ScoreSight - Predicting EPL Points Tally, Top Scorer

Project Statement:

The goal of this project is to build ML models to predict the outcomes of the upcoming English Premier League season. Predicting the results and winners of matches and the overall points tally and winner of the League. Based on a players previous years performances predict what their output (goals and assists) could be in the upcoming season.

Use Cases: The project addresses two primary use cases:

Predicting Result:

• Description: Based on previous years' results, historic data, home and away stadium and how a team tends to perform against another we can predict what the score outcome of that match could be. Based on the score prediction we can get winner and points achieve per match and can get the eventual points and Winner of the league.

Predicting the Top Scorer:

• Description: Based on previous years' performance and player output we can try to predict how many goals and assists and player will be able to achieve for the upcoming season. Players could be playing for the first time so their previous history with previous clubs.

Outcomes:

By the end of this project, students will:

• Understand the principles of data preprocessing, training classification and regression model and evaluating them

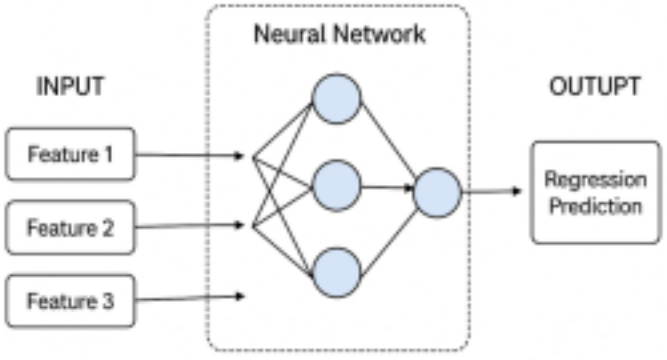
• Develop a model to predict scores and player performances.

• Implement regression models and measure their accuracy.

• Prepare detailed documentation and a presentation of their findings and results.

Dataset:

Model Architecture:



Modules to be implemented

1. Data preprocessing and feature selection.
    - The data available in the dataset might not be in the exact format required to train the model and might have missing values or categorical values.
    - We have to first make this data suited to our model. We need to get the columns that we specifically need for our use case.
2. Building a Regression model
    - Regression is type of machine learning model where we try to predict a **y** for a given **X**. Over here the **y** is the unknown and the **X** is the know variables.
    - In our case the **y** would be player performance or match score and the **X** would be team1, team2, home or away, previous history, etc.
3. Evaluation and Fine-tuning
    - Evaluate the regression model using metrics like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), etc. and tweaking the model to give us more accurate results.
4. Documentation and Presentation Preparation
    - Prepare documentation and presentation to showcase the project.

Week-wise module implementation and high-level requirements with output screenshots Milestone

1: Week 1: Project Initialization and Dataset Acquisition.

- Understand the project goals and outcomes.
- Collect datasets for EPL score prediction and player performance.
- Explore the dataset structure and sample data.

Week 2: Data Preprocessing and Feature Selection.

- Clean up the data and form new columns of data required as per the use case.
- Choose the columns most suited for the task and which are most relevant for the use case.

Milestone 2:

Week 3: Preliminary Model Training.

- Create some regression models and train them on the pre-processed data to predict outcomes. •

Look at the loss and look for where improvements can be made.

Week 4: Predictions and fine tuning the model.

- Make predictions with the model.
- Change different hyper parameters to give the model more opportunity to learn.

Milestone 3:

Week 5: Fine tune the data and try out more model architectures.

- Looking back at the data to see any more improvements that can be made.
- Retrain the model with new data and train the new models.
- Compare different models' performance.

Week 6: Inference with the test data.

- Load the model separately and test it out to see if the model's performance is satisfactory. • Go back to pre-processing and training and adjust parameters to see if better performance can be achieved

Milestone 4:

Week 7: Build a Full stack app for users to use your model.

- To let everyone use the model in a easier way with a GUI. Build a simple website with 2 screens to use the model.
- Load the models and encoders in the backend and make use of pipelines to make the process more streamlined.

Week 8: Documentation, Presentation, and Demo Preparation

- Compile project documentation, including methodology, results, and conclusions.
- Prepare a presentation summarizing the project.
- Conduct a final review and rehearsal of the presentation and demo.

Evaluation Criteria:

1. **Completion of Milestones:** Assess the extent to which each milestone was achieved within the designated timeline. This includes successful dataset acquisition, model development, integration, and evaluation.
2. **Accuracy of the Predictions:** Compare the model's prediction with real life data from the newer EPL season to see if the predictions were close.
3. **Clarity and Depth of Documentation and Presentation:** Review the final documentation for completeness, clarity, and technical depth. Assess the presentation and demo for their ability to clearly convey the project's objectives, methodology, results, and conclusions. This includes the quality of the visual aids, the coherence of the narrative, and the responsiveness to questions during the demo.

Model Performance - Quantitative Metrics:

- **Match Score Prediction And Player Performance Prediction.**
  - Metric: Metric: Mean Absolute Error (MAE)
  - Description: Measures the average absolute difference between the predicted and actual total runs scored in an inning. Lower MAE indicates better predictive accuracy.
  - Formula: $MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i|$

- y_i: Actual Value
- $\hat{y}_i$: Predicted Value

Additional Metrics:

- **Model Training Metrics:**
  - Metric: Loss Curves (Training and Validation Loss)
  - Description: Tracks the loss during training and validation to ensure the models are learning effectively and to prevent overfitting.

Example Quantitative Metrics for Evaluation

1. Match Score and Player Performance.

- Goal: Achieve a low Train and Test loss