NATIONAL UNIVERSITY OF SINGAPORE

BT5152: Decision Making Technology for Business | Assignment 3

Swetha Narayanan – A0074604J

Q1 Random Forest of a post pruned rpart

- Training Data: First 1500 rows in A3_train
- Test Data: Last 500 rows in A3_train
- Performance Metric : Accuracy

Model Results

Model Name	Training Accuracy	Test Accuracy
Random Forest with post-pruned rpart	0.787	0.754

Q2 Stacking of 3 Algorithms : C50 with default parameter values, KNN with k=3, and random forest in Task 1. Logistic regression is used for the level1 algorithm.

- Train L0 Data: First 1000 rows in A3_train
- Train L1 Data: Next 700 rows in A3_train
- Test Data : Last 300 rows in A3_train
- Performance Metric : Accuracy

Model Name	L0 Training Data Prediction Accuracy	L1 Training Data Prediction Accuracy	Test Data Prediction Accuracy
C50	0.921	0.741	0.753
KNN	0.797	0.621	0.593
Random Forest with post-pruned rpart	0.829	0.733	0.750
Stacked Model with GLM as Metaclassification model	NA	0.751	0.760

Comparing Stacking implementation and L0 models

Stacked model gives highest prediction accuracy on both training and test dataset compared to the base models. The meta classifier (Logistic Regression in this case) highlights each base model where it performs best and discredits it when it performs poorly.

Note: Stacking was performed manually and the GLM based meta classifier is re-usable for other classifications

Q3 Data Competition Predict true label of 2000 rows in test file

- Training Data: First 1500 rows in A3_train

- Validation Data: Last 500 rows in A3_train
- Test Data: 2000 rows in A3_test
- Performance Metric : AUC

Feature Selection

As we can see from the corrplot in html file , x1, x5, x8, x14 are highly correlated. In favour of the law of parsimony and to reduce multi-collinearity, we keep only 1 of the correlated features.

Feature Engineering

Using StepAIC, we do step-wise model selection by AIC and we see that the interaction between x4*x8 comes out to be significant. So we use this information to create additional feature x16. The addition of this new feature improves model performance by atleast 1-2%.

Model Results

Model built using k-fold cross validation	Training AUC	Validation AUC
Random Forest	0.753	0.802
XGBoost	0.745	0.810
C5.0	0.717	0.784
NNet	0.607	0.651
GLM Step AIC	0.585	0.627
Stacked Model with XGBoost as Metaclassification model	0.998	0.814

Model Correlation

Column1	rf	xgbTree	C5.0	nnet	glmStepAIC
rf	1	0.8895413	0.9399375	0.994254222	-0.087782702
xgbTree	0.8895413	1	0.9315069	0.929076925	0.365341273
C5.0	0.9399375	0.9315069	1	0.965385181	0.061684104
nnet	0.9942542	0.9290769	0.9653852	1	0.000372219
glmStepAIC	-0.0877827	0.3653413	0.0616841	0.000372219	1

Discussion

We observe that the output of Level 0 classifiers (L1 Training Predicted values) seem to be quite correlated. Since the Level 1 meta learner will have multi-collinearity problems while training the predicted Level 1 values, we use XGboost as the metalearner since its resilient to these issues. AUC of Stacked model outperforms base models in both training and validation datasets. This is due to the smoothing nature of stacking and the effectiveness of majority voting where it highlights good performing models for different types of input data. Stacking also helps stabilize prediction performance