**Career Recommendation System**

**Project Report**

**Abstract**

This project introduces a machine learning model for recommending career paths to students based on academic scores, personal interests, and extracurricular activities. Using a dataset of student performance and characteristics, several classification models were evaluated, with Random Forest achieving the best performance. The model is deployed via a web app, allowing students to input their data and receive career recommendations with associated probabilities. This tool provides valuable insights for students exploring suitable career options based on their unique profiles.

**1. Introduction**

Selecting an appropriate career path is a critical decision for students, and traditional career guidance methods may not always provide personalized suggestions. This project leverages machine learning to recommend career paths based on a student's academic performance, weekly study hours, and involvement in extracurricular activities. By inputting their details into an interactive web app, students receive personalized career suggestions, helping them make data-informed choices about their futures.

**2. Literature Review**

Machine learning algorithms, particularly classification models, have proven effective for recommendation systems in areas like job guidance and educational planning. Studies have shown that models such as Decision Trees, Support Vector Machines, and Random Forests can predict career pathways based on user characteristics. In this project, we employ the Random Forest Classifier due to its ability to handle multiclass classification tasks and its interpretability, making it suitable for career prediction based on a diverse set of features.

**2. Dataset Overview**

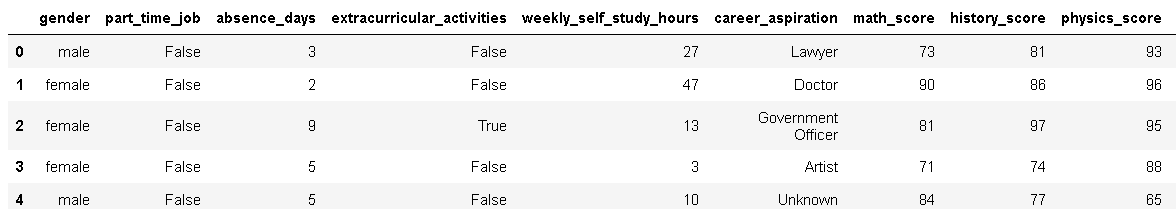
**Link:** [**https://www.kaggle.com/datasets/noorsaeed/student-studeis-recommendation**](https://www.kaggle.com/datasets/noorsaeed/student-studeis-recommendation)

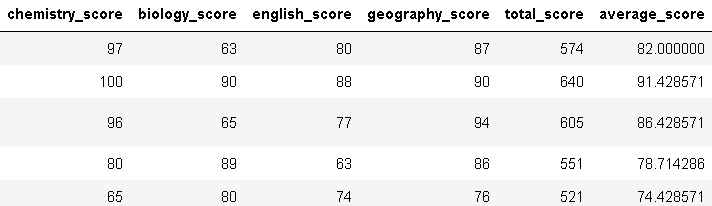
**Dataset Goal**: To predict career aspirations for students based on academic performance, personal interests, and extracurricular activities.

**Features**:

The dataset contains multiple attributes, including:

* **Input Variables**:
  + **Gender**
  + **Part-time Job** (Yes/No)
  + **Absence Days**
  + **Extracurricular Activities** (Yes/No)
  + **Weekly Self-Study Hours**
  + Scores for core subjects: Math, History, Physics, Chemistry, Biology, English, and Geography.
  + **Total Score** and **Average Score** calculated from the subject scores.





* **Target Variable**: Career aspiration with 17 possible categories, including:
  + Lawyer, Doctor, Government Officer, Artist, Software Engineer, Teacher, Scientist, etc.

