# Binary Prediction - Client Churn or No-churn

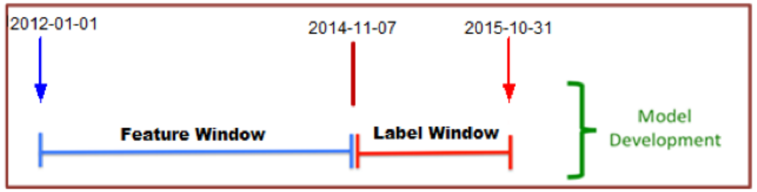



Table 5.1: Model Development Summary

| | |
|---|---|
| Dataset at Label Window | Train (4,130 obs.) & Test (1,000 obs.) with 74 variables |
| | Binary outcome variable (Churn or No Churn) |
| Feature Selection | Significant Logit vars. + RF important vars. - Pearson Correlated vars. |
| Candidate prediction models | Logistic Regression (Logit), Random Forest, C5.0 |
| Prediction probability threshold | 50% for all models and then |
| | to mean probability of (TP+TN) to adjusts for population bias |
| Model Comparison | Over-all prediction accuracy and others |
| | Area under ROC |
| Validation | 10-fold cross validation (90%-10% training/test split) |
| Attempted Misclassification penalty | 2 to 1 in favour of Type II(misclassified churns) |

R markdown document https://github.com/raulmanongdo/R-PredictionModel/blob/master/kc_client_churn_FINAL_G.md
Thesis publication at https://opus.lib.uts.edu.au/handle/10453/123179

# I’ll cover in this talk  my journey …

- Definitions – client churns and churn measure
- Literature Review
- Feature and model selection and comparison
- Improving accuracy of selected model

- Will dwell more on techniques/ R , not industry insights
- Coming from a data dev background

- Q &A

# Definition – Client Churn

In Strouse [1999], churn (aka ‘**attrition**’) is the annual turn-over of the market base.

$$ChurnFlag \Leftarrow function\,(Client\,Program\,Enrollments, Discharge\,Reason)$$

$$\Leftarrow \begin{cases} 1\ (\textbf{Churn})\ Discharge\,Reason \in R\ and \\ \quad Client\,Program\,Enrollment\ type \in P\ and \\ \quad Client\,Program\,Enrollment\ count = 0 \\ 0\ (\textbf{non}\ \ \textbf{Churn})\ otherwise. \end{cases}$$

*where*

*R is a set of Discharge Reason* $r_1, r_2, \ldots, r_n$ *and*

*P is a set of monitored Client Programs* $p_1, p_2, \ldots, p_n$ *and*

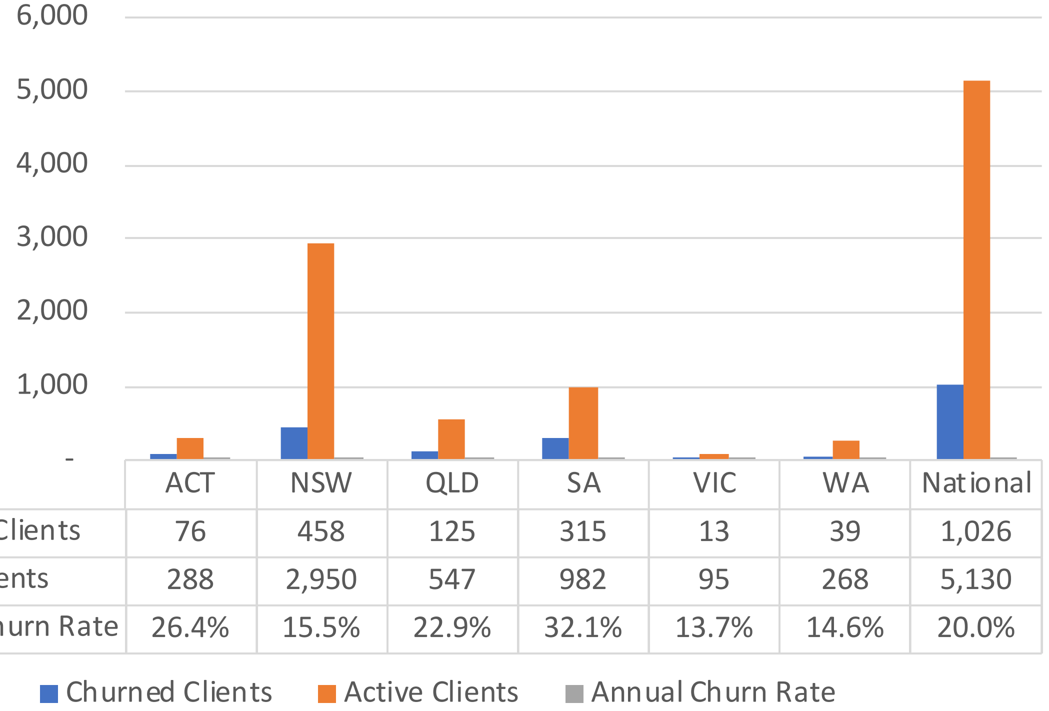
*Client Program Enrollment count is the remaining program(s) after discharge.*

# Measure - Churn Rate

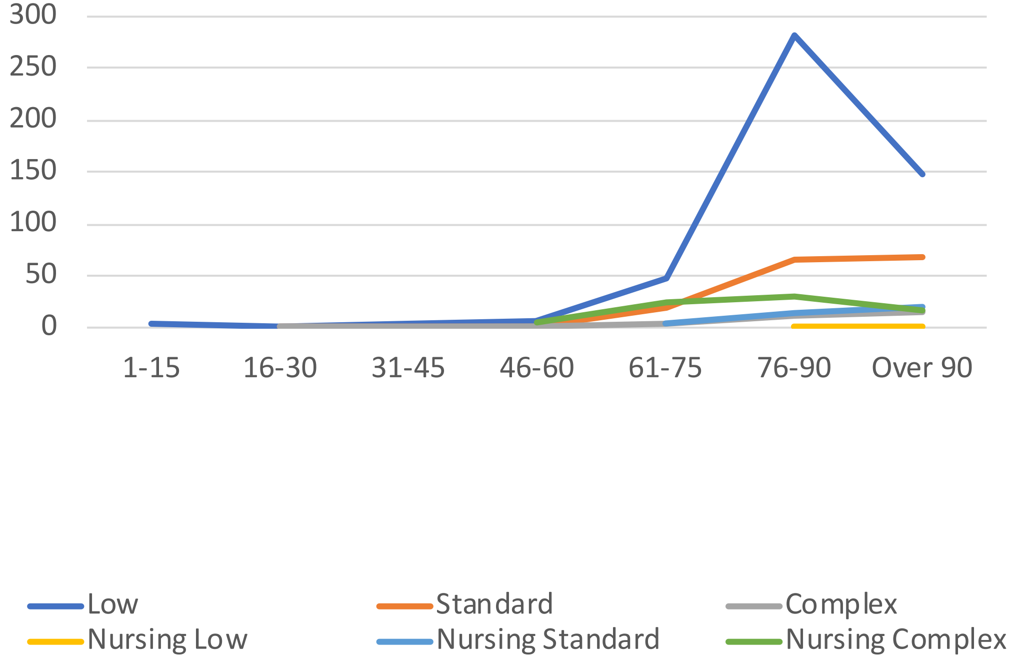
$$
\begin{aligned}
Active\ Clients_{(at\ end\,month)} = &\sum Active\ Clients_{(at\ begin\,month)} \\
&+ \sum new\ Clients_{(within\,month)} \\
&- \sum Churned\ Clients_{(within\,month)} \\
&- \sum Non\,churn\ Client\ Exits_{(within\,month)}
\end{aligned} \qquad (2)
$$

$$
Churn\ Rate\% = \sum_{i=1}^{k} \frac{Churned\ Clients^{k}_{(within\ month)}}{Active\ Clients^{k}_{(at\ end\,month)}} \qquad (3)
$$

where K is set of States $s_1, s_2.., s_k$




| | ACT | NSW | QLD | SA | VIC | WA | National |
|---|---|---|---|---|---|---|---|
| Churned Clients | 76 | 458 | 125 | 315 | 13 | 39 | 1,026 |
| Active Clients | 288 | 2,950 | 547 | 982 | 95 | 268 | 5,130 |
| Annual Churn Rate | 26.4% | 15.5% | 22.9% | 32.1% | 13.7% | 14.6% | 20.0% |



