A blue and white logo

AI-generated content may be incorrect.Department of Computer Science, CUI, Islamabad.

A colorful circular logo with a graph and arrow

AI-generated content may be incorrect. AI Income Predictor

Submitted by

Umar Hassan SP23-BSE-052

Zubaria Sajjad SP23-BSE-055

Supervisor

Ms. Farah Saeed

6th June 2025

**Table of Contents**

Abstract 2

1. Introduction 2

2. Proposed System 2

3. Advantages/Benefits of Proposed System 3

4. Scope 3

5. Modules 4

5.1 Module 1: 4

Data Preprocessing and Model Training 4

5.2 Module 2: 4

Graphical User Interface (GUI) & Input Management 4

5.3 Module 3: 5

Prediction Engine & Output Display 5

5.4 Module 4: 5

AI-Powered Recommendations 5

5.5 Module 5: 6

Data Visualization 6

6. System Limitations/Constraints 8

7. Tools and Technologies 8

8. Conclusion 9

9. References 9

**Project Category:**

* **A-** Desktop App

# Abstract

The AI Income Predictor is a desktop application designed to provide individuals with insights into their potential annual income bracket based on various demographic and socio-economic factors. Leveraging machine learning techniques, specifically a Random Forest Classifier trained on a publicly available dataset, the system offers a user-friendly graphical interface. It predicts whether a user's income is likely to be above or below $50,000 per year and provides personalized, rule-based recommendations on how to potentially increase their income, emphasizing actionable areas directly tied to their provided data and the model's insights.

# Introduction

This project proposal document outlines the design and functionality of the AI Income Predictor system. In today's complex economic landscape, understanding the factors that influence personal income is crucial for informed career planning and financial well-being. Many individuals lack clear insights into how their educational background, occupation, work hours, and other attributes correlate with earning potential. This software aims to solve this problem by providing a data-driven prediction and personalized advice system. Through the development of this project, we expect to gain comprehensive skills in machine learning model development (including data preprocessing, feature scaling, model training, and evaluation), user interface design using Tkinter, data visualization, and the implementation of rule-based recommendation systems.

# Proposed System

The proposed AI Income Predictor system addresses the aforementioned problem by offering a transparent and interactive platform for income estimation. It works by collecting key personal and professional data points from the user through a series of input fields and dropdowns. This input is then preprocessed (encoded and scaled) to match the format required by the trained machine learning model. The core of the system is a Random Forest Classifier, which, based on the processed input, predicts the user's income bracket (either less than or equal to $50,000, or greater than $50,000). Beyond mere prediction, the system calculates the model's confidence in its forecast and identifies the most influential factors from the user's input. It then leverages these insights to generate personalized, rule-based recommendations, guiding the user on potential strategies for income improvement.

# Advantages/Benefits of Proposed System

* **Personalized Insights:** Provides tailored recommendations based on the user's specific input data and the model's identified influential factors.
* **Data-Driven Decision Making:** Empowers users with quantifiable insights into how various personal and professional attributes impact their income potential.
* **User-Friendly Interface:** Offers an intuitive and accessible Graphical User Interface (GUI) built with Tkinter, making complex machine learning models approachable for non-technical users.
* **Proactive Planning:** Helps individuals identify actionable areas (e.g., education, work hours, investments) where changes might lead to income growth.
* **Educational Value:** Includes data visualizations that explain the model's performance and the relative importance of different features, fostering a better understanding of income analytics and most importantly it is not gender-biased predictor.
* **Privacy-Conscious:** All data processing and model predictions are performed locally on the user's machine, with prediction history saved to a local CSV file, minimizing external data sharing concerns.

# Scope

The AI Income Predictor project is designed to be a comprehensive desktop application focused on personal income prediction and guidance. Its main functionalities include: collecting structured demographic and financial inputs from the user via a guided form; applying a pre-trained machine learning model (Random Forest Classifier) to predict an individual's likely annual income bracket (less than or equal to $50,000, or greater than $50,000); calculating and displaying the model's confidence in its prediction; identifying and presenting the most influential input features for the specific prediction; and generating personalized, rule-based recommendations aimed at improving the user's income potential based on their provided data. The project also encompasses various data visualizations to explain the model's overall performance and key insights derived from the dataset. The scope of this project is limited to a standalone desktop application. It does not include real-time data fetching from external economic or job market APIs, nor does it incorporate advanced generative AI models for its recommendations. Furthermore, it is designed for single-user operation and does not feature multi-user capabilities, cloud synchronization, or web deployment.

# Modules

## Module 1:

## Data Preprocessing and Model Training

This module is responsible for preparing the raw adult.csv dataset for machine learning. It handles crucial steps such as identifying and replacing missing values, dropping irrelevant columns (fnlwgt), and converting categorical features into numerical formats using LabelEncoder. Numerical features are scaled using StandardScaler to ensure consistent data distribution for the model. Finally, the module splits the processed data into training and testing sets and either loads a pre-trained RandomForestClassifier model or trains a new one using GridSearchCV for hyperparameter optimization, saving the best model using pickle.

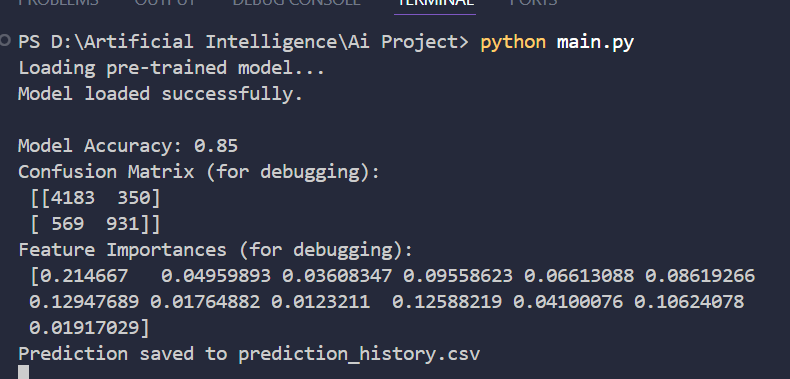


Figure 1 Terminal Screen (Loading Pre-Trained Model)

## Module 2:

## Graphical User Interface (GUI) & Input Management

This module develops the interactive user interface using the Tkinter library. It constructs the main application window, including dynamically generated input fields (Entry widgets for numerical data and Combobox widgets for categorical data) for all required features. It manages the layout and styling of these inputs for user clarity and aesthetic appeal. This module also incorporates input validation logic to ensure data integrity before feeding it to the machine learning model.

## Module 3:

## Prediction Engine & Output Display

At the heart of the application, this module takes the validated and preprocessed user input and feeds it into the trained RandomForestClassifier model. It calculates the income bracket prediction (<=50K or >50K) and the associated probability (confidence score). Additionally, it provides a heuristic estimation of the actual income value. The module then presents these prediction results, along with the most influential features derived from the model's feature importances, in a clear and informative new window, giving the user an immediate understanding of their income outlook.

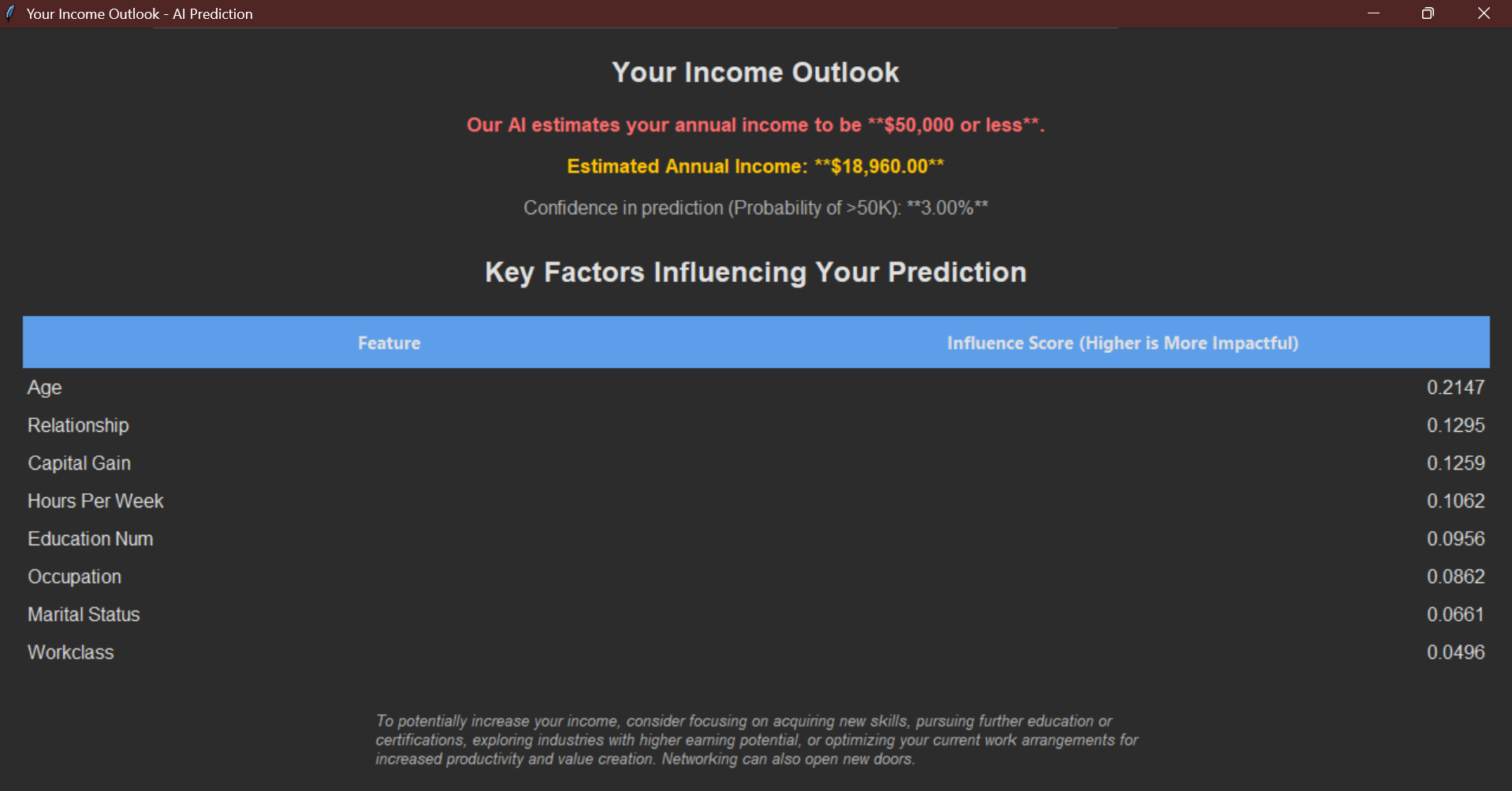


Figure 2 Output Screen (Predict Income)

## Module 4:

## AI-Powered Recommendations

This module provides personalized, rule-based recommendations designed to help users potentially increase their income. After a prediction is made, this module analyzes the user's specific input values in relation to the model's most impactful features. It identifies areas where changes could be beneficial (e.g., suggesting further education if education.num is low and important, or advising on investments if capital.gain is zero). The recommendations are tailored to the user's data and the predicted income class, offering actionable tips for improvement.

