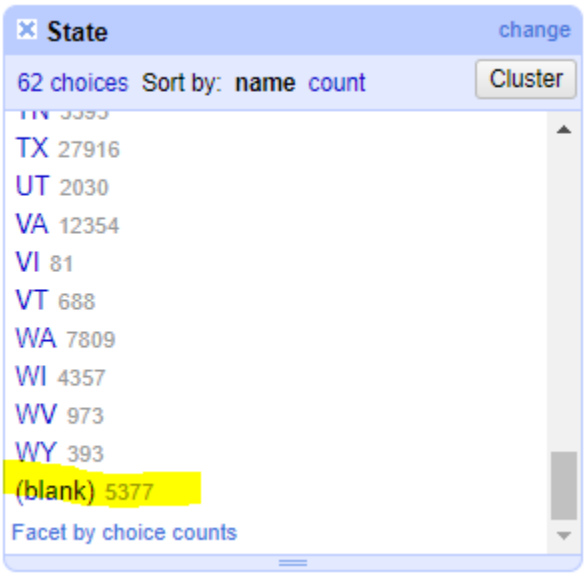
W205 summer 2017 section 2 – Lab08 - K Iwasaki

Step 1. Wrangling the Customer Complaints Data

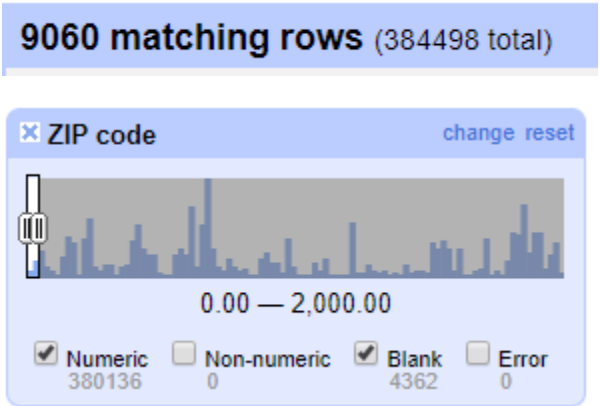
Submission 1: *How many rows are missing a value in the "State" column? Explain how you came up with the number.*

5377 rows are missing a value. I looked at text facet of the state column and found there were 5377 blanks.



SUBMISSION 2: *How many rows with missing ZIP codes do you have?*

In order to validate ZIP codes, I referred the ZIP codes list at *http://www.phaster.com/zip_code.html* . This is not exact but I consider ZIP code lower than 2000 invalid. There are 9060 rows missing ZIP codes.



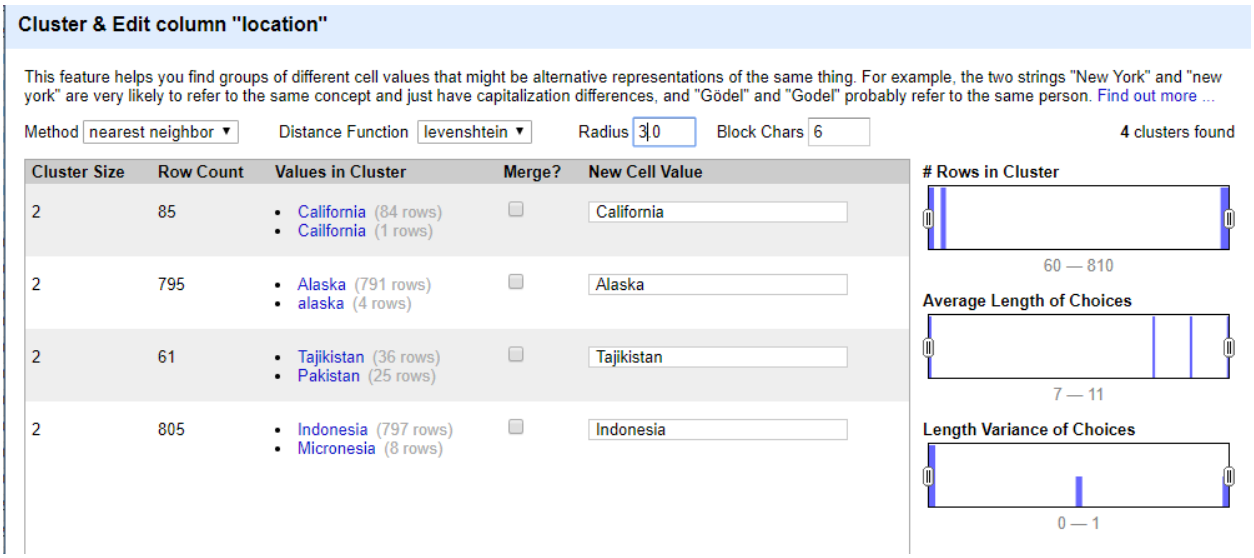
SUBMISSION 3: *If you consider all ZIP codes less than 99999 to be valid, how many valid and invalid ZIP codes do you have, respectively?*

There are 34,961 rows having invalid ZIP codes and 349,537 rows having valid ZIP codes

Step 2. Cleaning Up eq2015 Data

SUBMISSION 4: *Change the radius to 3.0. What happens? Do you want to merge any of the resulting matches?*

I found more clusters that are candidates for merging. Top two clusters in the list seems find to merge.



SUBMISSION 5: *Change the block size to 2. Give two examples of new clusters that may be worth merging.*

5	798	• Alaska (791 rows) • alaska (4 rows) • Alaksa (1 rows) • Alaka (1 rows) • Alska (1 rows)	Alaska
3	36	• Canada (33 rows) • Candaa (2 rows) • Cnaada (1 rows)	Canada

SUBMISSION 6: *Explain in words what happens when you cluster the "place" column, and why you think that happened. What additional functionality could OpenRefine provide to possibly deal with the situation?*

Hint: you may want to cancel the run.

It doesn't return the result. I think this is because it is very expensive to compute clusters for "place" column because the column has the strings which is much longer than the strings for "location" column. The longer the strings, the more expensive the computation is.

Additional functionality such as alert function that if the string is long and is expected to require lots of computation, it sends a message to consider shorten the string, would prevent this situation.

Step 3. Levenshtein Distance

SUBMISSION 7: *Submit a representation of the resulting matrix from the Levenshtein edit distance calculation. The resulting value should be correct.*

		1	2	3	4	5	6	7	8	9	10
			g	u	m	b	a	r	r	e	l
1		0	1	2	3	4	5	6	7	8	9
2	g	1	0	1	2	3	4	5	6	7	8
3	u	2	1	0	1	2	3	4	5	6	7
4	n	3	2	1	1	3	4	4	5	6	7
5	b	4	3	2	2	1	2	3	4	5	6
6	a	5	4	3	3	2	1	2	3	4	5
7	r	6	5	4	4	3	2	1	2	3	4
8	e	7	6	5	5	4	3	2	2	2	4
9	l	8	7	6	6	5	4	3	3	3	2
10	l	9	8	7	7	6	5	4	4	4	3

```
In [6]: distance("gumbarrel", "gunbarell")
Out[6]: 3
```