HCL Employee attrition prediction using Logistic regression

Kannu-priya

April 20, 2018

Contents

1. Data Import

```
#Import imputed data file
dt1 <- read.csv(file = 'D:/Kannu Priya/DataScience/DataSciencePrinciples/AttritionProject/data/dataR.cs

glimpse(dt1)
```

```
## Observations: 7,372
## Variables: 22
## $ per.id                    <int> 10102924, 10103073, 10114021, 101356...
## $ Status                    <fct> Active, Active, Active, Active, Acti...
## $ Age                       <int> 41, 54, 62, 64, 55, 46, 37, 38, 59, ...
## $ Gender                    <fct> Female, Male, Female, Male, Male, Ma...
## $ Band                      <fct> E3, E3, E3, E3, E3, E3, E3, E3, E2, ...
## $ Tenure                    <dbl> 17.03, 17.02, 14.05, 13.08, 17.07, 1...
## $ CTC                       <dbl> 102860, 93767, 107973, 103912, 10189...
## $ Nationality               <fct> Indian, Indian, American, American, ...
## $ L1                        <fct> Apps & SI, Infra, Infra, Apps & SI, ...
## $ Project.Tenure            <dbl> 4.4, 10.5, 37.9, 9.5, 1.7, 3.4, 22.5...
## $ Trainings                 <int> 0, 0, 0, 0, 13, 17, 0, 0, 0, 0, 3, 0...
## $ Worksite                  <fct> Texas, Pennsylvania, California, New...
## $ SSD.count                 <fct> 0, 0, 0, 0, 0, 0, 0, , , 0, 0, 0, 0,...
## $ Performance.Rating        <fct> Distinguished Performance, Exception...
## $ Expat.Local               <fct> local, local, local, local, local, l...
## $ Time.to.allocation.expiry <dbl> 9.0000000, 2.0000000, 16.0000000, 2....
## $ Time.to.promotion         <dbl> 3.8219178, 2.5397260, 2.7917808, 1.6...
## $ Active.Country            <fct> USA, USA, USA, USA, USA, USA, USA, I...
## $ Visa.Type                 <fct> H-1, GC/Citizen, GC/Citizen, GC/Citi...
## $ Validity.End.Date         <fct> 10/2/2020, Nil, Nil, Nil, Nil, Nil, ...
## $ meanCTC                   <dbl> 99606.70, 94618.06, 104294.11, 10391...
## $ Time.to.visa              <int> 894, 100000, 100000, 100000, 100000,...
```

---

2. Create Training & testing data set

```
#Setting a seed before taking random sample so that same sample is reproduced for future reference
set.seed(10000)

# Create training set using sample_frac
train_set <- dt1 %>% sample_frac(.7)
# let's create our testing set using the employee code (per.id) column
test_set <- dt1 %>% filter(!(per.id %in% train_set$per.id))
```

```
## Warning: package 'bindrcpp' was built under R version 3.3.3
```

