Sangha Lee
Serena Chen 67362 – a3

1) Percentage Confusion Matrix:

```
[['94%', '0%', '1%', '0%', '0%', '2%', '0%', '0%', '1%', '1%'],
['0%', '83%', '2%', '2%', '0%', '2%', '4%', '6%', '1%', '1%'],
['1%', '0%', '86%', '1%', '0%', '1%', '4%', '3%', '3%', '0%'],
['0%', '0%', '5%', '71%', '0%', '11%', '0%', '0%', '11%', '2%'],
['0%', '0%', '1%', '0%', '75%', '3%', '6%', '4%', '2%', '9%'],
['7%', '2%', '1%', '5%', '1%', '67%', '5%', '0%', '9%', '2%'],
['5%', '1%', '7%', '2%', '2%', '1%', '80%', '0%', '1%', '0%'],
['0%', '0%', '2%', '4%', '1%', '1%', '0%', '88%', '1%', '2%'],
['4%', '2%', '6%', '3%', '2%', '2%', '3%', '2%', '72%', '2%'],
['0%', '0%', '1%', '4%', '14%', '5%', '0%', '9%', '8%', '61%']]
```

```
The average accuracy performance is 76.7%
```

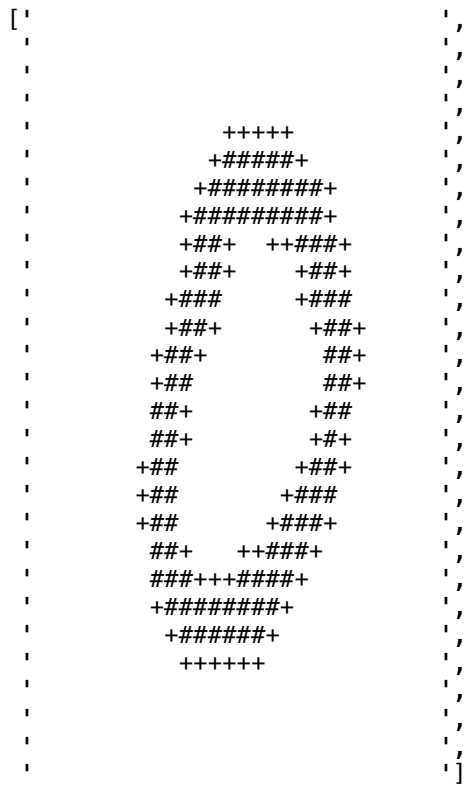
2) Confusion Matrix:

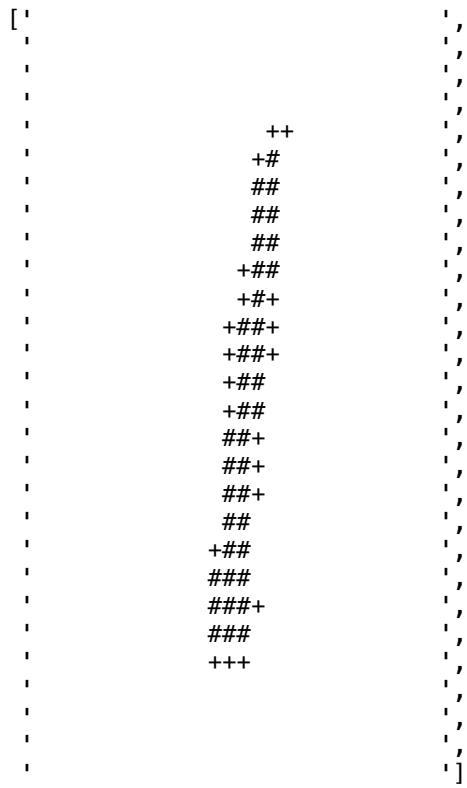
```
[[75, 0, 1, 0, 0, 2, 0, 0, 1, 1],
[0, 103, 3, 2, 0, 2, 5, 7, 1, 1],
[1, 0, 77, 1, 0, 1, 4, 3, 3, 0],
[0, 0, 6, 78, 0, 12, 0, 0, 12, 2],
[0, 0, 1, 0, 83, 3, 7, 4, 2, 10],
[6, 2, 1, 5, 1, 62, 5, 0, 8, 2],
[4, 1, 6, 2, 2, 1, 67, 0, 1, 0],
[0, 0, 2, 4, 1, 1, 0, 78, 1, 2],
[4, 2, 5, 3, 2, 2, 3, 2, 64, 2],
[0, 0, 1, 5, 18, 6, 0, 12, 10, 80]]
```

