Data Science Accelerator for Credit Risk Prediction

Fang Zhou, Data Scientist; Graham Williams, Director of Data Science, all at Microsoft

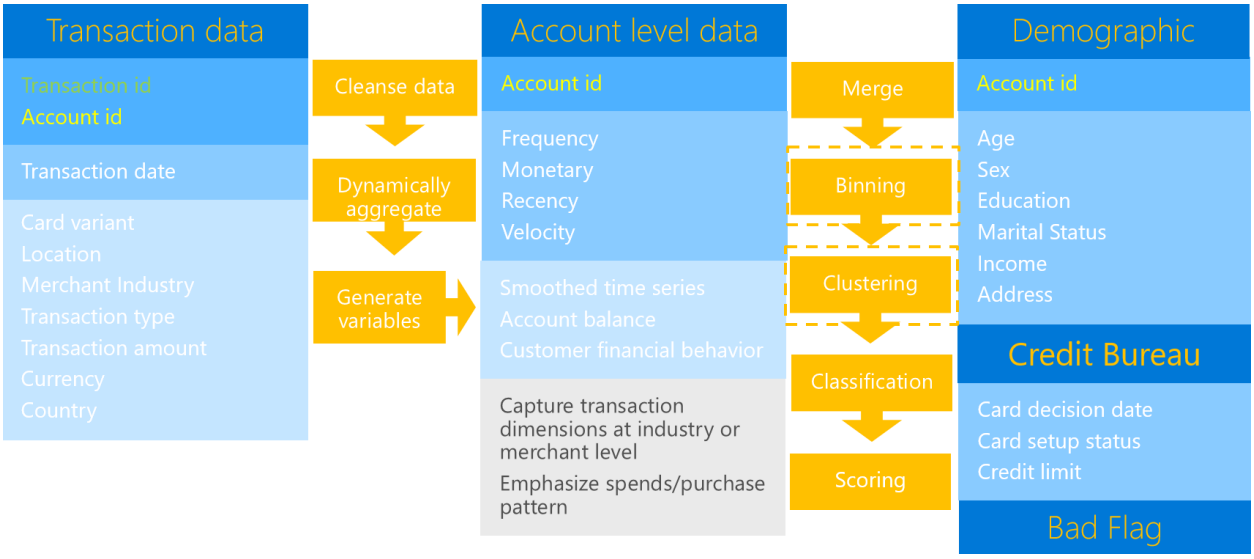
Credit Risk Scoring is a classic but increasingly important operation in banking as banks are becoming far more risk careful when lending for mortgages, credit card payments or other commercial purposes, in an industry known for fierce competition and the global financial crisis. With an accurate credit risk scoring model, a bank is able to predict the likelihood of default on a transaction. This will in turn help evaluate the potential risk posed by lending money to consumers and to mitigate losses due to bad debt, as well as determine who qualifies for a loan, at what interest rate, and what credit limits, and even determine which customers are likely to bring in the most revenue through a variety of products.

Many banks nowadays are driving innovation to enhance risk management. For example, a largest bank in one of the Asian countries by market capitalization is exploring opportunities to segment a million of active credit card customer population to improve risk scoring to then identify opportunities to offer increased limits. Using advanced analytics for credit risk scoring involves traditional scorecard building and modelling, and extends to machine learning and ensemble, but will also pursue an innovation on customer oriented aggregation of transactions, multi-dimensional customer segmentation and conceptual clustering to identify multiple segments across which to understand bank customers.

In the data-driven credit risk prediction model, normally two types of data are taken into consideration.

1. **Transaction data** Transaction records cover transaction id, account id, transaction date, transaction amount, merchant industry, etc. This transaction-level data could be dynamically aggregated and then provide transaction statistics and financial behavior information at account level.
2. **Demographic and bank account information** This type of data show the characteristics of individual customer or account credit bureau, such as age, sex, income, and credit limit. They are static and never change or solely increment deterministically over time.

The following graph shows the data schema and the workflow for credit card fraud risk prediction.



In recent years, R has been gaining in popularity over SAS among statisticians and data scientists in solving variety of industrial business problems, including Financial Services & Risk Management. A data science accelerator for credit risk prediction is now shared in the github repository. This accelerator consists of four R templates which walk through the process of model development, scale-up and speed-up, deployment, and application development.

- **CreditRiskPrediction** Data-driven credit risk prediction in R, covering techniques of exploratory analytics, data aggregation, merging and cleansing, feature engineering, but more importantly, model building and evaluation.

```
library(xgboost)
library(randomForest)
library(caret)
library(caretEnsemble)

tc <- trainControl(method="boot",
                   number=5,
                   repeats=3,
                   search="grid",
                   classProbs=TRUE,
                   savePredictions="final",
                   summaryFunction=twoClassSummary,
                   allowParallel=TRUE)

model_list <- caretList(bad_flag ~ .,
                        data=data[train, c(target, vars)],
                        trControl=tc,
                        methodList=c("xgbTree", "rf"))

model_ensemble <- caretEnsemble(model_list,
                                metric="ROC",
                                trControl=tc)

varImp(model_ensemble)
```

