

```

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

# Sample credit score dataset
data = {
    "Age": [25, 35, 45, 20, 52, 23, 40, 60],
    "Income": [30000, 60000, 80000, 20000, 90000, 25000, 70000,
100000],
    "LoanAmount": [5000, 10000, 20000, 3000, 25000, 4000, 15000,
30000],
    "CreditScore": ["Bad", "Good", "Good", "Bad", "Good", "Bad",
"Good", "Good"]
}

df = pd.DataFrame(data)
df

{"summary": "{\n  \"name\": \"df\",\n  \"rows\": 8,\n  \"fields\": [\n    {\n      \"column\": \"Age\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 14,\n        \"min\": 20,\n        \"max\": 60,\n        \"num_unique_values\": 8,\n        \"samples\": [\n          35,\n          23,\n          25\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"Income\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 30988,\n        \"min\": 20000,\n        \"max\": 100000,\n        \"num_unique_values\": 8,\n        \"samples\": [\n          60000,\n          25000,\n          30000\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"LoanAmount\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 10226,\n        \"min\": 3000,\n        \"max\": 30000,\n        \"num_unique_values\": 8,\n        \"samples\": [\n          10000,\n          4000,\n          5000\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"CreditScore\",\n      \"properties\": {\n        \"dtype\": \"category\",\n        \"num_unique_values\": 2,\n        \"samples\": [\n          \"Good\",\n          \"Bad\"\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    }\n  ],\n  \"type\": \"dataframe\",\n  \"variable_name\": \"df\"\n}"}

# Convert target labels to numerical values
df["CreditScore"] = df["CreditScore"].map({"Bad": 0, "Good": 1})

X = df[["Age", "Income", "LoanAmount"]]
y = df["CreditScore"]

```

```

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.25, random_state=42
)

model = LogisticRegression()
model.fit(X_train, y_train)

LogisticRegression()

from sklearn.metrics import accuracy_score, classification_report

y_pred = model.predict(X_test)

print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n",
      classification_report(y_test, y_pred, zero_division=0))

```

Accuracy: 0.5

Classification Report:

	precision	recall	f1-score	support
0	0.50	1.00	0.67	1
1	0.00	0.00	0.00	1
accuracy			0.50	2
macro avg	0.25	0.50	0.33	2
weighted avg	0.25	0.50	0.33	2

```

# New customer details (use DataFrame with same feature names)
new_customer = pd.DataFrame(
    [[30, 50000, 8000]],
    columns=["Age", "Income", "LoanAmount"]
)

```

```
prediction = model.predict(new_customer)
```

```

if prediction[0] == 1:
    print("Credit Score: Good")
else:
    print("Credit Score: Bad")

```

Credit Score: Bad