**Student Name: Chengjun Liu**

**Problem Description**

Our goal is to build Anti Money Laundering system based on pure statistic algorithms for Bank of America.

It is known that expert rule system is unable to reduce False Positive and improve efficiency, so improving AML with pure statistical models saves costs. The AML watches over business and consumer customers’ DDA accounts, monitoring the information and transactions with other accounts and linked credit cards, plus identity information such as customer/business name, SSN, address and more.

There are a few points to pay attention: We are mainly concerned about Fronting (processing money transactions under the disguise of legitimate business transactions) and Structuring (breaking money into small amount transactions into several different accounts). On the one hand, based on only internal account transactions we cannot investigate on whether goods are transacted or legally transacted, which hinders investigation of Fronting. On the other hand, because it is already required that transactions over $10,000 should be reported to US government, any large money laundering must be processed in fractions, which means many of Fronting laundering are also processed via Structuring. Because Structuring requires more frequent transactions between different accounts, it is easier to be detected using internal transaction records. Another thing is that our statistical models do not need case management, so our models are trained and applied purely by available transaction records.

Supervised or Unsupervised?

Our first question to investigate models is to decide whether it is a supervised or unsupervised question. We judge that supervised models need expert labeled records, which can make models learn and incorporate effective expert rules. Meanwhile, we think that unsupervised models can detect deviance for unchecked transaction records, which supplements supervised models. It is understandable that in real practice supervised models are limited by limited training samples, so applying only unsupervised model can detect outliers which are more possible to be money laundering. However, we expect supervised model can reliably improve the accuracy using history cases. As a result, we intend to combine supervised and unsupervised models as the final score to measure fraud if supervised models with expert-ruled samples are applicable, or we apply unsupervised models.

**Solution Approach**

Step 1: Request of data volume, features, and algorithm update frequency

The models are built to be available up to date, so we need to run algorithms on fixed time intervals. Because the strong need of records, relatively low frequency of money laundering audit in a year, and enough evidence of frequent transaction, we tend to set time interval to a longer period. Here we take records in 1 year as the window the algorithm runs on, and renew our algorithm for every 3 months, which means batch in 3 months. For available features, we try request existing and additional information based on the following 3 types:

1. Background information about sending and receiving parties: indicator of bankruptcy, customer/business name, SSN/TIN, address and contact information, type of identity and customer relationships (individual & corporation), sex & age for individuals, nationality, activity level, years of establishment, and industry type for corporations.
2. Transaction history for sending and receiving parties: balances, payments, transfers for all DDA accounts and linked credit cards, with more information such as type of currencies, type of transactions.
3. Alert or expert marked money laundering label: including whether the suspicious records are reported and marked as fraud.

Those information request does not guarantee all features can be provided, but the list makes it clear what variables we are expecting.

Regarding the labels of money laundering, we are not going to replace expert rule system but enhance the system. We still would like to get labels from expert rule system and alert report system. Referring to the cases in other money laundering reporting system like Norway, we assume that usually daily money laundering case report can be over hundreds, and we estimate around 10,000 labeled records for the rolling window of a year can support the supervised system. Unsupervised models are not restricted by training samples and can utilize all transaction records.

Step 2: Data Cleaning

After getting requested data, we conduct data cleaning in following ways:

1. For numeric missing variables, we intend to fill the values using mean or median in different groups, such as groups by sending and receiving parties, identities like nationality and industries. We can also combine KNN filling method based on groups. We make sure the groups are properly built and we balance each group to contain at least 5 records. For categorical missing values, we would not like to alter them because missing information is suspicious, and we fill “N” instead.
2. For invalid values like 170 years in age, we should apply corresponding logic. For numeric variables, we apply capping on 99% or specify numerical range to filter abnormal values. For categorical values, we check the distribution of the values and find values without regular patterns. Because of the existence of similar transactions in parties, identities, etc., we can verify the values by grouping similar transactions and fill the invalid values by deduction from other information of the same group.
3. For special cases, we would remove transactions over $10,000 to focus on small amount transactions. Meanwhile, we check the identity of parties and transaction types and remove certain parties and transactions. For example, we do not intend to check government parties or other well-known parties that have a lot of accounts for real business, or we are only concerned about certain type of transactions.

Step 3: Data Linking and Feature Construction

After data cleaning, we build our dataset based on transactions and focus on the summary of identity and transaction records of entities. For types of entities, we are concerned about both individual and corporation because money laundering can be conducted in both kinds of entities. For corporations, we pay attention to the frequency of transactions, intervals between receiving and sending, account savings, and more. For individuals, we pay attention to the number of accounts and linked credit cards, number of customers connected via transactions and time length of relationships, total amount of transactions, and more.

For data linking, we would like to group up available identity information to set unique identity for parties and conduct data aggregation for new variables. Based on various identity information of name, SSN, address and more, we can come up with various identity combination, like name-SSN, name-address, name-contact\_information, name-identity-industries, and more, based on the judgement of unique identity.

We maintain existing variables and create new variables based on aggregation of entities, and we consider 4 different types of explanatory variables:

1. Background Information: indicator of bankruptcies, the number of years since the last one, number of years since relationship establishment, type & number of customer relationships towards both corporation and individual, sex, age, nationality, activity level, number of years since corporation establishment, industry & sector type, etc.
2. All transaction history: maximum & total amount & number of transactions in different kinds of currencies and different kinds of transaction types like cash deposit, store purchase, salary, manual payment, etc.
3. Single transaction: the amount, currency and transaction type for one transaction.
4. Fraud alert: fraud label for single transaction, proportion of previous fraud transaction with party identity such as sender and receiver.

Step 4: Feature Selection and Data Split

For categorial variables, we use traditional solution of one-hot encoding to convert variables. Meanwhile, we remove non-informative variables like variables with only one single value across records to pre-compress the dimensions.

For further dimension reduction, it depends on the models we decide to apply. For supervised model, we propose using XGBoost, one of Gradient Boosting Tree algorithm highly recognized by data scientists. XGBoost can deal with feature scaling and selection. However, for unsupervised models, we propose Isolation Forest, which requires feature scaling and selection. Considering the conditions above, we would like to conduct feature engineering over transaction data.

For feature scaling, we suggest using Z-scale for large samples and distance measure. For feature selection, we suggest using PCA algorithm to find most important features influencing measure of distance from average or wrappers like RFECV for selection over possibly thousands of features.

For sample split, we split the dataset following transaction dates. Because rolling window is 3 months, we retail the latest records of 3 months as validation set and split the first 9 months into training and testing dataset in 80%:20% for model selection.

Step 5: Building models

For supervised models, we would like to use XGBoost directly for recognized reliability in various real-world problems, fast speed, and large-scale datasets. We would use all selected features to do grid-search over hyper-parameters over training dataset and verify both accuracy and Fraud Detection Rate to select best hyper-parameters. We use the best hyper-parameters to train the model on whole 9 months and validate out-of-date predictions over validation set to measure model’s availability to out-of-date records. We use all records to train our final mode to make predictions for new records and we retain fraud probability of records from XGBoost for ranking steps.

For unsupervised model, we would like to use Isolation Forest, which is easy to output scores, can deal with high dimensions, and often applied in fraud detection. By building isolation trees and finding the complexity of dividing certain points from others, we can judge whether points are outliers. We combine the history records with new records, guarantee enough isolation trees and calculate the complexity scores for new records, and we use the decision scores for ranking steps.

Step 6: Fraud Score and Application

We sort scores from both XGBoost and Isolation Forest and rank the records respectively by two scores. Scores from XGBoost should be ranked in descending order, making high possibility of fraud appear as smaller ranks. Scores from Isolation Forest should be ranked in ascending order, making low scores which means less splits appear as smaller ranks. We take the average of two rankings as the final fraud scores to judge whether the record is more like money laundering than other records, with lower values more likely to be money laundering. Based on estimated money laundering frequency, we concentrate on a small portion of small-score transactions as possible money laundering and take actions. Another available fraud score is to leverage on two scores using , where is random forest probability score, and is anomaly score of isolation forest. The alpha and criterion of judging records as money laundering should be evaluated by discretion.