KDD Cup 2004- Plus Protein Homology Prediction

Presented by

Puja Sonawane.

TABLE OF CONTENTS

1. Executive Summary
2. Introduction
3. Methodology
4. Modeling Performance Measures
5. Best Model Selection
6. Conclusions
7. Appendix A: ROC Curves
8. Appendix B: Summary Statistics
9. Appendix C: Others
10. References

Executive Summary

A. Best model selected and its performance-

In this study, four models are studied as expected as mentioned- Logistic Regression, Logistic Regression with Interaction, Random Forest Classifier, Gradient Boosting Classifier. Logistic regression with interaction gave better accuracy than logistic regression without interaction but the increase in accuracy was very minute. After that, Random Forest Classifier gave better accuracy that logistic regression and gradient boosting Classifier performs the best in all of the models. It gave 82.90% accuracy on testing dataset.

B. Set of important variables-

Depending on the predictive capacity of variables the set of important variables is as given below:

Feat4, feat8, feat12, feat13, feat14, feat15, feat20, feat31, feat56, feat63, feat66, feat69, feat70, feat71, feat75.

Logistic regression model with these variables gives the model with best accuracy.

C. The impact of important variables to the target variable-

Important selected features are more predictive.

Additionally, after calculating Pearson correlation of predictors with target variable, it is observed that important features are more correlated with the target variable that other non-important features.

Introduction

This project aims to build a classification model. Dataset used in this study came from 2004 KDD CUP competition.

Initially, getting idea of dataset was important as dataset is huge with seventy-eight features. After knowing data size, preprocessing was done like imputing missing values and transformations. Later, to start working on logistic regression model, variable selection was very important task to perform. Depending on predictive capacity of variables, LASSO regression and correlation of variables with target, fifteen variables were found to be very important depending. Next task was to split the data into training dataset and validation dataset with proportion 70% and 30% respectively.

Later, building the logistic regression model using sklearn library was done in python and fitted the training data in model. After fitting training data, it was crucial to check the validation accuracy of model and other performance measures as well including ROC curve, AUC, confusion matrix and precision.

Moving further, building logistic regression model with interaction was built as required. Some interaction columns were inserted in training dataset. Again, fitted logistic regression model considering five interaction terms. Validation accuracy was observed to be better than the logistic regression without interaction.

Coming to trees, third model built was random forest classifier using sklearn library. As there is no need to da vigorous data preprocessing and variable selection for tree-based algorithms, all features were considered building random forest classifier. Later fitted the training data and tested on validation dataset, calculating different performance measures

Next came Gradient Boosting classifier. The Gradient Boosting Classifier Model was built using sklearn library. Again, training data was fitted in model and model performances were tested on validation dataset. According to the performance measures of Gradient boosting, it is the best model of all models studied for this dataset.

Methodology

A. Data preparation including missing value imputation and possible transformation.

In this project, Python programming language (version 3.6) is used. Initial and crucial step of the project was Data Preprocessing. In this project, data preprocessing includes the Missing Value Indicator, Missing value imputation and Transformation of skewed data. After exploring the data, it was found that, features- feat20, feat21, feat22, feat29, feat44, feat45, feat46 and feat55 have missing values. For each feature, one missing value indicator column is created and appended in original dataset to keep track of missing even after missing value imputation. Coming to missing value imputation, missing value in a particular feature is replaced by the mean of all values in a feature. To be specific, univariate imputation algorithm is used with mean strategy. Later, Focus was on transformations. Some features like feat41 were highly skewed. To transform those features suitable transformation was applied.

