# CS 440: naïve Bayes Classification

## Part 1: Digit Classification

### Single Pixels as Features

Our implementation is quite simple. We created an object that would hold matrices for the binary 0 (white) and 1 (black) for each class. These matrices would be the size of an image. This meant that when we encountered a feature, we could simply find that exact location in a specific matrix and increment the count. For example, if we were looking at pixel (1,1) in a training image that represented the digit class 5 and the pixel was black, we would go to the black matrix for class 5 and increment its count. We followed this procedure for every feature of every training image. After doing this, we compared the testing data against the populated matrices for every single class, using the method described in the Assignment handout. For each testing image, we calculated its posterior probability for each class. We labelled it using the highest posterior probability. Our results are shown below.

We chose to use a smoothing factor of 1. This means that we assume that we have seen every value once more than its actual occurrence value. This made our results more accurate than using no smoothing factor. We chose not to use a larger smoothing factor; we did not want to over-count occurrences by more than 1. Using a smoothing factor of 1 means that if a particular feature never occurred for a given class, we could still operate under the assumption that it did occur once. This allows uncommon features to still be represented. Using a larger smoothing factor would have overrepresented features that do not occur very often. We did, however, build functionality to play around with smoothing factor. We made it possible to vary smoothing factor very easily. In our implementation, smoothing factor can be specified, which allows us to test different smoothing factors very easily. We found the best results with a smoothing factor of 1, as per our hypothesis. We want to represent all features, even if they occur uncommonly, but we do not want to over-represent uncommon features, which larger smoothing factors would do.

#### Results, as referenced above

Our overall accuracy rate for the digit data set is 77.1%

Below are the classification rates for each digit. The format is as follows:

*(digit class, classification rate – truncated after 3 decimal points)*

(0, 0.844)

(1, 0.962)

(2, 0.776)

(3, 0.790)

(4, 0.766)

(5, 0.673)

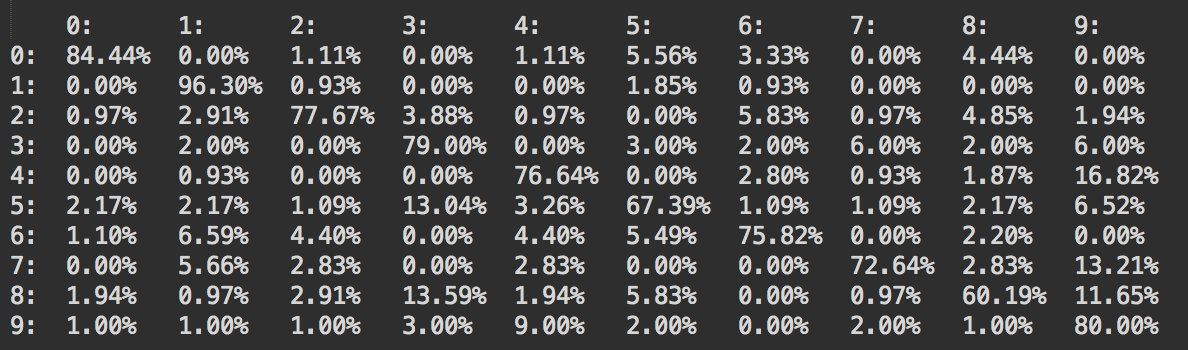
(6, 0.758)

(7, 0.726)

