Limited Data and Rare Events: Assessing the Performance of Statistical Methods at Predicting Food Inspection Outcomes in Allegheny County

Kaila Gilbert KJGILBER/ANDREW.CMU.EDU
Heinz College of Information Systems and Public Policy
Carnegie Mellon University
Pittsburgh, PA, United States

Matt Samach MSAMACH/ANDREW.CMU.EDU
Heinz College of Information Systems and Public Policy
Carnegie Mellon University
Pittsburgh, PA, United States

1. Introduction

This project analyzes five years of food inspection records provided by the Western Pennsylvania Regional Data Center (WPRDC). The initial goal of this project was to develop a tool for regulators (and potentially consumers) to understand key determinants of restaurant closures. Unfortunately, the positive event (restaurant closure) occurred only 1.1% of the time, presenting both a promising reality for regulators and a challenge for modelers. When limitations related to inconsistencies and rare events arose in the data, the project pivoted to examining how to extract insights using feature engineering and external data collection. We collected, cleaned, and merged business information form publicly available sources. Because we valued interpretability, we selected linear regression, logistic regression, and random forests as our key analytic methods. We considered two outcomes: the number of violations resulting from an inspection and the overall outcome of the inspection. We found that random forests performed better than basic linear and logistic regression for both prediction tasks. When classifying positive and negative events, the random forest outperformed logistic regression, although both provided rather dismal AUC curves on the imbalanced data. We received better results by "supersampling" rare events using the SMOTE package and training on those simulated data outputs. Across all models, features related to business age, geography, business type, and purpose of inspection frequently appeared as key determinants. We conclude with a discussion about limitations of the data and possible uses for the model.

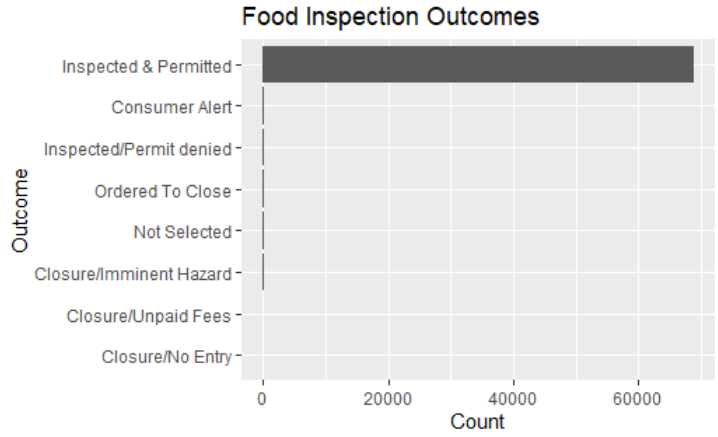
Our code is available at: https://github.com/kaigilly/FoodInspectionOutcomes

2. Background

This project was designed with hopes of creating such framework with regulators (and potentially consumers) in mind. Ideally, these models could be used by health and safety inspectors to optimize their targeting. It could also be used for the enlightenment of the typical hungry Pittsburgher. Developing these models, however, involved substantial data collection and pre-processing work.

2.1 Restaurant Closures: A Very Rare Event

Restaurant owners and Pittsburgh consumers both have a stake in better understanding any trends related to a health code violation or store closure. However, we found no preexisting research or methods for understanding key predictors of food inspection outcomes. A possible reason for the lack of comparative methods is the extent to which abnormal inspection outcomes are extremely rare. Nearly 98% of all inspections ended with the description of "Inspected & Permitted." To flag rare events and signal abnormality across inspection of outcomes, we considered collapsing all non-"permitted" inspections into a unique positive class. Even with this aggregation, the instances were very, very low.



3. Methods: Linear Regression, Logistic Regression, and Random Forest

Because the general public was one of our stakeholders, we wanted more interpretable models which could share more insight into key features associated with increased violations or an abnormal inspection. To blend the desire for both interpretable results and predictive power, we selected linear regression, logistic regression, and random forest models.

