CSC2062 AIDA - Assignment 2 Report

Charles Collins

40256761

19/03/2021

Introduction

Throughout this assignment I will be creating a dataset consisting of character images of the classes: digits, letters and maths symbols. Features will be identified for the images and the data will logged, this data will be used in a multitude of statistical tests and in the training of a basic machine learning algorithm.

Section 1 – Creating a Dataset

To create the images I used gimp to create a 25 by 25 pixel canvas and then drew the necessary characters with a brush thickness of 1 pixel, the image was also greyscale so that there were only two colours to represent, black and white. Next I exported the images from their standard file type to a .pgm file as it can opened and read as a text file. To create the image matrices I looped reading the pgm files as text files and checked each value within the file if the value was greater than or equal to 128 it was represented as white with a 0 and if it was less than 128 it was represented as black by a 1. These values were put into a 25 by 25 matrix and then written into a csv using gsub to change the .pgm to .csv.

Section 2 – Feature Engineering

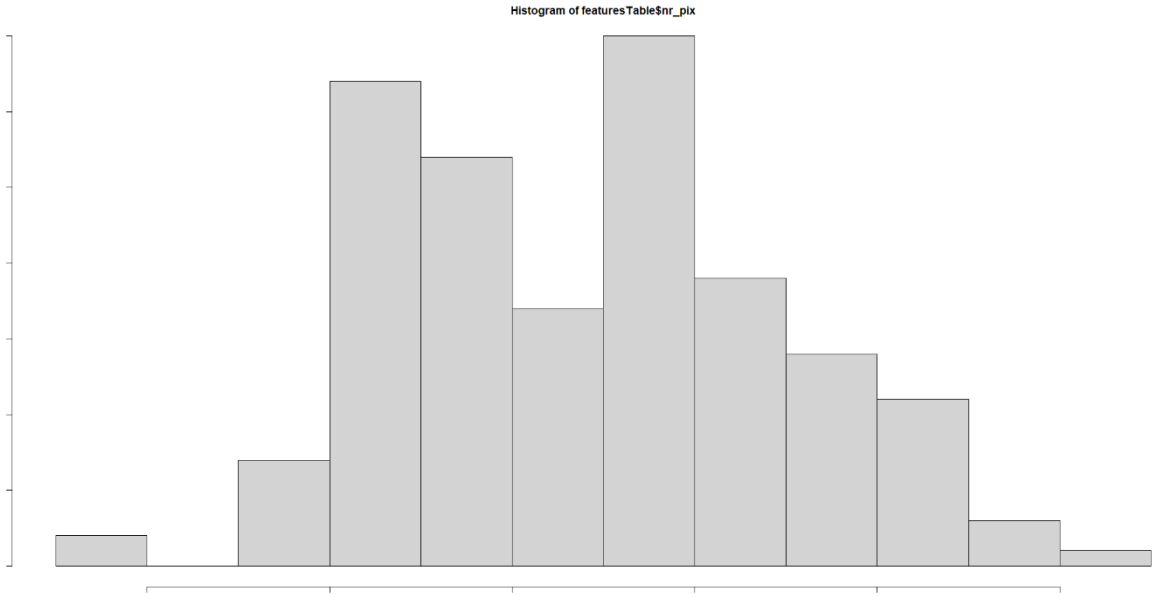
To acquire the necessary features of each character I first created an empty matrix with the correct column names that will eventually be filled with the corresponding feature values. I then loop to read in each csv file that contains the matrix for its unique character, within that loop I split the acquired file name so I can correctly assign the strings to the label and index features respectively. Next I loop within the unique character matrix to obtain values for each of the 14 features. The feature nr_pix is acquired by counting the number of black pixels in the matrix which is represented by the number of ones in it. To calculate the rows_with_2, cols_with_2, rows_with_3p and cols_with_3p features I take the mean of the row or column and if it is equal to 0.08 it has 2 pixels in that row or column, as 2/25 = 0.08 and if it is greater than or equal to 0.12 it has 3 or more pixels in that row or column, as 3/25 = 0.12. To get the height I find what row the first pixel is on by going top to bottom and I find what row the last pixel is on by going bottom to top, then I minus the first row from the last row to get the height. To get the width I find what column the first pixel is in by going left to right and I find what column the last pixel is in by going right to left, then I minus the first column from the last column to get the width. I calculate the left2tile feature by creating a nested for loop to iterate through the character matrix row by row checking if within a 2 by 2 matrix if the left two pixels are black, for right2tile the same loop also checks if the right two pixels are black. To find the top2tile and bottom2tile the loop checks if the top pixels are black and if the bottom two pixels are black respectively. Verticalness is calculated through adding left2tile to right2tile and dividing them by nr_pix. Horizontalness is similarly calculated by taking the sum of top2tile and bottom2tile and dividing them by nr_pix. The custom feature I chose to implement was isolated which checks if there is a black pixel surrounded by all white, which is represented by a 1 encircled by 0s within a 3 by 3 matrix. This feature would be useful as when the dataset is expanded to include all letters the isolated feature will assist in identifying lower case I as well punctuation such as full