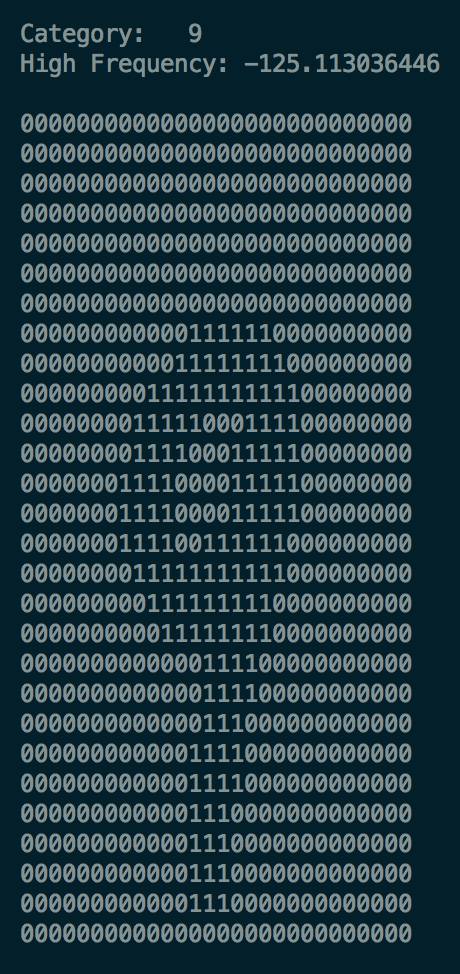
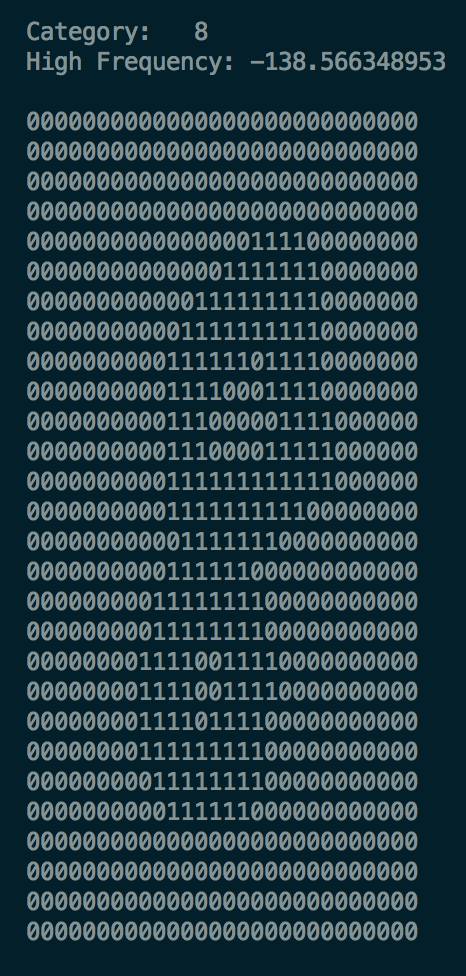
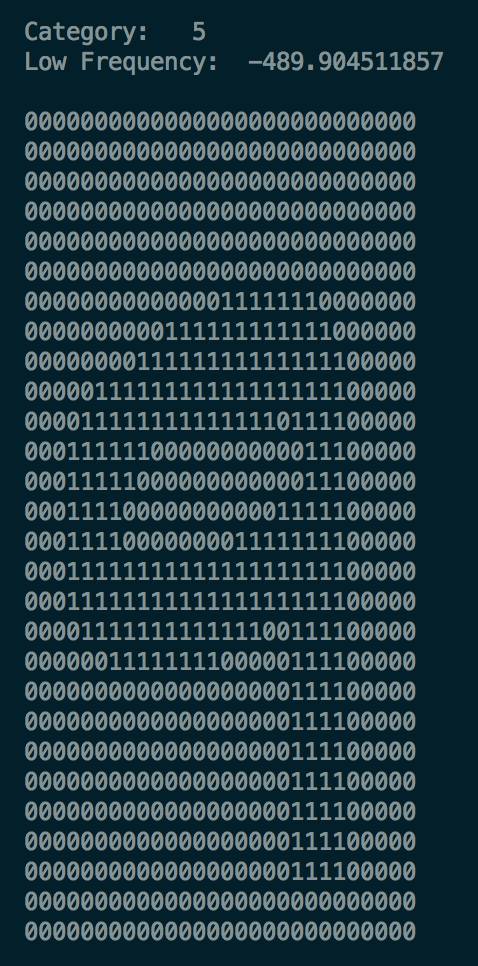
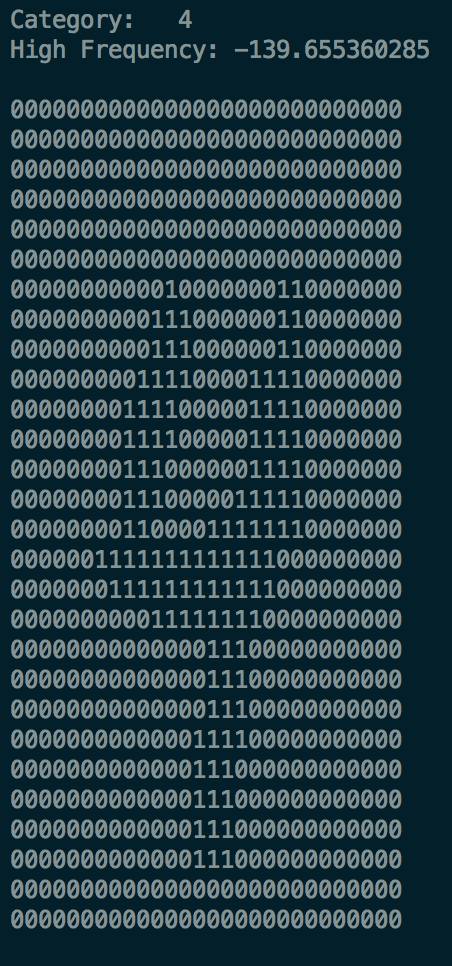
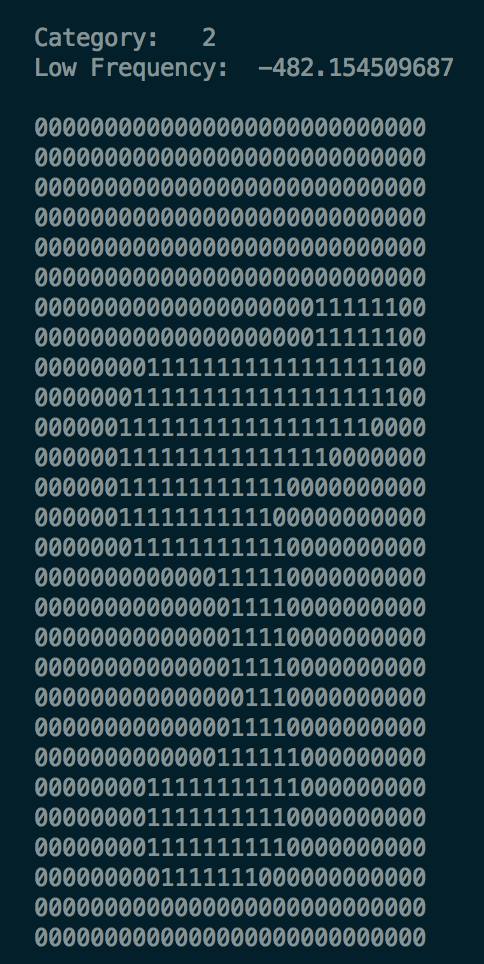
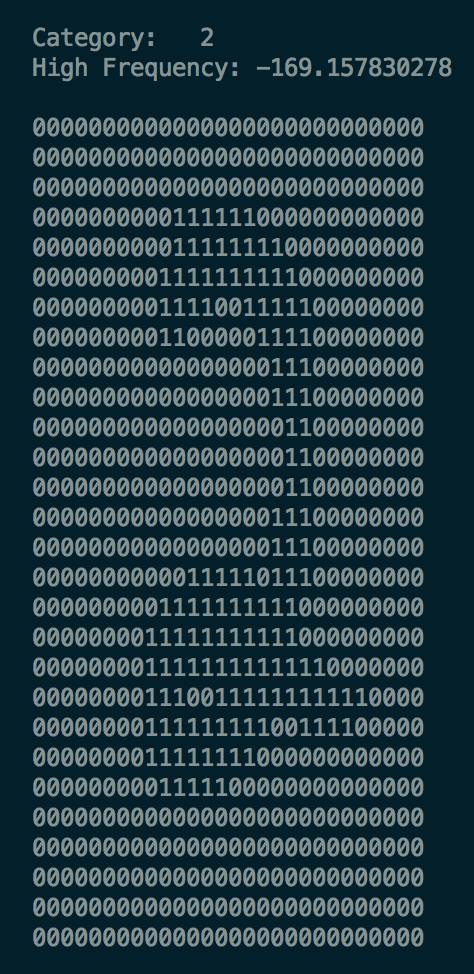
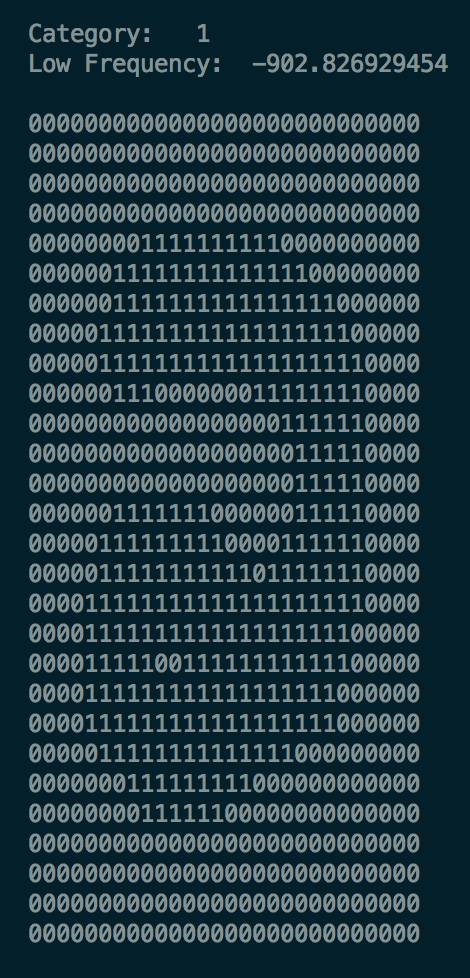
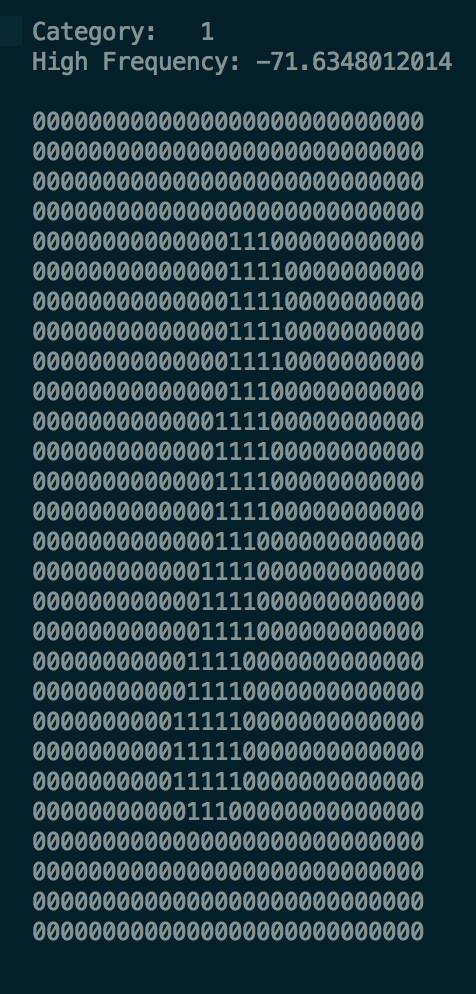
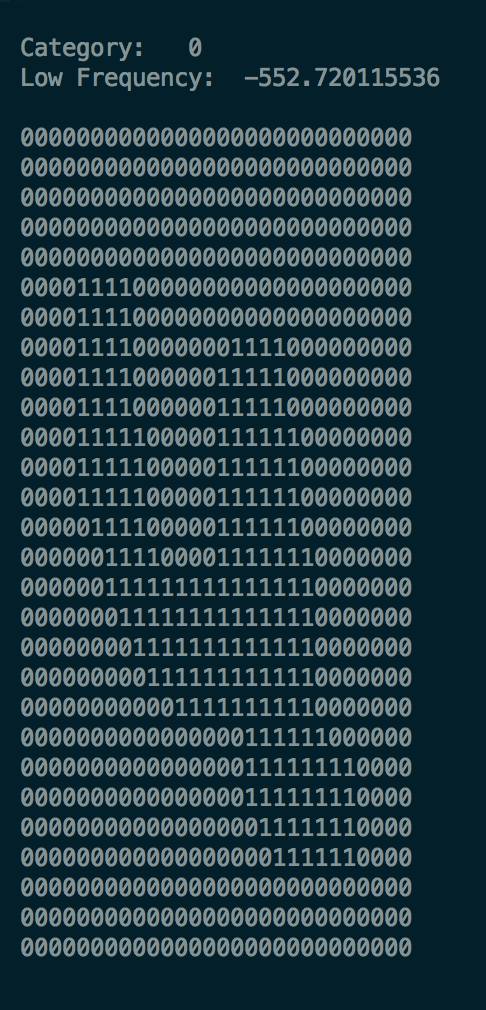
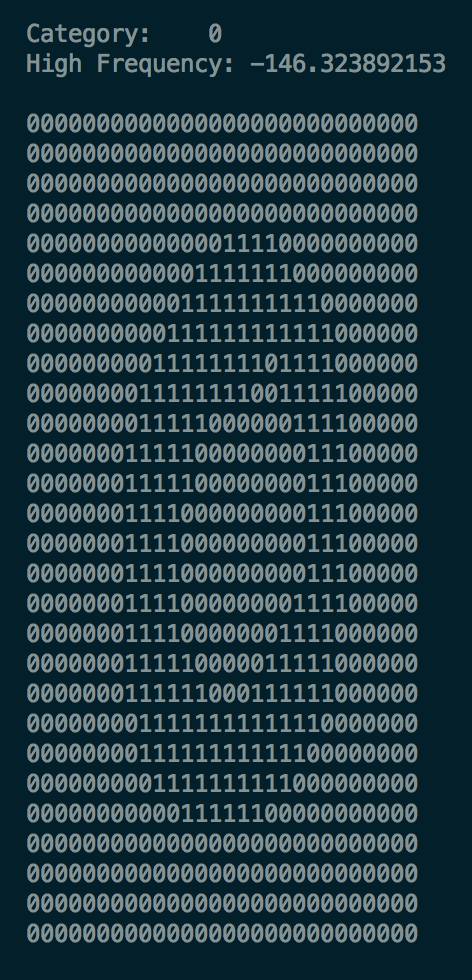
(8, 0.601)

