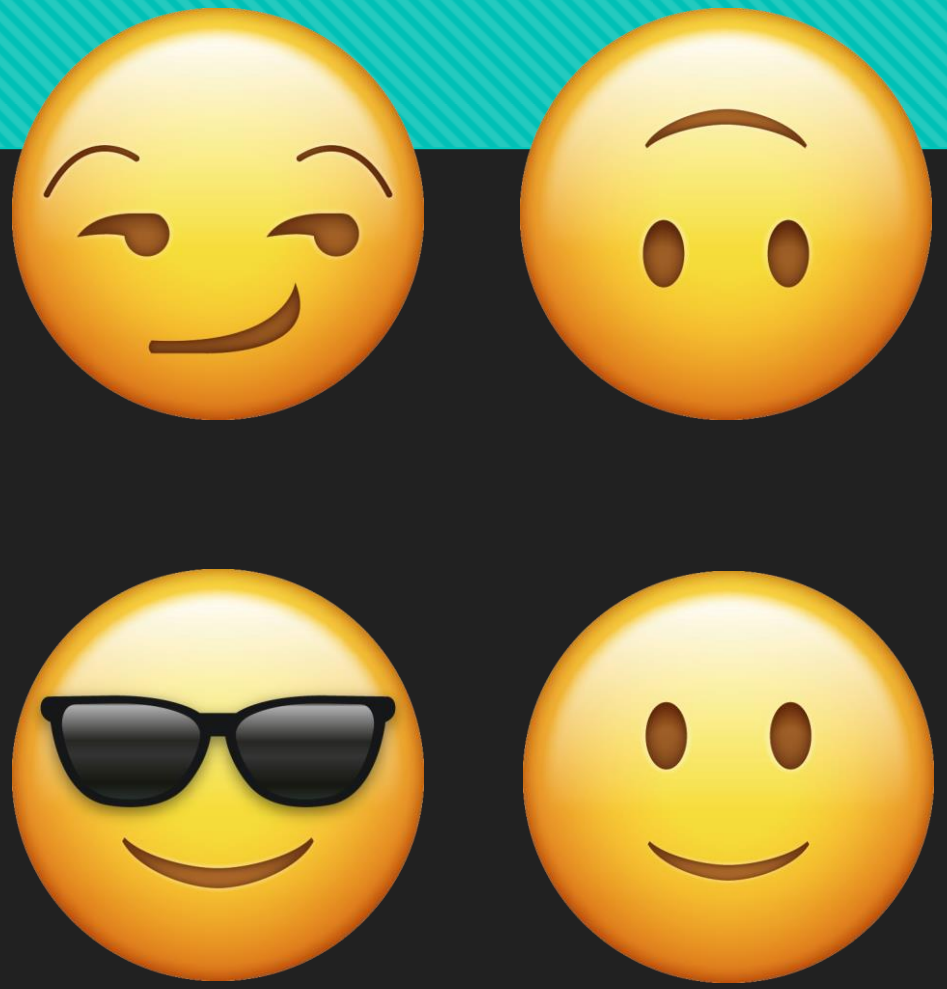
Emoji Prediction: A Survey of Classification Algorithms 

