Paper 4051-2019
Asking the Right Questions. A Study in Multiclass Imbalanced Classification for H1-B Visas

Matthew Bunker and Owen Kelly, Kennesaw State University

ABSTRACT

A shortage of skilled resident US workers has resulted in the need to import workers from other countries. The H1-B visa lottery system helps fill these needs. This paper investigates sampling methods that can be used to predict the outcome of an H1-B application with an aim to contain application processing costs for both government and private industry. It specifically investigates how the imbalanced nature of the outcome of cases can adversely affect the predictive power of differing models and ways to address this. Reclassifying the target into different binary variables for government and private industry showed benefits of different modeling techniques. The best model for the Government focuses on forecasting withdrawals with a Balanced Gradient Boosting model. For Private industry the best model focuses on predicting Certified results with an 80% Train Decision Tree model.

INTRODUCTION

In order to meet market's demand for skilled labor, American companies employ 85,000 foreign workers each year using H1-B visas (Jordan, 2018). To receive these visas employers, or potential employees, file a petition for a 3-year visa which can be extended for an additional 3 years. If the petition is approved the potential employee is entered into a lottery system for a chance to receive a visa. Due to the lottery aspect of the visa any money spent on the process may be wasted. Petition fees for a visa can cost between $1,250-$2,000 (Our Fees, 2018). This doesn't include the cost of an attorney or potential recruitment fees. Since the United States is tightening the requirements on accepted petitions predicting the likelihood of acceptance will prove important in reducing costs to business. Similarly, with the high number of petitions being submitted yearly, the Government may benefit from understanding which cases will be withdrawn.

Historically H1-B visa petitions are primarily certified which leads to imbalanced datasets. Such imbalanced datasets come with challenges when modeling predictive behavior. This paper analyzes different sampling methods to increase the combined precision and accuracy of the classification models.

DATA

The dataset *H-1B Visa Applications – 2017* (H-1B Visa Applications - 2017, 2018) comes from the Department of Labor and describes 624,650 different petitions (1 per row) for H-1B visas during the year of 2017. Each petition has 52 variables including items such as employer, SOC Job code, location of employment, worksite location, prevailing wage, and wage rate. A full list can be found in the data dictionary of the appendix (Table 1). The case_status (Case Status) variable is of interest when it comes to predictions. There are 4 different statuses a case can have. These are withdrawn, certified, certified-withdrawn, and denied. Certified cases are cases approved for entry in the lottery. Withdrawn cases are cases that the employer or potential employee withdrew prior to receiving a response. Certified-Withdrawn cases are cases that are certified but withdrawn by the employer or potential employee. Denied cases are cases that are denied participation in the lottery.

THE PROBLEM

One problem seen in this dataset is a class imbalance within the target variable Case Status. The data distribution has an imbalance of: 87.36% Certified, 7.96% Certified-Withdrawn, 1.36% Denied, and 3.33% Withdrawn. Due to this severe imbalance models

tend to automatically classify all targets as the majority class. Our goal is to investigate sampling methods and target variable binarization to improve model performance.

One method of reducing the effects of class imbalance is the use of inverse weights to allow correct classification of the minor classes. This makes them worth more in the model. Another option is to utilize a biased sampling technique that creates a balanced data set.

Because Case Status can result in 4 possible outcomes: withdrawn, certified, certified-withdrawn, and denied, multiple questions can be asked. For government and private industry, two different questions would be of particular interest -- one related to government concerns and one for those of private industry. Government benefits from understanding which applications result in withdrawals as these are as a waste of time for their employees. Whereas private industry benefits more from understanding what causes a petition to be certified. The question then becomes should the target variable, Case Status, be changed from a multiclass variable to a binary one according to the entity requesting the results?

The chosen approach to account for all concerns involves testing the effects of an 80% Training set, a smaller balanced set, and an inverse weighted set on the original target, a binary target based on certification, and a binary target variable based on case withdrawal. The classification models we would include in the comparison include Classification Tree, Regression (with stepwise variable selection), Gradient Boosting, and Neural Networks.

