# COVID Violations and Clusters

ANALYSIS

01

## VIOLATIONS

A Quick Glance at the Davidson County Violations Data

02

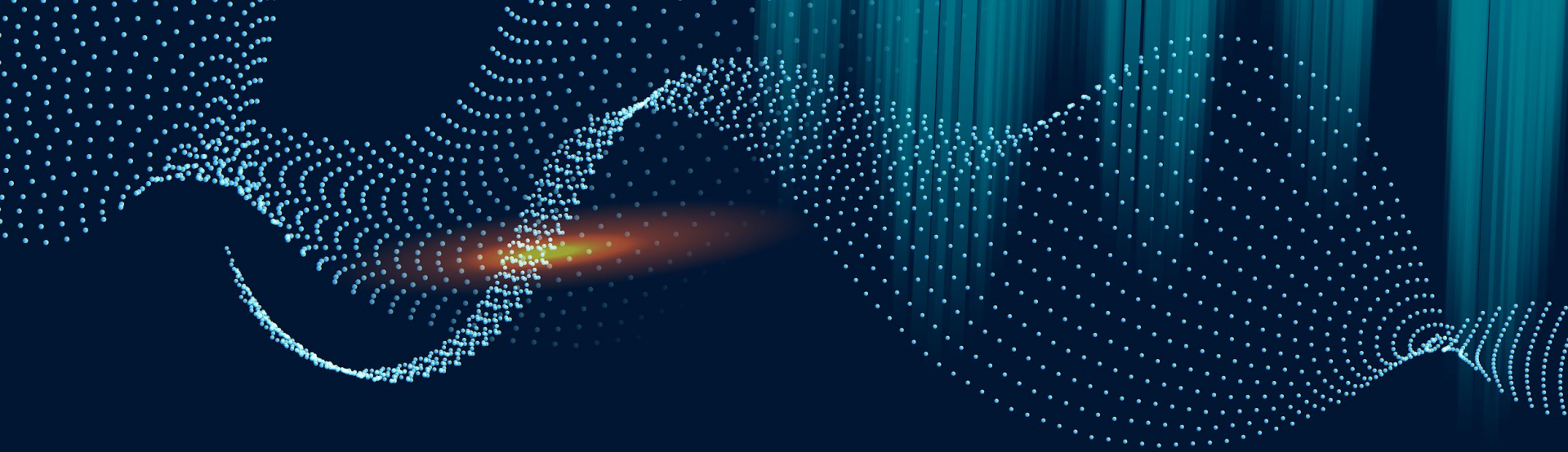
## CASES

A look at COVID cases in Davidson County and trying to align them with Violations

