

Sample Data Usage Guide

📁 Available Sample Files

1. **sample_single_request.json**

Single patient data for testing `/predict` endpoint

```
curl -X POST http://localhost:8000/predict \
-H "Content-Type: application/json" \
-d @sample_single_request.json
```

2. **sample_batch_request.json**

Batch of 5 patients for testing `/predict/batch` endpoint

```
curl -X POST http://localhost:8000/predict/batch \
-H "Content-Type: application/json" \
-d @sample_batch_request.json
```

3. **sample_patients.json**

Comprehensive dataset of 20 diverse patient profiles with descriptions

🧪 Testing Methods

Method 1: Using the Test Script (Recommended)

```
source venv/bin/activate
python test_samples.py
```

Interactive menu with options:

1. Test Single Prediction
2. Test Batch Prediction
3. Test All Sample Patients (20 patients)
4. Test Health Endpoint
5. Run All Tests

Method 2: Using cURL

Single Prediction:

```
curl -X POST http://localhost:8000/predict \
-H "Content-Type: application/json" \
-d '{
    "age": 63,
    "ca": 0,
    "chol": 233,
    "cp": 3,
    "exang": 0,
    "fbs": 1,
    "oldpeak": 2.3,
    "restecg": 0,
    "sex": 1,
    "slope": 3,
    "thal": 6,
    "thalach": 150,
    "trestbps": 145
}'
```

Batch Prediction:

```
curl -X POST http://localhost:8000/predict/batch \
-H "Content-Type: application/json" \
-d @sample_batch_request.json
```

Method 3: Using Python

```
import requests
import json

# Single prediction
with open("sample_single_request.json") as f:
    data = json.load(f)

response = requests.post(
    "http://localhost:8000/predict",
    json=data
)
print(response.json())

# Batch prediction
with open("sample_batch_request.json") as f:
    data = json.load(f)

response = requests.post(
    "http://localhost:8000/predict/batch",
    json=data
)
print(response.json())
```

Patient Profiles in sample_patients.json

Risk Level	Count	Examples
High Risk	6	Elderly with multiple symptoms, severe chest pain, post-MI
Medium Risk	7	Moderate symptoms, diabetic, hypertensive, borderline
Low Risk	7	Young and healthy, active lifestyle, good vitals

High Risk Patients (Likely Disease = 1)

- Age: 63-74 years
- Multiple risk factors (high cholesterol, chest pain, exercise-induced angina)
- Features: ca=2-3, cp=3-4, thal=7, low thalach

Medium Risk Patients (Varies)

- Age: 50-60 years
- Some risk factors present
- Features: ca=0-1, cp=2, moderate cholesterol

Low Risk Patients (Likely Disease = 0)

- Age: 29-48 years
- Minimal risk factors
- Features: ca=0, cp=1, thal=3, high thalach

Feature Descriptions

Feature	Description	Values
age	Age in years	29-77
sex	Sex (1=male, 0=female)	0, 1
cp	Chest pain type	1-4 (1=typical angina, 4=asymptomatic)
trestbps	Resting blood pressure	94-200 mm Hg
chol	Serum cholesterol	126-564 mg/dl
fbs	Fasting blood sugar > 120 mg/dl	0, 1
restecg	Resting ECG results	0-2
thalach	Maximum heart rate	71-202
exang	Exercise induced angina	0, 1
oldpeak	ST depression	0-6.2
slope	Slope of peak exercise ST	1-3

Feature	Description	Values
ca	Number of major vessels	0-3
thal	Thalassemia	3, 6, 7

Expected Results

Running all 20 sample patients should give approximately:

- **No Disease (0):** ~50-60%
- **Disease (1):** ~40-50%

Risk distribution:

- **Low Risk:** ~35%
- **Medium Risk:** ~35%
- **High Risk:** ~30%

Use Cases

1. **Dashboard Testing:** Run `test_samples.py` option 3 to populate Grafana metrics
2. **Load Testing:** Use batch endpoint with all 20 patients
3. **API Documentation:** Examples for Swagger/OpenAPI docs
4. **Unit Testing:** Base test cases for pytest
5. **Demo/Presentation:** Show diverse patient scenarios

Notes

- All data is synthetic but follows UCI Heart Disease dataset patterns
- Values are medically plausible but not from real patients
- Designed to test different model prediction scenarios
- Ensures comprehensive coverage of feature space