HEALTH SCORE PREDICTOR

GROUP 06

DATA SCIENCE WITH PYTHON

- Problem and Motivation
- Data Source
- Notebook map
- Data Exploration
- Feature Engineering
- Balancing and building dataset
- Model
- Results and Insights

Problem and Motivation

- Limited inspection bandwidth leads to reactive scheduling high risk restaurants are often found late due to fragmented data and manual heuristics.
- Future outcomes (pass/fail/conditional) depend on historical violations, time since last inspection, neighborhood risk, and inspection type hard to weigh consistently without a predictive tool.
- City agencies lack a unified, data-driven way to prioritize inspections, risking missed hotspots and inefficient routes.

- Use SF health inspections plus public signals (Google ratings) to forecast violation risk and proactively target HIGH-risk establishments.
- Provide an inspector-facing dashboard with trends, neighborhood risk, and per-business predictions to inform scheduling and resource allocation.
- Demonstrate ML (RF or XGBoost) can outperform baseline heuristics, improving public health outcomes and fairness with transparent, data-backed decisions.

Data Source

- SF Health Inspections (2020–Present):

Primary labeled records of inspections, violations, and outcomes. These are the raw records which are found separately in the SF health department website data.sfgov.org/Health-and-Social-Services

HealthInspection(2020-2023).csv HealthInspection(2024-present).csv

- Google Places Signals:

Ratings and review counts to proxy hygiene perception. This data is extracted using Google cloud API: Geocoding API and Places API.

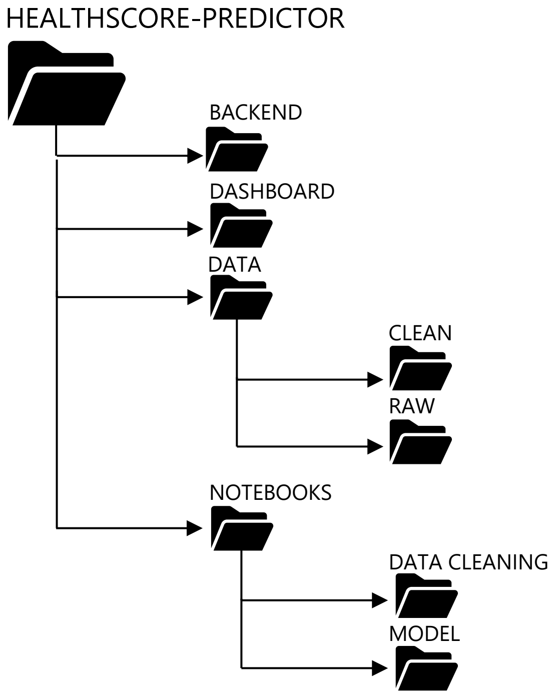
Google_clean.csv

- Neighborhood Crosswalk:

Mapping neighborhoods/ZIPs for spatial features and one-hot encoding. Cleaned up dataset for visualization and model run

model_dataset.csv

Notebooks Map



BACKEND – Consists of the End points, Data inputs and Model serving

DASHBOARD – React frontend framework for visualization and user interaction.

DATA/CLEAN – PreProcessed csv file useful for the model run and visual ease

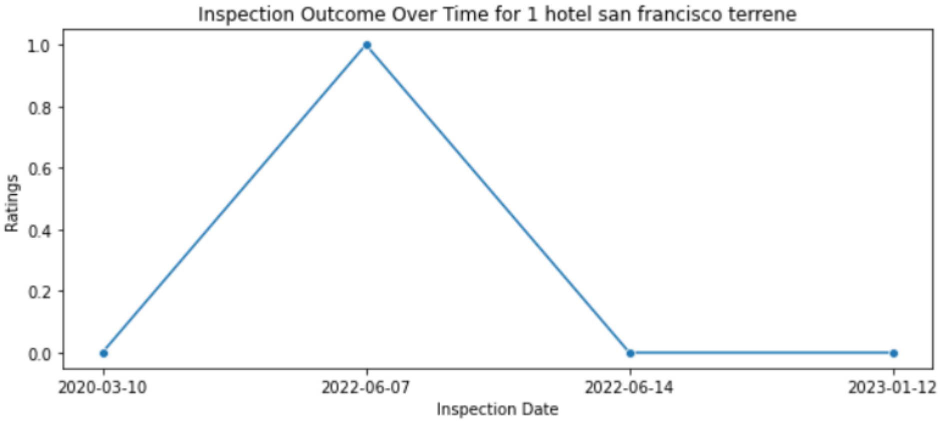
DATA/RAW – Untouch dataset extracted.

