# Risky Business: Predicting Cancellations in Imbalanced Multi-Classification Settings

Anand Deshmukh1, Meena Kewlani, Yash Ambegaokar, Matthew A. Lanham
Purdue University, Department of Management, 403 W. State Street, West Lafayette, IN 47907
deshmuk6@purdue.edu; mkewlani@purdue.edu; yambegao@purdue.edu;
lanhamm@purdue.edu

## ABSTRACT

We identify a rare event of a customer reneging on a signed agreement, which is akin to problems such as fraud detection, diagnosis of rare diseases, etc. where there is a high cost of misclassification. Our approach can be used in all cases where the class to be predicted is highly under-represented in the data (i.e. data is imbalanced) because it is rare by design; there is a clear benefit attached to this class' accurate classification and even higher cost attached to its misclassification. Pre-emptive classification of churn, contract cancellations, identification of at-risk youths in a community, etc. are potential situations where our model development and evaluation approach can be used to better classify the rare but important events. We use Random Forest and Gradient Boosting classifiers to predict customers as members of a highly underrepresented class and handle imbalanced data using techniques such as SMOTE, class-weights, and a combination of both. Finally, we compare cost-based multi-class classification models by measuring the dollar value of potential lost revenue and costs that our client can save by using our model to identify at-risk projects and proactively engaging with such customers. While most research deals with binary classification problems when handling imbalanced datasets, our case is a multi-classification problem, which adds another layer of intricacy.

**Keywords:** Predicting cancellations, Class imbalance problem, Rare class mining, Data imbalance, SMOTE, Random Forest, Gradient Boosting

[1] Corresponding author

# INTRODUCTION

The ability to predict future sales from various leads is a challenging problem. Usually the sales process has multiple stages, with competing interest among buyers and sellers. Identifying strong leads and allocating resources to potential customers is always an important problem for a sales team. If an associate obtains positive feedback or even a verbal commit to a purchase, it provides additional complexity when the customer reneges on the commitment at a later stage. Many firms must bear the sunk costs associated from a customer's decision change. Examples might include shipping costs, additional inventory costs, as well as wasted team member time. Moreover, larger the size of a project, greater are the number of resources allocated to it (Duran, 2008).

In this study, we partnered with a local business (hereon referred to as "partner company") to understand reasons for their customers cancelling a project after initially agreeing to it. These are resource-intensive and high-cost installation projects and such unforeseen cancellations pose a significant risk to the partner company. The sales pitch for our partner company is an intensive process that their sales force spends a considerable amount of time and resources on, hence, such cancellations also waste the time of the sales force and reduce morale.

Today, firms are employing analytics to tackle these problems of uncertain demand and resource allocation. Those firms that collect the right data at the right time have essentially invested into helping themselves improve processes and services in the future. For our problem, if a firm has several stored transactions, it has been shown that probability estimation techniques could be used to provide insights into an opportunities' potential ( (Duran, 2008); (Lodato, M. W. & M. W. Lodato, 2006); (Söhnchen & Albers, 2010)). Studies in the classification modeling domain have focused on B2B sales forecasting and organization learning using machine learning (Bohanec, Robnik-Šikonja, & Borštnar, 2017). Machine learning can outperform subjective association decision-making in the B2B space was shown by (Yan, et al., 2015).

Aspects of our problem have been seen in the healthcare realm. For example, (Sahraoui & Elarref, 2014) propose a problem of patients scheduling elective surgery at a hospital. Here they committed to have a surgery and the hospital has dedicated resources (e.g. allocated a room for surgery, a bed/room for a patient, scheduled a surgeon and/or anesthesiologist, and allocated time for surgical services). However, if the patient does not show up, the hospital has effectively lost business. In

their study, rather than building models, they take a theory of constraints approach to help identify underlying root causes for what led to a cancelation and better plan for future possible instances.

This study could be viewed as falling under the Customer Relationship Management (CRM) umbrella because we need to understand the information flow process in order to improve customer acquisition and retention (Chakravorti 2006).

The aim of this study is to identify at-risk projects so that our partner company's sales force can take pre-emptive measures to save the customer's business. We also identify projects that would get successfully completed as well as those which would be declined by the partner company, thus taking a step towards improving the sales pipeline.

We organized this paper by initially reviewing the past works on various topics related to customer cancellations and churn. Second, we discuss the data used in our study to help our partner company understand their customers better. Third, we outline the methodology we implemented and discuss the several models we investigated to predict the likelihood of a customer reneging on a signed project. Finally, we present our results, discuss our conclusions and how we plan on extending this study.

## LITERATURE REVIEW

A strategic goal for most businesses is to improve the productivity of its sales force. Identifying new sales opportunities and ensuring that sales professionals are deployed to serve the best potential-revenue generating accounts is critical to a company's revenue growth. An analytical challenge is to predict the likelihood of a customer buying a product or a service. If a large amount of stored transaction data is available, probability estimation techniques could be used to predict the outcome of an opportunity, based on its sales funnel ( (Duran, 2008); (Lodato, M. W. & M. W. Lodato, 2006); (Söhnchen & Albers, 2010)).

Sales forecasting is a complex process for several reasons. There are multiple stages involved, each stage has several participants (from the buyer and sellers side) who may not necessarily have the same objectives and interests. Sales forecasts are a critical cog in making managerial decisions and incorrect forecasting can lead to wasting of resources (Bohanec, Robnik-Šikonja, & Borštnar, 2017).

Customer cancellation is a classification type of problem and machine learning techniques can be employed to improve the accuracy with which the company can predict if a customer would cancel or not (Huang, Chang, & Ho, 2013). Additionally, stakeholders and decision makers in companies are not simply interested in the predictive performance of classification models, they also want to use it to support their decision making. Hence, the interpretability of the prediction models is also important along with the accuracy of prediction (Bohanec, Robnik-Šikonja, & Borštnar, 2017). Before applying a model, a user must trust it – this trust can be generated based on the transparency of the model. Hence, while sophisticated models such as random forests and support vector machines may demonstrate a stronger predictive model, they lack the interpretability of models such as of decision trees and logistic regression (Caruana & Niculescu-Mizil, 2006).

