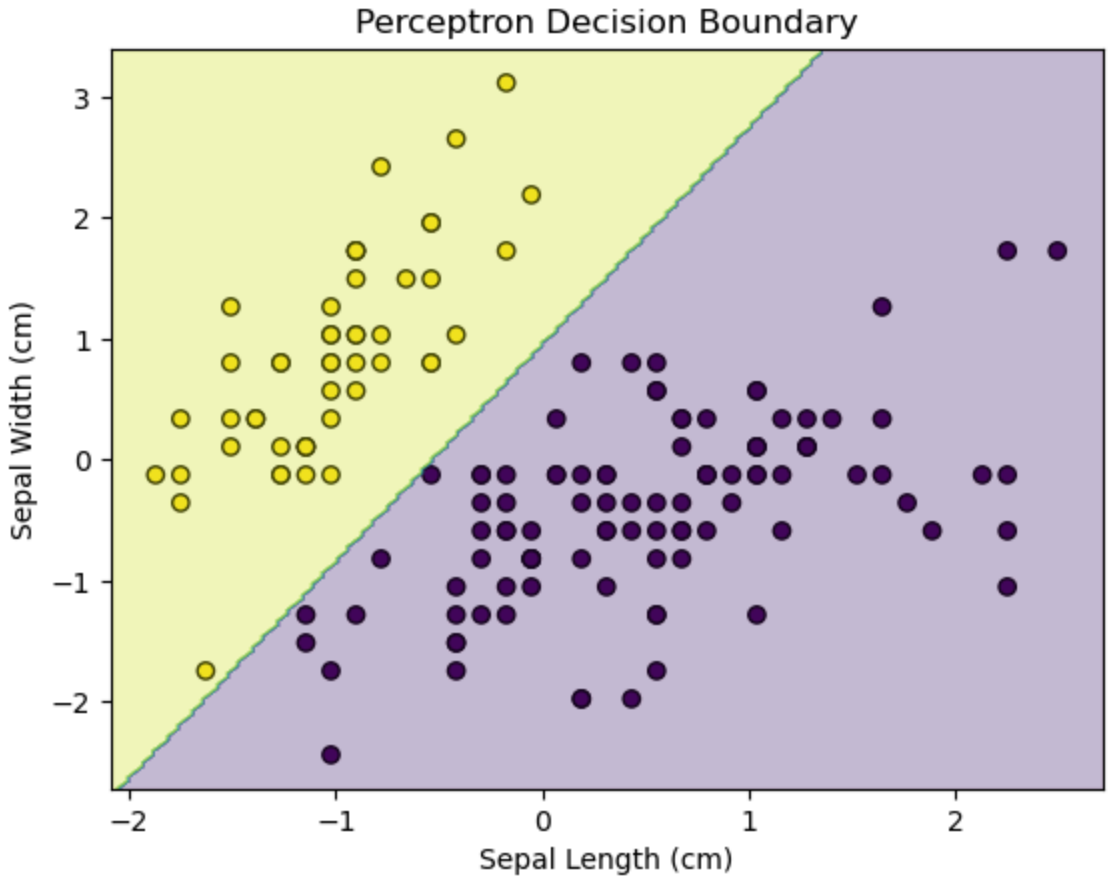
In [1]:
```
import pandas as pd
import numpy as np
from sklearn.linear_model import Perceptron
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt

# Load the dataset
df = pd.read_csv(r"C:\Users\tuhin\Downloads\archive (9)\Iris.csv")

# Define features and target
X = df[['SepalLengthCm', 'SepalWidthCm']]
y = np.where(df['Species'] == 'Iris-setosa', 1, -1)

# Standardize features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Initialize Perceptron model
perceptron_model = Perceptron()

# Train the model
perceptron_model.fit(X_scaled, y)

# Visualize decision boundary
plt.scatter(X_scaled[:, 0], X_scaled[:, 1], c=y, cmap='viridis', edgecolor='k'
xlim = plt.xlim()
ylim = plt.ylim()
xx, yy = np.meshgrid(np.linspace(*xlim, num=200), np.linspace(*ylim, num=200))
Z = perceptron_model.predict(np.c_[xx.ravel(), yy.ravel()])
Z = Z.reshape(xx.shape)
plt.contourf(xx, yy, Z, alpha=0.3, cmap='viridis')
plt.xlabel("Sepal Length (cm)")
plt.ylabel("Sepal Width (cm)")
plt.title("Perceptron Decision Boundary")
plt.show()

# Print the coefficients and intercept
print("Coefficients:", perceptron_model.coef_)
print("Intercept:", perceptron_model.intercept_)
```




```
Coefficients: [[-1.88617985  1.05056946]]
Intercept: [-1.]
```

In [ ]:

In [ ]:

In [ ]: