STRATEGIES FOR OPTIMIZING SALES TAX AUDIT RESOURCES

Ying Du, Michael Sherman, Nicole White

2013

EXECUTIVE SUMMARY

Two data-mining based approaches are discussed as possible strategies to increase sales tax audit efficacy. The first strategy uses a tree of classification rules to find returns most likely to be fraudulent, based on past patterns of known fraudulent claims. This tree was evaluated based on past audited claims, and was found to be more accurate than random audits. The second strategy predicts the expected revenue from auditing a return based on past tax collections from successful audits. This strategy was evaluated based on past data of successful and unsuccessful audits, and was found to produce audit profits significantly greater than auditing returns at random. Both strategies are ready for implementation in the field.

Introduction to Business Data Analytics
MS IROM: Business Analytics
October 16th, 2013

INTRODUCTION

The state has found that many businesses file improper sales tax returns. These fraudulent returns deprive the state of taxation income, and create an atmosphere among businesses that tax compliance is not important. To address this problem, the state is seeking a data-driven solution to the problem of how to best utilize its limited audit resources to enforce sales tax compliance and increase taxation revenues.

Thus, the state has two primary goals. The first goal is to audit as many fraudulent returns as possible. This goal is not concerned with the potential income the state may collect, it is only concerned with the sheer count of fraudulent returns. The second goal is to maximize the state's income from auditing fraudulent returns. For this second goal, the number of returns accurately audited is not as important as the income the state generates. However, every audit (in this case desk audit) has a cost. We will consider two scenarios: one where a desk audit costs the state $8,000, and another where a desk audit costs $15,000.

To help fulfill these goals, we have induced and evaluated two predictive, data-driven models.

MAXIMIZING THE NUMBER OF SUCCESSFUL AUDITS

In order to target as many non-compliant returns as possible, we need to find out which returns are most likely to be non-compliant. This is done by developing a model to predict the probability of fraud based on characteristics of the filing company and the tax return. These characteristics are aggregated, and we create a model by discovering patterns in the data that are more likely to appear in fraudulent returns vs. compliant returns.

The given dataset of returns includes both the success/failure of a past audit, as well as the revenue collected from the audit (if any). If we use the original data without cleaning, the REVENUE attribute will become the single predictor of the model. So REVENUE should be taken out of the dataset when inducing this first predictive model, since we are only concerned with the presence of fraud, not the extent.

In order to predict the probability of fraud, we used a J48 model induction algorithm with 10-fold cross validation. This induced a complicated classification tree model when we used all attributes from the dataset (except for REVENUE). We also noticed some attributes are derived from other attributes—for instance LG_<REDACTED> comes from <REDACTED>. After sorting the attributes by their information gain, we saw that the information gains of the derived attributes (the log transformations) are significantly higher than the info gains of the attributes from which they are derived. We also noticed that some attributes' info gains are very low, which means such attributes add unnecessary complexity to the model while not significantly adding to the model's accuracy. Based on this analysis, we hypothesized that we can induce a model without such low info gain attributes and that such a model will not be significantly worse that the model induced with all attributes. Thus, after eliminating unnecessary attributes, we decided to keep the following fields: <REDACTED>.

Using these, we created a J48 classification tree using 10-fold cross validation. The output measurements and a diagram of the tree (next page):