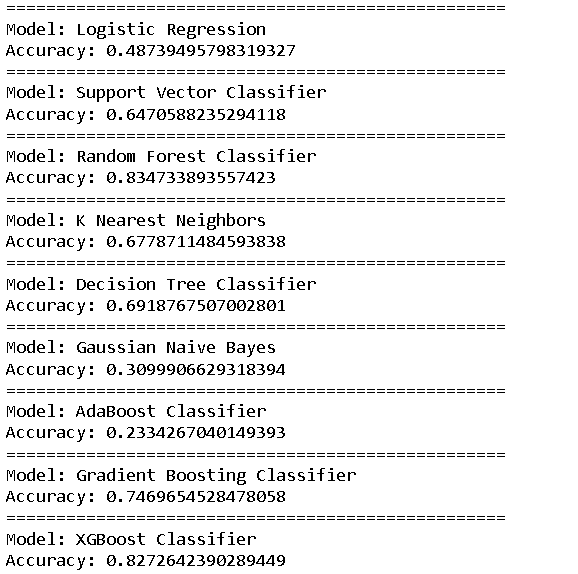
**Data Characteristics**:

* **Multiclass Classification**: Predicts one of 17 career categories.
* **Data Preprocessing**:
  + Categorical encoding for variables like Gender and Part-Time Job.
  + Balancing classes with SMOTE to ensure fair representation of each career category.

**3. Model Selection and Development**

**Modeling Approach**:

To accurately recommend careers, several classification models were evaluated, including Logistic Regression, Decision Trees, and Random Forest. The Random Forest Classifier demonstrated the highest accuracy, making it the model of choice for deployment.



**Data Preprocessing**:

* **Feature Encoding**: Converted categorical variables like Gender and Part-Time Job to numerical values.
* **Scaling**: Applied StandardScaler for standardizing features.
* **Train-Test Split**: Data was split into training and testing sets in an 80:20 ratio.