Alexandra Gamez

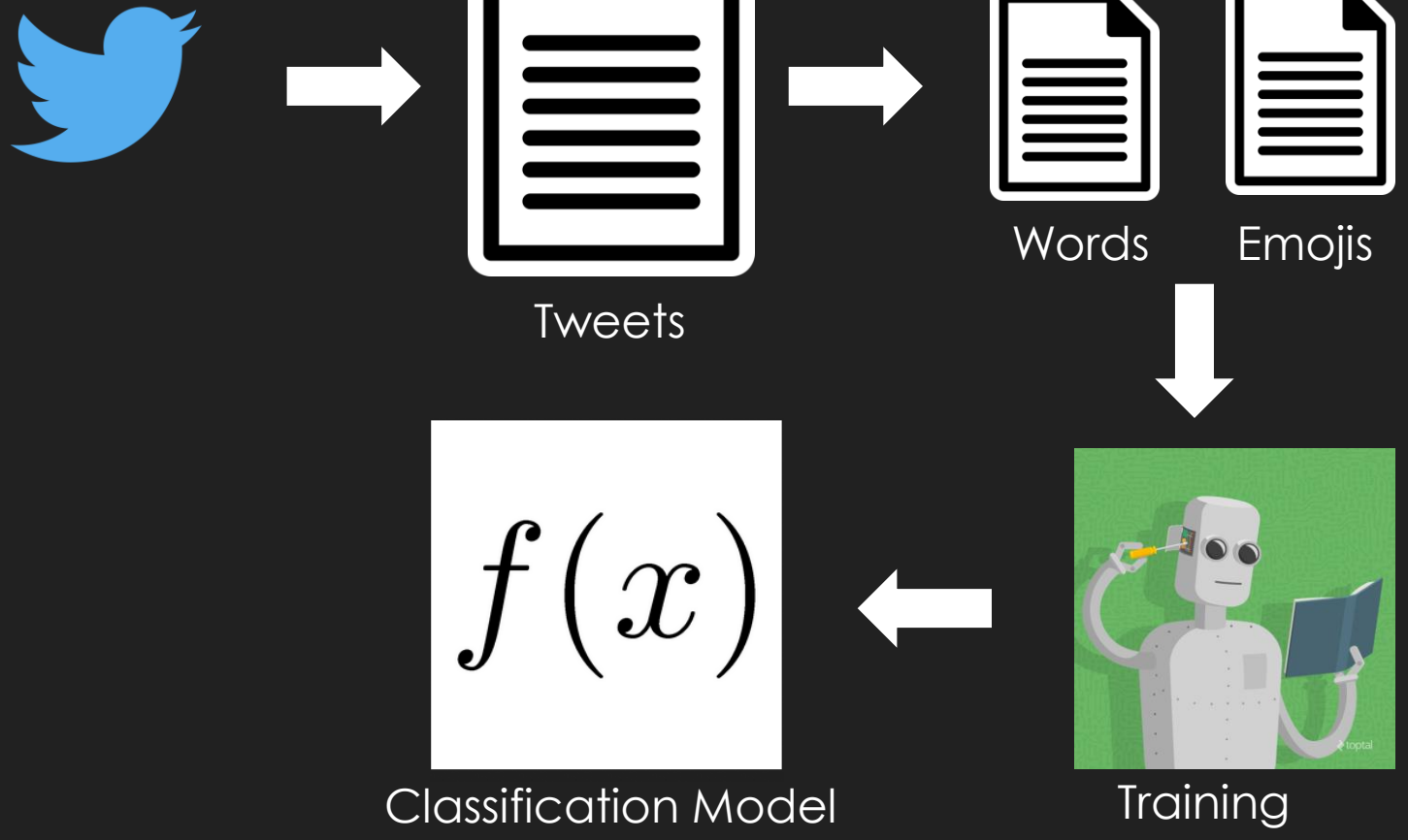
Background

- Twitter is a place where valuable information such as sentiments, popularity, and opinions of various topics are located
- A perfect place for Natural Language Processing
- Emoji prediction is a classification problem



Problem & General Approach

- Task
  - Emoji prediction
- Approach
  - Get data
  - Prepare data
  - Train data
  - Test



Dataset Example

Tweet content

75	PS I U @ Beaver Stadium
76	Get ready for a bunch of annoying pictures of Dallas @ Dallas, Texas
77	@user aww love ya laura!!!
78	Hoes never get cold @ Downtown Los Angeles
79	National Siblings Day #WeAreFamily #HappyNationalSiblingsDay #SistersLikeUs @ Time Square…
80	"Don't you hate working holidays?"... they asked. #roseparade2016 #ilovemyjob #colorfulfloats…
81	#taromilkteaboba for this hot LA day. @ McDonald's at 2810 South…
82	S N ~ They're saying it's the hottest day of the year today. So we're hiding in a cabana by the…
83	Our fierce party crew! @ Blarney Stone Pub
84	Got to visit my grandpa K today :) @ Greenwood, South Carolina
85	"what's in the ice box, if you don't mind me asking sir?".....Our Hearts …

Emoji Label

75	8
76	5
77	0
78	7
79	9
80	0
81	4
82	12
83	6
84	3
85	8

Emoji Mapping

0	🤍	_red_heart_
1	😍	_smiling_face_with_hearteyes_
2	🤩	_face_with_tears_of_joy_
3	💕	_two_hearts_
4	🔥	_fire_
5	😊	_smiling_face_with_smiling_eyes_
6	😎	_smiling_face_with_sunglasses_
7	✨	_sparkles_
8	💙	_blue_heart_
9	😘	_face_blowing_a_kiss_
10	📷	_camera_
11	US	_United_States_
12	☀	_sun_
13	💜	_purple_heart_
14	😉	_winking_face_
15	💯	_hundred_points_
16	😁	_beaming_face_with_smiling_eyes_
17	🎄	_Christmas_tree_
18	📸	_camera_with_flash_
19	😜	_winking_face_with_tongue_

Algorithms Used

- Naïve Bayes (Multinomial Naïve Bayes)
- Stochastic Gradient Descent
- Support Vector Machines
- Logistic Regression
- K Nearest Neighbors
- Decision Tree
- Neural Networks
- My Naïve Bayes ***

