FOOTBALL PREDICTIONS

https://github.com/aczapka/football-predictions

Project Definition

Project Overview

Betting in football matches has a long tradition in many countries. In this project we will focus on one of the easiest forms of betting: trying to guess the full time result of the match. The one who manages to systematically make better guesses than the bookmaker, could earn money betting.

Problem Statement

In football betting, bookmakers need to predict to some extent the results in order to set correct initial odds before the match. This prediction of course is not perfect. Maybe analyzing its weaknesses it is possible to find a betting strategy that will be more profitable than random guessing.

For this strategy to be profitable, it is not necessary to be globally better than the bookmarker. It would be enough to simply do better in some particular aspect and take advantage of that particular bookmaker weakness.

Metrics

The most realistic metric in order to evaluate the performance of the algorithm is to perform a walk-forward analysis with historical data and estimate how much profit would be made using the algorithm predictions for a certain period of time.

Analysis

Data Exploration

There are many datasets available online. The one chosen for this project is https://www.football-data.co.uk data from Spanish First Division (" Liga de Primera División" a.k.a. "La Liga").

The initial dataset downloaded includes all matches from the last 22 years: from the 2000-2001 season to the 2021-2022 season. Typically, one season is composed of 380 matches (38 weeks with 10 matches a week). However, this is not always true so our dataset includes 8248 rows instead of the initially expected 380 * 22 = 8360.

For each match, the data most interesting data for this project is:

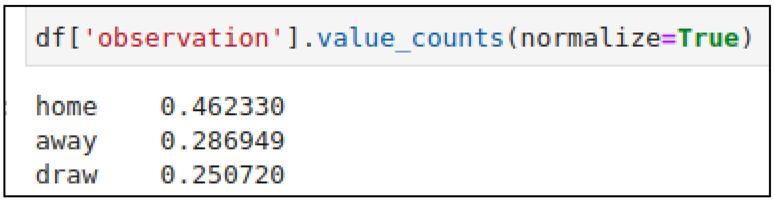
1. Date.
2. Home and away teams.
3. Full time home and away goals.
4. Home, draw and away initial odds of the bookmakers.

There are odds data of many bookmakers. It is interesting that they do not differ much. So it is better to focus on only one bookmaker with long historical data. The bookmaker “Bet365'' was chosen. Its historical odds data covers almost all our dataset.

It is important to keep in mind that bookmaker odds are linked to a particular time. In this case, the dataset notes claim that “betting odds for weekend games are collected Friday afternoons, and on Tuesday afternoons for midweek games“.

Data Visualization

The first fact to consider is that almost half of the matches end up with a local victory. Around a quarter of matches end in a draw and another quarter in away victory.



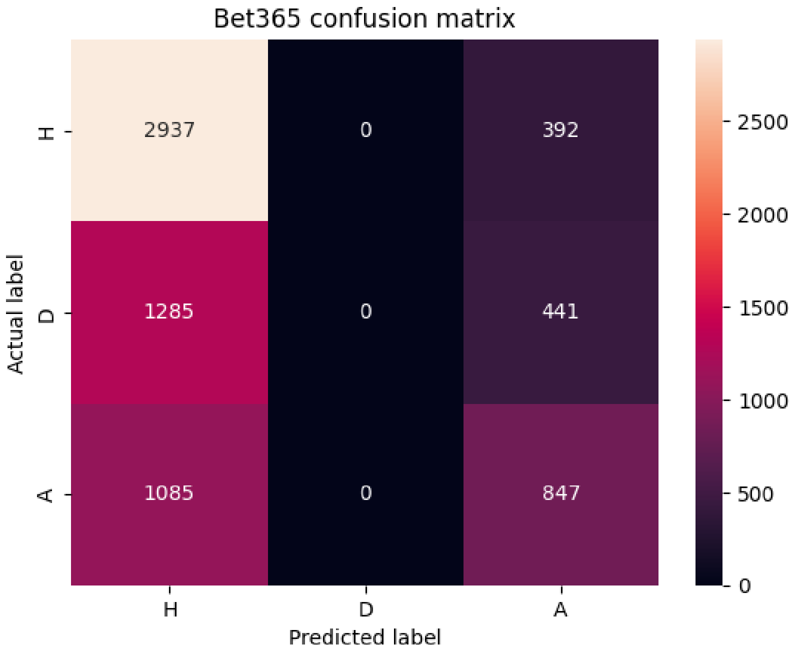
```
df['observation'].value_counts(normalize=True)

home    0.462330
away    0.286949
draw    0.250720
```

This draws two interesting conclusions:

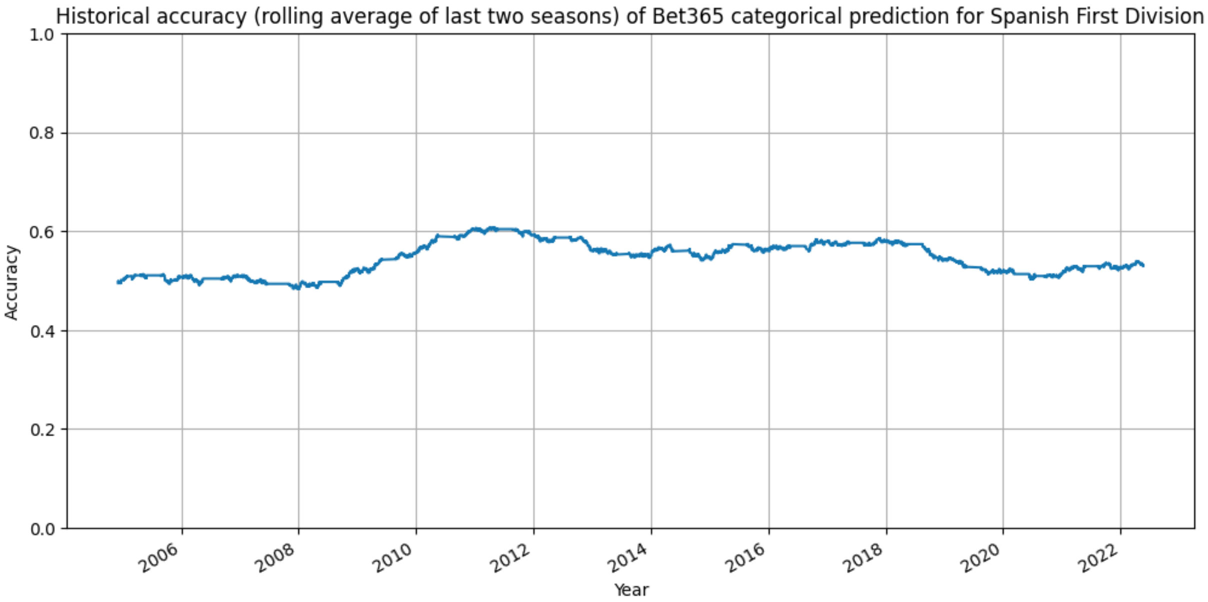
- It is a fact that it is not the same playing at home than away.
- The draw result is fairly common.

Now let’s check the bookmaker predictions, keeping in mind that the inverse of the odds are their probabilistic multi-class prediction. The confusion matrix for all the historical data is the following:



It is very surprising that although there are several thousands of matches and knowing that a lot of them end up in a draw, the bookmaker never predicts a draw.

That being said, their average global accuracy is pretty stable and moves around 0.5 - 0.6:



Methodology

Idea

The idea behind this project is to build a neural network meta-model which will be feed with the following data:

1. Bookmaker odds found on the internet (which are the output of the bookmaker models).
2. An alternative to the bookmakers odds calculated independently with a Poisson bivariate regression model (code in this project).
3. Additional data which may help the predictions.

Data Preprocessing

The first step is to download the historical data mentioned in the first section. After selecting only one of the bookmakers (“Bet365”) because all of them are very similar, this data is joined with the output of our Poisson regression model. After that, data is enriched with some additional features:

- Distance between team homes..
- Season progress.

More additional features could be tried in the future development of this project.