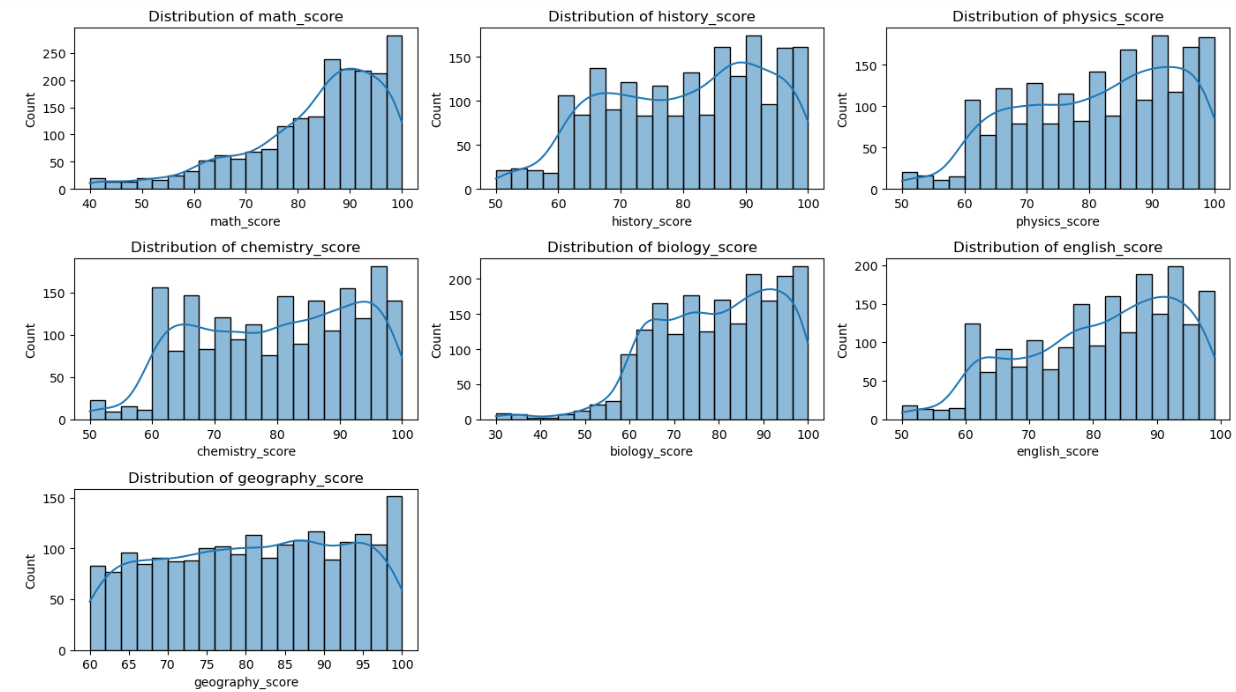


**Model Performance**:

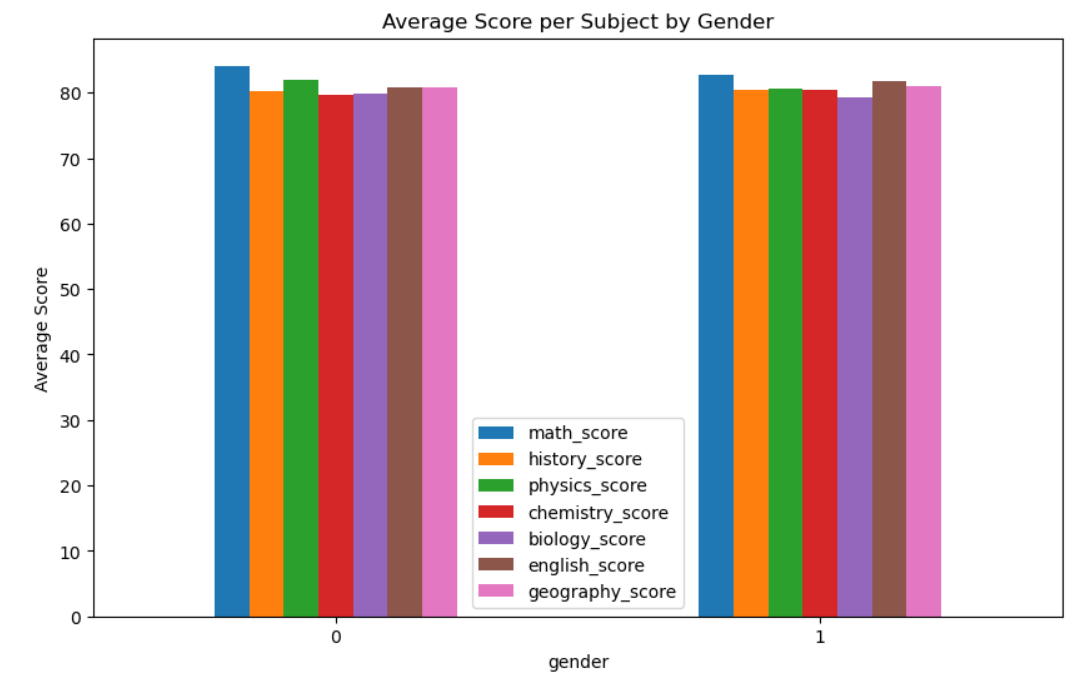
* **Accuracy**: The Random Forest Classifier achieved high accuracy on the test set, validating its suitability for the multiclass classification task.
* **Evaluation Metrics**: Performance was evaluated using accuracy, precision, and recall scores, as well as a classification report.