The study conducted by (Kotsiantis, 2007) describes various supervised machine learning classification techniques and compares them across several features. The important take away from this paper is that it is essential to understand under which conditions would a technique outperform the others, for a given problem. A modified version of a comparative study of these techniques, as shown by the paper, is as follows:

| | Machine Learning Techniques | | | | |
|---|---|---|---|---|---|
| Comparison Metrics | Decision Trees | Neural Networks | Naïve Bayes | kNN | Support Vector Machines |
| Accuracy in general | ** | *** | * | ** | **** |
| Speed of learning with respect to number of attributes and the number of instances | *** | * | **** | **** | * |
| Speed of classification | **** | **** | **** | * | **** |
| Tolerance to missing values | *** | * | **** | * | ** |
| Tolerance to irrelavant attributes | *** | * | ** | ** | **** |
| Tolerance to redundant attributes | ** | ** | * | ** | *** |
| Tolerance to highly interdependent attributes (eg. parity problems) | ** | *** | * | * | *** |
| Dealing with discrete/binary/continuous problems | **** | *** (not discrete) | *** (not continuous) | *** (not directly discrete) | ** (not discrete) |
| Tolerance to noise | ** | ** | *** | * | ** |
| Dealing with danger of overfitting | ** | * | *** | *** | ** |
| Attempts for incremental learning | ** | *** | **** | **** | ** |
| Explanation ability/transperancy of knowledge/classifications | **** | * | **** | ** | * |
| Model parameter handling | *** | * | **** | *** | * |

Table 3.1: Comparing learning algorithms (ranked from * to **** (best model))

A very peculiar limitation of predicting customer churn or cancellation is that the data is usually imbalanced. Typically, a very small percentage of customers fall into this category and this small percentage of customers – the minority class – are very often the class of customers we are interested in predicting (Zhao, Li, Li, Liu, & Ren, 2005). Some other examples besides customer churn are fraud detection, diagnosis of rare diseases, so on and so forth. However, according to (Chen, Liaw, & Breiman, 2004), most classification algorithms are built to minimize the overall error and not to focus on this minority class. Two approaches used in tackling imbalanced data are (1) down-sampling the majority class or over-sampling the minority class or both, and (2) cost-sensitive learning i.e. assigning a high cost to misclassification.

A resampling technique that was developed by (Chawla, Bowyer, Hall, & Kegelmeyer, 2002) is SMOTE (Synthetic Minority Over-Sampling Technique). Typically, the minority class is over-sampled with replacement which means that its data points are replicated at random. In the SMOTE technique, the minority class is over-sampled by creating "synthetic" samples instead of creating samples via replication, thereby increasing the information along with the weight of the minority samples. To summarize the SMOTE technique, *k* minority class nearest neighbors are identified and, on the line segments joining any or all of these *k* minority class nearest neighbors' synthetic examples are introduced.

Aside from techniques that could be employed to reduce the class imbalance, certain algorithms were found to perform well on such data. (Chen, Liaw, & Breiman, 2004) discovered that the following two approaches did a significantly better job at prediction of the minority class than the existing algorithms: (1) Weighted Random Forests (based on cost-sensitive learning) (2) Balanced Random Forest based on down-sampling the majority classes. However, they could not discern a difference between the two approaches to identify a winner.

Thereafter, (Xie, Li, Ngai, & Ying, 2009) proposed a new learning method called Imbalanced Random Forests (IBRF) and used it to predict churn in the banking industry. Their study integrates the effectiveness of random forest in prediction of customer churn behavior while incorporating balanced and weighted random forests. Their approach alters the class distribution as well as penalizes the misclassification of the minority class. They find that IBRF has a better accuracy than traditional random forest algorithms. Additionally, they find that the top-decile lift of IBRF

is better than other classification methods like decision tree, artificial neural network and class-weighted core support vector machines (CWC-SVM).

The performance of nine different Boolean classification evaluation metrics was compared by (Caruana & Niculescu-Mizil, 2006) across different settings and machine learning algorithms. Their paper finds that learning methods that perform well on one criteria may not perform well on another, hence, picking the correct evaluation metrics for your models is imperative.

For data with class imbalance, (Tang, Zhang, Chawla, & Krasser, 2009) find that overall accuracy isn't an appropriate model evaluation metric as it cannot appropriately evaluate a model that is ineffective in detecting rare positive samples and assigns the model a high overall accuracy when it predicts all samples to be negative. Instead, they recommend the use of Precision and Recall. The table 3. 2 below illustrates a confusion matrix for a binary classification problem – the columns highlight the predicted classes and the rows highlight the actual classes. True positive and true negatives imply that the predicted and actual class are the same. False positive and false negative indicate the cases where the positive and negative cases were misclassified. The formula and interpretation of Accuracy, Precision and Recall (Larose & Larose, Data Mining and Predictive Analytics, 2015) are listed in table 3.3.

| | | Predicted Class | |
|---|---|---|---|
| | | Positive | Negative |
| Actual Class | Positive | True Positive (TP) | False Negative (FN) |
| | Negative | False Positive (FP) | True Negative (TN) |

Table 3.2: Confusion matrix

| Evaluation Metric | Formula | Interpretation |
|---|---|---|
| Overall Accuracy | $\frac{(TP+TN)}{(TP+TN+FP+FN)}$ | This metric says how often is the classifier correct |
| Sensitivity / Recall | $\frac{(TP)}{(TP+FN)}$ | When an instance actually falls within a class, how often does the model correctly classify it as falling in this class |
| Positive Prediction Value (PPV) / Precision | $\frac{(TP)}{(TP+FP)}$ | When the model predicts an instance to fall within a class, how often does it actually fall within the class |

