Steps To reproduce the results

**DATA DEPENDENCIES**

**Monthly Data**

You’ll be downloading the necessary data for the month from the MA court website. The data folder of this file already contains some downloaded files. For the month of the data you’re adding, you will need to create a new folder within this existing file structure (see below). Make sure the names of these folders have NO SPACES:

data

|-> 2017

|-> Jan

|-> Feb

|-> March

…

|-> 2016

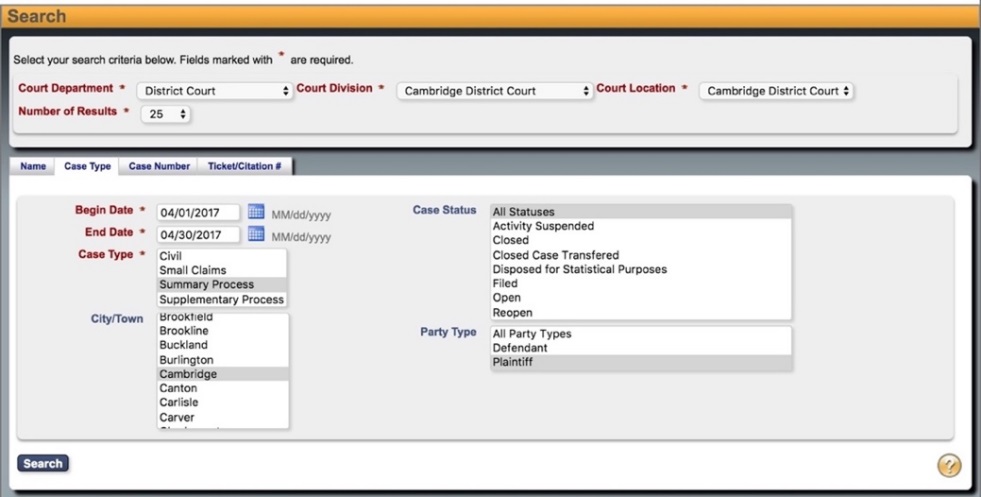
|-> Jan

|-> Feb

…

Click this link to access the MA State Court website and enter the captcha to gain access: <https://www.masscourts.org/eservices/home.page.2>

To search the website, you’ll want to use the following options for Cambridge Courts:

Court Department -> District Court

Court Division -> Cambridge District Court

Court Location-> Cambridge District Court

Choose the “Case Type Tab”:

Case Type -> Summary Process

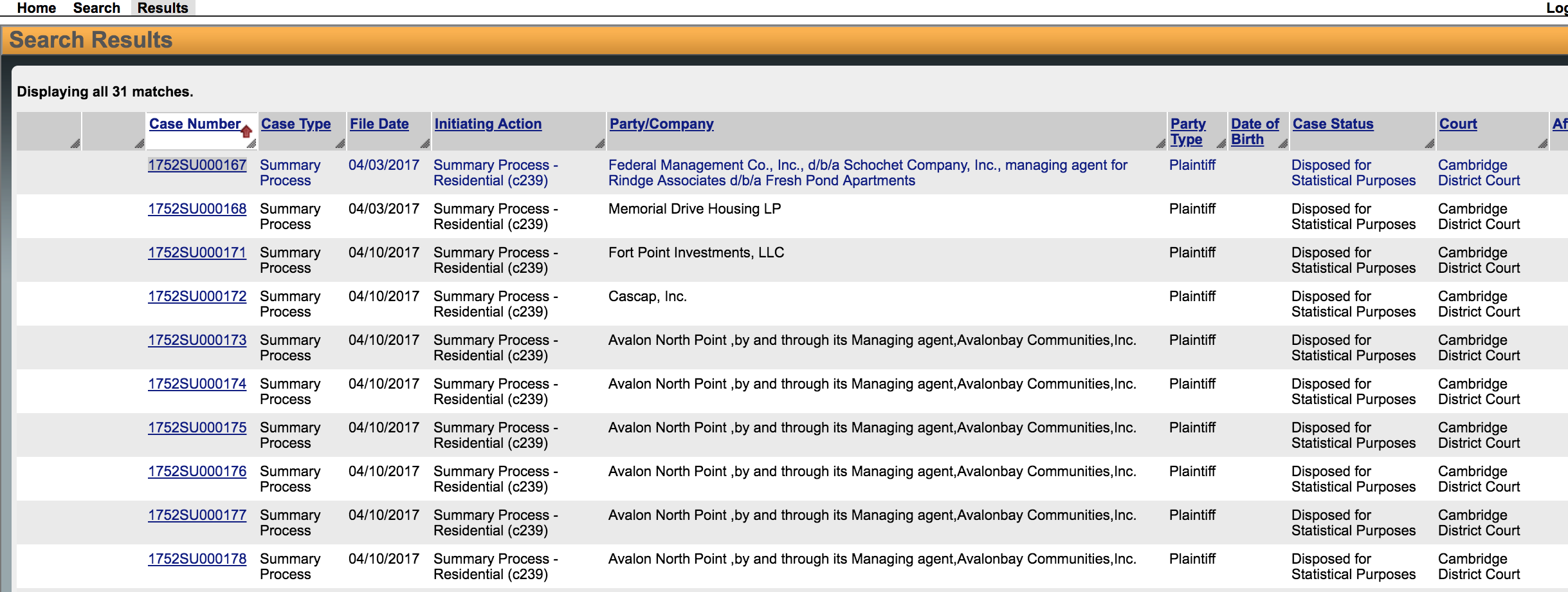
City/Town -> Cambridge

Case Status -> All Statuses

Party Type-> Plaintiff

Be sure to specify the begin and end date of the month you are adding. Only one month of data can be searched at a time.

Once you click Search, the results page should look like this:



Click on each case number to open the file and save each individual case to the month folder you’ve created (File->Save Page As, or Ctrl + S), choosing the format “Webpage, HTML only”. The file name doesn’t matter so long as it is unique to the folder and doesn’t overwrite an existing file. If you accidentally save with the other file format, simply delete the folder it creates and resave with the correct format.

**Running Scrapper to build initial csv from court dockets**

Run scrapper.py

It will create initial data file in the form of csv

/csv/csvdata.csv



**Clean up and adding additional information**

1. Cleanup and removing duplicate

Drop duplicate case number by using pandas

df = df.drop\_duplicates(subset=’Case Number’, keep="first")

Change the data type in judgement total and execution total to float by

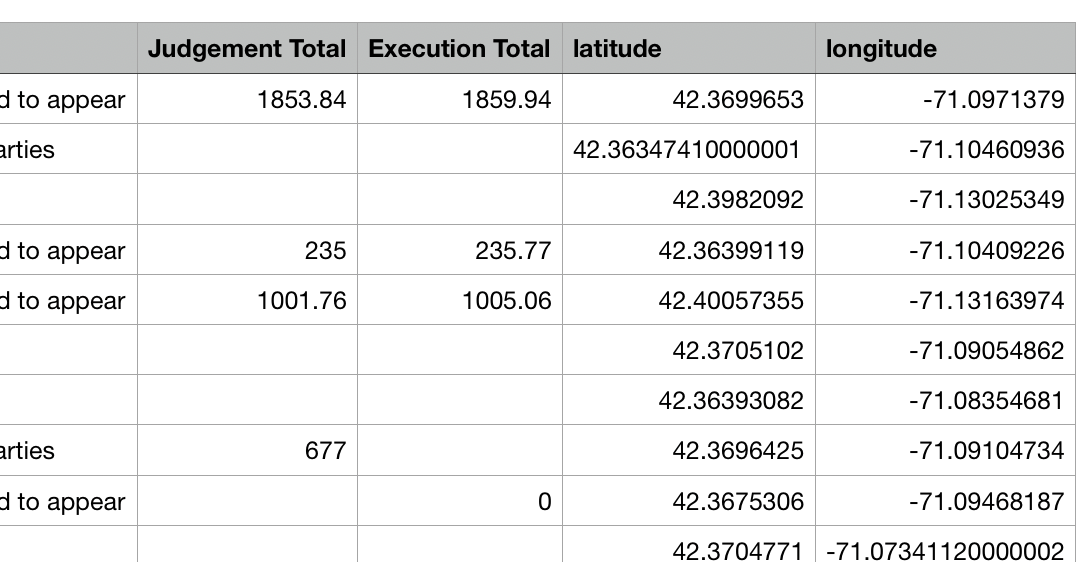
* Replace coma(,) with ‘’
* Change data type from string to Float

1. Adding Longitude & Longitude

From GeoMaps.py, call function addGeo(data)

The function will find the longitude and longitude based on address. It will append new 2 columns (longitude and latitude)

For address where the coordinates cannot be located, use google maps to find the coordinate



