Shelby Vanhooser

Data Science

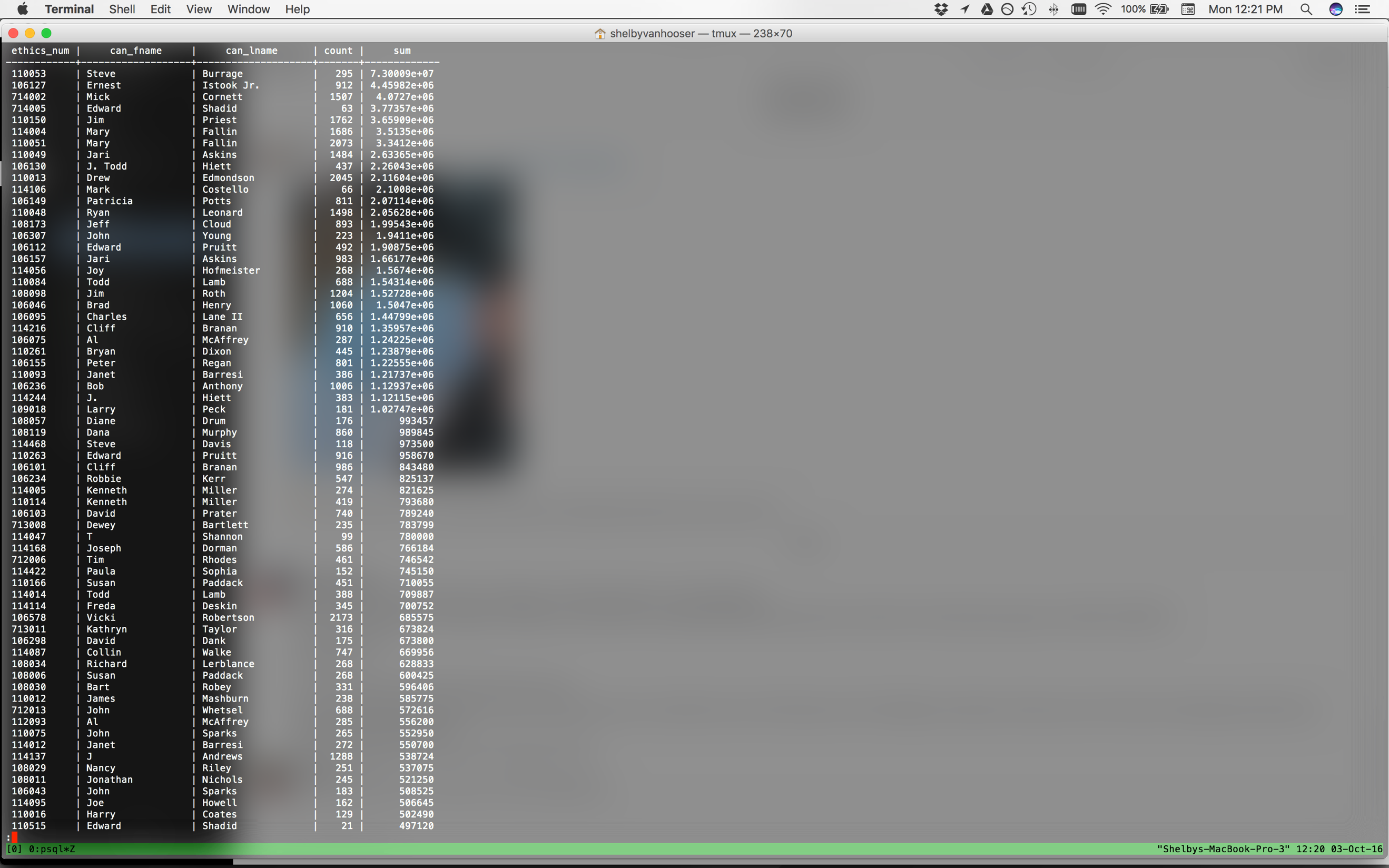
4 October 2016

Summary of Research

Take a deep breath, this is going to be a lot.

Let us begin by examining the work I initially had to complete to join all political transactions together and group them appropriately by candidate that received each. This was accomplished by looking at the brute transaction table that listed the origin identification number, destination report number, and transaction amount for all contributions. These had to be summed together by report number and origin identification number such that total contributions to each candidate by and individual could be understood. Then, these summations had to be linked against a registry of all individuals with the origin identification numbers so that we could read the addresses of each donor as well as his or her first/last name. Then, now that the names, addresses, report number, and totals were noted in a large set of rows, these rows *then* had to be joined on a map of report numbers to actual candidate identifications so that *now* we know what candidate actually received the money. Because obviously it would make too much sense just to note directly which candidate received these funds.