03

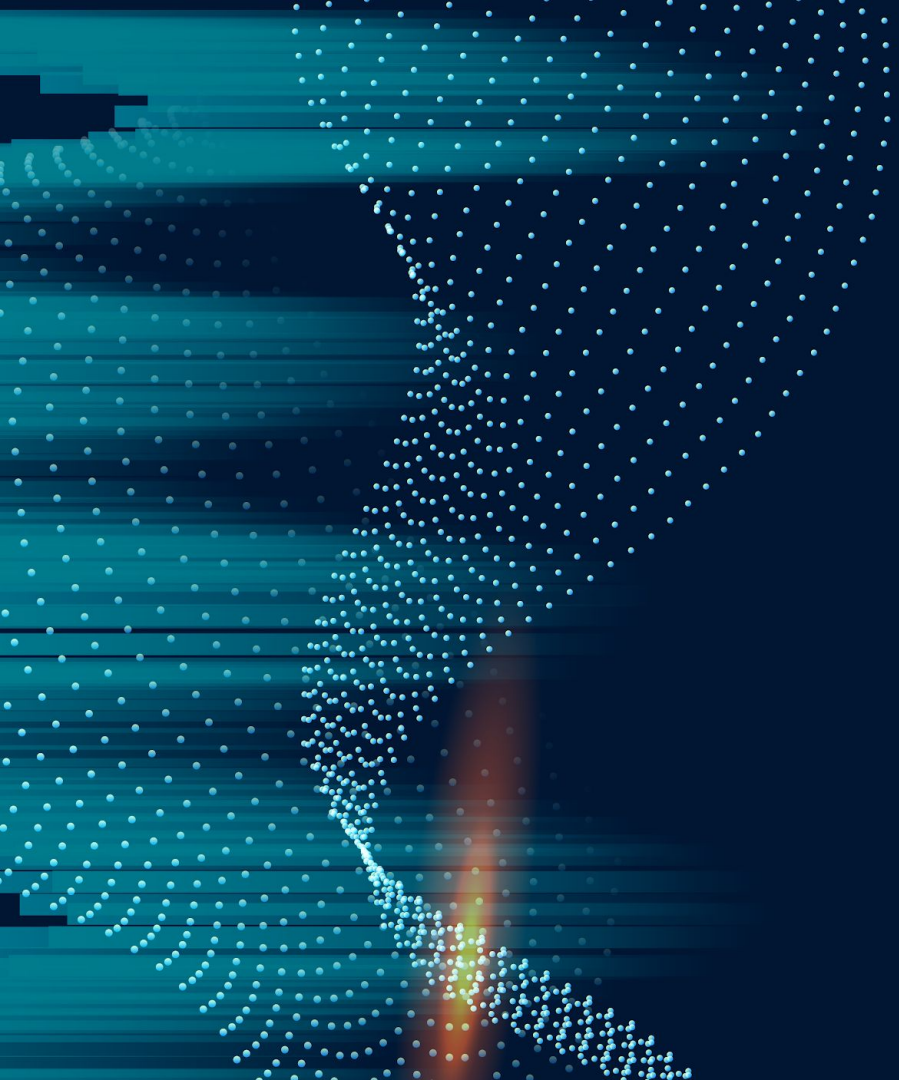
## CLUSTERS

Investigating COVID Clusters and trying to understand them in relationship to violations and clusters.



# 01 VIOLATIONS

A Quick Glance at the Davidson County Violations Data



**334**

Average per week

**754**

Max in a week (excludes week 1)