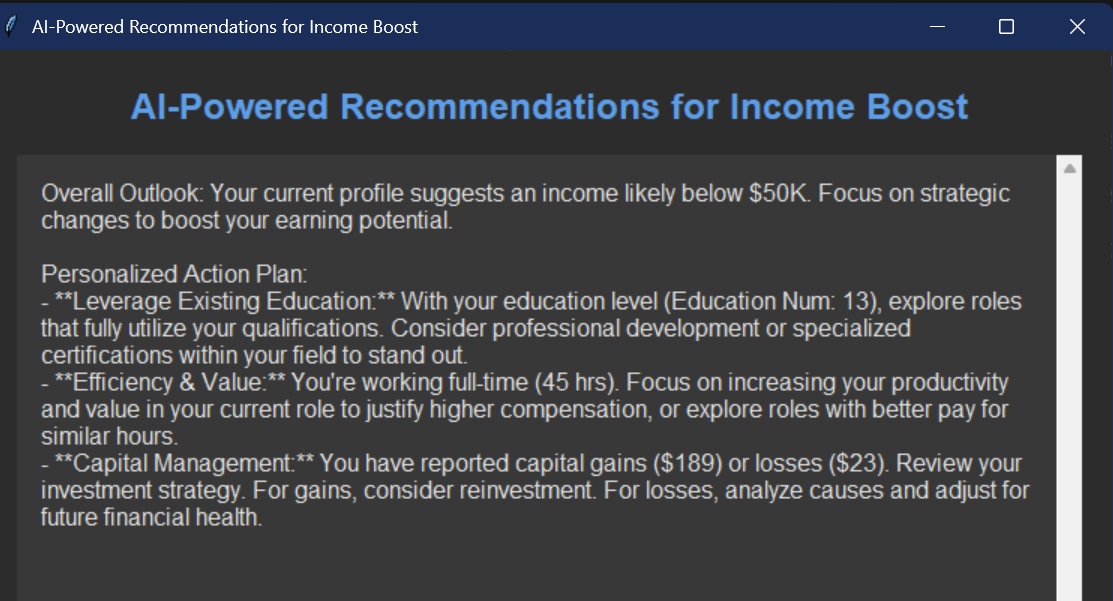


Figure 3 Recommendation Screen

## Module 5:

## Data Visualization

This module utilizes Matplotlib and Seaborn to generate various insightful graphs that explain the model's performance and underlying data characteristics. These visualizations include a Feature Importance plot (showing which features are most critical to the prediction), a Probability Histogram (distribution of prediction probabilities), a Confusion Matrix (evaluating prediction accuracy), a Correlation Heatmap (showing relationships between numerical features), and a Cumulative Probability plot. These visuals help users and developers understand the model's behavior and the dataset better.

A graph on a computer screen

AI-generated content may be incorrect.

Figure 4 Cumulative Probability of Earning> $50K

A screenshot of a computer

AI-generated content may be incorrect.

Figure 5 Correlation Mapping

A screenshot of a computer

AI-generated content may be incorrect.

Figure 6 Confusion Matrix Showing Accuracy of Model

A screen shot of a graph

AI-generated content may be incorrect.

Figure 7 Probability Histogram (>$50K)

