**Predicting Food Inspection Outcomes of Chicago Restaurants**

Andrew Brennan and Joe Reilly

Introduction

In one year in Chicago, more than 15,000 food vending establishments are inspected by 30 - 35 Department of Public Health (DPH) inspectors. Many of these inspections uncover no serious issues, but roughly 15% of inspections detect establishments failing to adhere to DPH guidelines. The longer a restaurant or other location operates with a serious violation, the greater the risk of food poisoning or other illness for Chicago’s citizens. This project focuses on predicting which establishments are likely to have serious violations and aims to prioritize visiting these locations soonest.

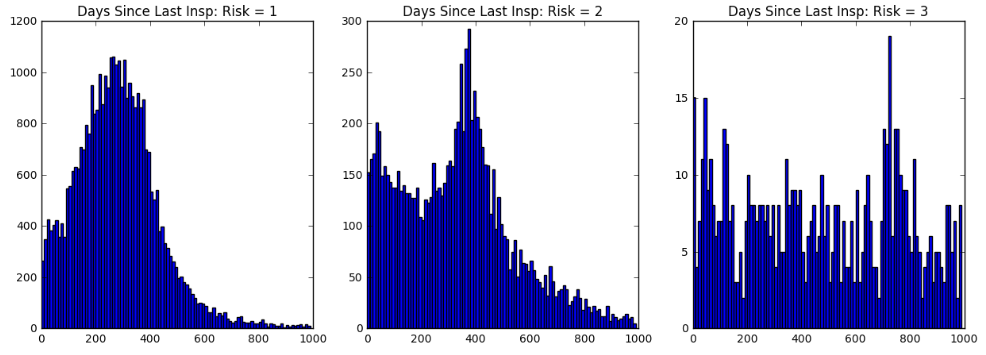
Far from being a hypothetical exercise, the City of Chicago partnered with Allstate Insurance in 2014 to tackle this problem. Chicago is a pioneer of providing open-source data and was recently awarded $1M from the Bloomberg Philanthropies Mayor’s Challenge in which they proposed to aggregate and analyze data to make better and faster decisions. This led to the creation of the SmartData Platform, a place to access well-structured and up-to-date data from many sources including business licensing, crime statistics, 311 complaints about sanitation, and prior food inspection outcomes. The open data portal that exists to improve transparency for city government, allow developers to make apps to aid residents, and to increase commercial activity for small and large businesses. The portal aided the city’s dissemination of a large volume of data and proved to be an effective tool for allowing collaborative research.

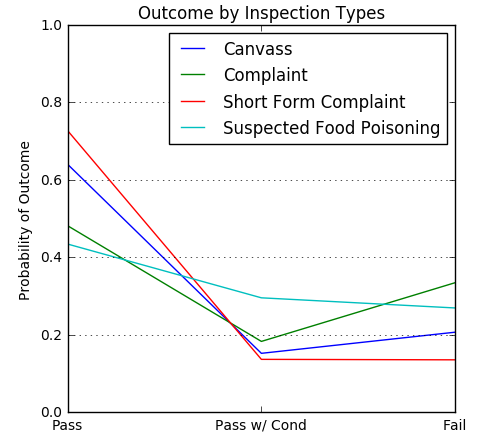
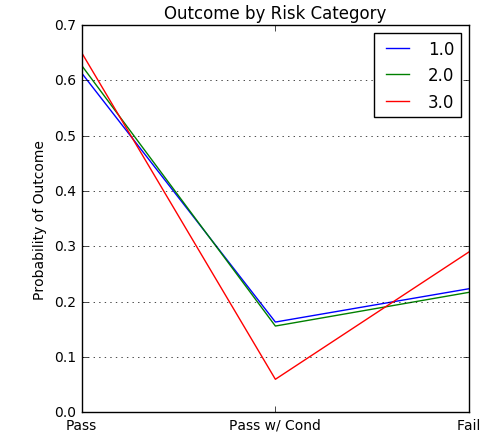
The City of Chicago has released its data and code for this project with the hopes that other cities can implement similar procedures or help them improve their own. However, most other cities would struggle to aggregate the data from many sources since they do not have a program similar to the SmartData Platform. Our goal in this project was to study their process, develop our own model, and see what performance gains we could achieve.