Table 3.3: Formulae and interpretation of accuracy, precision and recall scores

The findings from the papers related to treatment of imbalanced classes and customer churn prediction are summarized in table 3.4, below:

| **Studies** | **Motivation for the research** | **Algorithms Used** | **Class Imbalance Treatment** | **Results/Findings** |
|---|---|---|---|---|
| (Chawla, Bowyer, Hall, & Kegelmeyer, 2002) | Introducing SMOTE resampling technique | 1. C4.5 Decision Tree<br>2. Ripper<br>3. Naïve Bayes classifier | 1. SMOTE with under-sampling<br>2. Only under-sampling | Combination of SMOTE and under-sampling performs better than only under-sampling |
| (Chen, Liaw, & Breiman, 2004) | Treating imbalanced data with Random Forest classifier | 1. Random Forest<br>2. Ripper | 1. Balanced Random Forests<br>2. Weighted Random Forests<br>3. SMOTE with under-sampling<br>4. SHRINK<br>5. One-sided sampling<br>6. Boosting | 1. Balanced and weighted Random Forests perform significantly better than standard Random Forests.<br><br>2. No clear winner between balanced and weighted Random Forests |
| (Burez & Van den Poel, 2009) | Class imbalance in customer churn prediction | 1. Logistic Regression<br>2. Random Forest<br>3. Gradient Boosting Classifier | 1. Weighted Random Forest<br>2. Under-sampling (random and with CUBE algorithm) | 1. Under-sampling, Boosting and Cost-sensitive learning improve the standard classifier's performance<br><br>2. You don't need to make the sample size the same for the classes<br><br>3. Best performing class distribution depends on the method and case<br><br>4. Weighted random forests perform significantly better than regular random forests |
| (Xie, Li, Ngai, & Ying, 2009) | Customer churn prediction using improved balanced random forests | 1. Artificial Neural Network<br>2. Decision Tree<br>3. Support Vector Machine<br>4. Improved | 1. Balanced Random Forests<br>2. Weighted Random Forests | IBRF has a better accuracy and top-decile left |

| | | | | |
|---|---|---|---|---|
| | | Balanced Random Forest (IBRF) | | |
| (Longadge, Dongre, & Malik, 2013) | Class imbalance in data mining | 1. AdaBoost<br>2. AdaBoost.NC<br>3. SVM | 1. Random Under-sampling<br>2. SMOTE | 1. Boosting improves the performance of weak classifiers<br><br>2. Hybrid techniques (applying two or more techniques) improve performance |
| (Prasasti & Ohwada, 2016) | Machine Learning techniques for customer defection | 1. Multiple Perceptron (MLP) Neural Network<br>2. J48 Decision Tree<br>3. Sequential Minimal Optimization (SMO) Support Vector Machine<br>Random Forest | | 1. Performance of algorithms differed based on characteristics and type of data<br><br>2. J48 Decision Tree and SMO Support Vector Machine had more stable results across datasets |

Table 3.4: Summary of literature review on treatment of class imbalance and customer churn prediction

We used the findings from the literature review to finalize the following aspects of our model:

1. Algorithms selected:
    a. Random Forest Classifier
    b. Gradient Boosting Classifier
2. Techniques to treat class imbalance:
    a. Resampling using SMOTE
    b. Cost-sensitive learning (assigning class weights)
    c. Combination of the two
3. Model Evaluation metrics:
    a. Precision
    b. Recall
    c. F1 score: Harmonic mean of Precision and Recall

While studies have been conducted on treatment of imbalance in classes, the response variable in the datasets were binary. We study impact of imbalance class treatment in a multi-class classification setting.

## DATA

### A. PROPRIETARY DATA

The data used in this project came from our partner company. The data set consists of the attributes of all the projects undertaken by them over the past one year and has approximately 300,000 observations. The database has various tables that capture information regarding the projects, customers, the product being fixed, leads, sources of leads, partner company's employees, representatives that are involved in the project, so on and so forth. Features of a few tables are discussed below:

**Project Information:** The projects table lists all the features related to project timelines, price, current state, payment mode, financial status and owner signatures.

The target variable for our study is the variable "current state", which is classified into four types:

1. Active: Current on-going projects (whose eventual project status we wish to predict)
2. Cancelled: Projects where the customer reneges on a signed contract
3. Closed: Projects that were approved and executed successfully
4. Declined: Projects where the partner company does not approve a customer's project proposal

For building our model we are concerned with projects that are either Cancelled, Closed or Declined.

**Product Information:** This table captures descriptive attributes of all the products in the partner company's database that have been installed or are currently marked as "Active" projects.

**Customer Information:** Customer data such as address, age, credit card scores, and so on are captured here.

**Leads and Lead sources:** This table lists all the past and potential customers of the client and the sources through which these customers were approached. The table gives useful insights on which

customer segment is targeted by the partner company and the marketing channels used to approach them.

**Partner company's Representatives:** This table records the ID's, role and starting year of the partner company's representatives who interact with the customers. We used the information from this table in combination with the information in the projects table to discern if certain representatives are more efficient and have higher a conversion ratio than others.

B. PUBLIC DATA

Apart from the data provided to us by the partner company, we also collected publicly available zip code level demographic data such income level, unemployment rate, education level and population. The purpose of collecting this data was to create clusters of zip codes that represent similar kind of people.

The motivation behind collecting this data was to explore the possibility of identifying behavioral patterns across different zip codes such as project cancellation rate. We hoped to then uncover underlying characteristics of customers within these clusters which could explain reasons for the cancellations.