stops. The isolated feature is calculated by using a nested for loop and iterating row by row across the matrix checking if within a 3 by 3 matrix the middle value is a 1.

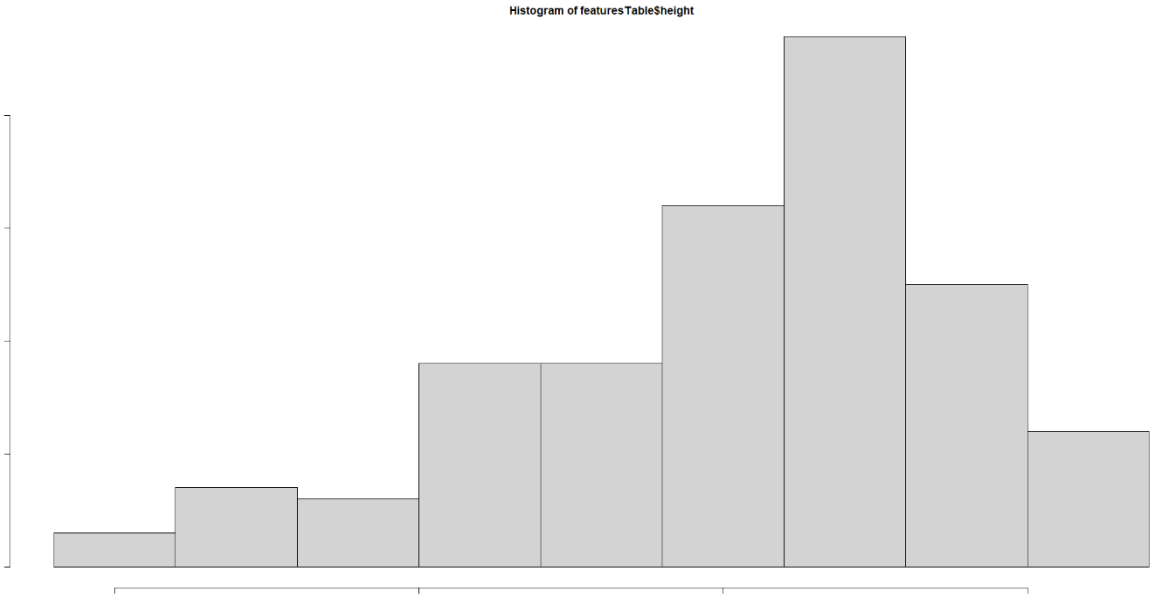
Section 3 – Statistical analyses of feature data

