CMSC 478
Project final submit

My project was an attempt to create a model that predicts the length of time an area is declared to be in a state of disaster, following a natural disaster as declared by FEMA. My data set comes from Kaggle, see my initial submission for the source and a full list of columns.

I decided to use a naive bayes classifier, because much of the data is categorical ie for the 'Disaster.Type' column the values are 'Fire','Tornado','Flood','Hurricane', etc. I decided to create my own column to use as the result variable. I could calculate the number of days between start and end date, by simple subtraction. I knew that I wanted to create categories based on these day lengths, so I decided to round the values. I rounded them to nearest 100 value. I also added columns for each rounded value to intervals of a hundred (100-1000). I noted that the error rate decreased steadily with each increased rounding range (see milestone 2). However, the usefulness of the prediction also decreases greatly. I also decided to attempt to make my own buckets of equal size. This did not produce results any better than rounding to 100. Unfortunately even rounding to 100 only got a fifty percent success rate. I also performed some 10-fold validation to make sure that my results are not based on my initial sample. I assessed model accuracy by creating a confusion table and summing the diagonal (as seen in 'create_model.r' and 'isolate_data.r').

During the preprocessing of the data, I did have to remove some incomplete rows. There were many rows that did not have entries in the county field, so I removed them. The data set also had some redundant columns, those were removed. I also opted to remove all of the rows that represented regions still in a state of disaster. All told, this removed about half of the entries (~46,000 to ~26,000).

In order to improve the model, I tried several different combinations of variables, but found that keeping all in makes for the best (still terrible) prediction. Despite the depressing result, I have found success with some fixed values. For example, I tried to predict based the length of time given a state, here are the results for the top ten most frequently appearing states:

	State	Freq	mean.length	range.from	range.to	round.100.error
51	TX	2210	2433.167	300	5000	0.29683258
29	MO	1948	2071.766	300	4500	0.16324435
53	VA	1433	2214.445	200	5000	0.07670851
42	OK	1240	2359.516	100	4700	0.21612903
21	KY	1187	3223.757	400	5200	0.16329966
16	IA	993	2625.478	500	5200	0.14084507
44	PA	940	2707.872	0	6600	0.07446809
19	IN	937	2414.088	500	4500	0.08955224
2	AL	906	2315.011	100	5500	0.15452539
41	OH	905	2011.160	500	4300	0.21633554

These results are significantly better than without knowing the state. I pulled out the range for each of these states to make sure these improved predictions weren't because there was a limited range for each state, on the contrary all of these exhibit a range of more than 4000.

Similarly, I attempted to predict the length of days given the type of disaster, here are the top 10 most frequently occurring disaster types:

	Type	Freq	mean.length	range.from	range.to	round.100.error
13	Storm	9283	2906.0325	0	7500	0.47802671
6	Flood	6682	2862.2119	0	8700	0.43070937
8	Hurricane	4941	2383.1411	200	6600	0.33346823
12	Snow	2661	1887.8617	0	5800	0.20435763
5	Fire	1720	1891.3372	100	6300	0.37325581
9	Ice	1327	2298.7943	400	5500	0.15361446
15	Tornado	1085	2442.5806	0	6600	0.16390424
3	Drought	920	618.6957	0	3000	0.03695652
11	Other	274	1957.6642	400	5000	0.10218978
20	Winter	258	2007.7519	400	4500	0.05426357

These results are pretty good. Clearly it is hard to predict on 'Storm' and 'Flood', possibly due to the large range of values.

Finally, I also tested to see predictions based on given initial year. These are the top 20 most frequently appearing start years:

	Year	Freq	mean.length	range.from	range.to	round.100.error
43	2005	3183	2033.0192	100	3700	0.24497487
34	1996	1466	2950.6139	400	6000	0.15961801
31	1993	1460	3046.5068	300	6600	0.05479452
41	2003	1343	2878.7044	100	5000	0.20535714
36	1998	1182	3717.8511	600	6400	0.18950931
42	2004	1139	2775.3292	100	4100	0.22456140
45	2007	1057	1981.3623	200	3400	0.33837429
37	1999	1031	2688.6518	600	6200	0.18217054
46	2008	1019	2091.0697	200	3200	0.32745098
38	2000	972	2981.0700	300	5800	0.08436214
15	1977	893	948.8242	0	6300	0.10067114
40	2002	877	3396.3512	100	4800	0.24145786
49	2011	828	1022.3430	200	2100	0.12560386
11	1973	750	1501.3333	900	2900	0.16800000
35	1997	681	3603.6711	700	6200	0.03225806
39	2001	664	3285.8434	400	5300	0.12650602

28	1990	652	3993.5583	1900	5600	0.14110429
47	2009	648	1782.5617	400	2800	0.20370370
50	2012	582	851.0309	200	1600	0.23024055
32	1994	570	4023.8596	1800	6600	0.02807018

These results seem to show that the year is very useful in predicting the length in days. It also doesn't seem to suggest that there is a great difference between the 1990s and 2000s. Having initial year in the model is not useful for trying to predict *future* disaster lengths, but it does provide insight into past data.

Ultimately I successfully created a model that could predict length of time in days rounded to 100 for disasters, given the state. I also gained some insight into the data, including value ranges for several of the columns, and which columns are important as predictors.

**My code has not drastically changed since the previous milestone, but I added some functions and loops to create the tables shown in this paper.