Overall, 0,1,2,6,and 7 have very high accuracy (above 80%). In comparison, 5 and 9 have low probability. (60%~70%)

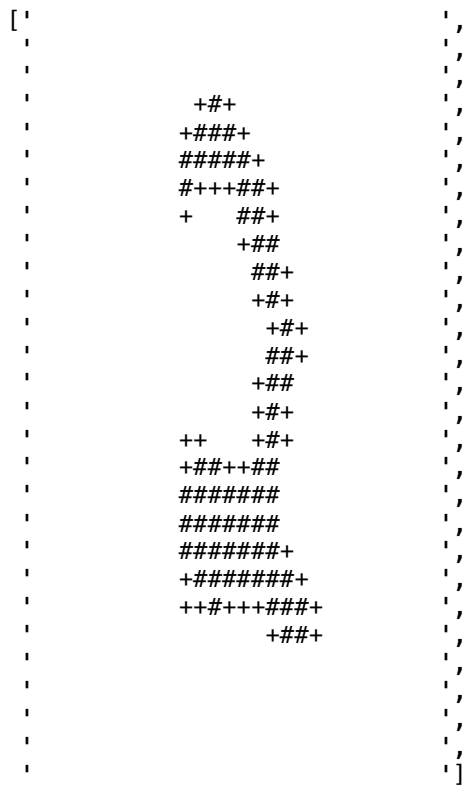
Sangha Lee
Serena Chen

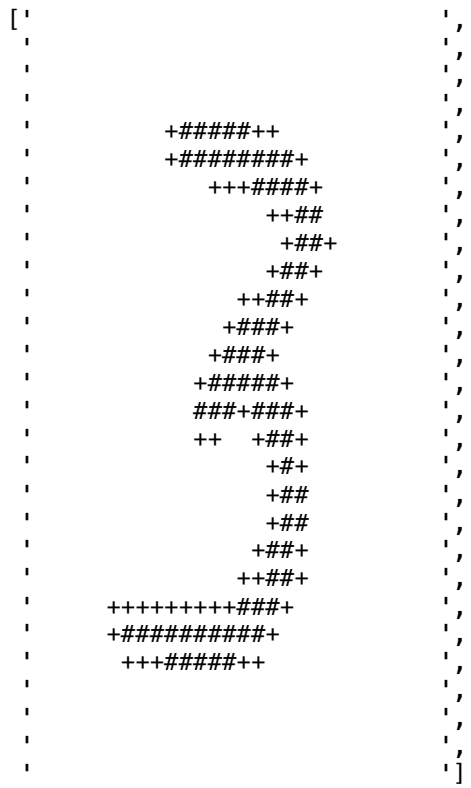
3) list of highest probability images



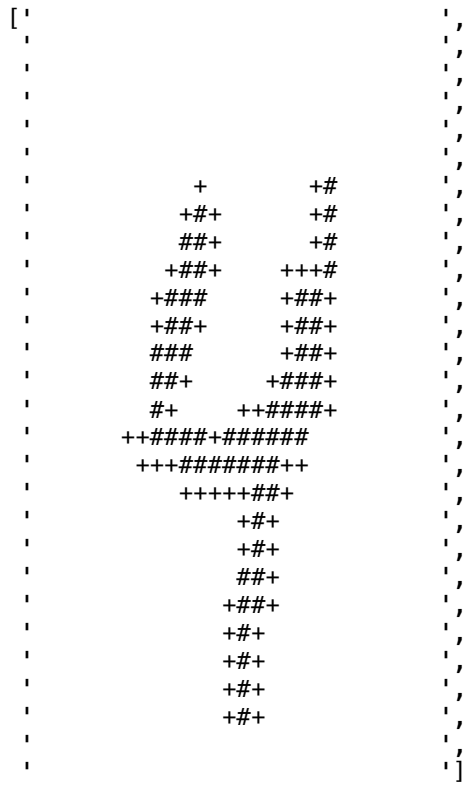


