

# Finding Bad Apples Before They Rot

Using Machine Learning Techniques to Predict Food Inspection Results

*PBPL 28820*

*Lily Li*

## Introduction

With millions of residents and tourists each year and thousands of facilities that store, produce, or serve food, the danger of a food safety violation in a Chicago establishment is significant. To address food safety violations, the Chicago Department of Public Health conducts tens of thousands of establishment inspections each year. However, the department is unable to conduct as many inspections as is required by law due to a lack of staff and funding. In 2015, less than half of Chicago's high-risk food establishments were inspected twice, as required by the state.<sup>1</sup> Under these conditions, it is critical for the Department of Public Health to maximize the impact of each inspection. It would be ideal if the city could determine which establishments are more at risk of violating the law and only target those establishments to increase the potential utility of each inspection. To accomplish this, the city would need to predict the probability that a given establishment will fail its inspection. Because this is a prediction problem, we can look to machine learning for a solution.

In fact, the city has already tried to do this. Chicago built a predictive analytics tool to identify the highest-risk establishments and began implementing it in 2015. I plan to build my own model and compare my results to the city's to see if I can improve upon the city's current analytics tool.

## Design

### Data

My main dataset is the "Food Inspections" dataset from the Chicago Data Portal, which gives the inspections of restaurants and other food establishments in Chicago from January 1, 2010 to present day. For each observation in the dataset, the name, license number, type of facility (ex. bakery, gas station, liquor store, restaurant, etc.), risk category of facility, address, inspection date, inspection type, result, and type of violation is given. Each observation in the "Food Inspections" dataset is a standalone inspection. I used the dataset to construct additional variables of "past\_inspect" and "past\_failed\_inspect" to show the inspection history of each establishment and how well it has done on inspections in the past. This also lets me account for how long a restaurant has been open, because those who have been open longer have more inspections.

I suspect that the likelihood of failing a food violation is dependent upon other variables not contained in this dataset. For instance, the cleanliness of a restaurant may be affected by the state of its surrounding area. To add additional variables to the "Food Inspections" dataset, I located the community area of each establishment based on its coordinates. I then added community-area-level crime statistics, socioeconomic indicators, and 311 data (Sanitation Code Complaints, Garbage Carts, Rodent Baiting, Abandoned Vehicles, and more).

I also performed data cleaning to improve the quality of the data. I cleaned the categorical variables of Facility Type and Inspection Type to correct for misspelling and repeated categories. I also cleaned the Results variable to turn it into a binary classifier. I removed inspections that had results of "Not Ready", "Out of Business", "No Entry", or "Business Not Located". In addition, I turned the "Pass w/ Conditions" result (which occurs when establishments are found to have critical or serious violations, but these violations

---

<sup>1</sup><http://www.chicagotribune.com/business/ct-chicago-restaurant-inspections-1130-biz-20161129-story.html>

are corrected during the inspection) to simply “Pass” because I am only interested in catching worst-case scenario inspections that lead to failure.

In addition, I restricted the types of inspections in my dataset. The city conducts several types of inspections, some of which are routine and required for a new business to open, and others that are done in response to specific complaints, such as food poisoning. I am not interested in studying the latter because they must occur and are not risk-based. Thus, I will only look at routine (canvass) inspections and re-inspections, the most common type of inspection performed at a frequency relative to the risk of the establishment.

## Model Construction

The effectiveness of my model depends on how well I do compared to the status quo. In this situation, the status quo is already better than randomly selected inspections, because the city has implemented its own predictive analytics tool. Thus, my model must perform better than the city’s to be effective.

I can examine my performance by keeping the number of failed inspections constant, seeing how many inspections I need to do to catch that number of failures, and comparing my total number of inspections to the city’s.

If inspections were randomly selected, I could simply split the food inspections dataset into two time periods and use the earlier time period as my training set and the later time period as my test set. However, the city’s predictive analytics tool complicates things.

The frequency of canvass inspections is driven by the risk level of the facility. Prior to 2015, risk levels were solely determined by the food handling practices required for each establishment. Establishments were generally categorized as riskier the more contact they had with preparing and serving food (so restaurants were riskiest and prepackaged food facilities were the least risky). After 2015, the city began using its own analytics model to quantify the risk of each individual establishment. It is unclear how robust the city’s risk model was prior to 2015, so I will assume that inspections were randomly selected for simplicity’s sake. After 2015, the city began targeting high-risk establishments. This means that data from 2015 onwards is biased by the city’s algorithm. If we were to construct a test set consisting of data from 2015-2016, our results would be skewed towards failure, because we would only be testing on establishments the city deemed risky.

