Recommending Potential Businesses using the Yelp Dataset

Daniel Saltiel
General Assembly Data Science Fall 2014

All code used for analysis available at http://github.com/drsaltiel/GA_final

Motivation

- Use the Yelp dataset to develop an indicator of future business success for a specific location
- Do this by quantifying the need for businesses of different types

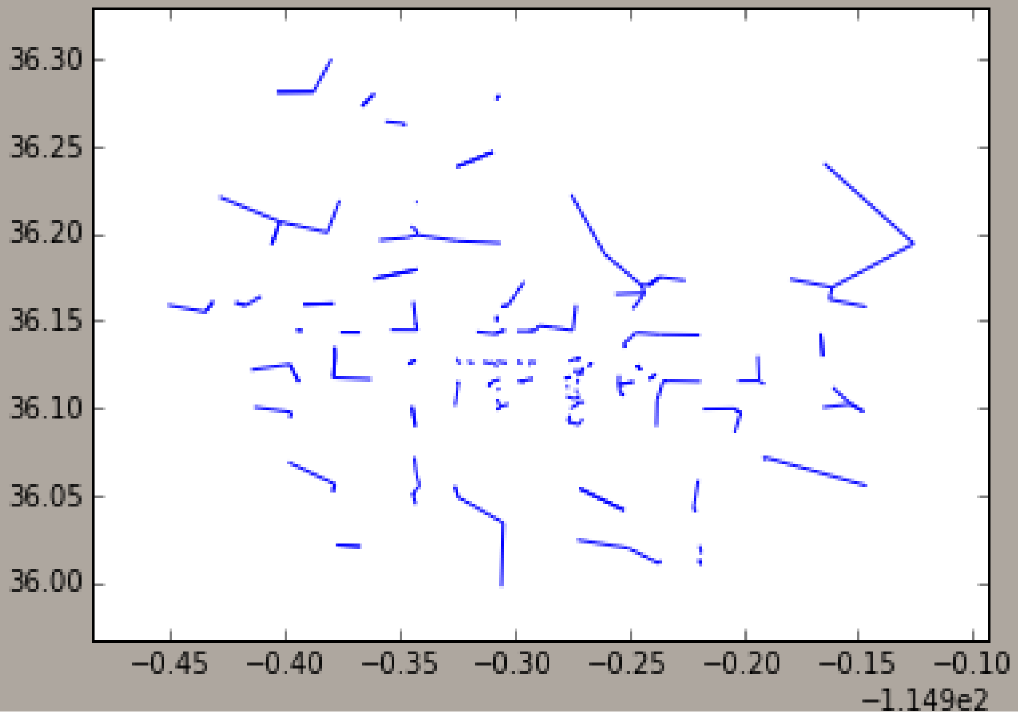
Our first feature:

Density of Businesses

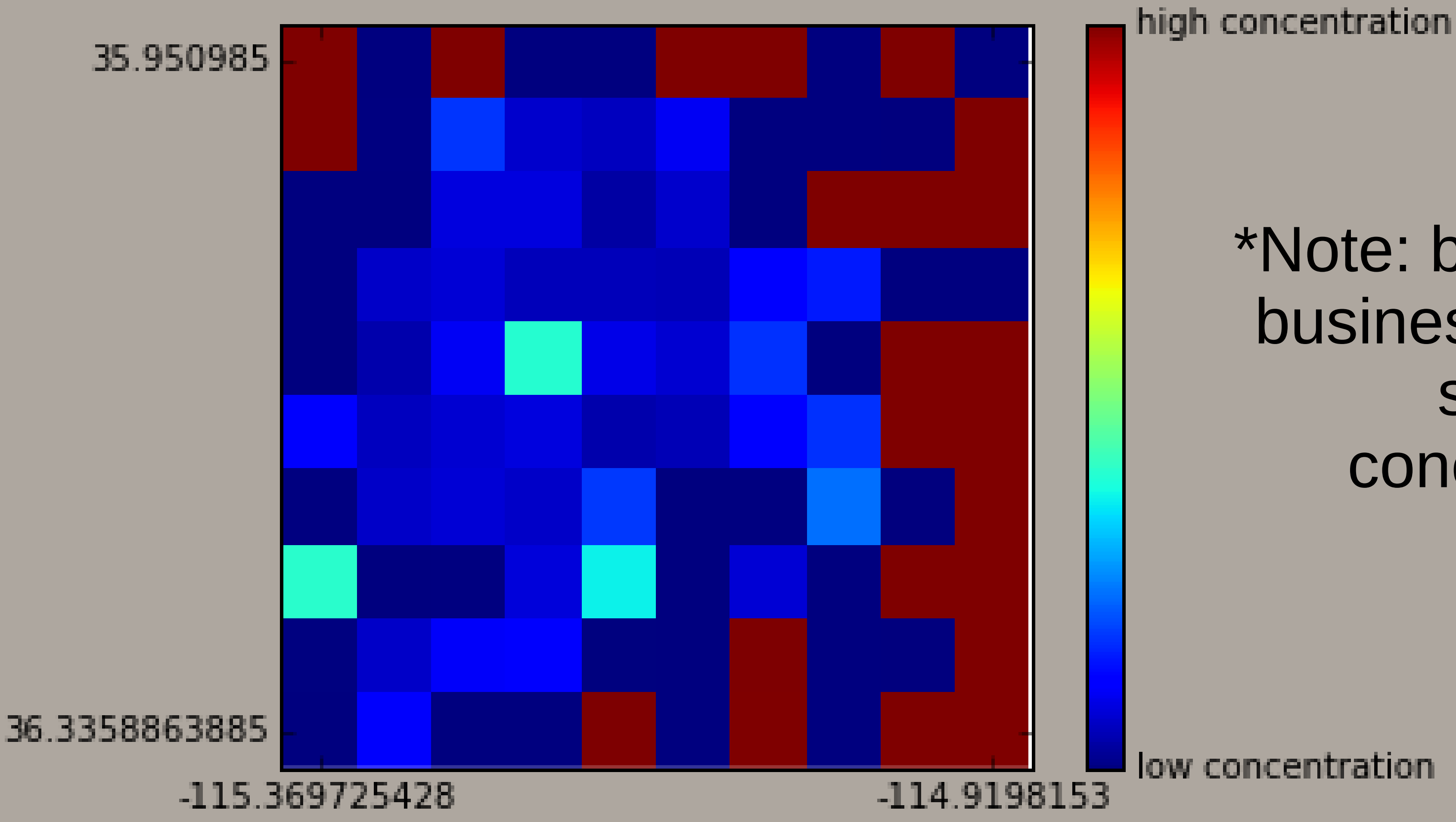
- Using the latitude and longitude of businesses we can determine the density of a certain business category

Example: Chinese food in Las Vegas

- 0.375 miles from a Chinese restaurant to another Chinese restaurant



Density of Businesses
Chinese Restaurants in Las Vegas



*Note: bins with 0 businesses were set to max concentration

But what if there is no interest in this type of business?