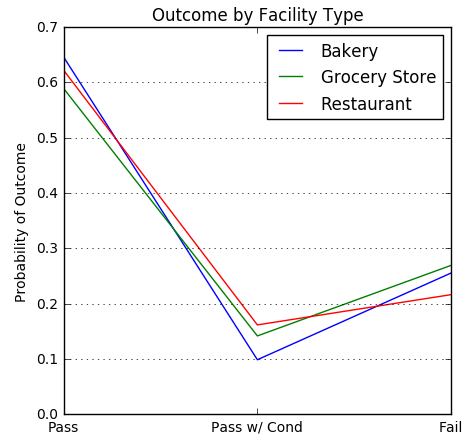
Data

Our primary source of data for the project was a repository of roughly 340,000 restaurant inspections in the City of Chicago from 2010 to the present. This repository is freely hosted on the city’s Data Portal and includes basic information such as name, location, time and date of inspection, type of facility, and license number. Location was represented as both zip code and as latitude/longitude coordinates. Facilities are classified into different risk tiers with Risk 1 (High) establishments presenting the highest risk to the population due to handling of raw ingredients (such as a restaurant) to Risk 3 (Low) facilities that typically reheat premade food (such as some daycares). We did not eliminate facilities by risk, but we chose to narrow our data to restaurants, grocery stores, and bakeries due to the assigned topic of the project on retail food establishments.

We calculated days since last inspection and noticed that Risk 1 establishments do get inspected more frequently than lower risk restaurants. We surmised that the length of time between inspections could be an important predictor of inspection outcome



Multiple types of inspection are possible, but many inspections such as those for establishing licensure must occur at certain times and are not able to be prioritized by our model. We narrowed our focus to inspections that occurred due to regular canvassing, as a result of a complaint, or due to suspected food poisoning. Inspection outcomes fall into three categories: pass, pass with conditions, and fail. The Chicago/Allstate team collapsed this into pass/fail, but we kept the outcome multiclass to see if this added nuance would aid the predictive power of our model. We also removed all inspections where no outcome was listed due to the establishment going out of business or being under renovations. After this filtering, we had 61,727 records remaining.



Historical weather data were available via the NOAA National Centers for Environmental Information. These data were recorded at Chicago O’Hare international airport and we limited ourselves to using the daily high, low, and average temperature, the amount of precipitation, the amount of snowfall, and the average wind speed. From the previous analysis done by the Chicago/Allstate team, we believed the temperature data may predict inspection outcomes to some degree. In lieu of using the daily high, we calculated the three-day rolling average high temperature.

We downloaded all the 311 sanitation complaints in the city, of which there were 112,000, or about 45 per day. These complaints may reflect areas of the city where sanitation standards are reduced and therefore may predict where restaurants will fail their inspections. To utilize these data in our task, we tallied the number of complaints within the last week and within a 0.5 km radius for each inspection.

Finally, we scraped rating and price data from Yelp for our retail food establishments due to a suggestion from the teaching team during our poster session. Due to limits on the number of calls allowed to the Yelp API per day and the short period of time between the poster session and the website due date, we were able to search for roughly 16,000 establishments. After removing missing data and narrowing our focus on certain types of establishment and inspections, we obtained a subset of 8,337 inspection records with price and review data. Analysis of this subset will allow us to determine if it would be worthwhile to continue scraping this data for more establishments.