(9, 0.800)

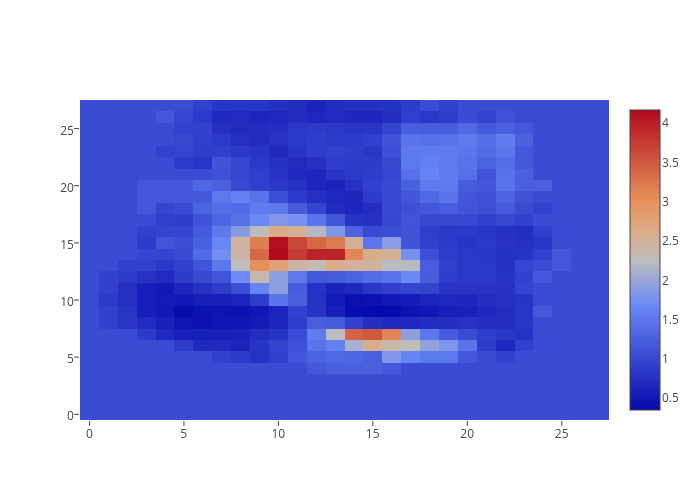
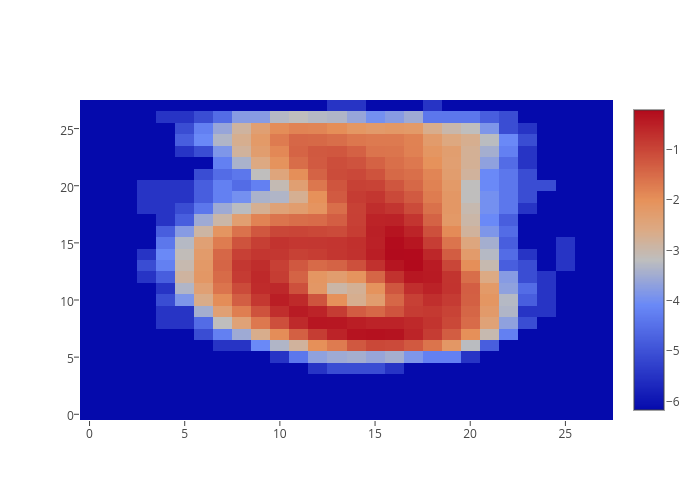
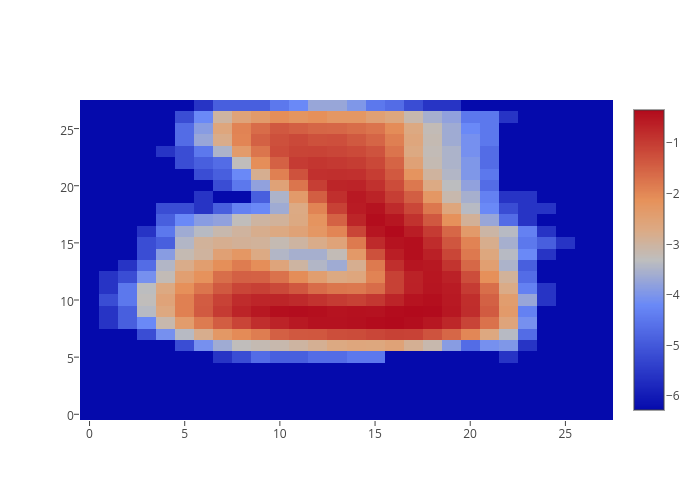
Below is the confusion matrix. Percentages are truncated after 2 decimal points.



