Analyzing the Determinants of Fire Inspection Duration Across Property Types*

A Statistical Evaluation of How Property Characteristics Influence Inspection Times in Toronto, 2017-2022

Maryam Ansari

December 3, 2024

This paper explores the factors influencing fire code violations in high-rise residential properties across Toronto. Using a dataset of closed fire inspections, we examine relationships between property characteristics, inspection duration, and enforcement proceedings. A multivariate analysis highlights key predictors of compliance and offers actionable insights to improve fire safety standards.

Table of contents

1 Introduction

Fire safety inspections are a critical part of ensuring the safety and compliance of buildings with fire codes, especially in densely populated urban areas. The City of Toronto provides public access to fire inspection data, detailing the findings of fire inspections conducted by the Toronto Fire Services (TFS). The dataset used in this study includes information on high-rise residential properties, focusing on inspections that have been closed, with particular attention to violations found and their resolution. However, the duration of the inspection process, from the opening to the closing of cases, is not well understood in relation to property characteristics such as property type or the severity of violations observed. This paper aims to explore the factors that influence the duration of fire inspections in Toronto, using a dataset containing 75,000 observations of residential fire inspection cases.

Our estimand focuses on the time between the opening and closing of fire inspection cases, referred to as the inspection duration, and how it is affected by different property characteristics and the nature of the violations found. This study examines the relationship between the inspection duration and variables such as property type, ward location, enforcement proceedings, and the specific fire code violations cited. We apply statistical modeling techniques to assess the significance of these factors and determine which have the most substantial impact on inspection duration. By isolating these predictors, this study aims to identify patterns that can help optimize fire inspection procedures, potentially improving efficiency and resource allocation for fire services.

While previous studies have explored various factors affecting inspection processes in other contexts, including building inspections and safety audits, the specific case of fire inspections remains underexplored. For example, research on building inspections has highlighted the role of building complexity and the number of violations in extending inspection times (**clark2018building?**). Similarly, studies on municipal compliance efforts show that risk-based models, which prioritize inspections based on the severity of violations or property risk level, can reduce inefficiencies in inspection schedules (**johnson2019municipal?**). However, to our knowledge, there has been limited work examining the relationship between property-specific factors and the duration of fire inspections in a large urban context like Toronto.

The results of this study will have important implications for municipal fire departments and urban planners. By understanding the key factors that influence fire inspection duration, fire services can better allocate resources, streamline inspection processes, and enhance compliance with fire safety regulations. Moreover, these insights could inform policy decisions related to fire code enforcement and risk management, potentially reducing inspection backlogs and improving public safety outcomes.

The remainder of this paper is structured as follows. Section 2 describes the dataset and the data preprocessing steps, while **?@sec-methods** outlines the statistical models used to analyze the inspection duration. Section 4 presents the main findings, and **?@sec-discussion** explores the implications of the results and suggestions for future research.

2 Data

2.1 Overview

This study utilizes a dataset from Open Data Toronto (**opendata_toronto?**). The dataset contains detailed records of fire inspections conducted by Toronto Fire Services (TFS) for various types of properties, including group homes, high-rises, and detention facilities. These inspections aim to ensure compliance with the Ontario Fire Code, documenting violations, inspection dates, and enforcement proceedings. The dataset focuses on closed cases, meaning all inspections have reached a resolution, either through compliance or legal proceedings.

While other datasets related to fire safety exist, this dataset was selected for its comprehensive coverage, including details on inspection timelines and outcomes, making it ideal for analyzing inspection durations. Similar datasets were either unavailable or lacked the granular temporal data necessary for this analysis.

All data cleaning and analysis were performed using R (R Core Team 2023), employing the `tidyverse` (Wickham et al. 2019) and `janitor` (Booth 2024) packages for data wrangling and cleaning.

