**Machine Learning Project Report**

**Aim of the model:** using randomforest model to predict the winner in the boxing matches by analyzing the data and insights of each fighter.

**Dataset’s brief description:** The first dataset used is fighters data, which includes the personal information of each boxer such as name, age, wins, and other features. The second dataset is popular matches, it contains information about the matches such as date, palace, opponents, the punches power, and others.

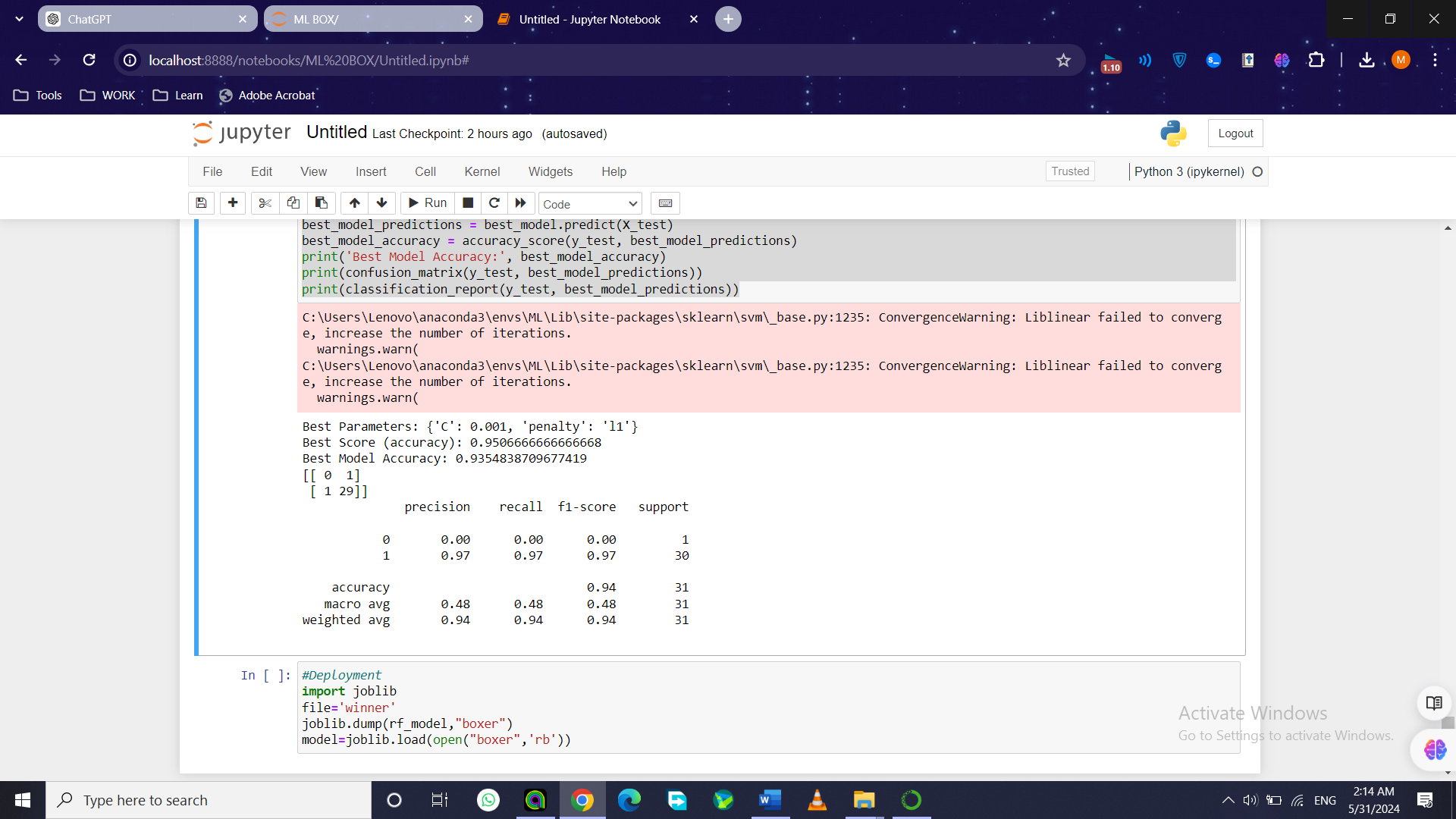
**Codes brief:**

1. **Libraries:** Import all the libraries which will be used such as pandas, seaborn, sklearn.linear\_model and all the used libraries.
2. **Datasets:** Read the 2 datasets files which are fighters and popular matches and their extension is CSV files.
3. **Data cleaning:** The data cleaning process included converting the age column in fighters\_df to numeric values, setting invalid and negative ages to NaN. In matches\_df, 'Unknown' entries were replaced with NaN, and several specified columns were converted to numeric, with invalid entries set to NaN. Negative values in the opponent\_1\_rounds\_boxed column were changed to NaN. A winner column was created based on the verdict to identify the winner. The cleaned datasets were then previewed using head().
4. A blue rectangular bar graph

   Description automatically generated with medium confidence**EDA:** some plots are produced: The first 2 plots show the distribution of all the numerical features in the 2 datasets via histograms. The 3rd plot shows the ratio of stances of the boxers via count plot. The 4th one shows the opponent 1 vs opponent 2 estimated punch power via a scatterplot.