**Graphs and Visualization:**

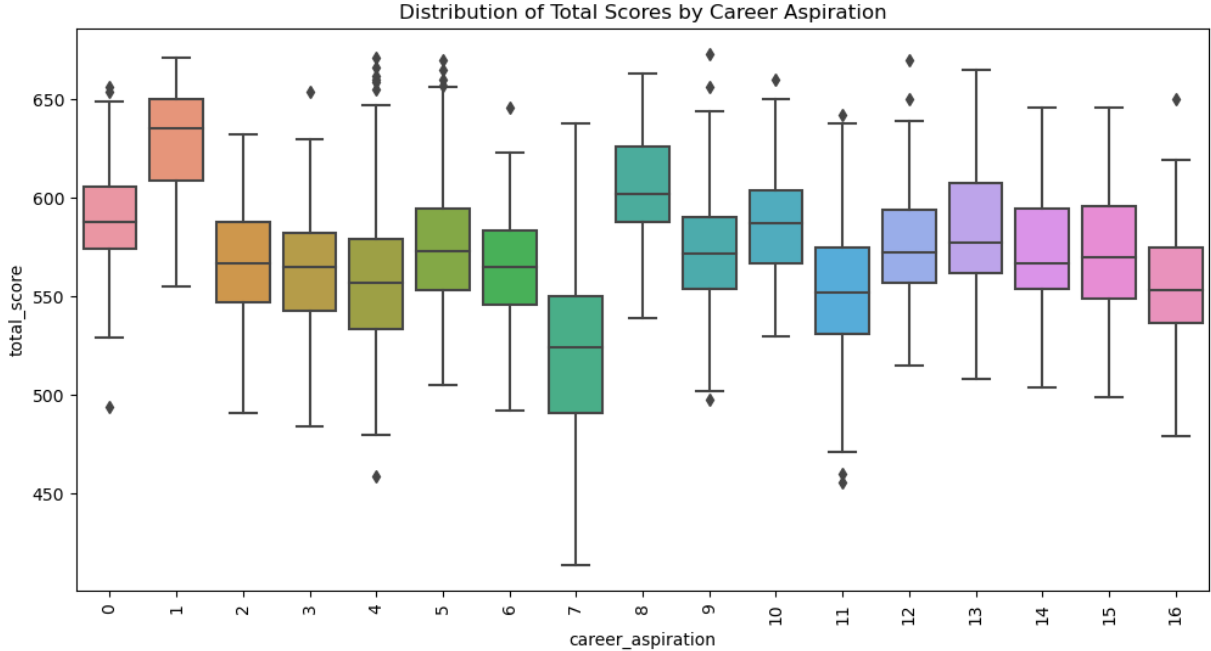
* **Distribution of Scores for Each Subject:**



**Average Score per Gender:**

****

**Box Plot of Total Scores by Career Aspiration:**

****

**4. Project Structure**

**Files**:

* **app.py**: Main script for the Streamlit app.
* **student-scores.csv**: Dataset containing student data and career aspirations.

**Libraries Used**:

* **Streamlit**: For the user interface.
* **Pandas**: For data manipulation.
* **Scikit-learn**: For machine learning modeling and evaluation metrics.
* **Imbalanced-learn**: For data balancing using SMOTE.

**5. Application Usage**

**How to Run the App**:

1. **Install Required Libraries**: Ensure you have the required packages installed by running:

pip install streamlit pandas scikit-learn imbalanced-learn

1. **Run the App**: Use the following command to start the Streamlit app:

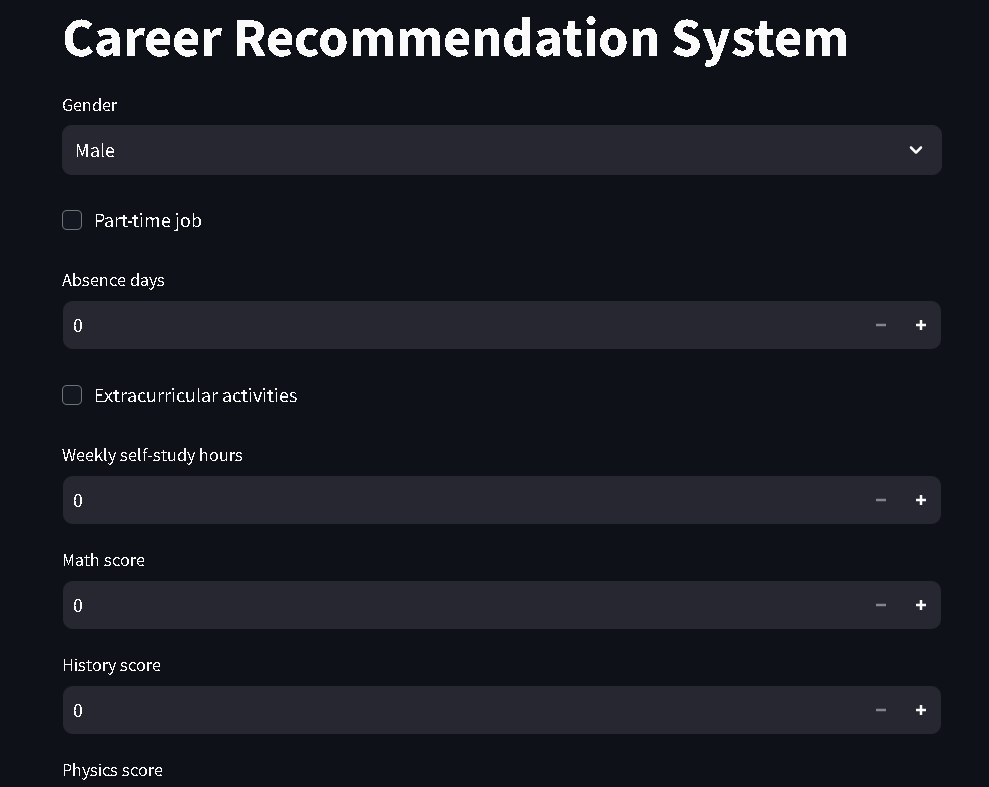
streamlit run app.py

**Interacting with the App**:

The app will prompt users to enter details such as gender, weekly study hours, and subject scores. After submitting the input, the app provides top career recommendations with associated probabilities, offering insights into which career paths align best with the student's profile.

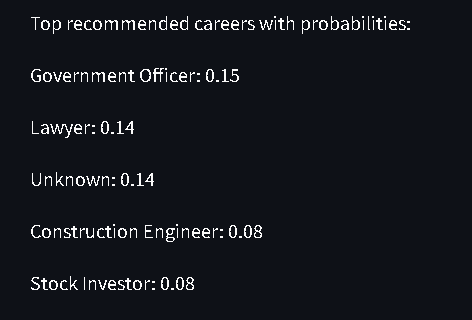
**Example Input**:

* **Gender**: Female
* **Weekly Self-Study Hours**: 8
* **Scores**: Math (78), History (85), Physics (90), etc.



**Example Output**:

* **Top Career Recommendations**:
  + Software Engineer (Probability: 0.72)
  + Teacher (Probability: 0.15)
  + Scientist (Probability: 0.08)



**6. Model Details**

**Model Training**:

* **Model**: RandomForestClassifier
* **Objective**: To predict the most suitable career paths for students based on their data.
* **Data Balancing**: Applied SMOTE to balance the target classes.

**Model Evaluation**:

* Model performance was validated using test data, achieving strong accuracy, precision, and recall metrics.

**7. Frontend and Backend Code**

**Frontend (Streamlit)**: The frontend interface is developed using Streamlit, allowing students to input relevant information and receive career recommendations.

**Backend (Random Forest Model)**: The backend consists of a trained Random Forest model. The model takes the input data, scales it, and returns the top career recommendations with probabilities.

**8. Future Work**

Future improvements to this Career Recommendation System could include:

* **Incorporating Personality Traits**: Adding more features, like personality scores, could improve recommendation accuracy.
* **Model Optimization**: Fine-tuning the Random Forest model and experimenting with other algorithms, like Neural Networks, could boost performance.
* **Deployment Enhancements**: Hosting the web app on a cloud platform (e.g., AWS or Heroku) would make it accessible to a wider audience.

**9. Conclusion**

This project successfully demonstrates the potential of machine learning in career counseling by predicting career paths based on student performance and personal attributes. The interactive app provides students with data-driven recommendations, supporting informed career decisions and potentially enhancing career satisfaction.

**10. References**

* Random Forest in Classification Tasks: Scikit-learn Documentation.
* Imbalanced Dataset Handling: Imbalanced-learn Library.