Naïve Bayes

Likelihood

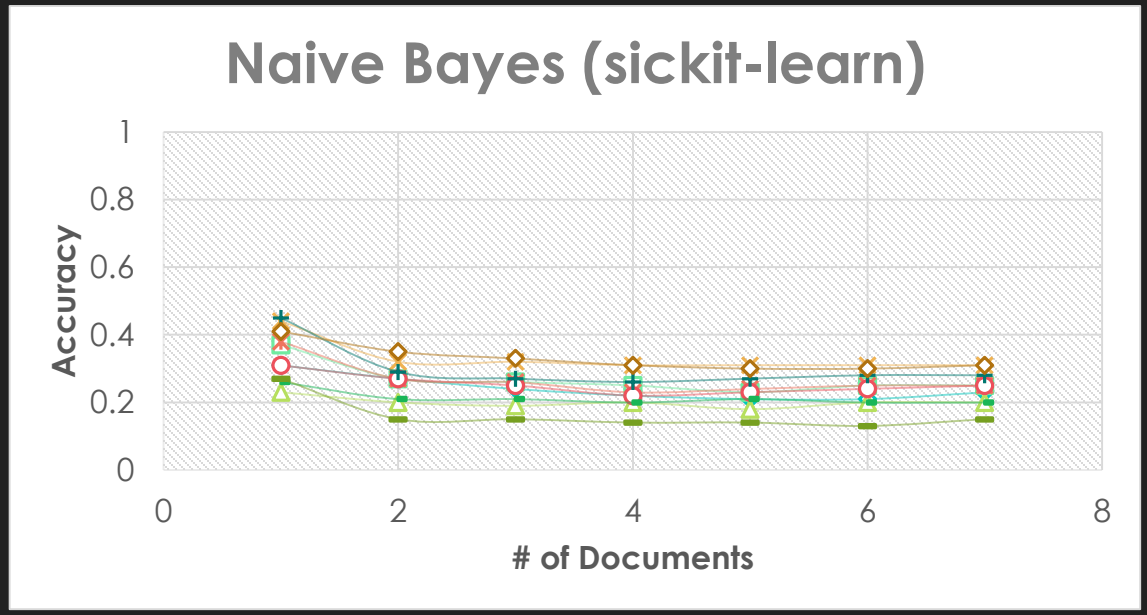
Class Prior Probability

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}$$

Predicted prior probability

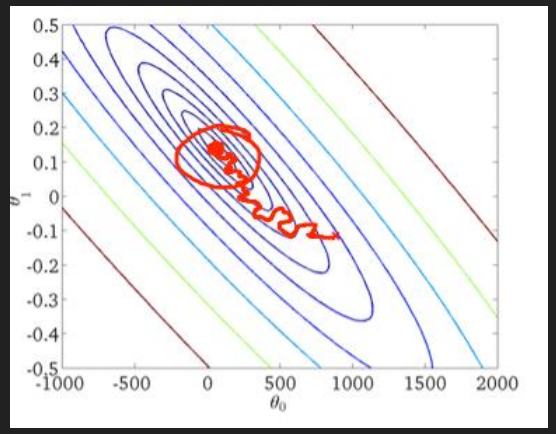
- Fast training
- Fast classification
- Terrible results

Naïve Bayes	Documents						
Training Set Size	1k	5k	10k	20k	30k	40k	50k
0-100	0.31	0.27	0.24	0.22	0.21	0.21	0.23
100-200	0.37	0.27	0.26	0.25	0.23	0.25	0.25
200-300	0.23	0.2	0.19	0.2	0.18	0.2	0.2
300-400	0.44	0.32	0.32	0.31	0.31	0.31	0.31
400-500	0.38	0.27	0.26	0.23	0.24	0.25	0.25
500-600	0.31	0.27	0.25	0.22	0.23	0.24	0.25
600-700	0.45	0.29	0.27	0.26	0.27	0.28	0.28
700-800	0.26	0.21	0.21	0.2	0.21	0.2	0.2
800-900	0.27	0.15	0.15	0.14	0.14	0.13	0.15
900-100	0.41	0.35	0.33	0.31	0.3	0.3	0.31
Training Time	0.015	0.046	0.071	0.125	0.266	0.33	0.224
Elaspsed time	0.171	0.171	0.188	0.205	0.161	0.63	0.268



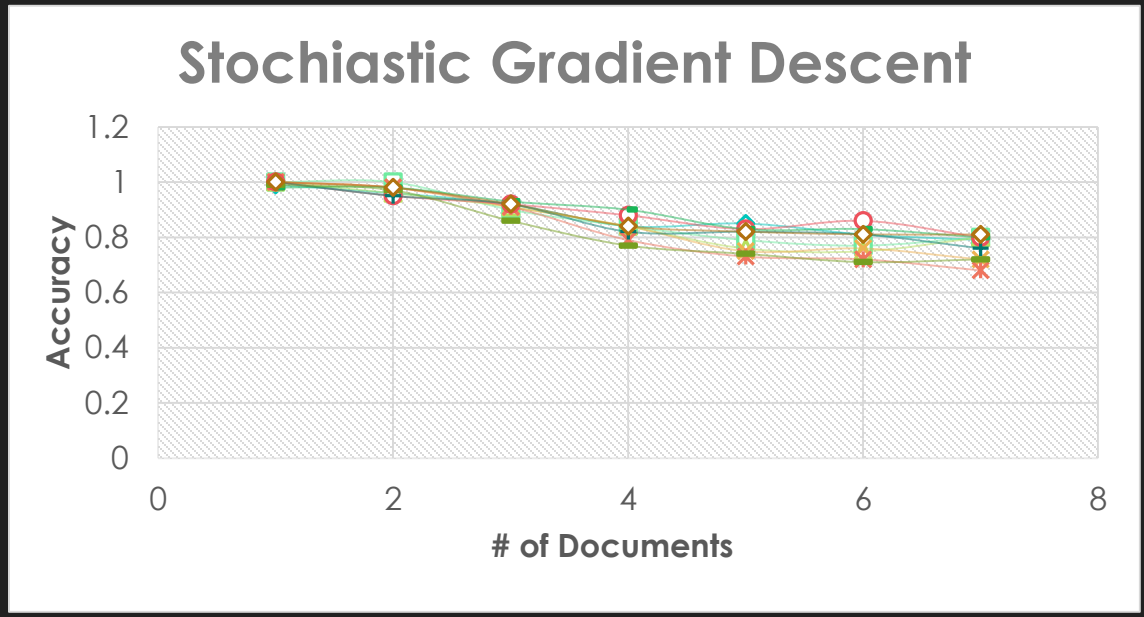
Stochastic Gradient Descent

- Trained by assigning weights and then updating iteratively until convergence at a maximum



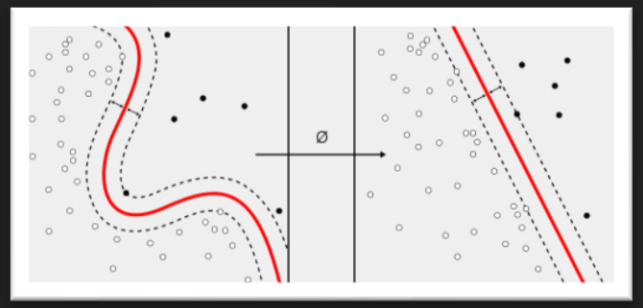
- Fast training
- Fast classification
- Sufficiently accurate

