Sign Language Digits Classification

Multivariate Analysis Project Report

Anton Potapchuck, Khatia Kilanava, Novin Shahroudi
Institute of Computer Science, University of Tartu

June 1, 2018

1 Introduction

This project is a classification task on a set of sign language images which includes pictures of hands showing a number in sign language. Figure 1 demonstrate different sample types in the dataset. Our goal is to classify these images with the help of discriminant analysis methods such as LDA and QDA. In Section 1 we demonstrate some of the characteristics of the data, then we explain the methods in Section 2, and demonstrate our final result in Section 3.3 and finally discuss the conclusions of this effort.

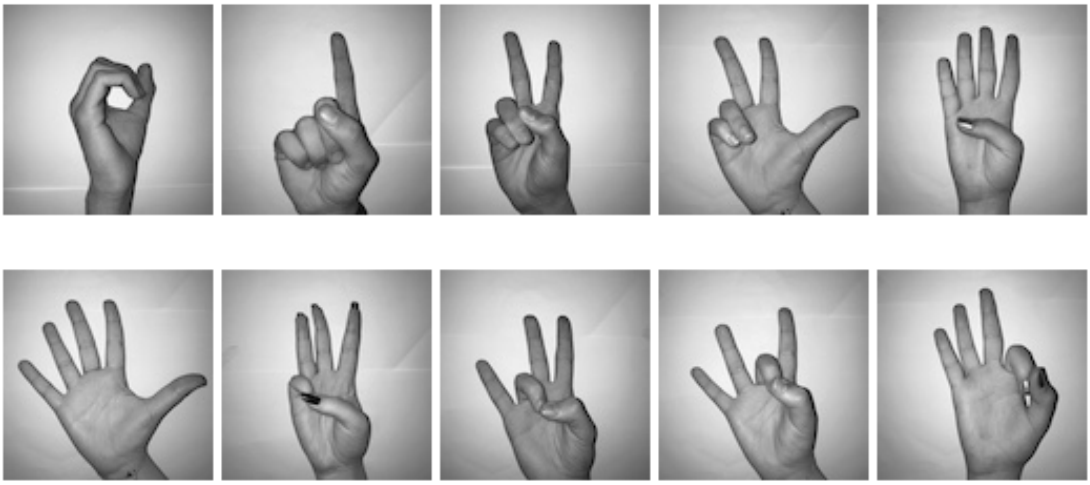


Figure 1: Different samples in the dataset representing digits 0 to 9 from top left to the bottom right respectively

2 Dataset

The dataset is obtained from Kaggle competition[1]. The dataset includes 2062 images with corresponding labels of 10 digits from 0 to 9. All the images are in Grayscale each in the dimension of 64x64. Hence, we have 4096 variables for each image that represents pixels of the image. Since the image is grayscale, each pixel is in a range of 0 to 1 which is from the darkest to the highest brightness.

The dataset is balanced meaning that there is an almost same number of labels from each class, hence roughly 205 labels for each class. It is hard to study the distribution of the features as there are 4096 pixels (features) existing. However, we picked 9 pixels from 9 different parts of each image as depicted in Figure 3 to have more insight on the distribution of these pixels with the hope that it gives a general insight on the distribution of the data. Figure 2 demonstrates the distribution of these 9 pixels.