**102**

Fewest number of violations

# Interactive Visualizations

## Weekly Violations

A bar chart to look at violation reporting trends

## Violations Map

Marker clustered map to look at reported violations

## Violations Timeline

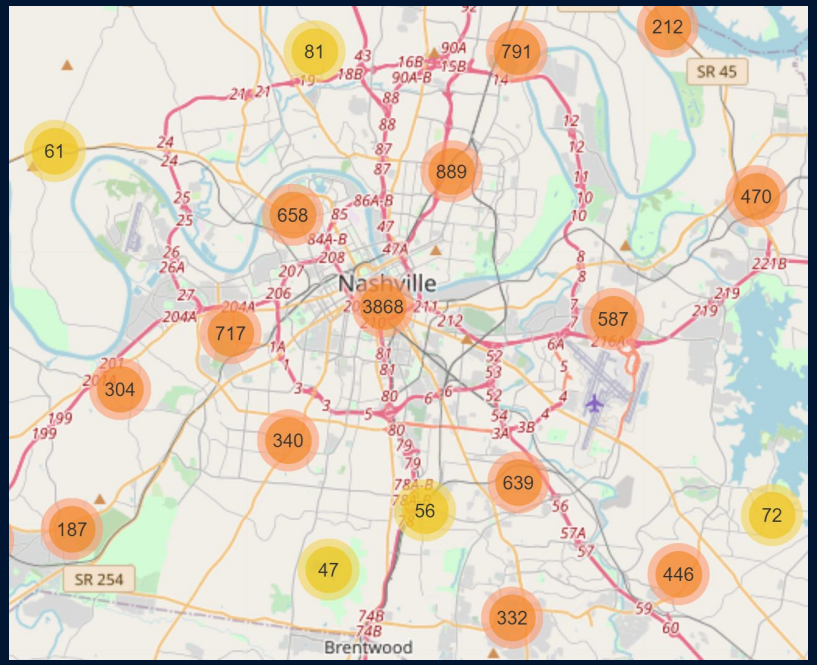
A daily look at plotted violations

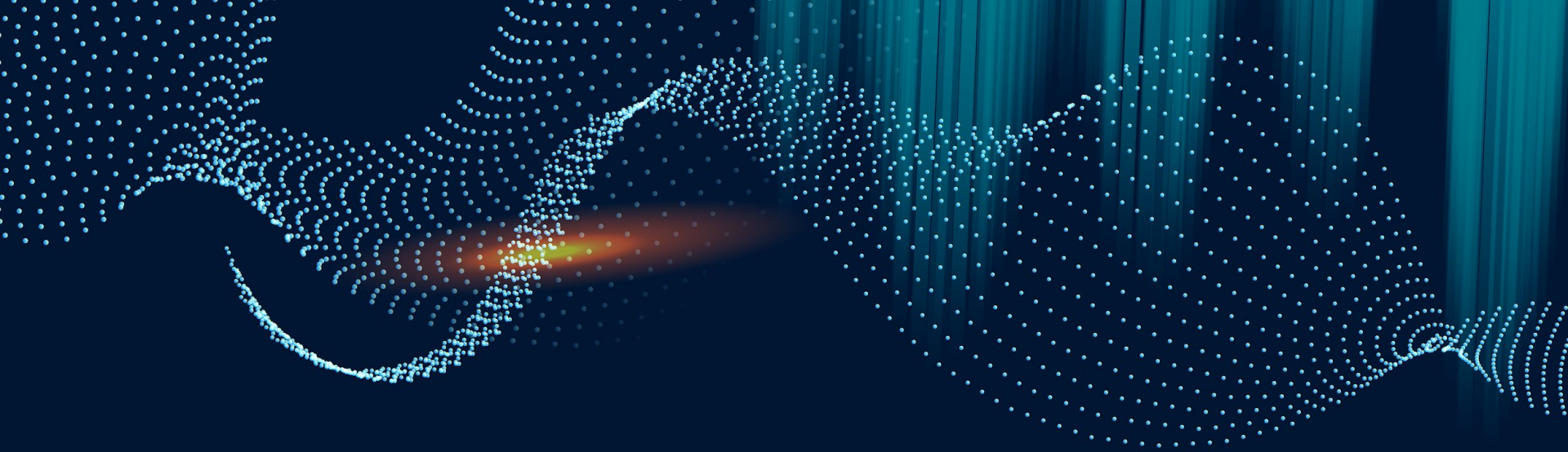
