**data set 2.**

(Devin)

Question 1.

**Use PCA to reduce dimensions. How many components do you need to keep to reproduce the digits reasonably well? what is your final matrix?**

To capture ~94% of the variance, 75 PCs is enough, and based on plots of the cumulative variance, the elbow at which diminishing returns for including additional PCs starts about here. The drawn images are recognizable at this level, and even at lower thresholds around 20~30 PCs the images can be visually interpreted. However, the background is not correctly reflected as completely white, and the label assignments are not correct even at 75 PCs or higher, despite cumulative variance totaling ~99%. To reach the highest level of accuracy at which the background and labels are correct, 567 PCs are needed. This amount is still 75% of the original number of components, which isn’t an excellent reduction of dimensions. Excluding the labels could potentially improve the dimensionality reduction, though they are a single column with fairly low variance.

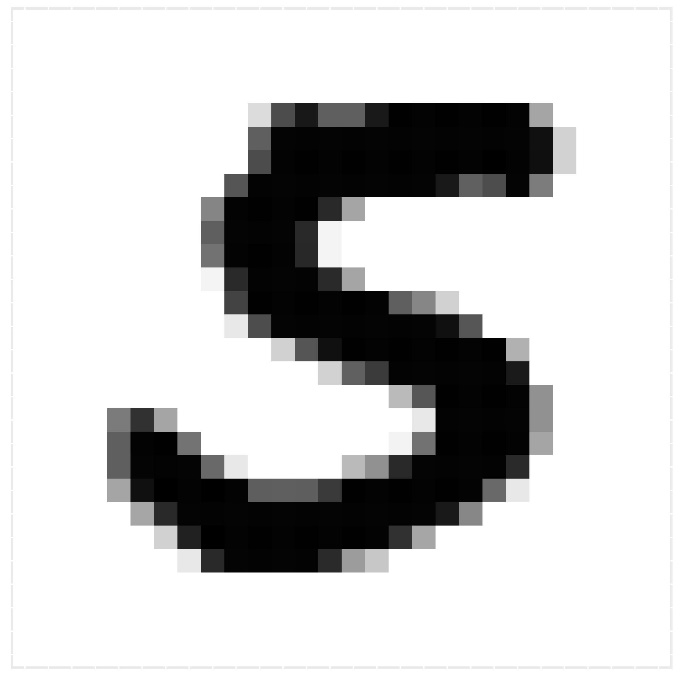


Figure 3 Digit drawn with original data for reference

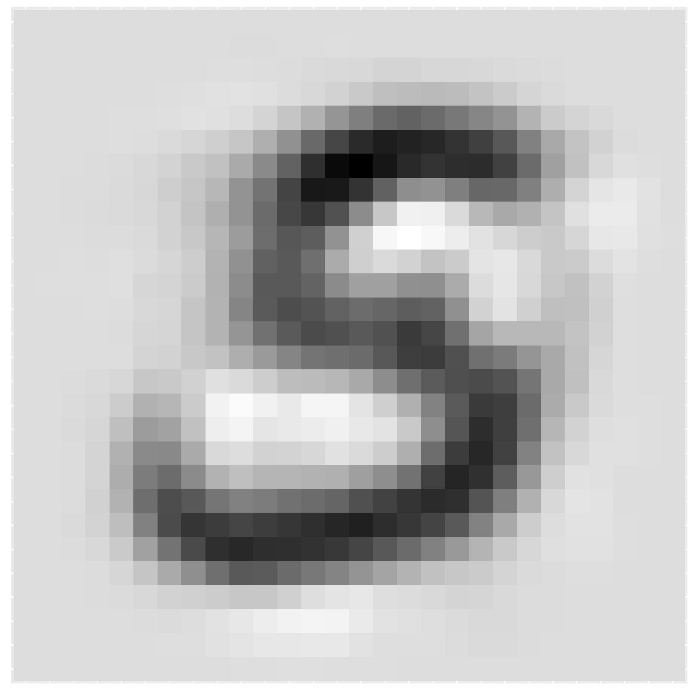
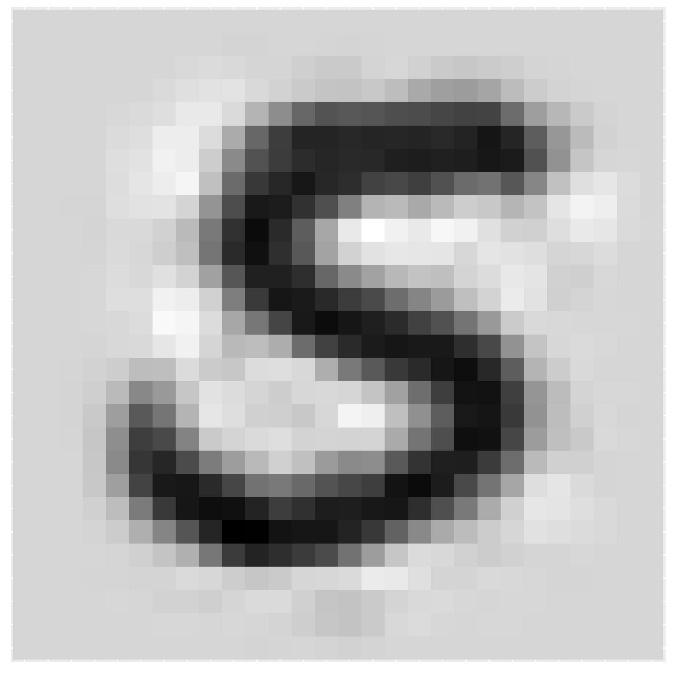


