

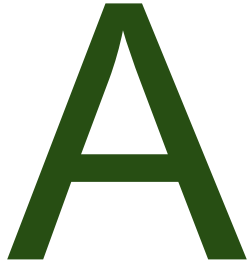
# Predicting Restaurant Health Inspection Grades

By: Ipek Sayar

# Project Goal

- Food safety is an essential public health concern that affects everyone
  - Consumers looking to make healthier dining choices by identifying safer restaurants.
  - Restaurant owners who strive to meet health standards
- Results take 1-2 weeks to come out + not frequently performed
- Classify restaurant GRADE
  - Grade that the restaurant got on their health inspection

# Class

A large, bold, green letter 'A' centered within a green square border.

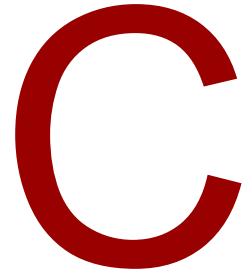
Met the highest health +  
safety standards

68%

A large, bold, yellow letter 'B' centered within a yellow square border.

Some violations, no  
immediate health risks

15%

A large, bold, red letter 'C' centered within a red square border.

Multiple serious  
violations, near closure

17%

# Original Dataset

NYC Food Establishment  
Health Inspections 2012-2023

Instances: 103,426

Attributes: 26

CAMIS	SCORE
DBA	GRADE DATE
BORO	RECORD DATE
BUILDING	INSPECTION TYPE
STREET	Latitude
ZIPCODE	Longitude
PHONE	Community Board
CUISINE DESCRIPTION	Council District
INSPECTION DATE	Census Tract
ACTION	BIN
VIOLATION CODE	BBL
VIOLATION DESCRIPTION	NTA
CRITICAL FLAG	GRADE

# Preprocessing

## Handling Missing Values:

- Class
  - 40% missing.
  - Used "Score" column (no missing values) to calculate grades per NYC Health guidelines:
    - 0-13 → A
    - 14-27 → B
    - 28+ → C
  - Filled missing grades using Excel IF-THEN statements.
  - Removed "Score" column.
- Attributes
  - 20% of "Grade Date" values were missing; replaced with placeholder (1/1/2024).

# Preprocessing

## Removing Irrelevant Columns:

- Removed CAMI (restaurant ID) and Phone Number as they are unique to each establishment.

## Weka Compatibility:

- Removed punctuation using Excel's Find and Replace.
- Used StringToNominal filter in Weka.
  - Most ML models require nominal or numeric data to function properly.

## Sampling for Efficiency:

- Used Python for stratified sampling to reduce dataset to 1500 instances.
  - Maintaining class proportions for accurate representation.

# Dataset after Preprocessing

Instances: 1,500

Attributes: 23

Class Proportions:

A: 68 %

B: 15 %

C: 17%

DBA	GRADE DATE
BORO	RECORD DATE
BUILDING	INSPECTION TYPE
STREET	Latitude
ZIPCODE	Longitude
CUISINE DESCRIPTION	Community Board
INSPECTION DATE	Council District
ACTION	Census Tract
VIOLATION CODE	BIN
VIOLATION DESCRIPTION	BBL
CRITICAL FLAG	NTA

# Preprocessing 2

- The dataset has too many "A" grades, causing classifiers to mostly predict "A" and perform really poorly on minority grades ("B" and "C").
  - Bad accuracy (60%) , precision (0.4<), and recall (0.5<)
- Weka's SMOTE filter generates synthetic instances of minority classes, helping balance the class while still being representative of the population.
  - Don't overuse so models don't overly depend on synthetic instances instead of the actual ones.
- Reduces bias toward the majority class, allowing for better performance and more accurate predictions across all grades.