3.1

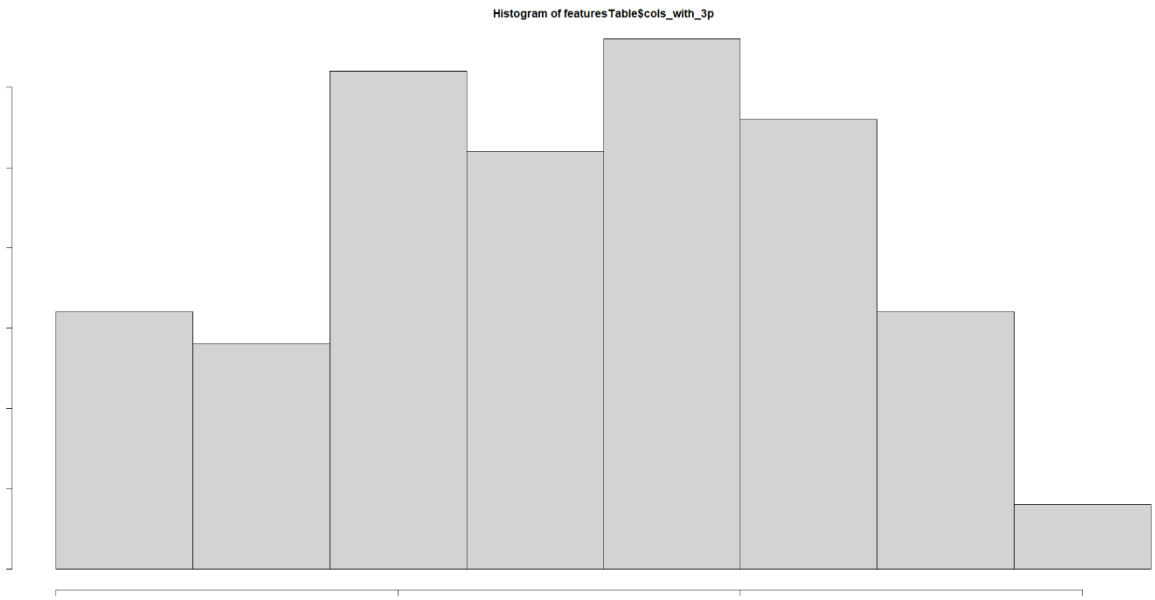
The first histogram show the distribution of the nr_pix feature which is the number of pixels within each character matrix, this histogram is bimodal as there are clearly two peaks. It forms a slight left skewed distribution as the right tail containing the larger values is longer than the left tail. The assumptions I made when plotting this histogram was that the data came from a single statistical distribution, the distribution is a normal distribution and the errors are uncorrelated over time.



The next histogram shows the distribution of height across the 168 characters, it shows a right skewed unimodal distribution as it has a single peak and the right tail is shorter than the left tail. The assumptions I made when plotting this histogram was that the data came from a single statistical distribution, the distribution is a normal distribution and the errors are uncorrelated over time.



The final of these histograms shows the distribution of the rows_with_3p feature. This histogram forms a plateau distribution as there are multiple peaks close together, while still having slight tails at either end of it. The assumptions I made when plotting this histogram was that the data came from a single statistical distribution, the distribution is a normal distribution and the errors are uncorrelated over time.



3.2

If you randomly sample a digit image from the full set of images the probability that that digit image has a pixel count greater than or equal to 20 is 96.4285714285714 % as calculated by R. This was done by counting the number of digit files with a pixel count greater than or equal to 20 and then dividing that file count by the number of digit files, which was 56. When conducting this test I assumed there were no outliers within the data that would skew the result.