DATA CLEANING & VALIDATION

During the data cleanup process, no records are removed from the dataset. There are concerns that incomplete forms or incorrect data for each submission contribute to Case Status outcome (Withdrawn or Denied). The variables used are described in the data dictionary (Table 1). The following target variables are created to assist with proper outcome prediction:

> **BiW_Status** – This binary variable converts the status variable into a binary variable favorable for government. Whether or not the application ends up as Withdrawn is of interest. Withdrawn and Certified-Withdrawn are classed as Withdrawn. Certified and Denied are classed as Not Withdrawn.
>
> **BiC_Status** – This binary variable converts the status variable into a binary variable favorable for industry. Whether or not the application ends up as Certified is of interest. Certified and Certified-Withdrawn are classed as Certified. Withdrawn and Denied are classed as Not Certified.

ANALYSIS

SAS®

Using SAS® the dataset is cleaned as described in the cleaning section. Then the following datasets are created. Data is sampled into an 80% Training Set and a 20% Validation Set stratified by case_status. From here sample sets are created in the following manner:

- 80% Training – Stratified 80% of the full data set. This is used to create the Balanced and Inverse weight set further.
- 20% Validation – The remaining 20% of the data set is used for validation for all models.
- Balanced – the 80% Training set is sampled into 4 equally sized selections representing the 4 output target classes. Each grouping is a collection of 5000 observations randomly selected using a uniform distribution.
- Inverse Weight – no changes are made to the 80% Training Set in SAS®. In SAS® Enterprise Miner™ the Decision Matrix is set to inverse prior weights (Table 2).

SAS® ENTERPRISE MINER™

Each of the data sets is imported into SAS® Enterprise Miner™ and trained using the following models: HP Tree, HP Regression (Stepwise), HP Gradient Boost, and HP Neural Network. Results are compared across the different models to create model-independent conclusions on sampling. Figure 1 below shows the workflow for one of the target variables.

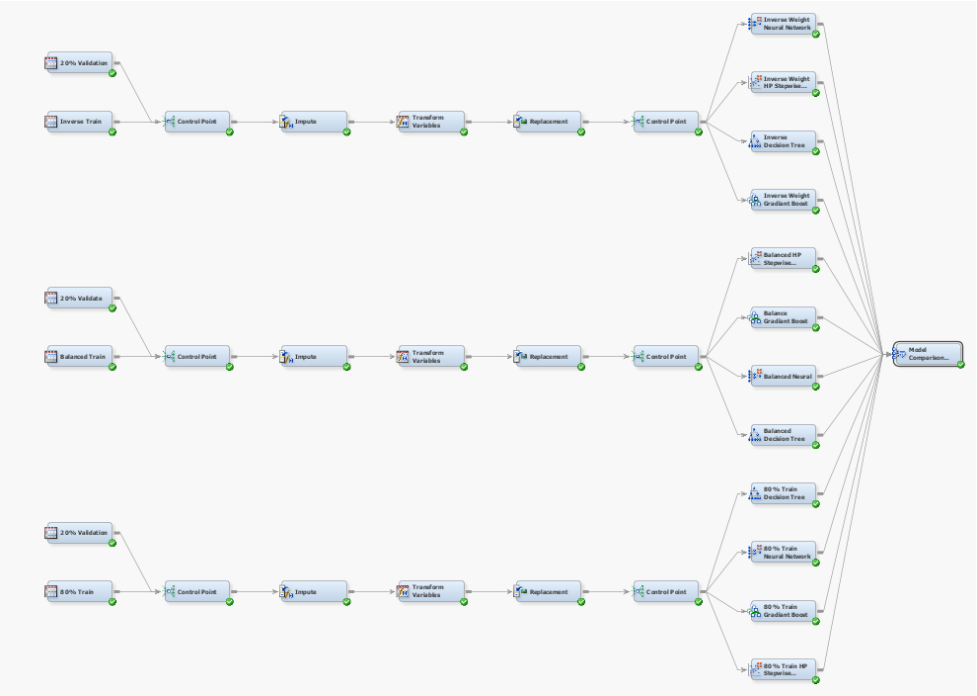

Figure 1 - SAS® Enterprise Miner™ Overall Diagram for One of the Target Variables (Case Status)

The following steps are taken to create the models:

1. The chosen training set is imported from SAS® and attached to a control point.
2. The chosen target variable is selection (Case Status, Binary Withdraw Status, or Binary Certified Status).
3. The 20% Validation set is imported from SAS® and attached to the same control point.
4. Final data cleaning steps are completed:
    a. Imputation node: The Calculated Prevailing wage is imputed to the median for missing values. Categorical variables with missing values are imputed to a “missing” class.
    b. Transformation node: Due to their skewness, the Calculated Prevailing Wage and Calculated Wage Rate are log transformed (Before and After histograms can be seen in Figure 2, Figure 3, Figure 4, Figure 5).
    c. Replacement node: Set the outlier for Calculated Prevailing Wage and Calculated Wage rate to the mean value
5. Final control point is made before sending through the 4 models (Decision Tree, HP Regression with stepwise variable selection, HP Gradient Boost, and HP Neural Network).
    a. Defaults are used for all settings except the Decision Tree for the Inverse weight is set to use decisions in split searches.
6. All of the models for this target variable are connected to a single model comparison node
7. Steps 1-6 are completed for each target variable and each training set.

The models are compared using the misclassification, average precision, and average recall of each model. They are also evaluated using the comparison node as Average Recall and Average Precision for each target class.

RESULTS

A holistic comparison is created using the various modeling methods to discover the best sampling method. In addition, both the Binary Certified target variable and the Binary Withdrawn target variable are compared against the multiclass target (Case Status).

The Gradient Boost model in Table 3 shows the best sampling method for precision and recall, regardless of target variable, is the Balanced sampling method. For Case Status average precision is 0.525 and average recall is 0.614. The Binary Certified target method shows no improvement for recall or precision, but the Binary Withdrawn target shows an increase in precision to 0.985 and Recall to 0.876. Misclassification rate changes based on the target variable without a pattern.

The Decision Tree model in Table 4 shows the best overall sampling method for precision, limiting false positives, is 80% Train and for recall. When limiting false negatives, the best is the Inverse sampling method. For Case Status target has highest precision on 80% train (0.873) and recall with Inverse sampling method (0.613). The Binary Certified showed improvements with 80% Train sampling method precision as 0.978 and Inverse with recall 0.623. Binary Withdrawn showed even more improvement with Balanced precision 0.985 (a close second is 80% Train with precision 0.984) and 80% Train recall 0.878 (a close second being Inverse with recall 0.876). The misclassification rate stays the same regardless of model for Case Status target (0.041) and Binary Withdrawn (0.030). For Binary Certified misclassification is best at 0.040 for 80% Train and Inverse sampling methods.

The Neural Network model results in Table 5 show Case Status has least misclassification rate 0.041 and precision 0.738 using the 80% Train sampling method and best recall 0.634 with Inverse sampling method. Values improved for the Binary Certified target 80% Train misclassification rate 0.040 and precision 0.973. Recall did not improve with the Binary Certified Inverse method at 0.634. Binary Withdrawn further improved with 80% Train misclassification 0.030, precision 0.985 and recall 0.876.

The Stepwise Regression model in Table 6 is the lowest performing model. The misclassification and average precision for Case Status is tied at 0.126 and 0.218 respectively for both 80% Train and Inverse sampling methods. The average recall remained at 0.250 for all sampling methods. For Binary Certified the misclassification rate and the precision is tied for 80% Train and Balanced sampling methods at 0.050 and 0.477 respectively, while all the recall results are 0.500. Binary Withdrawn had no difference in precision with all sampling methods at 0.443 and recall with 0.500 for all three sampling methods. One exception is misclassification rate where the 80% Train sampling method is best with 0.110.

GENERALIZATION

Redefining a multiclass target variable as a binary target variable is an effective approach to improve model accuracy. Multiple questions can be answered if each binary target variable is defined to answer a specific question. Domain knowledge may be required to define the questions. Different binary target variables may result in differing optimal models which can require additional processing time. This needs to be considered prior to using binary target variables.

SUGGESTIONS

If the target variable needs to retain all the classes additional investigation can be done with the data. One method to correct the data imbalance is the use of SMOTE with Tomek undersampling (Boardman, Biron, & Rimbey, 2018). This process generates artificial data for the target variable example of a minority target class to help balance the class imbalance within the target variable. This process can be extended to generate additional

examples for multiple minority target classes. Additional data processing needs to be done to convert categorical data into multiple binary classifiers for each class in each categorical variable. To make this process more efficient reductive binning of categorical variables is suggested. The drawback to SMOTE is the use of the MODECLUS procedure. This is not a very efficient procedure which can result in lengthy processing times.