## METHODOLOGY

Our study is divided into 4 distinct phases:

A. Data exploration and hypothesis development
B. Data cleaning and pre-processing
C. Model building
D. Model evaluation and comparison

Figure 5.1 (below) illustrates this process flow.

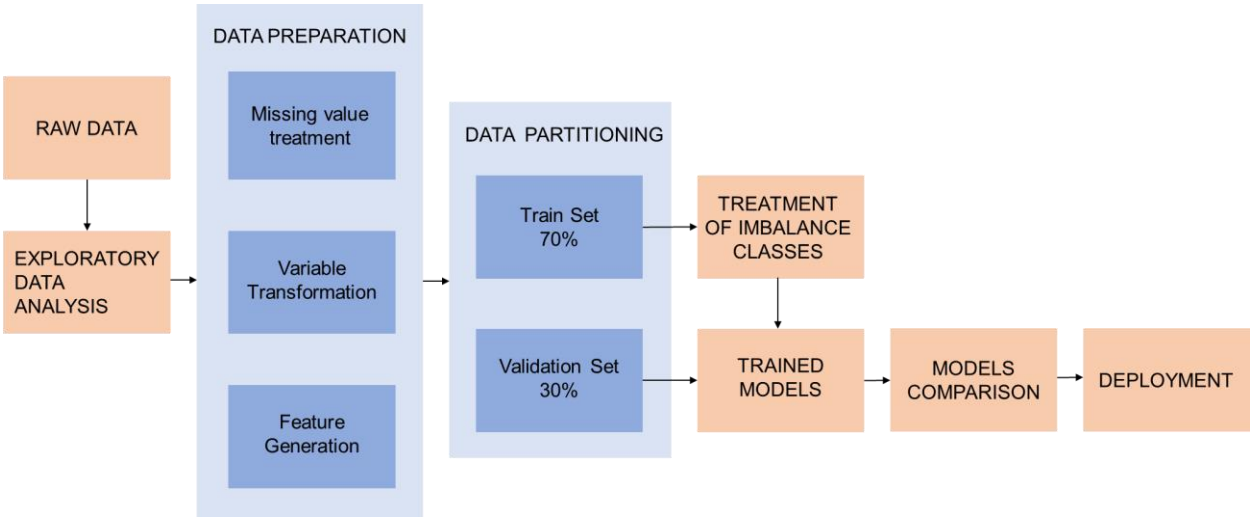


Figure 5.1: Methodology

## A. DATA EXPLORATION AND HYPOTHESIS DEVELOPMENT

### Exploratory Data Analysis

We first explored the data to understand the following:

- Interrelationship between tables
- Interrelationship between features
- Distribution and fill-rate of features
- Associations between predictors as well as between the response variable ("current status") and predictors

### Hypothesis Generation

After understanding the data, we developed hypotheses to understand the predictors and the associations between the predictors and response variable better. They were as follows:

- $H_1$: Do projects which have a higher discounted price have lower cancellation rates?
- $H_2$: Do projects in certain cluster of locations have more cancellations than other locations?
- $H_3$: Do representatives with higher conversion rates have lower project cancellation rates?

## B. DATA CLEANING AND PRE-PROCESSING

During this stage we performed the following tasks:

- Data validation: Ensuring the correctness and relevance of data; identifying and treating outliers
- Treatment of missing values and nulls
- Eliminating features with high correlations or near zero variance
- Variable transformations such as encoding and standardizing features data

**Feature Generation**: During the feature engineering phase we created several features which could be directly inputted into the model and checked for significance. The features created were linked to the hypotheses we generated as well as to account for neighborhood effect (where the response of customers is based on external influences that affect their decisions).

Some of the features we created are:

- Offered Price Ratio: Comparison of Offered Price with the Market Price (a measure of discount offered to the customer)
- Cluster of zip codes: Clustering based on publicly available, zip code level data sources on income, population and unemployment rates
- Conversion Ratio: A metric to measure the performance of the partner company's representatives
- Referral count: Number of referrals a customer received

C. MODEL BUILDING

**Data Partition**

We used the validation set approach, partitioning our data 70-30% into train and validation sets respectively.

**Treatment of Class Imbalance**

The imbalance between projects that are Cancelled, Closed or Declined is treated using the following techniques:

1. Resampling using Synthetic Minority Over-Sampling Technique (SMOTE):
2. Assigning class weights (cost-sensitive learning)

3. A combination of the two

The treatment of class imbalance using the SMOTE resampling technique is performed on the train set only and not the validation set. This is done for the following reason:

- In the SMOTE algorithm, which is used for over-sampling the minority classes, k-nearest neighbors for the minority class are identified and synthetic observations of the minority class are created on the line joining any or all the k nearest class neighbors.
- If SMOTE is performed on the entire dataset, information from the validation set would bleed into the train set, thereby inflating the precision and recall of the model.

**Algorithm Selection**

We selected the Random Forest classifier and the Gradient Boosting classifier to train the models for this multi-class classification problem.

D. MODEL EVALUATION

Once the models are built, they are evaluated based on certain parameters so that one of them can be picked to be the final model. Since this is a multiclassification problem where there is a class imbalance, we use Precision and Recall as the evaluation metrics. Precision and Recall are defined and can be interpreted as follows (Larose & Larose, Data Mining and Predictive Analytics, 2015):

| Evaluation Metric | Formula | Interpretation |
|---|---|---|
| Precision | $\frac{(True\ Positive)}{(True\ Positive + False\ Positive)}$ | When the model predicts an instance to fall within a class, how often does it actually fall within the class.<br><br>A Precision of 0.85 means that out of the 100 times the model classifies the project as falling within a particular project status, 85 times the model would be correct. |
| Recall | $\frac{(True\ Positive)}{(True\ Positive + False\ Negative)}$ | When an instance actually falls within a class, how often does the model correctly classify it as falling in this class.<br><br>A Recall of 0.85 means that out of the 100 projects that fall within a given project status, the model correctly classifies that 85 of them would fall within that project status. |