=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.1 -M 100

=== Classifier model (full training set) ===

J48 pruned tree

<REDACTED>

Number of Leaves : 7

Size of the tree : 13

=== Stratified cross-validation ===

=== Summary ===

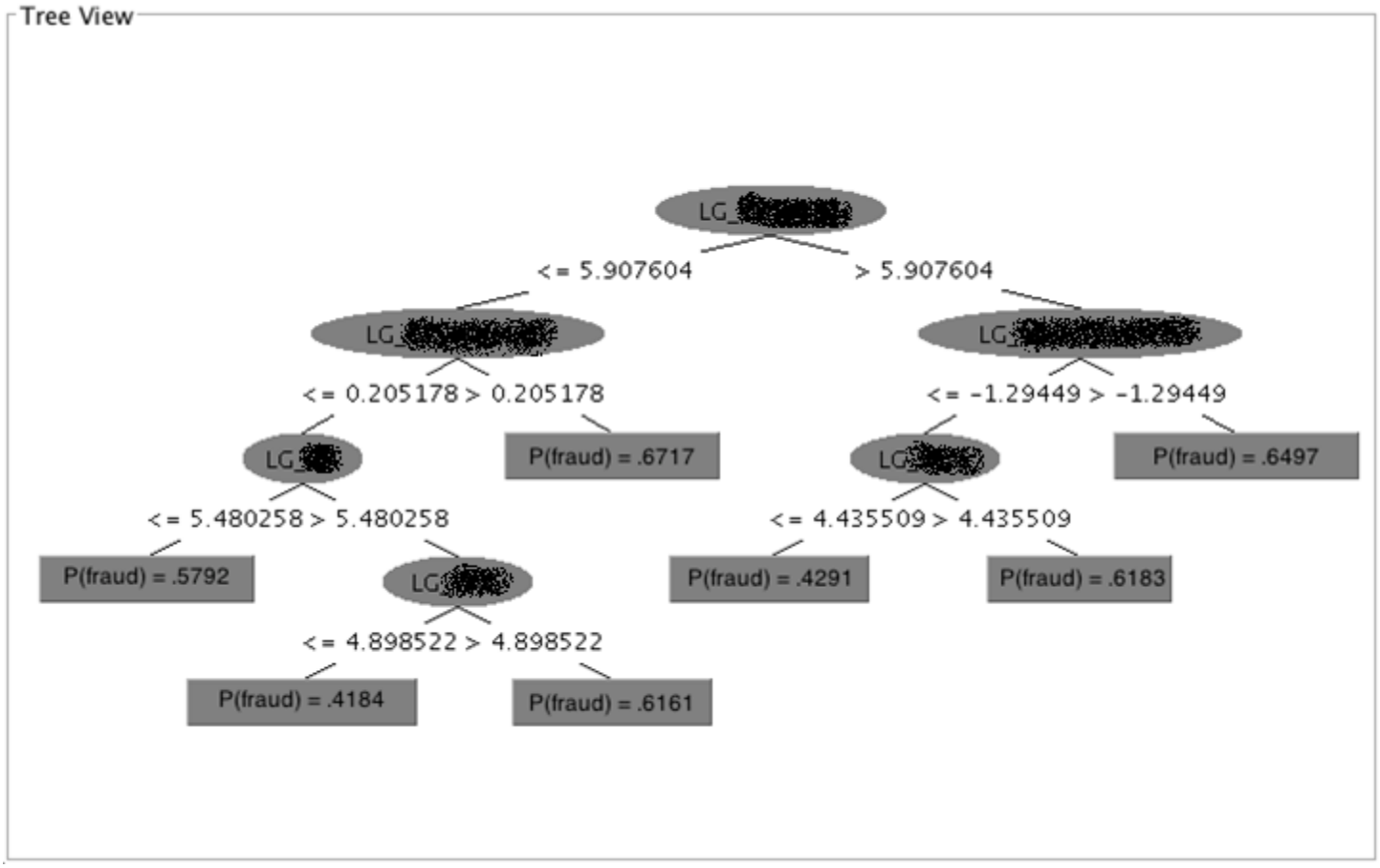
Correctly Classified Instances	18495	62.0742 %
Incorrectly Classified Instances	11300	37.9258 %
Kappa statistic	0.1799	
Mean absolute error	0.4679	
Root mean squared error	0.4842	
Relative absolute error	96.0779 %	
Root relative squared error	98.1197 %	
Total Number of Instances	29795	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.366	0.196	0.575	0.366	0.448	0.593	0
	0.804	0.634	0.637	0.804	0.711	0.593	1
Weighted Avg.	0.621	0.45	0.611	0.621	0.601+	0.593	

=== Confusion Matrix ===

a	b	<-- classified as
4579	7917	a = 0
3383	13916	b = 1



This tree can be deployed in the field. In order to use the tree, the relevant attributes must be filtered through the tree so that each return will arrive at an end node and assigned a probability of being fraudulent. For example, consider a hypothetical return A with the following attributes:

```
LG_<REDACTED1> = 6
LG_<REDACTED2> = 0.4
LG_<REDACTED3> = 4
LG_<REDACTED4> = -1.3
LG_<REDACTED5> = 3
```

The first step is to look at the top level of the tree, which splits on `LG_<REDACTED3>`. Because `LG_<REDACTED3>==4` is less than 5.91, we move to the left in the tree. Because `LG_<REDACTED2>==0.4` is greater than 0.21, we move to the right and reach an end node where P(fraud) = 0.6717. This model would generate a probability of fraud equal to 0.6717 for this return.