Since this study used the Enterprise Miner default settings for Neural Network, Gradient Boost, and Decision Trees a future study can investigate other values of hyperparameters. For example, the number of hidden nodes and number of nodes in each layer could be increased for Neural Networks. Additionally, with Gradient Boost and Decision Trees the maximum branch, maximum depth, and minimum categorical size can be adjusted to examine different outcomes.

CONCLUSION

In conclusion, it is best to understand what question is being asked and how the results affect the need before starting the problem. The entity evaluating the results matters. In our case this is government or private industry. Most cases showed both Binary Withdrawn and Binary Certified held better in terms of misclassification, average precision, and average recall. In cases where it is important to know the outcome of all classes of the target variable the best case is utilizing the default sampling method (80% Train).

For our proposed problem: Is one model enough? The answer is no. The best-case scenario would be to build separate models for each question. It is important to limit false positives and focus on precision. This will limit the amount of wasted resources on individuals the model falsely predicts as Certified or Not Withdrawn. The best model for the Government predicating withdrawals is the Balanced sampling method with Gradient Boost with misclassification 0.030, average precision 0.985, and average recall 0.876. For private industry the best model predicting Certified results is 80% Train sampling method with Decision Tree model with a misclassification 0.40, precision 0.978, and recall 0.623.

REFERENCES

34.3 Stratified Random Sampling. (2018, November 16). Retrieved from Penn State Eberly College of Science - STAT 482: https://onlinecourses.science.psu.edu/stat482/node/31/

Boardman, J., Biron, K., & Rimbey, R. (2018, May 8). *MITIGATING THE EFFECTS OF CLASS IMBALANCE USING SMOTE AND TOMEK LINK UNDERSAMPLING IN SAS.* Retrieved from SAS: https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2018/3604-2018.pdf

H-1B Visa Applications - 2017. (2018, November 20). Retrieved from Enigma: https://public.enigma.com/datasets/h-1-b-visa-applications-2017/e1ee0ae8-13f4-444f-804e-9a429b32f424

Jordan, M. (2018, April 6). *What Are H-1B Visas, and Do They Hurt American Workers?* Retrieved from The New York Times: https://www.nytimes.com/2018/04/06/us/what-are-h1b-visas.html

Matthews, R. (2018, July 27). *Tips:Program run time*. Retrieved from sasCommunity.org: http://www.sascommunity.org/wiki/Tips:Program_run_time

Our Fees. (2018, November 20). Retrieved from U.S. Citizenship and Immigration Services: https://www.uscis.gov/forms/our-fees

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Matthew Bunker
mbunker@students.kennesaw.edu
Owen Kelly
okelly1@students.kennesaw.edu

