**Predicting Monthly Survey Submissions with Machine Learning**

**Executive Summary**

We try several approaches to maximize recall on predicting non-participants in a monthly survey. The primary method used was a decision tree classifier, with iterations of parameters to optimize for our goal. Our final model did not achieve our minimum acceptable benchmark but further steps for experimentation are prescribed. The model will be used on a prototype basis to begin predicting non-participants, but results may not always be actionable. We recommend continued analysis and research.

**Introduction**

The bane of every monthly survey is participant retention. If participation drops, survey coverage decreases, reliability suffers, and the remaining participants find the survey less valuable and will question the time commitment needed to participate. Slowly but surely, if no remedy is taken and no extraneous forces intervene, all parties begin to abandon ship, and the survey enters a calamitous tailspin and collapses in a pile of rubble and ruin, a far cry from its former glory and grandeur.

In light of the importance of participant retention to survey success, we have decided that a key element of survey longevity is the ability to predict member participation on a monthly basis. If a member is predicted to stop submitting, we can stage an intervention and bring them back.

Our model will attempt to predict the probability that a participant in our survey will stop submitting, and subsequently require human contact to remind them to submit. We can then anticipate which participants will need more attention and tailor follow-up (i.e. emails, phone calls, etc.) to them, that is customized based on the model results. This follow-up will not only reduce the amount of contact needed over time by prescribing effective remedies to the problem and saving human resources for the administrator of the survey, but it should improve the participation in our survey. The client is the administrator of the survey.

**About the Survey**

The survey collects data from US-based manufacturing companies and compiles orders data to capture a monthly picture of the US manufacturing market. The survey is free and currently collects data from around 180 different companies every month. It is the duty of the analysts at our company to ensure that current participating companies submit their orders data and to reach out to as many manufacturing companies as possible for recruitment in free time. Due to the other various duties of analysts, we do not always have the time and resources to selectively reach out to companies that are in danger of not submitting.

As previously stated, the goal of this project is to assess the likelihood that a participant will submit so that our analysts can focus their attention on the companies that are most likely to not submit and drop from the survey. We are assuming that companies that take longer to submit are more at risk to drop from the survey, but this is a hypothesis that is investigated further in the report.

This would also give us more time to recruit new companies to the program so the survey can get back to the numbers it once had. Greater participation would result in a better, more reliable survey.

**Defining the Goal**

The value-add behind this model is the ability to capture non-participants and intervene in a significant way. Due to the close relationships between the client and the participants of the survey, excess participant contact is not a big deal. Participants embrace communication and seek to build a stronger relationship with the client. However, we still need to balance over-classifying with under-classifying, as the cost to over-classifying is excessive time dedication on the analysts’ part towards reaching out, and analysts cost money. The cost of under-classifying is obvious - losing participants and taken to the extreme - the entire survey (which is a major product of the client’s) if we can’t catch any non-participants.

Therefore, we decided to define a successful classifier which can predict with:

* At least 90% recall on non-participants (will not submit).

To reiterate, we decided on this benchmark because the business problem demands that we need to capture the vast majority of non-participants. This is key to giving us the information necessary to “intervene” when non-participation becomes evident. Precision for non-participants, while important, is not necessarily the most important metric, as occasional contact with those who would have submitted anyways does not outrank capturing non-participants (low precision would just give us many false positives, which may cause participants to get annoyed, but otherwise would not impact the business problem).