## Violation Locations

What places did the violations happen

# Visualizing all the tags

# Visualizing Business Types by Google Places Tags

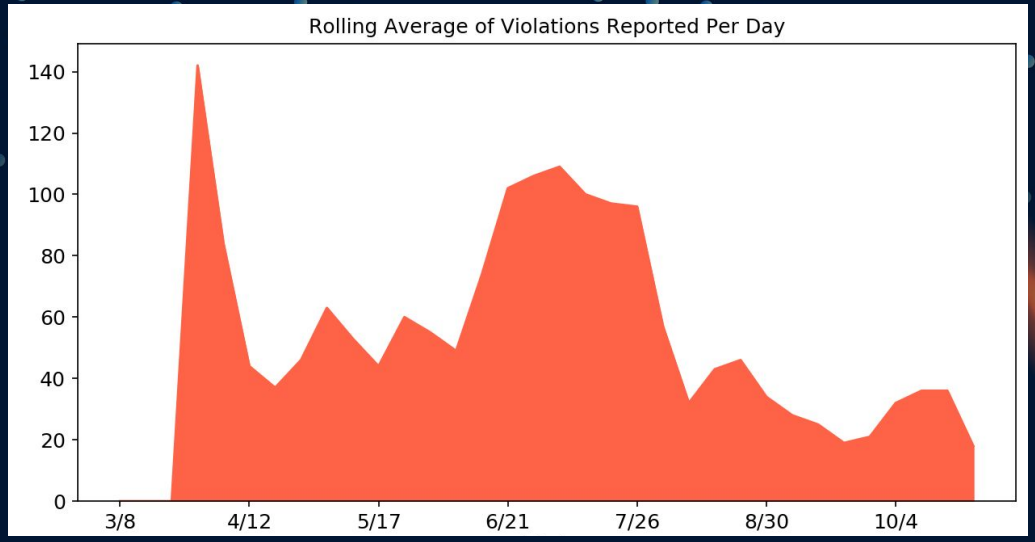


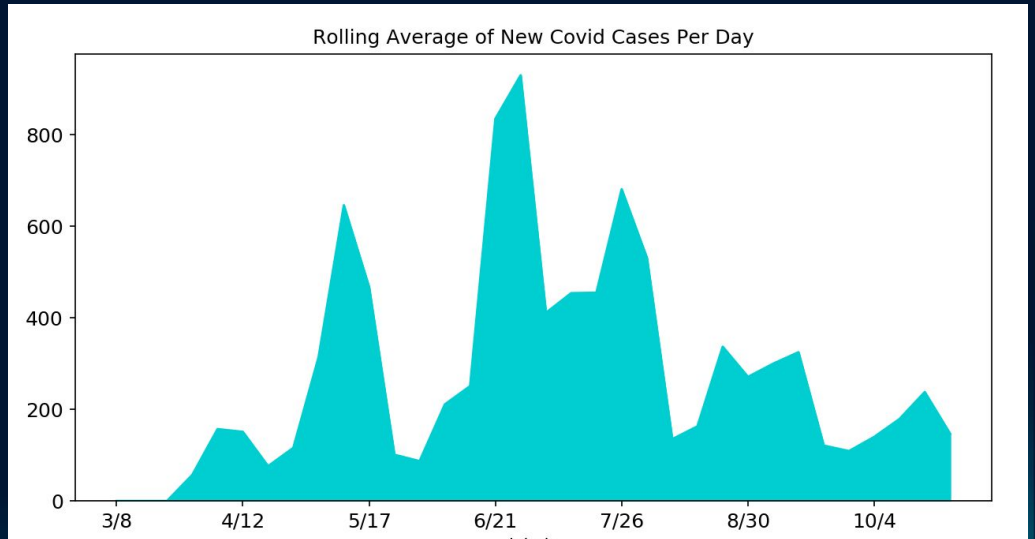


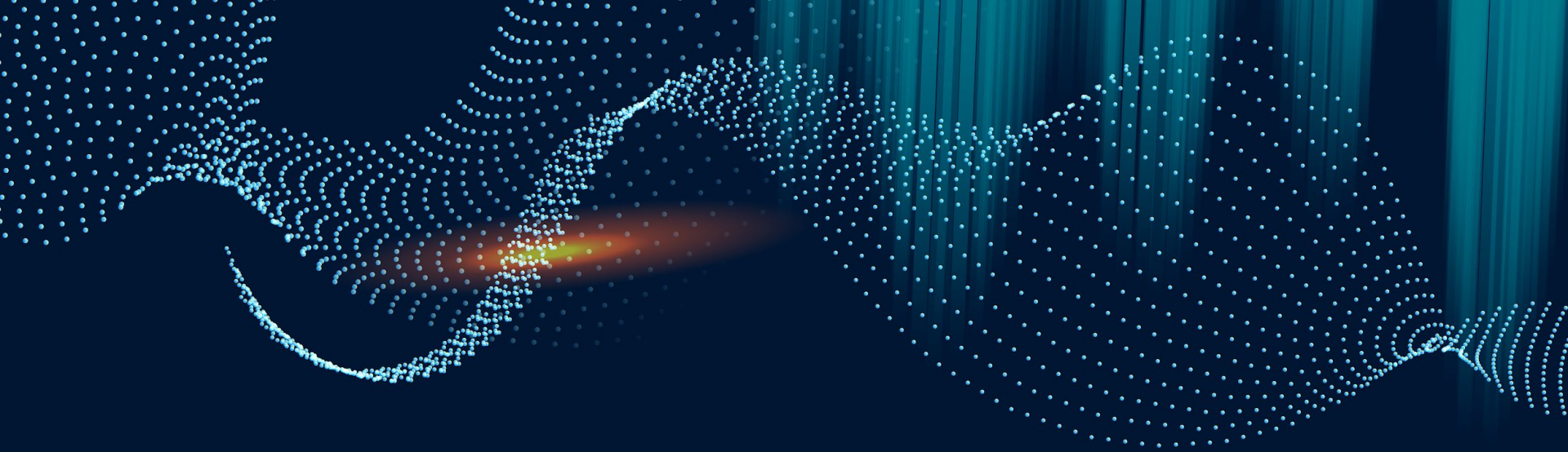
# 02 CASES

A look at COVID cases in Davidson County and trying to align them with Violations