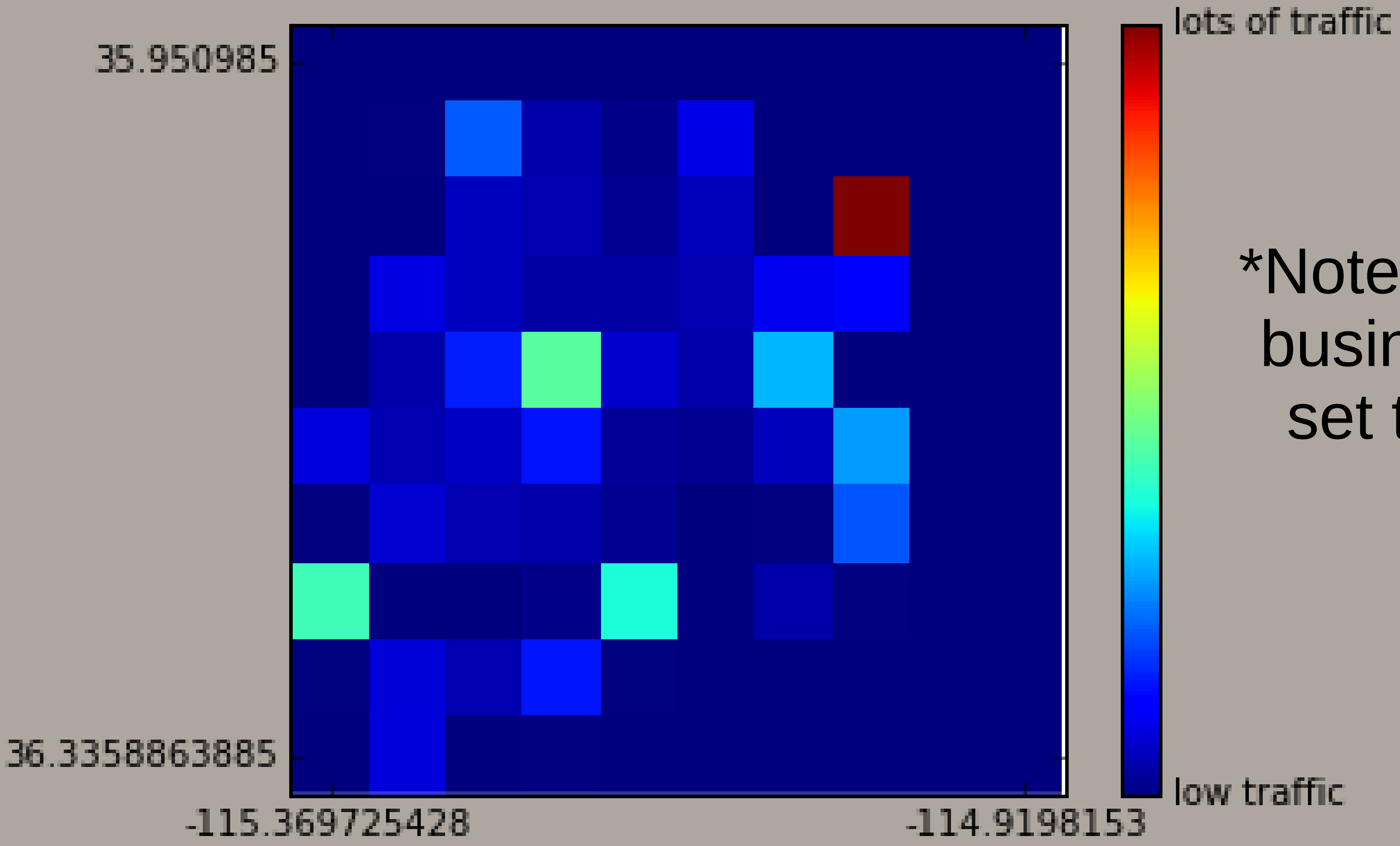
Traffic to Businesses

- How do we know how much interest there is in a given business category at a location?
  - Traffic to businesses of that category in the area
  - How do we know how much traffic there is?