To address this issue, I will attempt to recreate the city’s model by observing which establishments the city chose to inspect in 2015, as this is when the city began implementing its tool. My recreation of the city’s model will serve as a proxy for how the city is measuring an establishment’s risk of failure, and if we assume the city is doing a good job of predicting failures, my “inspected\_in\_2015” variable should closely mirror the results of these inspections.

I will run two models, my own and my approximation of the city’s. For both models, the training set is data from 2010-2013, and the test set is data from 2014. I use 2014 data as my test data, because 2014 is the last year before the model was put in use, so these inspections are still “neutral” and randomly selected. My own model will use the results of past inspections to predict the results in 2014. The city’s model will see which establishments in the training set were inspected in 2015, and use that information to predict the likelihood of establishments in the test set being inspected in 2015. I will run a ridge regression and lasso for both models.

My final training dataset had 35,573 observations with 94 predictors, while my test dataset had 12,144 observations with 94 predictors. A full list of my variables is located in the appendix of this paper.

## Results

In 2014, there were 12,144 canvass inspections. 2,003 of those inspections resulted in failure, leading to a hit rate of 16.5%. I would like to improve that hit rate by holding the number of failed inspections constant and reducing the total number of inspections.

## Ridge Regression Models

I first fit a ridge regression model on the training set to predict an inspection's result. For my model,  $\lambda = 0.01$  results in the smallest cross-validation error. Out of 126 coefficients, the 10 largest by absolute value are presented below. As expected, none of the 126 coefficients are zero, since ridge regression does not perform variable selection.

Table 1: 10 Largest Coefficients in My Ridge Regression

Variable	Coefficients
(Intercept)	192.657
PUBLIC_INDECENCY_crimerate	-4.644
NON_CRIMINAL_crimerate	2.797
Facility_Type: church	-2.301
Facility_Type: movie theater	-2.203
Facility_Type: cafeteria	-2.114
Facility_Type: navy pier kiosk	-2.059
Inspection_Type: canvass re-inspection	-1.499
Facility_Type: other	1.438
Facility_Type: rooftop	-1.383

I rank ordered the probabilities of failure in descending order to select the riskiest establishments. To get 2,003 failures, my model would require 12,132 inspections. This is essentially no improvement from the baseline rate, as this is only 12 fewer inspections than what actually occurred in 2014.

I then fit a ridge regression model on the training set to predict if an establishment was inspected in 2015, to mimic the city's model. For this model,  $\lambda = 0.01$  results in the smallest cross-validation error. Out of 126 coefficients, the 10 largest by absolute value are presented below.

Table 2: 10 Largest Coefficients in City Ridge Regression

Variable	Coefficients
(Intercept)	-552.065
Facility_Type: kiosk	4.066
Facility_Type: church	3.790
Facility_Type: children's services facility	-3.565
Facility_Type: daycare	-3.451
Facility_Type: butcher	3.279
Facility_Type: college	3.063
Facility_Type: nursing home	-2.201
NON_CRIMINAL_crimerate	2.151
Risk: Risk 3 (Low)	-2.125

To target the 2,003 riskiest establishments as determined by the city, the city's model would require 2,186 inspections. Clearly, the city's model is doing much better at predicting the likelihood of an establishment being inspected in 2015 than my model is at predicting the likelihood of a failed inspection.

## Lasso Models

I then fit a lasso model on the training set to predict an inspection's result. For this model,  $\lambda = 0.01$  results in the smallest cross-validation error. Out of 126 coefficients, the 10 largest by absolute value are presented below.

Table 3: 10 Largest Coefficients in my lasso Regression

Variable	Coefficients
(Intercept)	-5.373
Inspection_Type: canvass re-inspection	-1.103
Facility_Type: other	1.065
PUBLIC_INDECENCY_crimerate	-1.033
Facility_Type: bar	0.568
Risk: Risk 3 (Low)	0.107
Facility_Type: grocery	0.107
past_failed_inspect	0.040
INTERFERENCE_WITH_PUBLIC_OFFICER_crimerate	0.040
BURGLARY_crimerate	0.015

Unlike ridge regression, the lasso does perform variable selection, so it will turn some of the coefficients to exactly zero. In this case, the model turns 110 of the 126 coefficient estimates to exactly zero.

To get 2,003 failures, my model would require 12,133 inspections. Once again, there is very minimal improvement from what actually occurred in 2014.

I then fit a lasso model on the training set to predict inspected\_in\_2015. For this model,  $\lambda = 0.01$  results in the smallest cross-validation error. Out of 126 coefficients, the 10 largest are presented below.

