Appendix: Project Challenges

Candace Grant

2025-12-15

Project Challenges & Solutions

NYC 311 Service Request Analysis

Challenge 1: Shiny App Development — From Static Analysis to Interactive Decision Tool

The most significant challenge I encountered was learning how to build, update, and modify a Shiny application from scratch. As a data science student with a background in statistical analysis and R programming, I found that Shiny development requires an entirely different mental model—one rooted in **reactive programming** rather than sequential script execution.

Specific Obstacles Encountered

Obstacle	Description	Resolution
Reactive Dependencies	Understanding when and why UI elements update based on user input required debugging invisible data flows	Created detailed flowcharts mapping input → reactive → output relationships before coding
UI/UX Design Decisions	Balancing information density with visual clarity across multiple tabs	Iteratively removed clutter (e.g., eliminated redundant charts when filters made them unnecessary)
Layout Responsiveness	Dashboard elements rendering differently across screen sizes	Used `fluidRow()` and `box()` with explicit width parameters
Filter Synchronization	Ensuring all visualizations updated consistently when users changed borough, complaint type, or date filters	Centralized filtering logic in a single `reactive()` expression that all outputs reference

The process felt analogous to developing a **User Experience (UX) application**—I had to consider not just *what* data to display, but *how* users would interact with it, *what questions* they might ask, and *which visual pathways* would guide them to insights. This required multiple rounds of user-centered iteration: building a feature, testing it from a stakeholder's perspective, identifying friction points, and refining the interface.

Key Learning

Shiny development taught me that data science deliverables extend beyond statistical accuracy—they must be *usable*, *intuitive*, and *actionable* for non-technical audiences. This skill directly translates to industry roles where communicating findings to stakeholders is as critical as generating them.

Challenge 2: Extracting Meaningful Insight from API-Limited Data

My second major challenge was developing meaningful, generalizable insights from data constrained by API limitations. I integrated **three distinct data sources** in this project:

Data Source	API Limitation	Records Retrieved	Time Coverage
NYC Open Data (311)	100,000 record cap per query	~80,000 records	~30-60 days
Open-Meteo Weather	Date range limited to available historical data	Daily aggregates	Matched to 311 range
U.S. Census (ACS)	5-year estimates only	5 borough records	2022 snapshot

The Core Problem

NYC's 311 system processes over **80,000 service requests per day** during peak periods. My 80,000-record dataset therefore represented only **1-3 days of citywide activity** rather than the longitudinal trends I initially envisioned. This fundamentally limited my ability to detect seasonal patterns, year-over-year changes, or long-term geographic shifts.

Strategies I Employed to Maximize Insight

1. **Shifted from temporal to cross-sectional analysis:** Rather than asking "How have complaints changed over time?", I pivoted to "How do complaints differ across boroughs *right now*?"

2. **Leveraged weather data for short-term correlations:** Even with limited days, temperature variation within the dataset allowed meaningful heating/noise correlations.

3. **Normalized by population:** Converting raw counts to "complaints per 1,000 residents" enabled fair borough comparisons despite different population sizes.

4. **Focused on high-frequency complaint types:** By filtering to the top 20 complaint categories, I ensured sufficient sample sizes for statistical testing.

5. **Iterative visualization refinement:** Through trial and error, I discovered that:

   - Scatter plots required more data points than I had → switched to binned bar charts
   - Daily aggregations were sparse → temperature range groupings revealed clearer patterns
   - Borough-level demographics created only 5 data points → supplemented with per-capita calculations

Key Learning

Real-world data science rarely involves clean, unlimited datasets. The ability to **adapt analytical strategies to data constraints** while still delivering actionable insights is a critical professional skill. This project forced me to be creative within limitations rather than waiting for ideal conditions.

Challenge 3: Translating Statistical Findings into Actionable Policy Insights

The third challenge—and perhaps the most intellectually demanding—was transforming raw statistical outputs into insights that could meaningfully inform municipal decision-making. It is one thing to calculate a correlation coefficient or generate a Random Forest variable importance plot; it is another to articulate *why it matters* and *what should be done about it*.

The Gap Between Analysis and Action

What the Data Showed	Initial Interpretation	Deeper Policy Question
Heating complaints correlate negatively with temperature (r = -0.67)	"People complain more when it's cold"	*Which buildings repeatedly generate complaints? Should inspections be proactive before winter?*
Bronx has the longest median response time	"Bronx is slower"	*Is this due to complaint volume, staffing levels, or complaint complexity? What resource allocation would equalize service?*
Renter-heavy boroughs have more housing complaints per capita	"Renters complain more"	*Does this reflect worse housing conditions, better tenant awareness, or landlord neglect? What tenant protections are needed?*
Complaint type is the strongest predictor in the Random Forest model	"Type matters most"	*Which specific complaint types drive delays? Can workflow redesign reduce resolution time?*

Key Learning

Data science creates value not through technical sophistication alone, but through **translating complexity into clarity**. A Random Forest model with $R^2 = 0.71$ means nothing to a city council member—but "we can predict which complaints will take longest and staff accordingly" is actionable. This project taught me to always ask: *So what? Who cares? What should they do differently?*

Reflection: Growth Through Challenge

Each challenge contributed to my development as a data scientist:

Challenge	Skill Developed	Professional Application
Shiny Development	Interactive data product design	Building stakeholder dashboards
API Limitations	Adaptive analytical thinking	Working with imperfect real-world data
Insight Translation	Policy communication	Bridging technical and executive audiences

The final product—a comprehensive analysis with statistical rigor, predictive modeling, and an interactive decision-support tool—represents not just technical achievement but genuine intellectual growth. I am proud to submit this project as evidence of my readiness to contribute meaningfully to data-driven organizations.

Candace Grant | M.S. Data Science | CUNY School of Professional Studies | December 2025