Core model

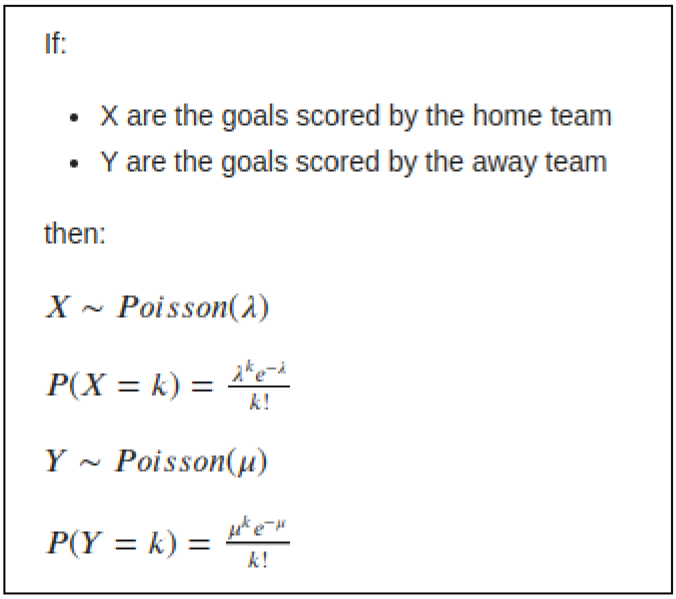
Implementation of the Poisson regression model

This is the core of this project. A big part of the theoretical background is inspired in following paper:

Dixon, M. J., & Coles, S. G. (1997). Modelling Association Football Scores and Inefficiencies in the Football Betting Market. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, *46*(2), 265–280. http://www.jstor.org/stable/2986290

It is a fact that most bookmakers have historically used the Poisson regression technique in order to get an initial probabilistic forecast of the results of a match. The betting odds are just the inverse of these predicted probabilities, a little bit adjusted so the bookmaker will have some profit margin.

The code in this project implements a particular version of such a model. In short, the idea is to consider that a goal in a match scored by a team is just a “particular event” whose occurrence distribution over time follows a Poisson distribution. A key idea of the paper is that this distribution is independent of the rival team. This simplification assumes that, during a match, a team's performance is not directly affected by the goals scored against them, which of course is not exactly true.



If:

- X are the goals scored by the home team
- Y are the goals scored by the away team

then:

$$X \sim Poisson(\lambda)$$

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

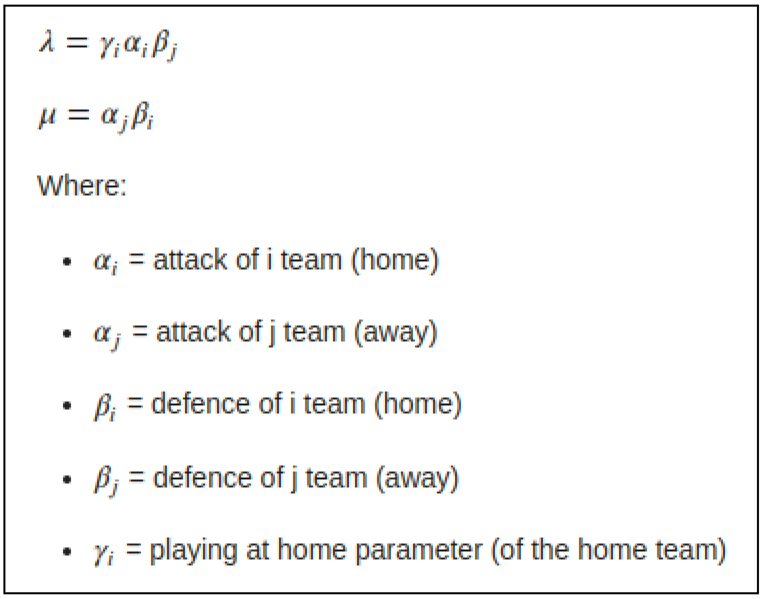
$$Y \sim Poisson(\mu)$$

$$P(Y = k) = \frac{\mu^k e^{-\mu}}{k!}$$

The next step is to consider which parameters define the particular Poisson distribution of each team, which of course may vary over time. At first, the selected parameters are

- Home team attack and defense parameters
- Away team attack and defense parameters.
- Home effect.

The attack parameter of a team increases the probability of scoring a goal against any other team. This parameter is multiplied by the rival team's defense parameter, which reduces the probability of scoring a goal. In the case of the home team, an additional multiplicative parameter quantifies the advantage of playing at home, normally increasing the probability of scoring a goal.



$$\lambda = \gamma_i \alpha_i \beta_j$$

$$\mu = \alpha_j \beta_i$$

Where:

- α_i = attack of i team (home)
- α_j = attack of j team (away)
- β_i = defence of i team (home)
- β_j = defence of j team (away)
- γ_i = playing at home parameter (of the home team)

The home attack, away defense and home effect parameters determine the Poisson distribution of the goals scored by the home team. The home defense and the away attack parameters determine the Poisson distribution of the goals scored by the away team. If we assume the mentioned independence of these two distributions, then all the possible combinations of these two distributions generate a probability matrix with all the possible results of the match.

$$\boxed{P(X = x, Y = y) = \frac{\lambda^x e^{-\lambda}}{x!} \cdot \frac{\mu^y e^{-\mu}}{y!}}$$

The parameters of this model are optimized using the maximum likelihood technique. Assuming independence between games, the likelihood of the outcomes of many of them would be the product.

$$\boxed{L(X = x, Y = y) = \prod_{i=1}^{n} \frac{\lambda^x e^{-\lambda}}{x!} \cdot \frac{\mu^y e^{-\mu}}{y!}}$$

For a consistent numerical implementation, actually what the algorithm really does is minimizing the minus logarithm of likelihood.

$$\boxed{l(X = x, Y = y) = \sum_{i=1}^{n} \log(\frac{\lambda^x e^{-\lambda}}{x!}) + \log(\frac{\mu^y e^{-\mu}}{y!})}$$

Refinement

As the problem can be considered as a time series prediction, the algorithm is trained on past data in order to evaluate against supposedly future data. The current implementation uses the data from all the last season to train and predict the whole following season.

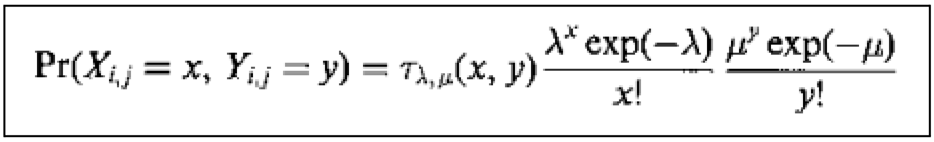
However, this should be a hyperparameter of the model and in the future implementation it is worth considering optimizing this behavior.

Another important refinement explained in the Dixon & Coles cited paper is that the assumption of independence of Poisson distributions is not true. The teams dynamically adapts and changes their playing style during a match and this results in a higher probability of a draw than the model predicts.

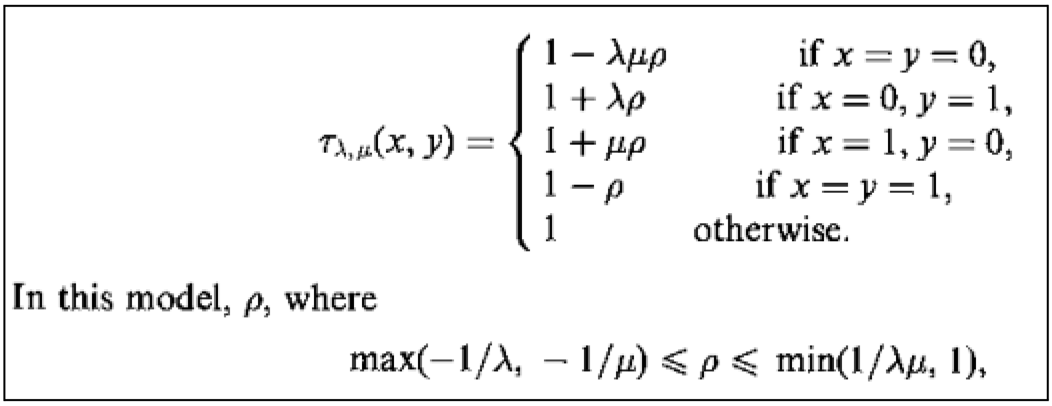
The bookmakers do not consider this and as result they almost never predict a draw as the most probable result. If our Poisson regression of this project could take this imperfection into account, it could be the advantage needed to beat the bookmakers.

In order to try to achieve this, an additional parameter is added to the model: the dependence. This parameter multiplies the product of the two Poisson distributions . It is a global variable for all teams and it is used to correct the probability matrix trying to “thicken the diagonal”. Which means that it increases the probability of a draw to compensate for the model’s bias.

Formulas shown on the cited Dixon, M. J., & Coles, S. G. paper:



$$\Pr(X_{i,j}=x,\ Y_{i,j}=y)=\tau_{\lambda,\mu}(x,y)\frac{\lambda^{x}\exp(-\lambda)}{x!}\frac{\mu^{y}\exp(-\mu)}{y!}$$



$$\tau_{\lambda,\mu}(x,y)=\begin{cases}1-\lambda\mu\rho & \text{if } x=y=0,\\ 1+\lambda\rho & \text{if } x=0,\ y=1,\\ 1+\mu\rho & \text{if } x=1,\ y=0,\\ 1-\rho & \text{if } x=y=1,\\ 1 & \text{otherwise.}\end{cases}$$

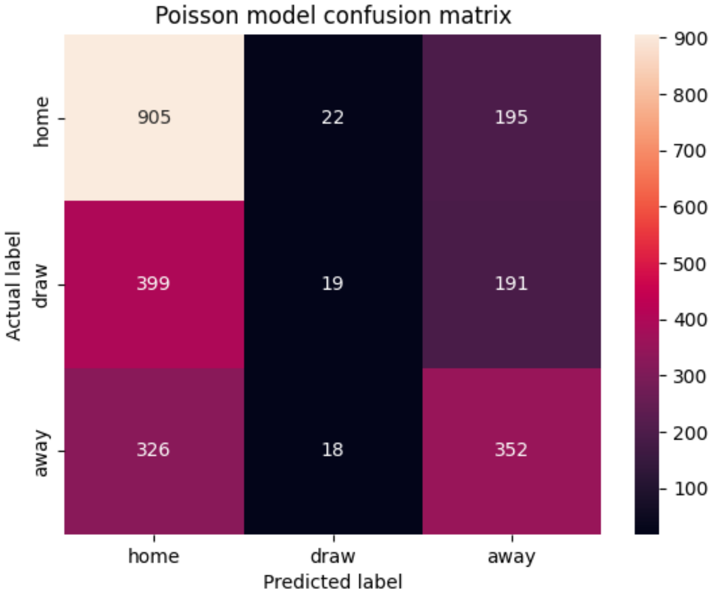
In this model, ρ, where

$$\max(-1/\lambda,\ -1/\mu)\leqslant\rho\leqslant\min(1/\lambda\mu,\ 1),$$

As the authors of the paper explain, the resulting marginal distributions remain Poisson.

After training our bivariate Poisson model taking into account this improvement, the resulting confusion matrix:

- Is similar to the one of the bookmaker model, which suggests that their approach is not very different.
- Has some more tendency to predict draws.



