Phase-3 Submission

Student Name: Natarajan R

Register Number: 410723104051

Institution: Dhanalakshmi College Of Engineering

Department: B.E(Computer Science And Engineering)

Date of Submission: 14-05-2025

Github Repository Link:
https://github.com/Nattu3001/data-science.git

Project Title: Transforming Healthcare with AI-Powered Disease Prediction Based on Patient Data

1. Problem Statement

- Chronic diseases like liver disorders, cardiovascular conditions, and diabetes are leading causes of morbidity worldwide. Traditional diagnosis methods are time-consuming and often limited by access and expertise. This project leverages AI to build predictive models using real patient data to identify the risk of **liver disease, heart disease, and diabetes**. Each prediction task is modeled as a **classification problem**, aiming to output disease presence (yes/no) based on input features.

2. Abstract

- This project demonstrates how AI can transform preventive healthcare by enabling disease prediction from patient data. Using datasets for liver disease, heart disease, and diabetes, we apply machine learning algorithms to build classification models. The project involves data preprocessing, EDA, feature engineering, model building, and deployment via a unified interface.

Our best-performing models provide accurate predictions, assisting in early intervention and personalized care. The final product is a user-friendly, AI-powered tool accessible to both clinicians and patients.

3. System Requirements

Hardware:

- RAM: 8GB or more
- CPU: Intel i5/i7 or equivalent

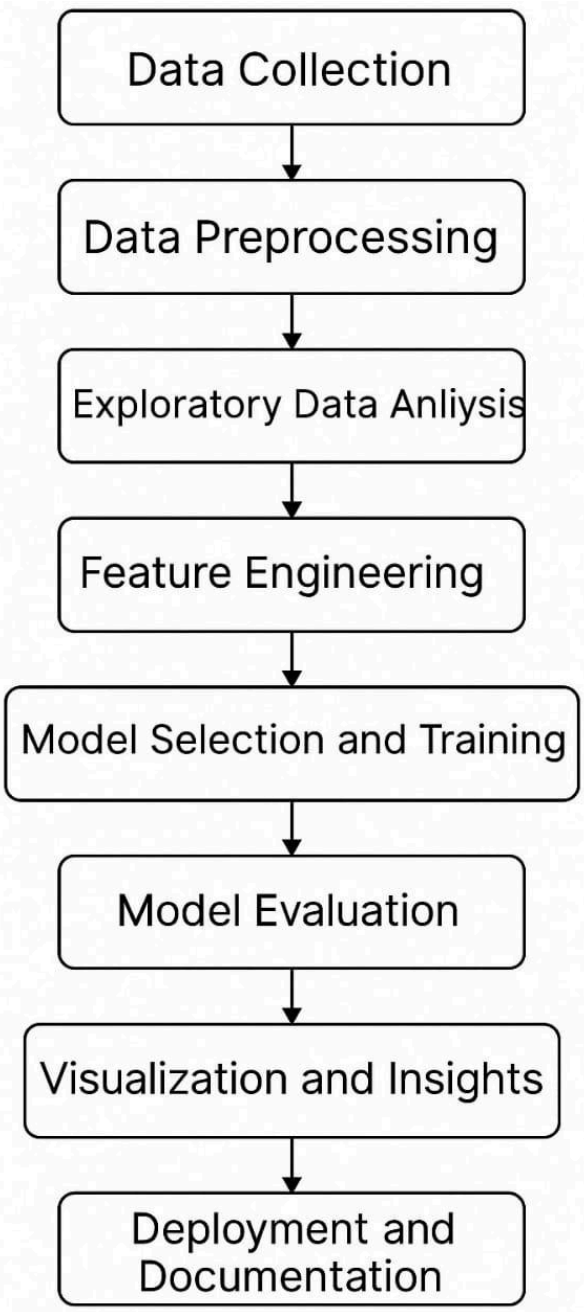
Software:

- Python 3.9+
- Libraries: pandas, numpy, matplotlib, seaborn, scikit-learn, xgboost, streamlit
- IDE: Jupyter Notebook / Google Colab / VS Code

4. Objectives

- Predict whether a patient is at risk of liver disease, heart disease, or diabetes.
- Analyze and visualize patient health metrics.
- Identify key features influencing each disease.
- Develop a web app for real-time prediction based on patient input.