Table 5.1: Formulae and interpretation of precision and recall

**Cost Matrix**

Additionally, to compare the models based on their business impact, we built a cost matrix to quantify the gain or loss of correctly classifying or misclassifying a project's status:

| | | Predicted Status | | |
|---|---|---|---|---|
| | | Cancelled | Closed | Declined |
| Actual Status | Cancelled | $3,700 | -$500 | $0 |
| | Closed | -$500 | $0 | $0 |
| | Declined | -$500 | -$40 | $40 |

Table 5.2: Model evaluation cost matrix

These costs were based on the assumption that out of all the projects our model would identify as being eventually "Cancelled", 10% could be saved by taking pre-emptive measures. While formulae behind the cost matrix cannot be disclosed, some elements of the matrix are discussed below:

- Columns are the project statuses as predicted by the model
- Rows are the actual project statuses
- If our model classifies a project as "Cancelled" and it eventually does get cancelled, there is a gain since:
    - 10% of these projects could get saved. Hence, the revenue from these projects would get realized.
    - The partner company can be cautious about deploying resources once the customer signs the contract, hence saving money and resources in the eventuality that the customer reneges at a later stage. The representatives of the partner company could get reassurance from the customer that they indeed want to proceed with the project before the company proceeds with planning the installation projects.
    - We have accounted for the opportunity loss of projects that could not be saved even after representatives of the partner company connect with the customer.
- If our model classifies as project as "Cancelled" and it eventually gets "Closed" or "Declined" there's an opportunity loss of having sent a representative to the customer to save the business.

- If our model classifies a project as “Declined” and it eventually gets “Cancelled” or “Closed” there isn’t any additional gain or loss as we recommend that the company doesn’t take any action before approving the project.

**Cost-Saving Per Project**

For each model, a 3 x 3 Confusion matrix is generated: ($\sum_{i=1}^{3}\sum_{j=1}^{3} CF_{i\,x\,j}$)

The Cost Matrix is given by table 5.2

Hence, we calculate the cost saving per project for each model, using the following formula:

$$\text{Cost Saving Per Project} = \sum_{i=1}^{3}\sum_{j=1}^{3} \left(\frac{(C_{i\,x\,j}) \times (CF_{i\,x\,j})}{Number\ of\ Projects\ (N)}\right)$$

Finally, we pick a model that has the best performing evaluation metrics across the three classes and helps our industry partner save the highest potential revenue and cost by correctly classifying the projects that would close, get declined by management, or get cancelled by customers.

E. DEPLOYMENT

The final step is deploying our model into production and predicting the status of active projects.

## MODELS

Our study requires us to classify a project as one of three eventual statuses:

1. Cancelled: Projects where the customer reneges on a signed contract
2. Closed: Projects that were approved and executed successfully
3. Declined: Projects where the partner company does not approve a customer’s project proposal

This is a multiclassification problem and we use the following two machine learning techniques to solve them.

**Random Forests:**

Random forest is a learning technique that consists of bagging un-pruned decision trees with a randomized selection of features at each split. Initially it draws n_tree bootstrap samples from the original data. For each bootstrap sample, it grows an un-pruned classification or regression tree.

Finally, the class which has the most number of votes across all trees in the forest, is used to classify the case (Breiman, 2001).

**Gradient Boosting:**

This is an ensemble technique that starts with weak learners, usually decision trees, and combines them into a single stronger learner (Brownlee, 2016). Once the initial weak model makes a prediction, subsequent boosting steps predict the error residuals. These error residuals are minimized using the gradient decent approach. Hyperparameters specific to this algorithm can tune the individual trees or manage the boosting procedure according to requirements (Jain, 2016). These can be optimized using a grid search or a randomized search. Finally, the algorithm uses a weighted sum of the predictions to provide an overall prediction (Gorman, 2017).

## RESULTS

FINDINGS FROM EXPLORATORY DATA ANALYSIS

Some interesting findings from the exploratory data analysis are discussed in this section.

**Imbalanced Distribution of Project Statuses**

As discussed earlier, we eliminate "Active" projects from our database of the over 300,000 projects while building our models. The projects that we are interested in studying are either "Cancelled", "Closed" or "Declined". The pie chart in figure 7.1 illustrates the distribution of the projects across these three classes (project statuses).

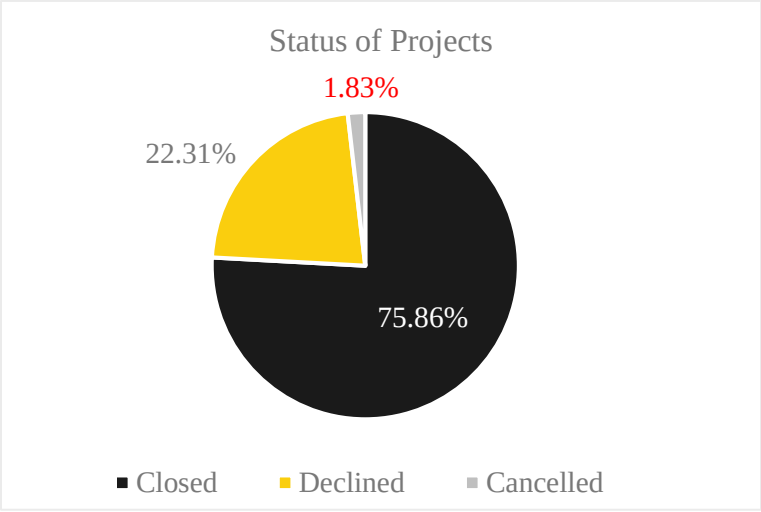

Figure 7.1: Class distribution across projects

As visible, one can see a clear imbalance in data across the three classes. Cancelled projects form only 1.83% of all the projects. Hence, the decision to treat the class imbalance before feeding the data into the Random Forest and Gradient Boosting classifiers.

