NATIONAL UNIVERSITY OF SINGAPORE

BT5152: Decision Making Technology for Business | Assignment 2

Swetha Narayanan – A0074604J

Q1 Model Selection Using Caret and C5.0 Algorithm to predict credit default

Model Results: With Stratified Sampling

Model Name	Trials	Accuracy
2 Fold Cross Validation	11	0.75
10 Fold Cross Validation	12	0.76
10 Fold Cross Validation with 5 repeats	25	0.83
10-fold cross validation with selectionFunction = "oneSE"	6	0.74
10-fold cross validation with selectionFunction = "tolerance"	9	0.74

Model Results: Without Stratified Sampling

Model Name	Trials	Accuracy
2 Fold Cross Validation	32	0.83
10 Fold Cross Validation	29	0.82
10 Fold Cross Validation with 5 repeats	30	0.84
10-fold cross validation with selectionFunction = "oneSE"	8	0.71
10-fold cross validation with selectionFunction = "tolerance"	6	0.74

Repeated Stratified 10-Fold Cross Validation gives highest accuracy which is consistent with the extensive experiments done by researchers that it's the best choice of K to get accurate estimate. With and without stratification gives similar results. But stratification is preferred because each fold is representative of the outcome of the data as a whole and it also reduces variance. Repeated CV is also preferred because the results are averaged and that reduces variance which reduces overfitting and gives better overall prediction accuracy. Also higher the number of trials i.e boosting iterations, better the accuracy.

Q2 Neural Network Classification to predict credit default using logistic activation function

Data Preparation: We use 80:20 rule to split data into training and test set. We preprocess the data by scaling numerical variables and encoding categorical variables.

Model Results

Model Name	Threshold Value	Hidden Node Config	AUC
NeuralNet - Logistic	0.01	1	Default =1 0.7336677116 Default =2 0.7362382445

Q3 2 Level Neural Network Regression to predict price of diamond using tanh activation fn

Data Preparation: We use 80:20 rule to split data into training and validation set. We preprocess the data by scaling numerical variables and encoding categorical variables.

Model Results

S.No	Hidden Node Config	Threshold Value	Stepmax	RMSE	RSquared	MAE	Training RMSE	Validation RMSE
1 (log price)	10,10	0.02	2e+05	0.13	0.94	0.087	508.023	6774.822
2	10,10	0.02	2e+05	0.14	0.92	0.082	527.906	3546.056
3	15,10	0.02	2e+05	0.17	0.88	0.098	517.269	3804.17
4	12,12	0.02	2e+05	0.19	0.87	0.094	541.166	7500.488

Best performing model: I tried various combinations of hidden layers. The best performing model with lowest training and validation RMSE is the model with 10 nodes each in 2 hidden layers.

Discussion

1) **Standardization helps**: I observed that Neural network performance was better and it speeded up convergence when the variables where standardized.

2) **Log normal of prices:** Exploration of the diamond prices indicated that the data was right-skewed. So taking log of the prices resulted in a more normal distribution of the prices. It also resulted in a better model.

	RSquared	RMSE
With raw prices	0.92	0.14
With log prices	0.94	0.13

3) **Parsimonious model:** As the number of layers or the number of nodes increased, there was an over fitting problem (low training RMSE, high validation RMSE) which means that the model does not require that level of complexity. Hence, reducing the number of nodes gave better validation RMSE.

Other Improvements possible in future

1) **Try Different learning rates**: We went with default learning rate here= 0.25
2) **Log:** Carat feature is slightly right skewed. We could convert that to log normal as well.
3) **Feature Engineering and Feature Selection:** One hot encoded categorical variables don't really capture the **ordinal** nature of some features in the dataset like Clarity, Color and Cut which are correlated with price. To better incorporate this information, we could convert these features into numerical data and check if that improves prediction performance. We could try removing predictors that don't improve accuracy as well.