3.3

I calculated the mean and standard deviation about each feature for the full set of letters, the full set of digits and the entire set of 168 characters. The mean gives a value for the average for each of the features, this average value can be used to compare between the sub-sets(letters and digits) as well as the entire dataset. The standard deviation is a measure of the spread of the data around the mean value, so the higher the standard deviation the less useful the feature as the data is less concise and probably not indicative of a trend throughout the respective subset. For the number of pixels feature the mean values for the letters and digits are both lower than the mean value for the entire dataset, 44.02, 44.57 and 44.67 respectively. This does show that the average number of pixels in a picture is slightly lower for the letter images than the digit images. The standard deviation shows that there is a larger spread in nr_pix value for the letters than for the digits or the total dataset, while the digits have less spread than the dataset as a whole shown by the lower standard deviation. The standard deviation is 10.34, 10.89, 10.70 for digits, letters and the total respectively. I do not believe this feature will be useful in differentiating between digit and letter despite the clear difference in summary statistics due to the fact that it is independent from the shape of the picture and just shows how large the picture may be. The row_with_2 average for the digits was 4.5, while

for the letters it was 3.77. The average across the entire dataset was 4.23. These averages show that the letters images are less likely to have rows with exactly two pixels in it. The standard deviations are 3.26 for digits, 2.72 for letters and 3.17 for the dataset. This shows that the variation in the data is much higher for the digits images with this feature. The standard deviation is quite high meaning that the data values vary heavily from the mean of the data which is why I think this feature is not the most suitable for differentiating between letters and digits. The means of the cols_with_2 for digits and letters are both lower than the mean for the whole dataset, this is likely due to the various forms of equals sign used in the symbols set skewing the total mean to be higher, also resulting in a high standard deviation for the whole dataset. The means for digits and letters have a large difference between them being 1.25 and 3.11 respectively, while the standard deviations for them are 1.73 and 2.69 which is quite low, meaning there is minimal spread in the data from the mean values. It is due this difference in means and low standard deviation that I think that cols_with_2 is a useful feature to distinguish between digits and letters. The rows_with_3p feature had averages of 4.29, 7.23 and 5.57 for digits, letters and total dataset respectively. The significant difference for them is 2.63, 3.05 and 3.18. The cols_with_3p feature had means of 8.43 for digits, 6.91 for letters and for the entire dataset the mean is 7.96. The standard deviations are 2.63, 3.04 and 3.18 respectively. These summary statistics show that letters have less columns with 3 or more pixels, but have a greater spread of data when compared to the same statistics for digits. The mean and standard deviation for the height of the digits is 16.91 and 2.30, for letters it is 15.88 and 4.12, and for the total it is 15.74 and 3.71. There is a low standard deviation across all three, but due to the minimal difference between the means of the digits and letters I think there are better features to use to distinguish between them. 11.75, 10.09 and 12.5 are the width averages of digits, letters and the total dataset respectively, while 3.63, 3.26 and 3.91 are the respective significant differences. Similar to the height feature there is not a large difference between the means and the feature is not indicative of the shape of the image so it is not useful in differentiating between the images. Left2tile had means and standard deviations of 12.12 and 5.93, 14.16 and 6.54, and 9.59 and 7.40 for digits, letters and total respectively. Similarly for right2tile the means and standard deviations were 11.73 and 5.98, 14.18 and 6.19, and 9.47 and 7.24 for digits, letters and the whole dataset respectively. Both these features have similar values for the means and standard deviations with a clear difference forming between digits and letters, it is due to this that I think verticalness will be a good feature to use to distinguish between the images as it is calculated using both left2tile and right2tile. The means for verticalness are 0.58 for digits, 0.65 for letters and 0.44 for all 168 images with standard deviations of 0.38, 0.27 and 0.37 respectively. As it was with left2tile and right2tile, top2tile and bottom2tile have similar averages for digits, letters and the dataset as a whole with them being 15.52 and 15.09, 10.11 and 9.71, and 15.55 and 15.20 with standard deviations of 7.97 and 8.07, 4.60 and 4.33, and 8.15 and 8.18. As previously stated I believe as both features would make relatively useful indicators of if the image was a letter or a digit the feature calculated with both of them, horizontalness, would be a good way to differentiate between the images. The final feature is isolated, but as there are currently no images that meet the parameters of the isolated feature it will be useless in distinguishing between the letters and digits in the current dataset, but will be imperative in identifying characters such as the letter I or punctuation like a full stop when the dataset is expanded. When analysing the mean I worked under the assumption that there were no extreme outliers within the dataset which would unfairly impact the result. Likewise, when calculating and analysing the standard deviations I assumed the endpoints of the distribution were known and that there were no extreme outliers that would skew the resultant standard deviation to a noticeable degree.