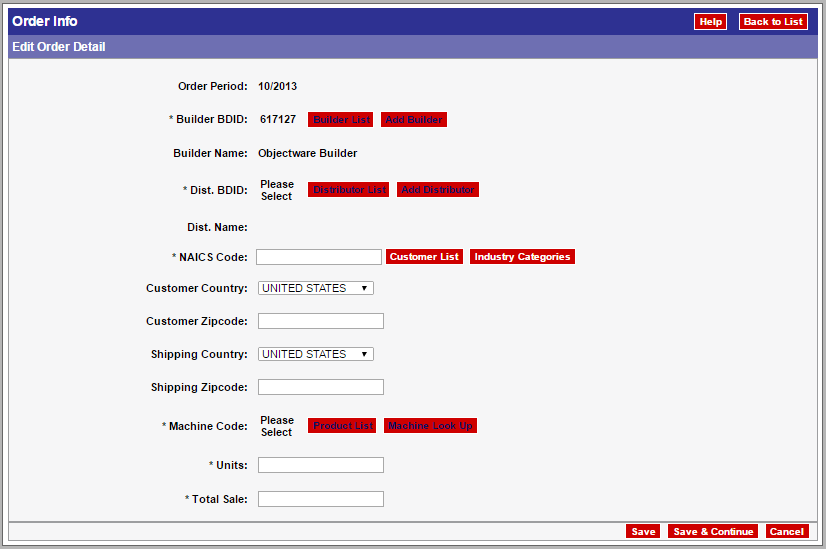
Precision and recall for participants are even less important. This is because we do not take any action on those who would have submitted anyways; capturing all those who would have submitted and being precise on who has submitted is subservient to capturing non-participants, and much of the value add in precision/recall for participants is already captured and expanded upon in precision/recall for non-participants.

**The Data Set**

Our dataset consists of survey data, industrial production (IP) data, and S&P 500 data. The main dataset, survey data, is rich with features and spans back to 1997.

On a monthly basis, participants submit 10 data points: builder ID, Distributor ID, NAICS code, customer country, customer zip code, shipping country, machine code, units, and total sale. From this information, numerous other data points are automatically populated in the database and complemented with details from reference tables. Some of these other data points include order period, company name, parent category, machine category, submit date, and created by.

Note - Customer and shipping zip code/country are distinct fields because a machine can be ordered from a company headquarters, but shipped to another location.



All company orders are originally added to to tbl\_Order as soon as they submit their month data. At the end of the month when we process reports, tbl\_Order is processed and tbl\_Fact is created. Tbl\_Fact is the table where all the processed numbers are stored. A large part of this process involves duplicate elimination. Since orders can be submitted by both a Distributor and Builder, there is a need to remove order duplicates in the database. An algorithm tracks duplication by looking at the total sale, the zip code, and machine code of an order. If these fields match, the duplicate is eliminated so that just the builder’s order remains and the order is not double-counted in final reports.

All of this data is stored in Amazon cloud with Amazon Web Service and is processed using a SQL server. There is a database interface in Access that is linked to this SQL server where queries can be run to examine the data and be linked to outside sources such as Excel or Tableau.

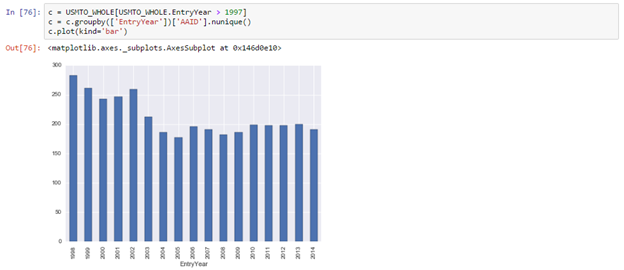
The columns we decided to use in our initial analysis are as follows:

1. Company ID - a unique identifier for each company
2. Order Date - the applicable month for each set of data
3. Total Sale value of the current month
4. Units Sold in the current month
5. Days Late they submitted in the current month
6. S&P 500 open price of the current month
7. Industrial production value of the current month
8. Received first reminder email dummy variable
9. Received second reminder email dummy variable
10. Received personalized call (target) dummy variable
11. Dummy variable for if company backfilled data
12. Average of company’s last three month’s sales
13. Average of company’s last three month’s days late
14. Average machine value sold by month by company

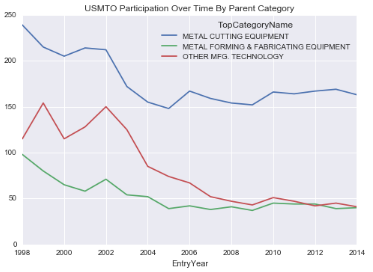
**Exploratory Data Analysis**

After the data wrangling and feature engineering, the next step was exploring our dataset. This started by reviewing the participating companies and learning more about them.

Our first look at participation shows survey participation trending in a bad direction (# of participants on the y axis).

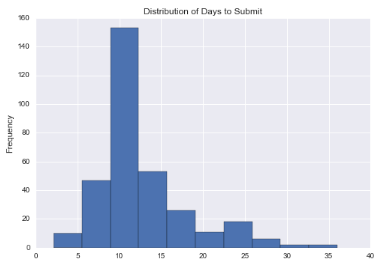


This trend is seen across categories as well.