In this fashion, the tree model provides a predicted probability for each return. Now, we need to find out which returns should be targeted. In the cost matrix of the J48 model, we assign a benefit of 1 for true positives and a benefit of -1 for false positives:

	Actual FRAUD	**Actual NO FRAUD**
Predicted FRAUD	1	-1
Predicted NO FRAUD	0	0

The reason for using 1 and -1 is that revenue is not involved at this current stage. We only need to differentiate the correctly predicted and falsely predicted instances. The recommended threshold from WEKA is 0.57, which means any return with predicted probability of fraud higher than 0.57 should be targeted for audit.

We also tried other techniques to induce a model, such as J48 with bagging or boosting using the same 0.10 confidence level and minimum of 100 objects in end nodes. There are slight improvements on overall accuracy, but not on recall. Given a budget, only a certain number of returns can be audited. Thus, improving the accuracy of targeting actual non-compliant companies is the primary goal. In other words, we want a model with higher recall given similar accuracies. Because our original J48 tree has the highest recall and is the easiest to use in practice (it is difficult to generate business rules with bagging and boosting techniques), we decided to move forward with the J48 classification tree in order to produce estimates of the probability of fraud for each company.

No additional evaluation of this model is necessary. The n-folds cross validation technique used to induce the model also evaluates the model by leaving a portion of the data out from each model induced from each fold, and evaluates the induced model against this left-out data.

MAXIMIZING AUDIT PROFITS

Now that we have a probability of fraud for each return, we must produce a model by which we can estimate the money received from an audit of a potentially fraudulent return. In order to do so, we first removed all instances where `FRAUD = 0` and then removed the `FRAUD` variable itself. We performed a linear regression with the same five variables used in the J48 classification tree as the independent variables and with revenue as the dependent variable. We decided to use the same five variables in the regression for purposes of consistency. Doing so yields the following regression equation:

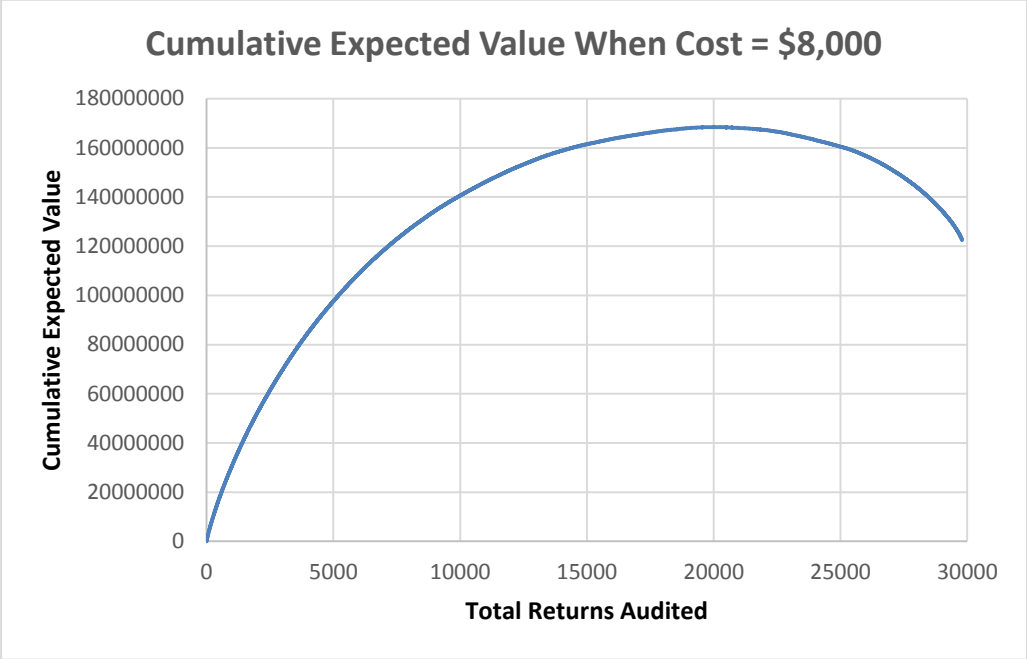
$$REVENUE = 31125.68 - 2921.65REDACTED + 24111.35REDACTED$$
$$+1000.17REDACTED + 1456.09REDACTED + 19153.08REDACTED$$

