NYC Airbnb Data Profiling: From Raw Chaos to Clean Insights

A Comprehensive Analysis of Data Quality Issues and Remediation Strategy
VOIS Internship Project

September 28, 2025

Abstract

This comprehensive data profiling report presents a systematic analysis of the NYC Airbnb dataset containing 102,599 listings across 26 feature columns. Through rigorous statistical examination and quality assessment, we uncover critical data integrity issues including widespread missing values, physically impossible data points, and structural inconsistencies that render the raw dataset unreliable for immediate analysis. This report documents these findings with detailed statistical evidence and establishes a clear, actionable 5-step remediation plan to transform this flawed dataset into a trustworthy foundation for market analysis and predictive modeling.

Contents

1 Executive Summary

The Central Challenge

The NYC Airbnb dataset represents a substantial data resource with over 100,000 property listings, yet beneath this apparent richness lies a maze of quality issues that threaten any analytical work. This profiling phase reveals that using this data in its current state would inevitably lead to:

- Skewed statistical results from corrupted numerical values
- Incorrect business conclusions from missing critical information
- Failed model training from illogical data patterns
- Unreliable insights that erode stakeholder trust

1.1 Dataset Overview: The Raw Numbers

Attribute	Value
Total Listings	102,599
Feature Columns	26
Initial Memory Size	95.9 MB
Data Source	NYC_Airbnb.xlsx
Analysis Date	September 28, 2025
Project Phase	Phase 1, Day 1

Table 1: NYC Airbnb Dataset Overview

1.2 Critical Findings Summary

Four Major Data Quality Issues Identified

The profiling process revealed four categories of critical problems:

1. **Significant Missing Data:** Critical columns like `license` (99.9% missing) and `house_rules` (53.4% missing) contain insufficient information for meaningful analysis
2. **Illogical and Invalid Data:** Physically impossible values including negative minimum nights (-1223), impossible availability values (3677 days), and future review dates (year 2058)
3. **Structural Inconsistencies:** Column naming conventions violate best practices with mixed capitalization and spaces (e.g., `host id`, `Construction year`)
4. **Redundant Information:** Columns like `country` and `country code` provide zero unique information for a NYC-focused dataset

2 Initial Data Inspection

2.1 DataFrame Structure and Memory Analysis

DataFrame Information Summary

The `df.info()` output provides our foundational understanding of the dataset's structure:

- **Shape:** 102,599 rows (individual listings) × 26 columns (features)
- **Data Types:** Mixed composition with int64 (identifiers), float64 (numerical measurements), object (text/categorical), and datetime64[ns] (temporal data)
- **Memory Footprint:** Approximately 95.9 MB in initial state
- **Optimization Potential:** Significant memory reduction possible through column removal and data type optimization

2.2 Visual Data Inspection

Head and Tail Analysis

Examining the first and last five rows confirms proper data loading while immediately revealing quality issues:

- **NaN (Not a Number):** Visible throughout for missing numerical and text values
- **NaT (Not a Time):** Appears frequently in date columns indicating temporal data gaps
- **Inconsistent Column Names:** Space-separated names like `host id`, `host name`, and `neighbourhood group` prevent dot notation access

Critical Observation: The inconsistent naming convention is not merely aesthetic—it forces verbose bracket notation (`df['host id']`) instead of clean dot notation (`df.host_id`), increasing code complexity and error probability.

3 Statistical Summary and Anomaly Detection

3.1 Comprehensive Descriptive Statistics

The `.describe()` method reveals the numerical distributions and immediately exposes data integrity failures:

Metric	price	service_fee	minimum_nights	availability_365	last_review
Count	102,352	102,326	102,190	102,151	86,706
Mean	$625.3	$125.0	8.1	141.1	Jun 12, 2019
Std Dev	$331.7	$66.3	30.6	135.4	—
Min	$50	$10	**-1223**	**-10**	Jul 11, 2012
25th % (Q1)	$340	$68	2	3	Oct 28, 2018
50th % (Median)	$624	$125	3	96	Jun 14, 2019
75th % (Q3)	$913	$183	5	269	Jul 5, 2019
Max	$1200	$240	**5645**	**3677**	**Jun 16, 2058**

Table 2: Statistical Summary Revealing Critical Anomalies (Invalid values highlighted in red)

3.2 Detailed Anomaly Analysis

Anomaly 1: Negative Minimum Nights

The Issue:

- Minimum value: -1223 nights
- Physical interpretation: Impossible—a booking policy cannot require negative nights
- Maximum value: 5645 nights (15.5 years)

Implication: These values represent catastrophic data corruption. Any statistical measure (mean, median, standard deviation) calculated from this column is mathematically compromised. Visualizations like box plots or histograms would be dominated by these extreme outliers, completely obscuring the true distribution of legitimate minimum stay requirements.
Action Required: Immediate row removal for all entries with `minimum_nights < 1`.

Anomaly 2: Invalid Availability Range

The Issue:

- Expected range: 0 to 365 days
- Observed minimum: -10 days
- Observed maximum: 3677 days (over 10 years)

Implication: This represents a fundamental breakdown in data validation. The column name explicitly refers to availability within a 365-day window, yet contains values that violate this constraint by orders of magnitude. Any booking availability analysis or occupancy rate calculation would produce nonsensical results.
Action Required: Apply strict filter: `0 <= availability_365 <= 365`.