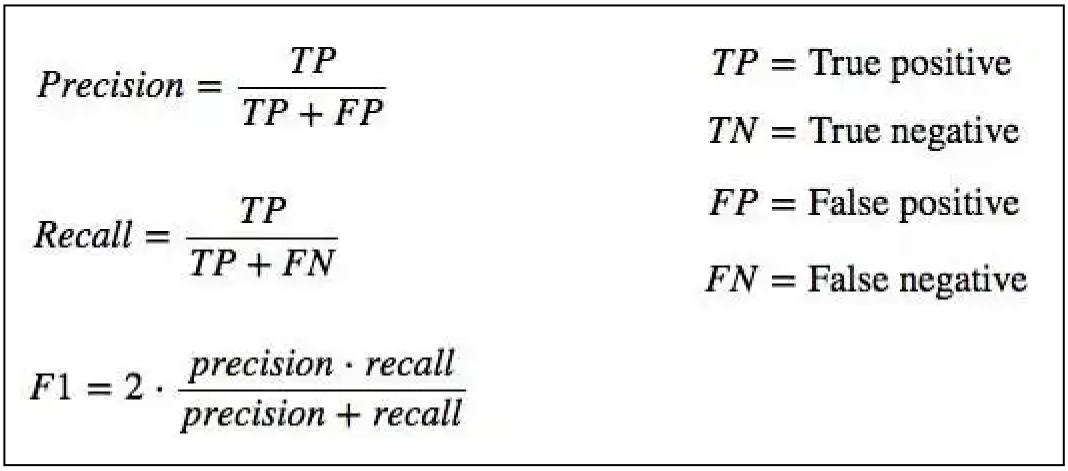
Metamodel

The metamodel is just a conventional neural network which takes the output of the bookmaker model, the output of the Poisson model of this project and some additional data and tries to combine it all in the best possible way.

The final strategy is a simple binary classification: trying to predict whether a match will end up as a draw or not. If a good classification system could be trained, the bookmakers odds would be beaten because they almost never predict a draw (so they pay a big reward for correctly guessing an actual draw).

Train - test split of 0.8 - 0.2 has been used. The dataset is imbalanced because only around one quarter of the matches end up in a draw. In order to overcome this, weights are applied to data. Also dropout technique is used to prevent overfitting to train data.

The binary classification threshold prioritizes precision metric over accuracy. This means that when the model predicts a draw, there is a really high probability of an actual draw. On the other hand, the bookmaker will not predict such a high probability of a draw because of their biased model. Thus, it would be a good idea to place a bet on a draw expecting that (on average) the reward will be bigger than the risk.