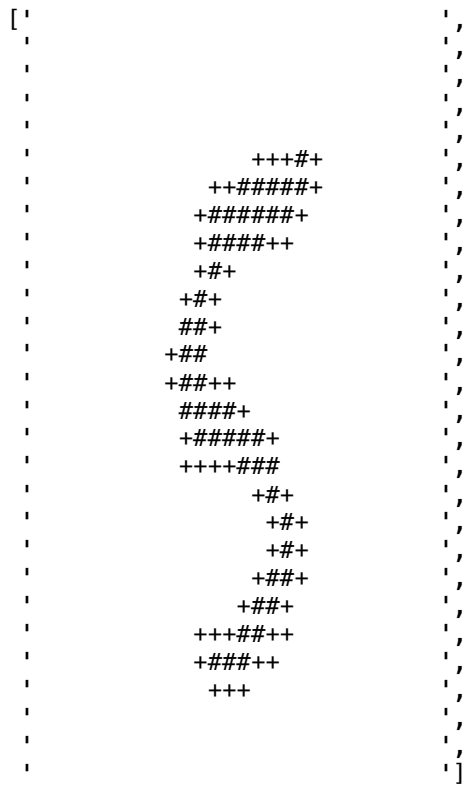
Sangha Lee
Serena Chen



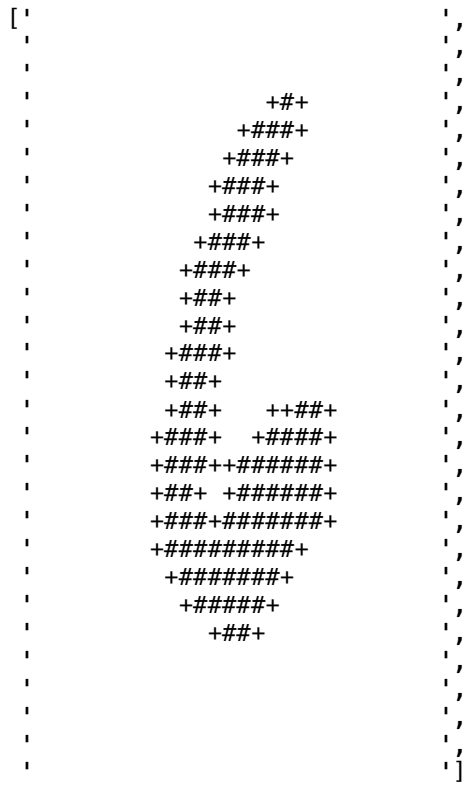


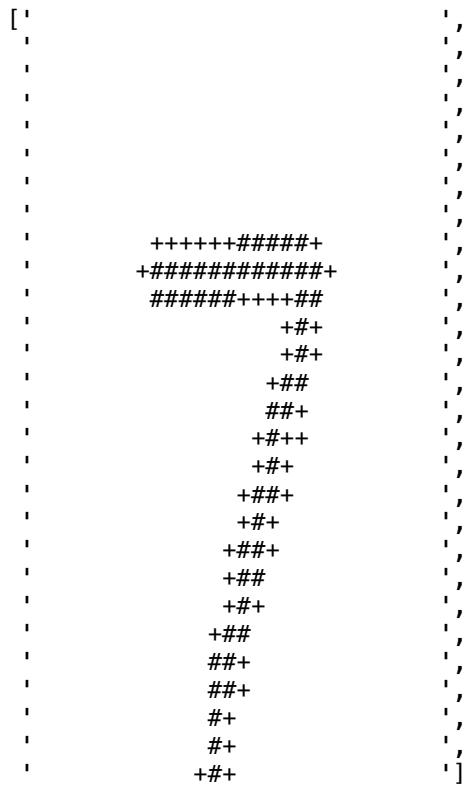
Sangha Lee
Serena Chen



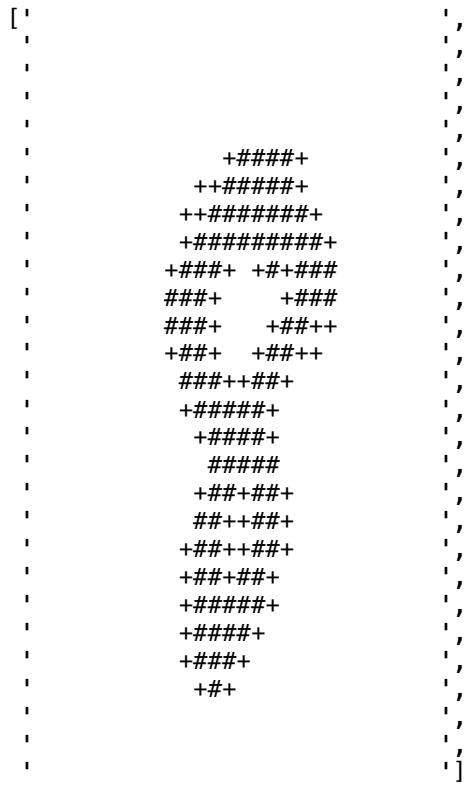


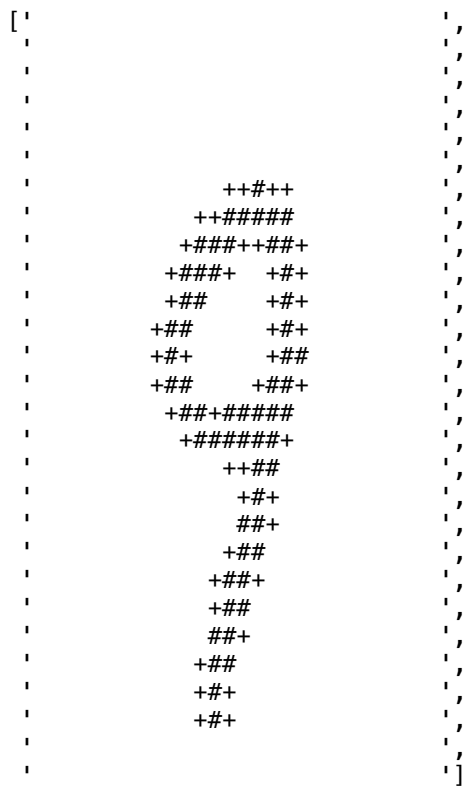
Sangha Lee
