ROBUST L_p-NORM LINEAR DISCRIMINANT ANALYSIS WITH PROXY MATRIX OPTIMIZATION

Navya Nagananda & Andreas Savakis
Department of Computer Engineering
Rochester Institute of Technology
Rochester, NY 14623, USA
{nn3264,Andreas.Savakis}@rit.edu

Breton Minnehan
Air Force Research Laboratory
WPAFB, OH 45433, USA
breton.minnehan.1@us.af.mil

ABSTRACT

Linear Discriminant Analysis (LDA) is an established supervised dimensionality reduction method that is traditionally based on the L_2-norm. However, the standard L_2-norm LDA is susceptible to outliers in the data that often contribute to a drop in accuracy. Using the L_1 or fractional p-norms makes LDA more robust to outliers, but it is a harder problem to solve due to the nature of the corresponding objective functions. In this paper, we leverage the orthogonal constraint of the Grassmann manifold to iteratively obtain the optimal projection matrix for the data in a lower dimensional space. Instead of optimizing the matrix directly on the manifold, we use the proxy matrix optimization (PMO) method, utilizing an auxiliary matrix in ambient space that is retracted to the closest location on the manifold along the loss minimizing geodesic. The L_p-LDA-PMO learning is based on backpropagation, which allows easy integration in a neural network and flexibility to change the value of the p-norm. Our experiments on synthetic and real data show that using fractional p-norms for LDA leads to an improvement in accuracy compared to the traditional L_2-based LDA.

1 INTRODUCTION

Dimensionality Reduction (DR) techniques are used to obtain a lower dimensional representations of higher dimensional data, which are then used for feature extraction in pattern recognition Bishop (2006); Fukunaga (1990) and computer vision applications Belhumeur et al. (1997a); Lu et al. (2003); Moon et al. (2017). Linear Discriminant Analysis (LDA) Belhumeur et al. (1997b) is a supervised DR method that has been widely used for classification tasks, where the projection matrix of the L_2-norm LDA is obtained by maximizing the ratio of the between-class and within-class scatter matrices. However, the performance of L_2-LDA breaks down in the presence of outliers as the method tends to be dominated by samples with large norms. Many works have tried to improve upon L_2-based DR methods by using L_p-norms, where $p \leq 1$ to increase the robustness De La Torre & Black (2003); Aanæs et al. (2002); Ding et al. (2006); Markopoulos et al. (2017); Ye et al. (2018); Zhong & Zhang (2013); Wang et al. (2014).

Traditional LDA is usually solved using a generalized eigenvalue solution which can be sub-optimal. Cunningham and Ghahramani Cunningham & Ghahramani (2015) cast linear DR as an optimization over the Grassmann manifold (GM). Recently, GILDA Nagananda et al. (2020) made use of the proxy matrix optimization (PMO) to solve the L_2-norm LDA problem more effectively. In this paper, we propose the Robust L_p-norm LDA with PMO (L_p-LDA-PMO), based on the L_p-norm optimization technique as a generalized version of Fishers LDA. Our solution of LDA is based on the more general L_p-norm instead of the traditional L_2-norm to obtain a robust version of LDA. Further, non-iterative methods require that all the data be available during training and cannot gracefully handle additional data samples that are acquired incrementally. Due to the iterative nature of L_p-LDA-PMO, it is possible to adapt it so that the statistics of new data arriving incrementally are captured to compute a more accurate lower dimensional projection matrix. The main contributions of this paper are:

1. The proposed L_p-LDA-PMO method casts L_p-LDA in terms of an objective function integrated into a generalized Grassmann manifold optimization framework that does not require analytical computations of gradients.
2. L_p-LDA-PMO learning is based on backpropagation with Stochastic Gradient Descent (SGD), which enables the realization of L_p-LDA as a layer in a neural network. Using automatic differentiation allows to easily change the value of the p-norm.
3. Experiments performed on real and synthetic data show that L_p-LDA-PMO with $p < 2$ has advantages when dealing with outlier data.

