Dinesafe

Within Toronto, all establishment serving food requires up to 3 health inspection by Dinesafe, a program under Toronto Public Health. During these inspections, Dinesafe records data relating to the type of establishment, details and severity of infractions, fines if necessary, and whether the establishment passed inspection. An inspection status can have on of three values: pass, conditional pass, or closed. Full closure of establishments are exceedingly rare, so for the sake of this report I consider a “conditional pass” a fail.

Also note that a pass status does not necessarily imply an establishment is without health violations. In a majority of cases, establishments can have several minor or even crucial health violations before a conditional pass is given. Because of this, later sections focus on whether an establishment has at least one health violation, rather than if an establishment received a pass.

Data Set Used

Dinesafe (https://ckan0.cf.opendata.inter.prod-toronto.ca/dataset/ea1d6e57-87af-4e23-b722-6c1f5aa18a8d/resource/815aedb5-f9d7-4dcd-a33a-4aa7ac5aac50/download/Dinesafe.csv) (updates daily)

Description of Features

Variable	Description
_id	Unique row identifier for Open Data database
Rec #	Unique ID for an entire record in this dataset
Establishment ID	Unique identifier for an establishment
Inspection ID	Unique ID for an inspection
Establishment Name	Business name of the establishment
Establishment Type	Establishment type ie restaurant, mobile cart
Establishment Address	Municipal address of the establishment
Establishment Status	Pass, Conditional Pass, Closed
Min. Inspections Per Year	Minimum inspections of an establishment per year depending on establishment type. Can be 1, 2, 3, or “O” (Other)
Infraction Details	Description of the Infraction
Inspection Date	Calendar date the inspection was conducted
Severity	Level of the infraction, i.e. S - Significant, M - Minor, C - Crucial, NA - Not Applicable
Action	Enforcement activity based on the infractions noted during a food safety inspection

Outcome	The registered court decision resulting from the issuance of a ticket or summons for outstanding infractions to the Health Protection and Promotion Act
Amount Fined	Fine determined in a court outcome
Latitude	Latitude of establishment
Longitude	Longitude of establishment

Pass Rates Within Establishment Types

Out of 56406 inspections, only 230 has lead to a “Conditional Pass” status. More surprisingly, however, are how these failed inspections are concentrated within a handful of establishment types. Below are the types that contain at least one fail, the number of establishments of that type, the number of failed inspections, and the percentage chance that an establishment of that type would pass an inspection.

Establishment Type	Total Number of Establishment Type	Number of Failed Inspections	Probability of Passing
Community Kitchen (Meal Program)	282	27	0.9042553
Food Depot	263	11	0.9581749
Bake Shop	234	6	0.9743590
Food Court Vendor	1866	39	0.9790997
Retirement Homes(Licensed)	391	8	0.9795396
Nursing Home / Home for the Aged	284	5	0.9823944
Boarding / Lodging Home - Kitchen	318	4	0.9874214
Child Care - Food Preparation	973	11	0.9886948
Child Care - Catered	1296	5	0.9961420
Food Take Out	9282	35	0.9962293
Restaurant	29011	78	0.9966220
Student Nutrition Site	519	1	0.9980732

Interestingly enough, out of the 57 establishment types, only 12 contain at least one failed inspection. This concentration shows some connection between establishment type and the likelihood that an individual establishment will fail a health inspection.

