**CSCI 3320: Fundamentals of Machine Learning**

**Project: Horse Racing Prediction**

**[Name Here] (1155112154) & Atif Khurshid (1155085457)**

**2. The Dataset and pre-processing**

**2.2 Data pre-processing**

2.2.3 Indices and features for horses, jockeys and trainers

Number of horses: 2155

Number of jockeys: 105

Number of trainers: 93

**3. Classification**

**3.1 Training Classifiers in Scikit-Learn**

3.1.1 Logistic Regression

// Report your prediction result, prediction evaluation and running time

3.1.2 Naïve Bayes

// state your reason to choose the specific classifier

// Compare your implementation with the Naïve Bayes classifier implemented in scikit-learn in terms of the running time and prediction performances. Report your prediction results, prediction evaluations and running time

3.1.3 SVM

// choose a kernel function and state your reasons

// Report your prediction result, prediction evaluation and running time

3.1.4 Random Forest

// Report your prediction result, prediction evaluation and running time

**3.3 Evaluation of Predictions**

// As we mentioned in section 3.1, you also need to report your prediction result, prediction evaluation result and running time

**3.4 Writing A Report**

Q: What are the characteristics of each of the four classifiers?

Q: Different classification models can be used in different scenarios. How do you choose classification models for different classification problems? Please provide some examples.

Q: How do the cross-validation techniques help in avoiding overfitting?

Q: In addition to the Precision-Recall metric, there are many other metrics can be derived according to the confusion matrix, e.g., the true negative rate TNR=TN/TN+FP, the negative predictive value NPV= TN/TN+FN and so forth. How do you choose evaluation metrics for imbalanced datasets according to the class distribution? Please give your understanding and provide some examples.

**4. Regression**

**4.1 Training Regression Model in Scikit-Learn**

4.1.1 Support Vector Regression Model(SVR)

Kernel: We have selected the RBF kernel because it can handle both linear and non-linear relationships. Moreover, the number of features is small and RBF performs well for small number of features.

Hyperparameters: Epsilon is the width of SVM's soft boundary whereas C is the regularization parameter that controls the trade-off between epsilon and classification accuracy. Hence, a larger C and smaller epsilon result in a complex model which may overfit whereas smaller parameters mean that the model is more general with a large margin of separation between classes. We have selected epsilon=0.1 and C = 5000 because these parameters result in the lowest RMSE which means that they provide a good fit of the training data while ensuring the generality of the model.

4.1.2 Gradient Boosting Regression Tree Model(GBRT)

Loss: We have selected ls (squared distance) because it is similar to lad (absolute difference) and huber (ls + lad) and has no extra parameters unlike quantile. Furthermore, ls results in a model with the best RMSE result.

Hyperparameters: learning\_rate is a parameter of gradient descent and it determines the effect of new trees on the result of regression. n\_estimaters is the number of decision trees which are combined to form a final regression model and it is directly proportional to the complexity of the model. max\_depth is the maximum depth of each decision tree and a small max\_depth prevents overfitting. After a grid search, we assigned the values of learning\_rate=0.05, n\_estimaters=120, max\_depth=5 because these parameters result in the model with the least RMSE.

**4.2 Predicting on Test Data**

SVR: kernel='rbf', C = 5000, epsilon=0.1, gamma= 0.000001

(svr\_model, 1.567, 0.200, 0.473, 4.763)

GBRT: loss='ls', learning\_rate=0.05, n\_estimators=120, max\_depth=5, random\_state=42

(gbrt\_model, 1.550, 0.229, 0.517, 4.390)

SVR: Before normalisation: 1.567

After normalisation: 1.564

It appears that RMSE has decreased slightly but more importantly, the scaled\_svr\_model is far less complex than the unscaled model since it has a smaller C value and a larger gamma value. Hence, normalization had a positive impact on SVR because it is sensitive to differences in the ranges of values of different features.