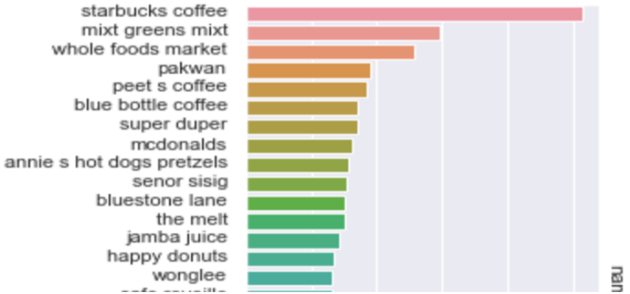
NOTEBOOKS/DATACLEANING – Data cleaning, standardizing, schema-aligned datasets ready for EDA, modeling, and the dashboard.a

NOTEBOOKS/MODEL – Execution of our chosen model over the built dataset

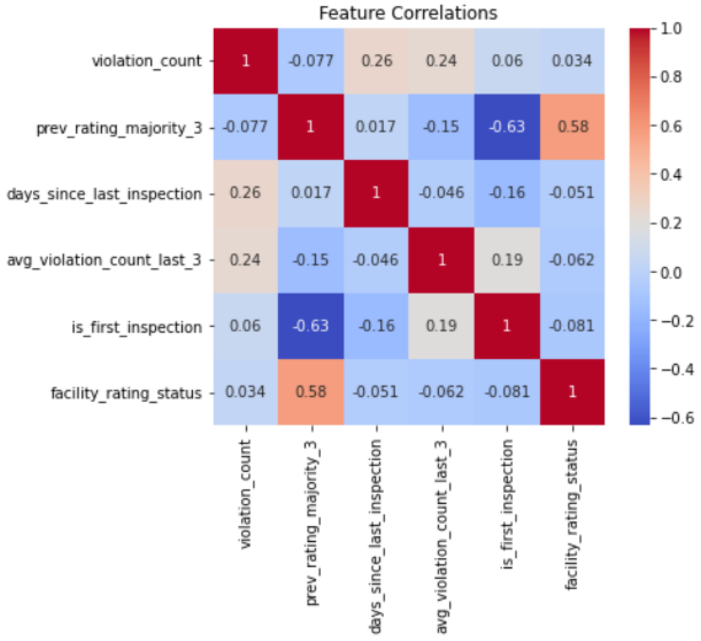
OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

Data Exploration





	percentage
facility_rating_status	
Pass	85.0
Conditional Pass	10.2
Closure	4.7



Data Cleaning and Feature Engineering

Text normalization

Column names:
Standardized to lowercase with underscores

String trimming:
Removed leading/trailing spaces in all text columns.

Address & name canonicalization:
Added canonical names dress by normalizing suffixes, removing unit markers, and slugifying text.

Corrections:
Standardized facility status variants (typos in "Conditional Pass").

Merging & Alignment (2020 vs 2024)

Column alignment:
Renamed 2020 columns to match 2024 schema

Column pruning:
Dropped administrative/geospatial columns not needed for modeling (e.g., inspector, district, point, etc.).

Dataset merge:
Concatenated aligned 2020 and 2024 frames into df_merge.

Feature Engineering

Recent rating:
Most common inspection outcome across the last three inspections for the same facility.

Recent Violation:
Average number of violations over the previous three inspections.

Inspection gap (in days):
Elapsed time in days since the facility's previous inspection.

First Inspection : Yes or NO.