# Literature Review – Prediction Models

Table 3.1: Client Churn Prediction Models reviewed

| Title of Paper | Models and Techniques |
|---|---|
| Model of Customer Churn Prediction on Support Vector Machine | Factor Analysis, SVM |
| Telco Churn Prediction with Big Data | Logit, C4.5, SVM Naive Bayes, ANN |
| Churn prediction in subscription services: An application of SVM while comparing two parameter-selection techniques | SVM, Logit, RF |
| Improving customer attrition prediction by integrating emotions from client/company interaction emails and evaluating multiple classifiers | SVM, Logit, RF |
| Customer churn prediction by hybrid neural networks | SOM Cluster, ANN |
| Variable selection by association rules for customer churn prediction of multimedia on demand | C5.0, ANN |
| Storm Prediction: Logistic Regression vs RFfor Unbalanced Data | Logit, RF |
| Prediction modelling and pattern recognition for patient readmission | CART, CHAID, C5.0 , ANN |
| Building comprehensible customer churn prediction models with advance rule induction techniques | AntMiner (Ant Colony Optimization) ALBA (SVM) |

# Literature Review – Home-based Care Services

Table 3.2: Churn associated studies on Home-based Care Services

| Title of Paper | Models Used |
|---|---|
| A data mining approach in home healthcare: outcomes and service use | CART(Classification and Regression Tree) |
| Data mining techniques for patient satisfaction data in home care | Box Analysis, Segmentation, CHAID and ANOVA |
| The home care satisfaction measure: a self-centred approach to assessing the satisfaction of frail older adults with home care services | Correlation and Common Factor Analysis |