GBRT: Before normalisation: 1.550

After normalisation: 1.549

RMSE is almost unchanged. Hence, normalization had no effect on the result of decision trees because they are insensitive to linear transformations.

**5. Betting Strategy**

Classification:

|  |  |
| --- | --- |
| Model Name | Money Won |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Regression:

|  |  |
| --- | --- |
| Model Name | Money Won |
| svr\_model | 172 |
| s\_svr\_model | 179 |
| gbrt\_model | 277 |
| s\_gbrt\_model | 277 |

Winning Horse:

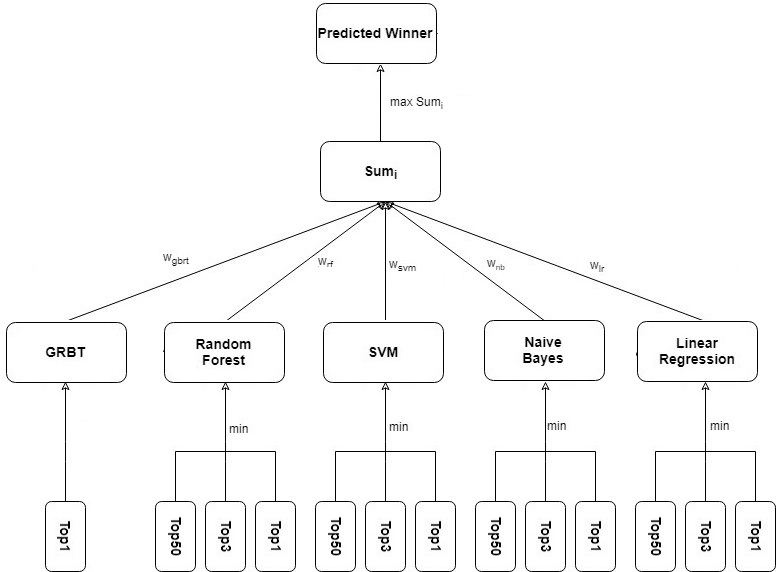
As the results above show, none of the algorithms is accurate enough in its prediction of the winning horse. Therefore, we consider a combination of the results of the 4 classification models and the GBRT regression model to predict the winning horse. This is because the classification models provide three independent labels (top1, top3 and top50) while the regression models’ results can only be processed into a top1 label. Thus, we choose only the most accurate regression model for the final decision.

First, we create labels of top1, top3 and top50percent for the regression model. We then create a vector, , the final prediction for each model, where an element of is 1 only if the predicted top1, top3 and top50percent labels of the model are 1:

Finally, we calculate a weighted sum of the predictions of each model for each horse and the horse with the highest sum is the predicted winner.

The weights are defined according to the relative top1 accuracy of each model. More specifically,

The following is a graphical representation of the selection process:



Betting Strategy:

// Please analyse the result and come up with your own betting strategy to see if you can improve the result (lose less money or win more) in prjreport.pdf. Hint: Sometimes we can choose to bet or not on one race based on our confidence of our result. For example, we are more confident if the score of the winning horse if much higher than others. Also, you can take the odds into consideration. Any attempts are encouraged and will get bonus scores accordingly. The bonus you will get is based on both your idea and your results.

**6. Visualization**

**6.1 Line Chart of Recent Racing Result**

// Select two horses that you are interested in, put down the plots of these two horses in your report, and briefly describe what you observe from the plots.

**6.2 Scatter Plot of Win Rate and Number of Wins**

// Put down the plot in your report, and write down the “best” horse and the “best” jockey in your opinion, and briefly explain why.

**6.3 Pie Chart of the Draw Bias Effect**

// Put down the plot in your report, briefly describe what you observe from the plots, and answer whether low draws really have a considerable advantage?

**6.4 Bar Chart of the Feature Importance**

// Put down the plot in your report, and briefly describe what you observe from the plots

**6.5 Visualize SVM**

// Put down the plot in your report, and briefly describe what you observe from the plots.