**Relationship Between Price Discounts Offered and Project Statuses: Tougher the customer, larger the discounts.**

We developed a metric "Offered Price Ratio" to measure the price offered to the customers compared to the market price. It was interesting to note that the customers who were offered the most discounts (i.e. lowest price ratio of 64% of the market price) most frequently cancelled the projects. A possible explanation for this could be that the partner company's representatives offered the greatest discounts to unwilling customers during their pitch meetings as an added incentive to sign the project's contract. This is visible from figure 7.2.

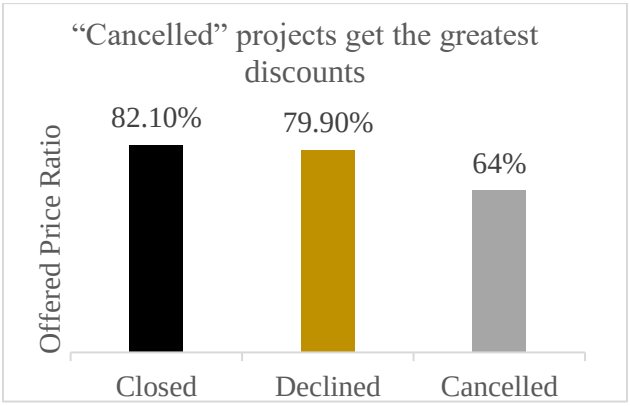


Figure 7.2: Relationship between price discounts offered and project statuses

**Clusters of zip codes**

We used the k-means algorithm to create a cluster of zip codes that we could use to test whether certain cluster of locations have more cancellations/declines/closed projects than other locations. The external income, population, unemployment rate data used to create the clusters was standardized and then fed into the k-means algorithm. Based on the elbow curve (figure 7.3) we created 5 clusters (figure 7.4). These clusters of zip codes were used as inputs in our models.

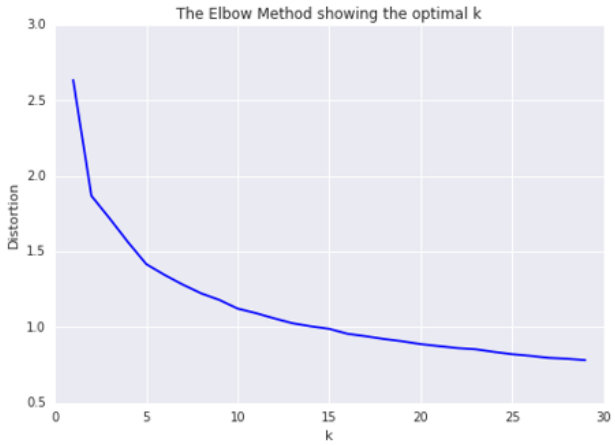


Figure 7.3: Elbow curve method to select the optimal k

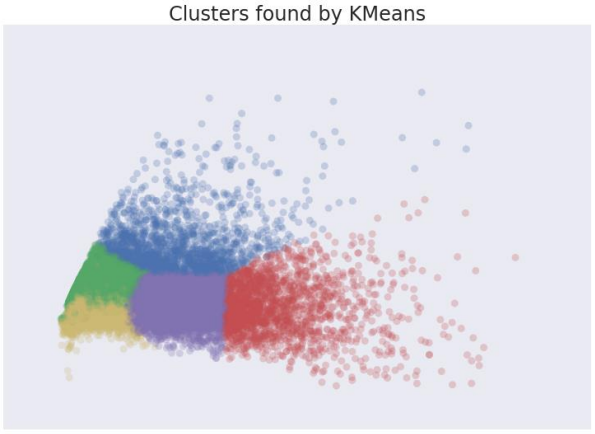


Figure 7.4: 5 clusters of zip codes

RESULTS OF MODELING

We ran the SMOTE algorithm on the Train set. The three types of up-sampling we tested are:

1. Auto: Over-sample all classes to match the majority class
2. Minority: Over-sample the minority class to match the majority class
3. Custom over-sampling using dictionary as an argument

The distribution of classes in the train set, before and after SMOTE is illustrated by the table below.

| SMOTE Technique | Description | Cancelled | Closed | Declined |
|---|---|---|---|---|
| None | No resampling | 1.82% | 75.85% | 22.32% |
| Auto | Over-sample all classes to match the majority class | 33.33% | 33.33% | 33.33% |
| Minority | Over-sample the minority class to match the majority class | 43.59% | 43.59% | 12.83% |
| Custom | Cancellation: Over-sample by 1200%<br>Declined: Over-sample by 115% | 16.23% | 51.31% | 32.46% |

Table 7.1: Class distribution before and after resampling using SMOTE

**Variable Importance**

Along with correctly classifying projects according to the three project statuses, it is also important to identify features which form good identifiers of the final status of a project. Our findings are summarized below:

- "Approval Time" is a feature that measures the time taken by the partner company to approve or reject a project proposal. During an initial iteration of the model, we found "Approval Time" to have the largest impact on the status of the project, explaining 70% of the variation in the data. This finding was in accordance with the hypothesis that, the greater the time spent to approve a project, larger the impact on its status. However, we needed to explore this variable further and more importantly, we realize that approval time isn't always a controllable factor. Hence, we had to drop it from our final model.
- For our final dataset, we found variables pertaining to price, such as "price", "retail price" and so on dominate the model. "Offered Price Ratio" (measure of discount provided to customers) was a good indicator of a project's final status, explaining up to 12% of the variation.
- We also found that attributes related to the partner company's representatives such as experience, conversion ratio and so on, are important identifiers in the models.