Upon generating this set of information, I was now able to take the set of all unique addresses from the YSOC database and lookup using the geocoding map of three weeks ago the lat/lon values *every* donor in their database resides at, then this had to be used as the lookup key for all the *transaction totals* I generated above. This allowed me to see only contribution totals for each candidate that YSOC donors engaged in. This set is what led to our first conclusion of candidates of interest to YSOC contributors. I was able to look at all the hits of candidates from YSOC’s data to find these main candidates that receive contributions:



Note that the **ethics\_num** field is the candidate’s identification number and the **count** represents the total number of transactions that that candidate received, regardless of monetary value. The sum represents the actual total number of gifts. This was exactly what we needed to now start looking into similar contributions across YSOC’s database.

I selected a set of 6 top politicians and their corresponding **ethics\_num** values to then link against every donor in YSOC’s database to see how much each contributed to these top politicians. This was done using a SUM function and JOIN function between the YSOC donor database, filtered through the geocoded map, and joined to this table. This now resulted in every YSOC donor having specific information about their donations both to YSOC over the past two years, one year, 90 days, and 30 days in total as well as their giving to these 6 top politicians.

Luckily, I had census data already preloaded on my Mac, so this allowed me to link in income data with two lines of SQL, which now added median household income and mean household income for every one of these linked donors.

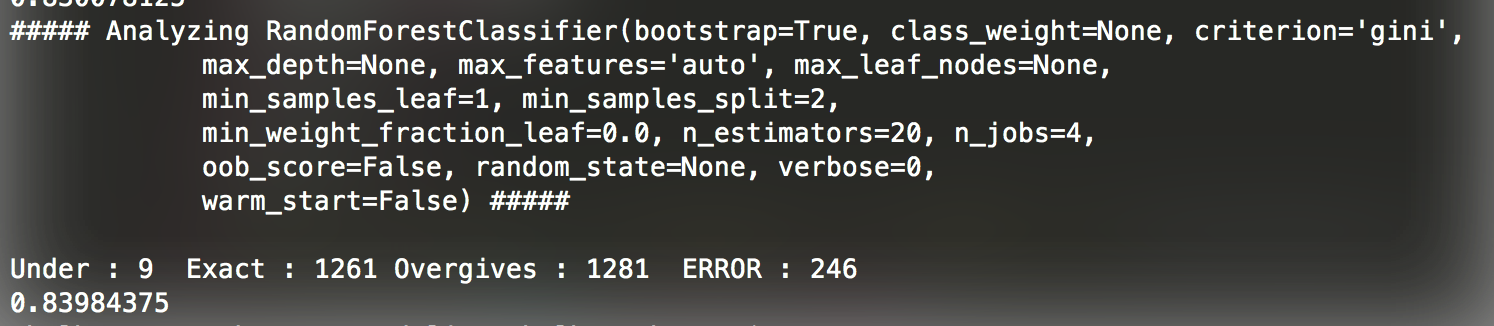
Thus, the machine learning began.

I first examined the distribution of **two\_year\_total** donations for YSOC donors (which from prior experiments consistently showed the greatest accuracies for my algorithms) and discovered that after using a simple logbase10 function of the real-valued column, a meaningful distribution of about 10 different classes resulted. That is, the function in Python