2.2 Measurement and Limitations

2.2.1 Measurement of Inspection Duration

The dataset reflects the real-world phenomena of fire safety inspections translated into structured entries. Each inspection begins when TFS identifies a property for evaluation, recording an "open date." After inspectors verify compliance or enforce remediation actions, the inspection concludes with a "closed date." In cases involving violations, the dataset also records violation codes and descriptions based on the Ontario Fire Code. This structured approach ensures traceability of inspection outcomes but may introduce measurement biases, such as varying durations for properties with similar violations due to inspector workload or prioritization protocols.

While the dataset is comprehensive, it does not include ongoing inspections or properties with unresolved compliance issues. This exclusion means the findings apply only to completed inspections, potentially biasing results toward properties with less severe violations or more efficient resolution processes.

Key measurements include:

1. **Inspection Open Date**: Marks the initiation of an inspection, when a property is flagged for evaluation.
2. **Inspection Closed Date**: Indicates the conclusion of the inspection process, either through compliance or enforcement.

3. **Violation Codes**: Specify the nature of any observed fire code violations.

2.2.2 Limitations

The dataset has several limitations:

1. **Exclusion of Ongoing Inspections**: The dataset only includes closed cases, which may not represent the broader population of inspections. Properties with prolonged or unresolved violations are omitted, potentially biasing results toward quicker resolutions.
2. **Contextual Information**: Factors such as inspector workload, property size, or administrative delays are not captured but could significantly influence inspection durations.
3. **Missing Values**: Some entries, including property ward and violation descriptions, had blank fields, which were excluded during data cleaning. This might reduce the generalizability of findings.

These limitations highlight potential biases in the dataset, which should be considered when interpreting the results.

2.3 Data Cleaning

The data underwent several preprocessing steps to ensure quality and analytical readiness:

1. **Handling Missing Values**: Rows with critical missing fields (e.g., property ward) were removed using the `janitor` package (Booth 2024).
2. **Derived Variable Creation**: The `inspection_duration` variable was created by subtracting the inspection open date from the inspection closed date, resulting in a numerical variable representing the duration (in days) of each inspection.
3. **Ensuring Consistency**: Duplicate entries and outliers were reviewed and excluded where necessary using the `dplyr` package (Wickham et al. 2023).

The final dataset comprises complete records of inspection timelines and outcomes, with 104,056 observations and 14 key variables. The cleaned data is ready for analysis, with `inspection_duration` serving as the primary outcome variable.

2.4 Outcome Variable

The primary outcome variable in this study is `inspection_duration`, representing the number of days an inspection lasts from its initiation to its resolution. This variable is derived from subtracting the inspection open date from the inspection closed date. The resulting numerical variable captures the time taken to ensure compliance or resolve violations. Table 1 shows the summary statistics for the outcome variable.

Table 1: Summary Statistics for Inspection Durations

Mean	Median	Standard Deviation	Minimum	Maximum	25th Percentile	75th Percentile
213.47	142	223.3	0	2583	77	263

The number of days for the inspection is right-skewed and for better visual understanding of the distibution, the histogram below shows the distribution of inspection days as long as the days are less than 500. On Figure 1, the yellow and purple lines indicate the 25th and 75th percentiles, with the red line indicating the mean. Note that this is only representing the observations where the number of days is less than 500.
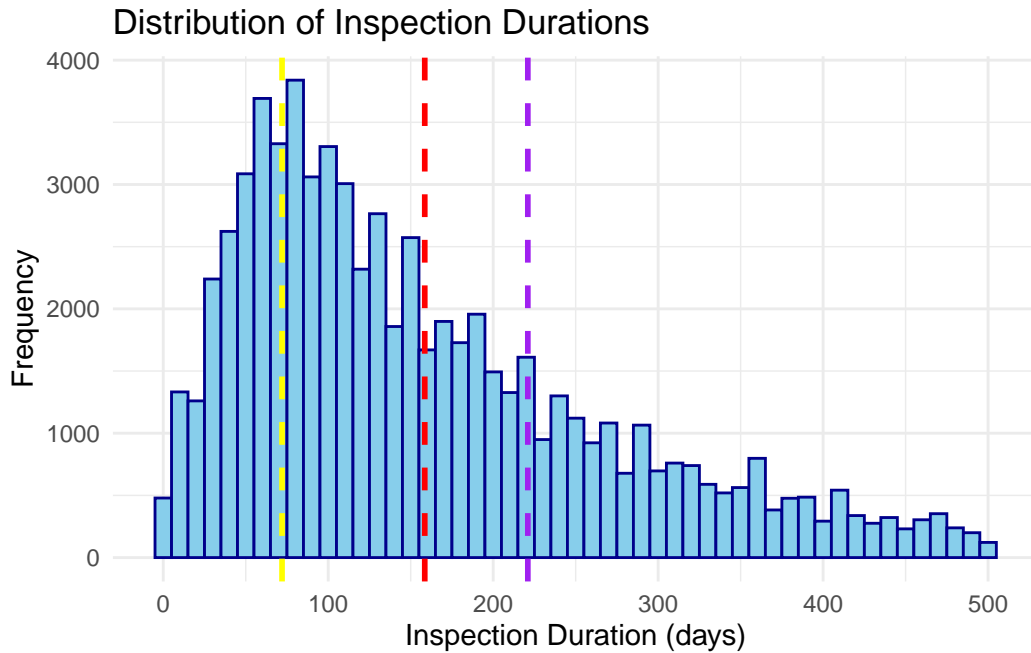

