

```

from google.colab import drive
drive.mount('/content/drive')

import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report

# Load the data
train_data = pd.read_csv('/content/drive/MyDrive/AI Dataset/mnist_train.csv')
test_data = pd.read_csv('/content/drive/MyDrive/AI Dataset/mnist_test.csv')

# Split data into features and labels
X_train = train_data.drop('label', axis=1)
y_train = train_data['label']
X_test = test_data.drop('label', axis=1)
y_test = test_data['label']

# Create and train a logistic regression model
model = LogisticRegression(max_iter=1000)
model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)

print(f'Accuracy: {accuracy}')
print(report)

```

Mounted at /content/drive
 /usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logistic.py:458: ConvergenceWarning: lbfgs failed to converge (status=1):
 STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

n_iter_i = _check_optimize_result(

Accuracy: 0.9203

	precision	recall	f1-score	support
0	0.95	0.97	0.96	980
1	0.95	0.98	0.97	1135
2	0.92	0.89	0.90	1032
3	0.90	0.91	0.90	1010
4	0.93	0.93	0.93	982
5	0.90	0.85	0.87	892
6	0.94	0.94	0.94	958
7	0.93	0.92	0.92	1028
8	0.87	0.88	0.87	974
9	0.91	0.91	0.91	1009
accuracy			0.92	10000
macro avg	0.92	0.92	0.92	10000

weighted avg	0.92	0.92	0.92	10000
--------------	------	------	------	-------