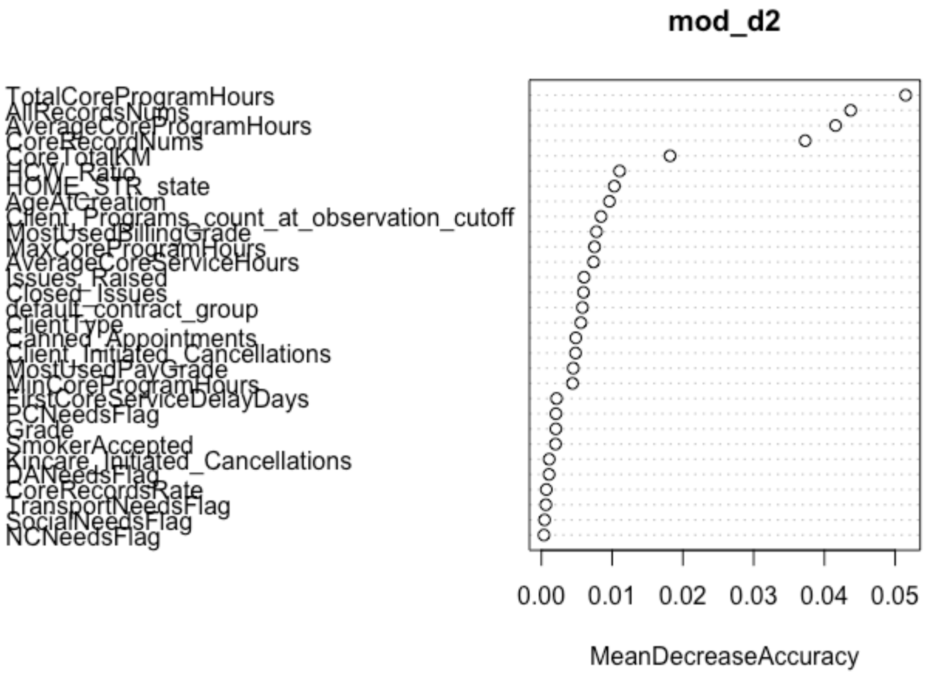
# Attributes and Feature Selection

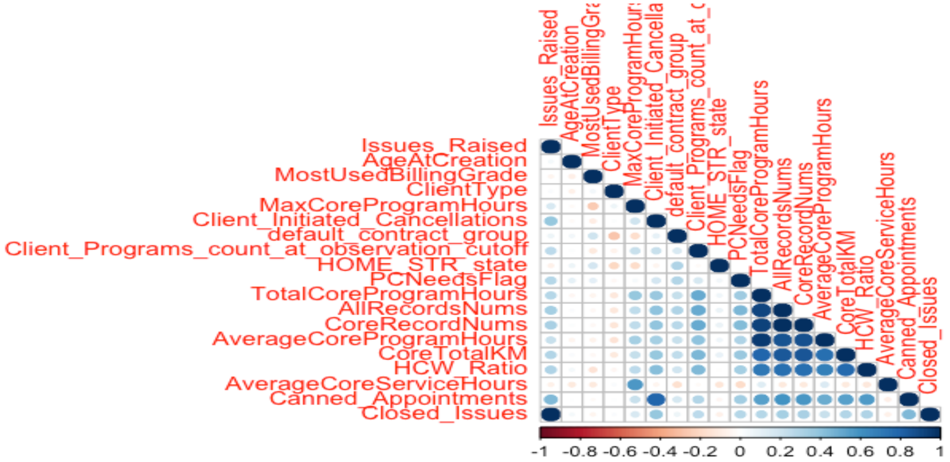
- Client Demographics *e.g. gender, home state, age*
- Client Program Enrollments
  - *e.g. enrolled program, services, discharges*
- Service Delivery and Expectations *e.g. time sheet, service preference*
- Client Interactions *e.g. customer complaints, communications*
- Client Satisfaction Survey *e.g. Net Promote Score survey*
- Derived Attributes –*e.g. home care worker ratio, issues raised and other RFM values*

Table 5.3: Logistic Regression significant variables

| | Estimate | $Pr(>\|z\|)$ | Signif. |
|---|---|---|---|
| ClientTypeYou | 4.4516080 | 3.084973e-03 | ** |
| MostUsedBillingGrade.L | 0.9074320 | 4.072909e-04 | *** |
| FrequentschedStatusGroupCancelled | 0.6448600 | 3.036731e-02 | * |
| Issues_Raised | 0.4031573 | 1.482545e-16 | *** |
| default_contract_groupPrivate/Commercial | 0.3977963 | 7.502689e-03 | ** |
| PCNeedsFlagY | 0.2815010 | 3.345663e-02 | * |
| Client_Programs_count_at_observation_cutoff | 0.1484453 | 7.854017e-03 | ** |
| MinCoreProgramHours | 0.1203954 | 2.865223e-02 | * |
| MaxCoreProgramHours | -0.2197452 | 2.671882e-03 | ** |
| Client_Initiated_Cancellations | -0.2415312 | 5.610230e-03 | * |
| AgeAtCreation | -0.3542742 | 7.664414e-17 | *** |
| RespiteNeedsFlagY | -0.4273024 | 3.742558e-02 | * |
| HOME_STR_stateNSW | -0.5914624 | 1.272792e-02 | * |
| MostUsedBillingGrade.C | -0.6602870 | 4.304246e-02 | * |
| HOME_STR_stateVIC | -1.1414711 | 1.011449e-02 | * |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1




*d2_corr <- cor(d2, method = "pearson")*

*corrplot(d2_corr, type = "lower")* # High collinearity (>=.814)

# Logistic Regression

Z-score normalization applied only to regression, not on tree based models

Table 5.5: Standardised Logistic Regression Coefficients

| | Estimate | Std. Error | $Pr(>\|z\|)$ | Significance | Odds |
|---|---|---|---|---|---|
| (Intercept) | 0.4337 | 145.78538 | 0.997626 | | |
| Issues_Raised | 0.34927 | 0.05183 | 1.60E-11 | *** | 1.42 |
| AgeAtCreation | -0.36461 | 0.04645 | 4.17E15 | *** | 0.69 |
| MostUsedBillingGrade.L | 0.93084 | 0.19525 | 1.87E-06 | *** | 2.54 |
| MostUsedBillingGrade.Q | -7.24166 | 445.36776 | 0.987027 | | |
| MostUsedBillingGrade.C | -0.26828 | 0.20914 | 0.199576 | | |
| MostUsedBillingGrade.4 | 7.59609 | 493.3888 | 0.987716 | | |
| MostUsedBillingGrade.5 | 0.58078 | 0.1972 | 0.003228 | ** | 1.79 |
| MostUsedBillingGrade.6 | -10.04761 | 671.41712 | 0.98806 | | |
| ClientTypeCAC | 1.68353 | 1.21489 | 0.165823 | | |
| ClientTypeCCP | 1.10212 | 1.16913 | 0.345839 | | |
| ClientTypeCom | -0.16902 | 1.13103 | 0.881206 | | |
| HOME_STR_stateVIC | -1.25828 | 0.47128 | 0.007587 | ** | 0.28 |
| HOME_STR_stateWA | -0.39709 | 0.36099 | 0.271331 | | |
| PCNeedsFlagY | 0.18885 | 0.14038 | 0.178552 | | |
| AverageCoreProgramHours | -0.72001 | 0.09461 | 2.74E-14 | *** | 0.49 |
| HCW_Ratio | -0.3221 | 0.09466 | 0.000667 | *** | 0.72 |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

