LAHORE UNIVERSITY OF MANAGEMENT SCIENCES
Department of Electrical Engineering

EE514/CS535 Machine Learning
Spring 2021

Project Description

Total Marks: 40

Contribution to Final Assessment: 10%

1 Project Description

1.1 Objectives

The objective of the project is to apply the different classification algorithms covered in the course to the problem of predicting the results of the English Premier League (EPL). This will help your retention of the material and significantly enhance the depth of your understanding. This will also develop your skills in reporting the performance of different machine learning algorithms. The project is expected to consume roughly two weeks of moderately concentrated effort. We encourage you to work in a group of (maximum) three students (every student in the group will receive same score).

1.2 Data Set Description

The data-set provided here comprises of two main parts: match statistics and final standings. All data set files are in CSV format.

1.2.1 Match Statistics

The match statistics data is organized into separate files for each year, and within each file, the rows are sorted according to match date. There are a total of 20 .csv files, one for each year from 2000-2019. The feature columns are explained in the text file provided, as shown below:

```
Attendance = Crowd Attendance
Referee = Match Referee
HS = Home Team Shots
AS = Away Team Shots
HST = Home Team Shots on Target
AST = Away Team Shots on Target
.
.
.
```

Be careful, feature columns are not uniform across all years! You will have to process them accordingly.

1.2.2 Final Standings

We also have all the final team standings for each year consolidated into a single file where each row represents a team. In this part, each attribute is simply the year of the standing. There are 43 teams and their finishing position for each year from 2000 till 2016.

1.3 Scope of Work

For the given data-set, we want to develop classifiers for the prediction the outcome of a match between two select teams, given their attributes. You will need to develop your own test and validation data accordingly.

The scope of the project includes

- formulation of the problem under consideration.
- cleaning and pre-processing the data.
- apply feature engineering (if needed).
- implement the following classifiers kNN, logistic regression, Bayes classifier, SVM, neural network for the problem under consideration. You are allowed to use Scikit-learn implementations of the algorithms.
- report the performance of different classifiers and presentation of analysis/findings.

2 Expectations

There are three components of assessment in the project:

- report (15 marks)
- code (15 marks)
- 3 minute video presentation summarizing your work (10 marks)

Final report must be prepared using the template provided (sample format will be uploaded on LMS). Your report is expected to have the following sections: 1) Abstract (executive summary), 2) Introduction, 3) Mathematical Formulation, 4) Data-preprocessing (extraction and cleaning), 5) Feature Engineering (e.g., dimensionality reduction), 6) Use of the following classification algorithms; subsections on kNN, logistic regression, Bayes classifier, SVM, neural network, 7) Performance Evaluation (plots, tables etc.) and 8) Conclusions.

Note that: We encourage Masters and PhD students to use the LaTeX template for their report.

3 Timeline of Deliverables

We want you to adhere to the following time-lines.

- Week 10, 26th March, Friday 23:55: Form your group.
  The spreadsheet for groups can be found here.
- Week 11, 2nd April, Friday 23:55 pm: Submit preliminary project report with the following sections populated. You can obviously change the content in these sections in your final submission.
  - Abstract, Introduction and Mathematical Formulation
- Week 12: 9th April, Friday 23:55 pm: Submit mid-term report and code with the following tasks completed and added in the report
  - Data-preprocessing (extraction and cleaning)
  - Feature Engineering (e.g., dimensionality reduction)
  - Implementation of at least two of the following algorithms: subsections on kNN, logistic regression, Bayes classifier, SVM, neural network
- Week 14: 23rd April, Friday 23:55 Submit code, final report and video (No extensions will be given)

Dataset Explanation

For a start, the dataset is divided into two parts:

a) Match-wise stats of nearly for each season in a CSV file with the name of that season. For example, the stats of all the matches in the 2017-2018 season would be in the file with that name.
b) A CSV file containing the standings of teams across nearly 20 years of the English Premier League.

Let us look at the standings file first. It is a fairly simple file with each row representing a team, and each column representing a season, with the number corresponding to where they finished in the league from 1-20. If a cell is empty, it means that the team did not participate in the English Premier League that season. For the purposes of a match predictor, a team that consistently finishes above the other team is **more likely** to win their match. This is not a surety, but if you can find a way to use it as a feature for your classifier(s), it can potentially be useful.

The match-wise stats file is where you will get most of your data/features from. Keep in mind that you are **not** allowed to use any of the stats from the match you are trying to predict. You can only use the betting odds of that particular match. You can find the full form of the abbreviations in the file "ML EPL Dataset-Explanation" but let us go through some of the data you have at your disposal. Every match has specified which team is the home team and which team is the away team. In football, there is a "myth" that home teams generally have an advantage over the away team, so that piece of information **might** be useful for your classifier(s). For each win, a team gets 3 points, for each loss 0 points, and for each draw 1 point. Teams that have a significant point advantage over the other teams are more likely to win. In the dataset, you are not given the points of a team at a certain point in time, but you can very easily calculate them if you have the results of all the matches of the season till that point in time. Furthermore, attributes such as a high number of goals scored, and a low number of goals conceded before the match we are trying to predict might have a big impact on the outcome of the match. Another thing that can potentially impact a team's chances of winning is the form they are in. Think of it like this: if a team has won their last 5 matches, and they are facing a team that has lost the last 3 and drawn 2, which team is more likely to win? The data does not explicitly have the form of a team, but you can very easily figure that out.

Here is a mind map of how you can potentially tackle the problem at hand:

- **Engineer the features:** The data is fairly raw; you will need to do a lot of work to clean it and get meaningful features out of it. Do a bit of research regarding what can contribute to a team having an advantage over the other, and how you can generate that information from the provided data.

- **Train your model:** In layman's terms, assuming the aforementioned features are used, you are essentially telling your model that if a team is on a winning streak, having a significant advantage over the other team in terms of points, and has scored a high number of goals in the past *x* matches, it is more likely to win. Choose a window of matches and use their data to train your model. Keep in mind that if you train your model to check the form of a team over the past 5 matches, you will have to do the same in your testing phase. Consistency in the train and test data is the key.

- **Test your model:** As mentioned earlier, the test data should be in exactly the same shape as the train data. One way of approaching this could be to try and run your classifier for each season separately, since the data across seasons can be inconsistent.

We understand that this must be a lot of information to process, but if you can divide your project into the three components, it can be very easily tackled. If you have any queries, please feel free to reach out. Best of luck!