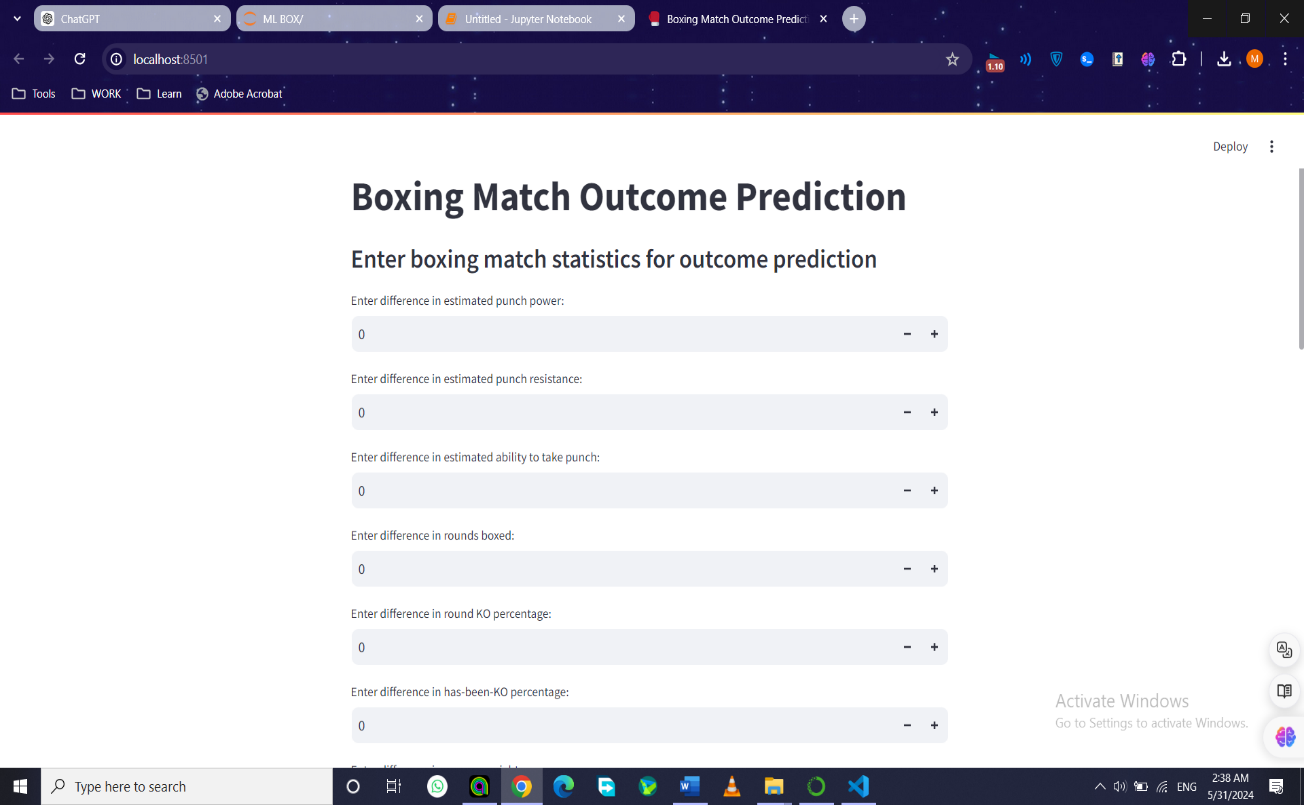
**Sample of the plots produced**

1. **Feature engineering:** Feature engineering in matches\_df entailed crafting difference and ratio features to represent variations and proportions between opponents across attributes like punch power, punch resistance, ability to endure punches, rounds boxed, round KO percentage, KO percentage, and average weight. Furthermore, interaction features were devised by computing the product of each opponent's punch power and punch resistance. These additions enrich the dataset by offering deeper insights into the comparative dynamics and interplay of fighters' characteristics. All of these features are added to improve the prediction accuracy and the model’s performance.
2. **Model:** During modeling, significant features and the target variable were chosen from the dataset, emphasizing variations, proportions, and interaction characteristics related to punch attributes, round numbers, KO percentages, and average weight. The dataset underwent splitting into training and testing subsets with an 80:20 ratio, ensuring an equitable representation of classes. Three classification methods—Logistic Regression, Random Forest, and Support Vector Machine (SVM)—were applied and assessed for their effectiveness in predicting match winners. Logistic Regression was trained on the training data and used for predictions on the test set, with accuracy scores calculated and compared across all models. The top-performing model, which achieved the highest accuracy, was selected as the optimal choice for predicting match outcomes.
3. **Logistic regression tuning:** This phase involved refining logistic regression parameters through grid search. Various values for regularization strength (C) and penalty types (penalty) were explored within the parameter grid. The logistic regression model was configured with the 'liblinear' solver and a random state set to 42. Grid search was conducted using a 5-fold cross-validation method, aiming to optimize accuracy. The best parameters and their corresponding accuracy score were determined through this process. Following that, predictions were made using the best model on the test set, and its performance in predicting match winners was assessed by computing accuracy, confusion matrix, and classification report metrics.