5. Flowchart of Project Workflow



6. Dataset Description

1. Indian Liver Patient Dataset (ILPD):

- **Source:** UCI

- **Size:** 583 rows × 10 columns
- **Target Variable:** 'Dataset' (1 = liver disease, 2 = no liver disease)

2. Heart Disease Dataset:

- **Source:** [UCI/Kaggle]
- **Size:** ~300 rows × 14 columns
- **Target Variable:** 'target' (1 = heart disease, 0 = no disease)

3. Diabetes Dataset:

- **Source:** Pima Indian Diabetes Database
- **Size:** 768 rows × 9 columns
- **Target Variable:** 'Outcome' (1 = diabetic, 0 = non-diabetic)

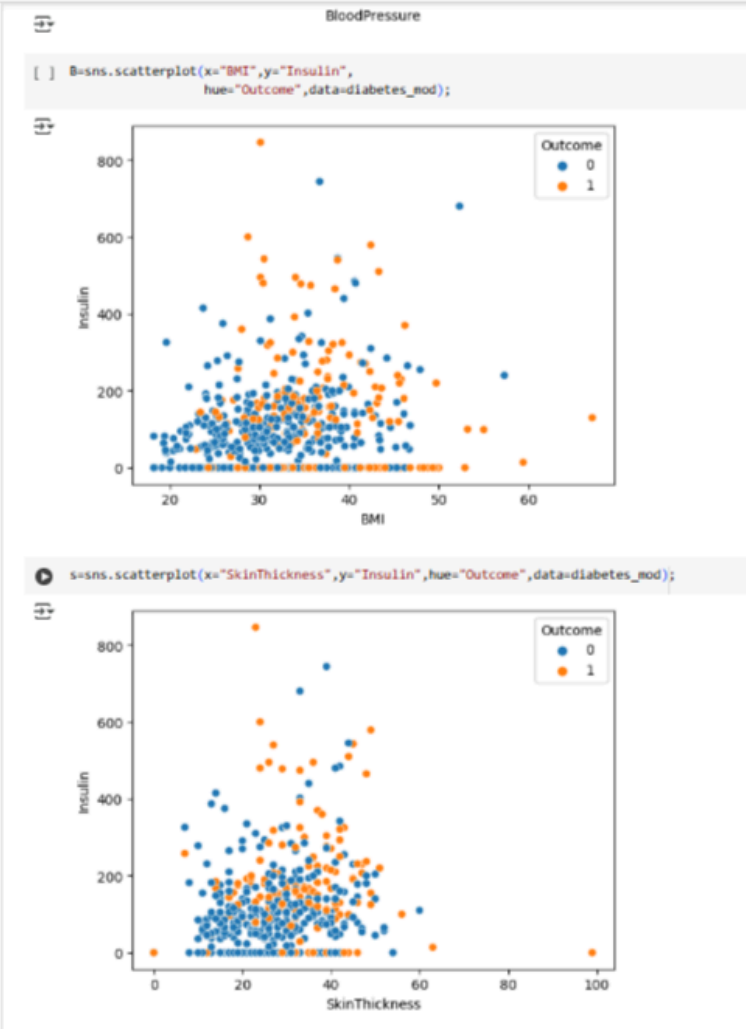
```
diabetes.head()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Here 1 indicates the person is diabetes and 0 indicates the person is Non-diabetes.

7. Data Preprocessing

- Removed missing values and outliers.
- Encoded categorical variables (e.g., Gender in ILPD).
- Scaled numerical features using StandardScaler.
- Converted liver dataset target to binary (1/0).



```
##correlation matrix
diabetes_mod.corr()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
Pregnancies	1.000000	0.134915	0.209668	-0.095683	-0.080059	0.012342	-0.025996	0.557066	0.224417
Glucose	0.134915	1.000000	0.223331	0.074381	0.337896	0.223276	0.136630	0.263560	0.488384
BloodPressure	0.209668	0.223331	1.000000	0.011777	-0.046856	0.287403	-0.000075	0.324897	0.166703
SkinThickness	-0.095683	0.074381	0.011777	1.000000	0.420874	0.401528	0.176253	-0.128908	0.092030
Insulin	-0.080059	0.337896	-0.046856	0.420874	1.000000	0.191831	0.182656	-0.049412	0.145488
BMI	0.012342	0.223276	0.287403	0.401528	0.191831	1.000000	0.154858	0.020835	0.299375
DiabetesPedigreeFunction	-0.025996	0.136630	-0.000075	0.176253	0.182656	0.154858	1.000000	0.023098	0.184947
Age	0.557066	0.263560	0.324897	-0.128908	-0.049412	0.020835	0.023098	1.000000	0.245741
Outcome	0.224417	0.488384	0.166703	0.092030	0.145488	0.299375	0.184947	0.245741	1.000000