3.1 Feature Expansion via External Data Collection

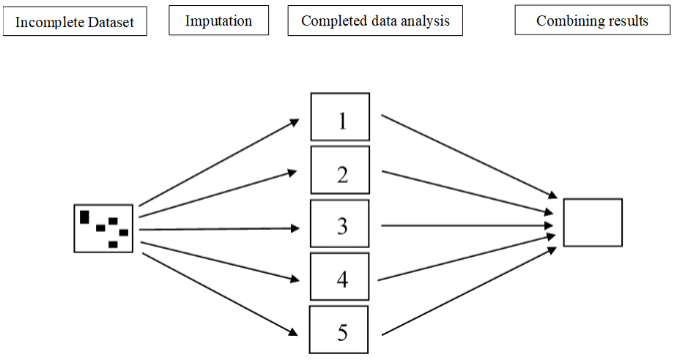
We collected, cleaned, and analyzed data from ReferenceUSA and Yelp to provide a richer feature set for both outcome variables. This information was merged to our data sets by the creation of unique address identifiers. The Yelp set is not included in the analysis, although that code is included in the Github link.

3.2 SMOTE Package

The SMOTE package is one way for dealing with rare events. The method has been used to detect credit card fraud. By using the ADASYN algorithm, the method generates synthetic positive instances in data set. Models are then trained on these data sets and then applied to the actual un-simulated data.

3.3 MICE package

To deal with missing values within a certain threshold, we utilized a method called multivariate imputation by chained equations. This method first uses mean imputation strategies for missing value, before allowing each cell



3.4 External Tools Used

We used the Selenium Python API to create a web scraper which could continually query ReferenceUSA. We merged this information with our inspections data sets by standardizing addresses using ArcGIS's Geocoding tool. Reference address files came from the Western Pennsylvania Regional Data Center.

3.5 R Packages Used

We used tidyverse for most of our standard processing, aggregating, and cleaning tasks. We used the DMwR library to implement the SMOTE algorithm, which "supersamples" rare events. We used the mice package to conduct multiple implementation for columns with up to 30% of values missing. Plots were generated using ggplot2. Dates and times were processed using lubridate. The randomForest and glmnet library were used for our modeling. The ROCR package was used to evaluate performance.

4. Experimental Setup

The core data sets utilized for this study comes from WPRDC and includes information between 2014 and 2019. Two central data sets form the basis of this study: a record of inspections at food facilities and a record of associated violations. The inspections data set contains details for 70,000 inspections conducted at a food facility, while the violations data set contains 290,000 records of resulting violations, including a description of the type and magnitude of the violation. Upon analysis, however, many features were inconsistent or incomplete. It became necessary to look elsewhere for supplemental data.

4.1 Cohort Selection

We analyzed the most recent year of inspections to see if some combination of historical information and business characteristics could predict outcomes. The number of observations was restricted to 12,000 rows.

4.2 Data Extraction

4.2.1 Merging with ReferenceUSA

In the absence of plentiful or informative features in the original data set, we decided to merge the restaurant information with other data sources.One such publicly available source was ReferenceUSA. This database contains valuable information about Pittsburgh businesses and their owners. New features include the characteristics of owners, employee size, budgetary expenses, square footage, and more granular food service information. Unfortunately, this site offered no official API and limited downloads to 250 entries at a time. Because of this, we needed to automate the data collection process to build our data set. Using a Python API, we built a custom web-scraper.We restricted ReferenceUSA results to restaurants (open or closed) associated with food service industries in Allegheny County.

4.2.2 Using ArcGIS to Standardize Address Data

With actual business data scraped from the ReferenceUSA website and loaded into a CSV, we began the process of matching instances with the Pittsburgh Food Inspection data set. First, both data sets were fed into ArcGIS Geocoding tool. The reference for the geocoding came from the Western Pennsylvania Regional Data Center address file. To deal with data coming from two different sources, we created a unique identifier for our standardized addresses. Once these instances were both "fuzzy matched" in a standardized library, we analyzed matches. Of the 12,104 unique restaurants in the inspection data set, we found unique matches for only 5,241 addresses. In many cases, several businesses were associated to a single address. We speculated that many of these addresses were tied to strip malls or large shopping centers. It is probable that the establishments that were dropped were in some way different than those were not dropped, indicating missingness not at random. However, for the purposes of our project, we preferred numerous features and a smaller data set to a large data set with few informative variables.