Figure 2Digit drawn with 20 PCs

Figure 1Digit drawn with 75 PCs

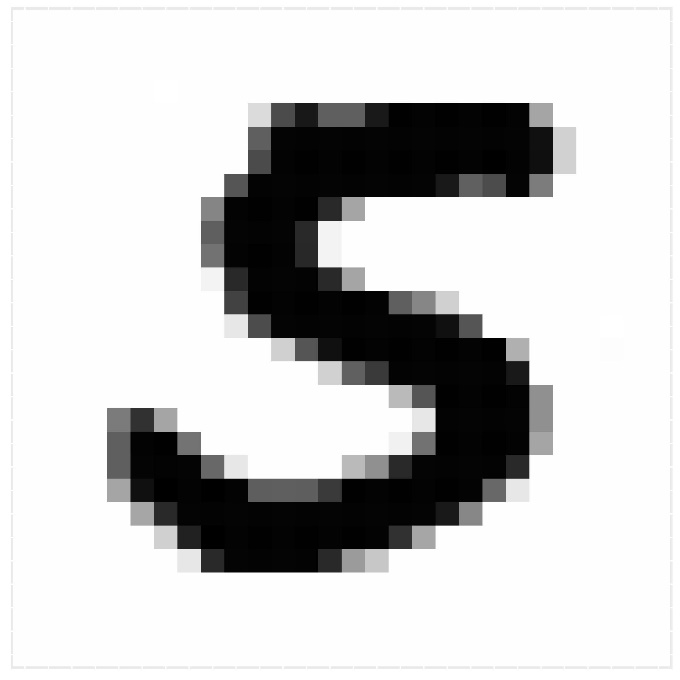
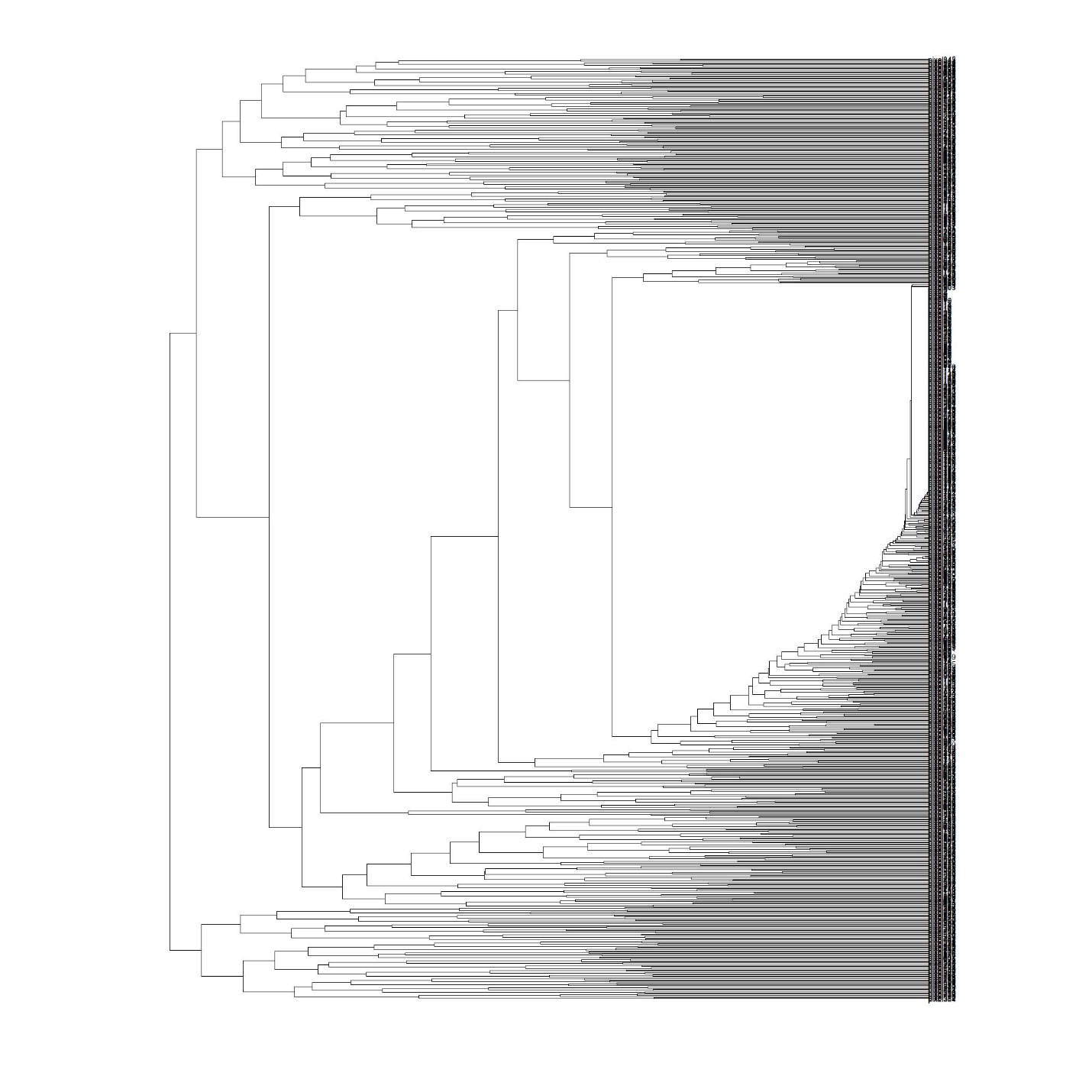