2 BACKGROUND

Let $X = [x_1, x_2, \ldots, x_N] \in \mathbb{R}^{m \times N}$ be the data matrix which is m dimensional and has N data points. To find the LDA projection, the between class scatter matrix (S_B) and the within class scatter matrix (S_W) are calculated as:

$$S_W = \sum_{i=1}^{N} (x_i - \mu_{c_i})(x_i - \mu_{c_i})^T \quad \text{and} \quad S_B = \sum_{i=1}^{C} N_c (\mu_{c_i} - \mu)(\mu_{c_i} - \mu)^T, \tag{1}$$

where μ is the mean of the entire dataset and μ_{c_i} is the class mean associated with x_i, and N_c is the number of samples belonging to class c. The LDA projection matrix $R \in \mathbb{O}^{m \times p}$ (p is the lower dimensional space) aims to maximize the between-class variability while minimizing the within-class variability, which leads to minimizing the following objective,

$$f = -\frac{trace(R^T S_B R)}{trace(R^T S_W R)}. \tag{2}$$

The projection matrix R is orthogonal under this objective function. The eigenvalue solution considers the top p eigenvectors of the objective ($S_W^{-1} S_B$).

Using linear transformations, Eq. 2 is re-written in terms of the L_2-norm as,

$$R^* = \arg\max_R \frac{tr(R^T S_B R)}{tr(R^T S_W R)} \tag{3}$$

$$= \arg\max_R \frac{tr(\sum_{i=1}^{C} N_c [R^T(\mu_{c_i} - \mu)][R^T(\mu_{c_i} - \mu)]^T)}{tr(\sum_{i=1}^{N} [R^T(x_i - \mu_{c_i})][R^T(x_i - \mu_{c_i})]^T)} \tag{4}$$

$$= \arg\max_R \frac{\sum_{i=1}^{C} N_c tr([R^T(\mu_{c_i} - \mu)][R^T(\mu_{c_i} - \mu)])}{\sum_{i=1}^{N} tr([R^T(x_i - \mu_{c_i})][R^T(x_i - \mu_{c_i})])} \tag{5}$$

$$= \arg\max_R \frac{\sum_{i=1}^{C} N_c \left\| R^T(\mu_{c_i} - \mu) \right\|_2^2}{\sum_{i=1}^{N} \left\| R^T(x_i - \mu_{c_i}) \right\|_2^2}, \tag{6}$$

where $\|.\|^2$ is the L_2-norm. Thus, traditional LDA can be represented using L_2-norm. The use of the L_2-norm distance makes it very sensitive to outliers. In order to reduce the sensitivity of outliers, LDA based on L_1-norm distance has been used. By extension, the objective function can be generalised as follows,

$$f = \frac{\sum_{i=1}^{C} N_c \left\| R^T(\mu_c - \mu) \right\|_p^p}{\sum_{i=1}^{N} \left\| R^T(x_i - \mu_{c_i}) \right\|_p^p} \tag{7}$$

with $R^T R = I$ and $\|.\|^p$ is the L_p-norm for ($p > 0$).

The objective function described in Eq. 7 can face stability issues for some values of p-norms such as when $p = 1$, this is because the absolute value is not divisible at zero. To get around the issue of singular points, the sign function is used, which is described as,

$$sign(x) = \begin{cases} 1, & \text{if } x > 0, \\ 0, & \text{if } x = 0, \\ -1, & \text{if } x < 0. \end{cases} \tag{8}$$

Eq. 7 is then re-written as,

$$f = \frac{\sum_{i=1}^{C} N_c[sign(R^T(\mu_c - \mu))R^T(\mu_c - \mu)]^p}{\sum_{i=1}^{N}[sign(R^T(x_i - \mu_{c_i}))R^T(x_i - \mu_{c_i})]^p}. \tag{9}$$

The presence of outliers in the data compromises the methods that make use of L_2-norm to find the lower dimensional representation as it is more prone to the outlier points. This has lead to the introduction of the L_1-norm and more generally L_p-norm based methods for dimensionality reduction.

This analysis can be extended to L_p-LDA, where the objective function described in Eq. 9 with the generalised L_p-norm is used instead of the conventional LDA objective function Eq. 2, and the value of p can be easily changed.

Oh et al. Oh & Kwak (2013) present a method for LDA that uses the L_p-norm instead of the L_2-norm to obtain a robust and rotation- invariant version of LDA. The objective function is formulated using the general L_p-norm in both the numerator and denominator and the optimal solution is found using the steepest-gradient method. They make use of the objective function described in Eq. 9. An et al. An & Xing (2014) make use of the same objective function by performing a greedy search method to find the projection vectors one-by-one and a gradient ascent method to obtain the optimal solution for the projection matrix. Both of these methods calculate the gradient of the objective function manually.

Li et al. Li et al. (2019) present a method called bilateral L_p-norm two-dimensional linear discriminant analysis (BLp2DLDA) which is more robust to outliers and noise by using the objective function in Eq. 9. The optimization problem of BLp2DLDA can be derived through the Bayes error bound optimization. Since the proposed BLp2DLDA involves the L_p-norm operation on both its numerator and denominator, a modified gradient ascent method is used to solve it. Ye et al. Ye et al. (2018) propose a robust linear discriminant analysis via simultaneous L_s-norm distance maximization and L_p-norm distance minimization (FLDA-Lsp), which utilizes L_s- and L_p-norm to respectively measure the between- and within-class scatter matrices.

3 Mathematical Foundation

In this work, we leverage the Grassmann manifold and the operations used on the manifold are defined in this section. A point on the Grassmann manifold is a linear subspace, specified by an arbitrary orthogonal basis represented by matrices of size $n \times k$ Edelman et al. (1998).

$$G(n,k) \triangleq \{Span(X) : X \in \mathbb{R}^{n \times k}, X^T X = I_k\}$$

The tangent space at a point Y on the manifold M, ($T_Y M$) is the linear approximation of the manifold at a point and it contains all the vectors tangential to M at the point Y. More specifically, it is the set of all point X that satisfy:

$$Y^T X + X^T Y = 0_k, \tag{10}$$

where 0_k is a matrix containing all zero entries. The tangent space is pivotal for the optimization over the manifold as the direction in which the points update lies, must exist in the points tangent space as well.

The gradients of the loss function pushes the point onto the ambient Euclidean space, and thus it is not possible to restrict the calculated gradients to the manifold tangent space. For an objective function F, the gradients with respect to a point on the manifold $Y \in M$ is defined as:

$$\nabla F = \frac{\partial F}{\partial Y} - Y(\frac{\partial F}{\partial Y})^T Y^T \tag{11}$$

Since the Grassmann manifold is a subspace of the Euclidean space, not every direction of motion in ambient Euclidean space will move a point on the manifold. In order to move on the manifold during the optimization process, a small step must be taken along a direction that exists in the tangent space of the manifold at the current location. If the desired direction of motion does not exist in the tangent space of the point, the direction must first be projected onto the tangent space. The equation for the

projection of a point Z that exists in ambient Euclidean space to the tangent space of the manifold at point Y is given as:

$$\pi_{T,Y}(Z) = Y\frac{1}{2}(Y^T Z - Z^T Y) + (I_k - YY^T)Z \tag{12}$$

The projection operation eliminates the components that are normal to the tangent space and only keeps the components that are on the manifold tangent space at the current point, Y.

The retraction r_k from ambient Euclidean space (or the tangent space) to the manifold is a mapping of the point in ambient space to the closest point in the manifold. In this work, we use the retraction operation defined in Cunningham & Ghahramani (2015), which makes use of the SVD of $Z = U\Sigma V^T$,

$$r_k(Z) = UV^T. \tag{13}$$

4 METHODOLOGY

We first describe the two-step optimization approach used by Cunningham et al. Cunningham & Ghahramani (2015) for LDR. The two-step approach retracts a matrix from the ambient space onto the manifold to find the optimal projection matrix. The projection matrix R_i at iteration i is updated based on the gradients that produce Z_{i+1} which is in the ambient Euclidean space at the next iteration. The point Z_{i+1} is projected onto the tangent space using Eq.12. The point is then retracted from the tangent space onto the manifold using Eq.13.

L_p-LDA-PMO is based on the proxy matrix optimization (PMO) which combines the manifold retraction into the optimization function unlike the two-step which does retraction after optimization. PMO uses an auxiliary or proxy matrix in ambient space which is retracted to the closest location on the manifold using Eq. 13 instead of directly optimizing over the manifold. The PMO process is illustrated in Fig. 1 and the corresponding L_p-LDA-PMO algorithm is outlined in Algorithm 1. The first step in the PMO process is to retract the proxy matrix, P_i to Y_i, its closest location on the manifold. Once the proxy matrix is retracted to the manifold, the loss is calculated based on the loss function at Y_i. This loss is then back-propagated through the singular value decomposition of proxy matrix using a method developed by Ionescu et al. (2015) to a new point P_{i+1}. This point is then retracted back onto the manifold using Eq. 13 to point Y_{i+1}. PMO leverages the autograd routine in Pytorch to back-propagate through the SVD and hence removing the need for any analytical gradient calculation.

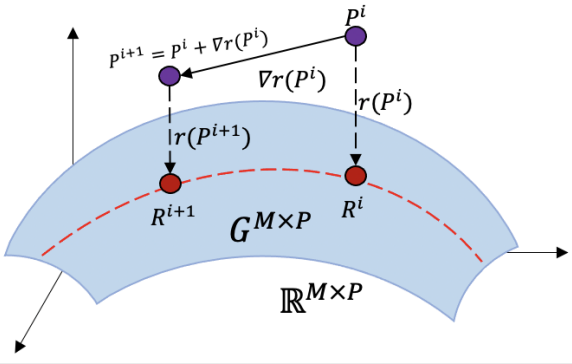

Figure 1: Illustration of the proxy matrix optimization method.

5 L_p-LDA-PMO COMPLEXITY AND CONVERGENCE

In this section, we provide a high level discussion on the convergence and complexity of L_p-LDA-PMO.

Algorithm 1: Proxy Matrix Optimization for L_p-LDA.

Data: $X \in \mathbb{R}^{M \times P}$
Result: Locally Optimal P such that $r(P_i)$ minimizes the loss f_X from Eq. 9
initialize $P \in \mathcal{R}^{m \times p}$ and $P \notin \mathcal{G}^{m \times p}$;
for $i > iter$ **do**
  $USV^T = P_i$; /* SVD */
  $r(P_i) = UV^T$; /* Retract P_i to $\mathcal{G}^{m \times p}$ */
  $\nabla r(P_i) = \frac{\partial}{\partial P_i} f_X(r(P_i))$; /* Calculate gradients for P_i using Eq.11 */
  $P_{i+1} = P_i - \beta \nabla r(P_i)$; /* Update P */

5.1 Complexity

L_p-LDA-PMO has three steps that are performed in every epoch. The first is calculating the means that are used in the objective function Eq. 7, which involves calculating the p-norms. The next step is to retract the projection matrix from Euclidean space to the closest point on the manifold. The final step is to calculate and backpropagate the loss function. We let the number of classes be C and the total number of samples be N. The input dimension of the data is D and the lower dimension to which the data is projected is d, which is $C - 1$ for LDA. We consider the number of samples N to be a lot greater than the number of classes C. Under this assumption, the computational complexity of calculating the numerator and denominator of Eq. 7 is $\mathcal{O}(NDp)$. The computational complexity of the second step, which is an SVD, is $\mathcal{O}(D^2d + d^3)$. Lastly, the complexity of the third step which involves taking the partials has a computational complexity of $\mathcal{O}(LDd)$. Thus, the overall complexity is $\mathcal{O}(2NDp + D^2d + d^3 + LDd)$.

5.2 Convergence

The two-step method is guaranteed to converge Cunningham & Ghahramani (2015) due to the fact that the gradients in the ambient Euclidean space can be decomposed into their normal and tangential components. The tangential components lie in the tangent space to the manifold. As the components are orthogonal to one another, setting the normal component to zero does not affect the tangential component. The gradients are defined to be in the loss minimizing direction and as a result, the tangential components of the gradients also lie in the loss minimizing direction. The points on the tangent space are then retracted to the manifold, which by definition of retraction is the closest point on the manifold and thus lies on the loss minimizing geodesic of the manifold.

In PMO, the optimization problem is setup for solving

$$R = \underset{R \in \mathbb{R}^{m \times p}}{\arg\min} F(UV^T)$$

where, $R = U\Sigma V^T$ is its SVD and F is the objective function Eq 9. Algorithm 1, and the definition of retraction indicate that retraction is embedded in the objective function of L_p-LDA-PMO. The derivative of the objective with respect to the proxy matrix as defined in Algorithm 1 is done using the method described in Ionescu et al. (2016). Since the derivative is taken with respect to the proxy matrix, moving the gradient in the loss minimizing direction will also move the retracted point along the loss minimizing geodesic on the manifold. This optimization will lead to the optimal value of the projection matrix, R. Unlike the two-step method, the search space is not restricted to the manifold and the magnitude of each optimization step is not limited. If the L_p-LDA-PMO method is initialised with the eigenvalue solution, the number of iterations taken to reach the minimum is lower than the two-step method due to this unconstrained optimization and the removal of the projection to the tangent space step.

6 RESULTS

We test out L_p-LDA-PMO on both toy and real world datasets. Each of the datasets have a certain percentage of outliers in the training data. The metric used for comparison is the classification accuracy for each of the p-norms for L_p-LDA.

6.1 TOY DATA EXPERIMENTS

The toy-data datasets are all generated using a Gaussian distribution. We show the results for two synthetic datasets described below:

TWO-DIMENSIONAL DATA

In this experiment, we make a synthetic two-dimensional dataset with two-classes with covariance matrices [0.05, 0; 0, 2]. There are 20 samples in the positive class and 19 in the negative class. An outlier data point with coordinates [10, 10] is designated to the negative class. The entire dataset of 40 points is used for training the model and finding the projection onto one dimension. The optimal projection of the dataset without the outlier point is [1, 0] or the x-axis. However, the optimal projection vectors for various p-norm values for LDA is shown in Fig. 2.

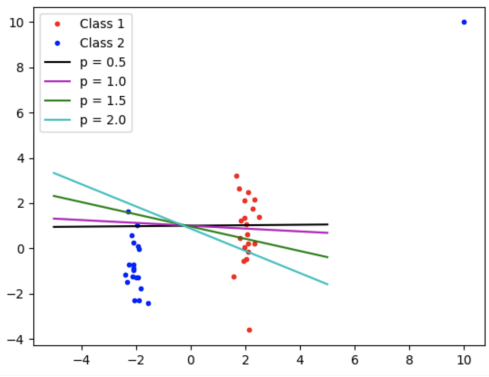

Figure 2: Projection vectors for the various p-norm values for L_p LDA on the two-dimensional dataset.

The line corresponding to the p-norm of 0.5 is closest to the optimal projection line while the slope decreases with increasing p-norms showing that smaller norms work better for outlier data.

THREE-DIMENSIONAL DATA

The three-dimensional dataset contains 300 data points in three classes. The covariance matrix of all the classes is [0.1, 1, 1; 0, 1, 0;, 0, 0, 0.1] with mean vectors of [0, 0, 0], [0, 4, 0], and [2, 2, 0]. Each class contains 100 datapoints and 20 data points from each class are chosen to construct the training set. The remaining 80 points are used as a test dataset. An outlier point [100, 100, 0] is used to substitute for one point in the first class of the training set. The dataset is projected onto two dimensions and the classification accuracy of the test set is obtained for each of the p-norm values of L_p-LDA. The 2D projection of the data for each of the p-norms along with the testing accuracy is shown in Fig. 3.

In the presence of outliers, L_2-LDA does not make the best lower dimensional representation as shown by the projection plots and classification accuracy. Lower values of p-norms tend to perform better and make more discriminate projections.

6.2 IRIS+NOISE EXPERIMENTS

In this experiment, we make use of the Iris dataset from the UCI database Dua & Graff (2017) and add random Gaussian noise to the training data split. The projection matrix for each of the p-norms

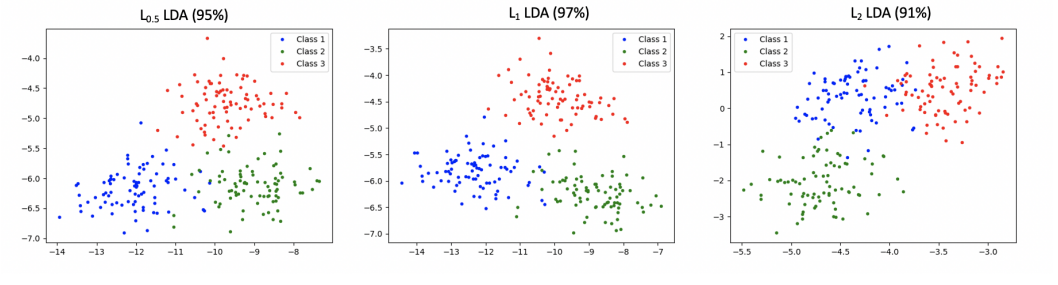

Figure 3: Projection vectors for the various p-norm values for L_p LDA on the three-dimensional dataset with testing accuracy in parenthesis.

[0.5, 1, 1.5, 2] are computed and the final test accuracy of the test dataset is recorded for each of the two methods L_p-LDA-PMO and BLp2DLDA Li et al. (2019). Fig. 4 shows that L_p-LDA-PMO does a lot better than BLp2DLDA for all the p-norms.

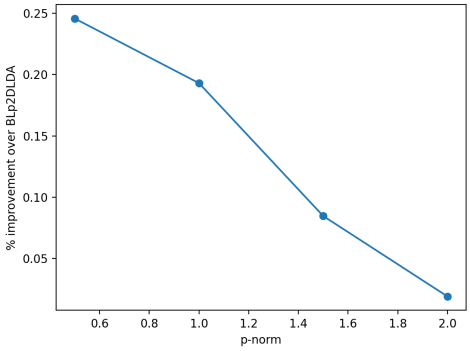

Figure 4: Comparison of the performance of L_p-LDA-PMO and BLp2DLDA Li et al. (2019) for the Iris+noise dataset.

6.3 Face Recognition Experiments

For the face recognition experiments, we make use of two datasets, the ORL database and the Yale database. Two types of noises are used to make the outlier data; salt and pepper noise (S&P), and Gaussian noise (Occlusion). For experiments with Gaussian noise, patches of random uniform noise that cover 30% of the total face are randomly placed on the training dataset. For the salt and pepper outliers, the entire image is subjected to salt and pepper noise of density 0.1. The percentage of outlier data for the training set is varied and the corresponding classification accuracies are noted for each of the p-norms in L_p-LDA-PMO. Examples of the Gaussian and salt and pepper noise are shown in Fig.5.

ORL Database Experiments

The ORL database consists of 400 images from 40 distinct classes. All the images are taken against a dark homogeneous background with the subjects in an upright, frontal position. Each image is of size 119x92 pixels which are resized down to 32x32 for the experiments. 70% of the images for each subject is taken to form the train dataset and the remaining images form the test set. [0, 5, 10, 50, 90]% of the training set is subjected to outliers in the form of Gaussian noise or salt and pepper noise. The projected dimension is 39 and the classification accuracy on the test dataset is calculated using an SVM for each of the $p = 1$ and $p = 2$ norms. The results from the experiment are in Table 1.

Figure 5: Example of corruption with Gaussian noise patch (top) and salt and pepper noise (bottom) for faces in the Yale dataset.

YALE DATABASE EXPERIMENTS

The Yale faces database consists of 165 images of 15 classes and each image is 225x195 pixels. Each class has 11 instances which are resized to 32x32 pixels. This database is used as it contains images with different facial expressions and lighting conditions. A random set of 10 images from each class is taken as the training dataset and the rest of the database is used as the test set. Outliers in the form of Gaussian and salt and pepper noise are added to the training set in the same procedure as the ORL database. The projected dimension in this case is 14 and the classification accuracy on the test dataset is calculated using an SVM for each of the $p = 1$ and $p = 2$ norms. The results from the experiment are in Table 2.

Table 1: Accuracy results of L_p-norm LDA with $p = 1$ and $p = 2$ norms from the ORL experiments along with the % improvement of the error of the L_1-LDA result over L_2-LDA.

	Occlusion			S&P		
Outlier %	p = 1	p = 2	% imp	p = 1	p = 2	% imp
0 %	1.00	0.99		1.00	0.99	
5 %	0.99	0.96	75	0.99	0.99	0
10 %	0.98	0.95	60	0.99	0.99	0
50 %	0.86	0.80	30	0.91	0.85	40
90 %	0.79	0.76	12.5	0.65	0.63	5.4

Table 2: Accuracy results of L_p-norm LDA with $p = 1$ and $p = 2$ norms from the Yale Faces experiments along with the % improvement of the error of the L_1-LDA result over L_2-LDA.

	Occlusion			S&P		
Outlier %	p = 1	p = 2	% imp	p = 1	p = 2	% imp
0 %	1.00	0.80		1.00	0.80	
5 %	0.85	0.79	28.57	0.88	0.76	50
10 %	0.85	0.79	28.57	0.82	0.82	0
50 %	0.79	0.76	12.5	0.82	0.79	14.28
90 %	0.70	0.64	16.66	0.76	0.73	11.11

From the faces experiments, as the percentage of outliers increases, the over-all test accuracy decreases. The advantage of using a smaller p-norm becomes apparent for higher outlier percentages. In each of the cases, using L_p-norm does better than or the same as using L_2-norm LDA.

7 CONCLUSION

We presented the robust L_p-LDA-PMO for fractional p-norm LDA using the flexible PMO framework. We also discussed L_p-LDA-PMO convergence and its advantage over the two-step optimization method. Our results illustrate the advantage of L_p-LDA-PMO compared to the conventional L_2-LDA when dealing with outliers in syntetic and real data. We have illustrated the advantage of L_p-LDA-PMO when using toy data and real-world data using the ORL and Yale datasets for the faces experiments with and without noise.

ACKNOWLEDGEMENTS

This research was partly supported by the Air Force Office of Scientific Research (AFOSR) under Dynamic Data Driven Applications Systems (DDDAS) grant FA9550-18-1-0121 and the National Science Foundation award number 1808582. We would also like to thank the reviewers for their comments.

REFERENCES

Henrik Aanæs, Rune Fisker, Kalle Astrom, and Jens Michael Carstensen. Robust factorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1215–1225, 2002.

L. An and H. Xing. Linear discriminant analysis based on Z_p-norm maximization. In *Proceedings of 2nd International Conference on Information Technology and Electronic Commerce*, pp. 88–92, 2014. doi: 10.1109/ICITEC.2014.7105578.

P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997a. doi: 10.1109/34.598228.

P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997b. doi: 10.1109/34.598228.

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

John P. Cunningham and Zoubin Ghahramani. Linear dimensionality reduction: Survey, insights, and generalizations. *Journal of Machine Learning Research*, 16(89):2859–2900, 2015. URL http://jmlr.org/papers/v16/cunningham15a.html.