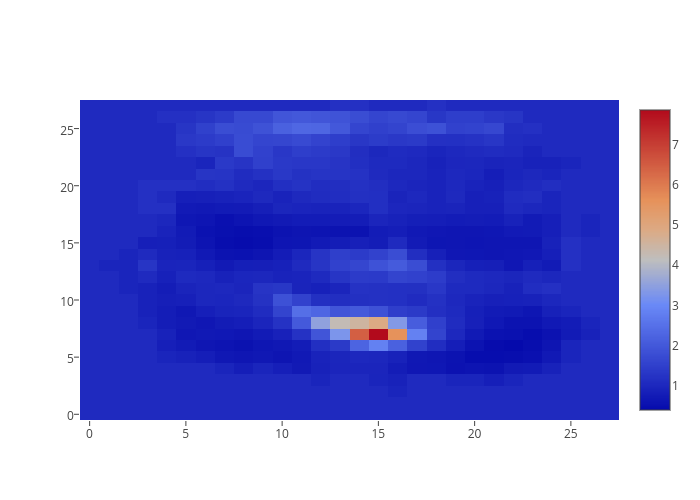
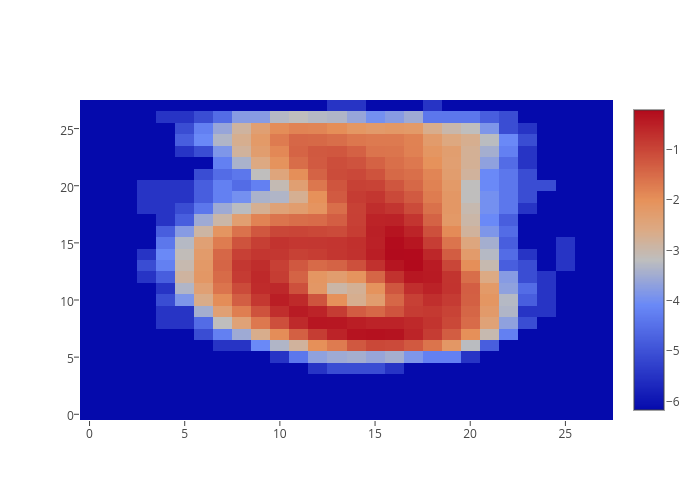
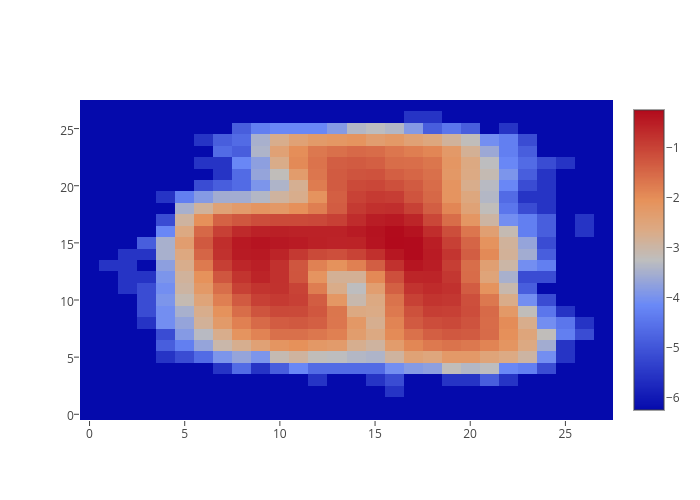
Below are the test examples with the highest and lowest posterior probabilities for each digit class. White regions are represented as zeros and colored regions are represented as ones. See below.



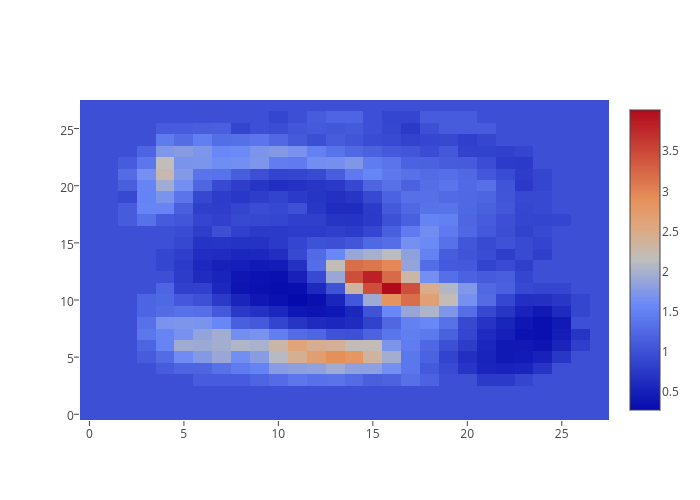
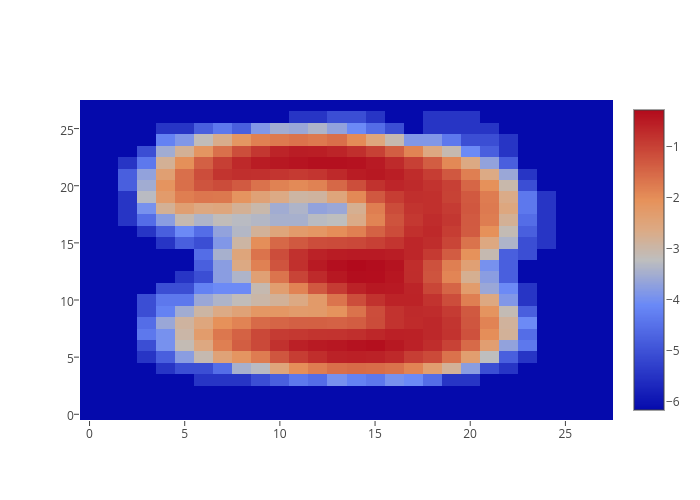
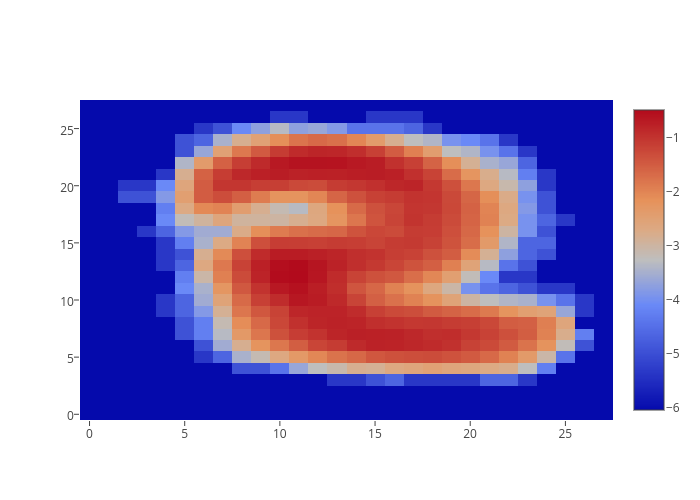
We also looked at odds ratios for the following sets of numbers: [(7,9),(4,9),(5,3),(8,3)]. The results are below, shown as heat maps of the log likelihoods.