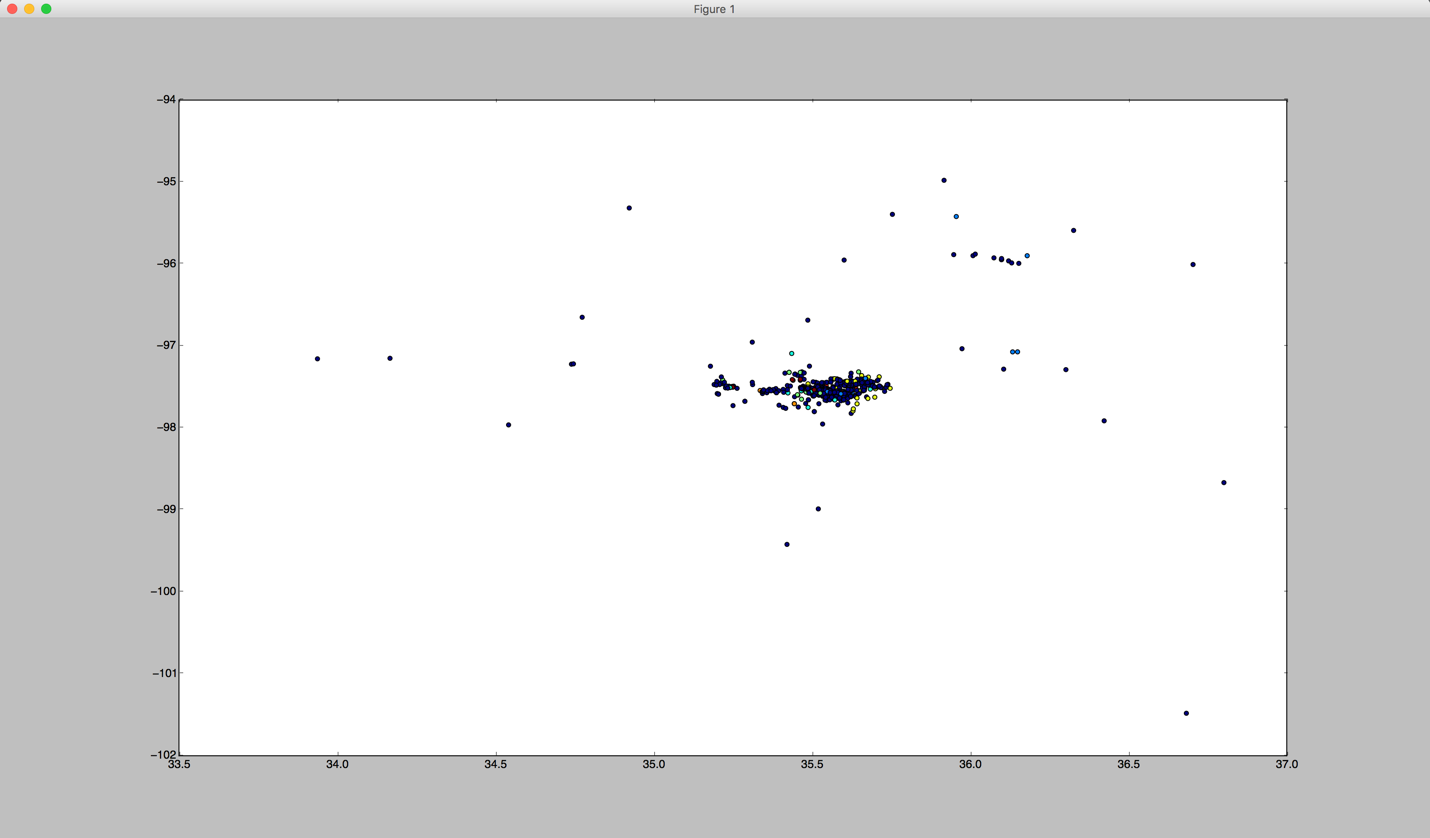
*categorized = list(map(lambda x: int(math.floor(math.log(x))) if x > 0.0 else 0, twoYear))*

was able to convert every value of contributions into a categorical label instead of continuous value. While the distribution is still mostly dominated by donors with $0 contributions to YSOC in the past two years, approximately 200 of 1500 resulted in nonzero labels of interest. It’s definitely not the worst dataset split I’ve ever seen, but it’s close.

These categorical labels then were put through the battery of tests I had constructed for prior workstreams and it resulted in an impressive ***84%*** accuracy of prediction. More importantly, the points and their output classes were airing on the side of pessimism in predictions. That is, if the model predicted a class of giver within 2 of the actual, it almost always aired on the side of being a *lower* class than a higher one, thus it rarely overestimated giving from donors. However, it was still able to identify donors who likely are being underasked and should be followed up with.