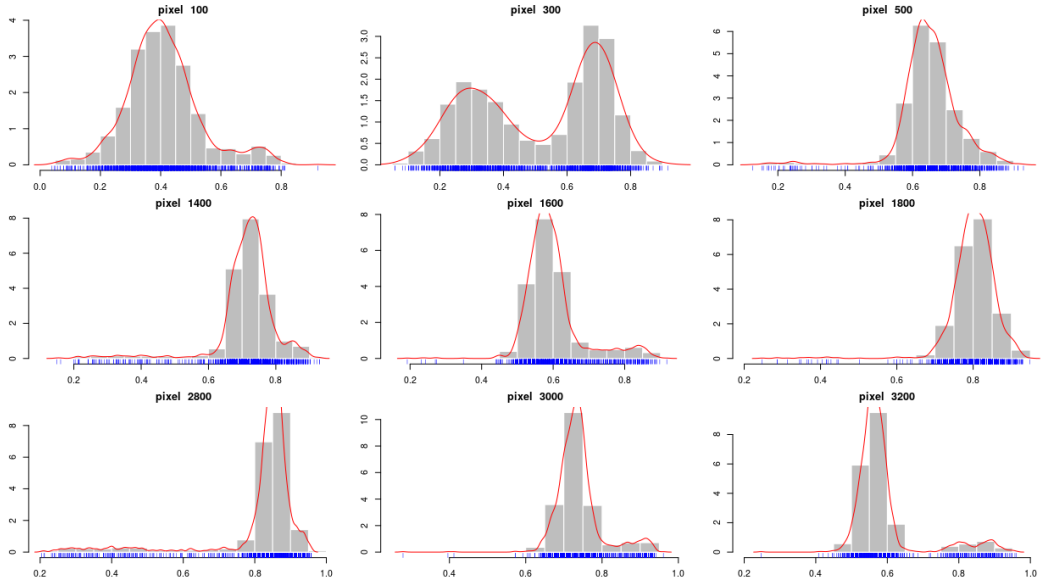


Figure 2: Distribution of 9 picked pixels

Distribution of these 9 pixels shows that some of the pixels represent roughly a normal distribution, and some a mixture of multiple normal distributions and other distributions.

3 Methods

Based on the findings from the dataset exploration we have a large number of pixels (almost twice the number of samples) which can degrade performance and even make the discriminant analysis impossible due to dimensionality and dependence of the variables. To this end, we need to reduce dimensionality of the problem. Also considering the fact that the pixels are not from normal distribution, we need to consider that in the preprocessing. This is necessary since the discriminant analysis methods that we use assume multivariate normality [1].

[1] `https://www.kaggle.com/ardamavi/sign-language-digits-dataset`

Figure 3: rough position of pixels where above distribution is demonstrated for

3.1 Dimension Reduction

We decided to perform Principal Component Analysis (PCA) to reduce the number of dimensions. We apply PCA on the pixels and obtain new components. Instead of using original pixels in the image we use the first principle components that explain the most variance. Since first principle components do not explain all the variance, the noise in the images are also reduced at the same time. Thus, the classification models will not learn the noise, which leads to better performance. We used criteria explained in [2] to choose appropriate number of principle components. First criteria was by thresholding the total variance explained from the first PC. Second criteria was to pick all principle components that are greater than the average eigenvalues.

3.2 Linear Discriminant Analysis

Linear discriminant analysis considers a discriminant score for each observation in order to classify the observation in each of the classes. Eq. 1 is an estimate of the discriminant function to calculate this score for observation x where μ_k is the class-specific mean vector, Σ is the covariance matrix that is common among all classes, and π_k the prior probability that an observation belongs to the kth class. Observation is classified as k for the highest score function δ_k.

$$\hat{\delta}_k(x) = x^T \Sigma_k^{-1} \hat{\mu}_k - \frac{1}{2} \hat{\mu}_k^T \Sigma^{-1} - \hat{\mu}_k + log \hat{\pi}_k \tag{1}$$

3.3 Non-linear Discriminant Analysis

We employed Quadratic Discriminant Analysis (QDA) which is similar to LDA that it assumes the observations for each class are of Gaussian distribution $X \sim N(\mu_k, \Sigma_k)$, however, it also considers different covariance Σ_k for each class k. This enables the model to construct non-linear discrimination line between classes.

$$\hat{\delta}_k(x) = -\frac{1}{2} x^T \Sigma_k^{-1} x + x^T \Sigma_k^{-1} \hat{\mu}_k - \frac{1}{2} log \mid \Sigma_k \mid + log \hat{\pi}_k \tag{2}$$

Eq. 2 shows the formula which calculates the score value for observation x. Observation with the highest value is assigned to the respective class. Each term is similar to the Linear case. We can also see that the first term includes squared of the x and that is why it is called quadratic discriminant analysis.