**Model Evaluation using Precision and Recall scores**

The models are evaluated based on their Precision and Recall scores as well as the cost savings per project. The models and their Precision and Recall scores for the validation set can be seen from the table 7.2 below.

| Classification Algorithm | SMOTE Ratio | Class Weight | Class | Precision | Recall | F1 |
|---|---|---|---|---|---|---|
| Random Forest Classifier | None | None | Cancelled | 0.96 | 0.58 | 0.72 |
| | | | Closed | 0.80 | 0.96 | 0.87 |
| | | | Declined | 0.58 | 0.20 | 0.30 |
| Random Forest Classifier | Auto | None | Cancelled | 0.86 | 0.58 | 0.69 |
| | | | Closed | 0.81 | 0.89 | 0.85 |
| | | | Declined | 0.44 | 0.29 | 0.35 |
| Random Forest Classifier | Minority | None | Cancelled | 0.87 | 0.58 | 0.70 |
| | | | Closed | 0.79 | 0.96 | 0.87 |
| | | | Declined | 0.54 | 0.17 | 0.26 |
| Random Forest Classifier | Custom | None | Cancelled | 0.91 | 0.58 | 0.71 |
| | | | Closed | 0.80 | 0.91 | 0.85 |
| | | | Declined | 0.46 | 0.27 | 0.34 |
| Random Forest Classifier | None | Balanced | Cancelled | 0.88 | 0.58 | 0.70 |
| | | | Closed | 0.80 | 0.91 | 0.85 |
| | | | Declined | 0.44 | 0.25 | 0.32 |
| Random Forest Classifier | None | Custom | Cancelled | 0.89 | 0.58 | 0.70 |
| | | | Closed | 0.80 | 0.91 | 0.85 |
| | | | Declined | 0.45 | 0.25 | 0.32 |
| Random Forest Classifier | Auto | Balanced | Cancelled | 0.84 | 0.58 | 0.69 |
| | | | Closed | 0.81 | 0.88 | 0.84 |
| | | | Declined | 0.42 | 0.31 | 0.35 |
| Random Forest Classifier | Auto | Custom | Cancelled | 0.82 | 0.58 | 0.68 |
| | | | Closed | 0.81 | 0.87 | 0.84 |
| | | | Declined | 0.41 | 0.30 | 0.35 |
| Gradient Boosting Classifier | None | None | Cancelled | 0.96 | 0.58 | 0.72 |
| | | | Closed | 0.84 | 0.95 | 0.89 |
| | | | Declined | 0.69 | 0.42 | 0.52 |
| Gradient Boosting Classifier | Auto | None | Cancelled | 0.94 | 0.55 | 0.69 |
| | | | Closed | 0.83 | 0.94 | 0.88 |
| | | | Declined | 0.63 | 0.37 | 0.47 |
| Gradient Boosting Classifier | Custom | None | Cancelled | 0.96 | 0.56 | 0.71 |
| | | | Closed | 0.84 | 0.94 | 0.89 |
| | | | Declined | 0.66 | 0.40 | 0.50 |

Table 7.2: Precision and Recall scores of the models

Summary of findings from table 7.2:

- The Gradient Boosting classification models outperform the Random Forest classification models, irrespective of the class imbalance treatment.
- The base models perform better than the models where class imbalance has been treated.
- SMOTE findings across both the classifiers:
    - Over-sampling the minority class (“Cancellation”) to match the class size of the majority class (“Closed”) degrades the precision and recall scores of the minority class.
    - Precision and Recall scores for Custom SMOTE over-sampling are better than “Auto” or “Minority” SMOTE over-sampling.
- Precision scores for the projects that are “Cancelled” are high across the models. This implies that the likelihood of the projects that are predicted to be cancelled actually being cancelled is high. Hence, our final model can be used by the partner company to save the projects that are predicted to be cancelled.

- Recall scores for the projects that are "Cancelled" need to be further improved. A significant number of projects where the customer eventually reneged on the signed contract are being predicted to be "Closed" by our model.

**Model Evaluation using Cost Savings Per Project**

The combinations of models we ran and the cost saving per project for each of them can be seen from table 7.3.

| Classification Technique | SMOTE | Class Weight | Cost Saving Per Project |
|---|---|---|---|
| Random Forest | None | None | $30.99 |
| Random Forest | Auto | None | $32.22 |
| Random Forest | Minority | None | $29.53 |
| Random Forest | Custom | None | $31.97 |
| Random Forest | None | Balanced | $32.39 |
| Random Forest | None | Custom | $32.22 |
| Random Forest | Auto | Balanced | $32.61 |
| Random Forest | Auto | Custom | $32.60 |
| Gradient Boosting | None | None | $35.44 |
| Gradient Boosting | Auto | None | $32.44 |
| Gradient Boosting | Custom | None | $34.09 |

Table 7.3: Comparison of models based on cost savings per project

We observe from the table above that while SMOTE and class weights increased the cost-savings for the Random Forest Classifier (in isolation as well as in unison), the base model of the Gradient Boosting classifier outperformed all the models and performed better without the treatment of class imbalance.

**Final Model Selection: Gradient Boosting Classifier (Base model)**

The Gradient Boosting classifier's base model (no class imbalance treatment) has the best precision and recall scores as well as the highest cost savings per project.

| Classification Technique | Classes | SMOTE | Class Weight | Cost Saving Per Project | Precision | Recall | F1 |
|---|---|---|---|---|---|---|---|
| Gradient Boosting | Cancelled | None | None | $35.44 | 0.96 | 0.58 | 0.72 |
| | Closed | | | | 0.84 | 0.95 | 0.89 |
| | Declined | | | | 0.69 | 0.42 | 0.52 |

Table 7.4: Performance of the Gradient Boosting classifier (base model)

The summary of this model's performance can be seen in table 7.4 above:

- Since the Precision of 'cancelled' is high (96%), if a project is classified as 'cancelled' it is highly likely to actually be cancelled
- But since the Recall value is low (58%) it implies that only 58% of the cancels were predicted out of all the cancelled projects
- 'Closed' has precision of 84% and recall of 95% implying that the model mostly predicts 'Closed' for most of the projects. This could also indicate an imbalance.