Serena Chen





Sangha Lee
Serena Chen





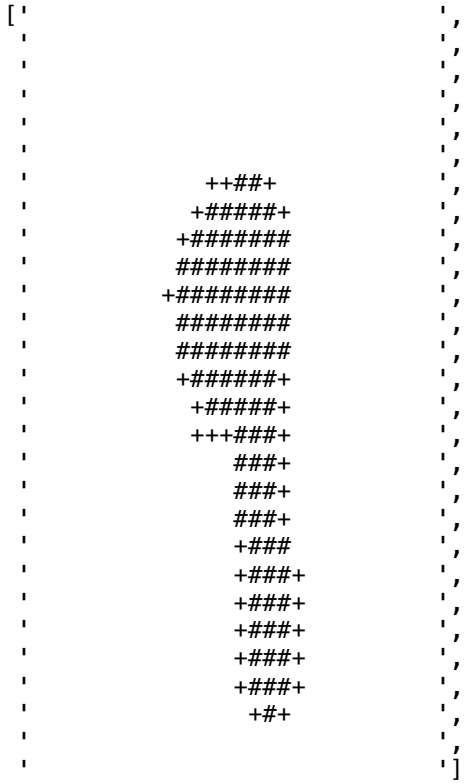
In order obtain the number text images which was the “ideal” image of the category, we found for the each image, the best estimated category and its probability from the classify function in the Naïve Bayes.