To ascertain, the p-value of less than 0.001 below tells us that our model was statistically and significantly better than the null model.

```
> with(kc.glm, pchisq(null.deviance - deviance,
          df.null - df.residual, lower.tail = FALSE))
 [1] 3.995819e-111
```

*kc.glm <- glm(Label ~ ., family = binomial(link = "logit"), train.scaled)*
*pr.kc.glm <- predict(kc.glm, newdata = test.scaled, type = 'response')*

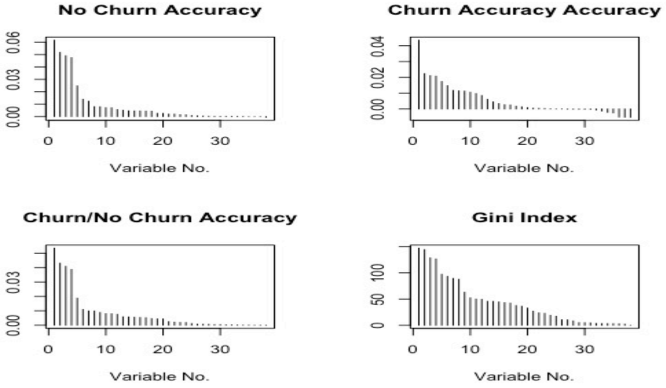
```
Call:
 glm(formula = Label ~ ., family = binomial(link = "logit"),
     data = train.scaled)

 Deviance Residuals:
 Min        1Q    Median       3Q      Max
 -2.2030  -0.6639  -0.4741  -0.1776   3.3285

 (Dispersion parameter for binomial family taken to be 1)     ›m

 AIC: 3563.8

 Number of Fisher Scoring iterations: 14
```

*fitted.results.glm <- ifelse(pr.kc.glm > .5, 1, 0)*
*confusion_maxtix <- table(test.scaled$Label, fitted.results.glm)*

| | fitted.results.glm | |
|---|---|---|
| | 0 | 1 |
| 0 | 3199 | 105 |
| 1 | 654 | 172 |

Common way of computing prediction
Accuracy across all candidate models

# Random Forest



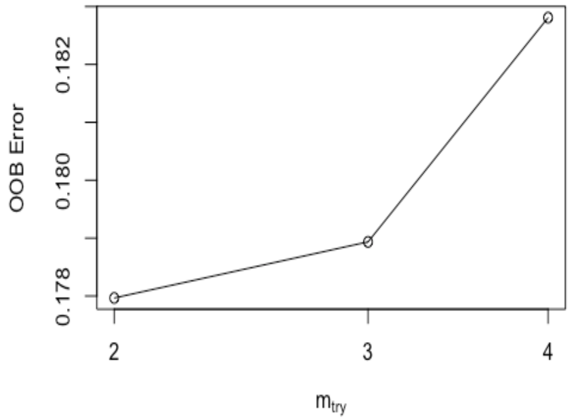

*kc.rf <- randomForest( Label ~ .,data = train, mtry= RFmtry_param, ntree = 1000, keep.forest = TRUE, importance = TRUE )*
*pr.kc.rf <- predict(kc.rf, newdata = test, type = 'prob')[, 2]*