Severity Score

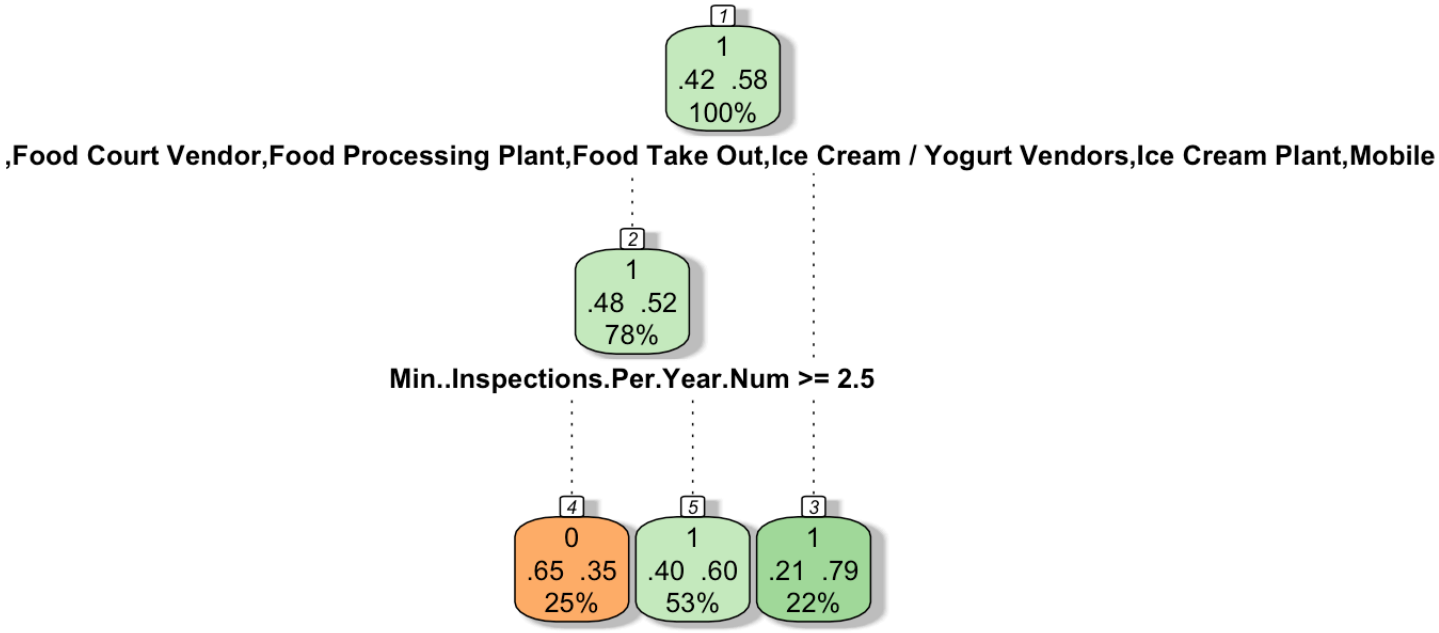
As mentioned before, a “Pass” status has no relation to the number of health violation an individual establishment has; an establishment can have multiple crucial violations and still pass. Because of this, a different metric is needed to determine how ‘clean’ or ‘dirty’ an establishment is. Infractions have 4 severity levels: minor, significant, crucial, and NA. NA deals with issues such as improper training or paperwork, while the other levels are related to the safety of the preparation, storage, and handling of food, along with the cleanliness of the area and its resistance to pests. By assigning values 0 for no infraction and 1-4 for NA to crucial respectively, we can calculate the “Severity Score” of each establishment by summing its infractions. Below are the establishments with the top ten highest severity scores.

Establishment Name	Establishment Address	Establishment Type	Min. Inspections Per Year	Severity Score
MB THE PLACE TO BE	3434 BATHURST ST	Restaurant	3	134
SEOUL HOUSE	3220 DUFFERIN ST, Unit-1a	Restaurant	3	88
MONKEY SUSHI	901 YONGE ST	Restaurant	3	83
THE GRILL COTTAGE	1468 QUEEN ST W	Restaurant	3	83
LA ROSE BAKERY & DELICATESSEN	140 LA ROSE AVE	Bakery	3	79
INDRAPRASTHA INDIAN CUISINE LTD	3300 LAWRENCE AVE E	Restaurant	3	77
QUEEN STREET WAREHOUSE	232 QUEEN ST W	Restaurant	3	77
SHUN HING NOODLES & FOOD PRODUCTS LTD	2200 MARKHAM RD, Unit-20-25	Food Processing Plant	3	74
SUNRISE CARIBBEAN RESTAURANT	1285 YORK MILLS RD	Restaurant	3	74
BTRUST SUPERMARKET	1105 WILSON AVE	Supermarket	3	72

Note that severity scores are calculated over all inspection in the past three years. Most establishments require up to three inspections per year, with the average being 2.31. Looking over all establishments, the average severity score is 4.67, with a SD of 8.17. When we only look at establishments with at least one violation, that average goes to 8.96 with a SD of 9.46. So while higher severity scores are biased towards establishments opened for at least three years, a score of about 20 or above indicate consistent moderate to major health violations over the past nine inspections.

