**CS 145 HW6**

**Allen (Akikazu) Miyazawa**

**1.**

**a.**

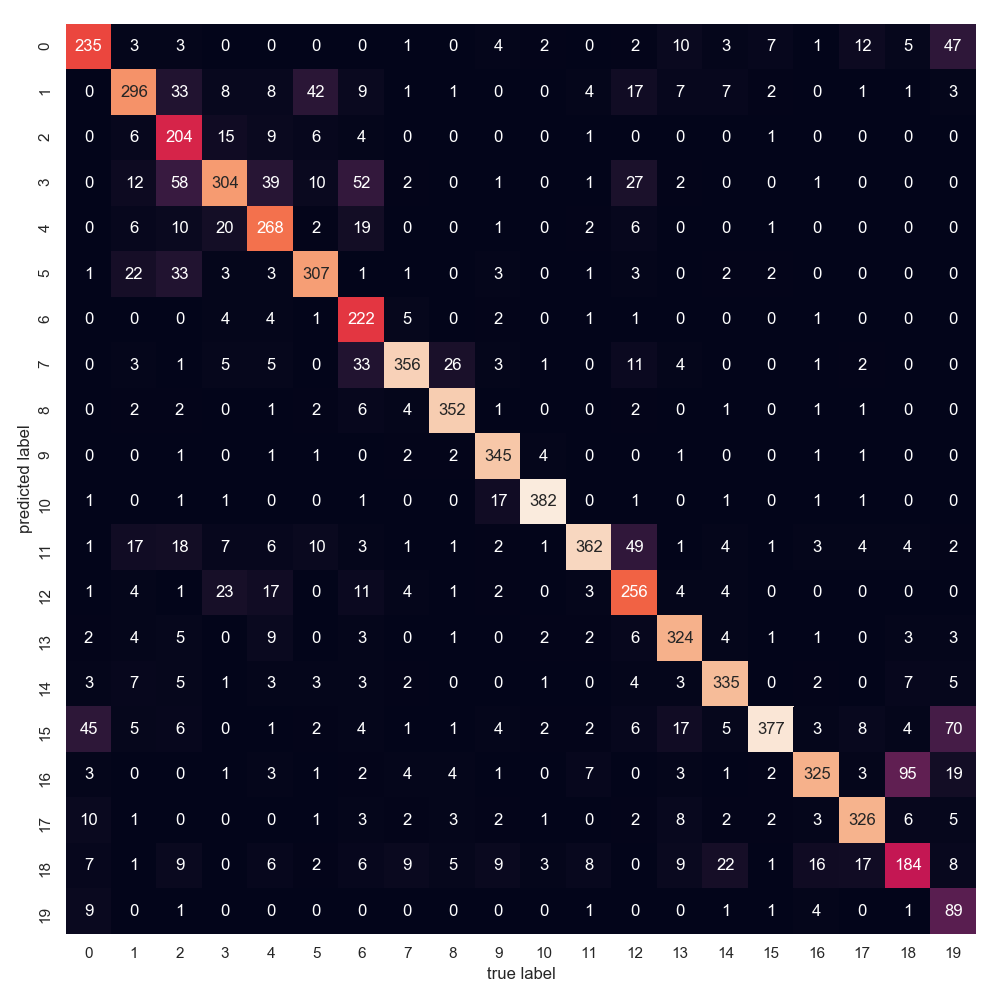
Completed (See code)

**b.**

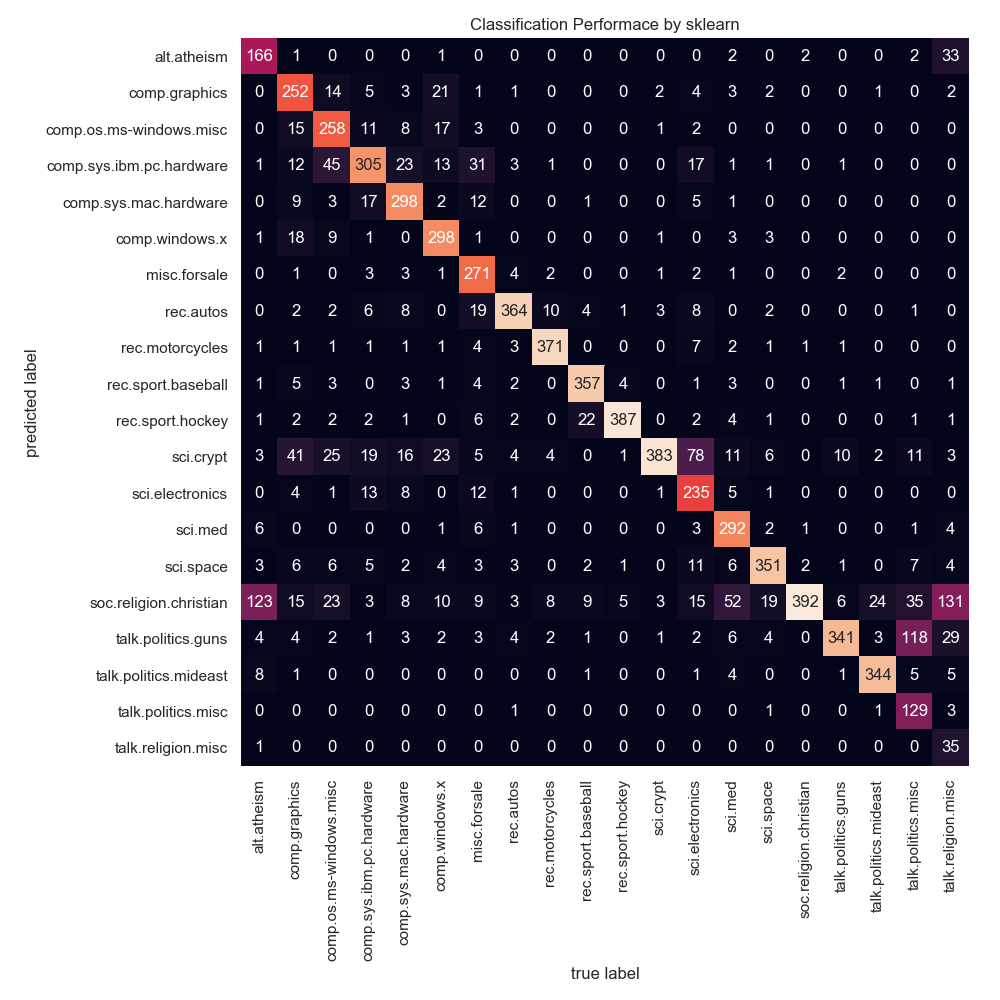
**Table 1: Report accuracy for Naive Bayes Model**