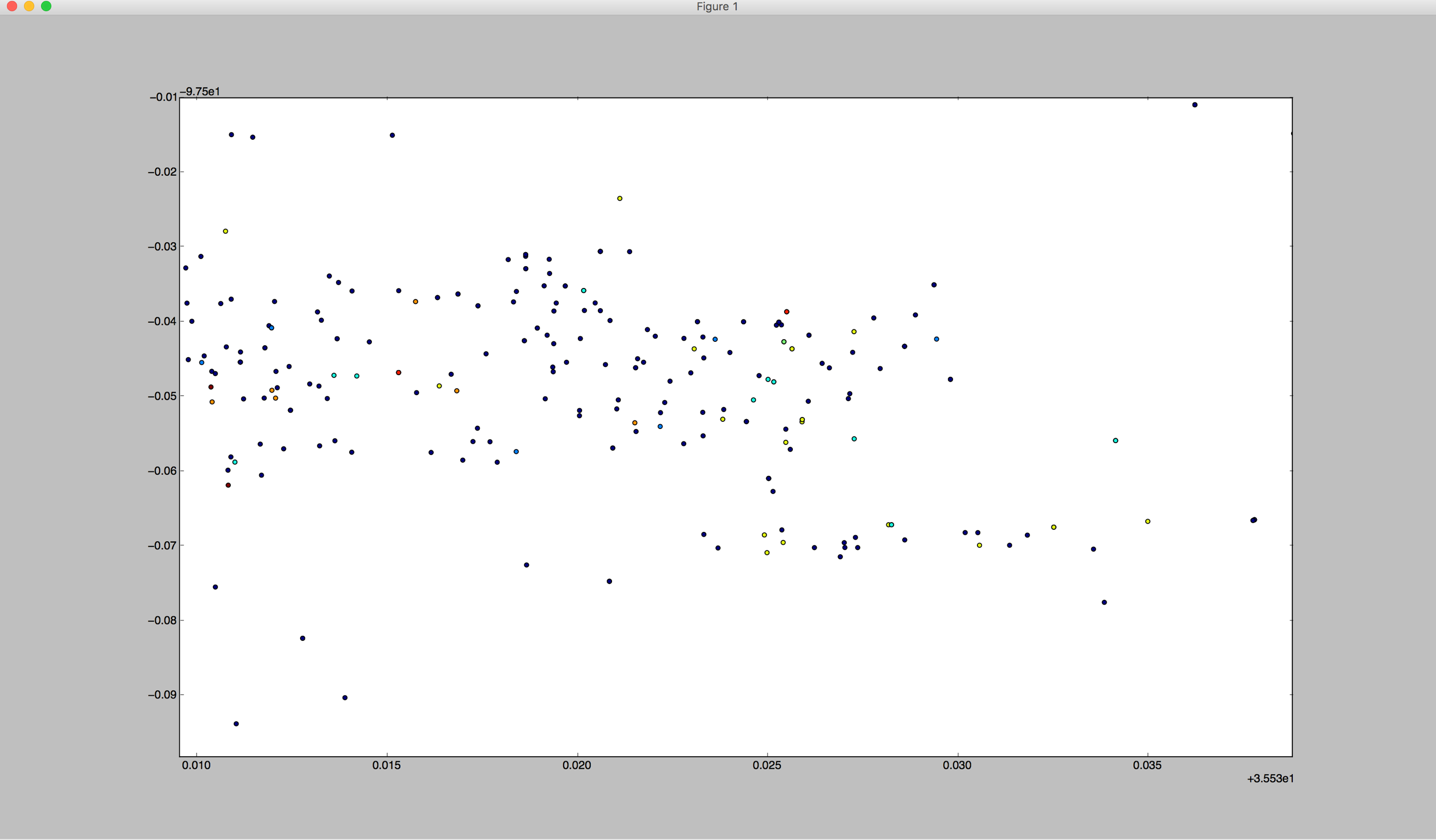


Now, let us explore graphically what this looks like.



Does that look like Oklahoma? Because it actually shows the concentration of donors for YSOC and the output classes the model predicted!

Zooming in and inspecting neighborhoods within the model, you can see there are actually pockets of similar classes clustered together in certain areas, seen below.



What is also curious to note is that there are points of dark blue (representing no giving in the past two years) surrounded by other points of homogenous color, and this makes me suspect these are donors that haven’t been stewarded effectively and have fallen off giving. Note in the lower right a cluster of three yellow points that have a neighbor registered with YSOC that hasn’t given at all!

Take Aways

We have a model. And it shows donors that aren’t being stewarded well, but more importantly it shows a good gradient of points that need to be analyzed further. I crashed my machine several times and had to rewrite a small portion of the operating system at one point trying to get it to show a true map of the state and roadways for better context, but Python and macOS Sierra made sure that didn’t happen.

The state is pretty clearly obscuring anyone from asking questions of their data. This is disturbing. It took a ridiculous amount of SQL programming to get transactions in donations linked together, and even when this was accomplished there was nothing short of 3 different identification systems that had to be synchronized. This is ridiculous.

Next Steps

Fix the map to actually show a map of Oklahoma and identify points of underasking. These would be lovely to come up with a set of addresses and predicted giving they *should* be donating. Yes this is essentially playing god, but I’d love nothing more than to show a specific case of the data showing how someone should be donating and it actually getting more money out of an existing donor!

Data Science. It works.