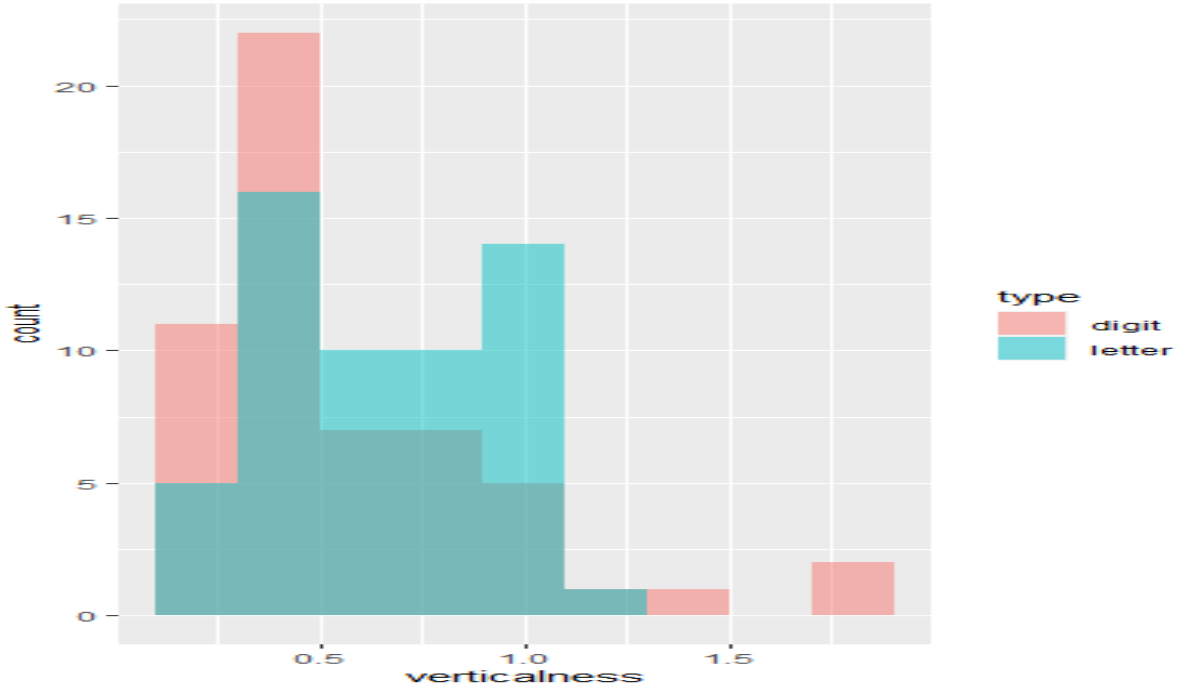
3.4

To investigate the degree of linear association between the height and verticalness features I conducted a Pearson correlation test, resulting in a t-value of 5.9394, degrees of freedom of 166 and a p-value of 1.63e-08. As the significance level is 0.05 and the p value is less than it, 1.63e-08<0.05, so we reject the null hypothesis. We can conclude that height and verticalness are significantly correlated with a correlation coefficient of 0.419 (3 significant figures). This correlation coefficient suggests a slight positive correlation between the two features, so as one increases the other is also likely to increase. I used a Pearson correlation test as it easily calculates the degree of linear association between two groups, which is exactly what was required for this sub-task. To complete the Pearson product moment correlation test I assumed the absence of outliers, and linearity.