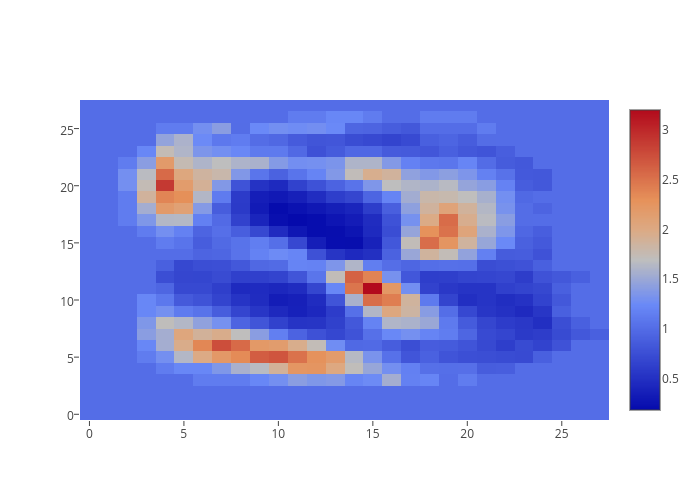
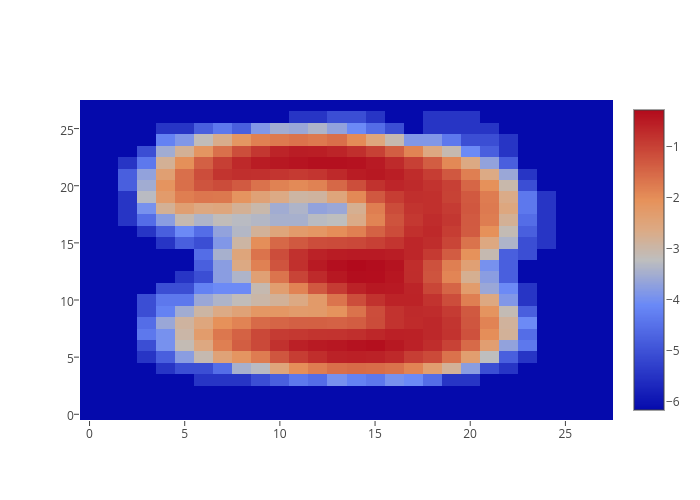
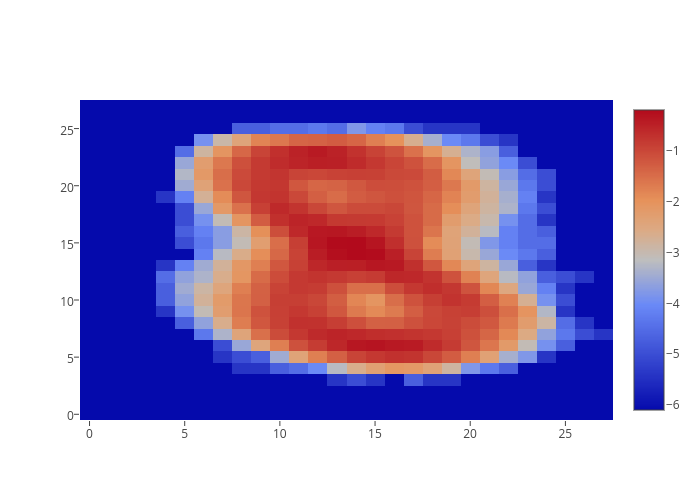
7 9 7 over 9



4 9 4 over 9



5 3 5 over 3



8 3 8 over 3

### Extra Credit for Part 1

We experimented with using ternary features. This is for bonus points. Our implementation was similar, but we had matrices to represent white, gray, and black rather than just white and black. Our results reflected a small improvement.

Our overall accuracy improved to 77.6%.

Below are the classification rates for each digit. The format is as follows:

*(digit class, classification rate – truncated after 3 decimal points)*

(0, 0.833)

(1, 0.953)

(2, 0.766)

(3, 0.800)

(4, 0.775)

(5, 0.684)

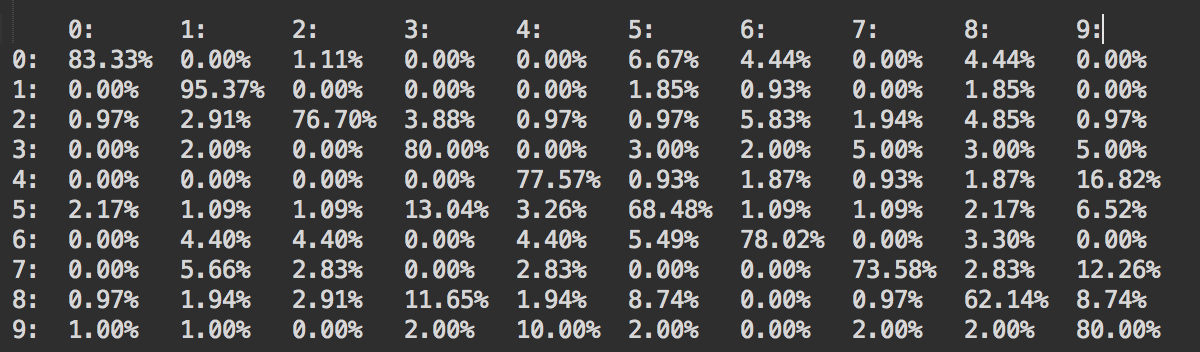
(6, 0.780)

(7, 0.735)

(8, 0.621)

(9, 0.800)

The confusion matrix is shown below.



It appears that ternary features did not make a significant improvement upon classification. However, we do believe that weighting gray values differently could be worth further exploration.

We applied our Naïve Bayes Classifier to the face data set. The implementation was very similar to the digit data set, except that there were only two classes and that there were more features per image. This is for bonus points.

Our overall accuracy was 90.6%.

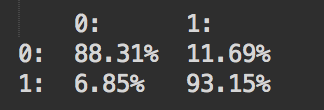
Below is the classification rate for each class. The format is as follows, with 0 being not face and 1 being a face:

*(face class, classification rate – truncated after 3 decimal points)*

(0, 0.883)

(1, 0.931)

The confusion matrix is shown below.



This whole section was for bonus points.

## Part 2: Document Classification

### 2.1: For Three-Unit Students

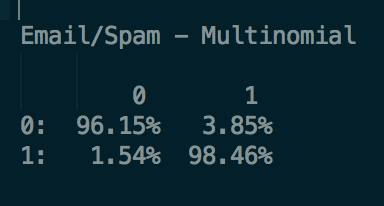
#### Results

##### SPAM DATA SET

###### Multinomial

Accuracy 97.30769%

Confusion matrix is below.



Top 20 Words for Normal Email

language

university

s

linguistic

de

information

conference

workshop

email

paper

e

english

one

please

include

edu

http

research

abstract

address

Top 20 Words for Spam Email

email

s

order

report

our

address

mail

program

send

free

money

list

receive

name

business

one

d

work

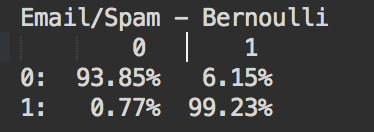
com

nt

###### Bernoulli

Accuracy 96.53846%

Confusion matrix is below.



Top 20 Words for Normal Email

language

university

s

information

linguistic

http

email

please

e

follow

fax

include

one

english

call

research

www

word

address

interest

Top 20 Words for Spam Email

our

s

free

please

email

mail

one

address

list

com

receive

http

us

send

day

information

remove

here

over

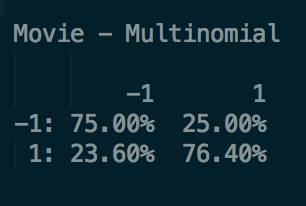
want

##### MOVIE DATA SET

###### Multinomial

Accuracy 75.70000%

Confusion matrix is below.



Top 20 Words for Negative Reviews

movie

film

like

one

--

bad

story

much

time

even

characters

good

little

would

comedy

never

nothing

makes

plot

make

Top 20 Words for Positive Reviews

film

movie

--

one

like

story

good

comedy

way

even

time

best

much

performances

funny

make

life

us

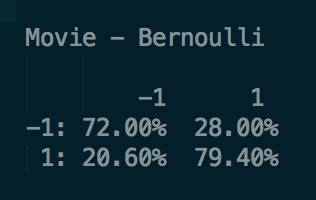
makes

characters

###### Bernoulli

Accuracy 75.70000%

Confusion matrix is below.