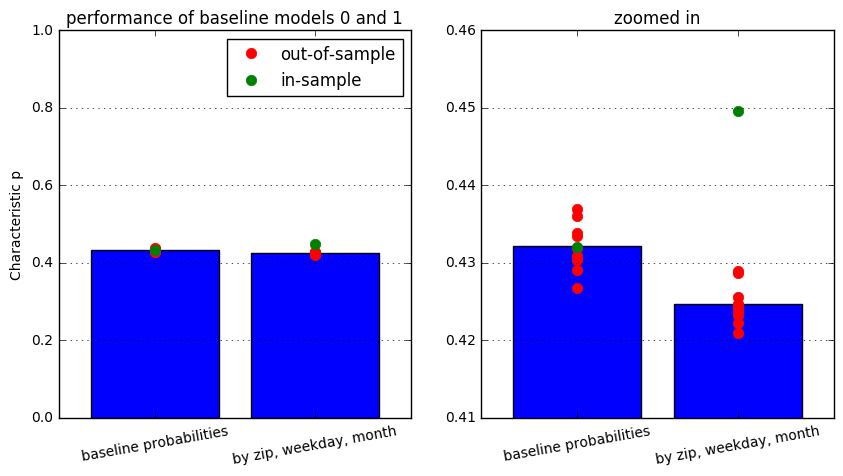
Model Building

We first defined our performance metric against which all our models would be evaluated. We aimed to maximize the likelihood of the observed data given the model. For computational reasons, we are technically minimizing the negative log likelihood as a loss function. The log likelihood function adds a positive number with a magnitude of roughly 1 for each additional observation. This is much less likely to hit machine precision issues than multiplying many numbers smaller than 1 as in the likelihood function, which gets quite small quite fast.

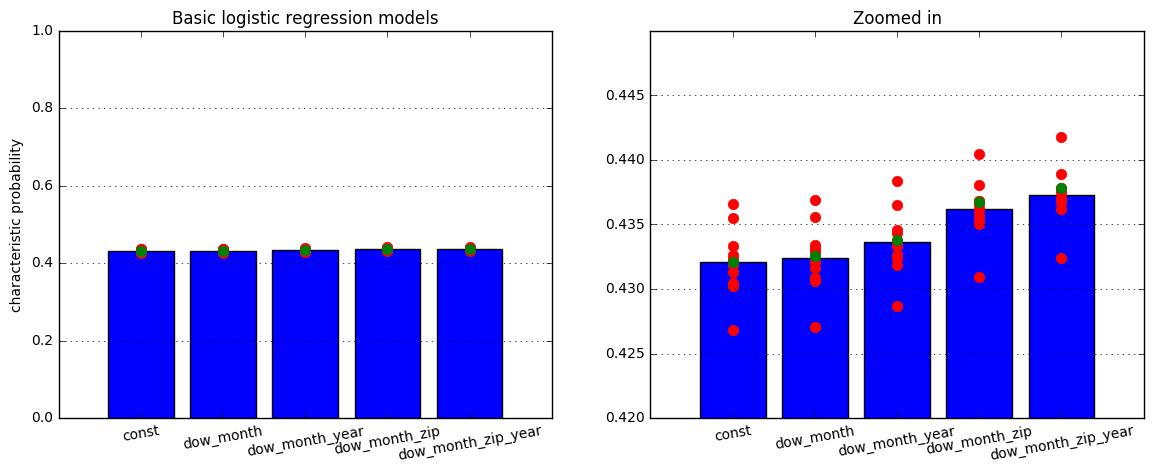
Represent the actual equation here?

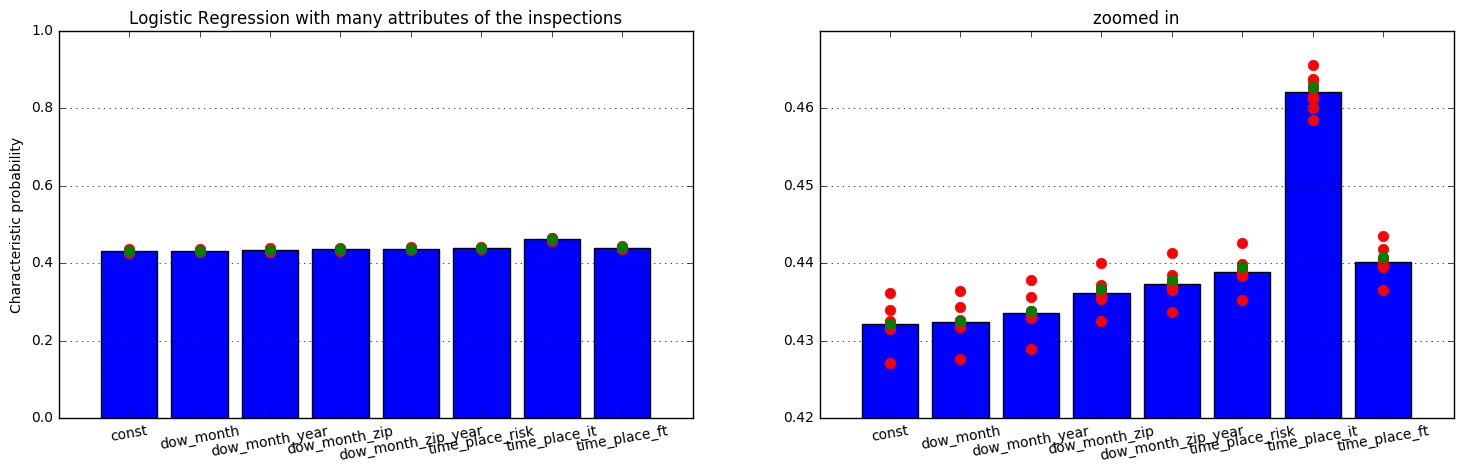
The model assigns a probability to each of the three classes of inspection outcome, and then the probability of the observed outcome (according to the model) is then just the class probability of the observed outcome. The joint probability of all the observed outcomes, assuming they are independent, is the product of the individual outcome probabilities. We are scoring the models based on this joint probability of obtaining the observed data from that model. This metric has two advantages: it is the same metric that logistic regression models maximize and it naturally extends to the multiclass classification we are doing here. All visuals in our model selection process plot this likelihood using the characteristic probability, which represents an average probability of observing the data across all the observations. Specifically, if model M has characteristic probability P, then the model that always predicts the observed outcome with probability P will have the same likelihood.