****

**That is the result of the tuning the regularization was strong as c=0.001 and it was Lasso regularization technique and this is the detailed report**

1. **Deployment on Streamlit:** Firstly, the Random Forest model was stored using the joblib library's dump function, with the filename 'boxer'. Subsequently, the model was reloaded into memory utilizing joblib's load function. This method facilitates the retention and retrieval of the trained model for subsequent predictions, eliminating the need for retraining. Then, the deployment code in written in an individual pyhton file called “Winner”. This web tool enables users to predict boxing match outcomes based on entered match details. Employing a trained Random Forest model stored as 'boxer', individuals input various match parameters, including disparities and proportions in estimated punch attributes, round figures, KO percentages, and average weight for each fighter. After input submission, the app delivers predictions. The 'About' section sheds light on the app's purpose, while the 'Contact' page offers contact information for queries. Data utilized for model training and testing is sourced from 'fighters.csv' and 'popular\_matches.csv'.



A screenshot of a computer

Description automatically generated**That is and example for the deployment all of the entered data are the information of the opponent 2 performance in the 14 row(index 13) in the dataset popular matches after cleaning and add the new features. The output is the winner is 0 which is opponent 2 if it printed 1 then opponent 1 is the winner is this match.**

**Limitations:** The used datasets are too small, the dataset fighters should include more important features such as championships and experience of each fighter to validate more accurate predictions.

**Conclusion:** In summary, this project utilized a Random Forest model to forecast winners in boxing matches by analyzing two datasets: one containing fighters' personal details and the other comprising match information. The project encompassed various stages, including data cleaning, exploratory data analysis, feature engineering, model selection, and deployment via Streamlit for user interaction. However, it's essential to recognize the project's limitations, notably the small size of the datasets and the absence of key features like height, reach, and championships in the fighters dataset. Despite these constraints, the project showcases the potential of machine learning in sports analytics and lays the groundwork for future enhancements and expansions of the model.