Top 20 Words for Negative Reviews

movie

film

like

one

story

much

--

bad

time

even

characters

little

good

would

comedy

nothing

makes

plot

never

make

Top 20 Words for Positive Reviews

film

movie

one

like

--

story

comedy

way

even

good

best

time

much

performances

funny

makes

life

make

characters

work

### Part 2.2: For Four-Unit Students (Extra Credit for Us)

Though we are three-unit students, we attempted the four-unit problem. This is for bonus points.

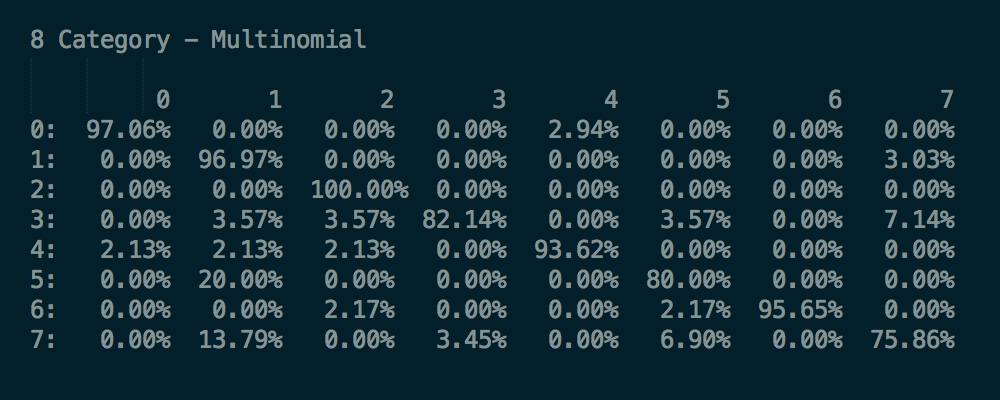
#### Results

##### 8 CATEGORY DATA SET

###### Multinomial

Accuracy 92.01521%

Confusion matrix is below.



Top 20 Words for sci.space

space

nt

would

one

launch

nasa

earth

subject

like

us

system

also

writes

could

time

data

first

orbit

edu

mission

Top 20 Words for comp.sys.ibm.pc.hardware

drive

scsi

nt

ide

one

card

drives

controller

system

disk

subject

use

would

edu

hard

bus

get

m

data

also

Top 20 Words for rec.sport.baseball

nt

would

year

edu

writes

one

game

good

team

subject

last

article

think

players

like

baseball

games

better

well

time

Top 20 Words for comp.windows.x

x

window

use

nt

subject

file

server

also

available

get

edu

motif

version

system

program

sun

c

one

m

windows

Top 20 Words for talk.politics.misc

nt

would

people

q

one

mr

think

writes

president

article

government

stephanopoulos

know

us

edu

like

subject

going

right

get

Top 20 Words for misc.forsale

new

edu

dos

sale

appears

art

subject

wolverine

shipping

cover

price

one

list

comics

drive

nt

hulk

good

vs

system

Top 20 Words for rec.sport.hockey

nt

game

team

hockey

would

play

subject

period

season

nhl

games

one

first

year

think

players

get

la

edu

like

Top 20 Words for comp.graphics

image

jpeg

edu

nt

file

images

data

also

graphics

software

available

use

one

program

files

format

get

version

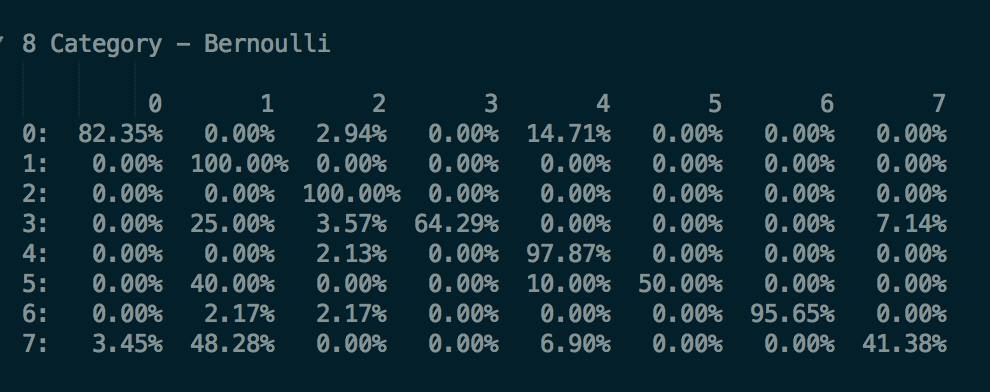
system

ftp

###### Bernoulli

Accuracy 84.41065%

Confusion matrix is below.



Top 20 Words for sci.space

subject

would

nt

space

writes

article

one

like

could

also

get

think

time

us

new

much

way

m

edu

see

Top 20 Words for comp.sys.ibm.pc.hardware

subject

nt

one

would

writes

use

get

know

article

card

also

like

m

two

system

edu

drive

work

problem

could

Top 20 Words for rec.sport.baseball

subject

nt

writes

article

edu

would

one

last

year

like

baseball

good

think

get

time

m

know

game

team

first

Top 20 Words for comp.windows.x

subject

x

nt

use

writes

get

window

article

using

one

like

would

also

problem

know

code

m

set

email

help

Top 20 Words for talk.politics.misc

subject

nt

writes

article

people

would

one

like

edu

us

even

think

m

get

government

could

know

make

time

much

Top 20 Words for misc.forsale

subject

sale

edu

new

shipping

please

email

price

nt

one

get

condition

like

list

want

used

good

use

sell

etc

Top 20 Words for rec.sport.hockey

subject

nt

team

game

hockey

writes

would

one

article

like

play

first

think

go

get

nhl

year

games

time

last

Top 20 Words for comp.graphics

subject

nt

one

would

writes

also

like

article

graphics

edu

use

get

know

computer

need

could

think

program

m

two

### Extra Credit for Part 2

We did word clouds for every single category for ever dataset. This is for bonus points. They are shown below.

