PREDICTING OUTCOMES OF FOOTBALL MATCHES USING BETTING ODDS AND DECISION TREES

1. INTRODUCTION

Football (soccer) is a popular sport and the outcomes of football matches are often bet on by fans of the sport. Bookmakers give odds to different outcomes based on how probable they believe each outcome to be. If the outcome of a football match could be predicted more accurately than the bookmakers, it would yield a significant advantage in sports betting and possibly other applications.

This report is organized as follows. Section 2 formulates our application as a Machine Learning problem. Next, section 3 describes the methods used in analyzing and implementing the problem and section 4 presents our results. Finally, section 5 presents concluding remarks.

2. PROBLEM FORMULATION

Our application can be modelled as a Machine Learning problem that is defined as follows. The data points of the problem represent football matches. Each match has the betting odds of the three possible outcomes (home team win, draw, away team win) as its features. The label of the data point is the actual outcome of the match.

3. METHOD

The data points for this problem were gathered from a data set in Kaggle that contains football match data from Europe's top 5 football leagues (English Premier League, German Bundesliga, Spanish La Liga, Italian Serie A, French Ligue 1). The data is from six different seasons (2011/2012 to 2016/2017) and is available from the webpage
`https://www.kaggle.com/prathamsharma123/comprehensive-football-dataset`.

We modified the data so that instead of team scores it contains the outcome of the match (1 for home win, 0 for draw, -1 for away win). We also got rid of the excess data not used in our application. Then, we created two datasets called ”Dataset1” and ”Dataset2”. ”Dataset1” contains 9135 data points from season 2011/2012 to season 2015/2016 and it will be used for training, validation and model selection. ”Dataset2” contains 977 data points from the season 2016/2017. It will be used as a test set to measure the performace of the selected model.

Since this is a classification task, we decided to compare Decision trees with different maximum depths. They are suited for classification tasks containing more than two discrete outcomes. Decision trees that have different maximum depths use different hypothesis spaces. We used eight different models which are defined by decision trees of maximum depths $d = 2, 3, ..., 9$.

Decision trees are described in Chapter 3 of [1]. The hypothesis represented by a decision tree is piecewise-constant over different areas of the feature space. We used the squared error loss as defined in Chapter 2.3 of [1] to assess the quality of the decision trees, since this allowed us to make use of a library. The squared error loss takes the difference between the predicted label and the true label and squares it.

We split "Dataset1" into a training set and a validation set with a single random split, so that the training set contained 67% of the data points. Then we computed the training and validation error on these sets for decision trees with different maximum depths. Next, we chose one of the models based on the errors and tested it on "Dataset2". We used the classification accuracy as the test metric instead of the squared error loss, since it can be better compared with other benchmarks. All of our computation was performed using Scikit-learn [2].

4. RESULTS

Table 1 shows training and validation errors for different maximum decision tree depths. We chose the maximum depth $d = 7$, since it gives the smallest validation error and is close to the training error. For depths larger than $d = 7$, the method seems to overfit the training data, since the validation error starts to grow.

	d = 2	d = 3	d = 4	d = 5	d = 6	d = 7	d = 8	d = 9
Training	1.105	1.096	1.082	1.067	1.019	1.000	0.989	0.955
Validation	1.133	1.134	1.134	1.128	1.077	1.073	1.107	1.113

Table 1. Training and validation errors for different maximum decision tree depths d.

We tested the resulting decision tree with maximum depth $d = 7$ on "Dataset2", since we wanted to evaluate its performance. For "Dataset2", the average squared error loss is 1.014, which is lower than the validation error. This seems to suggest that predicting the outcomes of football matches is possible using this method.

To be able to compare our method with other benchmarks, we computed the classification accuracy for our chosen model. The accuracy result was 55% for "Dataset2". This is close to the accuracy usually achieved by bookmakers, which is 53% [3].

Achieving a similar result to bookmakers was expected, since our predictions relied only on betting odds provided by them. This accuracy would probably not be enough to gain an advantage in sports betting, but our methods could be improved by considering additional features. These could be for example the previous performance of the teams and their players, as well as betting odds provided by more than one bookmaker. Statistics of a match in progress could also be used to predict the outcome of that match.

5. CONCLUSION

We compared eight different ML models regarding predicting the outcomes of football matches based on betting odds. The models were gathered from decision trees with eight different maximum depths. We split ”Dataset1” into a training set containing 67% of the data points and a validation set containing 33% of the data points. We chose the model where the decision tree had a maximum depth of 7, which yielded the lowest validation error 1.073 on the validation set. The respective training error on the training set was 1.0. The method seemed to not overfit with maximum depth 7 but larger depths would have resulted in overfitting.

The decision tree with maximum depth 7 was tested on ”Dataset2” resulting in the test error 1.014. This is lower than the validation error, suggesting that the method is not overfitting with maximum depth 7. The prediction accuracy on ”Dataset2” was 55%, which is similar to the results of bookmakers.

It seems that learning a hypothesis from betting odds could be useful in predicting outcomes of football matches and gaining an advantage in sports betting, even though our method does not achieve this at a higher accuracy than the bookmakers. The method could be improved using suggestions presented in section 4. Which improvements would be useful is a potential avenue for future work. Other works have achieved accuracies of up to 70% [4].

6. REFERENCES

[1] A. Jung, “Machine Learning: The Basics,” 2021, mlbook.cs.aalto.fi.

[2] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[3] H. Mathien, “European Soccer Database,” https://www.kaggle.com/hugomathien/soccer, [Referenced 28.3.2021].

[4] K. Odachowski and J. Grekow, “Using Bookmaker Odds to Predict the Final Result of Football Matches,” in *Knowledge Engineering, Machine Learning and Lattice Computing with Applications*, M. Graña, C. Toro, R. J. Howlett, and L. C. Jain, Eds., Berlin, 2013, pp. 196–205, Springer, doi: 10.1007/978-3-642-37343-5_20.