Our first baseline model simply uses the inspection outcome frequencies from our sample to make predictions. Regardless of the inspection, it predicts the same class probabilities. This gave a characteristic probability of 0.43 which we sought to improve upon. We added the day of the week, month of the year, and zip code to our model, but cross-validation revealed that this model overfits our data.

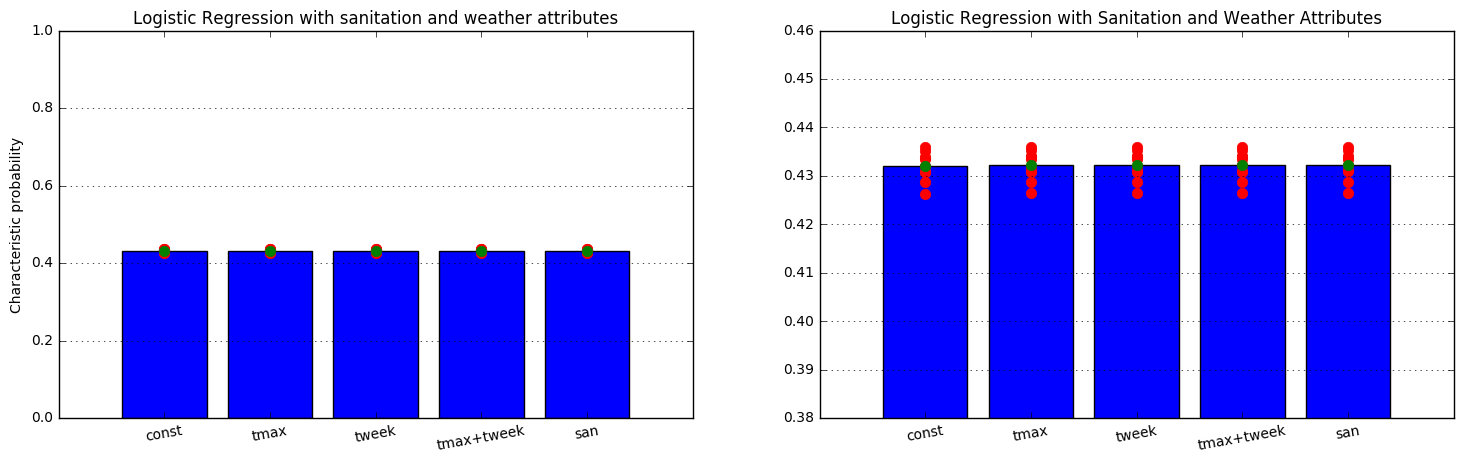


We next fit a series of logistic regression models where we added the month, year, and zip code. Adding simply the day of the week did not help, but year did, suggesting some long-term trends in inspection outcome. The inclusion of zip code greatly improved our model’s predictive power.

