Home | Untitled | ML 1st practical | Untitled1 | ml 2nd practical | Untitled2 | Untitled3 | Untitled4 | Untitled5

localhost:8888/notebooks/Untitled5.ipynb?

Google Chrome isn't your default browser Set as default

jupyter Untitled5 Last Checkpoint: 31 seconds ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python [conda env:base] *

[1]:
```
# Import necessary libraries
from sklearn.datasets import load_iris
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# Load the Iris dataset
data = load_iris()
X = data.data  # Features
y = data.target  # Labels

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Initialize the Decision Tree Classifier
clf = DecisionTreeClassifier(criterion='gini', max_depth=3, random_state=42)

# Train the model
clf.fit(X_train, y_train)

# Make predictions on the test set
y_pred = clf.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy of the Decision Tree: {accuracy:.2f}")

# Optional: Visualize the Decision Tree
from sklearn.tree import export_text
print("\nDecision Tree Structure:")
print(export_text(clf, feature_names=data.feature_names))
```

```
Accuracy of the Decision Tree: 1.00

Decision Tree Structure:
|--- petal length (cm) <= 2.45
|   |--- class: 0
|--- petal length (cm) >  2.45
|   |--- petal length (cm) <= 4.75
|   |   |--- petal width (cm) <= 1.60
|   |   |   |--- class: 1
|   |   |--- petal width (cm) >  1.60
|   |   |   |--- class: 2
|   |--- petal length (cm) >  4.75
|   |   |--- petal width (cm) <= 1.75
|   |   |   |--- class: 1
|   |   |--- petal width (cm) >  1.75
|   |   |   |--- class: 2
```

[ ]:

Hot weather Now Search ENG IN 2:38 PM 5/6/2025

Home | Untitled | ML 1st practical | Untitled1 | ml 2nd practical | Untitled2

localhost:8888/notebooks/Untitled2.ipynb?

Google Chrome isn't your default browser Set as default

jupyter Untitled2 Last Checkpoint: 17 seconds ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python [conda env:base] *

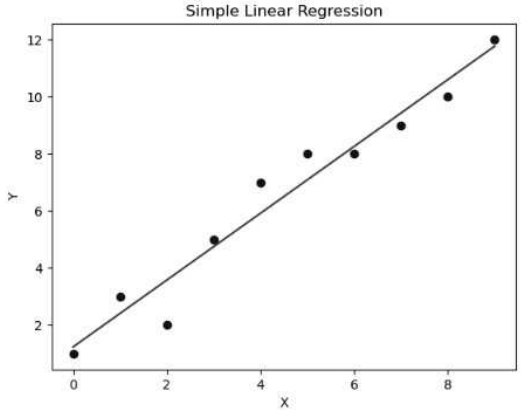
[1]:
```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

# Sample data
x = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9]).reshape(-1, 1)
y = np.array([1, 3, 2, 5, 7, 8, 8, 9, 10, 12])

# Create and train the model
model = LinearRegression()
model.fit(x, y)

# Predicting values
y_pred = model.predict(x)

# Plotting the results
plt.scatter(x, y, color='blue')
plt.plot(x, y_pred, color='red')
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Simple Linear Regression')
plt.show()
```



[ ]:

104°F Sunny Search ENG IN 2:33 PM 5/6/2025

Home | Untitled | ML 1st practical

localhost:8888/notebooks/ML%201st%20practical.ipynb?

Google Chrome isn't your default browser | Set as default

jupyter ML 1st practical Last Checkpoint: 1 minute ago

File Edit View Run Kernel Settings Help | Trusted

Code | JupyterLab | Python [conda env:base] *

```
# Example: Creating a DataFrame
data = {
    "Name": ["Alice", "Bob", "Charlie"],
    "Age": [25, 30, 35],
    "Score": [85, 90, 78]
}
df = pd.DataFrame(data)
print(df)

# Accessing a column
print(df["Name"])  # Output: Series of names
```

```
      Name  Age  Score
0    Alice   25     85
1      Bob   30     90
2  Charlie   35     78
0      Alice
1        Bob
2    Charlie
Name: Name, dtype: object
```