APPENDIX

Tables

Variable	Description	Type	Length
case_submitted	Time and date the application is submitted.	Numeric	8
total_workers	Total number of foreign workers requested by the Employer(s).	Numeric	8
change_previous_employment	Indicates requested worker(s) will be continuing employment with same employer without material change to job duties, as defined by USCIS I-29.	Numeric	8
change_employer	Indicates requested worker(s) will begin employment for new employer, using the same classification currently held, as defined by USCIS I-29.	Numeric	8
amended_petition	Indicates requested worker(s) will be continuing employment with same employer with material change to job duties, as defined by USCIS I-29.	Numeric	8
h1b_dependent	Y = Employer is H-1B Dependent; N = Employer is not H-1B Dependent.	Char	1
visa_class	Indicates the type of temporary application submitted for processing. R = H-1B; A = E-3 Australian; C = H-1B1 Chile; S = H-1B1 Singapore. Also referred to as "Program" in prior years.	Char	15
employer_region	Employer requesting temporary labor certification - Corporation/Main Address Region	Char	9
public_disclosure_location	Variables include "Place of Business" or "Place of Employment."	Char	1
full_time_position	Y = Full Time Position; N = Part Time Position.	Char	1
continued_employment	Indicates requested worker(s) will be continuing employment with same employer, as defined by USCIS I-29.	Numeric	8
support_h1b	Y = Employer will use the temporary labor condition application only to support H-1B petitions or extensions of status of exempt H-1B worker(s); N = Employer will not use the temporary labor condition application to support H-1B petitions or extensions of status for exempt H-1B worker(s);	Char	2
new_employment	Indicates requested worker(s) will begin employment for new employer, as defined by USCIS I-29.	Numeric	8
CleanEmployerCountry	Employer requesting temporary labor certification - Corporation/Main Address Country	Char	24
new_concurrent_employment	Indicates requested worker(s) will begin employment with additional employer, as defined by USCIS I-29.	Numeric	8
CalcPrevailingWage	Yearly Prevailing Wage for the job being requested for temporary labor condition.	Numeric	8
worksite_region	Region information of the foreign worker's intended area of employment.	Char	9
CalcWageRate	Maximum proposed yearly wage rate	Numeric	8
willful_violator	Y = Employer has been previously found to be a Willful Violator; N = Employer has not been considered a Willful Violator.	Char	1
SOC_Code_Name	Occupational code associated with the job being requested for temporary labor condition, as classified by the Standard Occupational Classification (SOC) System.	Char	48
labor_con_agree	Y = Employer agrees to the responses to the Labor Condition Statements as in the subsection; N = Employer does not agree to the responses to the Labor Conditions Statements in the subsection.	Char	1
naics_code	NAICS Code of Business	Char	1
Form_Complete	Yes = Complete Form; No = Incomplete Form	Char	3
BiW_Status	Withdrawn = Withdrawn and Certified-Withdrawn Statuses ;Not Withdrawn = Certified and Denied Statuses	Char	15
BiC_Status	Certified = Certified Withdrawn and Certified Statuses; Not Certified = Withdrawn and Denied Statuses	Char	15

Table 1 - Updated Data Dictionary (Original Data Dictionary (H-1B Visa Applications - 2017, 2018))

Decision Matrix	WITHDRAWN	DENIED	CERTIFIED-WITHDRAWN	CERTIFIED
WITHDRAWN	30.03	0	0	0
DENIED	0	72.9927	0	0
CERTIFIED-WITHDRAWN	0	0	12.5945	0
CERTIFIED	0	0	0	1.14469

Table 2 - Inverse Decision Matrix

Gradient Boost

	Case Status			Binary Certified			Binary Withdrawn		
Training Set	Misclassification Rate	Average Precision	Average Recall	Misclassification Rate	Average Precision	Average Recall	Misclassification Rate	Average Precision	Average Recall
80% Train	**0.126**	0.218	0.250	0.126	0.477	0.500	0.110	0.443	0.500
Balanced	0.343	**0.525**	**0.614**	0.240	**0.520**	**0.565**	**0.030**	**0.985**	**0.876**
Inverse	**0.126**	0.218	0.250	**0.050**	0.012	0.250	0.110	0.057	0.500

Best Result Bolded for each target variable

Table 3 - Gradient Boost Results

Decision Tree

	Case Status			Binary Certified			Binary Withdrawn		
Training Set	Misclassification Rate	Average Precision	Average Recall	Misclassification Rate	Average Precision	Average Recall	Misclassification Rate	Average Precision	Average Recall
80% Train	**0.041**	**0.873**	0.540	**0.040**	**0.978**	0.562	**0.030**	0.984	**0.878**
Balanced	**0.041**	0.739	0.538	0.050	0.477	0.500	**0.030**	**0.985**	0.876
Inverse	**0.041**	0.748	**0.613**	**0.040**	0.525	**0.623**	**0.030**	0.973	0.876

Best Result Bolded for each target variable

Table 4 - Decision Tree Results

Neural Network

	Case Status			Binary Certified			Binary Withdrawn		
Training Set	Misclassification Rate	Average Precision	Average Recall	Misclassification Rate	Average Precision	Average Recall	Misclassification Rate	Average Precision	Average Recall
80% Train	**0.041**	**0.738**	0.541	**0.040**	**0.973**	0.557	**0.030**	**0.985**	**0.876**
Balanced	0.051	0.487	0.493	0.090	0.527	0.531	0.040	0.980	0.839
Inverse	0.280	0.554	**0.634**	0.310	0.527	**0.634**	**0.030**	0.557	0.553