Review Count

More Reviews = More Customers

Traffic to Businesses



*Note: bins with 0 businesses were set to min traffic

Let's Combine Our Metrics

Business Suggestion Algorithm

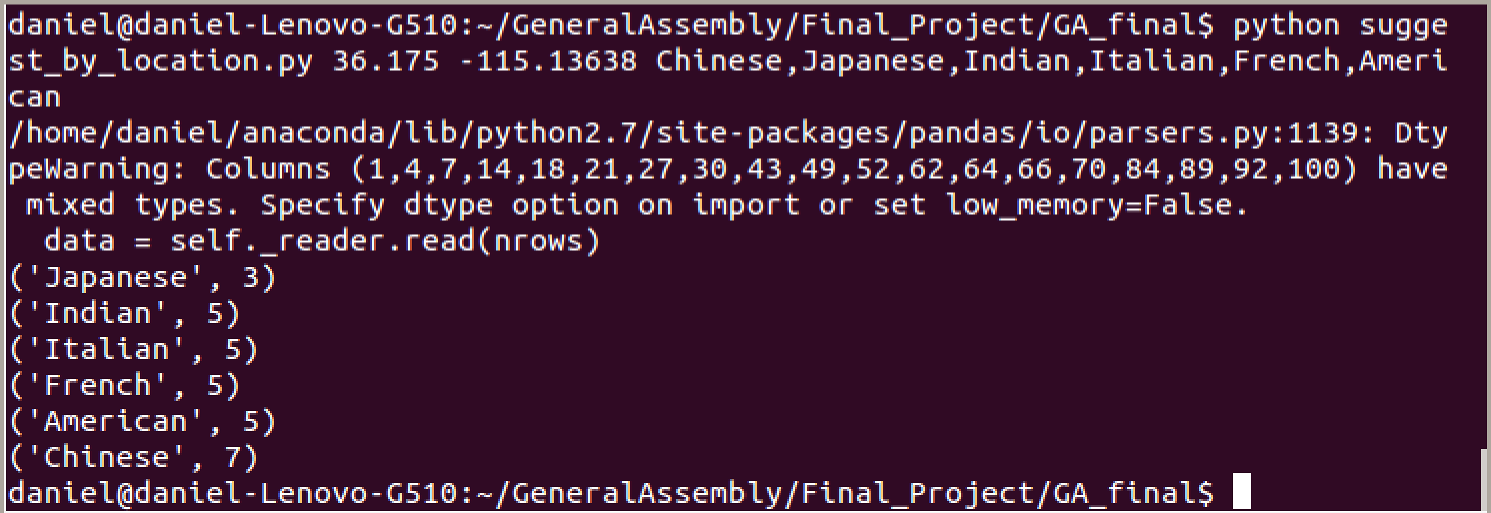
By Density

- ('Indian', 0.0024437074533077328)
- ('French', 0.0031419095828242277)
- ('Japanese', 0.017978704835049746)
- ('Italian', 0.020509687554547042)
- ('Chinese', 0.023826147669750393)
- ('American', 0.05445976610228661)

By Demand:

- ('American', 0.13385131970801337)
- ('Japanese', 0.04726248962550562)
- ('Italian', 0.0427402045787006)
- ('Chinese', 0.03587812071119758)
- ('French', 0.026954699397330025)
- ('Indian', 0.005607199492260039)

Combined



```
daniel@daniel-Lenovo-G510:~/GeneralAssembly/Final_Project/GA_final$ python sugge
st_by_location.py 36.175 -115.13638 Chinese,Japanese,Indian,Italian,French,Ameri
can
/home/daniel/anaconda/lib/python2.7/site-packages/pandas/io/parsers.py:1139: Dty
peWarning: Columns (1,4,7,14,18,21,27,30,43,49,52,62,64,66,70,84,89,92,100) have
 mixed types. Specify dtype option on import or set low_memory=False.
  data = self._reader.read(nrows)
('Japanese', 3)
('Indian', 5)
('Italian', 5)
('French', 5)
('American', 5)
('Chinese', 7)
daniel@daniel-Lenovo-G510:~/GeneralAssembly/Final_Project/GA_final$ 
```

Problem:
What if traffic matches density of businesses?

- We can't help ya.
  - If this is the case, there is no favored business category since the algorithm works by finding the disparity between number of businesses and interest in business category.
- Could potentially be fixed in the future
  - Using actual values instead of just ranks
  - Need a way to quantify absolute supply vs. demand

The Trouble with Validation

- Too many variables
  - Just because there is a demand for a business doesn't mean any given business of that type will thrive
- Most businesses will be in the middle
  - Would need to find a specific location and set of businesses in which the actual business was in the extremes.

Expanding the Concept

- Basic algorithm can be expanded to any categorical variable for which there is data
  - Price range
  - Ambiance
  - Etc...
- Corollary: the more categories the better
  - The fewer the number of categories, the more likely that traffic will perfectly mirror concentration of businesses
  - If category traffic and concentration are symmetrical,

    no useful recommendation is produced

Summary

Using only geographic location and number of reviews, we have developed a method to examine the interest in a specific type of business vs the concentration of that business, and from this suggest a certain category of business out of a group of categories for a given location.

Future Direction

- Using actual values instead of just ranks
  - More sensitive to differences, more information
  - Could create absolute scale could be used to gauge individual business category