Anomaly 3: Future Review Dates

The Issue:

- Latest valid date: September 28, 2025 (report date)
- Observed maximum: June 16, 2058 (33 years in the future)

Implication: Temporal data integrity is completely broken. Time-series analysis, recency features (e.g., "days since last review"), and any chronological modeling would fail catastrophically. These impossible dates suggest systematic data entry errors or database corruption.
Action Required: Remove all entries where `last_review > current_date`.

3.3 Price and Service Fee Analysis

Pricing Statistics

While price and service fee columns do not contain illogical values, their distributions provide valuable market insights:

- **Price Distribution:** Mean of $625.3 with standard deviation of $331.7 indicates substantial variability in listing costs
- **Median Price:** $624 suggests a roughly symmetric distribution (mean median)
- **Interquartile Range:** Q1=$340, Q3=$913 shows that 50% of listings fall within a $573 price range
- **Service Fee Pattern:** Mean of $125 represents approximately 20% of mean price, following typical platform commission structures

4 Missing Data Analysis

4.1 Missing Value Distribution

The `.isnull().sum()` analysis quantifies data completeness across all columns:

Column	Missing Count	Missing %
license	102,597	**99.9%**
house_rules	54,843	**53.4%**
last_review	15,893	**15.5%**
reviews_per_month	15,879	**15.5%**
Construction_year	532	0.5%
service_fee	273	0.3%
price	247	0.2%

Table 3: Missing Value Summary (Critical thresholds highlighted)

4.2 Missing Data Categorization

Three Categories of Missingness

Category 1: High Volume Missing (Column Removal Candidates)

- `license`: With 99.9% missing values, this column is effectively empty
- `house_rules`: Over 53% missing—too sparse for reliable text analysis
- **Decision:** Drop these columns entirely

Category 2: Moderate Volume Missing (Strategic Imputation)

- `last_review`: 15,893 missing values
- `reviews_per_month`: 15,879 missing values
- **Key Insight:** Nearly identical missing counts suggests these are not randomly missing—they're logically missing because listings have `number_of_reviews = 0`
- **Decision:** Impute `reviews_per_month = 0` where `number_of_reviews = 0`

Category 3: Low Volume Missing (Row Removal)

- All other columns: 8 to 532 missing values (¡ 1%)
- **Decision:** Drop rows containing these missing values—acceptable data loss for complete dataset

4.3 Missing Data Pattern Analysis

Understanding Missingness Mechanisms

The pattern of missing data reveals important insights about data collection:

1. **Structural Missingness (MNAR - Missing Not At Random):**
   - The `license` column's near-complete absence suggests regulatory compliance data was not systematically collected
   - This is likely a post-hoc addition to the data schema without backfilling historical records
2. **Logical Missingness (MAR - Missing At Random):**
   - Review-related fields are missing precisely when listings have zero reviews
   - This is not true missingness but rather the absence of applicable data
   - Can be safely imputed with zeros
3. **Random Missingness:**
   - Small percentages (¡ 1%) across remaining columns suggest random data entry failures
   - No systematic pattern—safe to remove via listwise deletion

5 Structural Issues: Column Naming Analysis

The Column Naming Problem

The dataset suffers from inconsistent naming conventions that violate Python best practices:

Current Problems:

- Mixed capitalization: `host id`, `Construction year`
- Embedded spaces: `neighbourhood group`, `last review`
- Inconsistent separators: Some use spaces, others use underscores

Code Impact:

```
# Current state forces verbose bracket notation
df['host id']  # Required
df['neighbourhood group']  # Required

# Desired state enables clean dot notation
df.host_id  # Fails with current naming
df.neighbourhood_group  # Fails with current naming
```

Standardization Example:

`host id`	→	`host_id`
`neighbourhood group`	→	`neighbourhood_group`
`Construction year`	→	`construction_year`

6 Next Steps: Day 2 Implementation

Action Items

Day 2 Workflow:

1. Execute the complete cleaning pipeline script
2. Generate before/after comparison visualizations
3. Document all removed records for audit trail
4. Create data quality report with:
   - Final row/column counts
   - Memory usage comparison
   - Distribution comparisons for key features
   - Summary of all applied transformations
5. Validate cleaned data through:
   - Statistical sanity checks
   - Visual inspection of distributions
   - Sample manual review of records
6. Proceed to exploratory data analysis (EDA) phase

7 Conclusion

Key Takeaways

This profiling phase has revealed that the NYC Airbnb dataset, while substantial in size, requires significant remediation before analysis. The systematic identification of:

- 99.9% missing data in critical columns
- Physically impossible values (negative nights, future dates)
- Over 175,000 individual missing values
- Structural inconsistencies in naming conventions

demonstrates the critical importance of thorough data profiling as the first step in any analytical project. Our comprehensive 5-step cleaning plan provides a clear roadmap to transform this flawed dataset into a reliable foundation for market analysis, predictive modeling, and business intelligence.
The investment in rigorous data quality enforcement will pay dividends throughout the project lifecycle by ensuring that all downstream analysis, visualizations, and models are built on a trustworthy data foundation.