Models

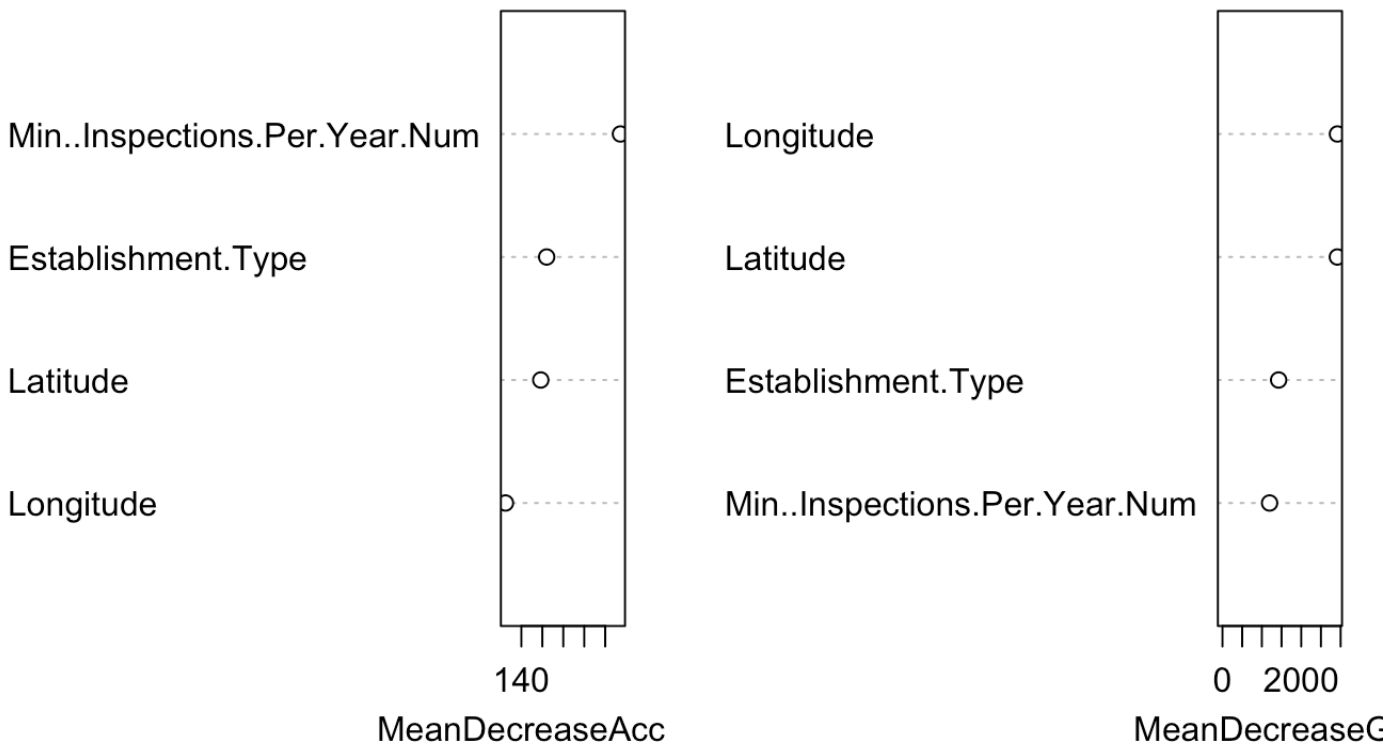
Using severity scores, I created a decision tree to see which features of the data relate most to whether an establishment has a severity score of at least one. Note that the data had to be adjusted, which will be explained later. From this we get,



Where the first variable are all establishment types with zero instances of health violations at any individual location, and 0 indicate at least one health violation.

Note that Establishments with three or more inspections per year have a higher chance of having at least one health violation. This could be since most establishments require three inspections, or that with more inspections brings a higher chance of finding a violation.

Using this insight, I created a random forest model that uses “Latitude”, “Longitude”, “Min..Inspections.Per.Year.Num”, and “Establishment.Type” to predict whether an establishment has at least one health violation. Note that other columns were not used either because they were unnecessary or too specific. For example, “Inspection.ID” is an arbitrary number used for tracking purposes, which would just add noise to the model. On the other hand, a feature such as “Action” is closely tied to a result; the “Action” value is filled only if there has been an infraction and the model would just check if “Action” was filled, rather than using other features. Below shows the features most associated with health violations.



Both the decision tree and random forest models showed some success in predicting if an establishment had a health violation, with DT having a success rate of 0.654, and RF having 0.671. These models shows how establishment type and the minimum number of inspections an individual establishment requires is predictive of whether an establishment will have at least one health violation.

