HiLabs Hackathon 2025: Patient Risk Identification

Problem Statement: Predict Patient Risk levels from multi-source healthcare data to enable targeted, proactive care management.

Introduction

HiLabs invites you to participate in an exciting hackathon centered on one of the most critical problems in value-based care operations – building a patient risk identification strategy from messy provider and member data

This challenge aims to simulate a real-world **Value-Based Care (VBC)** problem, where identifying at-risk patients enables care teams to prioritize interventions effectively. Participants will use structured and semi-structured data to build a **risk prediction score** by analyzing clinical, demographic, and utilization data to predict which patients are most likely to need immediate care intervention.

At HiLabs, we leverage AI to tackle real-world problems in the healthcare industry to drive real-world impact with our solutions. This challenge tests your ability to build intelligent, scalable solutions using data science to make a measurable impact on care quality.

Domain and Context

This use case is set in the U.S. healthcare and value-based care (VBC) ecosystem, where providers and payers work together to improve patient outcomes while reducing unnecessary costs.

In this model, healthcare organizations are incentivized to identify and manage high-risk patients proactively

Your task simulates how care management platforms (like HiLabs' PCMS) use clinical, demographic, and utilization data to generate a risk score for every patient. This risk score helps care teams prioritize outreach and interventions, ensuring that limited resources focus on patients who need attention the most.

The Challenge

Your task is to build a model that can:

1. **Ingest multiple CSV datasets** containing patient demographics, diagnoses, care records, visit history, etc.
2. **Identify and engineer relevant features** (e.g., chronic conditions, visit frequency, readmission rates).
3. **Generate a patient-level risk score.**

The output must include each patient's ID and a corresponding **risk score** that represents the likelihood of adverse health outcomes or care gaps.

Sample Input/Output Data Provided

Participants will receive the following training and test data sets:

1. **Patient table:**
   This table includes data related to the patient demographics

patient_id	age	hot_spotter_identified _at	hot_spotter_readmission_flag	hot_spotter_chronic_flag
276	23	2023-04-09	f	f
234	14	2025-01-03	f	f

2. **Risk Table:**
   This table includes the risk score assigned to each patient.

patient_id	risk_score
276	**15.51**
234	**36.45**

3. Care Table

This table includes information related to the care – the details of the procedure performed, the results, date of the same, and the care gap identification indicator to mark missed or delayed procedures.

care_id	patient_id	msrmnt_type	msrmnt_sub_type	msrmnt_value	last_care_dt	next_care_dt	care_gap_ind
167	276	SCREENING	COLORECTAL CANCER	0.0	2025-09-08	NaN	t
189	234	LAB TEST	HbA1c	6.5	2024-07-03	NaN	t

4. Diagnosis table

This table contains data pertaining to the condition and diagnoses of the patient and their conditions, along with a flag for whether the condition is chronic or not.

diagnosis_id	patient_id	condition_name	condition_type	condition_description	Is_chronic
343	500975	CANCER	CHRONIC	Cancer recent medical history	t
248	5129	HYPERTENSION	CHRONIC	Hypertension past medical history	t

IIT KANPUR 2025

5. Visit table

This table displays the data related to the patient's visit, the duration, and the diagnosis identified at that visit, and the readmission indicator

visit_id	visit_type	patient_id	visit_start_dt	visit_end_dt	follow_up_dt	prmry_proc_nm	prncpl_diag_nm	readmsn_ind
7698	URGENT CARE	56	CHRONIC	2023-12-23	2024-12-31	NaN	Acute upper respiratory	f

							infection, unspecified	
248	URGENT CARE	56	CHRONIC	2024-02-18	8888-12-31	NaN	Acute pharyngitis, unspecified	f

Data Relationship Summary

Relationship	Type	Example
patient.csv ↔ risk.csv	1:1	One patient → one risk label
patient.csv ↔ diagnosis.csv	1:N	One patient → multiple diagnoses
patient.csv ↔ care.csv	1:N	One patient → multiple care events
patient.csv ↔ visit.csv	1:N	One patient → multiple visits

Data Relationship Summary

A **CSV file** named Prediction.csv containing two columns: (case-sensitive)

- patient_id
- predicted_risk_score

Key Tasks

1. **Data Cleaning**
    - Handle missing, inconsistent, or duplicate records.
    - Standardize field values such as diagnosis codes, care types, and visit dates.
    - Merge the datasets on patient_id to create a unified patient profile.

HiLabs™ AI Quest IIT KANPUR 2025

2. **Feature Identification & Engineering**
    - Identify impactful features such as:
        - Readmission rates
        - Chronic condition flags
        - Open care gaps
        - Diagnosis severity
    - Derive composite indicators (e.g., "care adherence index" or "visit frequency ratio").
3. **Model Development**
    - Train a predictive model using the training data (with provided risk scores).
    - Apply techniques such as regression, decision trees, ensemble methods, or deep learning models to predict patient risk.
    - Optimize for interpretability and accuracy.
4. **Output Generation**
    - Generate a Prediction.csv file for the test dataset with the schema specified above.
    - The output should be replicable and formatted cleanly.

HiLabs™ AI Quest IIT KANPUR 2025

Deliverables

Each submission must include:

- **A Jupyter Notebook (.ipynb) that:**
    - Accepts the 5 CSV files (patient.csv; risk.csv; visit.csv; care.csv; diagnosis.csv)
    - Produces a prediction file (Prediction.csv) with **patient_id** and **predicted_risk_score** (column names are case sensitive)
- **A README.md detailing:**
    - Overall approach and data architecture
    - Feature selection logic and assumptions
    - Model architecture and parameter tuning
    - Setup and execution steps
- **A public GitHub repository link with complete runnable code and instructions.**

Evaluation Criteria

Criteria	Weightage	Description
Model Accuracy	**40%**	**How accurately the model predicts risk scores on the test data**
Feature Engineering Quality	**25%**	**Creativity and relevance of features used**
Code Quality & Reproducibility	**20%**	**Readability, modularization, and documentation**
Interpretability & Explainability	**15%**	**Ability to explain how risk scores are derived**

Rules

- The code must be **self-contained and reproducible** using the provided instructions.
- **No use of external APIs or LLM services** - all logic should be implemented locally.
- Use of Python libraries such as Pandas, NumPy, Scikit-learn, Matplotlib, and XGBoost might be helpful
- You may use open-source resources for address normalization, string matching, or data standardization.
- The final output will be verified for **accuracy, consistency, and interpretability**.

Submission Format

All deliverables must be submitted via a public GitHub repository link and a CSV file containing the patient ids and their corresponding risk scores by the deadline.

- Naming convention for the risk_score file is : TeamName_HiLabs_Risk_Score.csv
- Ensure the repository includes:

- A bash script to create the Python environment.
- A /notebooks folder containing all Jupyter Notebook(s) used.
- A README.md file with clear setup and execution instructions.
- Any model files required for generating predictions.
- A requirements.txt file listing all dependencies needed to run the code end-to-end.

HiLabs™
AI Quest
IIT KANPUR
2025