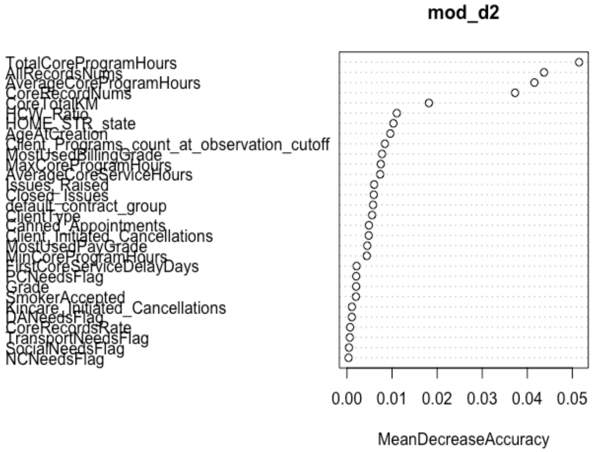
```
Call:
randomForest(formula = Label ~ ., data = train, mtry = RFmtry_param,
     ntree = RFnTree_param, keep.forest = TRUE, importance = TRUE)
Type of random forest: classification
Number of trees: 1000
No. of variables tried at each split: 2

OOB estimate of  error rate: 17.7%
Confusion matrix:
0    1 class.error
0 3164 140  0.04237288
1  591 235  0.71549637
```

Chosen accuracy measure is mean decease in accuracy for BOTH label class (*kc_RF_var_imp_measure*)

*bestmtry <- tuneRF( train[,-ndxLabel], train$Label, ntreeTry = 1000, plot = TRUE, type - kc_RF_var_imp_measure )*

*varImpPlot( kc.rf, sort = TRUE,type = k_RF_var_imp_type, class = NULL, scale = FALSE, main = '')*



# C5.0 Decision Tree

```
kc.c50 <- C50::C5.0( x = train[-ndxLabel], y = train$Label, trial = C5.0Trials_param, rules = FALSE, = C5.0Control(earlyStopping = TRUE) )
pr.kc.c50 <- predict(kc.c50, type = "prob", newdata = test[-ndxLabel])[, 2]
kc.c50.rules <-C50::C5.0( x = train[,-ndxLabel], y = train$Label,  trial = C5.0Trials_param, rules = TRUE,  control = C5.0Control(bands = 100, earlyStopping = TRUE))
```

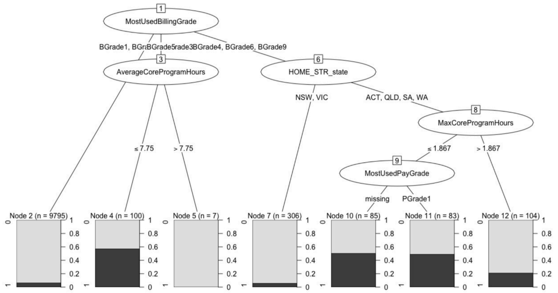



Fig. 5. C5.0 Model Decision Tree

n.b. Tree above is sample graph only.

```
AverageCoreProgramHours <= 46:
   ClientType = HAC:
   :...default_contract_group = Disability: 1 (1.7)

 AverageCoreProgramHours <= 40:
   Client_Programs_count_at_observation_cutoff <= 2:
     AgeAtCreation <= 83:
       Client_Initiated_Cancellations <= 3:
         AverageCoreProgramHours <= 13.75:
           Issues_Raised <= 0.7666531:
             ClientType in {Bro,CAC,Com,Dom,EAC,HAC,Per,Pri}:
               HCW_Ratio <= 2: [S1]
                 default_contract_group in {Disability,DVA/VHC}: 1 (1.5)
```

```
Number of boosting iterations: 10
Average tree size: 17.6
...
boost            641(15.5%)   <<

    (a)   (b)    <-classified as
   ----  ----
   3191   113    (a): class 0
    528   298    (b): class 1
    Trial        Decision Tree
     -----      ----------------
        Size      Errors

        0      25   671(16.2%)
        1      11   788(19.1%)
        2      15   946(22.9%)
        3      22   938(22.7%)
        4      15  1026(24.8%)
        5      15   881(21.3%)
        6      18   920(22.3%)
        7      20   823(19.9%)
        8      21   734(17.8%)
        9      14   745(18.0%)
     boost            641(15.5%)   <<
```

