

Data Automation & Workflow Optimization

[City of Boston] | [Department of Innovation Technology]

Anuraag Pandhi, Data Fellow | *Columbia University, Computer Science & Operations Research: Financial Engineering*

Keywords:

Urban Planning, Workflows, Processing, Time Series, Modelling

Summary:

This project focused on improving urban planning efficiency for the City of Boston through the automation of LiDAR data processing and administrative workflows. By leveraging Python scripts and GIS software, the project automated key processes such as data cleaning, filtering, and mapping, resulting in reduced processing time and enhanced accuracy. Additionally, Anuraag developed automated workflows using Jupyter Notebooks to streamline tasks like user account deletion, email table merging, and credit tracking. The project significantly reduced manual effort, minimized errors, and created scalable solutions with the potential for broader application across city departments.

Data Automation & Workflow Optimization

Department of Innovation Technology

City of Boston, MA

Guided by Jeffrey Kaplan & Emma Curran

coding it forward >

CITY of **BOSTON**

ANURAAG PANDHI

Columbia University
CS & OR:FE

Section 1

LiDAR Data Automation

3D Building Generation, Polygonization, Roof Segmenting



LiDAR Data and Its Importance

- **What is LiDAR?**
 - LiDAR (Light Detection and Ranging)
 - Uses laser pulses to measure distances
 - Creates detailed 3D maps of urban landscapes
 - Key tool for urban planning, infrastructure management
- **Why Automate?**
 - Manual processing is time-consuming and prone to errors
 - Automation improves accuracy, speed, and scalability
 - Enables quicker responses to urban planning needs

A View of Boston University



Workflow Automation Process

- **Data Acquisition**

- LiDAR data collected from aerial surveys and ground-based platforms.
- Imported into a central dataset for processing.

- **Data Processing**

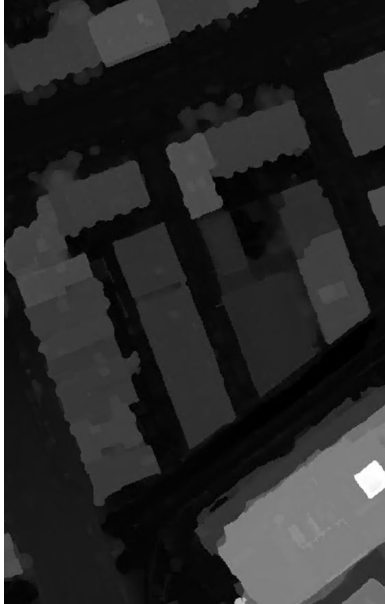
- Automated data cleaning to remove noise and irrelevant data points.
- Filtering and segmentation to focus on relevant structures and features.
- Generation of digital elevation models (DEMs) and 3D representations.

- **Automation Tools**

- Python scripts for processing and data manipulation.
- GIS software (e.g., ArcGIS) for spatial analysis and visualization.

The Process of Roof Segmenting

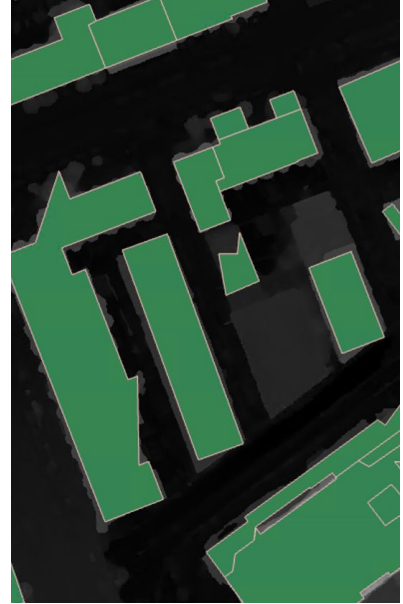
Elevation Layers



Polygonization



Segmentation



Key Achievements

- **Efficiency Gains**

- Method Calls
 - Chaining
 - Optimizing
- Streamlining
 - Detection
 - Mapping
- Drastic Reduction of Manual Work

- **Scalability**

- Size
 - Large Datasets
 - Multiple Locations
- Versatile
 - Low to High Density LiDAR
 - Residential vs Urban Areas
- Opportunities for Reproducibility