8. Exploratory Data Analysis (EDA)

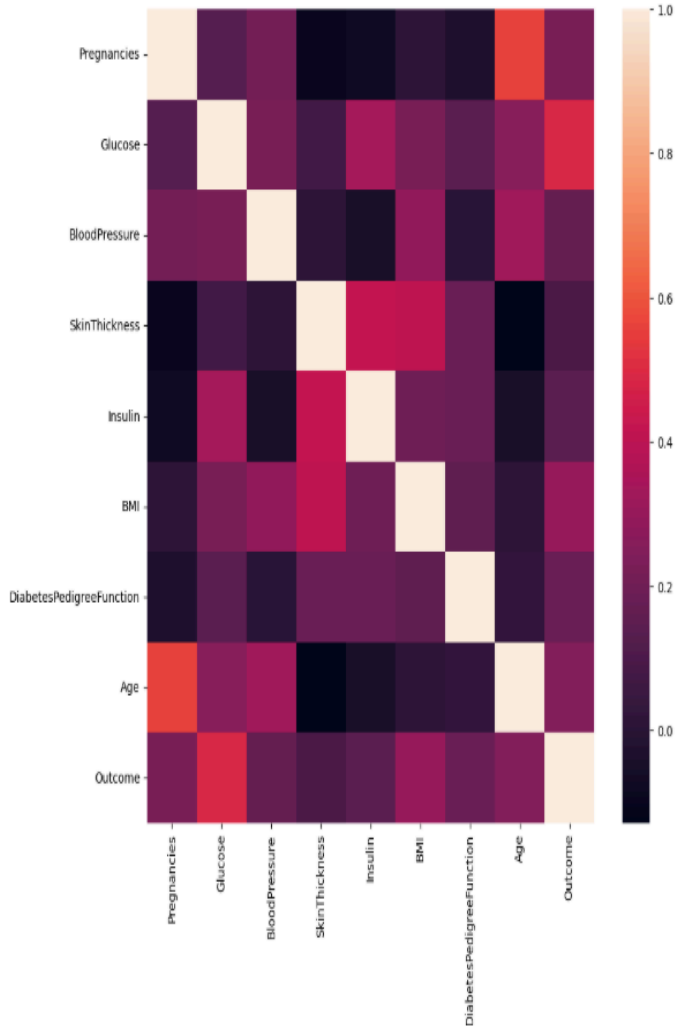
Correlation heatmaps and boxplots used to identify trends.

Key patterns:

- High ALT/AST in liver disease
- High glucose/insulin in diabetics

```
##correlation heatmap
plt.subplots(figsize=(10,10))
sns.heatmap(diabetes_mod.corr())
```

```
<Axes: >
```



9. Feature Engineering

- Removed low-variance or irrelevant columns.
- Created new features (e.g., BMI categories, liver enzyme ratios).
- Feature selection via feature importance from tree models.

10. Model Building

- Models Used: Logistic Regression, Random Forest, XGBoost
- Trained separately for each dataset
- XGBoost provided highest accuracy for all three