3.5

To investigate if any of the features would be useful in discriminating between letters and digits I conducted t tests at a 95% confidence level for each feature of the letter and digit images. Based on the results for the nr_pix feature there is not a significant difference between the letter and digit means as the p-value was 0.778 which is greater than 0.05. For the t test conducted for the row_with_2 feature we have to accept the null hypothesis of there is no significant difference as the resultant p-value of 0.185 is greater than 0.05. We can accept the alternate hypothesis for the cols_with_2 t-test as the p-value, 2.297e-05, is less than the 0.05 confidence value. The alternate hypothesis can also be accepted for the row_with_3p t-test as once again the p-value of 1.53e-07 is less than 0.05. Similarly, the p value for cols_with_3p was less than 0.05, so we can accept the alternative hypothesis as there is a significant difference in the means for letters and digits. At the 95% confidence level the null hypothesis is accepted for the height t-test as the p-value was 0.099 which is greater than 0.05, showing there is no significant difference in the tested mean values. There was a significant different difference between letters and digits for the average widths as the p-value, 0.011 was slightly lower than 0.05. There is no clear indication of a significant difference between the left2tile averages for letters and digits as the p-value was 0.085. The right2tile averages did show a significant difference as the t-test resulted in a p-value of 0.034, so we can reject the null hypothesis and accept the alternate one. The means of letters and digits for verticalness showed no significant difference as the null hypothesis is accepted due to 0.05 being greater than the resultant p-value of 0.253. The means for letters and digits showed a significant difference for both top2tile and bottom2tile with p-values of 1.93e-05 and 2.08e-05 respectively. As the p-value for the horizontalness was 2.902e-05 which is less than 0.05 we can accept the alternate hypothesis showing there is significant difference between the means which were 0.659 and 0.462 for digits and letters respectively. As previously stated the isolated feature is currently unpopulated as it is relevant for a future expanded dataset, so even though a t-test was conducted for the feature the results are not relevant. After analysing the t-tests for the features of letters and digits I believe the three most useful features are cols_with_2, rows_with_3p and horizontalness. All three of these features displayed a significant difference between digits and letters, showing that they could be used to differentiate between the two character types for the current dataset and potentially an expanded version of it if the trend is maintained. Another reason these specific features would be useful is because they give some indication of shape due to analysing pixel patterns or positioning whereas features such nr_pix or height merely show how large the image is irrespective of what the image is. I used t-tests over other statistical tests that compare between two groups as it was simple and gave the most streamlined results which were correct for the sub-tasks requirements. Assumptions were made when conducting the t-tests such as, normality of data distribution, adequacy of sample size and homogeneity of variance. The assumptions I made when plotting this

histogram was that the data came from a single statistical distribution, the distribution is a normal distribution and the errors are uncorrelated over time.



Section 4 – Regression and Machine Learning

4.1

Label	Index	Predicted_H	Actual_H
a	1	0.301527	0.363636
a	2	0.570702	0.5
a	3	0.475652	0.469388
a	4	0.637864	0.564103
a	5	0.543993	0.511628
a	6	0.441175	0.428571
a	7	0.241996	0.3
a	8	0.586649	0.457143
approxequal	1	0.860283	0.785714
approxequal	2	0.238613	0.353846
approxequal	3	0.520118	0.52459
approxequal	4	0.468075	0.484848
approxequal	5	0.507587	0.532258
approxequal	6	0.852175	0.753846
approxequal	7	0.391934	0.478261
approxequal	8	0.436686	0.492063
b	1	0.238427	0.27907
b	2	0.335413	0.375