4.2.3 Dealing with Missingness: MICE and Data Trimming

Of the data that survived the merge, even more data was trimmed or dropped. With the combined data set, we decided to retain features that had less than 30% of values missing.This correlated to about six columns This usually related to demographic information or ownership information. A few categorical variables only occurred a handful of times across thousands of records. For modeling simplicity, we removed feature characteristics that occurred less than 5 times. Due to a limitation of random forest models preferring less than 53 factor levels, we had to aggregate and convert when necessary. For values retained, we utilized the Multiple Imputation Chain Equation (MICE) package to fill the missing values.

4.2.4 Binary Classification: Dealing with Rare Events with SMOTE

Since our outcome of interest occurred less than 0.6% of the time in our expanded data set, we were worried about the imbalance in our data set. We used the DmwR's SMOTE package to help bulk up the instances of our minority class so our models could possibly learn patterns that differentiate our positive an negative class. For the purposes of this analysis, we trained models on a SMOTE data set with a 50-50 split between positive and negative outcomes. With more time and analysis, we would ideally run and evaluate these models on several different splits.

4.3 Feature Choices

We believed historical trends in addition to supplemental factors, could predict the outcome of inspections in the most recent year. Each item in our inspection data represented an inspection. Beyond variables featured below, we included data about business location, business type and start date, the purpose of the inspection, owner characteristics, and number of violations resulting from previous inspections. Our final data set for testing included 5,195 observations and 65 features.

4.3.1 Creating Historical Vectors

Because the unit of analysis was an inspection, and one instrument could have many inspections, we created running vectors which could capture the unique historical factors of a given inspection. This one-hot encoding method offered up-to-date violation audits at the time of a current inspection.

4.3.2 Rare Event Outcome Variable

Of key interest to the project was whether or not a given inspection would end in closure of an establishment. We created a binary outcome variable that flags if an inspection outcome was abnormal or not. Inspections that were inspected and permitted were assigned a 0 to indicate a negative event. Inspections that ended in a closure, a customer alert, an advisory, etc were flagged as 1 to indicate a positive event.

4.3.3 Continuous Outcome Variable

In the event we could not gain meaningful results from a model with only 1.1% in the minority class, we also produced a continuous alternative. This could potentially communicate issues by proxy, although we acknowledge the nature and type of violation could communicate more actionable information by regulators.

4.4 Evaluation Criteria

To compare outputs for the continuous outcome variable, we selected mean squared error as a reasonable metric of performance. To compare performance for the binary models, we selected Area Under a Curve(AUC) as our key metric, with sensitivities plotted in the final analysis. We selected this method because we wanted to ascertain overall trends in model

capability, so that regulators could choose an appropriate cutoff based on the issues they cared about.

5. Results

5.1 Predicting the Number of Violations

For both number of violations and the log of number of violations, the random forest outperformed the linear regression. The forest had an mean squared error of 3.00, compared to an mean squared error of 9.16 for the linear regression. The numbers were 0.48 and 0.70 for the logged number of violations, respectively.The variable importance plot in the appendix reveals that purpose of the inspection, city, and business start date (communicating the age of the business) played important roles in determining number of violations. The linear regression also associated business start date, purpose, and indicator variables related to cities relatively high. Age was negatively associated with number of violations, which there was great variation across geographies.

Dependent Variable	Model	Mean Squared Error
Count of Violations	Linear Regression	10.562
	Random Forest	9.576