B. Logistic Regression model formula without interaction terms

$Y = \beta 0 + \beta 1\ Feat4 + \beta 2\ Feat\ 8 + \beta 3 Feat12 + \beta 4\ Feat14 + \beta 5\ Feat15 + \beta 6\ Feat20 + \beta 7\ Feat31 + \beta 8\ Feat56 + \beta 9\ Feat63 + \beta 10\ Feat\ 70 + \beta 11\ Feat71 + \beta 12\ Feat75 + \beta 13\ Feat\ 13 + \beta 14\ Feat\ 69$

C. Logistic Regression model formula with interaction terms

$Y = \beta 0 + \beta 1\ Feat4 + \beta 2\ Feat8 + \beta 3\ \ Feat12 + \beta 4\ \ Feat14 + \beta 5\ Feat15 + \beta 6\ \ \ Feat20 + \beta 7\ Feat\ 31 + \beta 8\ \ Feat56 + \beta 9\ Feat63 + \beta 10\ Feat66 + \beta 11\ Feat70 + \beta 12\ Feat71 + \beta 13\ Feat75 +$

β14 Feat13 + β15 Feat69 + β16 (Feat66 * Feat12) + β17 (Feat71 * Feat31) + β18 (Feat69*Feat56) + β19 (Feat66*Feat14) + β20 (Feat17 * Feat13)

Model Performance Measures

After building model it is important to measure the performance on validation dataset. In this study several performance measures are considered for each model as shown in following table-

Model / Performance Measure	Logistic Regression without Interaction	Logistic Regression with Interaction	Random Forest Model	Gradient Boosting Model
Confusion Matrix	[[5404, 2207], [2205, 5184]]	[[5428, 2183], [2215, 5174]]	[[3845, 3766], [3782, 3607]]	[[3992, 3551], [3969, 3488]]
Accuracy Rate	0.8150	0.8173	0.8205	0.8290
True Positive Rate	5184	5174	3607	3648
True Negative Rate	5404	5428	3845	3820
False Positive Rate	2207	2183	3766	3791
Precision	0.7013	0.7032	0.7163	0.7404

Best Model Selection

From the above section of model evaluation, one can see how the accuracy metric, true positive rate, false positive rate and precision plays an important role in deciding which model performs best. To come to the conclusion, Gradient Boosting is best when compared to all other models particularly for this dataset as mainly it provided the predictive accuracy of 82.90% and precision as 0.74 that cannot be beaten by other model. Additionally, it has high flexibility working with several features with less computational time.

Therefore, Gradient Boosting model is selected as the best model in this scenario.

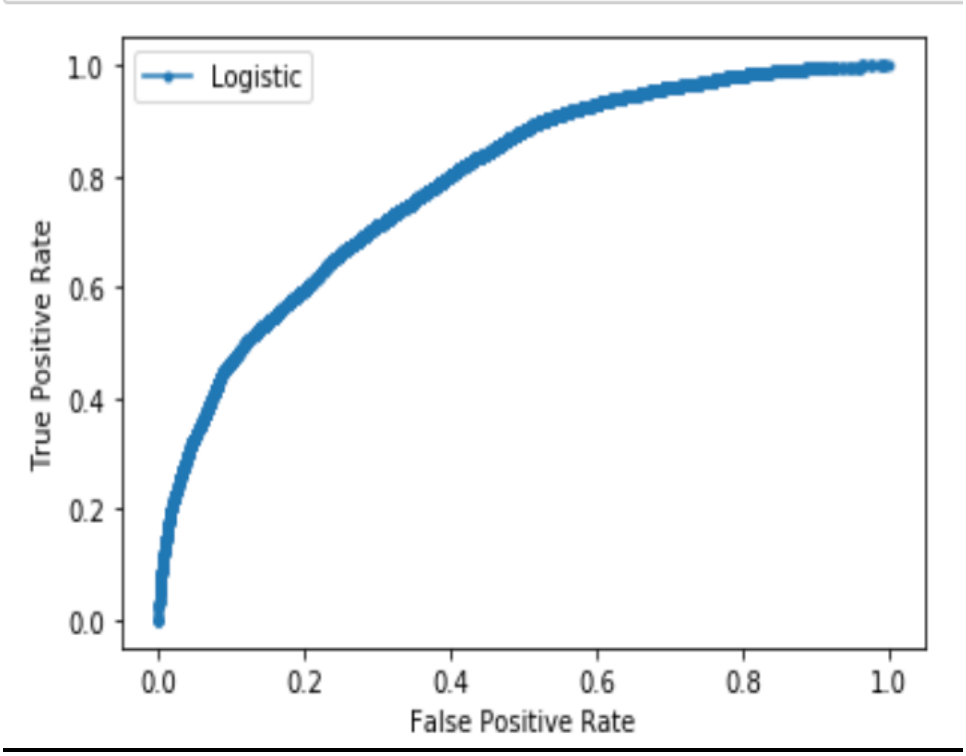
Conclusion

In this project, data exploration, data preprocessing – missing value indicators and imputations, transformations are done. Different models were considered for better classification results such as logistic regression, logistic regression with interaction, tree based- random forest classifier and gradient boosting. Additionally, all models built evaluated using different performance measures as discussed thoroughly in model performance measure section.

