CSCI 3320 Fundamentals of Machine Learning
Project Report

Preprocessing

Number of rows drop: 668

Number of horses: 2155

Number of jockeys: 105

Number of trainers: 93

Classification

All prediction results are shown on the folder predictions/*.csv .

Below is the prediction evaluation of different classification models:

Logistic Regression

Running time: 0.35s

	HorseWin	HorseRankTop3	HorseRankTop50Percent
Precision:	0.80	0.61	0.70
Recall:	0.02	0.34	0.74

Naive Bayes(MultinomialNB)

Naïve Bayes Classifier: MultinomialNB

Reason: Good for classification with discrete features

Running time: 0.29s

	HorseWin	HorseRankTop3	HorseRankTop50Percent
Precision:	0.14	0.36	0.63
Recall:	0.88	0.86	0.84

Support Vector Machine (SVM)

Kernel: rbf

Reason: Good for high dimensional projection and has less numerical difficulties

Running time: 95.18s

	HorseWin	HorseRankTop3	HorseRankTop50Percent
Precision:	0.50	0.64	0.72
Recall:	0.01	0.22	0.63

Random Forest

Running time: 3.55s

	HorseWin	HorseRankTop3	HorseRankTop50Percent
Precision:	0.43	0.56	0.69
Recall:	0.08	0.34	0.71

Characteristics of these four classifiers

Logistic Regression

Aims in finding the probability of success and fail events and used when dependent variable is binary (Either 0 or 1).

Naïve Bayes

The presence of a particular feature in a class is independent to any other features.

Support Vector Machine

Illustrate the extreme cases in different classes (Some points in class A are likely to be class B but somehow it belongs to class A)

Random Forest

Pick random points from training set repeatedly and train the model with a large amount of decision trees to form a forest.

Trade-off for choosing classification models

There are some qualities we may look for when choosing classification models:

(1) Speed and Simplicity

(2) Accuracy

(3) Size, number of features of dataset, data behavior (discrete / continuous values)

If we want to train a small amount of data, we may consider using a simple model which can still give us a good performance (Naïve Bayes). If we want to receive a higher accuracy, we may choose a more concrete and complex model as it involves complicated calculation and result a good precision. Similarly, we also take the size and number of features and the data behavior as the consideration of choosing a good model. For example, in this project, we may consider SVM and Random Forest would give us the highest accuracy since we have a large amount of data and features and they perform well when data size is big.

How do the cross validation techniques help in avoiding overfitting?

Cross validation split the training data into multiple train-test splits, which helps in tune our classification model. In general K-fold cross validation, we separate the train dataset into k subsets, then choose on fold for validation and the remaining for training. We iteratively train the model for k times. Finally, we pick the model which gives us the lowest error. This method can keep our test data unseen for selecting the final model.

Precision-Recall Metric

TPR (True positive rate) and TNR (True negative rate) are enough for us to evaluate the classification model. In the classification, we care about the Top_1, Top3, and Top50% in a horse racing, so we need to know how many correct classifications in those groups and how many correct positive labels within those groups.

Name: Ng Kam Fai (1155078625) Department of Computer Science and Engineering

Regression

Support Vector Regression Model

Kernel selected: rbf

Reason: Good for high dimensional projection and has less numerical difficulties

What roles do epsilon and C play in SVR

Epsilon: Margin of tolerance where no penalty is given to errors

C: Penalty of the error term

Value chosen for (1) Epsilon: 0.01 (2) C: 1

Reason: Choose smaller epsilon for smaller error admitted in the model, and C = 1 gives reasonable penalty when error occurs

Gradient Boosting Regression Tree Model (GBRT)

Loss function selected: ls

Reason: The most natural choice for regression. Calculated by the mean of the target.