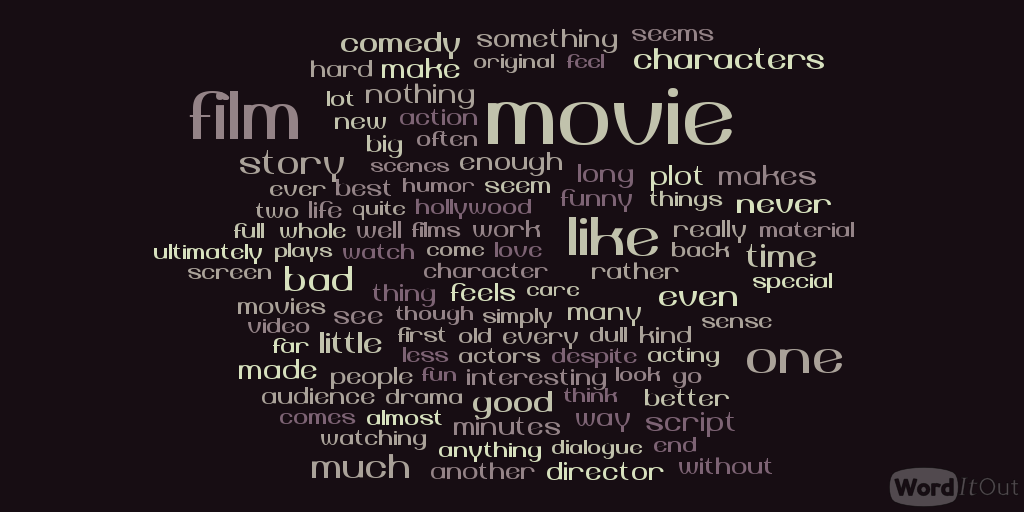
#### Normal Email



#### Spam Email



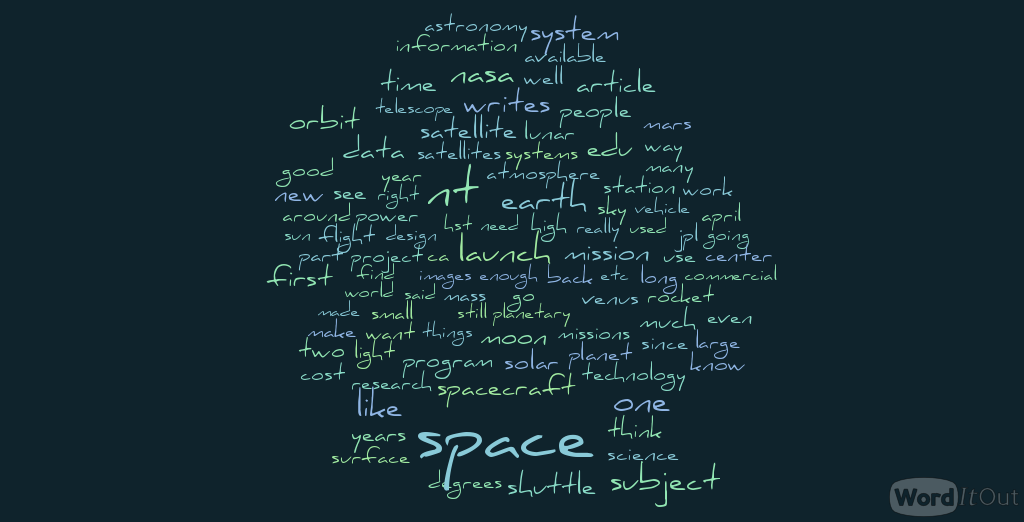
#### Negative Movie Review



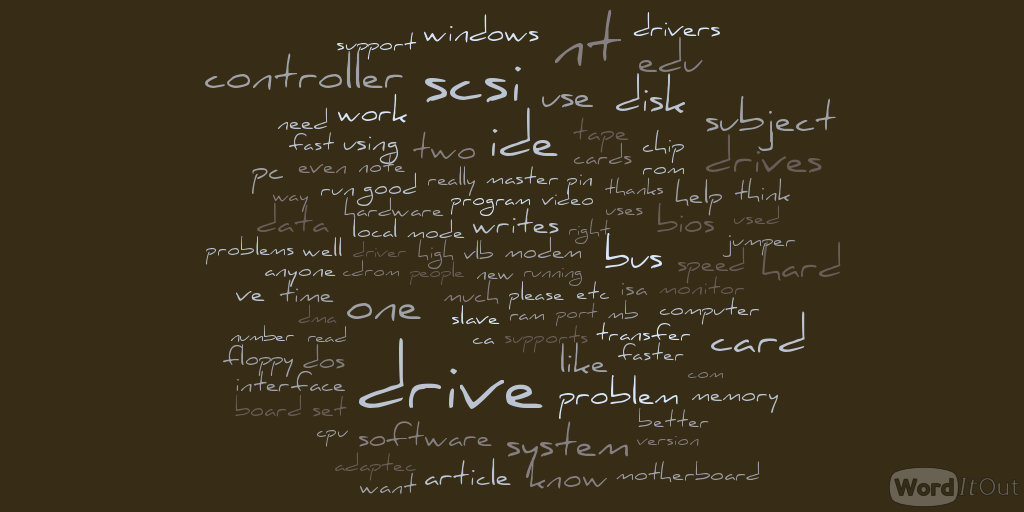
#### Positive Movie Reiview



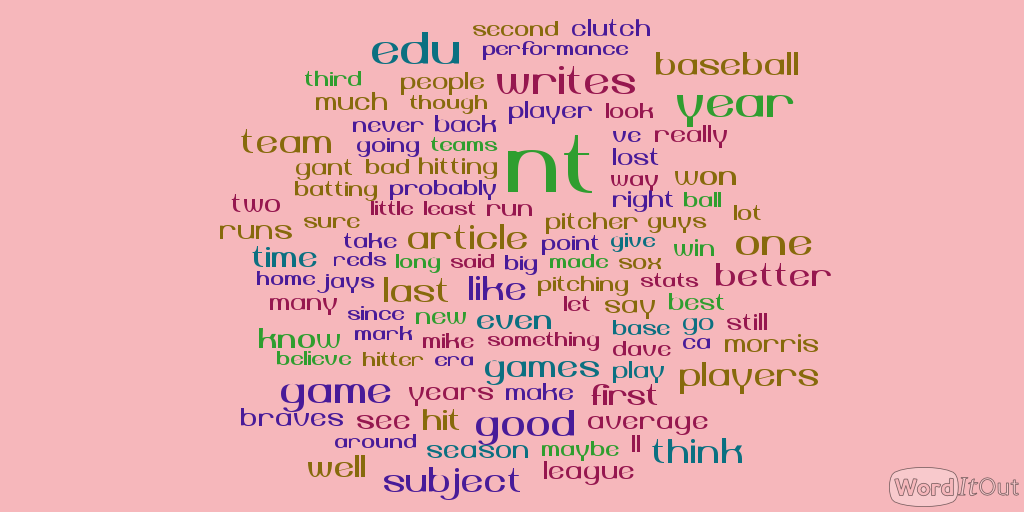
#### sci.space



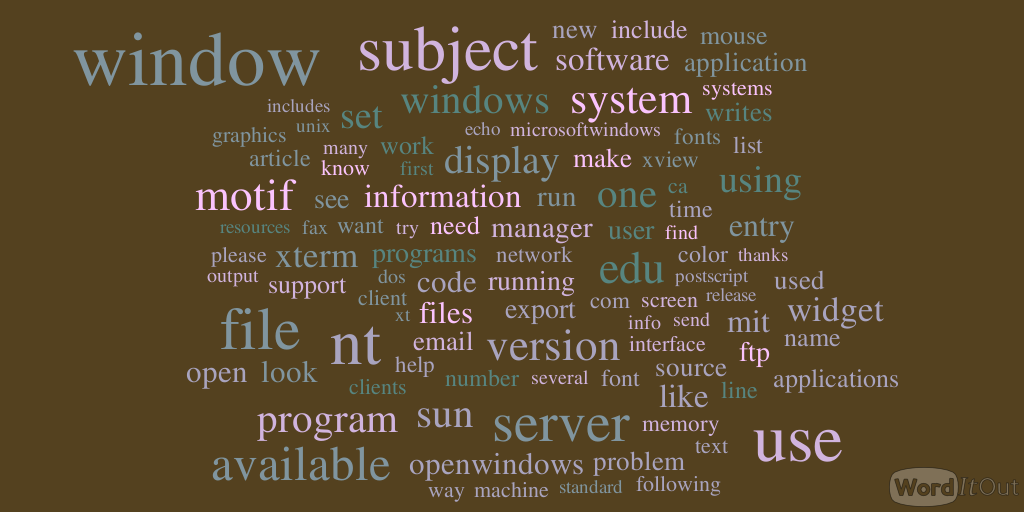
#### comp.sys.ibm.pc.hardware



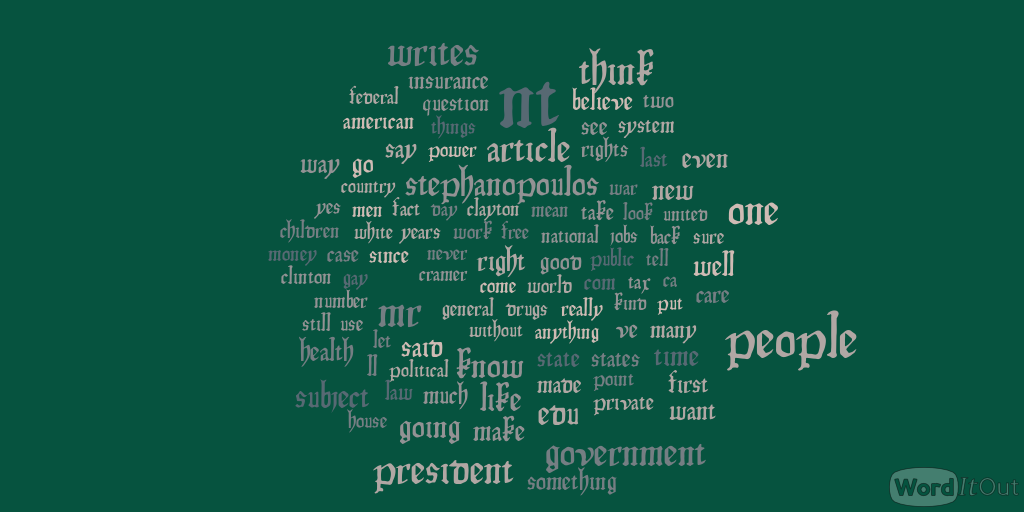
#### rec.sport.baseball



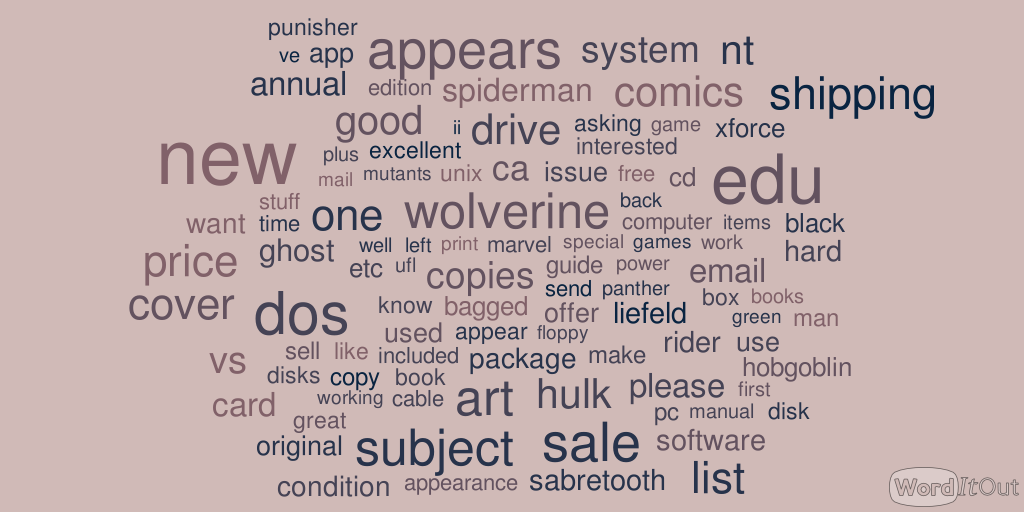
#### comp.windows.x



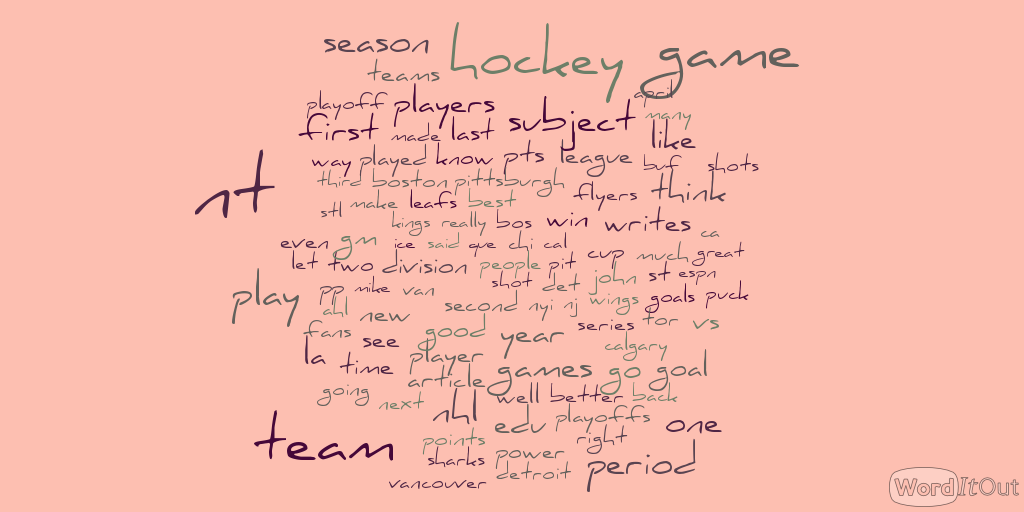