Challenges and Solutions

- **Technical Challenges**

- Handling large volumes of data with varying quality
- Integrating different software tools and platforms
- Ensuring compatibility with existing city planning systems

- **Overcoming Obstacles**

- Implemented robust data validation and error-checking routines
- Developed custom scripts to bridge gaps between software tools
- Navigated through a significant amount of outdated documentation

Impact and Future Applications

- **Current Impact**

- City of Boston now has a more efficient process for urban planning
- Faster turnaround times for project approvals and infrastructure updates
- Enhanced ability to respond to environmental and infrastructure changes

- **Future Applications**

- Expand automation to other areas like traffic management
- Explore integration with AI for predictive urban planning

Section 2

Workflow Optimization

User Account Management, Email Communications, Credit Tracking



Workflow Automation Tasks

- **User Account Deletion**

- Automated the process of identifying and deleting inactive user accounts
- Ensured compliance with data retention policies

- **Joining Email Tables**

- Automated merging of email tables for consistent communication records
- Improved data integrity across different departments

- **Credit Tracking**

- Automated credit allocation and tracking for various city programs
- Provided real-time updates and transparency

A Complex Bugfix in User Deletion

```
def content(user):  
    icount = 0  
    for item in user.items():  
        icount += 1  
    try:  
        for f in user.folders:  
            for f_item in user.items(folder=f):  
                icount += 1  
        print(f"{user.username} has {icount} items")  
    except Exception as e:  
        print(f"{user.username} has {icount} items")  
    return icount
```


Impact and Future Applications

- **Current Impact**

- Departments have access to more streamlined operations
- 650+ users deleted, 3 robust time-series models for visualization
- Feasibility for newsletter or organizational communications

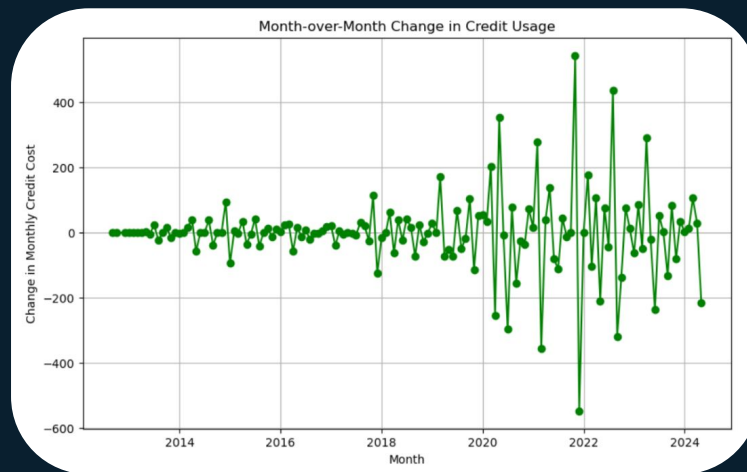
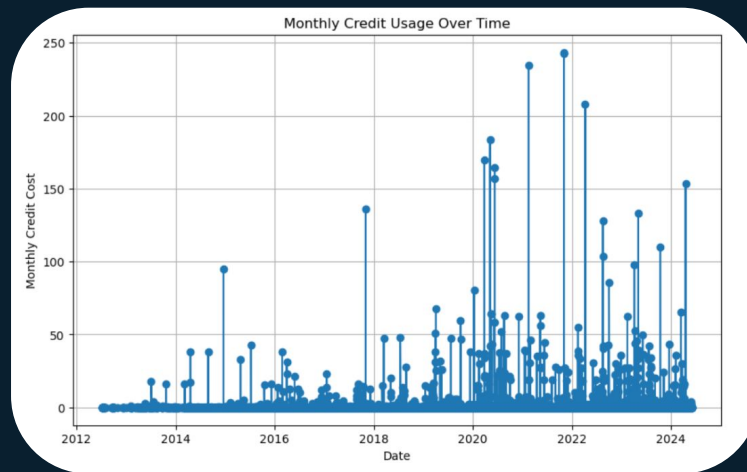
- **Future Applications**

- Expand automation to other routine tasks across city departments
- Potential for a centralized dashboard for monitoring all automated workflows



12 Years of Credit Utilization

A Quick Look



THANK YOU!

Special thanks to Jeff & Emma!