What roles do learning rate, n_estimators and max_depth play in GBRT

learning rate: Scale the step length, the smaller the better test error

n_estimators: Number of trees in the forest, the larger the better sometimes

max_depth: Maximum depth of the tree

Value chosen for (1) learning rate: 0.1 (2) n_estimators: 100 (3) max_depth: 1

Reason:

Learning rate has a high relationship with n_estimtors, smaller learning rate require larger n_estimators. When tuning different values of them, I found that learning with 0.1 and n_estimators with 100 already gives us the highest accuracy of the model. Also, max_depth of 1 is good enough to fit the model.

Predicting on test data

Best result for SVR: (kernel=rbf, RMSE=2.16, Top_1=0.23, Top_3=0.55, Average_rank=3.96)

Best result for GBRT: (lost function=ls, RMSE=1.59, Top_1=0.57, Top_3=0.55, Average_rank=1.81)

root mean squared error of svr model: 2.16

root mean squared error of svr model after normalization: 0.087

root mean squared error of gbr model: 1.59

root mean squared error of gbr model after normalization: 0.085

Name: Ng Kam Fai (1155078625) Department of Computer Science and Engineering

Prediction Evaluation:

----Without normalization----

Support Vector Regression:

Top_1: 0.23, Top_3: 0.55, Average_rank: 3.96

Gradient Boosting Regressor:

Top_1: 0.57, Top_3: 0.55, Average_rank: 1.81

----With normalization----

Support Vector Regression:

Top_1: 0.28, Top_3: 0.60, Average_rank: 3.54

Gradient Boosting Regressor:

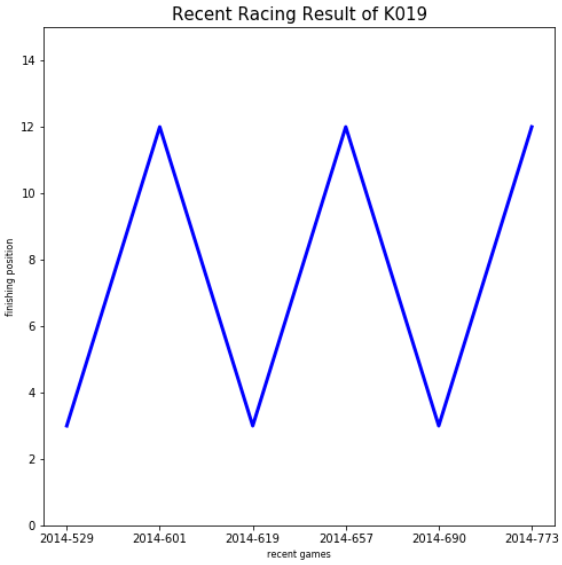
Top_1: 0.57, Top_3: 0.60, Average_rank: 1.81

Visualization

Line Chart of recent racing result

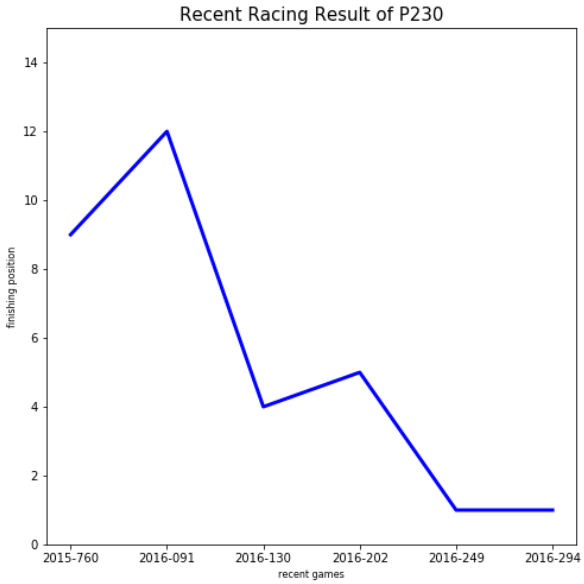
Now we choose two horse with ID K019 and P230 for demonstrating the result of line chart:

Horse 1: K019



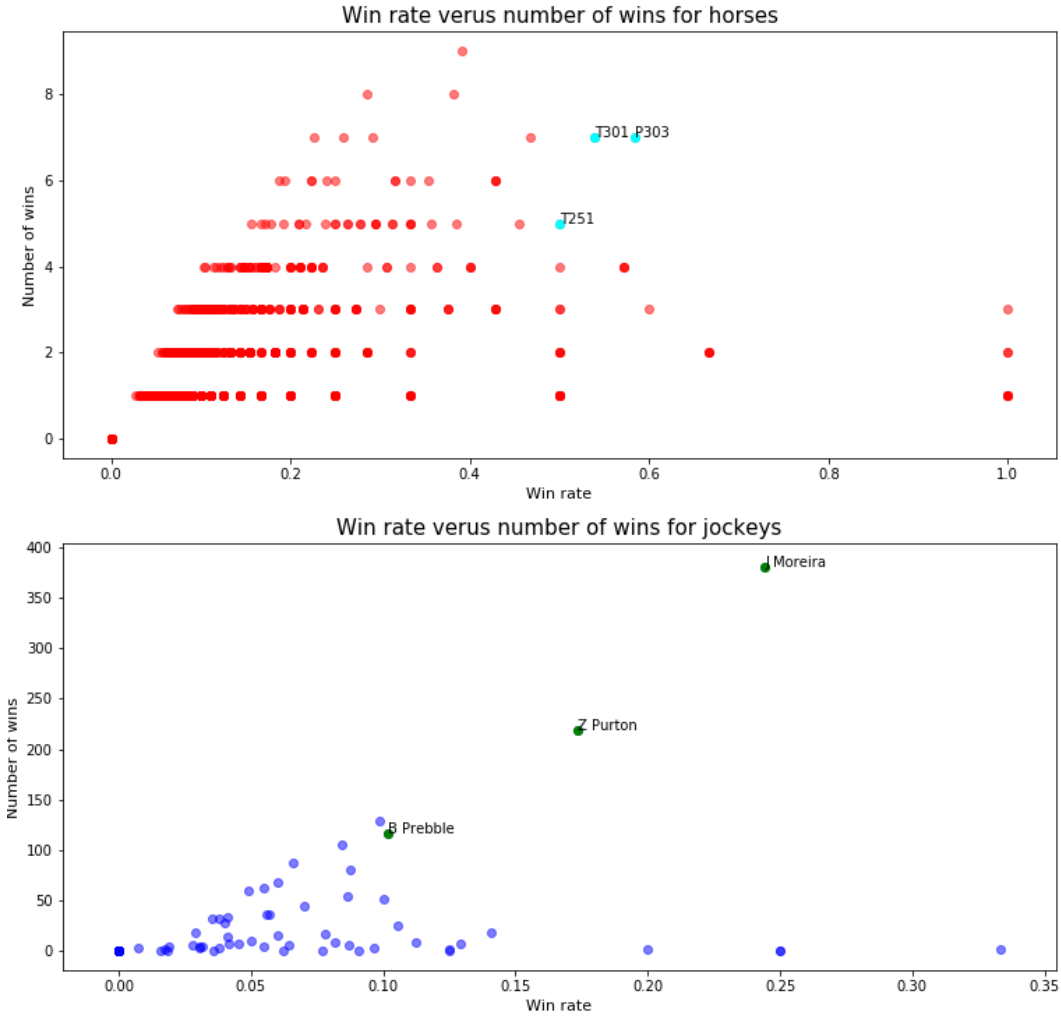
From the plot, we can see that the horse with ID K019 has unstable condition through out the recent games. Half of recent plays performing quite well while the remaining are bad

Horse 2: P230



From the above plot, we can observe this horse is keep improving, especially in last 2 games, it wins the racing.