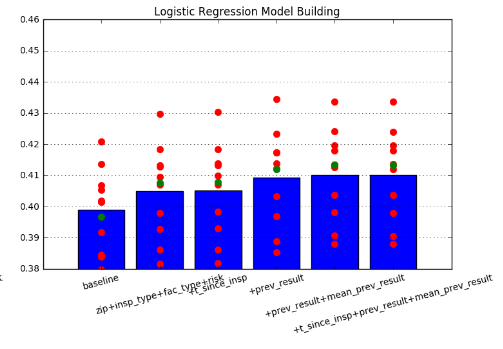
 Our next series of models took this information and incorporated risk, facility type, and inspection type. The type of inspection being carried out holds the most predictive power of these variables with high-risk inspections more often resulting in critical failures. Including risk rating and facility type also serve to improve our model. The model that included these three additional features performed much better, both in-sample and out-of-sample (cross-validation likelihood).



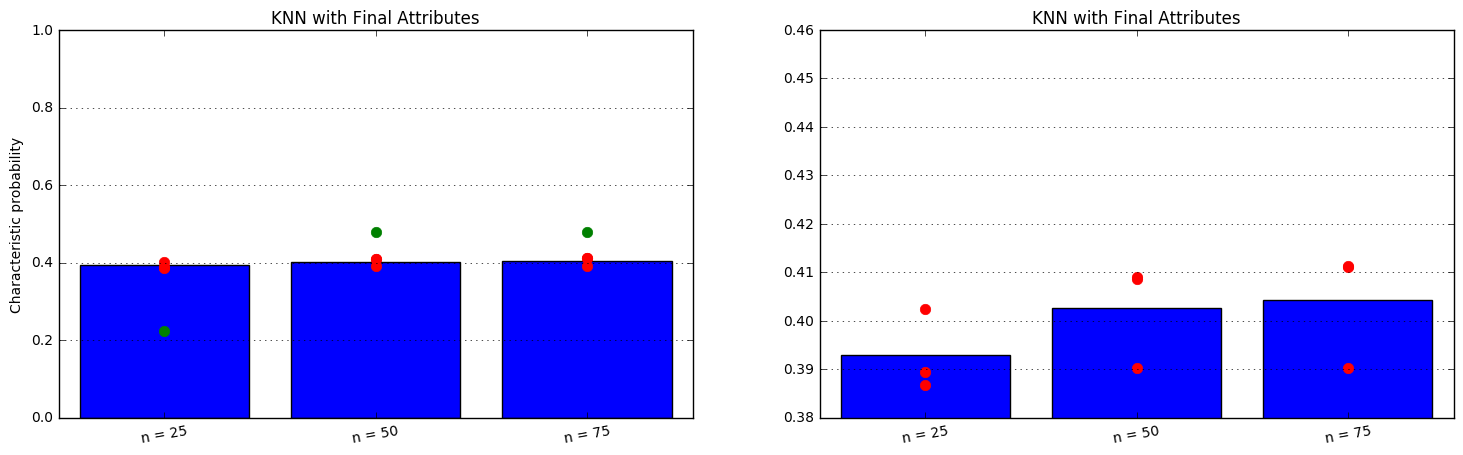
We next tried to improve this model by adding recent nearby sanitation complaints and weather data. For the sanitation complaints, we took complaints within 0.5km (much smaller than the ~3km radius we used before) and within the last week. For weather, we looked at both the day’s high and the mean high from the last week. However, none of these three features helped the models perform better than the models that used only the inspection data.



