Step 3: Building a REST API with FastAPI

University: Tek-Up
Group: ING-4-SDIA

Tutor: Haythem Ghazouani

1 Introduction

Once a model is trained, it needs to be "served" so other applications (like a web frontend or a mobile app) can use it. **FastAPI** is a modern, high-performance web framework for building APIs with Python 3.7+ based on standard Python type hints.

Implementation Note

The source code for our API is located in `code/app.py`.

2 Key Features of FastAPI

1. **Type Hints:** Automatic data validation and documentation.
2. **Asynchronous:** Support for `async` and `await`.
3. **Automatic Docs:** Generates interactive Swagger UI and Redoc documentation on the fly.

3 Designing our Churn API

API Concept: 1. Data Validation with Pydantic

We use **Pydantic** models to define exactly what data the API expects. If a user sends a string instead of an integer for "Age", FastAPI will automatically reject the request with a clear error message.

API Concept: 2. Real-time Prediction (/predict)

A `POST` endpoint that accepts a single customer's data and returns a churn prediction (0 or 1) along with the probability.

API Concept: 3. Batch Prediction (/predict_batch)

A more advanced endpoint that accepts a **CSV file** upload, processes thousands of rows, and returns the predictions for the entire list.

API Concept: 4. Health Check (/health)

A standard industry practice to ensure the service is running and the model pipeline is correctly loaded into memory.

4 Running the API

To start the server, use **Uvicorn**:

```
uvicorn code.app:app --reload
```

Once running, visit:

- `http://localhost:8000/docs` to interact with the API via Swagger UI.
- `http://localhost:8000/health` to check the status.

5 Exercise

1. Add a custom header to the API response that includes the model version.
2. Implement a `GET` endpoint that returns basic statistics about the dataset currently loaded in memory (if any).
3. **Hardware Challenge:** How would you modify the API to use a GPU if your model was a deep learning model?