# Dataset after Preprocessing 2

Instances: 2118

Attributes: 23

Class Proportions:

A: 48 %

B: 25 %

C: 27%

DBA	GRADE DATE
BORO	RECORD DATE
BUILDING	INSPECTION TYPE
STREET	Latitude
ZIPCODE	Longitude
CUISINE DESCRIPTION	Community Board
INSPECTION DATE	Council District
ACTION	Census Tract
VIOLATION CODE	BIN
VIOLATION DESCRIPTION	BBL
CRITICAL FLAG	NTA

# Intuition

- DBA (Chain Name) : Chain health standards vary.
- Building: Shared sanitation standards/issues.
- Street: Upscale = stricter standards, high-traffic = more wear and tear.
- Cuisine: Specialized handling affects violations.
- Action Taken: Reflects cleanliness management.
- Violation Code: Indicates specific health risks.
- Violation Description: Detailed violation reason..
- Critical Flag: Severe violations lead to lower grades.
- Longitude & Latitude: Location affects inspection outcomes.
- Community Board: Varies in regulatory strictness.
- Council District: Influences regulations and resources.

# InfoGainAttributeEval

Measures how much knowing an attribute reduces uncertainty about the target class.

Attributes with an information gain above 0.5 were selected.

$$\text{Entropy} = \sum_i -p_i \log_2 p_i$$

$$IG(A) = H(D) - \sum_{v \in \text{Values}(A)} \frac{|D_v|}{|D|} H(D_v)$$

1.2051	Latitude
1.2051	Longitude
1.1905	BBL
1.1861	DBA
1.1711	BIN
0.9961	BUILDING
0.6694	GRADE DATE
0.5982	INSPECTION DATE
0.564	Census Tract
0.5568	STREET

Location + Date Attributes

# GainRatioAttributeEval

Similar to InfoGain but adjusts scores to prevent attributes with many possible values from being favored (Normalization).

Attributes with a gain ratio above 0.25 were selected.

$$SI(A) = - \sum_{v \in \text{Values}(A)} \frac{|D_v|}{|D|} \log_2 \left( \frac{|D_v|}{|D|} \right)$$

$$GR(A) = \frac{IG(A)}{SI(A)}$$

Location + Date  
+ Violation

0.28012	ACTION
0.11639	DBA
0.11555	Longitude
0.11555	Latitude
0.11441	BBL
0.11273	BIN
0.10062	BUILDING
0.08119	GRADE DATE
0.07668	INSPECTION DATE
0.06548	STREET
0.0649	Census Tract
0.0447	VIOLATION DESCRIPTION
0.03475	VIOLATION CODE
0.03022	NTA
0.02594	CRITICAL FLAG

# ClassifierAttributeEval

Assesses attribute importance based on how well they improve a specific classifier's predictions.

- Default classifier was used: zeroR

The best 17 attributes were selected.

Location  
w/o long & lat +  
Date + Violation

NTA	
ACTION	
VIOLATION CODE	
CUISINE DESCRIPTION	
INSPECTION DATE	
ZIPCODE	
BBL	
BORO	
BUILDING	STREET
	VIOLATION DESCRIPTION
	CRITICAL FLAG
	GRADE DATE
	Census Tract
	BIN
	RECORD DATE
	Council District

# CorrelationAttributeEval

Evaluates the relationship between each attribute and the target class using correlation coefficients.

Attributes with a correlation of 0.25 or higher were selected.

$$r = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sqrt{\sum (x - \bar{x})^2 \sum (y - \bar{y})^2}}$$

ACTION
INSPECTION TYPE
CRITICAL FLAG
VIOLATION CODE
GRADE DATE
VIOLATION DESCRIPTION
CUISINE DESCRIPTION
ZIPCODE
BORO
Council District
Community Board
DBA
NTA

Location  
w/o long & lat + Violation +  
Governance

# OneR

Simple algorithm that creates classification rules based on a single attribute, selecting the one with the lowest error rate for surprisingly accurate results.

```
a    b    c  <-- classified as
1008  4    8 |    a = A
159  339   1 |    b = B
171   1  427 |    c = C
```

InfoGain

```
a    b    c  <-- classified as
1000  14    6 |    a = A
165  331    3 |    b = B
188   3  408 |    c = C
```

Correlation

```
a    b    c  <-- classified as
1007   5    8 |    a = A
154  344    1 |    b = B
171   2  426 |    c = C
```

Intuition

A lot of False As

```
a    b    c  <-- classified as
1007   5    8 |    a = A
149  349    1 |    b = B
164   2  433 |    c = C
```

GainRatio

```
a    b    c  <-- classified as
998  10   12 |    a = A
156  340    3 |    b = B
184   1  414 |    c = C
```

Classifier

# Random Tree

Builds multiple decision trees using random subsets of attributes, reducing overfitting and improving generalization to new data.