Stochiastic Gradient Descent	Accuracy						
Training Set Size	1k	5k	10k	20k	30k	40k	50k
0-100	0.99	0.96	0.92	0.84	0.85	0.81	0.79
100-200	1	1	0.9	0.84	0.79	0.77	0.8
200-300	1	0.96	0.91	0.84	0.76	0.75	0.8
300-400	1	0.98	0.91	0.84	0.75	0.76	0.72
400-500	1	0.98	0.91	0.79	0.73	0.72	0.68
500-600	1	0.95	0.92	0.88	0.83	0.86	0.8
600-700	1	0.95	0.92	0.82	0.82	0.81	0.76
700-800	0.98	0.98	0.93	0.9	0.83	0.83	0.8
800-900	0.99	0.97	0.86	0.77	0.74	0.71	0.72
900-100	1	0.98	0.92	0.84	0.82	0.81	0.81
Training Time	0.032	0.141	0.328	0.702	1.105	1.569	1.827
Elaspsed time	0.017	0.174	0.213	0.233	0.253	0.266	0.296



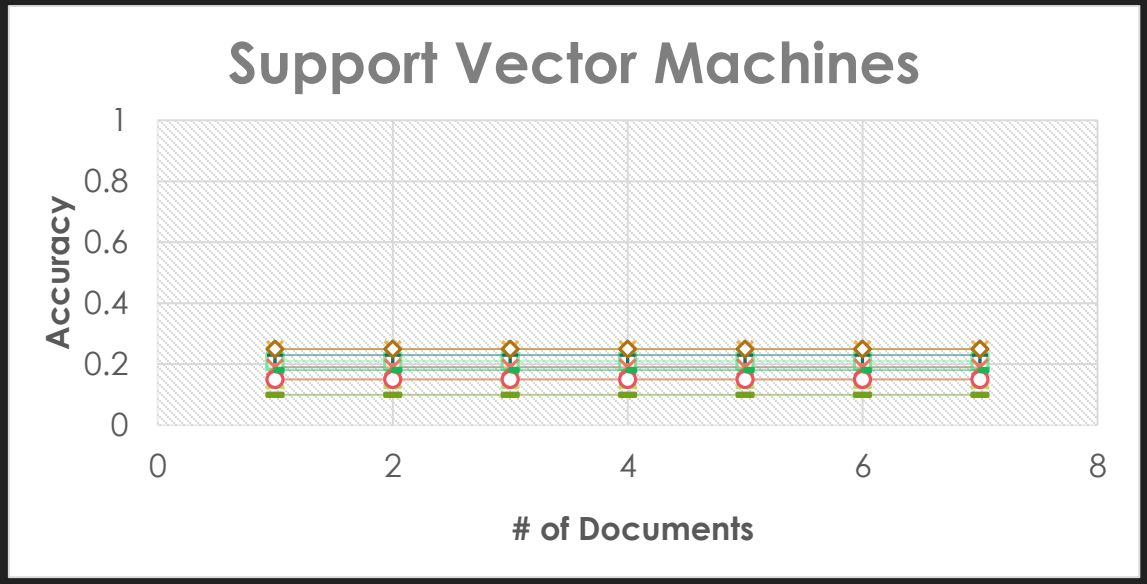
Support Vector Machines

- Points in space where categories are separated by gaps



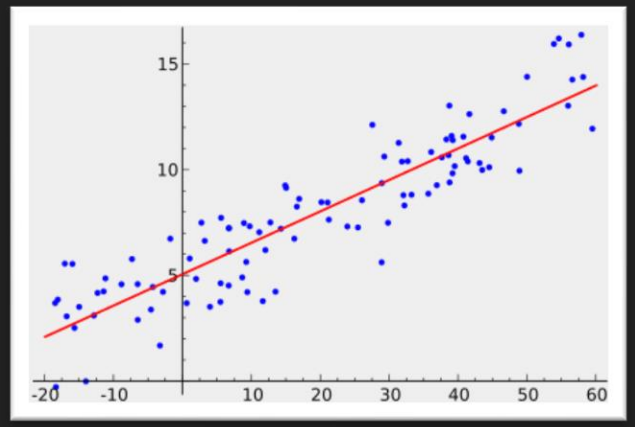
- Slow training
- Slow classification
- Miserable results

Vector Machines	Accuracy						
Training Set Size	1k	5k	10k	20k	30k	40k	50k
0-100	0.19	0.19	0.19	0.19	0.19	0.19	0.19
100-200	0.21	0.21	0.21	0.21	0.21	0.21	0.21
200-300	0.15	0.15	0.15	0.15	0.15	0.15	0.15
300-400	0.25	0.25	0.25	0.25	0.25	0.25	0.25
400-500	0.19	0.19	0.19	0.19	0.19	0.19	0.19
500-600	0.15	0.15	0.15	0.15	0.15	0.15	0.15
600-700	0.23	0.23	0.23	0.23	0.23	0.23	0.23
700-800	0.18	0.18	0.18	0.18	0.18	0.18	0.18
800-900	0.1	0.1	0.1	0.1	0.1	0.1	0.1
900-100	0.25	0.25	0.25	0.25	0.25	0.25	0.25
Training Time	0.674	14.68	58.409	253.598	572.56	1041.821	1344.291
Elaspsed time	0.673	2.44	5.304	9.736	18	19.182	21.89