b	3	0.303383	0.27027
b	4	0.480344	0.419355
b	5	0.231552	0.276596
b	6	0.346957	0.346939
b	7	0.431361	0.391304
b	8	0.315541	0.326531
c	1	0.799718	0.827586
c	2	0.551564	0.451613
c	3	0.864246	0.8125
c	4	0.965972	0.944444
c	5	0.473263	0.52
c	6	0.660137	0.655172
c	7	0.580165	0.512821
c	8	0.724191	0.657895
d	1	0.32822	0.333333
d	2	0.390888	0.378378
d	3	0.174664	0.265625
d	4	0.140916	0.133333
d	5	0.381612	0.375
d	6	0.327706	0.340909
d	7	0.303197	0.289474
d	8	0.411023	0.435897
e	1	0.959606	0.9375
e	2	0.864605	0.869565
e	3	0.881075	0.857143
e	4	0.844383	0.828571
e	5	0.603762	0.580645
e	6	0.790685	0.864865
e	7	0.62121	0.555556
e	8	0.680188	0.62963
equal	1	1.718456	1.866667
equal	2	1.662934	1.757576
equal	3	1.768324	1.875
equal	4	1.73703	1.870968
equal	5	1.768324	1.875
equal	6	1.774832	1.875
equal	7	1.720501	1.8125
equal	8	1.875812	1.885714
f	1	0.216062	0.193548
f	2	0.350721	0.382353
f	3	0.382119	0.351852
f	4	0.396474	0.344828
f	5	0.302089	0.277778
f	6	0.404835	0.388889

f	7	0.450334	0.416667
f	8	0.337503	0.3125
five	1	0.88195	0.893617
five	2	0.742712	0.744681
five	3	0.983694	1.022727
five	4	1.390898	1.233333
five	5	0.955431	0.949153
five	6	1.12938	1.191489
five	7	0.804693	0.784314
five	8	0.972934	1.05
four	1	0.622069	0.596154
four	2	0.510246	0.538462
four	3	0.481056	0.469388
four	4	0.469416	0.487179
four	5	0.699223	0.612903
four	6	0.341125	0.418605
four	7	0.518184	0.534483
four	8	0.521019	0.560976
g	1	0.47865	0.470588
g	2	0.154156	0.206897
g	3	0.421441	0.460317
g	4	0.275073	0.313725
g	5	0.319523	0.343284
g	6	0.264087	0.366197
g	7	0.393024	0.4
g	8	0.246621	0.306452
greater	1	0.51296	0.512821
greater	2	0.664805	0.605263
greater	3	0.556436	0.514286
greater	4	0.754671	0.69697
greater	5	0.367315	0.258065
greater	6	0.80059	0.714286
greater	7	0.555217	0.514286
greater	8	0.500821	0.541667
greaterequal	1	1.094806	1.090909
greaterequal	2	1.001644	1.054054
greaterequal	3	0.935678	0.963636
greaterequal	4	1.070462	1.035088
greaterequal	5	0.965899	0.981132
greaterequal	6	1.041633	1.041667
greaterequal	7	0.92333	0.977778
greaterequal	8	1.015816	1.042553
less	1	0.638351	0.641026
less	2	0.616362	0.611111

less	3	0.575869	0.55
less	4	0.971504	0.903226
less	5	0.680656	0.642857
less	6	0.803011	0.724138
less	7	0.843359	0.764706
less	8	0.713415	0.647059
lessequal	1	1.133257	1.051724
lessequal	2	1.101381	1.130435
lessequal	3	0.978	0.94
lessequal	4	1.237684	1.173077
lessequal	5	1.096958	1.085106
lessequal	6	1.068728	1.097561
lessequal	7	1.015529	1.041667
lessequal	8	0.986615	1.041667
notequal	1	1.034063	1
notequal	2	1.021905	0.949153
notequal	3	1.033316	0.964912
notequal	4	0.888412	0.911111
notequal	5	1.043161	0.979167
notequal	6	1.166849	1.122449
notequal	7	1.084473	0.982143
notequal	8	0.976725	1
one	1	0.387128	0.388889
one	2	-0.02423	0
one	3	0.5359	0.439024
one	4	-0.01762	0
one	5	0.107155	0.030303
one	6	0.47433	0.483871
one	7	0.431641	0.441176
one	8	0.571118	0.5625
seven	1	0.630051	0.673913
seven	2	0.750271	0.813953
seven	3	0.697389	0.757576
seven	4	0.736705	0.794118
seven	5	0.569957	0.630435
seven	6	0.801128	0.861111
seven	7	0.57809	0.676471
seven	8	0.699948	0.714286
six	1	0.429863	0.508475
six	2	0.575911	0.574468
six	3	0.686311	0.7
six	4	1.434675	1.21875
six	5	0.739242	0.763636
six	6	0.435612	0.512821