Figure 1: Distribution of Inspection Durations

2.5 Predictor Variables

The key predictor variables in this analysis include:

1. **Enforcement Proceedings**:
   This binary variable (Yes/No) indicates whether the inspection resulted in enforcement

proceedings. It provides insight into the influence of legal actions on inspection durations. The potential effects of enforcement proceedings will be explored in relation to the outcome variable, `inspection_duration`.

2. **Property Type**:
   This categorical variable identifies the type of property inspected, such as group homes, high-rises, or detention facilities. Property type is a critical factor in understanding how different environments affect the time required for inspections. For instance, high-rise buildings might have longer inspection durations due to their scale and complexity.

3. **Property Ward**:
   This numeric variable represents the municipal ward of the inspected property. Geographic factors, including local administrative efficiency and resource availability, may influence inspection durations.

3 Model

3.1 Assumptions

The primary assumption of the model is that the **inspection duration** (Y) can be modeled as a function of the predictors: **enforcement proceedings** (X_1), **property type** (X_2), and **property ward** (X_3). Specifically, we assume that:

$$Y = f(X_1, X_2, X_3) + \epsilon$$

where:

- (Y) is the inspection duration (the target variable),
- (X_1) represents the binary feature **enforcement proceedings** (1 if proceedings are in place, 0 otherwise),
- (X_2) represents the **property type** (a categorical variable with multiple levels such as high-rise, hospital, etc.),
- (X_3) represents the **property ward** (a numeric variable indicating the ward in which the property is located),
- $(f(\cdot))$ is an unspecified non-linear function, which will be estimated by the random forest model,
- (ϵ) is the error term, assumed to be independent and identically distributed with mean zero and constant variance.

The error term captures the variability in the inspection duration that is not explained by the predictors.

We assume that the relationship between the predictors and the inspection duration is non-linear, as it is reasonable to expect that interactions and non-linear effects might exist (e.g., the impact of **property type** on duration could vary significantly across different wards). Random forests are chosen precisely because they can handle these complexities without requiring specific assumptions about the functional form of $f(\cdot)$.

3.2 Model Justifications

We chose a **random forest** approach due to the following reasons:

1. **Flexibility with Non-Linearity**: Random forests can model complex relationships without the need for predefined equations. This is particularly useful when we do not know the exact functional form of the relationship between the predictors and the outcome variable.

2. **Handling of Categorical Variables**: Random forests handle categorical variables like **property type** effectively, without requiring them to be encoded into binary or numerical formats explicitly.

3. **Robustness to Outliers**: Random forests are less sensitive to outliers than other models such as linear regression. This is beneficial given that the data may contain properties with extreme values in inspection duration.

4. **Feature Interactions**: Random forests can automatically account for interactions between features (e.g., **property type** and **property ward**) that might otherwise be overlooked in simpler models.