On an average, our partner company processes over 300,000 projects annually. By deploying our best performing model, which has a cost saving of $35.44 per project, our partner company can save $10.63 million per annum.

| Classification Technique | SMOTE | Class Weight | Annual Cost Saving |
|---|---|---|---|
| Gradient Boosting | None | None | *$10.63 Million* |

## CONCLUSION

In this study, we develop a model to predict if a project undertaken by our industry partner would successfully get completed, get declined by the company, or if the customer would renege on the contract and cancel the project. Along with predicting the status of projects we were able to identify key features that determine the status of a project.

We use a Random Forest classifier and a Gradient Boosting classifier for this multi-class classification problem. The imbalance in classes is treated using SMOTE, setting class weights, and a combination of the two. We find that over-sampling the minority class ("Cancellation") using SMOTE to match the class size of the majority class ("Closed") degrades the precision and recall scores of the minority class. We find more encouraging results when we define a custom class distribution. It would be interesting to optimize the customized selection of class distribution, so that the base classifiers are beaten by models where the imbalance classes are treated.

The models are evaluated by comparing the potential revenue and costs they save as well as the precision and recall scores of predictions. The precision and recall scores of the highest cost saving model is also the highest amongst the models developed.

By deploying our best performing model (Gradient Boosting classifier without any treatment for class imbalance), our industry partner can **save $10.63 million annually**.

## References


Bohanec, M., Robnik-Šikonja, M., & Borštnar, M. K. (2017). Organizational Learning Supported by Machine Learning Models Coupled with General Explanation Methods: A Case of B2B Sales Forecasting. *Organizacija 50(3)*.

Breiman, L. (2001). Random Forests. *Machine Learning*, 5-32.

Brownlee, J. (2016, September 9). *A Gentle Introduction to the Gradient Boosting Algorithm for Machine Learning*. Retrieved from Machinelearningmastery.com: https://machinelearningmastery.com/gentle-introduction-gradient-boosting-algorithm-machine-learning/

Burez, J., & Van den Poel, D. (2009). Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, 4626 – 4636.

Caruana, R., & Niculescu-Mizil, A. (2006). An Empirical Comparison of Supervised Learning Algorithms. *23rd International Conference on Machine Learning*. Pittsburg, PA.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 321-357.

Chen, C., Liaw, A., & Breiman, L. (2004). *Using Random Forest to Learn Imbalanced Data*. University of Berkeley.

Duran, R. (2008). "Probabilistic Sales Forecasting for Small and Medium-Size Business Operations." . *Soft Computing Applications in Business*, pp. 129-146.

Gorman, B. (2017, January 23). *A Kaggle Master Explains Gradient Boosting*. Retrieved from Blog.Kaggle.com: http://blog.kaggle.com/2017/01/23/a-kaggle-master-explains-gradient-boosting/

Huang, H.-C., Chang, A. Y., & Ho, C.-C. (2013). Using Artificial Neural Networks to Establish a Customer-cancellation Prediction Model. *PRZEGLĄD ELEKTROTECHNICZNY*, pp. 178-180.

Jain, A. (2016, February 21). *Complete Guide to Parameter Tuning in Gradient Boosting (GBM) in Python*. Retrieved from Analyticsvidhya.com: https://www.analyticsvidhya.com/blog/2016/02/complete-guide-parameter-tuning-gradient-boosting-gbm-python/

Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. *Informatica 31*, 249 - 268.

Larose, D. T., & Larose, C. D. (2015). *Data Mining and Predictive Analytics.* John Wiley & Sons, Inc.

Larose, D. T., & Larose, C. D. (2015). Neural Networks. In D. T. Larose, & C. D. Larose, *Data Mining and Predictive Analytics* (pp. 339 - 358). John Wiley & Sons, Inc.

Lodato, M. W., & M. W. Lodato. (2006). Integrated sales process management: a methodology for improving sales effectiveness in the 21st century. *AuthorHouse*.

Longadge, R., Dongre, S. S., & Malik, D. (2013). Class Imbalance Problem in Data Mining: Review. *International Journal of Computer Science and Network (IJCSN)*.

Prasasti, N., & Ohwada, H. (2016). Applicability of Machine-Learning Techniques in Predicting Customer Defection.

Sahraoui, A., & Elarref, M. (2014). "Bed crisis and elective surgery late cancellations: An approach using the theory of constraints.". *Qatar medical journal: 1*.

Söhnchen, F., & Albers, S. (2010). Pipeline management for the acquisition of industrial projects. *Industrial Marketing Management 39(8)*, 1356-1364.

Tang, Y., Zhang, Y.-Q., Chawla, N. V., & Krasser, S. (2009). SVMs Modeling for Highly Imbalanced Classification. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS*, 281-288.

Xie, Y., Li, X., Ngai, E., & Ying, W. (2009). Customer churn prediction using improved balanced random forests. *Expert Systems with Applications*, 5445–5449.

Yan, J., Zhang, C., Zha, H., Gong, M., Sun, C., Huang, J., . . . Yang, X. (2015). On Machine Learning towards Predictive Sales Pipeline Analytics∗. *Twenty-Ninth AAAI Conference on Artificial Intelligence*, (pp. 1945 - 1951).

Yeh, I.-C., & Lien, C.-h. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications 36* , 2473–2480.

Zhao, Y., Li, B., Li, X., Liu, W., & Ren, S. (2005). Customer churn prediction using improved one-class support vector machine. *Lecture Notes in Computer Science*, pp. 300-306.