- **CreditRiskScale** Faster and scalable credit risk models with Microsoft R Server , using the state-of-the-art machine learning algorithms provided by the MicrosoftML package.

```
library(MicrosoftML)

model_rxtrees <- rxFastTrees(
    formula=form,
    data=data_split[[1]],
    type="binary",
    numTrees=100,
    numLeaves=20,
    learningRate=0.2,
    minSplit=10,
    unbalancedSets=FALSE,
    verbose=0
  )

model_ensemble <- rxEnsemble(
    formula=form,
    data=data_split[[1]],
    type="binary",
    trainers=list(fastTrees(),
                  fastTrees(numTrees=500),
                  fastTrees(numTrees=500, learningRate=0.3),
                  fastTrees(numTrees=500, learningRate=0.3, unbalancedSets=TRUE)),
    combineMethod="vote",
    replace=TRUE,
    verbose=0
  )
```

- **CreditRiskScale** Train multiple ML models with hyper-parameter selection in parallel by using rxExec().

```
models <- list(name=c("rxLogisticRegression", "rxFastForest", "rxFastTrees"),
               para=list(list(list(l1Weight=0,
                                   l2Weight=0),
                              list(l1Weight=0.5,
                                   l2Weight=0.5),
                              list(l1Weight=1,
                                   l2Weight=1)),
                         list(list(numTrees=50,
                                   numLeaves=15,
                                   minSplit=10),
                              list(numTrees=100,
                                   numLeaves=20,
                                   minSplit=10),
                              list(numTrees=500,
                                   numLeaves=25,
                                   minSplit=10)),
                         list(list(numTrees=50,
                                   learningRate=0.1,
                                   unbalancedSet=FALSE),
                              list(numTrees=100,
                                   learningRate=0.2,
                                   unbalancedSet=FALSE),
                              list(numTrees=500,
                                   learningRate=0.3,
                                   unbalancedSet=FALSE))
                         ))

trainModel <- function(formula, data, modelName, modelPara, algoIndex) {

    tunePara <- function(formula, data, modelName, modelPara) {
      model <- do.call(modelName, c(list(formula=formula,
```

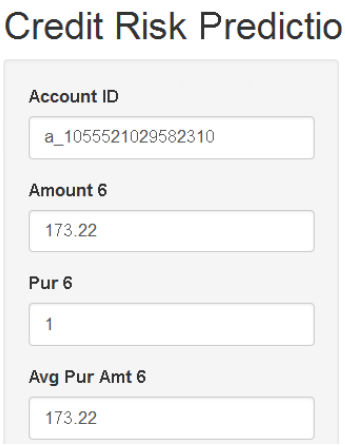
```
                                        data=data),
                                    modelPara))
      return(list(model=model))
    }

    output <- rxExec(tunePara,
                     formula=formula,
                     data=data,
                     modelName=modelName[algoIndex],
                     modelPara=rxElemArg(
                               modelPara[[which(modelName == modelName[algoIndex])]]))

    return(list(output=output))
  }

  rxSetComputeContext("localpar")

  result <- rxExec(trainModel,
                   formula=form,
                   data=data_train,
                   modelName=models$name,
                   modelPara=models$para,
                   algoIndex=rxElemArg(c(1:3)))
```

- **CreditRiskDeploy** Deploy a credit risk model as a web service with R Server Operationalization, leveraging the mrsdeploy package.

```
library(mrsdeploy)

creditRiskPrediction <- function(account_id, amount_6, pur_6, avg_pur_amt_6,
avg_interval_pur_6, credit_limit, marital_status, sex, education, income, age)
{
  newdata <- data.frame(account_id=account_id,
                        amount_6=amount_6,
                        pur_6=pur_6,
                        avg_pur_amt_6=avg_pur_amt_6,
                        avg_interval_pur_6=avg_interval_pur_6,
                        credit_limit=credit_limit,
                        marital_status=marital_status,
                        sex=sex,
                        education=education,
                        income=income,
                        age=age)

  pred <- rxPredict(modelObject=model_rxtrees, data=newdata)[, c(1, 3)]
  pred <- cbind(newdata$account_id, pred)
  names(pred) <- c("account_id", "scored_label", "scored_prob")
  pred
}

api <- publishService(
       "crpService",
        code=creditRiskPrediction,
        model=model_rxtrees,
        inputs=list(account_id="character",
                    amount_6="numeric",
                    pur_6="numeric",
                    avg_pur_amt_6="numeric",
                    avg_interval_pur_6="numeric",
                    credit_limit="numeric",
```

```
                marital_status="character",
                sex="character",
                education="character",
                income="numeric",
                age="numeric"),
    outputs=list(pred="data.frame"),
    v="v1.0.0")
```

- **CreditRiskShinyApp** Credit risk application through REST API, with integration with the *Shiny* Framework.




```
          account_id scored_label        scored_prob
1 a_1055521029582310           no 0.0156644005328417
```

Complete version of [the data science accelerator for credit risk prediction] is ready for share in this [github repository]. Code for these templates are displayed in R markdown files which can yield files of various formats, such as PDF, html, Jupyter Notebook, etc. By no means is this the endpoint of the data science journey. The accelerator is under regular revision and improvement and is provided as-is.