[5]:
```
# Example: List
fruits = ["apple", "banana", "cherry"]
print(fruits[1])  # Output: banana

# Adding an item
fruits.append("date")
print(fruits)  # Output: ['apple', 'banana', 'cherry', 'date']
```

```
banana
['apple', 'banana', 'cherry', 'date']
```

[7]:
```
import numpy as np

# Example: NumPy Array
arr = np.array([1, 2, 3, 4, 5])
print(arr[2])  # Output: 3

# Performing operations
print(arr * 2)  # Output: [2 4 6 8 10]
```

```
3
[ 2  4  6  8 10]
```

[ ]:

104°F Sunny | Search | ENG IN | 2:27 PM 5/6/2025

Home | Untitled | ML 1st practical | Untitled1 | ml 2nd practical

localhost:8888/notebooks/ml%202nd%20practical.ipynb?

Google Chrome isn't your default browser Set as default

jupyter ml 2nd practical Last Checkpoint: 16 seconds ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python [conda env:base] *

[3]:
```
# my_module.py

import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import matplotlib.pyplot as plt

def generate_data():
    """Generate synthetic data using NumPy."""
    np.random.seed(42)
    X = np.random.rand(100, 1) * 10  # 100 random numbers between 0 and 10
    y = 3 * X + np.random.randn(100, 1) * 2  # Linear relation with noise
    return X, y

def create_dataframe(X, y):
    """Create a Pandas DataFrame from the data."""
    data = pd.DataFrame({'Feature': X.flatten(), 'Target': y.flatten()})
    return data

def train_model(X, y):
    """Train a simple linear regression model using Scikit-learn."""
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    model = LinearRegression()
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    mse = mean_squared_error(y_test, y_pred)
    return model, mse, X_test, y_test, y_pred

def plot_results(X, y, model):
    """Visualize the data and the regression line using Matplotlib."""
    plt.scatter(X, y, color='blue', label='Data Points')
    plt.plot(X, model.predict(X), color='red', label='Regression Line')
    plt.xlabel('Feature')
    plt.ylabel('Target')
    plt.title('Linear Regression')
    plt.legend()
    plt.show()

if __name__ == "__main__":
    # Generate synthetic data
    X, y = generate_data()

    # Create a DataFrame
    df = create_dataframe(X, y)
```


104°F Sunny Search ENG IN 2:30 PM 5/6/2025

Home | Untitled | ML 1st practical

localhost:8888/notebooks/ML%201st%20practical.ipynb?

Google Chrome isn't your default browser | Set as default

jupyter ML 1st practical Last Checkpoint: 1 minute ago

File Edit View Run Kernel Settings Help | Trusted

Code | JupyterLab | Python [conda env:base] *

[1]:
```
# Example: Dictionary
student_scores = {"Alice": 85, "Bob": 90, "Charlie": 78}
print(student_scores["Bob"])  # Output: 90

# Adding a new key-value pair
student_scores["David"] = 88
print(student_scores)  # Output: {'Alice': 85, 'Bob': 90, 'Charlie': 78, 'David': 88}
```

```
90
{'Alice': 85, 'Bob': 90, 'Charlie': 78, 'David': 88}
```

[3]:
```
import pandas as pd

# Example: Creating a DataFrame
data = {
    "Name": ["Alice", "Bob", "Charlie"],
    "Age": [25, 30, 35],
    "Score": [85, 90, 78]
}
df = pd.DataFrame(data)
print(df)

# Accessing a column
print(df["Name"])  # Output: Series of names
```

```
      Name  Age  Score
0    Alice   25     85
1      Bob   30     90
2  Charlie   35     78
0      Alice
1        Bob
2    Charlie
Name: Name, dtype: object
```

[5]:
```
# Example: List
fruits = ["apple", "banana", "cherry"]
print(fruits[1])  # Output: banana

# Adding an item
fruits.append("date")
print(fruits)  # Output: ['apple', 'banana', 'cherry', 'date']
```

```
banana
['apple', 'banana', 'cherry', 'date']
```

[7]:
```
import numpy as np
```


104°F Sunny | Search | ENG IN | 2:27 PM 5/6/2025

Home | Untitled

localhost:8888/notebooks/Untitled.ipynb?

