Machine Learning Project Proposal

DTSC691: Applied Data Science

Nilesh Kadale

Project Overview

This project aims to develop a machine learning model that predicts the likelihood of diabetes in individuals based on key medical parameters such as glucose level, BMI, blood pressure, age, and insulin levels. By identifying early indicators of diabetes, this project seeks to contribute to preventive healthcare decision-making. The final deliverable will be an end-to-end machine learning pipeline integrated into a Flask web application where users can input medical data and receive a real-time prediction result.

Project Goals

Purpose:

The primary purpose of this project is to explore the application of machine learning in healthcare diagnostics. It focuses on improving early detection of diabetes to assist healthcare professionals and patients in timely interventions.

Project Focus:

- Analyze and preprocess healthcare data for model readiness.
- Experiment with various supervised learning algorithms for diabetes classification.
- Evaluate and deploy the best-performing model as a web-based prediction tool.

Specific Goals:

- Perform exploratory data analysis (EDA) to understand data characteristics and relationships.
- Handle missing values, outliers, and feature scaling for optimal model input.
- Train and fine-tune models including Logistic Regression, Random Forest, XGBoost, and a TensorFlow Neural Network.
- Compare model performance using multiple evaluation metrics and interpret model results.
- Deploy the model via a Flask web application.

Expected Outcomes:

- A high-performing machine learning model capable of predicting diabetes risk.
- A functional Flask-based web interface for interactive prediction.
- Complete documentation of data processing, modeling, and deployment steps.

Project Description

Project Objective and Scope:

The objective of this project is to develop an end-to-end machine learning solution that predicts diabetes based on health indicators. The project will utilize the PIMA Indians Diabetes Dataset and will encompass data preprocessing, model training, validation, and deployment.

Data Description:

- Dataset: PIMA Indians Diabetes Database (Kaggle).
- Size: 768 records and 9 columns (8 features and 1 target variable).
- Features: Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, Age.
- Target: Outcome (1 = Diabetic, 0 = Non-Diabetic)

Exploratory Data Analysis:

EDA will include feature correlation analysis, outlier detection, and visualizations such as histograms, pair plots, and heatmaps to better understand relationships.

Data Preparation and Cleaning:

- Replace zero values in key features (Glucose, Insulin, BMI) with NaN.
- Use median imputation for missing data.
- Normalize features with StandardScaler.
- Split the dataset into training (80%) and testing (20%) subsets.

Model Training:

The following algorithms will be trained and compared:

- Logistic Regression – baseline classification.
- Random Forest – handles non-linear feature relationships.
- XGBoost – gradient boosting for higher predictive power.
- TensorFlow Neural Network – deep learning model for pattern recognition.

Hyperparameter tuning (GridSearchCV and early stopping) will be used to optimize models.

Model Evaluation:

Evaluation metrics will include Accuracy, Precision, Recall, F1-Score, and ROC-AUC. Visualization through confusion matrices and ROC curves will support interpretation. Model interpretability tools such as SHAP and LIME will be explored to understand feature influence.

User-Interface Integration:

- Flask web app with an HTML form for user input.
- Preprocess input, generate prediction, display result.
- Deploy locally and optionally on cloud platforms like Render or Railway.
- I will also include the biographical information, a resume page and general project page.

Capstone Complexity

This project demonstrates master's-level complexity by incorporating:
• Multiple advanced models including XGBoost and TensorFlow Neural Networks.
• Integration of machine learning, data preprocessing, and web deployment.
• Use of hyperparameter tuning and interpretability frameworks.
• Combining statistical analysis, machine learning engineering, and UI development in one system.

Software

This project will be developed using Python 3.11 as the primary programming language due to its extensive machine learning ecosystem and efficiency.

• Python 3.11 – chosen for compatibility, community support, and access to ML libraries.
• Pandas and **NumPy** – for data handling, transformations, and statistical computations.
• Scikit-learn – for classical ML algorithms, cross-validation, and metrics.
• XGBoost – for advanced boosting techniques offering speed and interpretability.
• Matplotlib and **Seaborn** – for visualizing data distributions and correlations.
• Flask – lightweight web framework for deploying the trained model.
• Jupyter Notebook / VS Code – environments for code development and experimentation.
• Git / GitHub – version control to manage revisions and maintain project history.

These tools are chosen based on scalability, open-source access, and ability to integrate across stages of data science workflow.

Project Completion Plan

• Week 1 – Environment setup, data collection, and exploratory data analysis.
• Week 2 – Data preprocessing, handling missing values, and feature scaling.
• Week 3 – Train baseline models (Logistic Regression, Random Forest).
• Week 4 – Implement advanced models (XGBoost, TensorFlow Neural Network) and tune hyperparameters.
• Week 5 – Begin personal website development with Flask routes for Homepage, Resume, Projects, and Diabetes Model pages.
• Week 6 – Complete Flask website UI integration and testing of all four pages.
• Week 7 – Final system testing, documentation, and video presentation preparation.

Presentation Plan

The final presentation will be a 30-minute video that walks through the full project lifecycle. It will include:

- An introduction explaining the project motivation and real-world importance.
- A technical overview covering data preprocessing, model training, and evaluation.
- A **live walkthrough of the Flask application code**, showing the structure of routes, templates, and model integration.
- A **demonstration of the deployed web app**, highlighting the input form and real-time prediction process.

- A brief discussion of model results, interpretability, and next steps.

Presentation materials will include PowerPoint slides, a code walkthrough, and a recorded demo of the working application.

Resources

• PIMA Indians Diabetes Dataset – https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database
• Python 3.11 Documentation – https://docs.python.org/3/
• Scikit-learn Documentation – https://scikit-learn.org/stable/
• TensorFlow Tutorials – https://www.tensorflow.org/tutorials
• XGBoost Documentation – https://xgboost.readthedocs.io/
• Flask Documentation – https://flask.palletsprojects.com/