Scatter Plot of Win Rate and Number of Wins



We set a threshold value for both two plots:

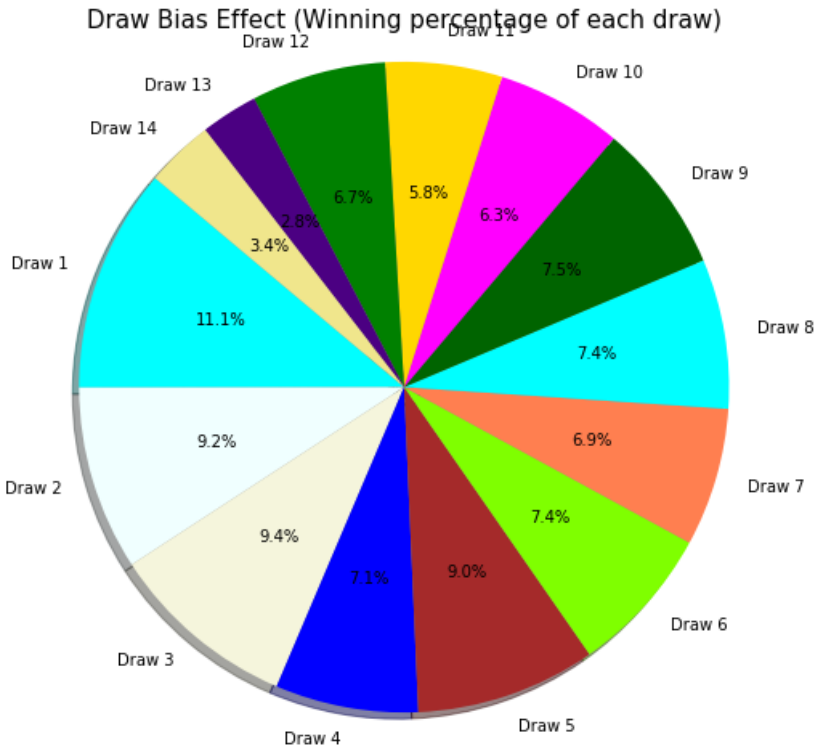
Threshold for horse: Win rate >= 0.5 and Number of wins >= 5

Threshold for jockey: Win rate >= 0.1 and Number of wins >= 100

The cyan/green points indicate the horses/jockeys who greater than the threshold.

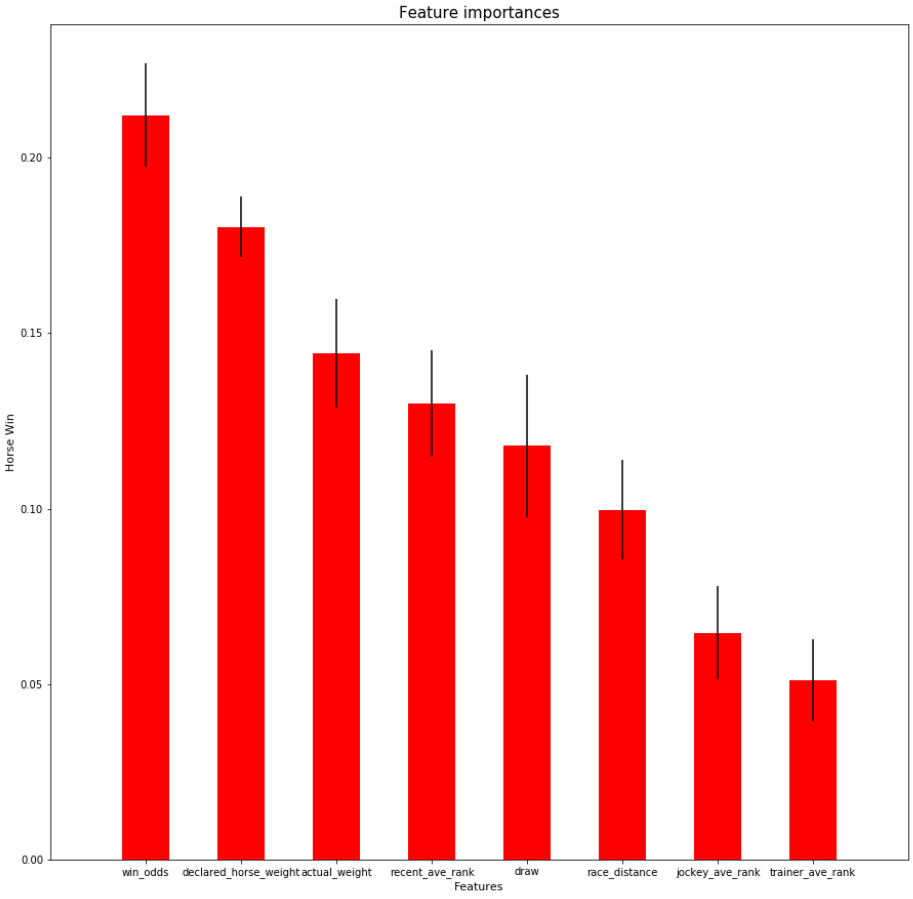
Then, we may say the best horse would be P303 and the best jockey would be J Moreira.