Print (best,probs[best])

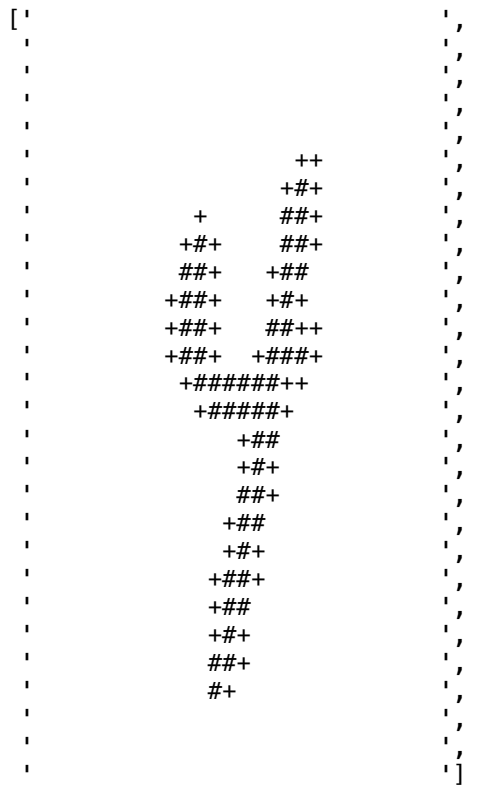
Then, we used the tuple output index to find the image using the show_nth_image_function. The output is displayed above.

4) Interesting Examples

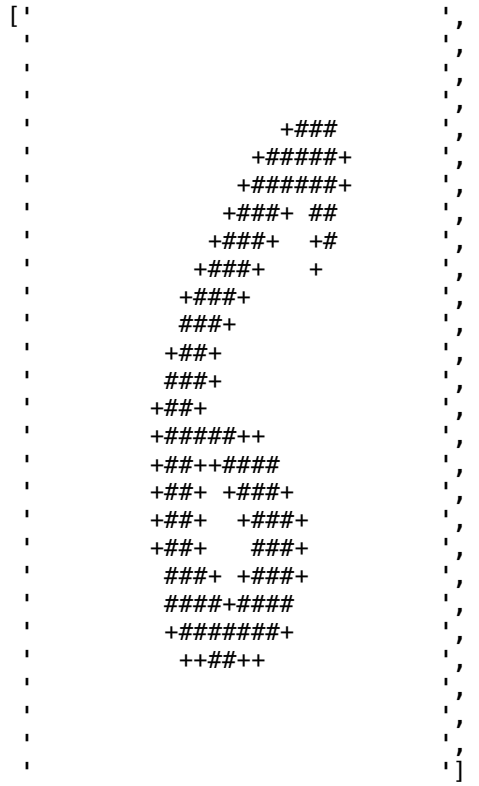
We chose to find some interesting examples from images either classified as 5 or 9 because the two numbers had the lowest accuracy. Then, we compared the correct classification and what the classifier classified. The misclassified images either had features which possibly have tricked the classifier (ex) 6 with longer head similar to head of 5), or had lack of features to be even classified.



Sangha Lee
Serena Chen



(label: 4 classified: 9)



(label: 6 classified: 5)

0,1,2,6,and 7 have very high accuracy (above 80%), and this is due to its distinguishable shape among with other one digit numbers. 5 and 9 have low probability. 5 was hard to be distinguished among other numbers with curved features (ex) 0, 8 and 6). 9 had the lowest probably of 61%. 14% of 9 were misclassified as 4. We examined the 4 and 9 images in the text files and realized that they were indeed very similar, except 9 had more curvature in the front while 4 did not.

Thoughts and Possible Improvements

We tried to improve the accuracy by grouping 4 pixels as a feature. The 9 and 5 still remained low in accuracy. Making the larger grouping can improve accuracy.

Looking at the interesting examples and examining the confusion matrix, we realized that the training could be improved when the classifier can distinguish the curve and the straight line. For example, 4 and 9 were unable to be distinguished easily due to their only difference in the curvature. Instead of looking only at the individual “pixel,” looking also at the group of “pixel” and use it as a feature will highly increase the accuracy.