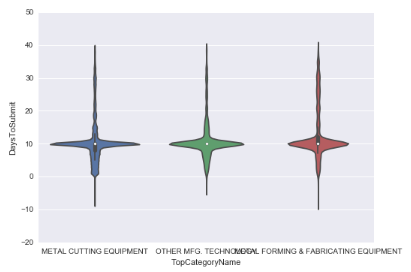


From this, we can see a clear downward trend in participation in the survey. The next step involved analyzing submission trends. In looking at this, we cleaned the data set to only include relevant values. We determined this to be values between -10 and 40 days to submit. This was done to try to remove outliers caused by incorrect data entry (submitting too early thinking it is different month of data) and backfills when a company enters historical data when they enter the survey to gain access to these months of data.

Visualizing within our cleaned dataset gives us insights that late submissions generally happen about 10-15 days after the deadline. This is due in part to our analysts’ efforts to reach out and contact non-submitters a few days after the deadline. This also reinforced our belief in the value of this project, as reaching out perhaps a week before the deadline would maybe cause the maximum frequency of days late and the distribution to shift closer to 0.



A quick look at a violin plot can also give us a few interesting insights. From this visualization, one can see that Metal Cutting and Other Manufacturing Technology have the skinniest upward tails, indicating that they have less participants submitting late. Metal Cutting also has the widest center, showing that most participants are submitting in a good range of around 10 days. The key insight to glean from this is that these three categories are inherently different, a point which will be discussed later in this paper.

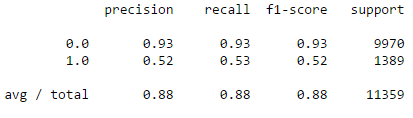


**Decision Tree Classifier (CART)**

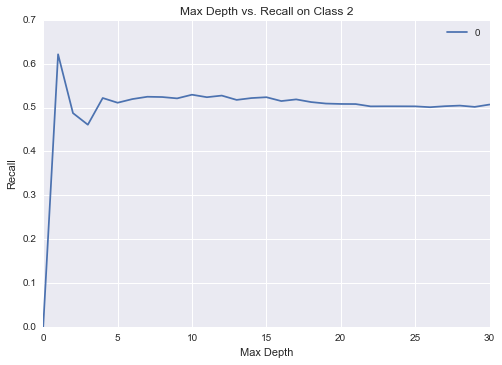
We decided to use a decision tree because they are easy to understand and visualize, in addition to being well-adapted to process large amounts of data without “outlier classes”. That is, we only have two classes with about a 70/30 split, so no classes will be pruned out during the tuning phase. With the initially relatively high performance, our large data set and desire to understand the problem better, decision trees seemed like an obvious choice to focus on.

Decision trees also describe relationships between the target and each feature well, and does so in a visual way. It has the added benefit of giving valuable insights even if the classifier does not reach the intended goals.

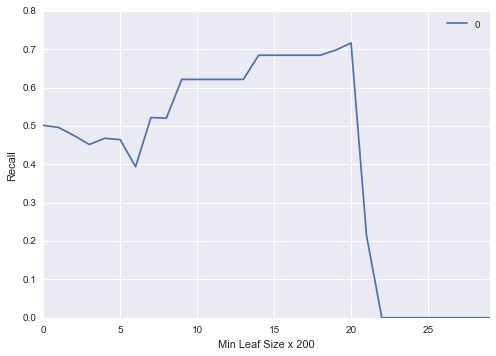
Our results, though better than kNN on many key measures, still did not achieve our business goals. *(0.0 indicates submitters, 1.0 indicates non-submitters)*