# Nashville Covid Violations and Cases over Time



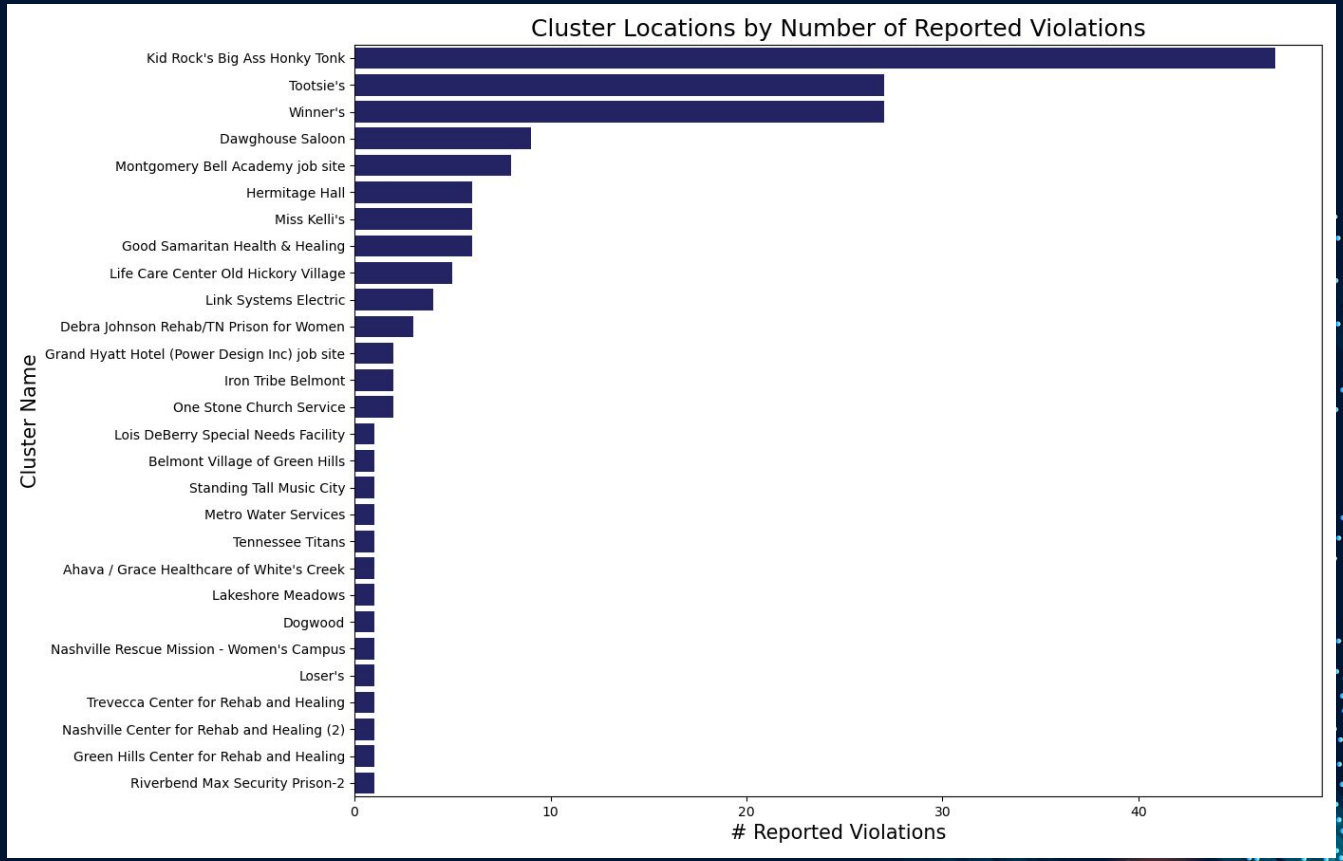





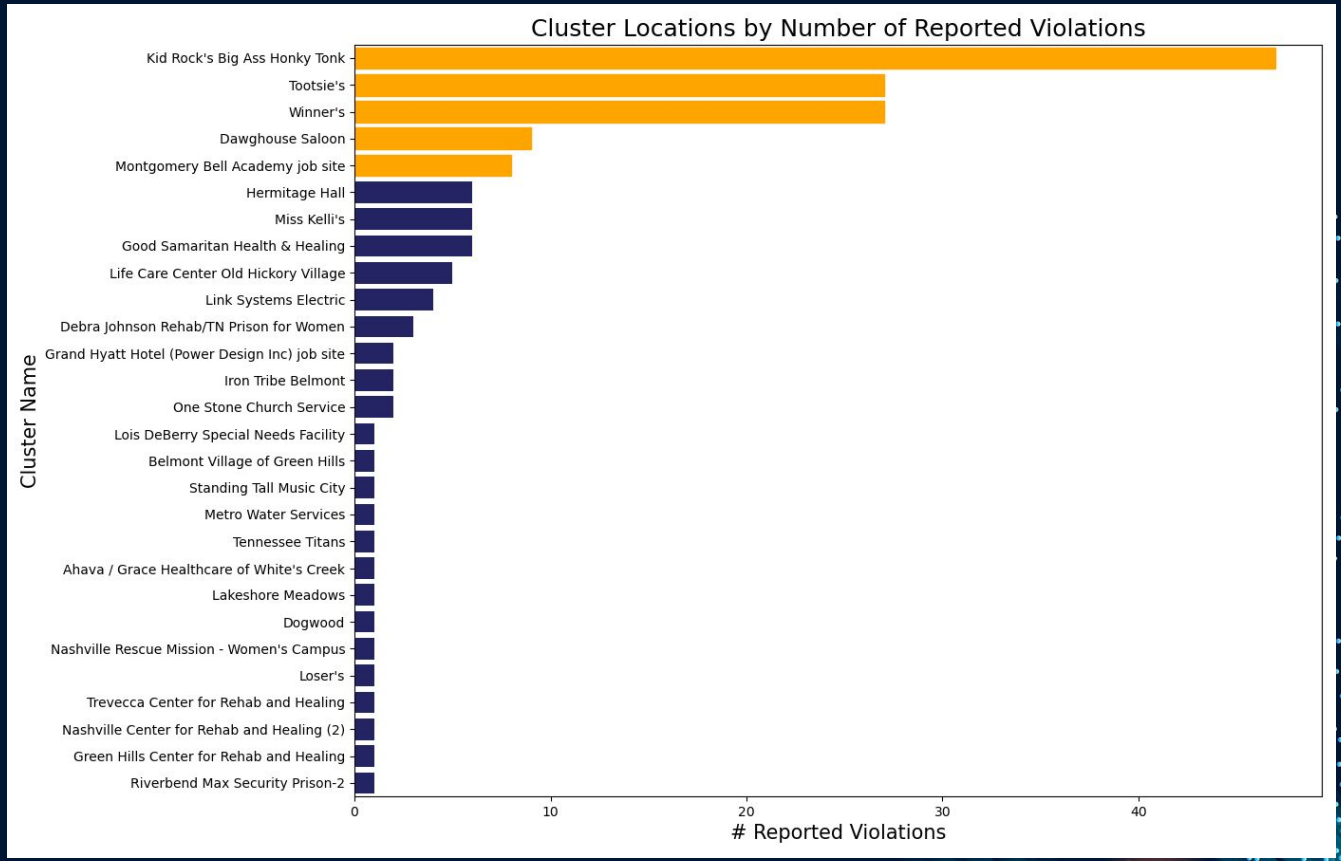
# 03 CLUSTERS

Investigating COVID Clusters and trying to understand them in relationship to violations and clusters.