six	7	0.536354	0.574074
six	8	0.560732	0.56
three	1	0.824923	0.775862
three	2	0.467365	0.509804
three	3	0.713548	0.673913
three	4	1.363285	1.259259
three	5	0.442088	0.491228
three	6	0.638409	0.666667
three	7	0.470643	0.529412
three	8	0.541398	0.5
two	1	0.708996	0.684211
two	2	0.574882	0.588235
two	3	0.793339	0.780488
two	4	1.181597	1.191489
two	5	0.689366	0.681818
two	6	0.660133	0.757576
two	7	0.460915	0.489796
two	8	0.631225	0.542857

This table shows the value predicted for horizontalness by the multiple regression model alongside the actual horizontalness value and the respective label and index. The model used all other features to assist in the prediction of the horizontalness value. As the data shows most of the predicted values are relatively near the actual value, indicating there may be some link between horizontalness and the other features used for the regression model. A multiple linear regression analysis makes several assumptions like, there must be a linear relationship between the outcome variable and the independent variables and the variance of error terms are similar across the values of the independent variables(Homoscedasticity).

4.2

```
Generalized Linear Model

112 samples
  3 predictor
  2 classes: '0', '1'

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 90, 90, 90, 89, 89
Resampling results:

  Accuracy   Kappa
  0.7940711  0.5890549
```

The accuracy value given (0.79) is an indicator of what percentage of the instances were correctly classified, so with an accuracy of 79% and there being 112 samples the model using the given predictors correctly predicted the type of 88.5 of the 112 samples. The Kappa value is a statistic used to measure the inter/intra-rater reliability for categorical items, which is a score of the consistency of the ratings from different or same judge(s) respectively. The Kappa value is more robust than accuracy as it considers random chance, meaning when taking into the account the chance for a random correct prediction the classification accuracy is 59% or 66.1 samples out of 112. The basic

assumptions that had to be met for the logistic regression include independence of errors, linearity in the logit for continuous variables, absence of multicollinearity, and lack of strongly influential outliers.

4.3

Using the binomial distribution I tested the probability of the random model and the logistic regression model correctly predicting all 112 image types, I can say the logistic regression model from 4.2 more accurately distinguishes between letters and digits than a 50/50 random assignment model. This is because the probability for the logistic regression model to predict all 112 types correctly was 6.023e-12, whereas the random model only had a probability of 1.926e-34. This means the logistic regression model is 3.1277259e+22 times more likely to correctly predict every type. To prove there is a significant difference between the two models I used the binomial distribution to check the probability that the random model gets the same result as the logistic model because we know the logistic model is more accurate as 79.4%>50%. The resultant p value (1.19e-10) from this was less than 0.05 so at a 95% confidence level we can say there is a significant difference between the two models. The underlying assumptions of the binomial distribution are that there is only one outcome for each trial, that each trial has the same probability of success, and that each trial is mutually exclusive or independent of each other.

Conclusion

In conclusion, for section 1 I believe I was able to properly create and reformat the dataset images as requested. The 2nd section required the creation of features for each image within the dataset, which I completed to the standard requested within the assignment document. Section 3 involved the data analysis of the features from section 2 with the goal to begin determining which features will useful in the discrimination between letters and digits. Finally, section 4 built upon section 3 in terms of distinguishing between letters and digits, but by using modelling and machine learning to discriminate.