|  |  |  |
| --- | --- | --- |
|  | **Train set accuracy** | **Test set accuracy** |
| **Sklearn implementation** | 0.932649814389 | 0.77389803505 |
| **My implementation** | 0.941077291685 | 0.779347101932 |

**Classification matrix of my implementation:**



**Classification matrix of sklearn:**



**Table 2: Incorrect Example: Document 3**

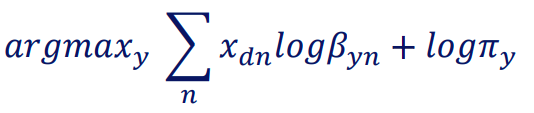
|  |  |  |
| --- | --- | --- |
| **Words (count) in the example document** | **Predict label** | **Truth label** |
| of(3), from(1), and(3), other(1), are(1), etc… (abbreviated to due its length) | 16 | 1 |

**c.**

Naïve Bayes is a generative model because it explicitly models the joint probability distribution p(x,y) and then uses Bayes rule to compute p(y|x). A discriminative model, in contrast, directly models p(y|x). In other words, a generative model like Naïve Bayes models the joint distribution of the feature x and target y and then predicts the posterior probability of P(y|x), and a discriminative model, like logistic regression directly models the posterior probability of p(y|x) by learning the input to output mapping by minimizing error. In terms of model assumption, Naïve Bayes assumes all features are conditionally independent (so it does not work as well when the features are dependent), but logistic regression splits feature space linearly (works well even when some variables are correlated. Naive Bayes for task classification task is good because it is computationally fast, simple to implement, works well with high dimensions, and not sensitive to irrelevant features. A disadvantage is that it relies on the independence assumption (assuming features are independent of each other), so when features are actually dependent, performance may not be as good.

**d.**

Yes, we can apply Naïve Bayes model to identify spam emails from normal ones. Let there be 2 classes, 0 and 1, representing not spam and spam. So the prior probabilities are p(h=1) = p and p(h=0) = 1-p. We have a collection of documents (emails) that are labeled 0 or 1 (not spam or spam). Ultimately, we try to maximize the following equation:



Where 𝑥dn is the number of words for nth word in the vocabulary, Byn parameter of Bernoulli distribution for each word in the sequence, and piy is the prior probability of class y.

Essentially, the only main difference between the model we build for this homework and a model that classifies spam is that the homework model has 20 classes while a spam model has 2 classes.

**2.**

**a.**

Completed (See code)

**b.**

I chose K = 6 and think it is reasonable because topics are strongly correlated at K = 6.

**Table 3: Topics words**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset 1 | | | | | |
| Topic 1 | Topic 2 | Topic 3 | Topic 4 | Topic 5 | Topic 6 |
| luffy crew dressrosa straw pirates franky flame government robin hat | luffy pirates island pose captain magnetic log roger set grand | sea grand mountain bur red blue baroque alabasta called series | luffy crew ace navy animals sabaody archipelago den whitebeard rayleigh | pirates haki island piece color treasure fishman straw crew king | devil fruit user sea fruits manga zou luffy powers water |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset 2 | | | | | |
| Topic 1 | Topic 2 | Topic 3 | Topic 4 | Topic 5 | Topic 6 |
| bank company jewish time congress president season iraq florio east | bush official noriega president campaign scientists magellan peres people summit | percent prices company government duracell union oil california york thursday | people city dukakis fire rating roberts police monday administration farmer | soviet barry children polish waste nelson people moore police war | percent gorbachev soviet people rose economy national month rate manufacturing |

**c.**

GMM, where the models are multivariate Gaussian distribution, uses EM algorithm. pLSA is another EM algorithm. Both uses parameters and a set of weights to update the parameters. pLSA is more flexible than GMM since a different set of mixture weights is set for each data point while GMM uses the same set of mixture weights for all data points in a certain class.

**d.**

pLSA is not a well-defined generative model, so there is no way of generalizing to new, unseen documents. Also, the number of parameters grows linearly with the size of training documents. The linear growth in parameters suggests that the model is prone to overfitting.