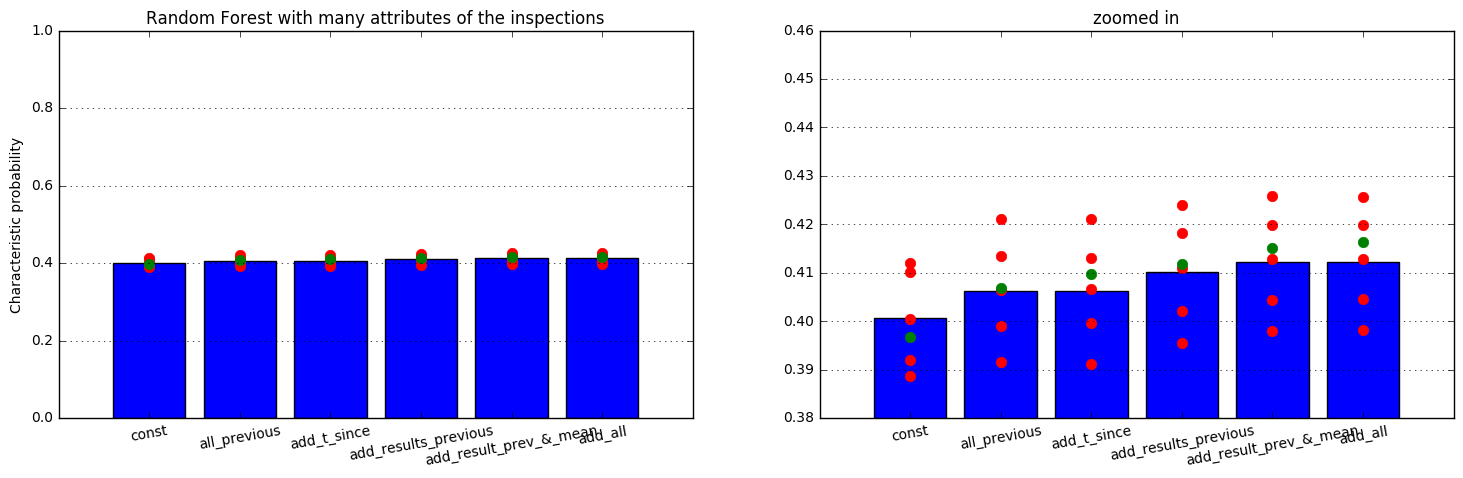
To delve into inspection data further, we added time since last inspection, the result of the previous inspection, and the mean of all previous results. This led to our final logistic regression model that included location by zip code, inspection type, facility type, and risk rating alongside our newly added predictors.



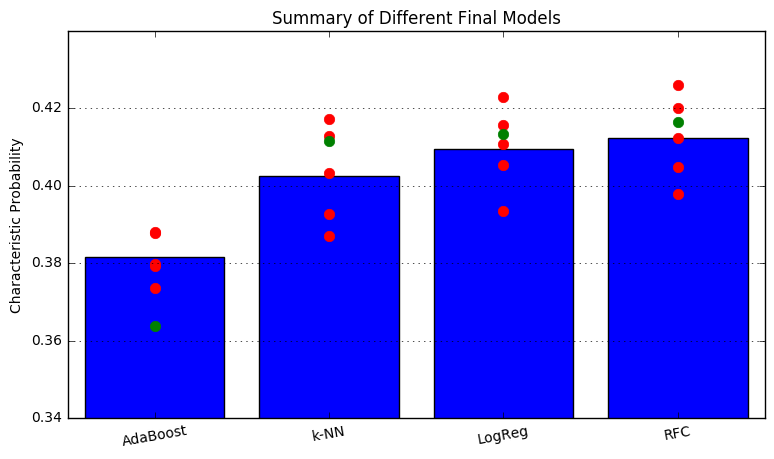
We next attempted to fir several models using different classifiers. We first attempted to use linear and quadratic discriminant analysis, but these did not perform nearly as well as logistic regression. This is likely due to their difficulty dealing with large amounts of categorical data. We next attempted a *k*-nearest neighbors approach to our classification problem. As the number of nearest neighbors considered increased past 75, k-NN performed roughly as well as our logistic regression model. Two issues arose, however: overfitting of our data was detected and the model was very computationally expensive to run.



We next turned to tree-based classifiers. The Random Forest classifier slightly outperformed our logistic regression model on characteristic probability after careful tuning with cross-validation. Further discussion in the “Evaluation” section will compare the performance of these two models. The AdaBoost classifier was also utilized but did not perform as well with our predictors as the Random Forest.

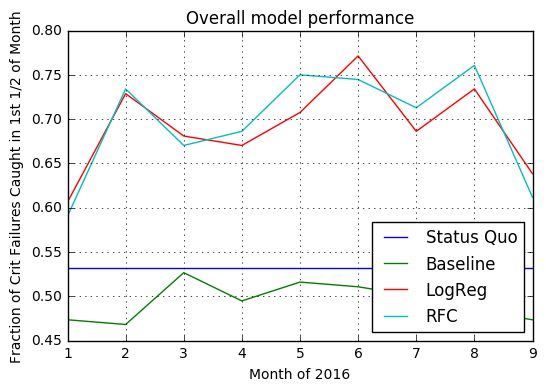


A visualization of the performance of our final models is provided below:



Evaluation

Random Forest and LogReg about the same.



Results