#### talk.politics.misc



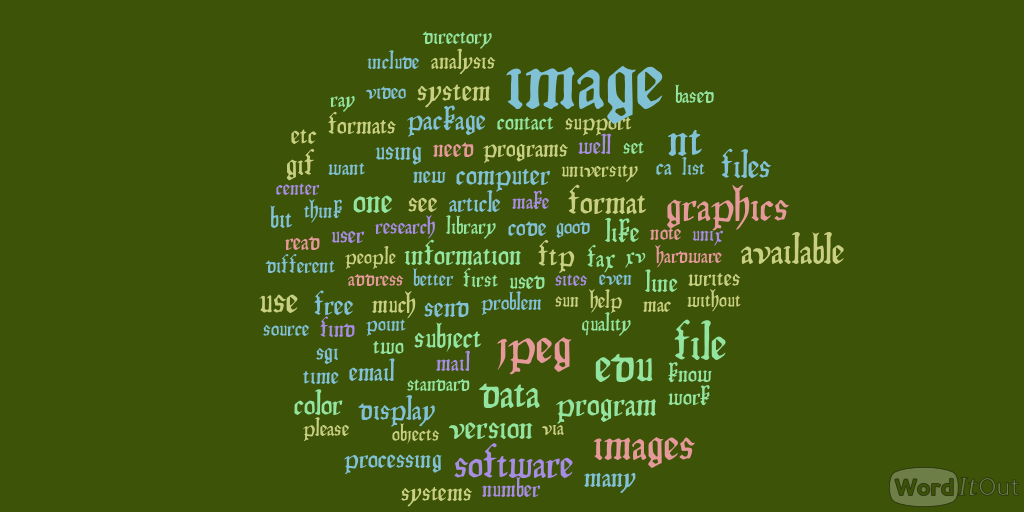
#### misc.forsale



#### rec.sport.hockey



#### comp.graphics



## Individual Contributions

### Abhishek Nigam

I setup the infrastructure for part 1 and developed the training model for part 2. I generated the Top 20 Words list. We worked on the rest together.

### Jakub Klapacz

I setup the infrastructure for part 2 and developed the training model for part 1. I generated the confusion matrices. We worked on the rest together.