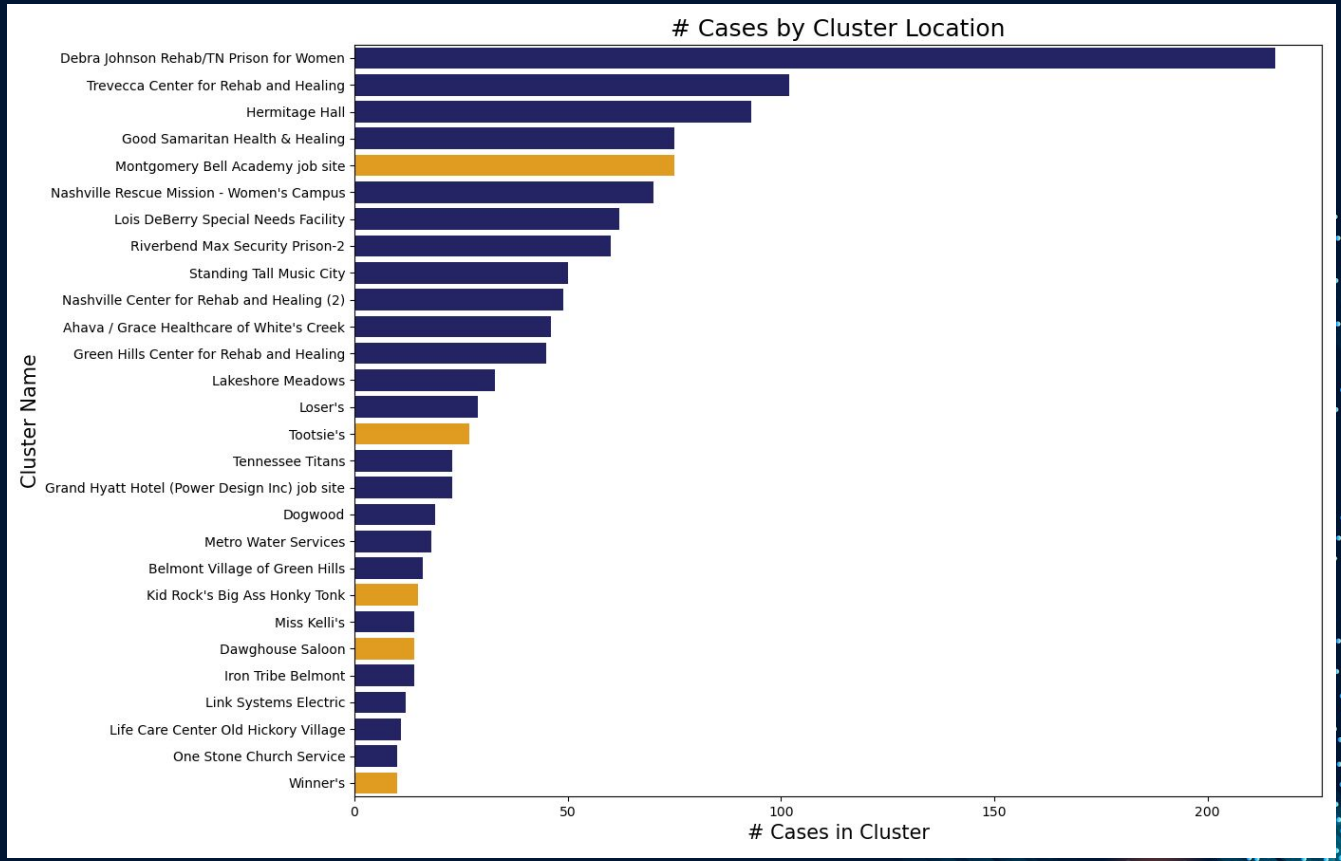
# Clusters with Highest Number of Violations



