# Income Prediction

# ( Web\_App + Gemini\_API )



# Session 2023 - 2027

# Submitted by:

Muhammad Talha 2023-CS-12

# Supervised by:

Muhammad Kabir Ahmed

# Course:

Artificial Intelligence

# Department of Computer Science

# University of Engineering and Technology, Lahore, Pakistan

Table of Content:

[1. Introduction 3](#_Toc11719)

[2. Dataset 3](#_Toc3546)

[3. Model Training 3](#_Toc6447)

[4. Web Application 3](#_Toc14770)

[5. Gemini API Integration 3](#_Toc276)

[6. Example Output 4](#_Toc7900)

[7. Technologies Used 4](#_Toc23982)

[8. Conclusion 4](#_Toc15591)

[9. Project Poster 5](#_Toc24480)

[10. Linkedin Reference POST: 6](#_Toc15349)

Income Prediction + Gemini AI Assistant

# 1. Introduction

This project combines traditional Machine Learning and modern Generative AI to create a web application that predicts a user's income category and uses Google's Gemini API to provide insights or recommendations based on the prediction. The aim is to enhance understanding and transparency in income-related decision-making.

# 2. Dataset

The dataset used is 'adult.csv', containing 32,000+ records labeled with income classes (≤50K or >50K). It includes features like age, education, occupation, and hours per week. The data was preprocessed using label encoding and standard scaling.

# 3. Model Training

Various machine learning models were tested, including Logistic Regression, SVM, Decision Tree, Random Forest, K-Nearest Neighbors, and Gradient Boosting. The Random Forest classifier delivered the best performance.  
  
• Encoding: sklearn.LabelEncoder  
  
• Scaling: sklearn.StandardScaler  
  
• Final Model: Random Forest Classifier  
  
• Accuracy: **88**%

# 4. Web Application

The web application was developed using Streamlit for a user-friendly interface. It allows users to input personal and professional details, predicts their income category using a trained Random Forest model, and then uses Gemini AI to provide helpful insights and recommendations based on the prediction.

# 5. Gemini API Integration

The prediction result is passed to the Gemini 1.5 Flash model via the Google Generative Language API. The endpoint used was:  
https://generativelanguage.googleapis.com/v1beta/models/gemini-1.5-flash-latest:generateContent  
This API call returns a natural language explanation and tailored recommendations based on the predicted income. The integration was implemented using Python’s requests library.

# 6. Streamlit UI

Figure : Prediction & Gemini Recommendations

Figure : Input from User

# 7. Technologies Used

• **Python** – Core programming language for model development and web app.  
• **Scikit-learn** – For preprocessing, training the Random Forest model, and evaluation.  
• **Streamlit** – To build and deploy the interactive web application.  
• **Google Gemini API (1.5 Flash)** – For generating intelligent, natural language insights based on predictions.  
• **Pickle** – For saving and loading the trained model, scaler, and encoders.  
• **LabelEncoder & StandardScaler** – Used for data preprocessing before prediction.

# 8. Conclusion

This project demonstrates the power of combining traditional machine learning with modern generative AI. It offers an enhanced user experience by not only predicting income categories but also generating suggestions and recommendations through Gemini AI.

# Project Poster

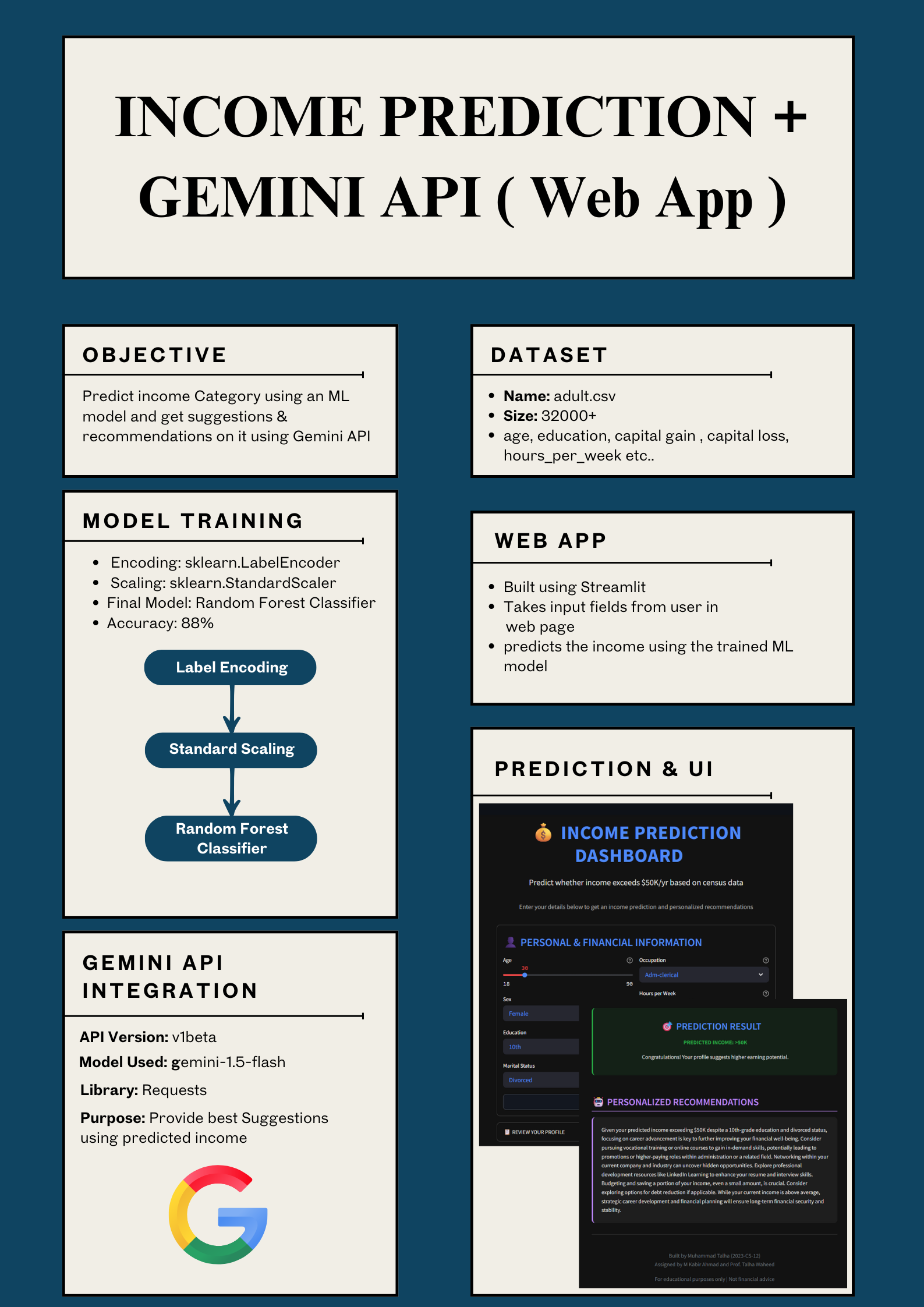


Figure : Poster

# References:

**Deployed App Link:** https://incomepredictionai.streamlit.app/

**Linked In:**