Gurugram, India · +91 93114 71700
gillharshit710@gmail.com
harshit-gill.github.io/portfolio
linkedin.com/in/harshit-gill · github.com/Harshit-Gill

Harshit Gill

B.Tech, Artificial Intelligence & Machine Learning

EDUCATION

B.Tech, Artificial Intelligence & Machine Learning — The NorthCap University, Gurugram 2025 – 2029 (expected)
CGPA: 7.14 / 10 · Coursework: Machine Learning, Data Structures & Algorithms, Object-Oriented Programming (Java), Programming in C

Class XII — MR Bharti Model Sr Sec School, Mundka, Delhi · 75.6% 2025

Class X — CCA School, Sector 4, Gurugram · 90.8% 2023

TECHNICAL SKILLS

Languages	Python, Java, C, C++
Machine Learning	scikit-learn — linear and logistic regression, KNN, random forest, gradient boosting, Isolation Forest (anomaly detection)
ML Practice	Feature engineering, categorical encoding (one-hot, target), scaling, cross-validation, hyperparameter tuning, model evaluation (RMSE, R^2, precision, recall, F1), avoiding data leakage, scikit-learn pipelines
Deep Learning	PyTorch, torchvision, CNNs, data augmentation, Grad-CAM
Data & Visualization	NumPy, pandas, Matplotlib, Seaborn, Plotly
Deployment & Tools	FastAPI, REST APIs, Streamlit, Render, GitHub Pages, HTML/CSS/JavaScript (basics), Git, GitHub, Jupyter

PROJECTS

MPLADS Fraud & Anomaly Detection (Smart India Hackathon, SIH26102) — scikit-learn, FastAPI, Render, GitHub Pages **GitHub · Live**

- Built an unsupervised detector for irregular MP Local Area Development Scheme spending, combining 4 audit rules with an Isolation Forest over 11 engineered vendor and spending features. No labels were used to build it.
- Found that the round-number-invoice rule was flagging honest bills more often than fraudulent ones (7% precision vs. an 11.6% base rate). Demoting it to a risk-score signal raised hold-out precision from **20% to 47%** and cut flagged works from 46% to 13%.
- Deployed a FastAPI scoring API on Render and an auditor dashboard that scores 8,658 works with explainable 0–100 risk scores.

Bengaluru House Price Predictor — scikit-learn, FastAPI, Render, GitHub Pages **GitHub · Live**

- Built a gradient boosting model that predicts house prices across 28 Bengaluru localities, reaching **R^2 0.81** and cutting test RMSE by 56% against a mean-price baseline (₹63.3L to ₹27.7L).
- Found that locality was encoded by listing frequency rather than price level, which ranked cheaper areas above premium ones. Fixing it raised R^2 from 0.50 to 0.81. Chose the model by 5-fold cross-validation, comparing linear regression, KNN and gradient boosting.
- Served the model as a single scikit-learn pipeline through a FastAPI REST API on Render with input validation, behind a GitHub Pages frontend that shows live estimates and a model-evaluation dashboard.

Tomato Leaf Disease Classifier — PyTorch, Grad-CAM, Streamlit **GitHub · Live**

- Built a 95K-parameter CNN from scratch that reaches **96.0% test accuracy** (macro F1 0.955) across 10 tomato disease classes on 16K PlantVillage images. The 390 KB model runs in real time on a CPU.
- Used Grad-CAM to inspect errors and found that diseased leaves predicted as healthy came from attention drifting to the background. Documented the gap between lab and field images.
- Deployed a Streamlit web app that shows top-3 predictions with confidence scores and a Grad-CAM heatmap for each uploaded photo.

ACHIEVEMENTS

- Smart India Hackathon: cleared the internal college round with an MPLADS fraud detection solution (problem statement SIH26102, MoSPI).