4 Results

PCA

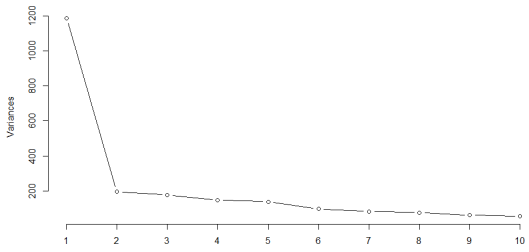
This is the result, demonstrating how much of the cumulative proportions are explained by principal components:

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11	PC12	PC13
Standard deviation	34.4357	13.96419	13.31933	12.17784	11.76110	9.86473	9.10166	8.76936	7.88638	7.5455	6.90726	6.75298	6.62532
Proportion of Variance	0.2895	0.04761	0.04331	0.03621	0.03377	0.02376	0.02022	0.01877	0.01518	0.0139	0.01165	0.01113	0.01072
Cumulative Proportion	0.2895	0.33711	0.38042	0.41663	0.45040	0.47416	0.49438	0.51316	0.52834	0.5422	0.55389	0.56502	0.57574

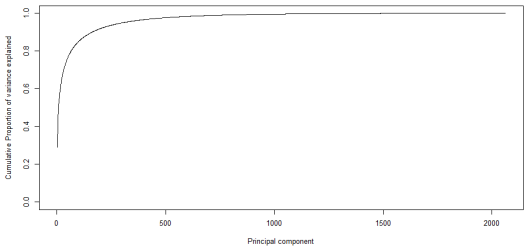
	PC14	PC15	PC16	PC17	PC18	PC19	PC20	PC21	PC22	PC23	PC24	PC25	PC26	PC27
Standard deviation	6.54627	6.2385	6.15934	6.06544	5.83957	5.59791	5.53651	5.43595	5.29662	5.14115	5.02158	4.82826	4.7891	4.69003
Proportion of Variance	0.01046	0.0095	0.00926	0.00898	0.00833	0.00765	0.00748	0.00721	0.00685	0.00645	0.00616	0.00569	0.0056	0.00537
Cumulative Proportion	0.58620	0.5957	0.60497	0.61395	0.62227	0.62992	0.63741	0.64462	0.65147	0.65792	0.66408	0.66977	0.6754	0.68074

Figure 4: Statistics of the first 30 principle components

As a result, we can see that first two PCs account for 33% of total variance, first four for 41%, first thirteen for 57% of total variance. For a better understanding of the result of PCA, we can visualize explained variance and cumulative proportion of explained variance for each principal component. As we can see from the Figure 5 (a), the first PC explains 6 times more variance than the second one. In Figure 5 (b), we can see, that we can explain more than 80 percent of explained variance by using just first 100 PCs. Probably, it can be a good result. In this case, we reduce the number of features 40 times.



(a) 10 first PC variance



(b) cumulative variance

Figure 5: Result of the principle component analysis

We decided to take several number of PCs that account for different amount of variances and perform QDA as well as LDA. Furthermore, study of the principle components suggest a normal distribution for each components which means that despite of original pixel values the resulting components are appropriate as an input for the discriminant analysis. Figure 6 demonstrates distribution of the first 9 principle components obtained.

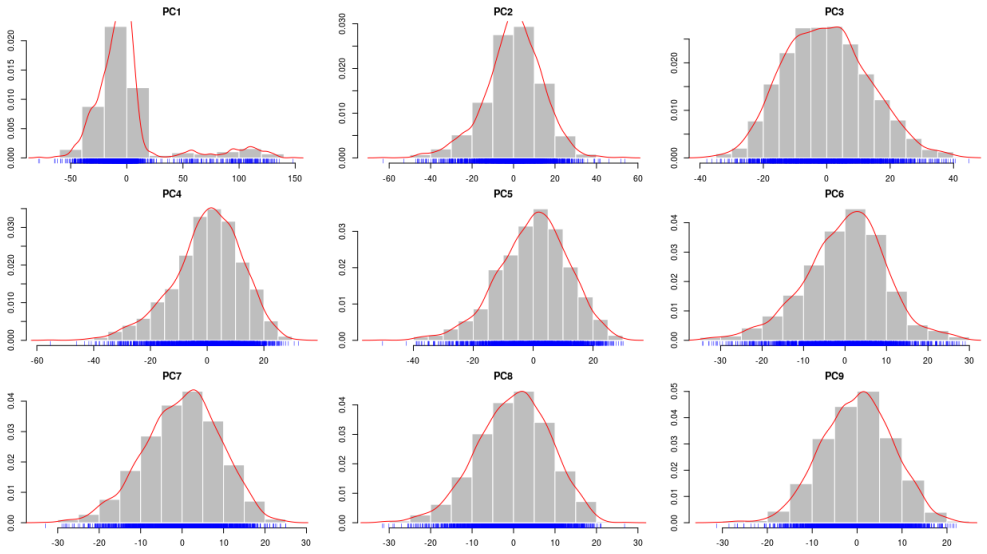


Figure 6: Distribution of first 9 principle components

Nr Components	302	181	**65**	30
Accuracy	0.73	0.762	**0.769**	0.69
APER	0.27	0.238	**0.231**	0.31

Table 1: LDA results with cross-validation

Discriminant Analysis

As a result we received the best accuracy of 85% with QDA by taking first 50 PCs. We observed the down performance of the models were partly due to the dataset mislabeling for some of the samples. Figure 10 demonstrates this observation.

