ML2 - Semestral Project Assignment

[redacted]

September 28, 2024

1 Introduction

In the realm of sports, football is the most popular sport in the world. The game is played at a professional level all over the world and millions of people regularly go to football stadium to follow their favourite team, while billions more watch the game on television. A lot of people bet on football matches, hoping to win some money. **In this project** we will try to predict selected statistics of a football match using machine learning. We will use the data from the past seasons to train our model and then we will use the model to predict the statistics in the next season. We will use two different approaches to predict the statistics.

The goal of this project is to build a model that will be able to predict selected statistics of football matches with high accuracy that could potentially be used to help people make better (betting) decisions.

2 Data and Resources

You are provided with a dataset containing 23 seasons of data from 21 top european football leagues from 11 countries. The data contains match statistics, results, betting odds and other information. The data is provided in the form of `csv` files, one for each season. Each file contains information about all matches played in the given season in the given league.

The data is split into two parts: **train** and **test**. The training data contains all seasons from 2000/01 up to and including season 2022/23. The test data contains only the season 2023/24. This is an example of the structure of the data (similarly for the test set):

```
train
├── england
│   ├── 0
│   │   ├── 0001.csv
│   │   ├── ...
│   │   └── 2223.csv
│   ├── 1
│   │   ├── 0001.csv
│   │   ├── ...
│   │   └── 2223.csv
│   └── ...
└── ...
```

Data files are stored in the path `(train/test)/{country}/{league}/{season}.csv`, where `league` is a number (the lower the number, the higher the league) and `season` is a string representing the season in the format `{start_year}{end_year}`. For example, the file `train/england/0/2122.csv` contains data from the season 2021/22 from the highest English league - Premier League.

In the directory `data_description`, you can find the file `notes.txt`, which contains a description of the data including the meaning of each column, as well as an example of the data in the file `example.csv`.

The data is available at [redacted]

3 Assignment Tasks

3.1 Task 0: Data preparation

The initial task is to prepare the data for the next tasks. This includes all necessary steps to prepare the data for training a model of your choice. You can use any model you want. You can use the `notes.txt` and `example.csv` files in the `data_description` folder to get an overview of the data.

Feel free to find and use any external data like weather data or news articles. You can also create new features based on existing ones.

Be aware that the data is not perfect. It contains missing values, outliers, and other inconsistencies. In fact, the data is manually corrupted to simulate problems we discussed in the preprocessing lecture. You have to decide how to handle these problems. You can use any preprocessing technique you want. The only requirement is that you have to explain your decisions and the preprocessing steps you took. But keep in mind that you have to train your model on the data prepared in this task.

This task serves as a checkpoint for you. You have to create well–documented code with your solution. This task is not graded but you have to include the code in the submission together with a solution of the next tasks.

3.2 Task 1: Classification task

The first task is to train a model to **predict over/under 2.5 goals scored in a match**. Possible outcomes are Over 2.5 goals scored (Class 1), Under 2.5 goals scored (Class 0). Class 1 includes matches with 3 and more goals scored in a match (sum of goals scored by both teams), class 0 includes matches with 0, 1 or 2 goals scored. You can use any classification model you want. You have to

- train your model on the data prepared in the initial task,
- explain your decisions and the steps you took to train your model,
- evaluate your model, explain how you evaluated your model and present the performance across the countries and leagues.

Bonus task: create a betting strategy based on the machine learning model you developed for predicting over/under 2.5 goals scored. Pick one of the betting providers for over/under 2.5 goals scored, distribute a budget of 10 000$ per league and estimate what would be the benefits or losses incurred by applying your strategy.

3.3 Task 2: Regression task

The second task is to train a model to **predict the number of shots on target in a match** (sum of shots on target by both teams). Possible outcomes are integers larger than or equal to zero. You can use any regression model you want. You have to

- train your model on the data prepared in the initial task,
- explain your decisions and the steps you took to train your model,
- evaluate your model, explain how you evaluated your model and present the performance across the countries and leagues.

4 Submission Guidelines

You have to submit your well–documented code with your solution as well as files with your predictions. You have to submit one `csv` file for each league. Each file must contain the same number of rows as the test data for the league. Each row must contain your prediction for the corresponding row in the test data. The file must not contain any header. The folder with the csv files must be named following the pattern: `<team_number>_task<task_number>`. The csv files must be named following the pattern: `<country_name>_<league_number>.csv`. The country names and league numbers are the same as in the data folder. Python scripts / Jupyter Notebook files does not have to follow any naming pattern. Example of the files you have to submit is described in the `data_description/submission_example` folder.

Please, follow the instructions precisely, your results will be processed automatically. Send your solutions to the following e–mail: [redacted]

There is a single deadline for online submission of both tasks:

- 27.10.2024

5 Evaluation and Grading

You can earn up to 30 points for your solution based two criteria:

1. 15 points: modeling process - steps, methods, techniques, reasoning and their correct use.
2. 15 points: performance of your models on the test data,

Note: semestral project is 30% of the final grade.

5.1 Modeling part of grading

The modeling process will be assessed based on the quality of the performed steps, selected methods, techniques, and reasoning applied throughout the project. Up to 15 points can be awarded for demonstrating a thorough understanding of the problem, correct application of machine learning algorithms, appropriate data preprocessing, model selection, hyperparameter tuning, and evaluation methods. Clear and logical reasoning behind each decision, along with proper use of validation techniques and error analysis, will be key in earning full points in this section.

Issues such as poorly justified decisions, incorrect use of methods or techniques, lack of clarity in reasoning, incomplete steps in the analysis, failure to properly validate models, neglecting essential steps in preprocessing or feature selection, or insufficient explanation of choices made during hyperparameter tuning and model evaluation could lead to a reduction in points.

5.2 Performance part of grading

Tasks 1 and 2 will be evaluated for performance separately. The performance of your model will be compared to the performance of other teams using the following scheme:

- teams with score in interval $(P_T - (P_T - P_M) \cdot 0.25, P_T]$ will get 7.5 points,
- teams with score in interval $(P_T - (P_T - P_M) \cdot 0.5, P_T - (P_T - P_M) \cdot 0.25]$ will get 6 points,
- teams with score in interval $(P_M, P_T - (P_T - P_M) \cdot 0.5]$ will get 4.5 points,
- teams with score in interval $(P_M - (P_T - P_M) \cdot 0.25, P_M]$ will get 3 points,
- teams with score in interval $(P_M - (P_T - P_M) \cdot 0.5, P_M - (P_T - P_M) \cdot 0.25]$ will get 1.5 points,

where P_T is the performance of the best team, P_M is the median performance of all teams. So the maximum that a team can obtain is 15. That is 7.5 points for Task 1 and 7.5 points for Task 2. The performance of a team is determined by **F1–score** for the classification task and **MSE** for the regression task. The scheme is designed to reward teams that perform better than the median. Only the teams with very bad performance will end up without any points.

5.3 Bonus task

Solving the bonus tasks could get your team another up to 5 points – if the computation and evaluation is both correct and extensive (detailed error analysis). This way you can possibly cross the 30–point base for the semestral project.