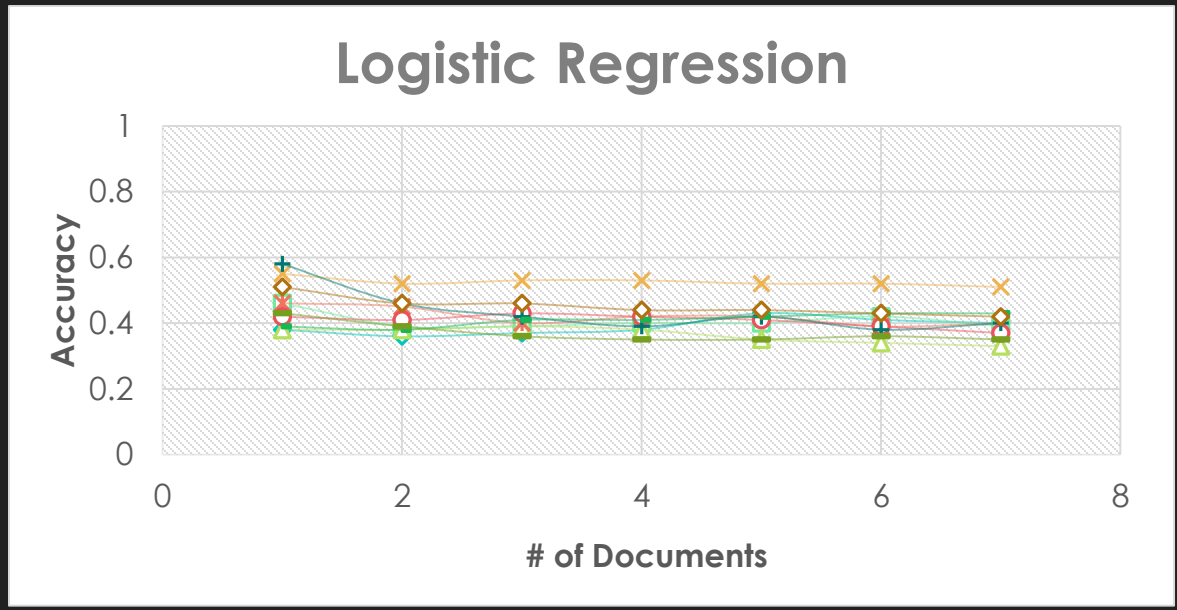
Logistic Regression

- Multinomial logistic regression also known as MaxEnt
- Features, scores, weights



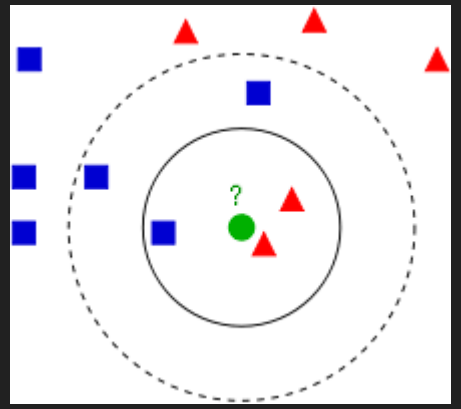
- Fast training
- Fast classification
- Subpar results

Logistic Regression	Accuracy						
Training Set Size	1k	5k	10k	20k	30k	40k	50k
0-100	0.38	0.36	0.37	0.38	0.43	0.41	0.4
100-200	0.46	0.39	0.39	0.4	0.4	0.42	0.39
200-300	0.38	0.38	0.39	0.38	0.35	0.34	0.33
300-400	0.55	0.52	0.53	0.53	0.52	0.52	0.51
400-500	0.46	0.45	0.4	0.42	0.42	0.39	0.4
500-600	0.42	0.41	0.43	0.42	0.41	0.39	0.37
600-700	0.58	0.46	0.42	0.39	0.42	0.38	0.4
700-800	0.39	0.38	0.41	0.41	0.42	0.43	0.43
800-900	0.43	0.39	0.36	0.35	0.35	0.36	0.35
900-100	0.51	0.46	0.46	0.44	0.44	0.43	0.42
Training Time	0.212	1.12	2.4	6.08	13.339	18.722	20.227
Elaspsed time	0.085	0.078	0.078	0.094	0.309	0.107	0.082