$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

TP = True positive

TN = True negative

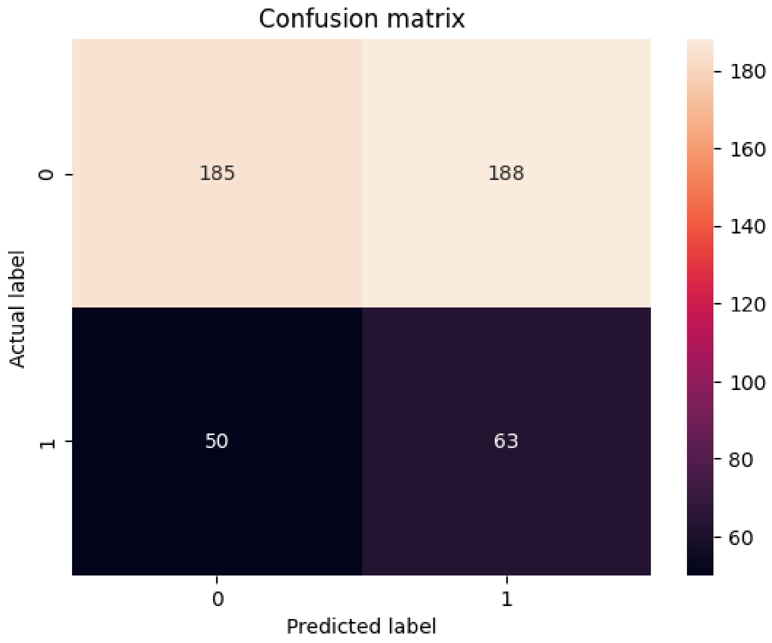
FP = False positive

FN = False negative

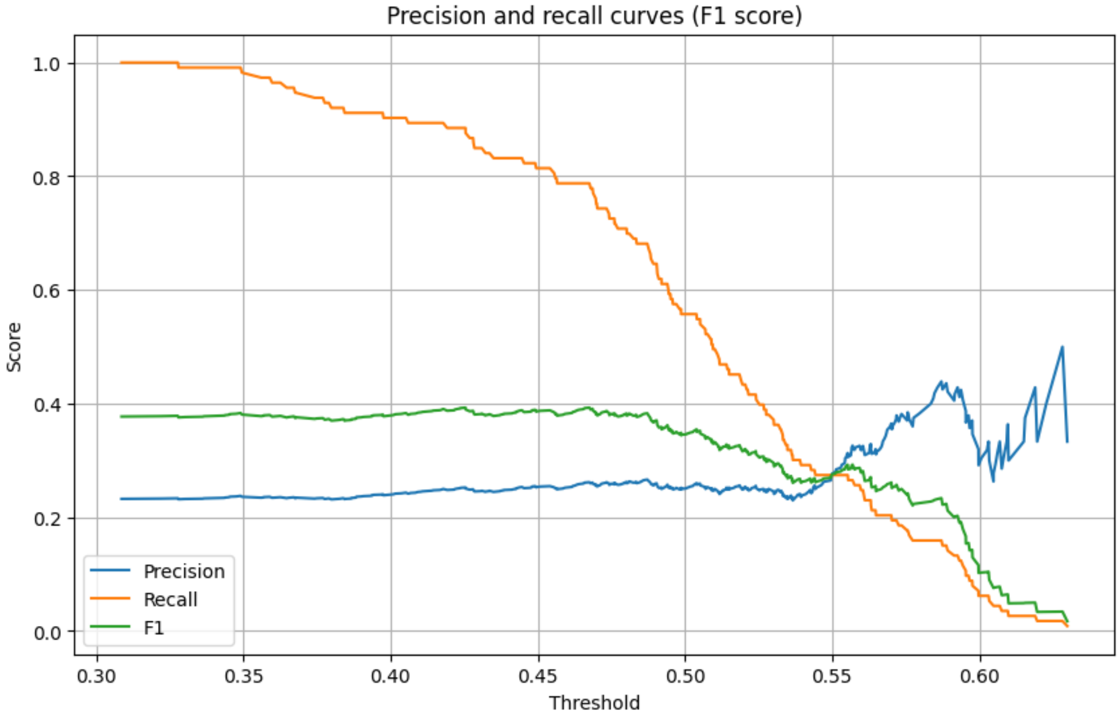
Results

Model Evaluation and Validation

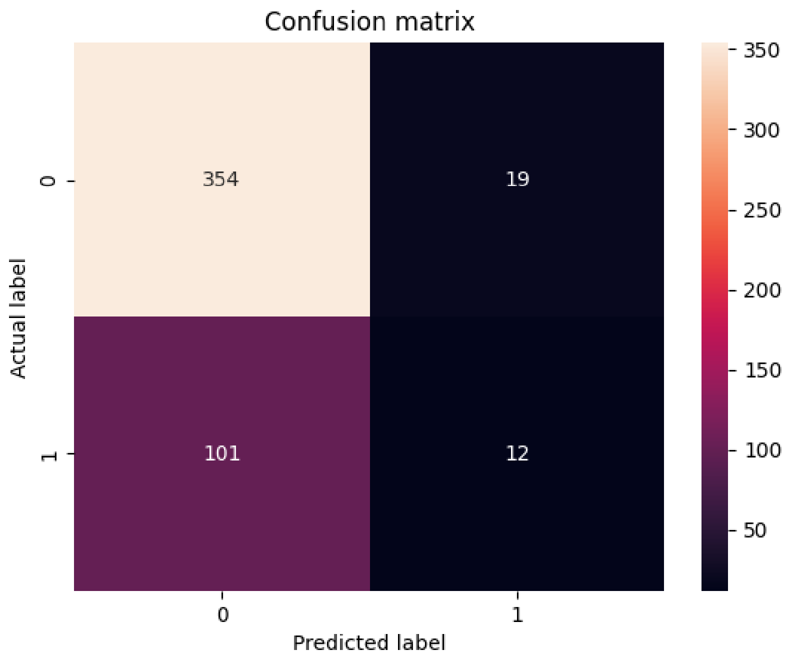
Just after the training and with a default decision threshold of 0.5 the confusion matrix of the metamodel looks as follows:



This means an accuracy of around 25 %, which is not enough.
In order to improve this, the precision-recall curves has been calculated:



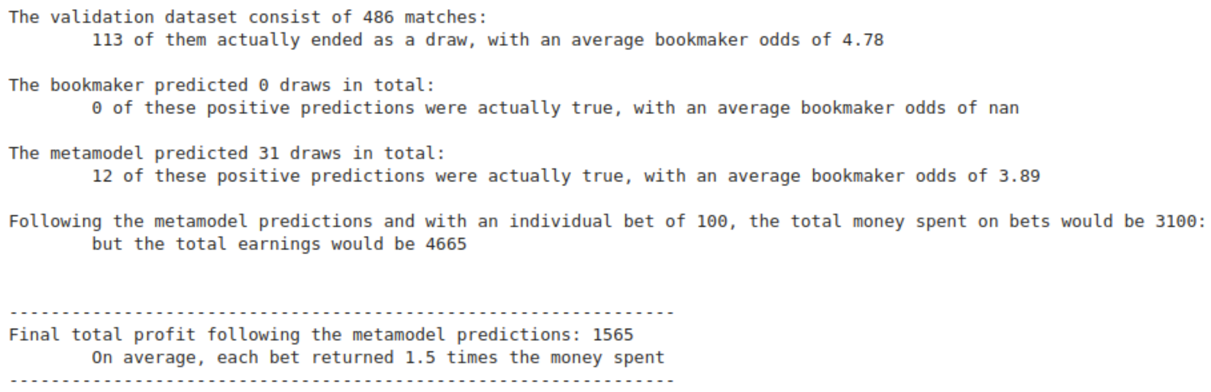
This shows that the precision metric would be better with a decision threshold of 0.595, which lead to this new confusion matrix:



Now the precision has been boosted to almost 40 %.

Justification

If this model would be used to bet on the validation dataset, the result would be:



```
The validation dataset consist of 486 matches:
        113 of them actually ended as a draw, with an average bookmaker odds of 4.78

The bookmaker predicted 0 draws in total:
        0 of these positive predictions were actually true, with an average bookmaker odds of nan

The metamodel predicted 31 draws in total:
        12 of these positive predictions were actually true, with an average bookmaker odds of 3.89

Following the metamodel predictions and with an individual bet of 100, the total money spent on bets would be 3100:
        but the total earnings would be 4665


----------------------------------------------------------------
Final total profit following the metamodel predictions: 1565
        On average, each bet returned 1.5 times the money spent
----------------------------------------------------------------
```

So the achieved precision may be enough to build a successful betting strategy, with an average return of x1.5 on each bet on a draw prediction.

Conclusion

This project reveals one clear flaw of the bookmakers' predictions: they under-estimate the probability of a draw. This weak point could be used to work out a model that specializes in predicting draws with high precision. With that forecast in hand, it would be possible to beat the bookmaker on average.

However, the model achieved in these projects struggles to deliver such a high precision consistently.

Improvements

Maybe the Poisson model of this project is too similar to the models used by bookmakers, not providing enough extra information. Further development could consider giving more importance to the “dependence” parameter of the Poisson model in order to correct more the draw bias.

Also, the data enrichment process could include more additional variables to build a more powerful neural network meta-model.

Some hyper-parameters, like the train and test period of the walk-forward analysis, could also be studied and probably improved.

Finally, this project is only focused on a very particular sport and championship. It may work better in other scenarios.

Reflection

Bookmaker odds are the output of a forecasting model which is far from perfectly accurate. However, it is good enough and balanced enough so it is not easy to systematically beat their categorical predictions. After all, bookmakers have been around for many years and have been and are profitable.

Author: Alberto Czapka, 2022
https://www.linkedin.com/in/alberto-czapka/