---

3. Logistic regression model training

Using the generalized linear model, glm() function, make a logistic regression analysis using 'Status' feature as outcome, with the rest of key features in the training dataset as independent predictors

```
# Fit a logistic regression model with intercept only
mod_1 <- glm(Status ~ Age + Gender + Band + Tenure + CTC + L1 + Project.Tenure + Performance.Rating + Ti

# review the features and coefficients
tidy(mod_1)
```

```
##                                                  term      estimate
## 1                                         (Intercept) -2.482087e+00
## 2                                                 Age -2.825255e-02
## 3                                          GenderMale  1.059649e-01
## 4                                              BandE1  1.068328e+00
## 5                                              BandE2  1.230906e+00
## 6                                              BandE3  1.568778e+00
## 7                                              Tenure -1.827206e-01
## 8                                                 CTC -1.752675e-05
## 9                                              L1ERS -1.802107e-01
## 10                                           L1Infra -4.440534e-01
## 11                                    Project.Tenure  1.070365e-02
## 12       Performance.RatingExceptional Performance -3.027602e-01
## 13              Performance.RatingGood Performance -5.017199e-02
## 14 Performance.RatingPerformance Needs Improvement  5.404647e-01
## 15         Performance.RatingThreshold Performance  3.411922e-01
## 16                      Performance.RatingUnrated  7.814482e-01
## 17                      Time.to.allocation.expiry -1.958148e-01
## 18                              Time.to.promotion  2.579562e-01
## 19                              Expat.Locallocal -4.559630e-01
## 20                                    Visa.TypeH-1  2.345762e+00
## 21                                    Visa.TypeL-1  1.582788e+00
## 22                                 Visa.TypeOthers  2.207298e-01
## 23                                    Time.to.visa  3.010423e-05
##       std.error   statistic      p.value
## 1  1.282796e+00  -1.9349032 5.300218e-02
## 2  6.469766e-03  -4.3668584 1.260464e-05
## 3  1.435318e-01   0.7382676 4.603519e-01
## 4  1.140353e+00   0.9368392 3.488413e-01
## 5  1.145127e+00   1.0749079 2.824160e-01
## 6  1.162849e+00   1.3490815 1.773108e-01
## 7  2.852542e-02  -6.4055379 1.498402e-10
## 8  3.522606e-06  -4.9755076 6.507689e-07
## 9  1.434989e-01  -1.2558332 2.091764e-01
## 10 1.775637e-01  -2.5008116 1.239091e-02
## 11 4.533386e-03   2.3610717 1.822221e-02
## 12 2.082818e-01  -1.4536087 1.460548e-01
## 13 1.902788e-01  -0.2636762 7.920295e-01
## 14 3.872834e-01   1.3955279 1.628567e-01
## 15 2.169470e-01   1.5726983 1.157887e-01
## 16 2.353723e-01   3.3200513 9.000089e-04
## 17 1.592727e-02 -12.2943130 9.717946e-35
## 18 3.517933e-02   7.3326072 2.257177e-13
## 19 2.269156e-01  -2.0093944 4.449533e-02
## 20 5.058773e-01   4.6370173 3.534729e-06
## 21 5.492849e-01   2.8815422 3.957343e-03
## 22 2.814904e-01   0.7841468 4.329540e-01
## 23 4.905281e-06   6.1371062 8.403823e-10
```

3.1 Regression coefficients

The output of the logistic regression object, mod_1, shows that categorical features, 'Expat.Local' and 'Visa.Type', and the numerical features 'Age', 'Tenure','CTC', 'Project.Tenure', 'Time.to.allocation.expiry', 'Time.to.promotion' and 'Time.to.visa' are significant features for predicting active Status outcome at alpha = 0.05 level. The rest of the model coefficients suggests insignificant contribution in status prediction. For example, increase one unit in age will decrease the log odd of status by 0.028; being on H1 visa will increase the log odd of inactive status by 2.35 compared to GC/Citizens.

The rest of the model coefficients suggests insignificant contribution in status prediction. These features are: *Gender, Band, Performance.Rating, L1 (except Infra) and VisaType.Others.*

3.2 Test individual features - Wald Test

Wald test on features coming as insignificant shows high p-values for Gender and Band thus confirming insignificance but Performance.Rating, L1 and Visa.Type are coming as significant with p value <0.05.