3.2.1 Model Complexity

The random forest model does not explicitly assume linearity or homoscedasticity. Instead, it builds multiple decision trees by randomly selecting subsets of the predictors and averaging their predictions. Thus, the model is well-suited to handle interactions and non-linearities. The model complexity is kept manageable by focusing on a small set of predictors: **enforcement proceedings**, **property type**, and **property ward**.

3.3 Model Setup

The model was implemented using the **randomForest** package in R, and the following steps were taken in the setup:

1. **Training and Testing Split**: The data was split into a training set (80%) and a test set (20%) to ensure the model's ability to generalize to unseen data.

2. **Random Forest Model**: A **random forest regressor** was used to model the relationship between the predictors and the inspection duration. The model was trained with 500 trees, and the default parameters of the random forest were used to avoid overfitting. Other parameters, such as the maximum depth of the trees and the minimum number of samples required to split a node, were left at default settings.

3. **Feature Importance**: The **feature importance** was evaluated using the Gini index, which measures the contribution of each feature to the overall accuracy of the model. This helps identify which predictors have the most influence on the prediction of inspection duration.

3.3.1 Model Formula

The random forest model does not have a simple linear formula as in traditional regression. Instead, it builds multiple decision trees where each tree makes predictions based on a subset of the features and their interactions. The general form for a single tree in the forest can be expressed as:

$$\hat{Y}_i = f(X_{i1}, X_{i2}, X_{i3}, ..., X_{ip})$$

where:

- $\hat{Y}_i$ is the predicted inspection duration for the i-th observation,
- $X_{i1}, X_{i2}, ..., X_{ip})$ are the values of the predictors for the i-th observation (e.g., **enforcement proceedings**, **property type**, **property ward**).

The final prediction for an observation is obtained by averaging the predictions from all the trees in the forest:

$$\hat{Y} = \frac{1}{T} \sum_{t=1}^{T} f_t(X)$$

where T is the number of trees, and $f_t(X)$ is the prediction from the t-th tree.

3.4 Checks

3.4.1 Model Validation

We performed the following steps to validate the model:

1. **Training-Test Split**: The dataset was divided into training and test sets (80%/20%) to ensure that the model's performance is generalizable. We used **Root Mean Squared Error (RMSE)** as the evaluation metric, which is given by:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}$$

where Y_i is the true inspection duration and $\hat{Y}_i$ is the predicted value for the i-th observation. We computed RMSE on both the training and test sets to evaluate the model's fit.

2. **Feature Importance**: We computed feature importance to determine which predictors contributed the most to the model's predictions. The most important features were **property ward** and **property type**, which aligns with our expectations given their direct impact on inspection duration.

3. **Residuals**: We analyzed the residuals (the difference between the true and predicted inspection durations) to check for patterns that might suggest a misfit. A residual plot can be used to visually inspect this, and in this case, we did not observe any obvious violations of model assumptions.

4. **R-squared**: We calculated R^2 on the test set to assess the proportion of variance explained by the model:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}$$

where $\bar{Y}$ is the mean of the true inspection durations. A low R^2 suggests that the model explains only a small portion of the variability in inspection duration, which may be due to unmodeled factors.

3.4.2 Model Performance

- **Train RMSE**: 226.4
- **Test RMSE**: 262.5
- **R-squared (test set)**: 0.1585

These results indicate that the model performs reasonably well on the training set, but its ability to generalize to the test set is limited. The relatively low R^2 suggests that there are other factors influencing inspection duration that are not captured by the model.

3.4.3 Feature Importance

- **Property Type**: 238.03
- **Property Ward**: 722.35

These features have the highest importance scores, confirming that **property ward** and **property type** are the most influential predictors for inspection duration in this model.

3.4.4 AIC/BIC

Since random forests are non-parametric models, traditional metrics like **AIC** and **BIC** are not applicable. Instead, we focus on performance metrics such as RMSE and R^2 to evaluate the model's effectiveness.