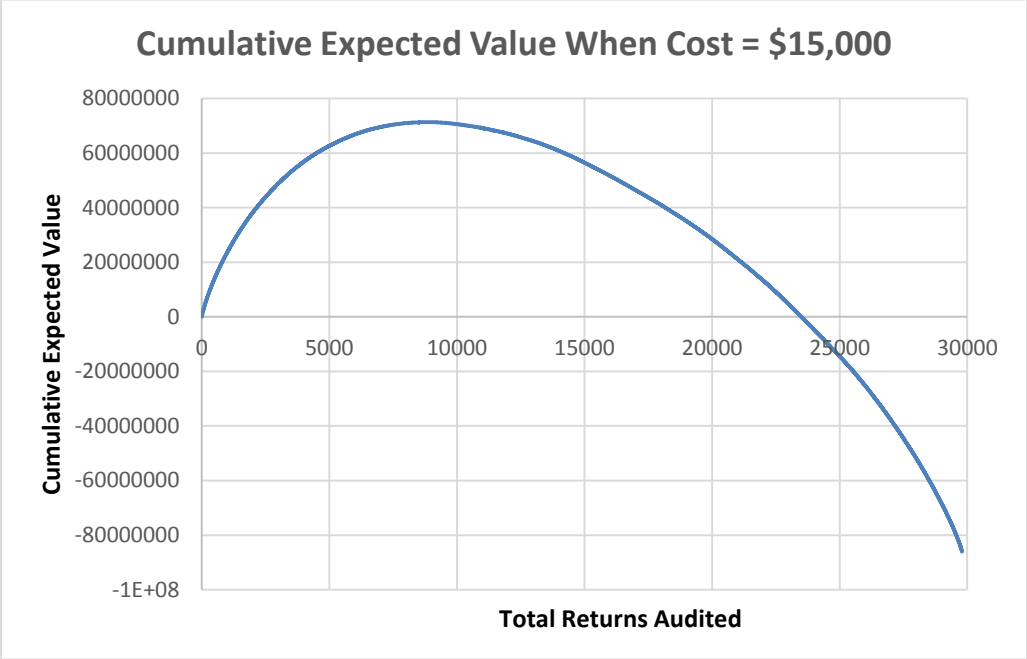
When producing this regression in a statistical software package, all p-values were zero or very close to zero, leading us to believe all coefficients are significantly different from zero at a 5% significance level.

Using this equation, revenue was estimated for each return. Combining this revenue estimate with the probability of fraud found earlier in the J48 classification tree, we can calculate an expected value for each instance and sort descending by this expected value to find the most profitable returns. The expected value for each return was calculated with the following:

$$E[profit] = P(fraud) \times (E[revenue] - cost) - P(no\ fraud) \times cost$$

Cost can either be $8,000 or $15,000 depending on the type of audit. After calculating the expected value of each return, we sorted the returns by this expected value so that the companies are in descending order from top targets to lowest targets. The following charts show cumulative expected value as a function of the number of companies targeted for audit at a cost of $8,000 and $15,000, respectively.





At a cost of $8,000, our cumulative expected value starts to decline after targeting beyond our top 20,000 returns. At a cost of $15,000, the decline occurs after targeting beyond our top 9,000 returns. Clearly, the higher cost of $15,000 limits the number of returns we would hypothetically want to target. And of course, the tax authority can only audit as many returns as provided in the auditing budget.

