**EE 456 Final Project Written Report**

**Abhinav Reddy Sama**

**Table of Contents**

**Abstract 3**

**Outline 4**

**Project Discussion 5**

**Results 7**

**Conclusion** 9

**Abstract**

This project investigates the use of a feedforward neural network to predict football match outcomes (Win, Draw, Loss) based on historical data and derived team statistics. The dataset includes key performance indicators such as goals scored, goals conceded, venue type (home/away), and recent form. Notably, the project incorporates advanced team-based features, such as rolling averages of goals scored/conceded, win rates, goal differences, and recent form scores.

A key highlight of the project is the implementation of cross-validation for model evaluation, coupled with hyperparameter tuning to optimize the neural network’s architecture and learning process. The best-performing model, with 128 neurons in the first hidden layer and 64 neurons in the second, achieved a cross-validation accuracy of 51%.

This result, while modest, emphasizes the predictive challenge inherent in football outcomes due to their stochastic nature. Feature importance analysis further revealed that recent team form and average goals significantly influence predictions, highlighting the importance of historical performance data.

**Outline**

**Main Idea**

The primary objective of this project was to design a machine learning pipeline capable of predicting football match outcomes. This was motivated by the complex and unpredictable nature of sports analytics, where outcomes depend on team performance, venue, and historical trends. Neural networks were chosen due to their ability to capture complex patterns in numerical data.

**Tools Utilized**

* Python: Programming language for data preprocessing, modeling, and evaluation.
* Libraries:
  + PyTorch: Deep learning framework for building and training the neural network.
  + Scikit-learn: For preprocessing, cross-validation, and model evaluation metrics.
  + Pandas and NumPy: For data manipulation and feature engineering.
  + Matplotlib and Seaborn: For visualizing results (confusion matrices, feature importance, and accuracy).
* Stratified K-Fold Cross-Validation: Ensured robust evaluation by splitting data into multiple folds.

**Project Discussion**

**Data Preparation and Feature Engineering**

The dataset consisted of historical match statistics, including goals scored (gf), goals conceded (ga), and venue type (home/away). To improve the model’s predictive ability, additional team-based features were engineered:

* Average Goals Scored (avg\_gf): Rolling average of goals scored over the past 3 matches.
* Average Goals Conceded (avg\_ga): Rolling average of goals conceded.
* Goal Difference (avg\_goal\_diff): Average difference between goals scored and goals conceded.
* Win Rate: Proportion of wins in the past 3 matches.
* Recent Form Score: Weighted rolling average of recent results (Win = 3 points, Draw = 1 point, Loss = 0 points).
* Recent Scored and Conceded Goals: Goals scored and conceded over the past 5 matches.

**Assumptions:**

* The rolling averages are calculated using a window of 3-5 matches, assuming recent form better reflects team performance.
* Venue (home or away) has a psychological and tactical impact on match outcomes.

**Neural Network Design:**

The feedforward neural network comprised:

* Input Layer: 8 features representing the team statistics.
* Hidden Layers:
* First layer: 128 neurons with ReLU activation and batch normalization.
* Second layer: 64 neurons with ReLU activation and dropout regularization (0.3).
* Output Layer: 3 neurons corresponding to the classes (Win, Draw, Loss) with softmax activation.

**Training Configuration:**

* Loss Function: Cross-Entropy Loss.
* Optimizer: Adam optimizer with a learning rate scheduler.
* Batch Size: Tested with 16 and 32.
* Epochs: Limited to 30 epochs per fold to balance time and performance.

**Cross-Validation and Hyperparameters Tuning**

Stratified K-Fold Cross-Validation with 5 folds was implemented to ensure robust model evaluation across all classes. Hyperparameters, including the number of neurons, learning rate, and batch size, were optimized through grid search.

**Hyperparameter Grid:**

Hidden Layer Sizes: [64, 128] for the first layer and [32, 64] for the second layer.

Learning Rates: [0.001, 0.0005].

Batch Sizes: [16, 32].

**Results**

**Cross-Validation Accuracy**

The best model configuration was found to have:

* Hidden Layer 1: 128 neurons
* Hidden Layer 2: 64 neurons
* Learning Rate: 0.001
* Batch Size: 16

The corresponding cross-validation accuracy was 51.02%, highlighting the challenge of predicting football match outcomes.

**Confusion Matrix**

The confusion matrix below indicates the model’s performance across the three classes (Win, Draw, Loss):

A blue squares with numbers and a number

Description automatically generated with medium confidence

The model performs best in predicting Draws, but struggles with Losses and Wins, which could be due to class imbalance.

**Feature Importance**

Feature importance was derived from the input layer’s learned weights:

A graph of blue bars

Description automatically generated with medium confidence

Recent Form and Average Goals Scored were the most influential features. Venue type and average goals conceded also contributed significantly.

**Conclusion**

This project successfully demonstrates the application of a feedforward neural network to predict football match outcomes using historical and derived team-based features.

**Key Findings:**

* Feature Engineering: Incorporating team-based statistics, such as rolling averages, win rates, and recent form scores, significantly enhanced model performance.
* Cross-Validation: A stratified K-Fold approach ensured robust evaluation, yielding a best accuracy of 51.02%.
* Feature Importance: Recent form and goals scored emerged as the most critical factors in determining match outcomes.

**Limitations:**

* The accuracy is constrained by the stochastic nature of football matches.
* Class imbalance (fewer losses) impacted the model’s ability to generalize.

**Future Work:**

* Incorporate external data such as player statistics, team rankings, and weather conditions.
* Experiment with ensemble methods like Random Forests or XGBoost to combine predictions.
* Address class imbalance using oversampling techniques or weighted loss functions.

This project highlights the complexities of sports analytics and demonstrates a foundation for further exploration of machine learning in predicting match outcomes.