



Practical 6: Develop a Gradient descent of linear regression using sample dataset.

Code:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris

iris = load_iris()
X = iris.data[:, :1]
y = iris.data[:, 1:2]

m = 0
c = 0

n = float(len(X))
learning_rate = 0.01
epochs = 500

for i in range(epochs):
    y_pred = m * X + c
    D_m = (-2/n) * np.sum(X * (y - y_pred))
    D_c = (-2/n) * np.sum(y - y_pred)
    m = m - learning_rate * D_m
    c = c - learning_rate * D_c

    if i % 100 == 0:
        print(f"Epoch {i}: m = {m}, c = {c}")

print(f"Final m = {m}, c = {c}")
```



```
plt.scatter(X, y, color='red')  
plt.plot(X, m * X + c, color='blue')  
plt.xlabel('X as Petal Length')  
plt.ylabel('y as Petal Width')  
plt.title("Linear Regression with Gradient Descent")  
plt.show()
```



Output: Linear Regression with Gradient Descent

```
Epoch 0: m = 0.35645733333333324, c = 0.061146666666666676  
Epoch 100: m = 0.4769392403595542, c = 0.20936496800591392  
Epoch 200: m = 0.4568212595597973, c = 0.3292005763030532  
Epoch 300: m = 0.4374544203622572, c = 0.44456190302517357  
Epoch 400: m = 0.4188106775216284, c = 0.5556160036568425  
Final m = 0.4010391498282945, c = 0.6614746309563504
```