Best Result Bolded for each target variable

Table 5 - Neural Network Results

Stepwise Regression

Training Set	Case Status			Binary Certified			Binary Withdrawn		
	Misclassification Rate	Average Precision	Average Recall	Misclassification Rate	Average Precision	Average Recall	Misclassification Rate	Average Precision	Average Recall
80% Train	**0.126**	**0.218**	0.250	**0.050**	**0.477**	**0.500**	**0.110**	**0.443**	**0.500**
Balanced	0.920	0.020	0.250	**0.050**	**0.477**	**0.500**	0.890	**0.443**	**0.500**
Inverse	**0.126**	**0.218**	0.250	0.950	0.023	**0.500**	0.890	**0.443**	**0.500**

Best Result Bolded for each target variable

Table 6 - Stepwise Regression Results

Figures

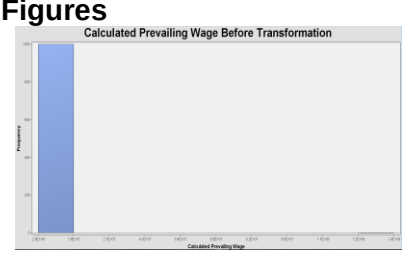

Figure 2 - Calculated Prevailing Wage - Before Transformation - After Replacement

Figure 3 - Calculated Prevailing Wage - After Transformation - After Replacement

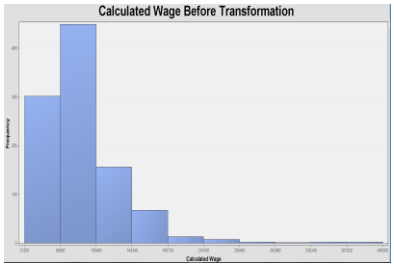

Figure 4 - Calculated Wage - Before Transformation - After Replacement

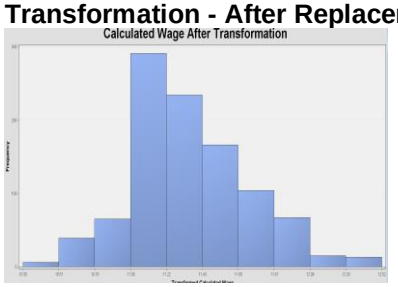

Figure 5 - Calculated Wage - After Transformation - After Replacement

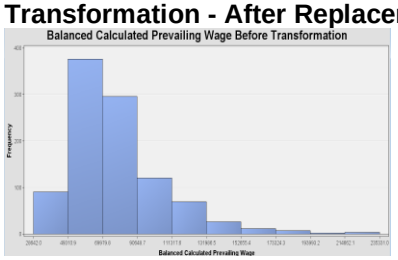

Figure 6 - Balanced Calculated Prevailing Wage - Before Transformation

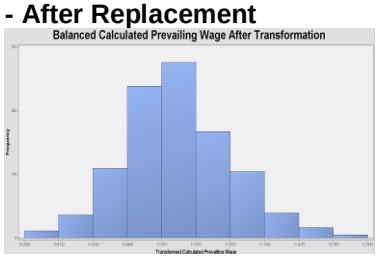

Figure 7 - Balanced Calculated Prevailing Wage - After Transformation

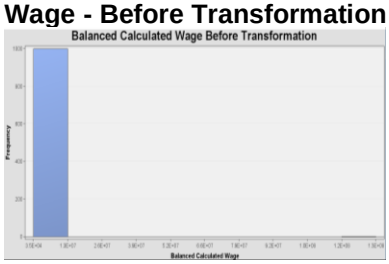

Figure 8 - Balanced Calculated Wage - Before Transformation

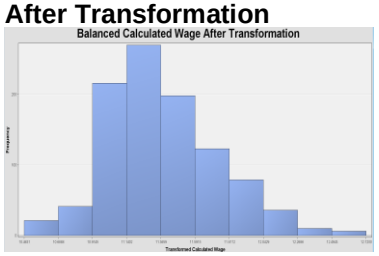

Figure 9 - Balanced Calculated Wage - After Transformation