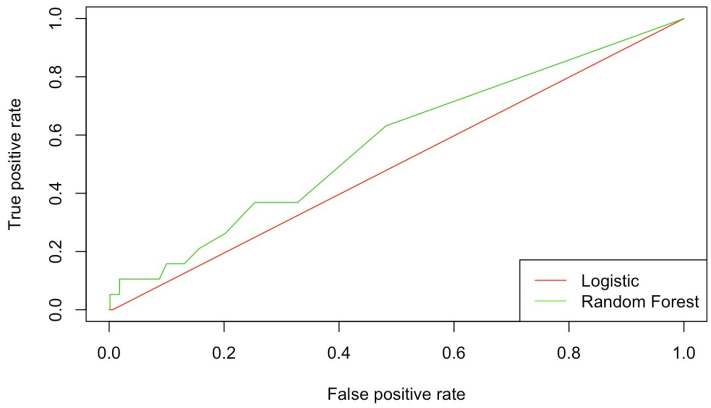
Dependent Variable	Model	Mean Squared Error
Number of Violations	Linear Regression	0.465
	Random Forest	0.427

5.2 Predicting Abnormal Inspections

5.2.1 Performance on the Imbalanced Data set

Accordingly, just 0.6% of all inspections ended in an abnormal inspection. While we were not optimistic (classifying everything as an non-event would be pretty good), we explored what a logistic regression could deem from the imbalanced data set. The logistic regression we ran produced a nearly perfectly diagonal ROC "curve", indicating that few insights could be gleaned.

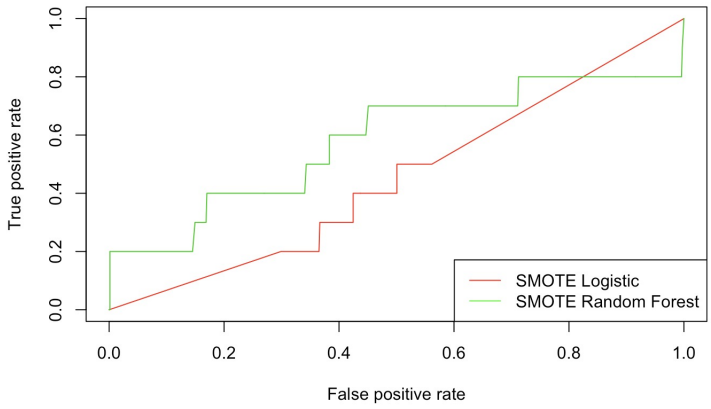
Normal / Abnormal	Model	AUC
Imbalanced	Logistic Regression	0.497
	Random Forest	0.574



5.2.2 Performance on the SMOTE-enriched Data set

The random forest performed better than the logistic regression, which performed worse than chance. The benefits are better seen when looking at sensitivity. In this case, our random forest exhibited much higher sensitivity, a better evaluation method for detecting the positive event

Normal / Abnormal	Model	AUC
Balanced Training	Logistic Regression	0.451
	Random Forest	0.580



6. Discussion

6.1 Missingness Probably Not At Random

When geocoding and merging the initial data set with ReferenceUSA, we only retained 43.2% of the original data. We decided to persist with a smaller, fuller data set. However, the restaurants we did not retain differed in some systematic way from the values were

able to retain. Likewise, we trimmed rows that occurred less than 5 times in the data, and removed columns with more than 30% of the data missing. Any of these decisions could affect the external validity of our study while also adding error and bias in our final predictions.

6.2 Causality vs. Correlation

While our models could assist in predicting past outcomes, we are careful not to suggest causal relationships between the models and their findings.

6.3 External Validity

Results from this analysis may not be generalizable to other locales, as it encompasses the inspection process unique to Allegheny County.

6.4 Inconsistencies in Initial Data

7. References

- Chapter 4 Multiple Imputation: Book_MI.utf8.md. (2019, November 14). Retrieved from https://bookdown.org/mwheymans/bookmi/multiple-imputation.html.
- Buuren, S. van. (2019). Package 'mice.' Retrieved December 2019, from https://cran.r-project.org/web/packages/mice/mice.pdf.
- Torgo, L. (2019). SMOTE Algorithm For Unbalanced Classification Problems. Retrieved December 2019, from https://www.rdocumentation.org/packages/DMwR/versions/0.4.1/topics/S
- Sugar, R. (2019). Allegheny County Restaurant/Food Facility Inspections and Locations. Retrieved 2019, from https://data.wprdc.org/dataset/allegheny-county-restaurant-food-facility-inspection-violationsutm_source=reply&utm_medium=email&utm_content=read_morecomm 4681054402.

8. Appendix