Imputation & Null handling

Violations (2024):
Imputed violation_count missing to 0 imputed violation_codes missing to empty string.

Violations (merged):
Filled violation_count missing to 0 and

created has_violation_count to differentiate two datasets

Lat and Log impute:
Data extracted from google_data and imputed for missing lat and log

Handling Imbalance

Balancing Techniques Evaluated
SMOTE
BorderlineSMOTE
Random Over-Sampling
Random Under-Sampling

Model-Based Imbalance Handling
class_weight used in **Random Forest**
sample_weight used in **XGBoost**

Key Observations

- Using class weights on the original dataset outperformed methods that introduced synthetic data
- Random Under-Sampling produced results similar to the original dataset
- Over-sampling and SMOTE degraded model performance Synthetic data introduced noise
- Led to poorer generalization

Model Benchmarking

The Challenge: Class Imbalance

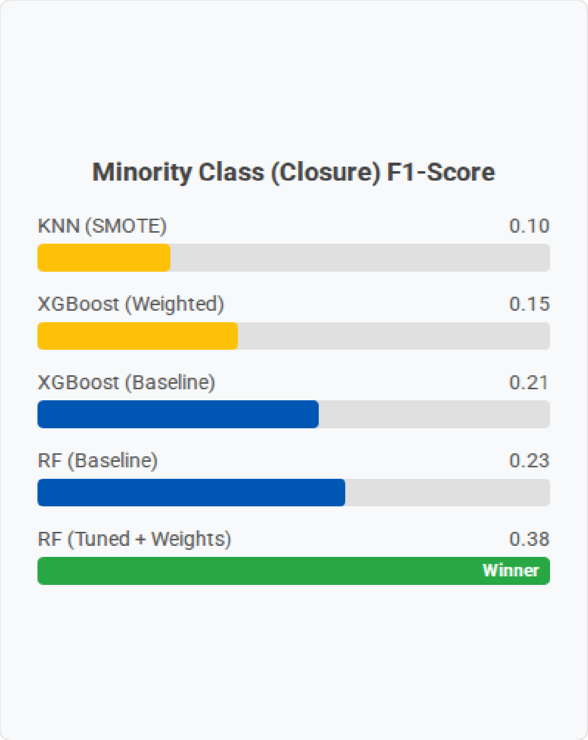
The dataset is heavily skewed towards "Pass" ratings. Detecting "Closures" (Class 2) is difficult but critical for public safety.

Selection Rationale

XGBoost (Baseline): High accuracy (90%) but failed to detect closures (Recall ~0.25).

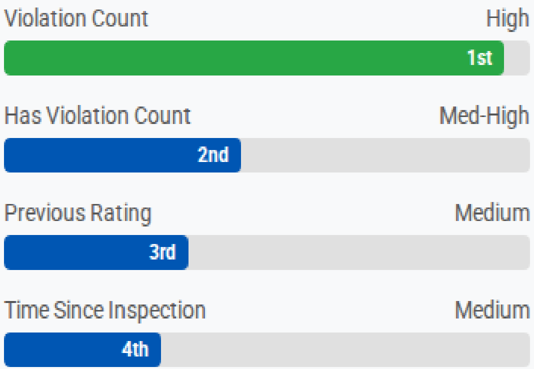
RF (Undersampling): Improved recall but lost too much training data.

Winner: RF (Tuned + Weights): We used custom class weights {0:1, 1:2, 2:4}. This provided the best balance of stability and precision.



Results & Strategic Insights

Key Drivers (Feature Importance)



94.1%
OVERALL ACCURACY

0.85
CLOSURE PRECISION

0.68
OPTIMAL THRESHOLD

Business Recommendations

Prioritize Risk: The sheer volume of current violations is the #1 predictor. Facilities with high violation counts should trigger immediate senior officer review.

History Matters: Past performance is predictive. "Conditional Pass" history correlates with future Closures.

Deployment: Use the model to flag "At-Risk" facilities *before* inspection to allocate resources efficiently.

OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG

QUESTIONS?

OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG