4. Results and In-depth analysis using machine learning

Prior to employing machine learning models. We performed several feature engineering to the original data. First, we need to create more independent variables out of the current useful features, drug name and total sale. We expanded these two features by creating a matrix with providers as indices and each column of the row is a drug name with a value of sale portion of that drug. In this way, we construct independent features that reflect sale behavior of each provider. In should be noted that since there are more than 2 million providers. It does not make sense to use all drugs as a part of feature construction. Herein, we only used drugs that were sold (at least once) by fraud providers. This reduces the number of unique drugs from 2191 to 553 types. The example of the resulting table is shown in Table 2.

Table 2. Example of the features generated from drug names and its sale portion for different providers.

	ABILIFY	ACEBUTOLOL HCL	ACETAMINOPHEN-CODEINE	ACTONEL	ACYCLOVIR	ADEFOVIR DIPIVOXIL	...
npi							
1003000126	0	0	0	0	0	0	...
1003000142	0	0	0	0	0	0	...
1003000167	0	0	0.01	0	0	0	...
1003000407	0	0	0	0	0	0	...
1003000423	0	0	0	0	0	0	...

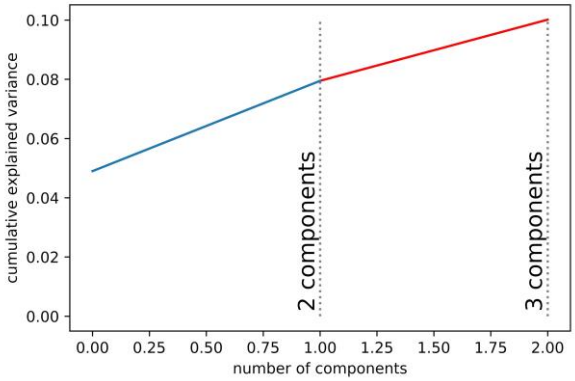

Fig. 9. Explained variance using 2 and 3 principal components.

As one might have expected, the generated features that consist of 553 columns are low-rank matrix and is filled mostly with zeros. To get rid of unnecessary features as well as avoiding the curse of dimensionalities, we applied principal component analysis (PCA) to the data. In Fig. 9, it is shown that by using 2 and 3 components, the variances explained are 8% and 10% respectively. This may sound like a small number, but consider there are more than 553 features, this means that the rest of the 550 features would contribute to less than 0.17% each. This implies that the rest of the components may be negligible.

To test if the engineered data could be better separated than the original ones as shown in Fig. 7 (right), scatter plots of the data with 2 and 3 components with label are shown in Fig. 10. In both of these figures, we can observe the agglomeration of the fraud providers, although not totally separated from typical providers. In the case where only 2 principal components were used, majority of the fraud providers exhibited PC2 lower than 0.1, with significant amount of those data points lying in a negative PC1 and negative PC2 region. Similarly, in the case of 3 components we can see that most fraud providers exhibited PC2 of lower than 0.2 and significant amount of those data points lying in a negative PC1 and negative PC3 region. This shows a promising result that machine learning models may be able to classify these points. However, judging from the shape and degree of overlaps of the data, it is expected that decision tree-based algorithms would do well in this situation. Not only because the data is highly non-linear, non-Gaussian, but the amount of red data point (fraud providers) is significantly less than the light blue ones (typical providers).

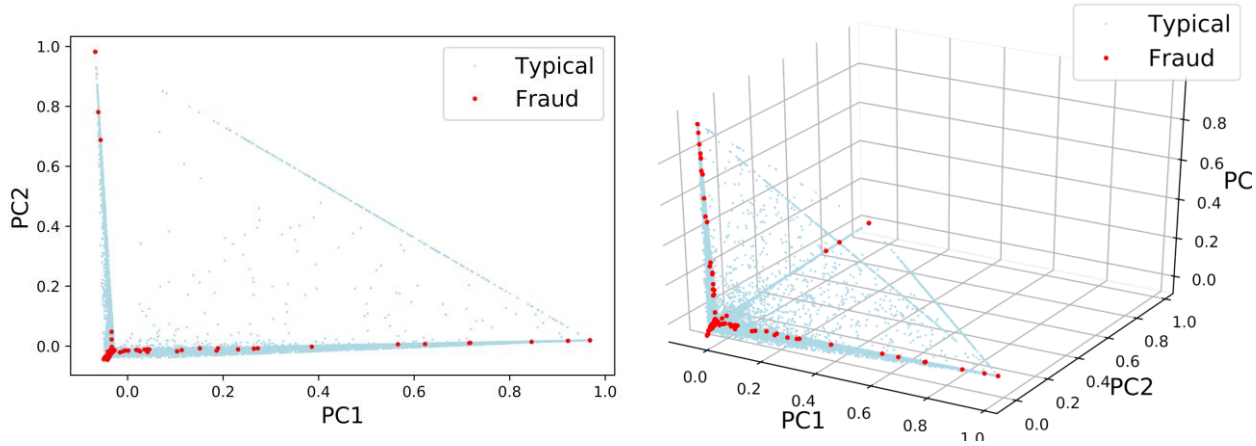

Fig. 10. Labelled scatter plot of typical (light blue) and fraud (red) providers plotted in its corresponding 2 principal components (left) or 3 principal components (right).

In summary, the number of typical providers is 486675 providers, as opposed to just 566 fraud providers. This would a heavy class imbalance for our classification model that even a tree-based algorithm may not be able to identify. To address this problem, we investigated several bootstrapping technique including up-sampling, down-sampling, and combination of both using an ensemble method, random forest with 50 trees, as a benchmarking algorithm. The bootstrapped samples will be separated 66/33 for training and test set. The results are then judged based on confusion matrices. It should be noted here that simple accuracy is not used as a scoring metric because even if the classification model predict that all providers are typical, the accuracy will still be above 98%. Using confusion matrix, which encompass both true/false positive and negative, enable us to clearly see where the model performs best and which needs to be improved.

Table 3. The confusion matrix results with precision and recall score for no and different types of bootstrapping methods. The results are obtained using random forest algorithm with 50 trees.

	No Resampling	**Up-sampling 10x**	**Down-sampling 10x**	**Combined**
Typical providers	486675	486675	48670	48670

Fraud providers	566		5480		566		5480	
Train	326078	0	326054	3	32608	0	32553	76
	116	257	757	2929	93	287	635	3016
Prec.	1		0.9999		1		0.9754	
Recall	0.6890		0.7946		0.7553		0.8261	
Test	160596	1	160586	32	16055	7	15924	117
	193	0	415	1379	185	1	314	1515
Prec.	0		0.9773		0.1250		0.9283	
Recall	0		0.7687		0		0.8283	

From Table 2, we can see that bootstrapping to increase the number of fraud providers is an important step. On the major column, we can see that with no resampling at all, the random forest algorithm can only capture the fraud provider in the training set, while it utterly failed to detect a single fraud provider in the test set. By up-sampling the fraud providers by about 10 folds, the algorithm performed significantly better, with a very precision of 98% and satisfactory recall of 77% in the test set, which is also very close to the training set results. However, down sampling data by ten times (down-sampling the number of typical providers) did not yield improved results. On the contrary, the outcomes are slightly below no sampling results. It should be noted here that down-sampling further to 100 folds were investigated and the results exacerbate. Lastly, the results from combination of these two techniques showed an improved in recall, but lower precision compared to up-sampling only method.

For the present study, we favor a conservative approach and choose to perform only up-sampling. We chose to maintain high-precision over improved recall because we want to be able to identify with high precision which providers are fraud. Moreover, lower precision means

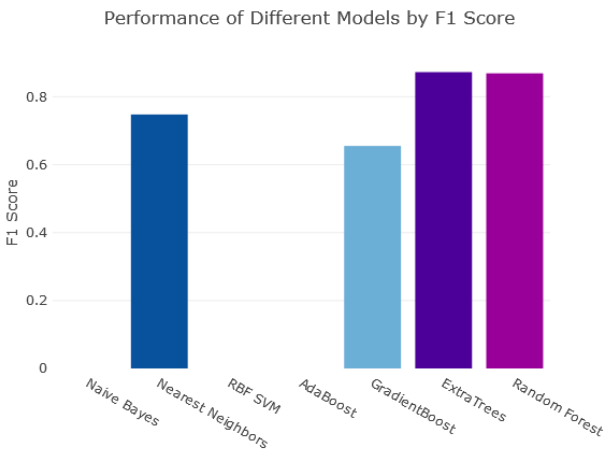

Fig. 11. F1 score from different classification models.

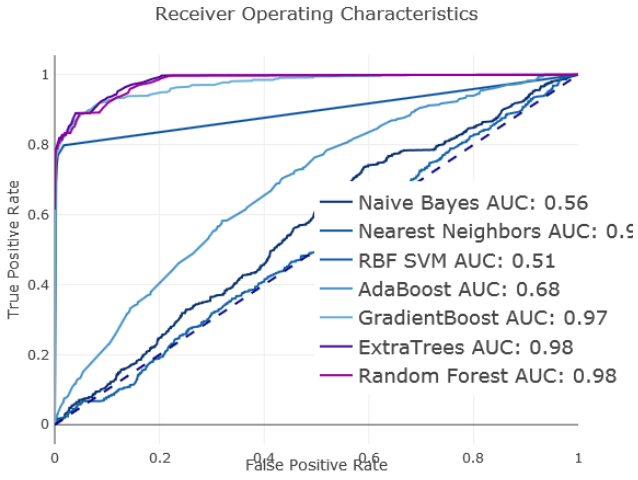

Fig. 12. Receiver operating characteristics (ROC) and area under curve (AUC) of each classification model.

significant more numbers of typical providers will be classified as fraud providers since there are way more of them than fraud providers. This would make verification process troublesome.

We first use data with 2 principal components and 5 fold cross validations to screen for the potential classifiers. In this step we used two criteria to judge the performance of classification models. These include F1 score and area under curve (AUC) of the receiver operating characteristics (ROC). Both of which, unlike accuracy, are metrics that take into account the effect of true/false positive and negative results. F1 score is defined as 2*(precision * recall) / (precision + recall), whereas ROC is the curve between false positive rate and true positive rate. Therefore, AUC is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one – the higher the better.

The results in Fig. 11 and Fig. 12 showed that random forest and extra trees algorithm performed the best among all studied models including naïve Bayes, K nearest neighbors, Gaussian SVC, Adaboost (decision tree base classifier), gradient boosting. Naïve Bayes, SVC, and Adaboost were not able to capture recall and hence the F1 score becomes zero (Fig. 11). Note that most models that yielded descent scores are based on "decision tree algorithms". This is because these trees are able to cope with class imbalance better by their hierarchical structure that allows them to learn signals from both classes given sufficient branches.[2] The results showed that extra trees method is slightly better than random forest with F1 and AUC of 0.873 and 0.98 versus 0.869 and 0.98 from random forest. Here, it is worth mentioning the difference between these two methods. Random forest is a commonly used ensemble method for many applications and it functions by, as the name suggest, having numbers of trees each takes in a different number of features and vote. The process is repeated with bootstrap sampling so that each tree is exposed to more data. Extra trees, short from extremely randomized tree, drops the idea of using bootstrapping copies, and with the split chosen at random from the range of values in the sample at each split, resulting in higher number of leaves. It is expected that extra trees tend to perform worse when there is a high number of noisy features (in high dimensional data-sets).[3]

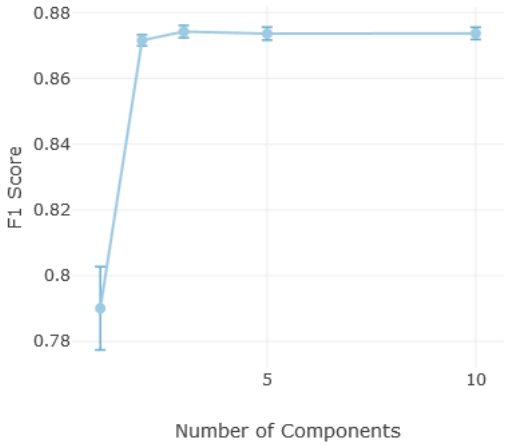

Fig. 13. F1 score of optimized extra tree model with increased number of principal components.

To further increase the performance, we optimized the extra trees method using grid search over different parameters including number of trees, number of features in each tree,

number of samples required to split nodes. The optimized model required up to 200 trees instead of just 50 trees in the original extra tree model. Lastly, the results were optimized by increasing the number of principal components. We can see that just 3 components are sufficient to obtain an F1 score of 0.874 (with 0.986 precision and 0.784 recall), which is considerably high.

For future studies, there are several ways that could be used to improve the performance of the model. For example, by incorporating Medicare Provider Utilization and Payment Data (https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports /Medicare-Provider-Charge-Data/Part-D-Prescriber.html), which contains information about which providers were paid by which drug companies and by how much in order to promote their drug sales. This is required by law. Unfortunately, the identifier provided in the payment dataset is purposefully altered so that one could not match with the identifier in the Medicare and exclusion dataset (npi). Although there are venues that one could back track these numbers, the process is heavily time consuming and potentially illegal. Therefore, this study did not pursue the use of payment dataset to add more features to it. Applying graph theory to these would likely yield very interesting insight as well.