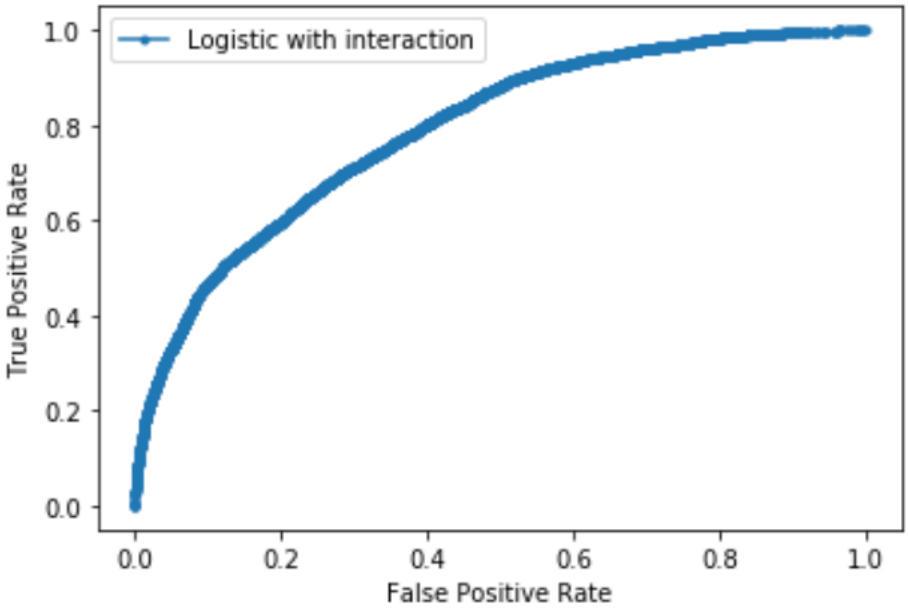
To conclude it in short, logistic regression model gave 81.51% accuracy. Later, considering the interaction between variables and building a revised logistic regression model gave the bit better accuracy of 81.73%. Tree based random forest classification model gave 82.05% accuracy. Finally, the best model concluded is Gradient Boosting Classification model gave the highest accuracy as 82.76% among all models.

Appendix A: ROC Curves:

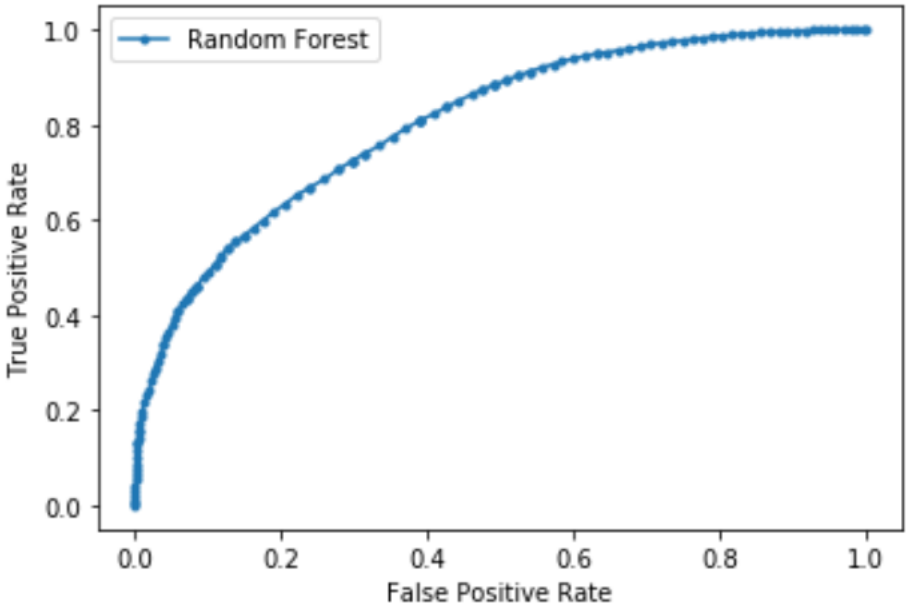
1. Logistic Regression without Interaction:



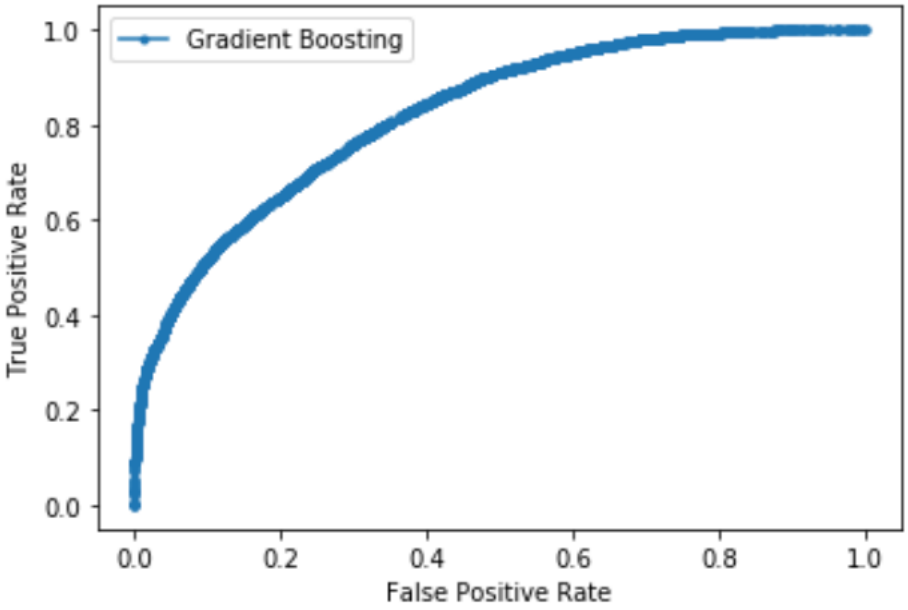
2. Logistic Regression with Interaction:



3. Random Forest Model:



4. Gradient Boosting Model:



Appendix B: Summary Statistics

Feature Name	Missing Value Percentage	Mean	Standard Deviation	Minimum	Maximum	Skewness
target	NA	0.497	0.4999	0.00	1.00	1.11
feat1	NA	0.1556	0.4148	0.00	2.639	2.826
feat2	NA	0.0848	0.2953	0.00	3.429	4.42
feat3	NA	-0.0503	0.2537	-1.00	0.999	-1.56
feat4	NA	0.000060	0.3929	-1.00	1.00	-5.30
feat5	NA	0.1265	0.4006	0.00	2.719	3.24
feat6	NA	0.0498	0.2237	0.00	3.054	5.76
feat7	NA	-0.038	0.2141	-1.00	0.999	-2.22
feat8	NA	0.0028	0.3220	-1.00	1.00	5.89
feat9	NA	0.848	0.453	0.00	6.699	0.862
feat10	NA	0.673	0.511	0.00	5.283	0.658
feat11	NA	-0.283	0.591	-1.00	0.999	0.605

feat12	NA	-0.010	0.993	-1.00	1.000	0.201
feat13	NA	0.072	0.649	-0.99	0.999	-0.010
feat14	NA	0.0008	0.303	-0.99	0.999	0.009
feat15	NA	-0.0008	0.119	-0.99	0.999	-0.319
feat16	NA	0.855	1.068	0.00	9.000	1.498
feat17	NA	0.168	0.410	0.00	4.000	2.464
feat18	NA	0.076	0.180	0.00	0.504	1.911
feat19	NA	0.121	0.440	0.00	6.077	0.676
feat20	68.40	0.001	0.590	-2.453	4.507	-0.026
feat21	68.40	-0.067	0.475	-0.999	1.000	0.288
feat22	68.40	-0.619	0.157	-0.999	-0.000028	0.905
feat23	NA	0.928	0.532	0.00	4.508	1.235
feat24	NA	0.826	0.530	0.00	4.790	1.407
feat25	NA	0.523	0.431	0.00	6.451	1.874
feat26	NA	0.404	0.391	0.00	0.999	0.224