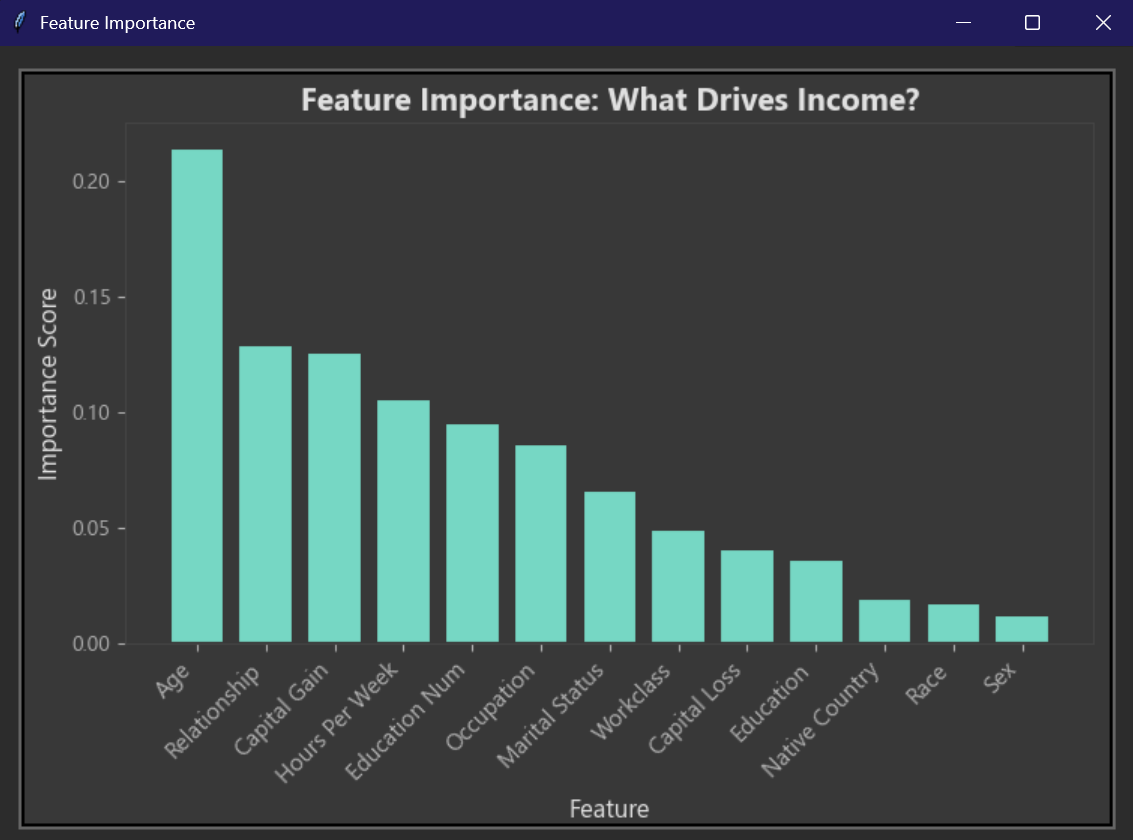


Figure 8 Feature Importance Graph

# System Limitations/Constraints

* **Data Dependency:** The accuracy and relevance of predictions are directly dependent on the quality, size, and representativeness of the adult.csv dataset used for training. Biases present in the dataset may be reflected in the model's predictions.
* **Generalization Limitations:** The model's performance is optimized for the patterns found in the training data. It may not generalize perfectly to individuals outside the dataset's demographic or socio-economic scope, or to future economic conditions.
* **Rule-Based Recommendations:** The AI-powered recommendations are based on predefined rules derived from feature importances and thresholds, rather than advanced generative AI. This limits their depth and nuance compared to a large language model.
* **No Real-time Data Integration:** The application does not pull real-time data from external APIs (e.g., current job market trends, inflation rates, real-time salary benchmarks), which could provide more up-to-date recommendations.

# Tools and Technologies

Table 1 Tools and Technologies for Project

|  |  |  |  |
| --- | --- | --- | --- |
| **Tools**  **And**  **Technologies** | **Tools & Technology** | **Version** | **Rationale** |
| Vs Code | 1.100 | IDE |
| Python | 3.13 | Programming language |
| Scikit-learn | 1.0 | Essential machine learning library for data preprocessing |

# Conclusion

The AI Income Predictor project successfully delivers a robust desktop application that empowers users with data-driven insights into their income potential. By effectively utilizing machine learning for prediction and providing personalized, actionable recommendations, the system serves as a valuable tool for career and financial planning. Its intuitive GUI and informative visualizations enhance user understanding, making complex analytical results accessible. This project stands as a testament to the practical application of AI in personal development, offering a foundational step towards more informed decision-making regarding professional and financial growth.

# References

* **UCI Machine Learning Repository:** Adult Dataset. Available at: <https://archive.ics.uci.edu/ml/datasets/Adult>
* **Scikit-learn Documentation:** Comprehensive documentation for machine learning algorithms and tools. Available at: <https://scikit-learn.org/stable/documentation.html>
* **Pandas Documentation:** Official documentation for the pandas data analysis library. Available at: <https://pandas.pydata.org/docs/>
* **Matplotlib Documentation:** Official user guide and examples for the Matplotlib plotting library. Available at: <https://matplotlib.org/stable/contents.html>
* **Real Python:** Tutorials and articles on various Python topics, including machine learning and GUI development. Available at: <https://realpython.com/>
* **Towards Data Science:** A popular Medium publication featuring articles on data science, machine learning, and AI. Available at: <https://towardsdatascience.com/>