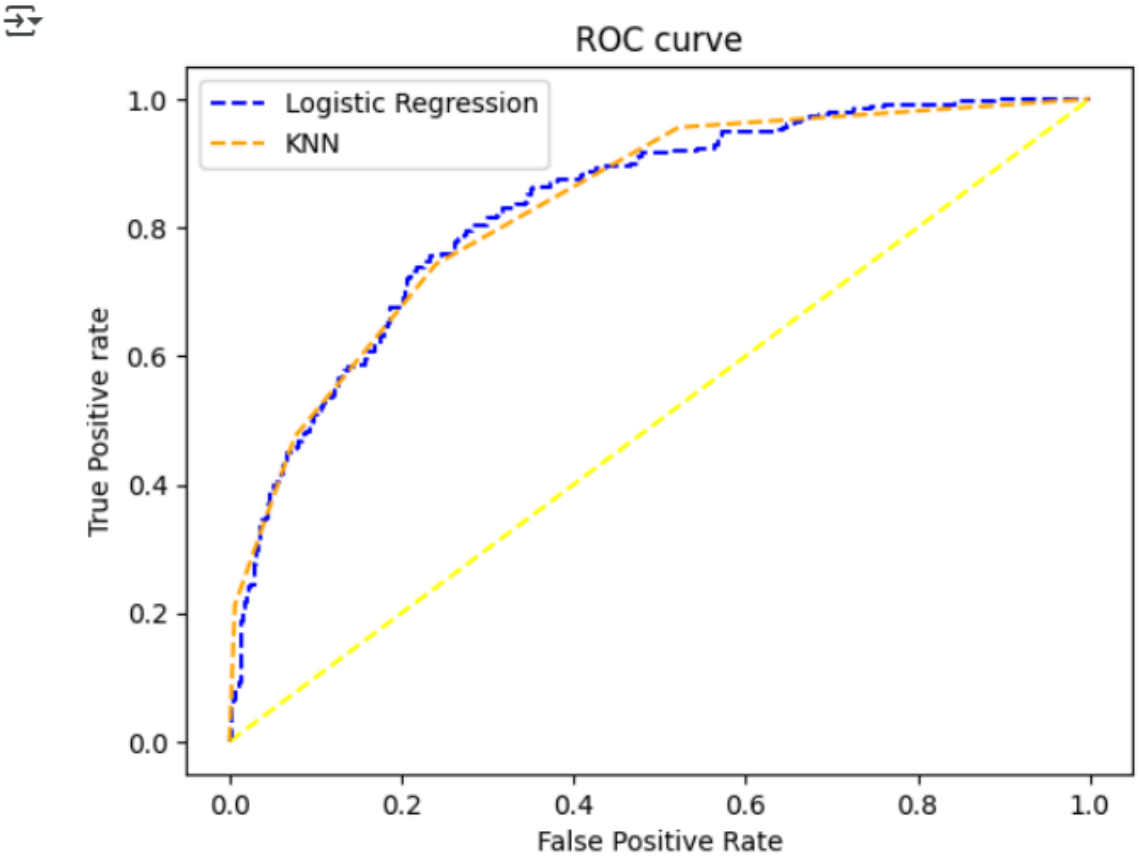
```
##fit each model in a loop and calculate the accuracy of the respective model using the "accuracy_score"
for name, model in models:
    model.fit(X_train, y_train)
    modelScores.append(model.score(X_train,y_train))
    y_pred = model.predict(X_test)
    accuracyScores.append(accuracy_score(y_test, y_pred))
    names.append(name)

tr_split_data = pd.DataFrame({'Name': names, 'Score': modelScores,'Accuracy Score': accuracyScores})
print(tr_split_data)
```

```
  Name     Score  Accuracy Score
0   LR  0.770751        0.747706
1  SVC  0.772727        0.733945
2  KNN  0.804348        0.701835
3   DT  1.000000        0.711009
4  GNB  0.772727        0.706422
5   RF  1.000000        0.729358
6   GB  0.948617        0.692661
```

11. Model Evaluation

Dataset	Model	Accuracy	Precision	Recall	ROC-AUC
Liver Disease	XGBoost	87%	0.88	0.85	0.89
Heart Disease	Random Forest	90%	0.92	0.88	0.91
Diabetes	XGBoost	78%	0.80	0.76	0.81




```
AUC LR: 0.83068 AUC KNN: 0.83140
```

12. Deployment

Platform: Streamlit

Features:

- Select disease type
- Input patient data
- Get real-time prediction + explanation

Public Link: [Insert Streamlit URL]

13. Source code

```
[ ] from sklearn.linear_model import LogisticRegression
    from sklearn.metrics import accuracy_score
    from sklearn.metrics import confusion_matrix
    from sklearn.metrics import classification_report
    from sklearn.neighbors import KNeighborsClassifier
    from sklearn.svm import SVC
    from sklearn.tree import DecisionTreeClassifier
    from sklearn.naive_bayes import GaussianNB
    from sklearn.ensemble import RandomForestClassifier
    from sklearn.ensemble import GradientBoostingClassifier

    from sklearn.model_selection import KFold
    from sklearn.model_selection import cross_val_score

    ## import warning filter
    from warnings import simplefilter
    ## ignore all future warnings
    simplefilter(action='ignore', category=FutureWarning)
```

```
[ ] ## logestic regression model
    #LR Model
    model_LR = LogisticRegression(solver='liblinear')
    model_LR.fit(X_train,y_train)
```

```
LogisticRegression(solver='liblinear')
```

14. Future scope

- Integrate more diseases into one unified model.
- Use electronic health records and wearable data.
- Add explainability features (e.g., SHAP values).
- Enable doctor-patient communication via the app.

15. Team Members and Roles

Name	Responsibility
Natarajan R	Data preprocessing
Naveen Raj R	Feature engineering and modeling
Tarun V	Model evaluation and optimization
Yokesh K	Deployment and documentation
Suraj SK A	EDA