Pie Chart of the Draw Bias Effect



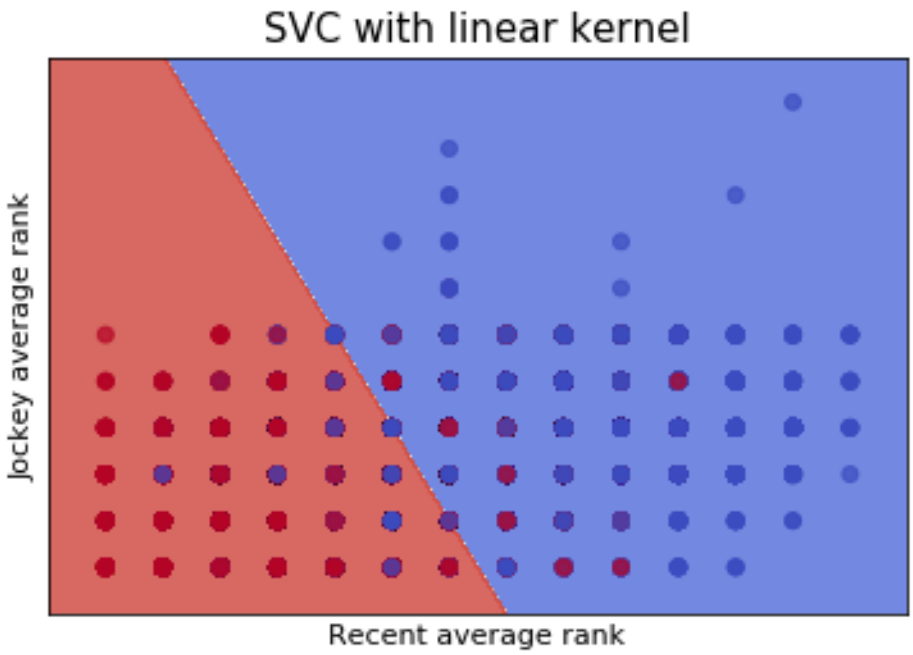
From the pie chart, we can observe the lower the rank is, the higher the possibility of that horse in the rank will win the game. So, we can conclude that low draws really have a considerable advantage.

Bar Chart of the Features Importances



Based on the features we train in classification, the win_odds feature gives us the highest relationship to the win rate, which means that it is the most important feature we should look at when we do a horse racing betting.

Visualize SVM



From the SVM plot, we can observe most of the points are in their correct region while some of them are not.(Red region for Top50Percent horse, blue region for the remaining) Thus, we say that the two features: recent average rank and jockey average rank cannot be used to predict the Top50Percent horses perfectly.