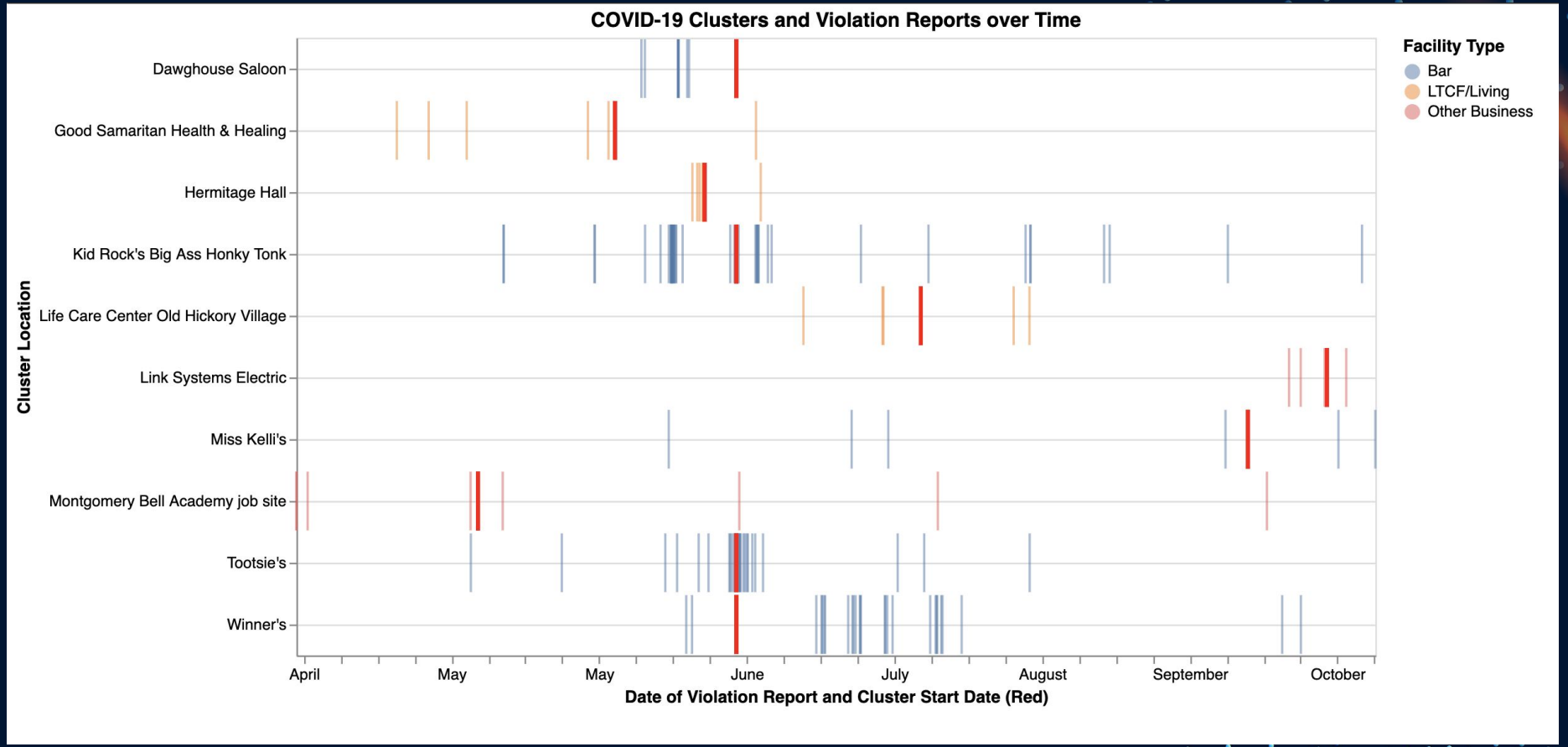
# Clusters with Highest Number of Violations



# Clusters with Highest Number of Cases



# Violations in Relation to Cluster Start Date



```
def fuzzy_merge(df_1, df_2, key1, key2, threshold=95, limit=1):
    s = df_2[key2].tolist()

    m = df_1[key1].apply(
        lambda x: process.extract(x, s, limit=limit))

    df_1['matches'] = m

    m2 = df_1['matches'].apply(
        lambda x: ', '.join(
            [i[0] for i in x if i[1] >= threshold]))

    df_1['matches'] = m2

    return df_1
```

```
hub_covid_clean = hub_covid[hub_covid['Address'].notna()]

matched_places_df[['orig_address_addr','orig_address_city']] =
    matched_places_df.orig_address.str.split(',',expand=True)

matched_places_df_dedupe =
    matched_places_df.drop_duplicates(subset='orig_address_addr')

places_violations_fuzzy = fuzzy_merge(hub_covid_clean,
    matched_places_df_dedupe, 'Address', 'orig_address_addr',
    limit=1)
```