Fernando De La Torre and Michael J Black. A framework for robust subspace learning. *International Journal of Computer Vision*, 54(1-3):117–142, 2003.

Chris Ding, Ding Zhou, Xiaofeng He, and Hongyuan Zha. R_1-PCA: rotational invariant L_1-norm principal component analysis for robust subspace factorization. In *Proceedings of the 23rd international conference on Machine learning*, pp. 281–288. ACM, 2006.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with orthogonality constraints. *SIAM journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.

Keinosuke Fukunaga. *Introduction to Statistical Pattern Recognition (2nd Ed.)*. Academic Press Professional, Inc., USA, 1990. ISBN 0122698517.

C. Ionescu, O. Vantzos, and C. Sminchisescu. Matrix backpropagation for deep networks with structured layers. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 2965–2973, 2015. doi: 10.1109/ICCV.2015.339.

Catalin Ionescu, Orestis Vantzos, and Cristian Sminchisescu. Training deep networks with structured layers by matrix backpropagation, 2016.

Chun-Na Li, Yuan-Hai Shao, Zhen Wang, and Nai-Yang Deng. Robust bilateral L_p-norm two-dimensional linear discriminant analysis. *Information Sciences*, 500:274 – 297, 2019. ISSN 0020-0255. doi: https://doi.org/10.1016/j.ins.2019.05.066. URL http://www.sciencedirect.com/science/article/pii/S0020025519304840.

Juwei Lu, Konstantinos Plataniotis, and Anastasios Venetsanopoulos. Face recognition using LDA-based algorithms. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, 14:195–200, 02 2003. doi: 10.1109/TNN.2002.806647.

Panos P Markopoulos, Sandipan Kundu, Shubham Chamadia, and Dimitris A Pados. Efficient L_1-norm principal-component analysis via bit flipping. *IEEE Transactions on Signal Processing*, 65(16):4252–4264, 2017.

Kevin R. Moon, David van Dijk, Zheng Wang, William Chen, Matthew J. Hirn, Ronald R. Coifman, Natalia B. Ivanova, Guy Wolf, and Smita Krishnaswamy. PHATE: A dimensionality reduction method for visualizing trajectory structures in high-dimensional biological data. *bioRxiv*, 2017. doi: 10.1101/120378. URL https://www.biorxiv.org/content/early/2017/03/24/120378.

Navya Nagananda, Breton Minnehan, and Andreas Savakis. Grassmann Iterative Linear Discriminant Analysis with Proxy Matrix Optimization . *NeurIPS workshop on Differential Geometry meets Deep Learning*, 2020.

Jae Hyun Oh and Nojun Kwak. Generalization of linear discriminant analysis using L_p-norm. *Pattern Recognition Letters*, 34(6):679 – 685, 2013. ISSN 0167-8655. doi: https://doi.org/10.1016/j.patrec.2013.01.016. URL http://www.sciencedirect.com/science/article/pii/S0167865513000202.

H. Wang, X. Lu, Z. Hu, and W. Zheng. Fisher discriminant analysis with L_1-norm. *IEEE Transactions on Cybernetics*, 44(6):828–842, 2014. doi: 10.1109/TCYB.2013.2273355.

Q. Ye, J. Yang, F. Liu, C. Zhao, N. Ye, and T. Yin. L_1-norm distance linear discriminant analysis based on an effective iterative algorithm. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(1):114–129, 2018. doi: 10.1109/TCSVT.2016.2596158.

Qiaolin Ye, Liyong Fu, Zhao Zhang, Henghao Zhao, and Meem Naiem. L_p- and L_s-norm distance based robust linear discriminant analysis. *Neural Networks*, 105:393 – 404, 2018. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2018.05.020. URL http://www.sciencedirect.com/science/article/pii/S0893608018301825.

F. Zhong and J. Zhang. Linear discriminant analysis based on L_1-norm maximization. *IEEE Transactions on Image Processing*, 22(8):3018–3027, 2013. doi: 10.1109/TIP.2013.2253476.