Appendix

```
knitr::opts_chunk$set(echo = TRUE)

##### import library and data set ###############################################

library(tidyverse)
library(knitr)
library(ggplot2)
library(rattle)
library(rpart)
library(pROC)
library(randomForest)
library(naniar)
library(simputation)

d <- read.csv("https://ckan0.cf.opendata.inter.prod-toronto.ca/dataset/ea1d6e57-87af-
4e23-b722-6c1f5aa18a8d/resource/815aedb5-f9d7-4dcd-a33a-4aa7ac5aac50/download/Dinesaf
e.csv")

# miss_var_summary(d), gives 602 missing for Inspection.ID

# clean up Min..Inspections.Per.Year, sometimes int, sometimes 'O'. Change 'O' to 0
# to uniquely identify 'O' status while keeping the entire col type int
d$Min..Inspections.Per.Year[d$Min..Inspections.Per.Year == 'O'] <- 0

# creating d1. We add values to severity levels, pass/value indicator values,
# and Severity indicator values for later evaluations.
d1 = d %>% mutate(Severity.Value = case_when(Severity == "" ~ 0,
                                             Severity == "NA - Not Applicable" ~ 1,
                                             Severity == "M - Minor" ~ 2,
                                             Severity == "S - Significant" ~ 3,
                                             Severity == "C - Crucial" ~ 4),
                  Pass.Value = ifelse(Establishment.Status == "Pass", 1, 0),
                  Fail.Value = ifelse(Establishment.Status == "Conditional Pass", 1, 0
),
                  Severity.Blank = ifelse(Severity == "", 1, 0),
                  Severity.NA = ifelse(Severity == "NA - Not Applicable", 1, 0),
                  Severity.M = ifelse(Severity == "M - Minor", 1, 0),
                  Severity.S = ifelse(Severity == "S - Significant", 1, 0),
                  Severity.C = ifelse(Severity == "C - Crucial", 1, 0),

                  # Min..Inspections.Per.Year clean up
                  Min..Inspections.Per.Year.Num = as.numeric(Min..Inspections.Per.Year
))

##### pass rate per Establishment.Type ##########################################

# group by establishment type, get nrow, calculate total num of fails, chance to pass
```

```
failed_establishment_type = d1 %>%
  group_by(Establishment.Type) %>%
  summarise(N=n(),
            Num.Fail = sum(Fail.Value),
            Chance.To.Pass = sum(Pass.Value)/n()) %>%
  filter(Num.Fail != 0)

failed_establishment_type = arrange(failed_establishment_type, Chance.To.Pass)

kable(failed_establishment_type, col.names = c("Establishment Type",
                                               "Total Number of Establishment Type",
                                               "Number of Failed Inspections",
                                               "Probability of Passing"))

# total number of establishment types
total = d1 %>% group_by(Establishment.Type) %>% summarise(N=n())

##### severity scores per establishment type ###################################

# calculate sum of all severity values per individual establishment
sev_table = d1 %>%
  group_by(Establishment.Name,
           Establishment.Address,
           Establishment.Type,
           Min..Inspections.Per.Year.Num) %>%
  summarise(Severity.Score = sum(Severity.Value))

sev_table = arrange(sev_table, desc(Severity.Score))

kable(head(sev_table, 10), col.names = c("Establishment Name",
                                         "Establishment Address",
                                         "Establishment Type",
                                         "Min. Inspections Per Year",
                                         "Severity Score"))

sev_avg_table = sev_table %>% filter(Severity.Score != 0)

# mean within only establishments where sev != 0
sev_mean = round(mean(sev_avg_table$Severity.Score), 2)

# standard deviation
sev_sd= round(sqrt(var(sev_avg_table$Severity.Score)), 2)

# mean sev of all establishments
sev_with_pass_mean = round(mean(sev_table$Severity.Score), 2)

# standard deviation
sev_with_pass_sd= round(sqrt(var(sev_table$Severity.Score)), 2)

##### models ###################################################################
```

```
linear = d1 %>%
  group_by(Establishment.ID,
           Inspection.ID,
           Inspection.Date,
           Latitude,
           Longitude,
           Min..Inspections.Per.Year.Num,
           Establishment.Type) %>%
  summarise(Severity.Score = sum(Severity.Value),
            Severity.Score.Pass = ifelse(Severity.Score == 0, 1, 0))

# through trial and error using varImpPlot, determined Inspection.ID, Establishment.I
D,
# Inspection.Date don't really help the model; when you think about it, almost random
values
# that shouldn't affect the outcome of an inspection.
linear2 <- subset(linear, select = -c(Severity.Score,Inspection.ID, Establishment.ID,
Inspection.Date))

# clean linear2 a bit
linear2 <- na.omit(linear2)

# decision tree
decision_tree <- rpart(Severity.Score.Pass~., data=linear2, method="class")

# show tree
fancyRpartPlot(decision_tree, main="", sub="", palettes = c("Oranges", "Greens"))

# create random forest model
m_rf <- randomForest(as.factor(Severity.Score.Pass)~., data=linear2, ntree=500, impor
tance=TRUE)

# MeanDecreaseAccuracy and MeanDecreaseGini
varImpPlot(m_rf)

# need to un-group linear2 for predict to work
linear2=linear2 %>% ungroup() %>% mutate(rpart.pred = predict(decision_tree, type="cl
ass"),
                                         rf.pred = predict(m_rf, data=linear2))

# get sensitivity, specificity matrix
dt = table(linear2$Severity.Score.Pass, linear2$rpart.pred)
mrf = table(linear2$Severity.Score.Pass, linear2$rf.pred)

# get prediction percentage
dt_value = round(sum(diag(dt)/sum(dt)), 3)
mrf_value = round(sum(diag(mrf)/sum(mrf)), 3)
```