Figure 4Digit drawn with 567 PCs

## Question 2.

**Draw a tree of the pixels, and see if you can explain the results based on geometry of the pixels (how far apart are they in the 2-d space). Try to Explain the PCA results in light of this.**

By creating the dendrogram, we can see that the pixels with little to no data in them (mostly the edges of the images) cluster together and make up about 25% of the data. This is consistent with the above finding that 75% of the PCs were required to recreate the images.



## Question 3.

**Can you use some of the tools you have learnt to build a classifier,so if you get a new set of pixels you can predict what is in the picture. This is a the start of a real project, but you don't have all the tools (such as neural networks) which might be more suited for this task. Split your dataset into two (a training set and a test set), build your classifier and figure out how well it does with the test data in predicting the digits. Define the sensitivity and specificity of your classifier. How well does it recognize your own handwriting? (make sure your handwriting is not in the training set)**

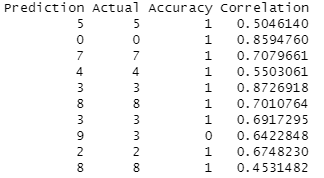
This will be addressed using three different levels of algorithmic complexity.

3.a. Machine learning algorithms/neural networks are the obvious choice for this task, but first we will try to classify by simply using the average for each digit and correlating against that. Using this method is about 80% accurate. When comparing to the handwriting samples I created, it was limited by the fact that there was only a single comparison, between the average and the sample for each digit. It was able to identify five of the digits correctly.

Sample classification based on correlation:



Results of first 10 samples:



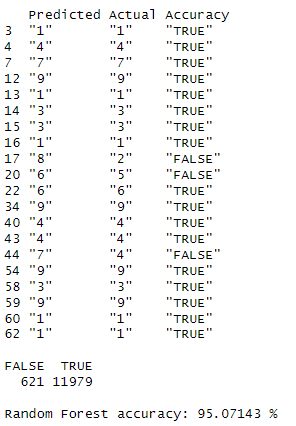


3.b. The results of the correlative approach seemed insufficient, so using a machine learning model with PCA was the next step. By reducing dimensions and using LDA, we hope to classify each digit using a smaller number of the PC values instead of all 784 pixel columns. However, the performance of this model is poor despite various attempts to optimize, including using up to 500 PCs. The specificity and recall (the term sensitivity only applies to binary classification) were both far below acceptable standards, approximately between 0 and 0.2 depending on the digit being evaluated.

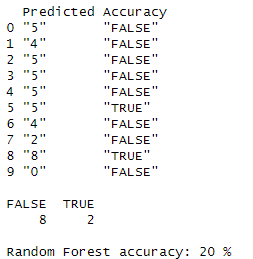


3.c. The PCA model was disappointing as well. Thus, a more proper machine learning algorithm seems to be required. We next evaluate a random forest model. This model uses much less time and lines of code, and has an accuracy of 95% when applied to the test set. However, it is completely unable to identify more than two of the handwritten samples. When applied to the averages for each digit, it can correctly identify 4 out of the 10. The failure to identify the handwriting samples may be due to the scale of the images in both position and intensity - the amount of whitespace outside the writing is not exactly the same, and the intensity of the black color is less in the samples. It may perform better if each image in both sets had any row or column in the 28x28 image with a maximum value below a certain threshold removed, and with the color normalized so that the darkest black part is always equal to the maximum.

Test data accuracy (First 20 predictions):



Handwritten digit accuracy:



As a final observation, the basic correlative approach performed surprisingly well compared to more advanced algorithmic techniques.

## Question 4.

**You can try simple things like take average of all data for each number and then take a "dot" product with your test set, and identify the pixels. This might work, maybe for some digits, and not others.**

Here we will see which of the digit averages is best classified by the Random Forest model. This model is only about to correctly identify 4 of the 10 digits.

Average digit accuracy:

