

i love pdf - Yahoo India Search | Download file | iLovePDF | FundamentalsOfMachineLearn | Harishma0523/FOML

colab.research.google.com/drive/1SdqBF8cPKJS7sJqrzuJMs1HcENosz?authuser=1#scrollTo=oLp9svLobhTd

Commands + Code + Text Run all ⌂

Connect ⌂

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.tree import DecisionTreeRegressor
np.random.seed(42)
X = np.random.rand(100, 1) - 0.5
y = 3 * X[:, 0]**2 + 0.05 * np.random.randn(100)
plt.scatter(X, y, color='blue', s=25)
plt.title("X vs y (Quadratic Relationship)")
plt.xlabel("X")
plt.ylabel("y")
plt.show()
def gradient_boost(X, y, n_estimators=5, learning_rate=0.8):
    """
    Simple Gradient Boosting Regressor (from scratch)
    """
    y_pred = np.full_like(y, np.mean(y))
    models = []
    for i in range(n_estimators):
        residuals = y - y_pred
        tree = DecisionTreeRegressor(max_depth=3, random_state=42)
        tree.fit(X, residuals)
        models.append(tree)
        update = learning_rate * tree.predict(X)
        y_pred += update
        x_line = np.linspace(-0.5, 0.5, 500).reshape(-1, 1)
        y_line_pred = np.full_like(x_line[:, 0], np.mean(y))
        for m in models:
            y_line_pred += learning_rate * m.predict(x_line)
        plt.figure()
        plt.scatter(X, y, color='blue', s=25, label='True y')
        plt.plot(x_line, y_line_pred, color='red', linewidth=2, label=f'Boost {i+1}')
        plt.title(f'Iteration {i+1}: Gradient Boosting Progress')
    return y_pred
```

Variables Terminal

Breaking news
Hidma hid well....

Search

15:51
ENG IN 18-11-2025

i love pdf - Yahoo India Search | Download file | iLovePDF | FundamentalsOfMachineLearn | Harishma0523/FOML

colab.research.google.com/drive/1SdqBF8cPKJS7sJqrzuJM-Os1HcENosz?authuser=1#scrollTo=oLp9svLobhTd

Commands + Code + Text Run all ⌂

Connect ⌂

```
plt.figure()
plt.scatter(X, y, color='blue', s=25, label='True y')
plt.plot(x_line, y_line_pred, color='red', linewidth=2, label=f'Boost {i+1}')
plt.title(f'Iteration {i+1}: Gradient Boosting Progress')
plt.xlabel('x')
plt.ylabel('Predicted y')
plt.legend()
plt.show()
print("\n\U2611 Training Complete!")
return models, y_pred
models, final_pred = gradient_boost(X, y, n_estimators=5, learning_rate=0.8)
plt.figure(figsize=(8,5))
plt.scatter(X, y, color='blue', s=25, label='True y')
plt.scatter(X, final_pred, color='red', s=25, label='Predicted y (Final)')
plt.title("Final Gradient Boosting Prediction")
plt.xlabel("x")
plt.ylabel("y")
plt.legend()
plt.show()
```

X vs y (Quadratic Relationship)

Variables Terminal