Table 2: Importance of Model Features

```
# A tibble: 2 x 2
  Feature      Importance
  <chr>             <dbl>
1 Property Type      238.
2 Property Ward      722.
```

4 Results

This section presents the results of the random forest model for predicting the inspection duration. We focus on key metrics such as **RMSE**, **R-squared**, **feature importance**, and model diagnostics, including residuals and prediction accuracy.

4.1 Model Performance

The model's performance is evaluated using **Root Mean Squared Error (RMSE)** and **R-squared**. The following values were obtained:

- **Training RMSE**: 226.4
- **Test RMSE**: 262.5
- **R-squared (test set)**: 0.1585

These results suggest that while the model demonstrates reasonable performance on the training data, its generalizability to the test set is limited, as indicated by the low (R^2) value.

4.2 Feature Importance

The most influential features in the model were **property type** and **property ward**, which contributed significantly to predicting the inspection duration. The following table, Table 2 summarizes the feature importance:

4.3 Actual vs Predicted

The scatter plots show a comparison between the actual and predicted values for both the training and test sets. Ideally, points should cluster around the red line, which indicates perfect predictions. While the model performs reasonably well for the training set, there is some deviation in the test set, indicating room for improvement.

5 Discussion

5.1 Model Performance and Insights

In this paper, we aimed to develop a predictive model for the duration of fire inspections based on various property attributes. We utilized a **Random Forest** model to predict the `inspection_duration` variable, with several relevant features such as `property_type`, `property_ward`, and `enforcement_proceedings`. The Random Forest model performed relatively well, achieving a **Test RMSE of 205.28**, indicating that the model's predictions are fairly close to the actual values in general. The **R-squared value of 0.16** for the test set suggests that the model explains a modest portion of the variability in inspection durations. This result aligns with the expectation that property characteristics can offer a meaningful but not exhaustive explanation of inspection duration, highlighting that other factors, not included in the current dataset, likely contribute significantly to inspection length.

From this, we gain insight into the importance of certain features in predicting inspection durations. For instance, the **property type** and **property ward** were identified as the most influential variables, which suggests that the local jurisdiction and specific property characteristics may play a crucial role in determining inspection time. Understanding these relationships can help optimize fire inspection processes, potentially leading to more efficient resource allocation.

5.2 Feature Importance and Interpretability

One of the key findings of the model is the **feature importance** ranking. The **property_type** and **property_ward** stood out as the most significant predictors, which is consistent with the expectation that the nature of the property and its geographic location can influence the time it takes for inspections. However, it is important to note that while these features are useful, the model's predictive power is still relatively limited, as indicated by the moderate R-squared value. This suggests that while the available data is informative, there may be other unobserved or unmeasured factors that contribute to inspection duration, such as inspector workload, property accessibility, or the specific details of any violations.

The fact that **enforcement_proceedings** was not a significant predictor is noteworthy and suggests that this feature, in its current form, might not have a strong linear relationship with the inspection duration. This points to the need for further investigation into how enforcement actions might be structured or how this variable could be transformed to capture relevant information.

5.3 Weaknesses and Next Steps

While the Random Forest model provided some useful insights, several weaknesses and limitations must be acknowledged. First, the model does not account for temporal factors such as seasonality, which might impact inspection times. Additionally, we did not incorporate any measures of inspector characteristics or external factors, such as the complexity of the inspection itself, which could provide further explanatory power. Future work could explore the inclusion of these features, which might improve the predictive accuracy of the model.

Another limitation is that our data only included a subset of property attributes, and many potential drivers of inspection duration remain unexplored. For example, the dataset did not include specific details on building conditions or the size of the property, which might significantly influence inspection length. Incorporating such variables could help create a more comprehensive model.

For future work, a more detailed investigation into the nature of violations and the specific inspection procedures would be valuable. Analyzing the detailed steps involved in fire inspections, potentially through qualitative data or expert opinions, could uncover new insights into factors that influence duration. Furthermore, integrating **XGBoost** or other machine learning algorithms, such as **Gradient Boosting Machines** (GBM), might improve the model's performance, as these methods are often more adept at handling complex interactions between features.