EVALUATION OF AUDITING BASED ON EXPECTED VALUE OF PROFIT

In order to compare our model to a base case, we found the average revenue of all instances from the raw dataset. We used this to estimate how much money we would expect to receive on average if we audited returns randomly. For a cost of $8,000, we would expect $13,673.30 - $8,000 = $5,673.30 profit per company targeted. For a cost of $15,000, we would expect to lose $1,326.70 per company. Our base cases can thus be represented with the following equations, where x is the number of returns targeted (audited):

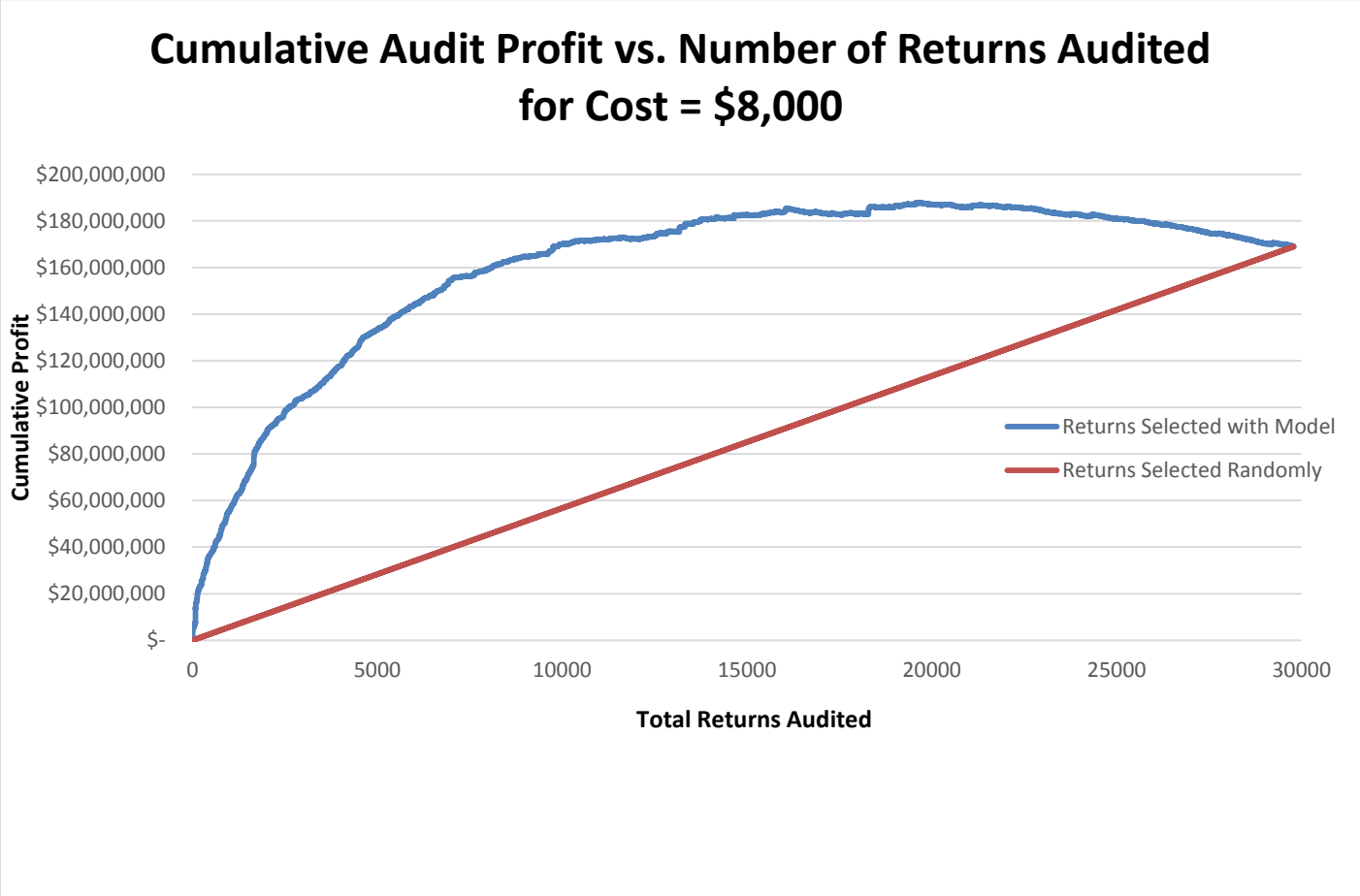
For cost = $8,000:

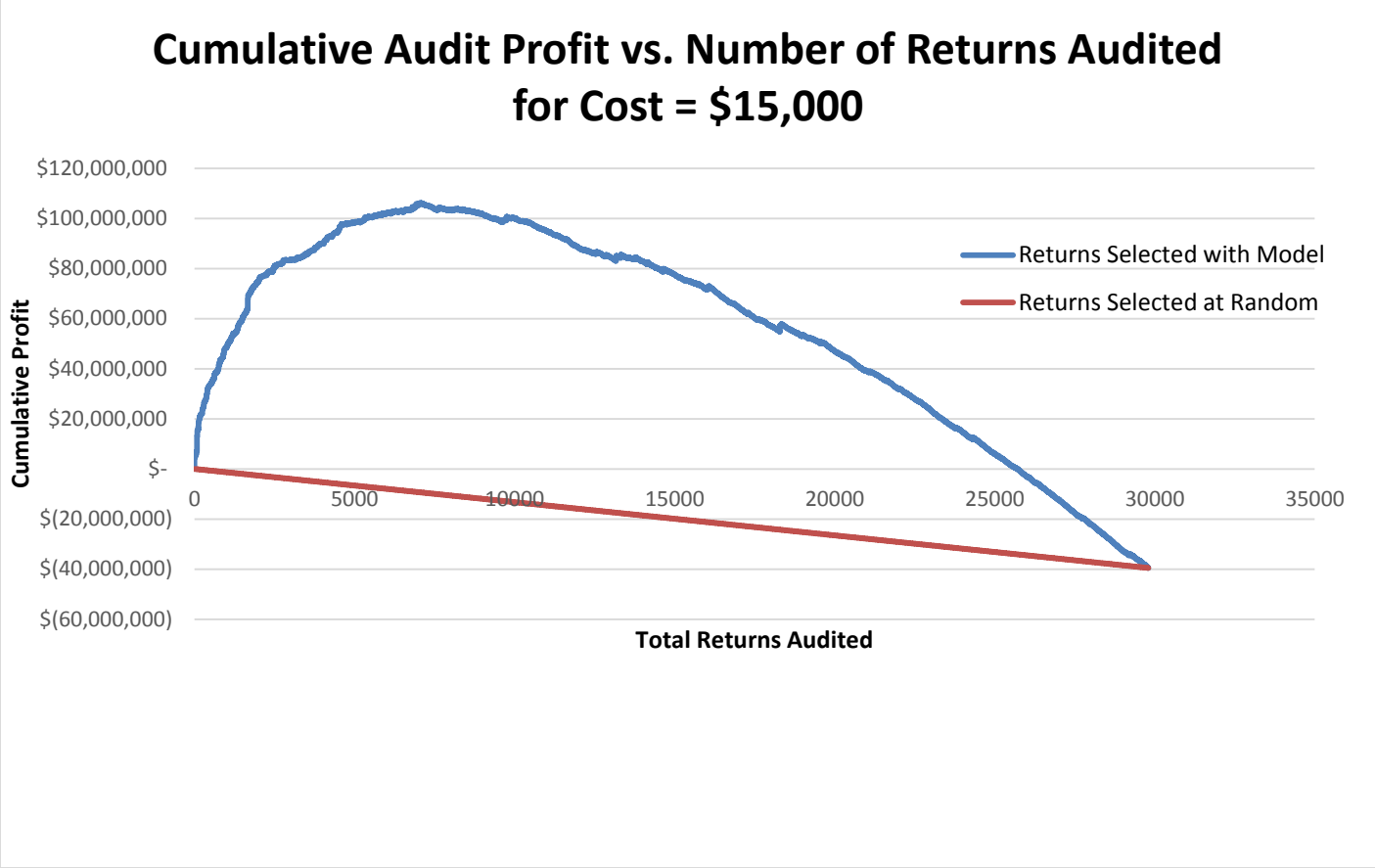
$$y = 5673.3x$$

For cost = $15,000:

$$y = -1326.7x$$

These base case lines are plotted next to our model's actual performance:





Note there are two lines on each graph, the blue line when you audit claims starting with the highest expected value as produced by our model (using the actual profit from the audit, not the expected value), and the red line from auditing claims randomly. Our "random" benchmark line comes from our provided data, as described above. In a real world situation it is likely this red line would look very different (and would almost certainly not produce a profit). But for the sake of evaluating our model, this baseline is sufficient.

For a scenario where an audit costs $8000, our model makes the state much more money than auditing returns at random. No matter how many returns the state is able to audit, using the model is always better. At the most extreme difference, the lift of our model (meaning how much our model outperforms random) is 4 (meaning our model makes the state four times as much money as auditing returns randomly, this lift of 4 is at around 4000 audits). At a certain point (around 20,000 returns audited out of around 30,000 total) our model has found most of the profitable returns and further audits lose more money for the state than they generate. But it is important to note this is ⅔ of the way through all the claims. While we do not have exact information on how many returns the state is able to audit, it will never approach 66.6% in any foreseeable scenario. For the normal volume of auditable returns (with a very optimistic cap of being able to audit 10% of all filed returns), our model is always better than random. Therefore, with an audit cost of $8000, we recommend the state use our model, and that the state audit as many returns as they can with their current resources, starting with the returns having the highest expected value and working down.

The $15,000 cost-per-audit scenario plays out differently. First, even in this fraud-dense dataset randomly auditing returns results in a loss of money to the state. Second, our model's profitability tops out at a much smaller percentage of the claims. In this case, at around 7,300 audits profit is maximized, meaning our model is only making money when we audit the returns with the top 25% of expected values. While our model consistently outperforms random, profit tops out much earlier in the set of returns vs. the $8000 scenario. However, we feel that given the state is realistically only able to audit a small percentage of returns and that our

model is still always better than random in the $15,000 cost scenario, the state should audit as many returns as they have resources for (as long as they follow our model and audit based on highest expected value).

One final point to note is that our data is very fraud-dense. This means that our comments about the point at which our model maximizes profitability are questionable--while we could make a bold claim such as "if an audit costs $8000 the state should audit ⅔ of returns", such a claim comes from a dataset that does not represent reality.

We would strongly prefer to evaluate our models on a set of data that better reflected the actual percentage of returns that contain fraud. Over 50% of returns being fraudulent seems unlikely in the real world, and we would likely gain better insight into the real-world performance of our model if given the chance to test it with more realistic data. At the very least, we would be able to provide a better estimate of the point where the state's profit is maximized. However, we still feel that our model is sufficient for the volume of returns the state can audit (no more than a few percentage of all returns), and that at either cost point the state should choose returns to audit based on our model's expected values.

CONCLUSION

Our recommended strategy to the state is to use our first model's tree and business rules if the intention of the state is the find as many fraudulent claims as possible. If the intention of the state is to maximize the profit generated by tax audits, they should use our second model, and audit the returns in order of expected value as generated by the model. At a cost of up to $15,000 per audit, as long as the state is only able to audit a small percentage of returns the model should produce a profit for the state significantly above auditing returns randomly.

While we are confident that our models outperform random selection, there are a few shortcomings we want to address. One is that our models were built from a dataset with an unrealistic proportion of fraudulent returns, and this could result in our model's performance being inconsistent on a dataset where fraud was less common. Unfortunately, we were not provided with more realistic data to investigate this. We highly recommend the state further evaluate our models using a test dataset where the incidence of fraud is closer to that of reality before deploying our model-informed strategies in the field.

Second, we feel we could have created better solutions for the state if we had better knowledge of the problem. We are not accountants or tax experts; our domain knowledge is minimal. Most of the fields in the data do not have any meaning to us other than as arbitrary labels. It is possible greater insight into the problem and our dataset would have led an even better set of models and procedures.

Finally, it is not certain this model will hold up over a long period of time. Taxation is a dynamic system, and with every change in the tax laws the construction of returns changes, and thus the patterns in a return that indicate fraud are likely to change as well. It is also possible that fraudsters will become savvy to the states' means for evaluating fraud, and will adjust their returns to avoid detection by the system. Given the number of people who will have access to the model, it is likely some of the operational details will leak into the accounting and business communities, and that new models will need to be induced every few years to keep pace with the changing nature of tax fraud.