Further looking at the Performance.Rating coefficients, it is observed that only 'Unrated' value is significant. This is not relevant and we can ignore Performance.Rating from the model for now.

```r
varImp(mod_1)
```

```
##                                                      Overall
## Age                                                4.3668584
## GenderMale                                         0.7382676
## BandE1                                             0.9368392
## BandE2                                           1.0749079
## BandE3                                           1.3490815
## Tenure                                           6.4055379
## CTC                                              4.9755076
## L1ERS                                            1.2558332
## L1Infra                                          2.5008116
## Project.Tenure                                   2.3610717
## Performance.RatingExceptional Performance        1.4536087
## Performance.RatingGood Performance               0.2636762
## Performance.RatingPerformance Needs Improvement  1.3955279
## Performance.RatingThreshold Performance          1.5726983
## Performance.RatingUnrated                        3.3200513
## Time.to.allocation.expiry                       12.2943130
## Time.to.promotion                                7.3326072
## Expat.Locallocal                                 2.0093944
## Visa.TypeH-1                                     4.6370173
## Visa.TypeL-1                                     2.8815422
## Visa.TypeOthers                                  0.7841468
## Time.to.visa                                     6.1371062
```

```r
regTermTest(mod_1, "Gender")
```

```
## Wald test for Gender
##  in glm(formula = Status ~ Age + Gender + Band + Tenure + CTC + L1 +
##     Project.Tenure + Performance.Rating + Time.to.allocation.expiry +
##     Time.to.promotion + Expat.Local + Visa.Type + Time.to.visa,
##     family = "binomial", data = train_set)
## F =  0.545039  on  1  and  5106  df: p= 0.46039
```

```r
regTermTest(mod_1, "Band")
```

```
## Wald test for Band
##  in glm(formula = Status ~ Age + Gender + Band + Tenure + CTC + L1 +
##     Project.Tenure + Performance.Rating + Time.to.allocation.expiry +
##     Time.to.promotion + Expat.Local + Visa.Type + Time.to.visa,
##     family = "binomial", data = train_set)
## F =  2.037454  on  3  and  5106  df: p= 0.10641
```

```r
regTermTest(mod_1, "L1")
```

```
## Wald test for L1
##  in glm(formula = Status ~ Age + Gender + Band + Tenure + CTC + L1 +
##     Project.Tenure + Performance.Rating + Time.to.allocation.expiry +
##     Time.to.promotion + Expat.Local + Visa.Type + Time.to.visa,
##     family = "binomial", data = train_set)
## F =  3.131166  on  2  and  5106  df: p= 0.043751
```

```r
regTermTest(mod_1, "Performance.Rating")
```

```
## Wald test for Performance.Rating
##  in glm(formula = Status ~ Age + Gender + Band + Tenure + CTC + L1 +
##     Project.Tenure + Performance.Rating + Time.to.allocation.expiry +
##     Time.to.promotion + Expat.Local + Visa.Type + Time.to.visa,
##     family = "binomial", data = train_set)
## F =  7.263521  on  5  and  5106  df: p= 8.6634e-07
```

```
regTermTest(mod_1, "Visa.Type")
```

```
## Wald test for Visa.Type
##  in glm(formula = Status ~ Age + Gender + Band + Tenure + CTC + L1 +
##     Project.Tenure + Performance.Rating + Time.to.allocation.expiry +
##     Time.to.promotion + Expat.Local + Visa.Type + Time.to.visa,
##     family = "binomial", data = train_set)
## F =  7.832597  on  3  and  5106  df: p= 3.2596e-05
```

```
regTermTest(mod_1, "Project.Tenure")
```

```
## Wald test for Project.Tenure
##  in glm(formula = Status ~ Age + Gender + Band + Tenure + CTC + L1 +
##     Project.Tenure + Performance.Rating + Time.to.allocation.expiry +
##     Time.to.promotion + Expat.Local + Visa.Type + Time.to.visa,
##     family = "binomial", data = train_set)
## F =  5.57466  on  1  and  5106  df: p= 0.01826
```

3.3 Refitting Logistic Model with significant features

Gender, Band and Performance Rating are removed from model.

```
# Fit a logistic regression model with intercept only
mod_2 <- glm(Status ~ Age + Tenure + CTC + L1 + Project.Tenure + Time.to.allocation.expiry + Time.to.pro

# review the features and coefficients
tidy(mod_2)
```

```
##                         term      estimate    std.error   statistic
## 1                (Intercept) -1.754777e+00 5.839186e-01  -3.0051743
## 2                        Age -2.459033e-02 6.286587e-03  -3.9115551
## 3                     Tenure -1.869597e-01 2.833076e-02  -6.5991780
## 4                        CTC -1.171379e-05 2.460074e-06  -4.7615608
## 5                      L1ERS -2.125478e-01 1.419748e-01  -1.4970819
## 6                    L1Infra -2.528573e-01 1.647535e-01  -1.5347620
## 7             Project.Tenure  6.796623e-03 4.425072e-03   1.5359351
## 8  Time.to.allocation.expiry -1.962489e-01 1.584929e-02 -12.3821844
## 9          Time.to.promotion  2.789923e-01 3.338695e-02   8.3563272
## 10          Expat.Locallocal -4.231619e-01 2.219319e-01  -1.9067195
## 11              Visa.TypeH-1  2.356859e+00 5.003472e-01   4.7104479
## 12              Visa.TypeL-1  1.664480e+00 5.450597e-01   3.0537578
## 13           Visa.TypeOthers  1.598558e-01 2.744712e-01   0.5824136
## 14              Time.to.visa  3.058586e-05 4.857120e-06   6.2971186
##         p.value
## 1  2.654287e-03
## 2  9.170372e-05
## 3  4.134436e-11
## 4  1.921014e-06
## 5  1.343720e-01
## 6  1.248423e-01
## 7  1.245543e-01
## 8  3.263381e-35
## 9  6.470108e-17
## 10 5.655693e-02
## 11 2.471730e-06
## 12 2.259944e-03
## 13 5.602881e-01
## 14 3.032293e-10
```

