**Data Wrangling**

Multiple datasets, including the open-source Vaastav GitHub Repository, the Understat API, and the Official Fantasy Premier League (FPL) API, were integrated to create a unified dataset for performance analysis. This required handling asynchronous API calls, aligning inconsistent data, and merging datasets with different structures and levels of granularity.

Understat data, containing advanced metrics like expected goals (xG) and expected assists (xA), was accessed using Python’s *understat module*, which simplifies API interactions. Additionally, asynchronous API requests were made using Python’s *asyncio* and *aiohttp* libraries, enabling efficient parallel data retrieval for match-level statistics across multiple seasons. The JSON responses from these APIs were parsed and transformed into DataFrames for further processing.

Vaastav’s data, available in downloadable CSV format, provided essential FPL-specific statistics. These datasets were loaded into DataFrames, where column names were standardized. Handling discrepancies in naming conventions between Understat and Vaastav required preprocessing player names by standardizing text case, stripping extra spaces, and resolving known mismatches manually. Player data was further cross-referenced by team and season to ensure alignment across datasets.

Merging the Understat and Vaastav datasets posed unique challenges due to differences in granularity. Vaastav’s data is organized by gameweek, while Understat operates on a match-level basis. This added a layer of complexity because gameweeks in FPL are not fixed and can vary based on match rescheduling. To address this, a custom counter was implemented to map Understat matches to corresponding FPL gameweeks, ensuring expected goals, assists, and goal involvements aligned correctly.

After resolving these challenges, the seasonal data was concatenated into a single DataFrame, and the wrangled dataset was saved as a master CSV file for further cleaning. The integration of the Understat module, asynchronous data retrieval, and meticulous data processing created a robust foundation for analyzing FPL player performance.

**Exploratory Data Analysis**

Data Cleaning

The master CSV file was further cleaned to transform raw player data into analysis-ready format. After loading the primary dataset and inspecting its structure, missing values and duplicate rows were dropped, and categorical columns, such as player position and team name, were standardized by resolving inconsistencies.

To handle non-alphanumeric characters in player names, a custom function replaced recurring symbols with their accent-less counterparts, ensuring uniformity. Time-related fields were extracted from the *kickoff\_time* column to derive features like hour, day of the week, and month. A custom function was created to assign seasons based on match dates.

Numerical data was rigorously checked for outliers and logical inconsistencies. For instance, invalid rows where goals scored were positive but expected goals (xG) were zero were removed. Negative values in the xP (expected points) column were imputed using the mean xP from adjacent gameweeks, ensuring continuity without jumping seasons.

Finally, dervied metrics like goals and assists per minute, as well as cumulative points, goals, assists, and their expected counterparts, were calculated for each player per season. The cleaned dataset was saved as a CSV file, *master\_cleaned*, to be used for visualization and modeling.