a	b	c	<-- classified as
1006	6	8	a = A
157	339	3	b = B
171	1	427	c = C

InfoGain

a	b	c	<-- classified as
863	77	80	a = A
127	347	25	b = B
159	25	415	c = C

Correlation

A lot of False As

a	b	c	<-- classified as
981	22	17	a = A
156	341	2	b = B
163	2	434	c = C

GainRatio

a	b	c	<-- classified as
929	46	45	a = A
152	325	22	b = B
156	15	428	c = C

Classifier

Intuition

a	b	c	<-- classified as
978	22	20	a = A
158	338	3	b = B
168	4	427	c = C



# Decision Table

Tabular representation that combines attribute values with class labels, simplifying classification by focusing on the most relevant features.

a	b	c	<-- classified as
1014	2	4	a = A
190	309	0	b = B
199	0	400	c = C

InfoGain

a	b	c	<-- classified as
1005	6	9	a = A
187	311	1	b = B
195	0	404	c = C

Correlation

a	b	c	<-- classified as
1011	4	5	a = A
192	306	1	b = B
202	1	396	c = C

Intuition

a	b	c	<-- classified as
1011	4	5	a = A
204	294	1	b = B
212	1	386	c = C

GainRatio

a	b	c	<-- classified as
1000	10	10	a = A
178	319	2	b = B
199	2	398	c = C

Classifier

A lot of False As

# NaiveBayes

Probabilistic classifier based on Bayes' Theorem. Assumes that each attribute is independent of the others.

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)}$$

a	b	c	<-- classified as
694	163	163	a = A
38	433	28	b = B
31	40	528	c = C

Intuition

a	b	c	<-- classified as
445	318	257	a = A
21	441	37	b = B
17	39	543	c = C

InfoGain

a	b	c	<-- classified as
686	180	154	a = A
43	405	51	b = B
36	42	521	c = C

Correlation

Lot of As classified wrong

a	b	c	<-- classified as
620	214	186	a = A
15	453	31	b = B
13	36	550	c = C

GainRatio

a	b	c	<-- classified as
619	214	187	a = A
27	441	31	b = B
22	33	544	c = C

Classifier

# Analysis

- Accuracy Trends
  - OneR: 82% - 84.5%, highest at 84.47% (GainRatio).
  - Random Tree & Decision Table: Low 80% range.
  - NaiveBayes: Low 70s Range, lowest at 67.47% (InfoGain).
- Precision and Recall
  - OneR: Precision 0.875, Recall 0.845.
  - Random Tree: Slightly lower than OneR in metrics.
  - NaiveBayes: Underperforms in Recall, misses more positive instances. (0.675 under InfoGain)
- False Positive Rate
  - NaiveBayes: Low FP rate (0.089 under GainRatio).
  - OneR: Slightly higher FP rate (0.123) but still effective.
- ROC Area
  - NaiveBayes: Highest ROC Area (0.964 under GainRatio), excels at distinguishing classes.
  - Everything else as mid range ROC Area, showing decent class separation.

# Results

Top 10 overall best performing:

1. GainRatio OneR
2. Intuition OneR
3. InfoGain OneR
4. GainRatio RandomTree
5. Classifier OneR
6. Correlation OneR
7. Intuition RandomTable
8. InfoGain RandomTree
9. GainRatio DecisionTable
10. Intuition DecisionTable

Want to minimize False Positives so rating is decided based on greater Precision.

- It's better for a healthy restaurant to be misclassified as unhealthy than for an unhealthy restaurant to be misclassified as healthy.

$$Precision = \frac{TP}{TP + FP}$$

# Best Classification Model

GainRatio OneR provides the best combination of critical metrics for predicting health inspection grades.

- Leads in accuracy at 84.47%.
- Achieves a the highest precision of 0.875 and recall of 0.845, balancing correctness and comprehensiveness.
- Maintains a false positive rate of 0.123, minimizing misclassifications.
- Its overall metrics make it a reliable choice.

# Future

I still noticed that there were quite a lot of False As which tells me that the classifier is still being affected by the abundance of the A class value.

- I want to perhaps use more of Weka's SMOTE filter to further better the class proportions
- Experiment with oversampling to get a dataset with better class proportions from the original population without creating synthetic values