feat27	NA	0.091	0.084	0.00	0.249	0.219
feat28	NA	1.491	1.183	0.00	9.999	7.864
feat29	60.12	0.00	0.00	0.00	0.00	0.00
feat30	NA	0.400	0.490	0.00	1.00	-0.026
feat31	NA	0.001	0.775	-1.00	1.00	0.288
feat32	NA	361.67	617.75	0.00	10000.00	0.905
feat33	NA	4.370	5.668	0.00	75.00	1.235
feat34	NA	0.110	0.108	0.00	0.526	1.407
feat35	NA	0.023	0.151	0.00	0.00	1.874
feat36	NA	0.0230	0.150	0.00	0.00	0.224
feat37	NA	0.002	0.280	-6.470	7.565	0.219
feat38	NA	0.004	0.289	-10.771	9.437	7.864
feat39	NA	-0.004	0.421	-17.321	14.763	0.00
feat40	NA	0.003	0.031	0.00	1.938	18.288
feat41	NA	0.003	0.037	0.00	3.548	27.959
feat42	NA	0.008	0.112	0.00	6.192	24.130

feat43	NA	0.601	0.489	0.00	1.00	-0.413
feat44	28.93	0.427	0.470	0.00	2.878	1.304
feat45	28.93	0.369	0.363	0.00	0.99	0.474
feat46	28.93	-0.268	0.303	-0.99	0.00	-1.023
feat47	NA	0.00	0.00	0.00	0.00	0.00
feat48	NA	0.00	0.00	0.00	0.00	0.00
feat49	NA	0.00	0.00	0.00	0.00	0.00
feat50	NA	0.00	0.00	0.00	0.00	0.00
feat51	NA	0.00	0.00	0.00	0.00	0.00
feat52	NA	3.72	4.38	0.00	10.00	0.529
feat53	NA	0.59	0.083	0.00	0.24	0.947
feat54	NA	1.28	1.09	0.00	9.99	8.485
feat55	37.20	0.00	0.00	0.00	0.00	0.00
feat56	NA	0.00	0.69	-1.00	1.00	-0.0013
feat57	NA	218.00	508.00	0.00	10000.00	7.084
feat58	NA	2.65	4.89	0.00	75.00	3.333

feat59	NA	0.072	0.10	0.00	0.54	1.236
feat60	NA	0.35	0.52	0.00	3.04	1.345
feat61	NA	0.31	0.41	0.00	0.99	0.672
feat62	NA	-0.21	0.33	-0.99	0.00	-1.183
feat63	NA	0.00	4.10	-22.00	22.00	-0.044
feat64	NA	0.01	0.01	0.00	0.09	2.750
feat65	NA	0.96	0.50	0.00	5.62	0.161
feat66	NA	-0.00	0.95	-1.00	1.00	0.0072
feat67	NA	0.78	0.28	0.00	1.00	-1.865
feat68	NA	0.15	0.47	-0.99	0.99	0.096
feat69	NA	0.008	0.76	-1.00	1.00	-0.013
feat70	NA	0.00	0.44	-0.99	0.99	0.012
feat71	NA	0.003	0.38	-0.90	0.90	-0.010
feat72	NA	0.05	0.18	0.00	0.99	3.475
feat73	NA	0.06	0.28	0.00	3.42	5.025
feat74	NA	-0.01	0.17	-1.00	0.99	-1.419

feat75	NA	-0.00	0.29	-1.00	1.00	-0.041
feat76	NA	0.09	0.31	0.00	3.00	3.464
feat77	NA	0.002	0.01	0.00	0.38	12.112
feat78	NA	0.06	0.22	0.00	1.00	3.223

Appendix C: Others (AUC)

Models	**AUC values**
Logistic Regression	0.795
Logistic Regression with Interaction	0.795
Random Forest Classifier	0.802
Gradient Boosting Classifier	0.818