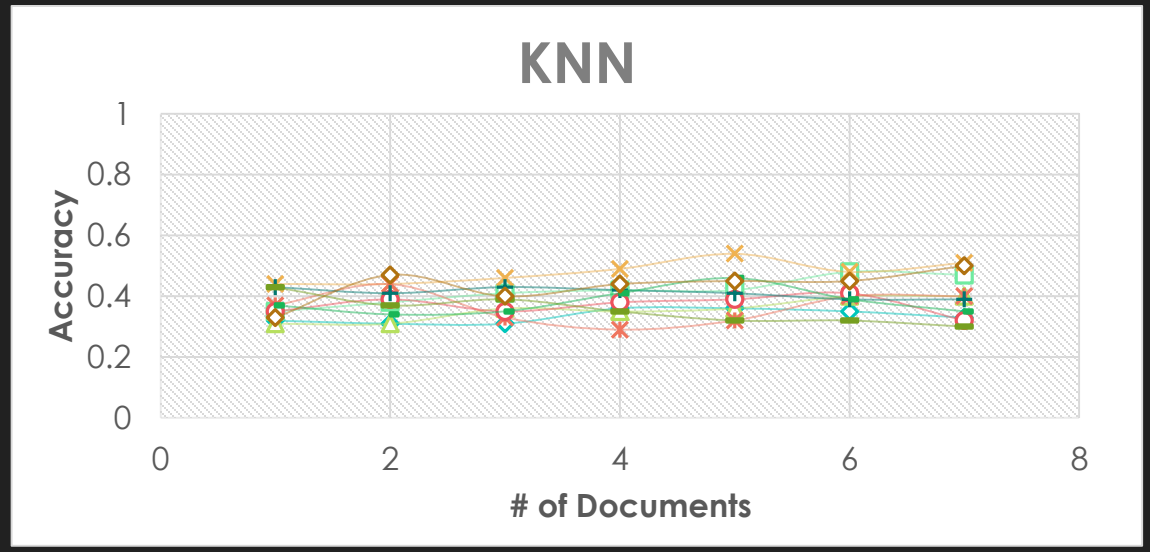


K Nearest Neighbors

- Classified by a majority vote of its neighbors n nearest neighbors



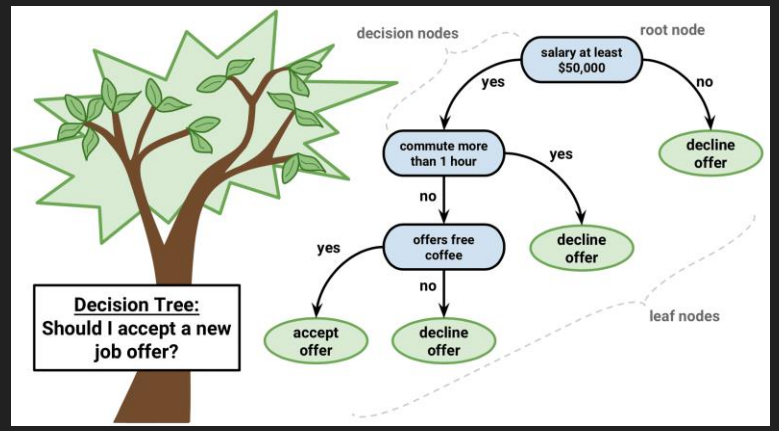
K Nearsest Neighbors	Accuracy						
Training Set Size	1k	5k	10k	20k	30k	40k	50k
0-100	0.32	0.31	0.31	0.36	0.36	0.35	0.33
100-200	0.35	0.38	0.41	0.42	0.42	0.48	0.47
200-300	0.31	0.31	0.36	0.35	0.36	0.4	0.4
300-400	0.44	0.44	0.46	0.49	0.54	0.48	0.51
400-500	0.37	0.44	0.33	0.29	0.32	0.4	0.4
500-600	0.35	0.39	0.35	0.38	0.39	0.41	0.32
600-700	0.43	0.41	0.43	0.42	0.41	0.39	0.39
700-800	0.37	0.34	0.35	0.41	0.46	0.39	0.35
800-900	0.43	0.37	0.39	0.35	0.32	0.32	0.3
900-100	0.33	0.47	0.4	0.44	0.45	0.45	0.5
Training Time	0	0	0	0.016	0.019	0.031	0.031
Elaspsed time	0.155	0.625	1.078	1.983	3.31	4.045	4.59