1. Adding district

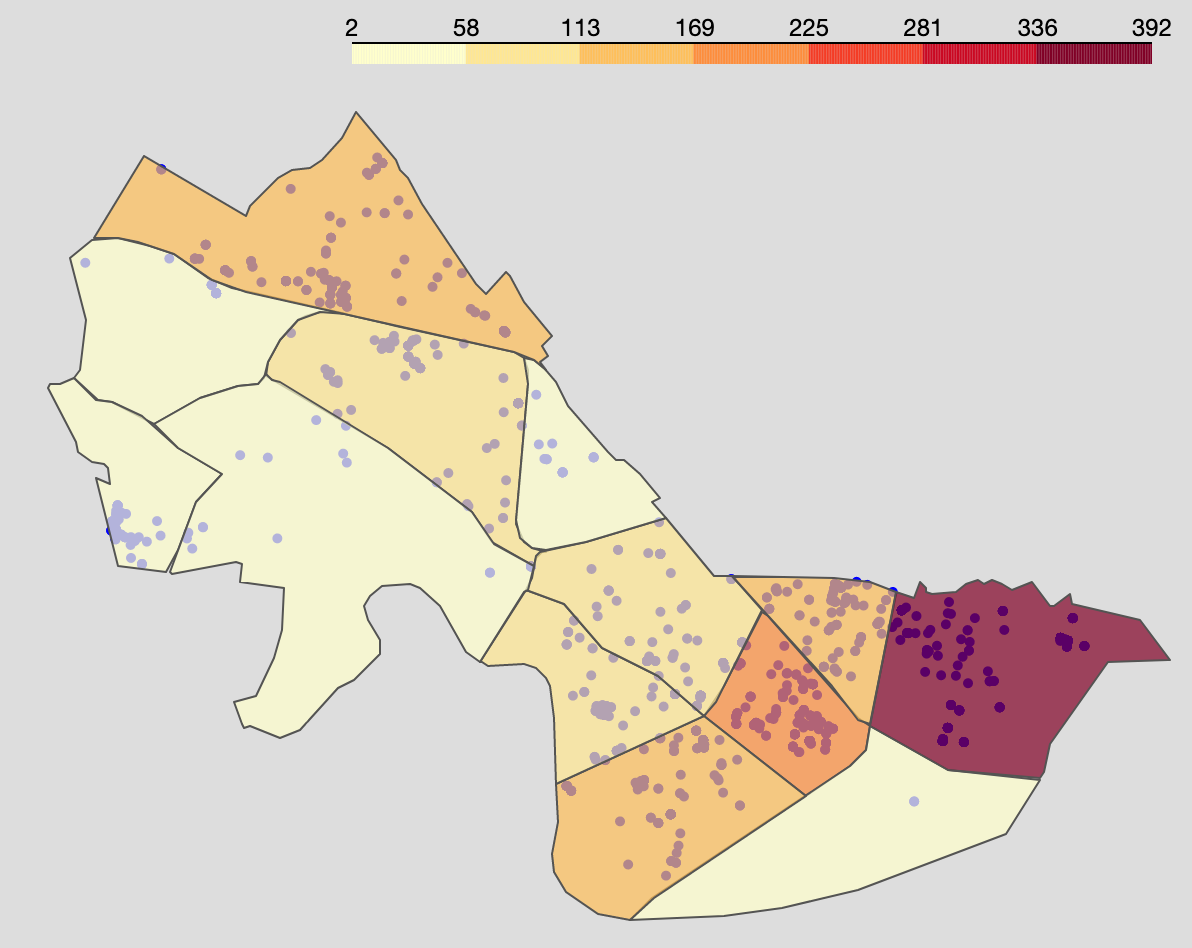
By combining with data from Neighbourhood polygon

From GeoMaps.py run addDistrict(data, districts)



1. Creating Map of Cambridge with number of cases

From GeoMaps.py run createMapPlot(data, district)



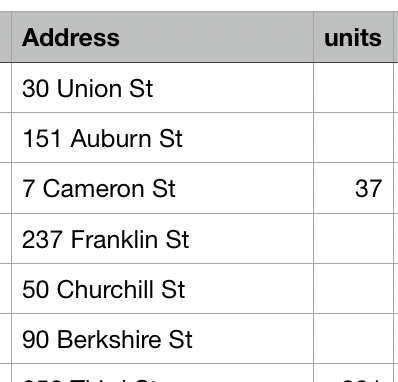
1. Adding unit

The unit is the number of units that is in the building.

By using the data we have from previous step together with housing stat.csv data

run findUnit.py

It will add another column Address (standardize address )and Units



1. Adding Median Income

BY using previous data (csv file and American\_Community\_Survey\_2013\_-\_17\_Estimates\_by\_Neighborhood\_\_Median\_Income) run get\_median\_income.py

**./csv/cleanData.csv is the final data we get from process 1 to 6**

1. To do data science analysis (clustering)

Run clustering.py

1. To run data analysis by using Power BI

From the analysis folder, open Eviction.pbix in power BI desktop.

If there is data error, import the latest/cleanest csv file from process 1 to 6 to as the data source

You might need to update the query in each report to reflect the data

The result of Power BI report is in analysis/data.pdf