Most of the observations in Figure 7 (b) are classified correctly but we can see the errors on the off-diagonal elements of the confusion matrix. For example class 3 was classified as class 2 four times. Class two was classified as class three seven times and as six eight times. This can be explained because those three classes images seem very similar.

LDA Figure 7 (a) labeled image with the class six as class three 9 times. But as we mentioned, those images are so similar, that it can be explained. But class 5 is labeled as class 2 nine times. This is not that realistic because image with class 5 is not similar to image with class 2. From both matrices we can conclude that both models performed the worst in classifying the images with the labels two, three and six. These images are very similar to others, so it is more difficult for the models to classify which one is which.

5 Discussion

As a result of our attempts, we observed mislabeling in the original dataset which influenced performance of both LDA and QDA models. However, training the models on the whole dataset, they were able to learn the data perfectly, despite of this mislabeling which shows

Nr Components	70	**50**	30	10
Accuracy	0.82	**0.84**	0.82	0.51
APER	0.18	**0.16**	0.18	0.49

Table 2: QDA results with cross-validation

y_test	0	1	2	3	4	5	6	7	8	9
0	41	0	0	2	0	3	3	1	0	2
1	0	33	0	2	2	1	3	0	0	0
2	0	2	36	2	2	5	6	0	3	0
3	0	1	2	48	0	2	4	0	4	0
4	0	1	2	1	38	2	2	0	3	0
5	2	1	9	0	1	33	1	0	0	1
6	0	2	5	9	0	2	39	0	1	0
7	0	0	0	1	3	0	1	38	3	0
8	1	0	5	2	3	0	1	1	36	0
9	1	0	0	0	0	2	1	7	0	44

(a) LDA

y_test	0	1	2	3	4	5	6	7	8	9
0	48	0	2	0	0	1	0	0	0	1
1	0	40	0	0	0	0	1	0	0	0
2	0	0	37	7	0	3	8	0	1	0
3	1	0	4	40	0	0	12	1	3	0
4	1	0	1	1	44	0	0	1	1	0
5	3	0	2	0	0	36	5	0	0	2
6	1	0	4	4	0	3	44	1	0	1
7	0	0	0	0	0	0	0	46	0	0
8	1	0	3	3	0	0	1	2	39	0
9	1	0	0	0	0	0	0	3	0	51

(b) QDA

Figure 7: Confusion matrix obtained as a result of QDA and LDA prediction on the test set, where rows represent true labels and columns the predicted class

capacity of the models for over-fitting. We also trained the models on the original features, and we observed that due to co-linearity of the features (pixels) it was not feasible to proceed with neither of the methods. Another observation in our effort was that it was not possible to run QDA model with more than about 130 principle components as the features. We believe that this has to do with a constraint on number of samples as we only had about 1500 samples on the training set.

6 Conclusion

As far we are working with high dimensional data, we applied a PCA to reduce a data dimension. We suppose that we can obtain a better performance using PCA because we can reduce a noise in the training set, so the model will not learn a noise. We consider different numbers of PCs based on the cumulative percentage of explained variance.

We have tried two different models to classify images: LDA and QDA. We achieved the highest accuracy with QDA 85% and 50 PCs whereas in case of LDA accuracy was equal to 75% and we used 65 PCs. As we mentioned above, in the original datasets, some images were mislabeled, so this can be a reason for the lower accuracy. In conclusion, we used 40 times fewer variables for classification. We obtained better accuracy on the test set using the smaller number of variables, that is a good demonstration of reducing a noise in the data.

The code used for this project is available online [3].

References

[1] G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning: with Applications in R.* Springer Texts in Statistics. Springer New York, 2014. [Chapter 4].

[2] A.C. Rencher. *Methods of Multivariate Analysis.* Wiley Series in Probability and Statistics. Wiley, 2003. [Page 397].

[3] Code release for this project. `https://github.com/novinsh/multivariate-project`, 2018.

Appendix I

	pred	true	0	1	2	3	4	5	6	7	8	9
1	9	9	9.0e-27	0.0e+00	0.0e+00	1.3e-137	1.9e-01	1.5e-95	1.6e-75	1.0e-10	8.1e-01	2.7e-147
2	9	9	1.8e-119	0.0e+00	0.0e+00	2.1e-87	3.9e-16	1.9e-38	4.4e-128	5.8e-16	1.0e+00	3.2e-09
3	4	4	7.2e-207	5.0e-83	8.1e-63	1.0e+00	5.2e-46	3.5e-61	3.0e-76	3.1e-33	4.0e-62	7.5e-83
4	1	1	1.0e+00	2.1e-147	8.3e-269	1.9e-130	2.0e-143	5.2e-133	6.5e-241	5.5e-251	2.0e-130	1.4e-100
5	8	8	1.1e-67	3.7e-49	8.4e-33	1.6e-55	1.0e-13	1.8e-32	1.1e-34	1.0e+00	9.6e-36	1.4e-34
6	2	2	3.6e-84	1.0e+00	5.4e-41	3.9e-39	3.5e-98	4.5e-54	2.4e-94	2.4e-85	1.6e-127	1.0e-126
7	2	2	3.6e-71	1.0e+00	1.5e-13	1.6e-48	7.0e-10	2.2e-25	2.1e-34	2.9e-13	1.7e-22	7.7e-13
8	7	7	3.1e-36	1.0e-47	2.5e-05	3.7e-27	8.8e-24	3.3e-119	7.6e-01	2.4e-01	2.4e-41	1.6e-86
9	8	9	1.4e-147	5.3e-298	2.9e-271	1.2e-49	1.1e-03	1.4e-15	2.1e-69	9.7e-01	2.4e-02	8.2e-11
10	5	5	0.0e+00	0.0e+00	8.3e-290	5.3e-193	1.0e+00	5.1e-60	3.6e-45	3.3e-29	1.8e-48	3.1e-132

Figure 8: Posterior probabilities for LDA; For each row, the label with maximum posterior probability is assigned to each label. From the table it is apparent that out of these 10 samples 9 were classified correctly. The sample predicted as 8 but had the label 9 is the misclassification. For this sample, the posterior probability of label 8 is 2.4e-0.2 and having 9 as 8.2e-11, much smaller than the first one.

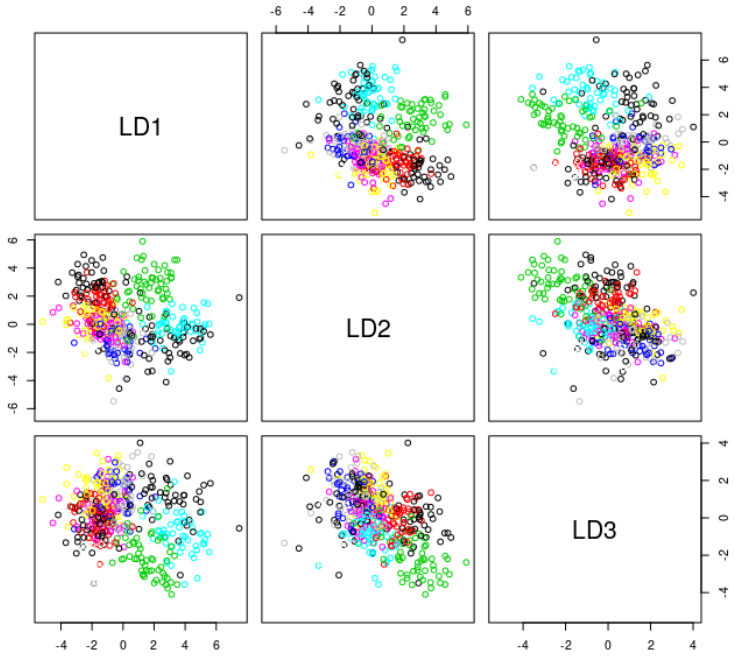


Figure 9: plot of linear discriminant coefficients - it is not possible to plot features or all of the coefficients, however, this pairwise plot can demonstrate and help in study of features that help in discrimination.

Figure 10: Some of the misclassification by LDA – left number: predicted class, right number: true label

Figure 11: Some of the misclassification by QDA – left number: predicted class, right number: true label