- Really fast training
- Relatively fast classification
- Pretty bad results

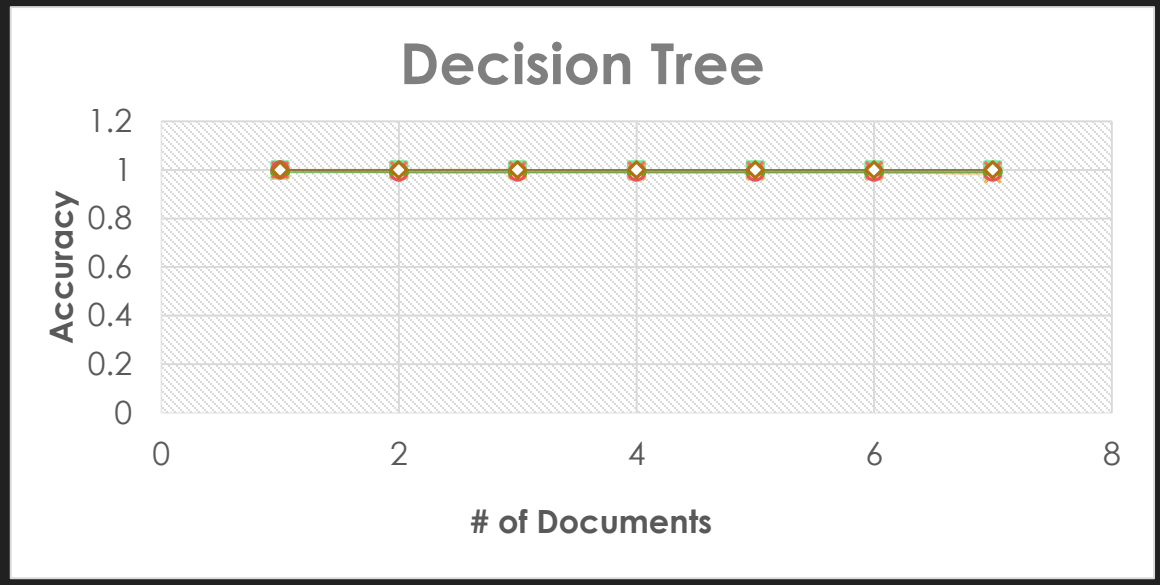
Decision Tree

- Data into subsets
- Decision nodes



- Slowish training
- Fast classification
- Really good results

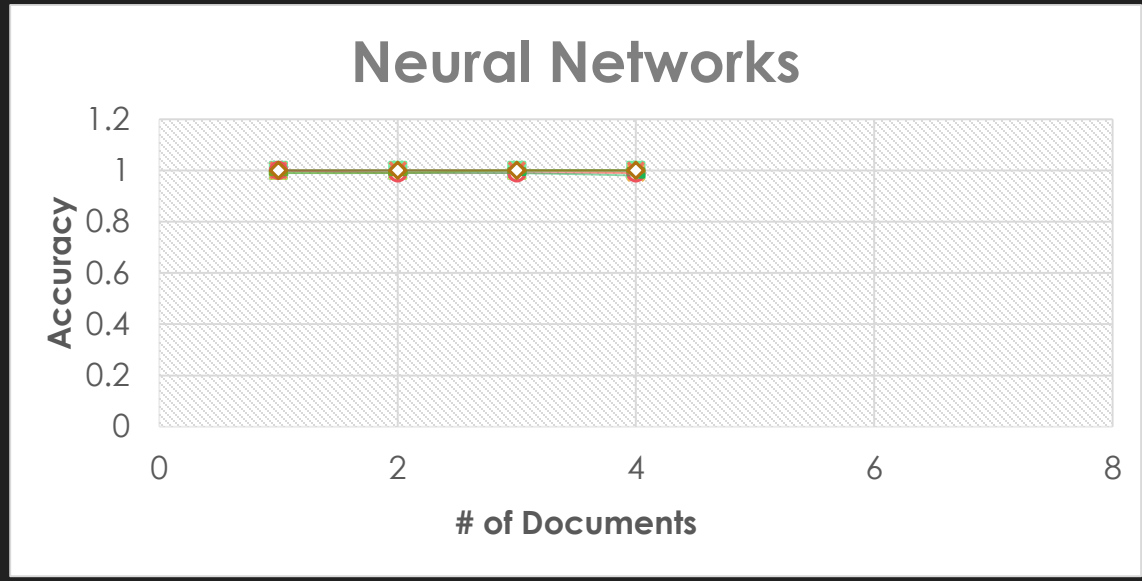
Decision Tree	Accuracy						
Training Set Size	1k	5k	10k	20k	30k	40k	50k
0-100	1	1	1	1	1	1	1
100-200	1	1	1	1	1	1	1
200-300	1	1	1	1	1	1	1
300-400	1	1	1	0.99	0.99	0.99	0.98
400-500	1	1	1	1	1	1	1
500-600	1	0.99	0.99	0.99	0.99	0.99	0.99
600-700	1	1	1	1	1	1	1
700-800	0.99	0.99	0.99	0.99	0.99	0.99	0.99
800-900	0.99	0.99	0.99	0.99	0.99	0.99	0.99
900-100	1	1	1	1	1	1	1
Training Time	0.806	7.974	23.993	56.48	99.828	159.71	198.07
Elaspsed time	0.075	0.092	0.154	0.078	0.102	0.094	0.0899



Neural Networks

- Process samples one by one
- Compare result to actual label
- Errors are from classification are used to make modifications
- Backwards prorogation, tuning

Neural Networks	Accuracy						
Training Set Size	1k	5k	10k	20k	30k	40k	50k
0-100	1	1	1	1			
100-200	1	1	1	1			
200-300	1	1	1	1			
300-400	1	1	1	0.99			
400-500	1	1	1	1			
500-600	1	0.99	0.99	0.99			
600-700	1	1	1	1			
700-800	0.99	0.99	0.99	0.98			
800-900	0.99	0.99	1	1			
900-100	1	1	1	1			
Training Time	56.315	435.316	1506.819	10349.03			
Elaspsed time	0.026	0.105	0.25	0.15			