Table 4: 10 Largest Coefficients in city lasso Regression

Variable	Coefficients
(Intercept)	-335.855
Facility_Type: daycare	-2.804
Facility_Type: children's services facility	-2.196
Risk: Risk 3 (Low)	-2.004
Facility_Type: kiosk	1.753
Facility_Type: nursing home	-1.492
Facility_Type: school	1.385
Facility_Type: golden diner	0.664
Risk: Risk 2 (Medium)	-0.504
Facility_Type: church	0.259

In this case, the model turns 105 coefficient estimates to exactly zero.

To target the 2,003 riskiest establishments as determined by the city, the city's model would require 2,233 inspections. Once again, the city's model is doing much better at its prediction problem than my model is.

## Conclusion

Neither of my machine learning models (ridge regression or lasso) for predicting inspection results performs well enough to justify the city replacing its current model with mine. My models perform about as well as the pre-2015 status quo. While my ridge regression model reduced the number of inspections by one more than my lasso model, this difference is not significant enough for me to favor one model over the other.

It is difficult to compare the success of my model to the city's, because my initial assumptions appear to be incorrect. I assumed that the city's prediction tool is very good at predicting which establishments will fail.

This led me to believe that the “inspected\_in\_2015” and “Results” variables, both 0/1 binary variables, were measuring the same thing (i.e. if an establishment was inspected in 2015, it also failed its 2015 inspection). I used this assumption to justify approximating the city’s model by predicting “inspected\_in\_2015.” However, this assumption does not seem to be true. In the test set, “inspected\_in\_2015” and “Results” are only the same for 3,871 inspections, or 31.9% of the total inspections.

Looking at my results, it seems much easier to predict an establishment’s likelihood of being inspected by the city than the actual result of the inspection. This suggests that the city is not solely concerned with maximizing its hit rate. The city may have other considerations for choosing inspections besides the final result of the inspection, like spreading out inspections evenly among geographic areas or facility types. This is evidenced by the city’s hit rates in 2015 and 2016. Even with the help of the prediction tool, the city was only able to achieve a hit rate of 18.11% in 2015 and 18.08% in 2016. This also explains why my model’s results are vastly worse than my approximation of the city’s model. Being inspected in 2015 is a poor proxy for failure, because the vast majority of inspections in 2015 passed.

Another possible explanation for the city’s low hit rates post-2015 is that the city is trying to predict pure results and is just doing a very poor job. Thus, the establishments the city predicts will fail often pass instead.

My unsatisfactory results for my results prediction model could be due to two issues: either I do not have the proper data needed to accurately predict food inspection results, or food inspection results are inherently difficult to predict. Looking at the actual city’s model<sup>2</sup>, the city used a few data sources that I did not use, like Business Licenses and weather data. If I had more time, I would add this data to my model to see if I could improve my results. I would have also liked to incorporate price and ratings information from Yelp. In addition, community-area-level statistics may be too broad. In the future, I would like to aggregate data at the block group level instead. It is difficult to determine if food inspection results are inherently hard to predict without running more models on more data. Additional research and work needs to be done in this area to determine how to best reduce the risk of unsafe food conditions in Chicago establishments.

---

<sup>2</sup>The city’s model can be found here: <https://github.com/Chicago/food-inspections-evaluation>. While their code is publicly available online, I did not look at it until after I ran my models to avoid influencing my predictors and model selection.

## Appendix

Listed below are all the variables in my training and test set:

- [1] "License\_\_"
- [2] "Facility\_Type"
- [3] "Risk"
- [4] "Zip"
- [5] "Inspection\_Type"
- [6] "Results"
- [7] "comm\_area"
- [8] "past\_inspect"
- [9] "past\_failed\_inspect"
- [10] "inspect\_year"
- [11] "num\_abandoned\_buildings\_complaints"
- [12] "num\_abandoned\_vehicles\_complaints"
- [13] "avg\_abandoned\_vehicles\_complaint\_length"
- [14] "num\_alley\_lights\_out\_complaints"
- [15] "avg\_alley\_lights\_out\_complaint\_length"
- [16] "num\_garbage\_carts\_complaints"
- [17] "avg\_garbage\_carts\_complaint\_length"
- [18] "num\_graffiti\_complaints"
- [19] "avg\_graffiti\_complaint\_length"
- [20] "num\_lights\_all\_complaints"
- [21] "avg\_lights\_all\_complaint\_length"
- [22] "num\_lights\_one\_complaints"
- [23] "avg\_lights\_one\_complaint\_length"
- [24] "num\_pot\_holes\_complaints"
- [25] "avg\_pot\_holes\_complaint\_length"
- [26] "num\_rodent\_complaints"
- [27] "avg\_rodent\_complaint\_length"
- [28] "num\_sanitation\_complaints"
- [29] "avg\_sanitation\_complaint\_length"
- [30] "num\_tree\_debris\_complaints"
- [31] "avg\_tree\_debris\_complaint\_length"
- [32] "num\_tree\_trims\_complaints"
- [33] "avg\_tree\_trims\_complaint\_length"
- [34] "num\_abandoned\_buildings\_complaints\_miss"
- [35] "num\_abandoned\_vehicles\_complaints\_miss"
- [36] "avg\_abandoned\_vehicles\_complaint\_length\_miss"
- [37] "num\_alley\_lights\_out\_complaints\_miss"
- [38] "avg\_alley\_lights\_out\_complaint\_length\_miss"
- [39] "num\_garbage\_carts\_complaints\_miss"
- [40] "avg\_garbage\_carts\_complaint\_length\_miss"
- [41] "num\_graffiti\_complaints\_miss"
- [42] "avg\_graffiti\_complaint\_length\_miss"
- [43] "num\_lights\_all\_complaints\_miss"
- [44] "avg\_lights\_all\_complaint\_length\_miss"
- [45] "num\_lights\_one\_complaints\_miss"
- [46] "avg\_lights\_one\_complaint\_length\_miss"
- [47] "num\_pot\_holes\_complaints\_miss"
- [48] "avg\_pot\_holes\_complaint\_length\_miss"
- [49] "num\_rodent\_complaints\_miss"
- [50] "avg\_rodent\_complaint\_length\_miss"

[51] "num\_sanitation\_complaints\_miss"  
 [52] "avg\_sanitation\_complaint\_length\_miss"  
 [53] "num\_tree\_debris\_complaints\_miss"  
 [54] "avg\_tree\_debris\_complaint\_length\_miss"  
 [55] "num\_tree\_trims\_complaints\_miss"  
 [56] "avg\_tree\_trims\_complaint\_length\_miss"  
 [57] "ARSON\_crimerate"  
 [58] "ASSAULT\_crimerate"  
 [59] "BATTERY\_crimerate"  
 [60] "BURGLARY\_crimerate"  
 [61] "CONC\_CARRY\_crimerate"  
 [62] "SEXUAL\_ASSAULT\_crimerate"  
 [63] "CRIM\_DAMAGE\_crimerate"  
 [64] "CRIM\_TRESPASS\_crimerate"  
 [65] "DECEP\_PRACT\_crimerate"  
 [66] "GAMBLING\_crimerate"  
 [67] "HOMICIDE\_crimerate"  
 [68] "HUMAN\_TRAFF\_crimerate"  
 [69] "INTERF\_PUB\_OFF\_crimerate"  
 [70] "INTIMIDATION\_crimerate"  
 [71] "KIDNAPPING\_crimerate"  
 [72] "LIQUOR\_crimerate"  
 [73] "MOTOR\_THEFT\_crimerate"  
 [74] "NARCOTICS\_crimerate"  
 [75] "NON\_CRIM\_crimerate"  
 [76] "OBSCENITY\_crimerate"  
 [77] "OFFENSE\_CHILDREN\_crimerate"  
 [78] "OTHER\_NARCOTIC\_crimerate"  
 [79] "OTHER\_OFFENSE\_crimerate"  
 [80] "PROSTITUTION\_crimerate"  
 [81] "PUBLIC\_INDEC\_crimerate"  
 [82] "PUB\_PEACE\_crimerate"  
 [83] "ROBBERY\_crimerate"  
 [84] "SEX\_OFFENSE\_crimerate"  
 [85] "STALKING\_crimerate"  
 [86] "THEFT\_crimerate"  
 [87] "WEAPONS\_VIOL\_crimerate"  
 [88] "PERCENT\_OF\_HOUSING\_CROWDED"  
 [89] "PERCENT\_HOUSEHOLDS\_BELOW\_POVERTY"  
 [90] "PERCENT\_AGED\_16\_\_UNEMPLOYED"  
 [91] "PERCENT\_AGED\_25\_\_WITHOUT\_HIGH\_SCHOOL\_DIPLOMA"  
 [92] "PERCENT\_AGED\_UNDER\_18\_OR\_OVER\_64"  
 [93] "PER\_CAPITA\_INCOME"  
 [94] "HARDSHIP\_INDEX"  
 [95] "month"  
 [96] "day"  
 [97] "inspected\_in\_2015"