# Model Comparison and Selection

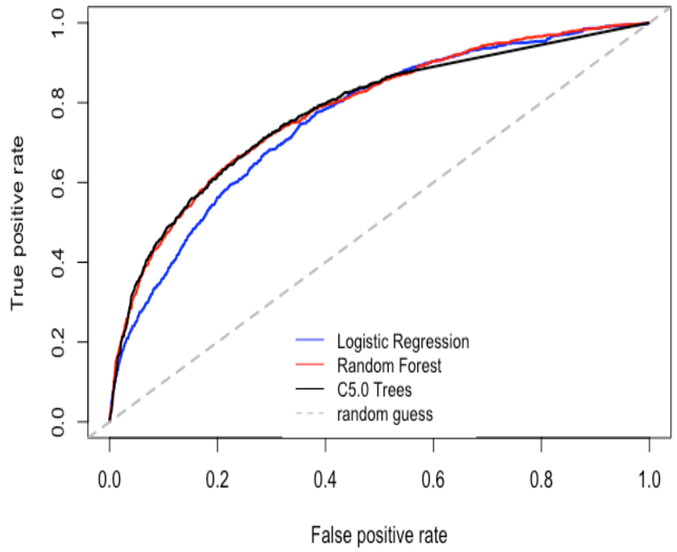



Table 5.8: Comparison of Prediction Model Performances

tp+tn/(tp+tn+tp+tn) tp/(tp+fn) tp/(tp+fp) 2*prec*rec/(1+rec)

| | Train | Test | | | | | 10-fold | |
|---|---|---|---|---|---|---|---|---|
| Population | 4,130 | 1,000 | | | | | Cross Validation | |
| | Accuracy | Accuracy | Precision | Recall | F-score | AUC | Accuracy | AUC |
| Logit | .8162 | 0.832 | 0.7353 | 0.250 | 0.3731 | 0.7993 | 0.8162 | 0.7594 |
| RF | .8447 | 0.831 | 0.8039 | 0.205 | 0.3267 | 0.7817 | 0.8259 | 0.7822 |
| C50 | .8230 | 0.838 | 0.9524 | 0.200 | 0.3306 | 0.7793 | 0.8283 | 0.7772 |

```
rocGLM <- roc(test.scaled$Label, pr.kc.glm)
rocRF <- roc(test$Label, pr.kc.rf)
roctest.eval.GLM.RF <- roc.test(rocGLM, rocRF, method = "delong", paired = TRUE)
```

Table 5.9: Pair-wise comparison of model significance (AUC)

| | p-Value Test | p-Value 10-X Test | 95% C.I. on Test? | 95% C.I. on 10-fold X Val |
|---|---|---|---|---|
| Logit to RF | 0.2912 | 3e-04 | No | Yes |
| Logit to C50 | 0.2074 | 0.0046 | No | Yes |
| RF to C50 | 0.8668 | 0.304 | No | No |

**Selected Model : C5.0**

- Highest over-all prediction accuracy at 83.8%
- Simplicity and interpretability

# Improving C5.0 Model accuracy

Rebuild model to cater for

- Population bias of minority class (i.e. Churns). Adjust threshold from default 50% to mean prob. value of TP&TN

  *fitThreshold.adj <- summary(y[, 2])[4] #Mean Probability value*
  *fitted.results.c50 <- ifelse(pr.kc.c50 > fitThreshold.adj, 1, 0)*

- Introducing misclassification penalty

  *miscosts <- matrix(c(NA, 2, 1, NA), nrow = 2, ncol = 2, byrow = TRUE)*
  *C50::C5.0( x = train[-ndxLabel], y = train$Label, trial = C5.0Trials_param, costs = miscosts, rules = FALSE, control = C5.0Control(earlyStopping = TRUE) )*

Table 5.10: C5.0 model parameter tuning

| | Accuracy | Precision | Recall | F-score |
|---|---|---|---|---|
| No tuning | 0.838 | 0.9524 | 0.200 | 0.3306 |
| Probability threshold set at 25% | 0.812 | 0.5405 | 0.400 | 0.4598 |
| Type 2 misclassification penalty of 2 to 1 | 0.829 | 0.8372 | 0.180 | 0.2963 |

Without rebuilding model, assign unequal weights to precision and recall and recompute F-score

# Conclusion

“In a major IT company-sponsored tournament in developing client churn systems participated by practitioners and academics alike, it was concluded that

- Logistic Regression and tree approaches perform well and were good techniques to begin with by companies starting up a predictive modelling function.
- Exploring several techniques to develop one model may not pay off. “

Neslin, S. A. et. al, ‘Defection detection: Measuring and understanding the predictive accuracy of customer churn models’, Journal of Marketing Research, 2006.

Thank you!

Raul Manongdo

raul@manongdo.com

https://www.linkedin.com/in/raulmanongdo/