- Slowest training I ever did see
- Fast classification
- Great results

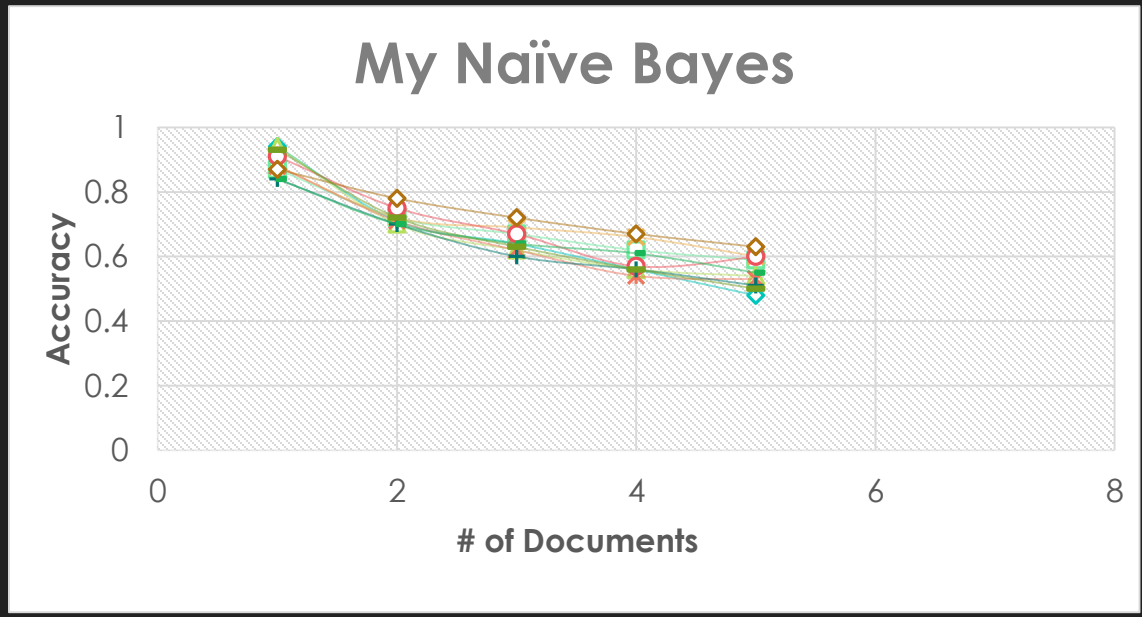
My Naïve Bayes

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}$$

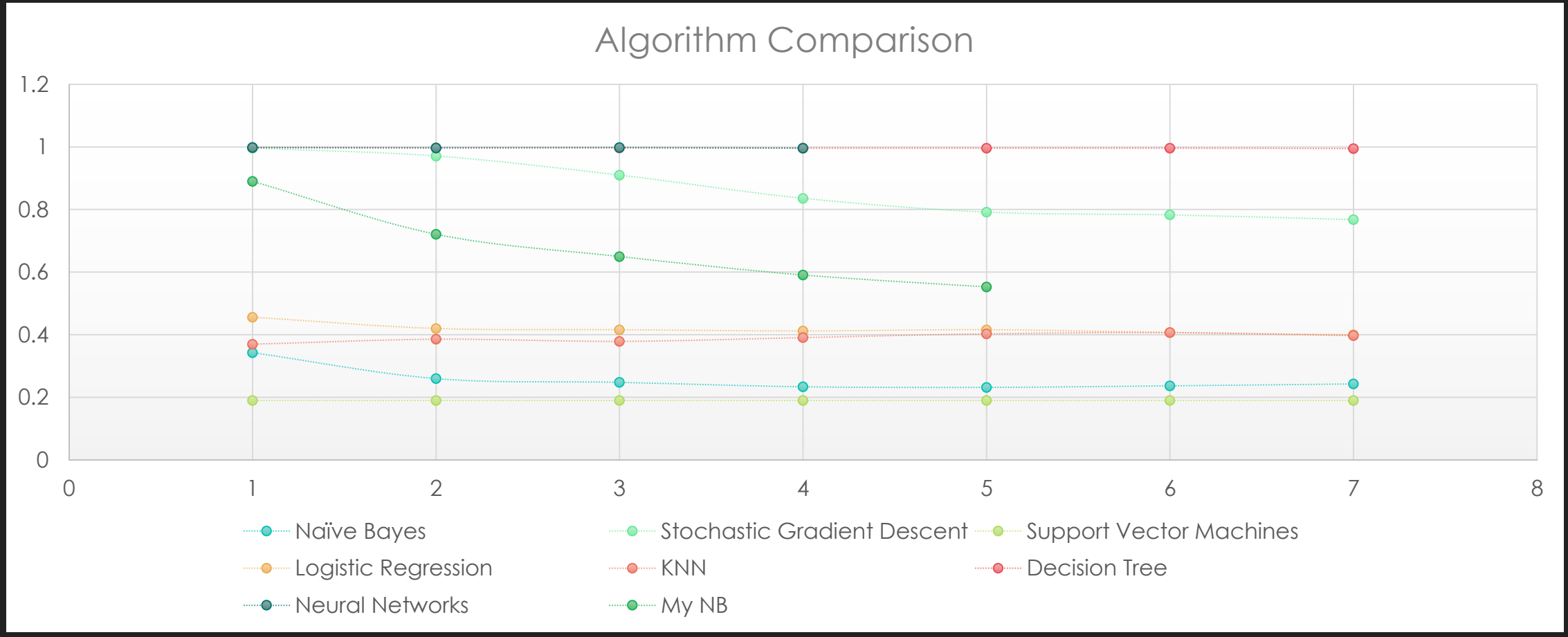
- Made by me
- Bag of words method

- Fast training
- SLOW classification
- Good then terrible results

MyNB	Accuracy						
Training Set Size	1k	5k	10k	20k	30k	40k	50k
0-100	0.94	0.71	0.64	0.56	0.48		
100-200	0.87	0.72	0.67	0.62	0.59		
200-300	0.94	0.7	0.62	0.56	0.54		
300-400	0.88	0.72	0.69	0.66	0.6		
400-500	0.88	0.71	0.62	0.54	0.53		
500-600	0.91	0.75	0.67	0.57	0.6		
600-700	0.84	0.7	0.6	0.56	0.51		
700-800	0.84	0.7	0.64	0.61	0.55		
800-900	0.93	0.72	0.63	0.56	0.5		
900-100	0.87	0.78	0.72	0.67	0.63		
Training Time	0.16	0.944	2.713	14.99	24.331		
Elaspsed time	132.059	877.13	2418.075	3427.908	41635.77		



Algorithm Comparison



My NB In Action

```
merry christmans ==> (u'17', 1.6138699208126408e-09)
hapy 4th of july ==> (u'11', 2.848003624086619e-18)
i hate you ==> (u'2', 1.1685893961947471e-14)
i love you ==> (u'0', 7.436597389420465e-13)
```

🎄
🇺🇸
😂
❤️

Emoji Mapping

```
0  🤍 _red_heart_
1  😍 _smiling_face_with_hearteyes_
2  🥲 _face_with_tears_of_joy_
3  💞 _two_hearts_
4  🔥 _fire_
5  😊 _smiling_face_with_smiling_eyes_
6  😎 _smiling_face_with_sunglasses_
7  ✨ _sparkles_
8  💙 _blue_heart_
9  😘 _face_blowing_a_kiss_
10 📷 _camera_
11 us  _United_States_
12 ☀ _sun_
13 💜 _purple_heart_
14 😉 _winking_face_
15 💯 _hundred_points_
16 😁 _beaming_face_with_smiling_eyes_
17 🎄 _Christmas_tree_
18 📸 _camera_with_flash_
19 😜 _winking_face_with_tongue_
```