---

4. Model Performance Evaluation

Applying the logistic regression model object and fitting all independent features of the tested dataset in the model. The test_logit is a vector that holds the predicted status outcomes of employees and we are adding the vector to test data set. The first few probability of Status are shown below.

```
#Use predict function to generate predictions for the testing set and add the predictions in test data
test_logit <- predict(mod_2, newdata = test_set, type = 'response')
test_set <- test_set %>% mutate(test_logit = test_logit)
head(test_set$test_logit)
```

```
##           1           2           3           4           5           6
## 0.001014978 0.008101578 0.114616392 0.115587854 0.091239540 0.006052766
```

The model has low probability values due to unbalanced data. We need to identify optimal cut-off value to predict 0/ Active and 1/ Inactive status so that miss classification error is minimized.

Lets look at ROC curve for getting to desired cut-off value.

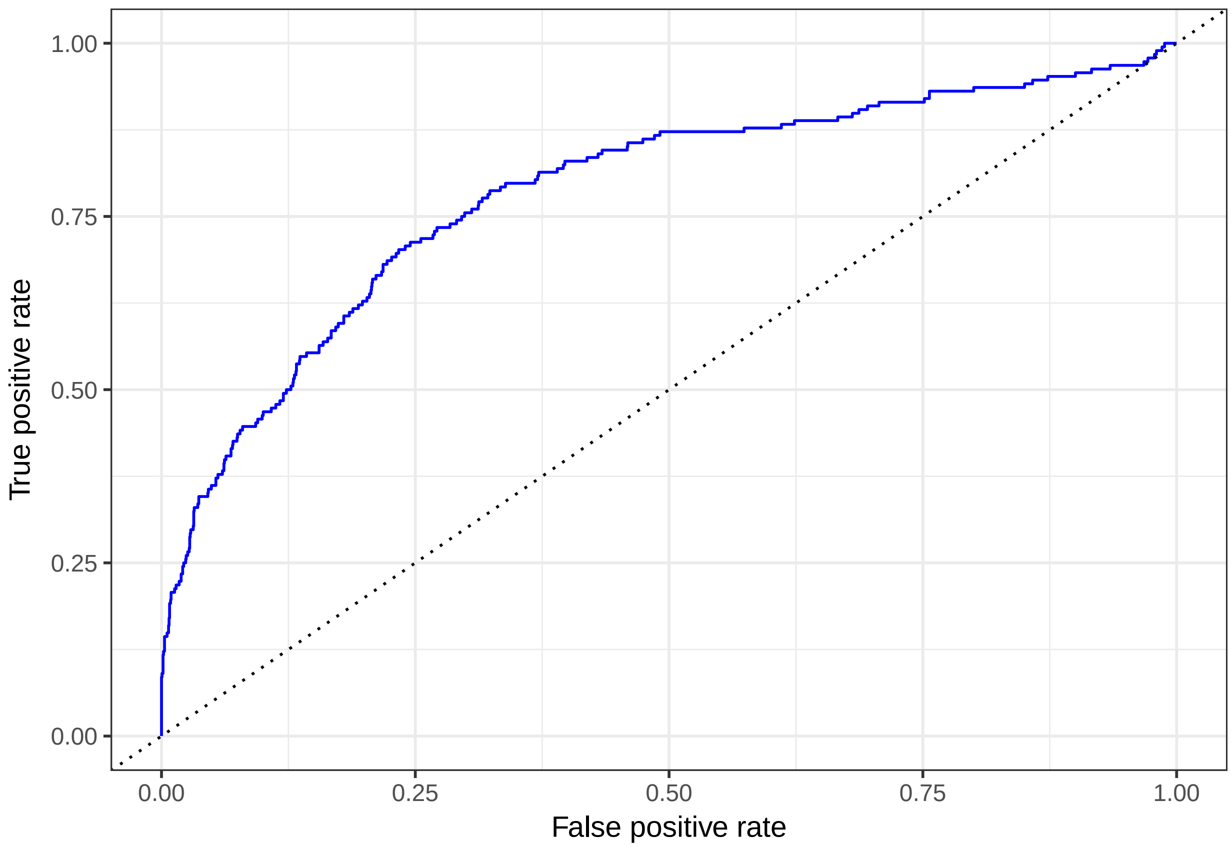
4.1 ROC curve plot

```
#create the prediction objects
pred_logit <- prediction(predictions = test_logit, labels = test_set$Status)

# get the FPR and TPR for the logistic model
perf_logit <- performance(pred_logit, measure = 'tpr', x.measure = 'fpr')
perf_logit_tbl <- tibble(perf_logit@x.values[[1]], perf_logit@y.values[[1]])

# Change the names of the columns of the tibble
names(perf_logit_tbl) <- c('fpr', 'tpr')

# Plotting function for plotting a ROC curve using ggplot
plot_roc <- function(perf_tbl) {
p <- ggplot(data = perf_tbl, aes(x = fpr, y = tpr)) +
geom_line(color = 'blue') +
geom_abline(intercept = 0, slope = 1, lty = 3) +
labs(x = 'False positive rate', y = 'True positive rate') +
theme_bw()
return(p)
}
# Create the ROC curves using the function above
plot_roc(perf_logit_tbl)
```



