ORIE 4741 - Midterm Report

atp44, wmb87, sc2267

November 8th, 2020

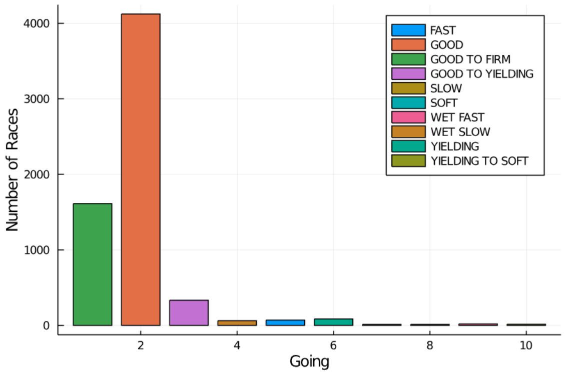
Digging into the Data Our project involves exploring a dataset to analyze how successfully we can create horse race betting suggestions. Our data consists of two .CSV files: one consisting of data about specific horse races, and the other one consisting of specific horse performances(runs) within those races. Across both datasets, there are 73 features which contain information about either a race or a specific horse's performance in a race. Every race consists of 14 horses, and every horse is ranked from first place to 14th place at the end of the race. Although the first place winner is not the only one that receives prize money, in order to be more accurate, we decided that we would only try to predict the probability that a certain horse would get first place. In our dataset, "Won" is the label that corresponds to whether or not a horse got first place. If the value is 1, it means the horse got first place and 0 otherwise.

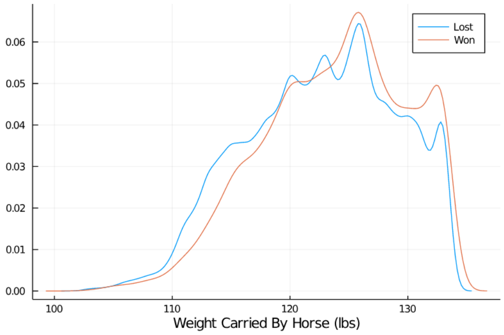
From a first scan, we identified a handful of features that could have a significant impact on our prediction variable, Won. These features include Going, which analyzes the ground condition of the track, Draw, which denotes the starting position of every horse, and actual weight, which is the total weight the horse carries. Other important features include Horse Type, Distance, and Horse ratings.

We first used some visualizations to better understand the data. For example, we created a histogram of horse types and found that an overwhelming majority of horses are Gelding, so this feature may not be very useful in predicting result outcomes for most horses.

The first plot is a histogram of the feature: "Going." This plot indicates the breakdown of the surface condition in every one of the races in our dataset. As shown, most of the race tracks have been in "good to firm" condition, meaning they are in solid condition for the race, while a sizable portion have also been in "yielding" condition, meaning they are starting to soften.

The second plot shows a very counter-intuitive insight into our data. This shows the distribution of winning horse and losing horses across the different 'actual weight', meaning the weight the horse is carrying during the race, i.e. jockey and gear. We hypothesized that the lighter the load a horse must carry during a race, the faster they run, however, this plot shows that is not the case. We can see if we condition on the horses that have won, the majority of them are carry around 125/126lbs. So it is less common that a horse, who won, has a lighter load during the race. This could be because the weight distribution of jockeys makes it less likely the horse has only 110lbs on their back during the race but it is interesting nonetheless.



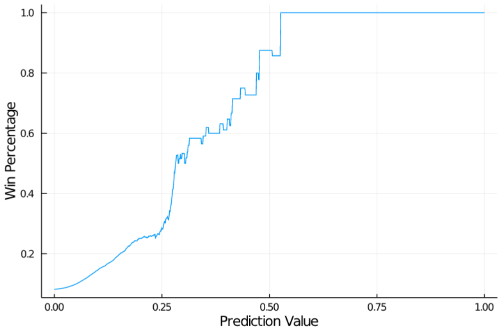


Some columns had missing values, such as the type of gear the horse was wearing. In that case, we decided that there was no logical way to fix this issue and removed this feature from our data matrix. For the total prize feature (award to horse owner), it had a low percentage of missing values, so we decided to replace those values with the column mean. From a more in-depth look at the datasets, there are a handful of features appear to be significant in . Some of these features include Going (Declared Weight, Actual Weight, Draw, Won, Race Data:

Cleaning It Up There were some significant challenges in getting this dataset into a 'clean' dataset. We identified 23 columns with at least one missing value. Fortunately, the majority of these columns were extraneous for our model's purposes and we ended up deleting all except 3 columns. We were able to drop the majority because most of them dealt with some sort of time factor during the horse race. However, we aim to predict the probability a certain horse wins the race, all before the race starts. Since the odds become static once the race begins there is no value in trying to manipulate the splits/section times of the horses during the race. This also helps our model reduce overfitting since we are deleting columns that we know cannot have any impact on our prediction. To handle the missing values in the prize column, which represents the total amount a winning horse owner and jockey will win if their horse takes first place, we inputted the mean of the column. We also had multiple categorical columns to preprocess. We used 'one-hot' encoding for each categorical, a total of 8, which added 436 binary columns. We chose 'one-hot' over 'many-hot' because we were not dealing with sets within our categorical columns. Our final step in cleaning our data was to change our 'date' column from a Julia Date object to a day-of-the-year. With this change we can account for the seasonality of races within a given year, which makes sense since there are 'seasons' for horse-racing.

Preliminary Models/Analysis We used a least squares regression model to predict the win value (0 or 1) of a given performance. The values we predicted are an "expectation of win" that are floats mostly on the interval [0,1], with a few outliers slightly less than zero or slightly greater than 1. Our mean square error is therefore the average squared difference between our prediction values and the win outcome of a performance (1 or 0. Our model has a train MSE of 0.0710 and a test MSE of 0.0722. Equivalently, our test predictions have an average difference of 0.27 from the win value of 0 or 1 for our test data. Given the difficulty of predicting horse race outcomes, we believe this is a reasonably good model, but plan to try new models to further improve our test error. These results show that our model does not overfit significantly and appears to generalize well.

To understand what our prediction value represents, we plotted the winning percentage of examples with a prediction value greater than or equal to values on the interval [0,1]. We can see that win percentage is positively correlated to our prediction value, so our prediction seems to be useful. We notice that the win percentage increases very steeply with our prediction value, reaching 100% at a prediction value around 0.55. At this point, there were very few examples (under 15), which explains why they all have the same result. This plot is shown below:



For our remaining work, we want to try new models (polynomial regression, random forest, etc.) to see if we can improve on our current model's error. We will also create a new model to predict the winning dividends of a race (which are constantly in fluctuation up to the actual start of a race). This prediction can then be used to find the expected payoff of making a specific bet, which will allow us to reach our overall goal of providing reliable betting suggestions.