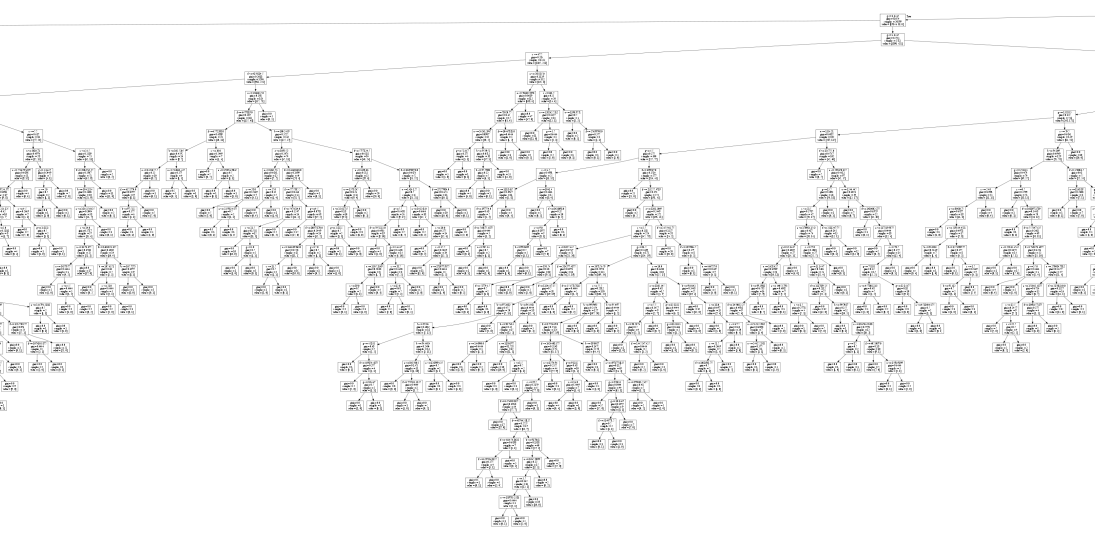
These results are on test data. One problem was immediately clear after investigating further - overfitting. We plotted max depth vs. recall and found that our current max depth of infinity produces a lower recall than the optimized depth of 10 (depth of 1 was ignored as a single-node decision tree is not a valid model)



We also tuned minimum leaf size - below is a plot of the curve which shows optimized leaf size, which was found to be 4000.



A small sample (by no means the complete picture, but illustrates the depth) of our initial massive decision tree is also below, which is neither very readable nor interpretable, and the length of which indicates probable overfitting. The remedies listed above are expanded upon in the next section and will greatly reduce the size of this tree, increasing recall and decreasing unproductive complexity:

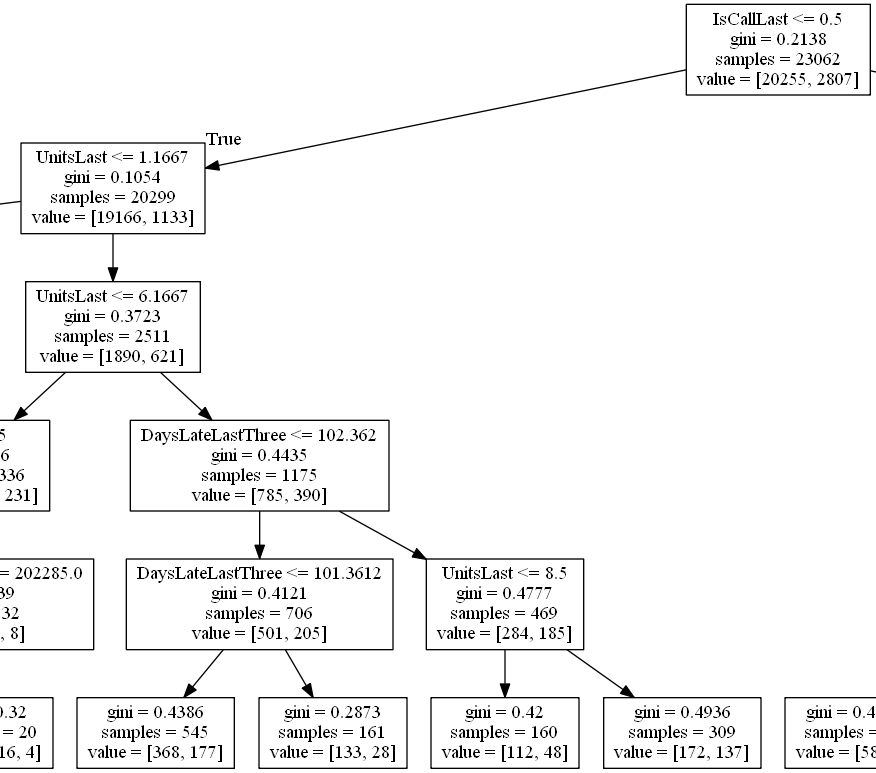


Overfitting and attempted remedies

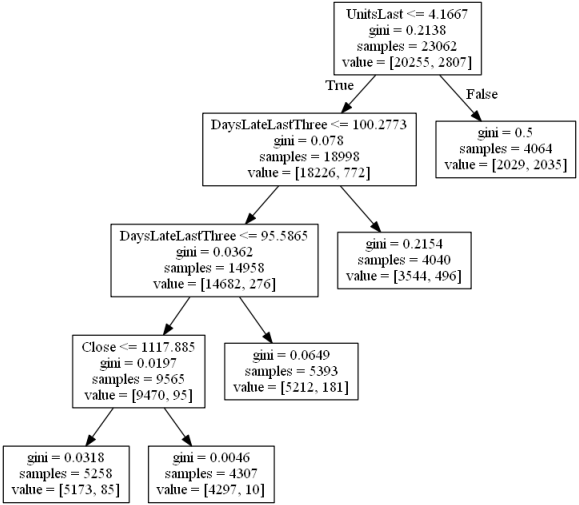
Given our specific problem, there are several parameters which we can tune to increase our model accuracy. These levers can be pushed and pulled in certain ways to optimize for our specific set of strengths and weaknesses.

Changing the maximum depth of the tree to 10 increases our recall on non-participants - as indicated in the curve plotted in the previous section.

This may be because our tree is suffering from overfitting and setting a maximum depth gives a constraint that helps the tree “not overfit” itself. Below is a figure (not the whole tree, but again illustrates the full depth) of our decision tree modified with a max depth - as you can see it is much cleaner, smaller, is only nine layers deep, and thus less likely to overfit.



We also raised the minimum samples per leaf up to 4000. This has the same idea of instituting a max depth, cutting the number of leaves and giving us a small, more concise tree that is less suspect to overfitting. See figure below:



This brought our final recall on non-participants up to 0.72 on test data, unfortunately still short of our target.

We also attempted several other tweaks to parameters which were minimally fruitful. The iterations are listed below:

1. Restrict to a maximum number of leaf nodes. Target metrics were not increased, possibly due to the fact that enacting a maximum depth already covered much of the efficiencies gained through this method.
2. Random splits on nodes instead of best split. This iteration split nodes, not based on the Gini index, but rather on a random sequence. We did not get closer to the goal with this method, possibly due to the incompatibility of our data set with this procedure (random splits are usually used for ensemble methods, which is not the method used in our model).
3. Splits on information gain (entropy) instead of Gini impurity. Again, this did not help us get closer to our goal – after further investigation, it was found that the choice of impurity measures has, across the board, little effect on outcomes).

**Further Feature Engineering with Domain Expertise**

We iterated through several sets of features but the most prominent was adding in three features, one for each category each observation fell into (cutting, forming, or other manufacturing technology [MT]). Remember that each observation is by company-month, meaning one observation is the aggregate of sales for one company, for one month. Remember also that metal cutting, metal forming, and other MT had different average days late from our EDA stage. Thus we hypothesized that this would be a predictor. We assigned each company-month observation to the greatest proportion of sales from each category for that month. For example if Company A had 70% sales from cutting, 20% from forming, and 10% from MT, we assigned it to the cutting category.

However, after adding these three features, recall actually fell significantly (15%), despite seeming like a good predictor at first.

Other modifications with features also proved fruitless, and as previously stated we had better luck with tweaking the models themselves to achieve higher metrics.

**Further Research**

Though we did not achieve our business goal, it does not mean there is no information to be scavenged from the model. Rather, we have constructed a fairly accurate model with a recall of 0.72 on non-participants with our most effective classifier, decision trees with max depth of 10 and leaf size restriction, but this still does not translate to something we would implement on a month to month basis. We have also visualized how features can determine class by looking at our decision tree, for example we now know that a small unit count the previous month may signal non-participation – an insight we can share with the team. This can facilitate discussion on why this may be and how this piece of information can be incorporated into the client’s business.

Because our model is close to our goal, there are several things we could try to help it reach the benchmarks we set. The classifier we used (CART), is notorious for overfitting. We can implement methods such as different types of pruning and selecting better features to reduce overfitting. This may be enough to push our recall values up to our goal.