Google Chrome isn't your default browser | Set as default

jupyter Untitled Last Checkpoint: 4 minutes ago

File Edit View Run Kernel Settings Help

Trusted

Code | JupyterLab | Python [conda env:base] *

[1]:
```
# Example: Dictionary
student_scores = {"Alice": 85, "Bob": 90, "Charlie": 78}
print(student_scores["Bob"])  # Output: 90

# Adding a new key-value pair
student_scores["David"] = 88
print(student_scores)  # Output: {'Alice': 85, 'Bob': 90, 'Charlie': 78, 'David': 88}
```

```
90
{'Alice': 85, 'Bob': 90, 'Charlie': 78, 'David': 88}
```

[3]:
```
import pandas as pd

# Example: Creating a DataFrame
data = {
    "Name": ["Alice", "Bob", "Charlie"],
    "Age": [25, 30, 35],
    "Score": [85, 90, 78]
}
df = pd.DataFrame(data)
print(df)

# Accessing a column
print(df["Name"])  # Output: Series of names
```

```
      Name  Age  Score
0    Alice   25     85
1      Bob   30     90
2  Charlie   35     78
0      Alice
1        Bob
2    Charlie
Name: Name, dtype: object
```

[5]:
```
# Example: List
fruits = ["apple", "banana", "cherry"]
print(fruits[1])  # Output: banana

# Adding an item
fruits.append("date")
print(fruits)  # Output: ['apple', 'banana', 'cherry', 'date']
```

```
banana
['apple', 'banana', 'cherry', 'date']
```

[7]:
```
import numpy as np

# Example: NumPy Array
arr = np.array([1, 2, 3, 4, 5])
print(arr[2])  # Output: 3

# Performing operations
print(arr * 2)  # Output: [2 4 6 8 10]
```

```
3
[ 2  4  6  8 10]
```

Home | Untitled | ML 1st practical | Untitled1 | ml 2nd practical

localhost:8888/notebooks/ml%202nd%20practical.ipynb?

Google Chrome isn't your default browser Set as default

jupyter ml 2nd practical Last Checkpoint: 47 seconds ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python [conda env:base] *

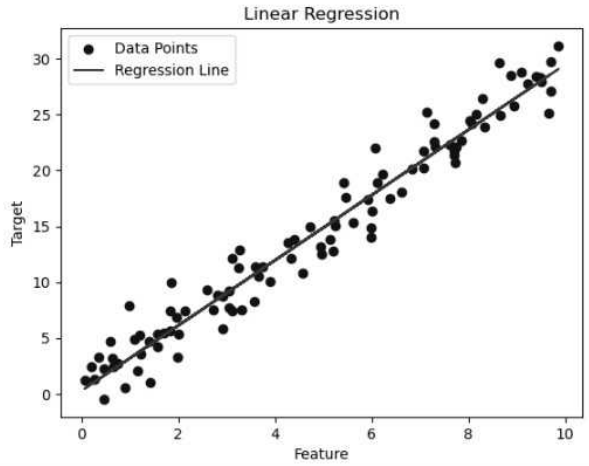
```
if __name__ == "__main__":
    # Generate synthetic data
    X, y = generate_data()

    # Create a DataFrame
    df = create_dataframe(X, y)
    print("Sample DataFrame:")
    print(df.head())

    # Train the model
    model, mse, X_test, y_test, y_pred = train_model(X, y)
    print(f"Mean Squared Error: {mse:.2f}")

    # Plot the results
    plot_results(X, y, model)
```

```
Sample DataFrame:
    Feature     Target
0  3.745401  11.410298
1  9.507143  27.923414
2  7.319939  22.143340
3  5.986585  13.984617
4  1.560186   4.241215
Mean Squared Error: 2.61
```



104°F Sunny Search ENG IN 2:31 PM 5/6/2025