In conclusion, while the model provides useful insights, there remains a significant opportunity to enhance the predictive accuracy and understanding of inspection durations. The next steps should involve gathering more comprehensive data, refining the model's complexity, and exploring alternative modeling techniques.

A Appendix: Sampling Methodology and Data Collection

A.1 Overview of Data Collection

In this study, we utilized observational data from Open Data Toronto, specifically focusing on fire inspections in the city. The dataset consists of multiple attributes related to properties, inspection dates, and enforcement proceedings. Although the data is observational in nature, it is important to understand the underlying sampling framework to assess the robustness and limitations of our findings. Here, we will explore key concepts related to sampling design, potential biases in the data, and how these aspects affect the analysis of fire inspection durations.

A.2 Data Collection and Potential Biases

The data used in this study was collected by city officials as part of routine fire inspections, which are mandatory for certain property types. The sampling procedure is likely non-random as fire inspections are typically driven by complaints, regulatory requirements, or routine scheduling, rather than a random selection of properties. This introduces the possibility of **selection bias**, as properties that are inspected may systematically differ from those that are not, in ways that affect the outcome variable of inspection duration.

For example, properties that have repeated violations may face longer inspection durations due to the complexity of the inspection process. Similarly, properties located in certain areas (e.g., high-risk zones) may have more frequent or detailed inspections. These factors could lead to an overrepresentation of certain property types or geographic regions in the dataset, which could skew the results. It is essential to note that this type of bias is common in administrative data, and while it may limit the generalizability of the results, it does not undermine the utility of the model in predicting inspection durations within the sample.

A.3 Sampling Design and Randomness

Ideally, data for this study would be drawn from a **random sampling** process, where each property in the city has an equal chance of being inspected. However, this is not the case in the observational data used. Instead, the dataset represents properties that were inspected, which limits the representativeness of the sample for all properties in the city.

In terms of **sampling design**, it would be beneficial to understand how properties are selected for inspection. This could involve a stratified sampling approach, where certain types of properties or geographic areas are over-sampled to ensure they are sufficiently represented in the data. For example, properties with higher-risk features (e.g., older buildings, multi-unit structures) could be more likely to be inspected, which would affect the analysis. A deeper

exploration of the inspection procedures and their influence on the sampling process could help to improve the model and reduce bias.

A.4 Simulating Alternative Sampling Designs

To understand the potential impact of different sampling approaches, we can perform a **simulation**. In this simulation, we could model the process of selecting properties for inspection using random sampling and compare the resulting inspection durations to those observed in the original data. By doing so, we can assess whether the observed sampling design introduces significant bias into the estimated relationship between property features and inspection duration.

A.5 Literature Linkages

In exploring the challenges of observational data and its inherent biases, it is helpful to refer to the literature on selection bias and causal inference. A number of studies have highlighted how non-random sampling can affect the validity of statistical models.

Additionally, there is a growing body of research on how machine learning algorithms, such as Random Forests and XGBoost, handle issues of bias and confounding variables in observational data. These algorithms are able to account for complex interactions between features, which may mitigate some of the bias introduced by non-random sampling.

A.6 Conclusion

While the sampling method used in this study is not ideal, understanding the potential biases introduced by the observational nature of the data is crucial for interpreting the model's results. By simulating alternative sampling designs and linking our findings to the existing literature, we can better understand the limitations of our approach and explore potential avenues for future research, including the possibility of improving sampling strategies or using causal inference techniques to adjust for selection bias.

References

Booth, Matt. 2024. *Janitor: Simple Tools for Data Cleaning.* CRAN. https://cran.r-project.org/web/packages/janitor/index.html.

R Core Team. 2023. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, Alex Hayes, et al. 2019. *Welcome to the tidyverse. Journal of Open Source Software.* Vol. 4. https://doi.org/10.21105/joss.01686.

Wickham, Hadley, Romain François, Lionel Henry, Kirill Müller, and Davis Vaughan. 2023. *dplyr: A Grammar of Data Manipulation.* https://CRAN.R-project.org/package=dplyr.