It may have also been that other models were better classifiers, but we lacked the necessary data and up-front know-how to use them effectively.

More external datasets can be brought in to better the accuracy of the model. An approach would be to comb through the FRED (Federal Reserve Economic Database) for more explanatory features. We could also combine our current features in such a way to provide better predictive power.

We can also ask our domain hoster for weblogs pertaining to user activity. These may have very high predictive power due to being an effective proxy for participant behavior. Such weblogs include information like -

1. Participant usage of reports (frequency of reports run by each participant, number of reports run, etc.)
2. Participant login time
3. Participant web traffic activity within the survey website (time spent on each page, links clicked, etc.)

These may be highly explanatory but would require significant cleaning to reach a usable format. They would be acquired as raw text files and need to be reformatted to make sense of the data.

**Business Case for Recommendations**

It will be good to test the model on new data points and see if predictions are accurate and live up to our predictions. Although the model is not ready to go into a fully automated system that can reach out to problem participants, we can still take the findings of the model to improve the efficiency of our participant outreach. By using the classifications of participants created by the models, we can focus our attention on the delinquent participants.

It will also be good to use report usage analysis to improve the survey. By observing what reports are used most by participants, we can see what data is most important to participants. Our model can be used to assess survey participation based on usage of these reports, and we can focus on improving or adding to reports that commonly predict high survey retention (indicating the report is useful and critical to why participants continue submitting). Using this information, we can work to improve these reports or add additional complementary reports to create an improved survey program. This would serve to improve survey participation by making a survey that participants value more. If participants value the survey more, they will be more likely to submit data.

Finally, we should implement A/B testing to determine best methods to contact participants. Contacting participants has been done using various methods: automatic email reminders, personal emails, and phone calls. Each of the methods has also taken different iterations. Throughout this, at no point has there been any work to test out what methods work best. Our model can be used to match contact methods with participants that have certain features - for example we may test using personal phone contact to alert a sub-group of members with low unit count(indicating they may be in a business slump). We could then test another group of low unit count members with personal emails to see which results in a higher re-participation rate, and repeat this method for every feature/contact pair.

Moving forward, I’d recommend that participant contact methods be monitored and recorded. We can test to see what methods are most effective in getting participants to submit data and then refine our methods for participant outreach based on their features.

**Conclusion**

Surveys of market conditions are helpful only if people participate. Unfortunately, people will only participate if it’s been shown that a meaningful portion of the market is already captured and the information they get from the survey will be helpful. This Catch-22 all too often is the Achilles Heel of many a survey. Often times, to achieve this initial baseline of participation, a concerted effort is made to gather key participants together and coerce them into participation through wining and dining. After this initial gathering, the survey administrator must dedicate resources to retain these contributors else suffer the consequences – a dead survey and lots of wasted wine, food, and venue rental fees. Participant retention is not only important to a survey’s success, but it is core. Our project centers around this principle – that we must find a way to retain participants if our survey is to survive. We try several machine learning methods to identify non-participants, based on past survey information and current macroeconomic indicators.

We defined a successful model as one which could identify 90% of all non-participants. In the end, we found that the methods used were not effective enough to reach this goal. The top classifier we found was the CART classifier, which was further tuned to maximize for recall on non-participants. Various methods were used to boost the recall of our decision tree quite effectively. We knew from prior knowledge that decision trees were notorious for overfitting, and we noticed that our decision tree may be suffering from it when we visualized the entire tree. The width of the tree indicated there may be some unproductive complexity occurring, and we restricted the depth and instituted a minimum leaf size to arrive at a simpler tree with greater recall. Even through our valorous attempts, we could not breach the impregnable fortress of 90% non-participant identification, reaching a formidable but ultimately insufficient 72%. Though the business goal was not ultimately reached, further identified sources – weblogs, click data, etc., may be used to boost our recall the 18% necessary to reach our threshold for use. We recommend the project be expanded upon and these sources integrated into the model and other sources identified.

Perhaps one day, our model can be used the end the cycle of non-participation in all surveys, in all industries, and in every format of delivery. As the legendary Captain Spock once said, “One can begin to reshape the landscape with a single flower.” We hope our model can be that single flower, and we hope it will end the ruination of surveys everywhere – yet another victory for machine learning.