

JupyterLite (2) WhatsApp

jupyter.org/try-jupyter/lab/

File Edit View Run Kernel Tabs Settings Help

notebooks /

Name Modified

- cpp-smallpt.ipynb 4mo ago
- cpp-third-party-li... 4mo ago
- cpp.ipynb 4mo ago
- Intro.ipynb 43s ago
- Lorenz.ipynb 4mo ago
- r.ipynb 4mo ago
- sqlite.ipynb 4mo ago

Intro.ipynb

[13]: # Logistic Regression (Jupyter Notebook - error-free)

```
import numpy as np
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# For reproducible random data
np.random.seed(0)

# Create simple synthetic dataset
X = np.random.rand(200, 2)
y = (X[:, 0] + X[:, 1] > 1).astype(int) # binary target values

# Split into train/test sets
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=42
)

# Create and train model
model = LogisticRegression()
model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)

# Accuracy
accuracy = accuracy_score(y_test, y_pred)
print("Logistic Regression Accuracy:", accuracy)
```

Logistic Regression Accuracy: 0.95

[14]: # Multiclass Logistic Regression using the Iris dataset

Simple Python (Pyodide) | Idle

Mode: Command

LN 30, Col 1 Intro.ipynb 2

3 2 cm of rain Tuesday

Search

10:15 15-11-2025

JupyterLite (2) WhatsApp

jupyter.org/try-jupyter/lab/

File Edit View Run Kernel Tabs Settings Help

Intro.ipynb Launcher Launcher

Notebook Python (Pyodide)

/ notebooks /

Name	Modified
cpp-smallpt.ipynb	4mo ago
cpp-third-party-li...	4mo ago
cpp.ipynb	4mo ago
Intro.ipynb	43s ago
Lorenz.ipynb	4mo ago
r.ipynb	4mo ago
sqlite.ipynb	4mo ago

```
from sklearn.datasets import load_iris
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# Load data
iris = load_iris()
X = iris.data      # 4 features
y = iris.target    # 3 classes

# Split dataset
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=1
)

# Create model (multinomial)
model = LogisticRegression(max_iter=200, multi_class='multinomial')

# Train
model.fit(X_train, y_train)

# Predict
y_pred = model.predict(X_test)

# Accuracy
acc = accuracy_score(y_test, y_pred)
print("Multiclass Logistic Regression Accuracy:", acc)

/lib/python3.12/site-packages/sklearn/linear_model/_logistic.py:1247: FutureWarning: 'multi_class' was deprecated in version 1.5 and will be removed in 1.7. From then on, it will always use 'multinomial'. Leave it to its default value to avoid this warning.
  warnings.warn(
Multiclass Logistic Regression Accuracy: 0.9777777777777777
```

Simple Python (Pyodide) | Idle Mode: Command Ln 30, Col 1 Intro.ipynb 2 10:15
2 cm of rain Tuesday ENG IN 15-11-2025