4.2 AUC

```
# calculate the AUC
auc_logit <- performance(pred_logit, measure = 'auc')

# extract the AUC value
auc_logit@y.values[[1]]
```

```
## [1] 0.7839966
```

The ROC curve and AUC of greater than 0.7 reveal that model is average fit to data.

4.3 Confusion Matrix

As per ROC curve, will take TPR (true positive rate) greater than 0.75 and FPR (false positive rate) of less than 0.4. The corresponding probability cutoff for classifying the prediction is calculated as below.

```
#create a tibble of fpr, tpr and cutoffs
perf_logit_tbl <- tibble(fpr = perf_logit@x.values[[1]], tpr = perf_logit@y.values[[1]], cutoffs = perf

perf_logit_tbl %>% filter(fpr < .375 & tpr > .80)
```

```
## # A tibble: 16 x 3
##      fpr   tpr cutoffs
##    <dbl> <dbl>   <dbl>
##  1 0.368 0.803  0.0795
##  2 0.369 0.803  0.0794
##  3 0.369 0.803  0.0793
##  4 0.370 0.803  0.0791
##  5 0.370 0.803  0.0788
##  6 0.371 0.803  0.0787
##  7 0.371 0.809  0.0786
##  8 0.371 0.809  0.0786
##  9 0.372 0.809  0.0786
## 10 0.372 0.814  0.0785
## 11 0.372 0.814  0.0784
## 12 0.373 0.814  0.0779
## 13 0.373 0.814  0.0779
## 14 0.374 0.814  0.0778
## 15 0.374 0.814  0.0776
## 16 0.375 0.814  0.0776
```

```
# Taking cutoff value
p_cutoff_logit <- 0.07846262

#Assign the prediction outcome using above cut-off value
test_set <- test_set %>% mutate(class_logit = case_when(
  test_logit < p_cutoff_logit ~ 'Active',
  test_logit >= p_cutoff_logit ~ 'Inactive')) %>% mutate(class_logit = as.factor(class_logit))

#Print confusion matrix
(cnf_mtrx <- test_set %>% count(class_logit, Status) %>% spread(Status, n))
```

```
## # A tibble: 3 x 3
##   class_logit Active Inactive
##   <fct>        <int>    <int>
## 1 Active        1267       36
## 2 Inactive       749      152
## 3 <NA>            NA        8
```

4.4 Performance against business target

RECAP: Model target as agreed with business

(1) Model should accurately predict 90% of actual attrites as high risk

(2) Precision (True positives & true negatives) should be greater than 85%

Lets calculate the performance against these measures.

4.4.1 Hit rate - Actual inactive predicted correctly

Total inactive employees in test set are 196 and correctly predicted as per confusion matrix are 152.

Inactive prediction accuracy rate thus is 77.5% (target - 90%).

```
#count of actual active and inactive employees in test data
test_set %>% group_by(Status) %>% count()
```

```
## # A tibble: 2 x 2
## # Groups:   Status [2]
##   Status       n
##   <fct>    <int>
## 1 Active    2016
## 2 Inactive   196
```

4.4.2 Precision - True positives & true negatives

Our precision rate as calculated from test_set is 64% as compared to the target of 80%.

```
correct_predic = test_set %>% filter(class_logit == Status) %>% count()

total_n = test_set%>% count()
(Precision = correct_predic / total_n * 100)
```

```
##          n
## 1 64.15009
```

---

5 Summary

Logistic regression model is fitted on employee attrition data to predict the employees at high risk of leaving. The binary logistic regression is first performed with the glm function and the model is evaluated on different parameters. We have obtained AUC value of less than 0.80 and prediction accuracy short of business target by close to 15%. There is scope to improve the performance further through alternate models and feature selection. The model should be revisited after feature selection and data sampling is updated.