Week 11 Assignment
Machine Learning Application
Part 2

Applied AI and Machine Learning

Boston College
Fall 2025

Objective

This assignment walks you through building, saving, and deploying a machine learning web app that forecasts next month's car sales using the previous six months of data. You'll use Python, FastAPI, a simple HTML form, and the Render hosting platform. All technical steps will be explained assuming no prior web development experience.

Overview of What You'll Build

- A trained regression model that predicts car sales
- A FastAPI server to handle prediction requests
- A static HTML form to enter the last 6 months of sales
- Deployment of your model online (free) using Render
- Logging your model as a .json file for compatibility

Steps and Instructions

1. Train Your Model

1. Use the car sales dataset provided
2. Create lag features (sales from last 6 months)
3. Use *XGBRegressor* to train the model

2. Save the Model in JSON Format

1. Unlike pickle, JSON is safer across environments
2. Command to save *model.get_booster().save_model("model/model.json")*

3. Train Your Model

1. Install FastAPI and Uvicorn: *pip install fastapi uvicorn xgboost*
2. Complete the sample *app.py*

4. Save the Model in JSON Format

1. Save this as *frontend/index.html*
2. It collects 6 months of sales and sends them to */predict*

5. Deploy with Render

1. Push your code to GitHub

2. Go to *https://render.com*, create an account
3. Create a new Web Service → Connect to GitHub repo
4. In **Build Command**, use: *pip install -r requirements.txt*
5. In **Start Command**, use: *uvicorn app:app --host 0.0.0.0 --port $PORT*

6. Test Your Live App

1. Visit the Render URL (e.g., *https://your-app.onrender.com*)
2. Enter sales data and click Predict
3. You should see the prediction instantly

Objective

May take a minute to spin up: https://car-sales-forecast.onrender.com

Overview of What You'll Build

- GitHub link to your repo (with model.json, app.py, and HTML)
- Render live URL
- Screenshots showing successful prediction

Extra Credit (Optional)

- Log predictions to a database (e.g., SQLite or Supabase)
- Add input validation to the frontend
- Style your HTML with colors/fonts