L1-norm Optimization in Subspace Learning Methods

Presented @ 2012 Korean CVPR Workshop

Nojun Kwak
nojunk@ajou.ac.kr
http://image.ajou.ac.kr

Department of Electrical Engineering
Ajou University, Korea

Nov. 2, 2012

Paper References

This presentation is mostly based on the two papers:

- Nojun Kwak, "Principal component analysis based on L1 norm maximization", IEEE TPAMI, vol. 30, no. 9, pp. 1672 – 1680, Sep. 2008.
- Nojun Kwak and Jiyong Oh, "Feature extraction for one-class classification problem: Enhancements to biased discriminant analysis", Pattern Recognition, vol. 42, no. 1, pp. 17 – 26, Jan. 2009.

Outline

- Introduction
  - L2-PCA: Two interpretations
  - Other flavors of PCA (L1-PCA, R1-PCA)
- PCA-L1
  - PCA-L1: Problem formulation
  - PCA-L1: Algorithm
  - PCA-L1: Examples & Experimental Results
- L1-BDA
  - L1-BDA: Application of the Theorem to BDA
  - L1-BDA: Experimental Results
- Conclusions and future works

Introduction

- PCA (Principal Component Analysis) [1]
  - Dimensionality reduction technique
  - Data visualization
  - Face recognition (eigenface)
  - A lot of applications
- Pros and Cons of PCA
  - Computationally efficient (SVD)
  - Prone to outliers
  - Instead of L2-norm, L1-norm is used here.

Notations

- $X = [\boldsymbol{x}_1, \cdots \boldsymbol{x}_n] \in \Re^{d \times n}$: dataset
  - d: dimension of input space
  - n: number of samples
  - $\{\boldsymbol{x}_i\}_{i=1}^n$ is assumed to have zero mean.
- $W \in \Re^{d \times m}$: projection matrix
  - m: dimension of feature space (no. of features to be extracted)
  - $\{\boldsymbol{w}_k\}_{k=1}^m$: set of m projection vectors
- $V \in \Re^{m \times n}$: coefficient matrix
  - $V = W^T X$
  - v_{ij}: ith coordinate of $\boldsymbol{x}_j$ after projection
- $E = X - WV = (I_d - WW^T)X$: error matrix in the original input space

Two interpretations of the conventional L2-PCA I

1. Minimizing the error of the projection (in L2-norm)

$$W^* = \underset{W}{\operatorname{argmin}}\, E_2(W) \tag{1}$$

subject to $W^T W = I_m$, where,

$$\begin{aligned} E_2(W) &= ||X - WV||_F^2 = \sum_{i=1}^{n} ||\boldsymbol{x}_i - \sum_{k=1}^{m} \boldsymbol{w}_k v_{ki}||_2^2 \\ &= \sum_{i=1}^{n} \sum_{j=1}^{d} (x_{ji} - \sum_{k=1}^{m} w_{jk} v_{ki})^2 \end{aligned} \tag{2}$$

Two interpretations of the conventional L2-PCA II

2. Maximizing the dispersion of the projection (in L2-norm)

$$W^* = \underset{W}{\operatorname{argmax}}\, D_2(W) \tag{3}$$

subject to $W^T W = I_m$, where

$$\begin{aligned} D_2(W) &= \sum_{i=1}^{n} ||W^T \boldsymbol{x}_i||_2^2 = \sum_{i=1}^{n}\sum_{k=1}^{m} (\boldsymbol{w}_k^T \boldsymbol{x}_i)^2 \\ &= ||W^T X||_F^2 = tr(W^T S_x W) \end{aligned} \tag{4}$$

$S_x = XX^T$: scatter matrix of X.

- The two are equivalent!! (Dual)

  $\rightarrow$ solved by EVD (of S_x) or SVD (of X).

PCA utilizing other norms than L2 I

- L2 norm (Frobenius) is prone to outliers.
- Instead of L2, use other norm in the optimization

1. Minimization of L1 projection error (input space) [2] – [4]

$$W^* = \underset{W}{\operatorname{argmin}}\, E_1(W) \quad \text{subject to} \quad WW^T = I_m. \tag{5}$$

$$\begin{aligned} E_1(W) = ||X - WV||_1 &= \sum_{i=1}^{n} ||\boldsymbol{x}_i - \sum_{k=1}^{m} \boldsymbol{w}_k v_{ki}||_1 \\ &= \sum_{i=1}^{n} \sum_{j=1}^{d} |x_{ji} - \sum_{k=1}^{m} w_{jk} v_{ki}|. \end{aligned} \tag{6}$$

- Not invariant to rotations of the input space.
- Exact solution of (5) is hard to achieve. (iterative solutions)

PCA utilizing other norms than L2 II

2. Minimization of R1-norm of the projection error [5]

$$W^* = \underset{W}{\operatorname{argmin}} \, E_{R1}(W) \quad \text{subject to} \quad WW^T = I_m. \tag{7}$$

$$E_{R1}(W) = ||X - WV||_{R1} \triangleq \sum_{i=1}^{n} \left(\sum_{j=1}^{d} (x_{ji} - \sum_{k=1}^{m} w_{jk} v_{ki})^2 \right)^{\frac{1}{2}}. \tag{8}$$

- R1-norm is a hybrid of L1 and L2.
- Optimal solution depends on m i.e., $W^{1*} \neq W^{2*}$.
- Solution is not intuitive, not exact. (Huber's M-estimator is used.)

Formulation of PCA-L1 [6] I

- Motivation:
  - Previous methods (L1-PCA, R1-PCA) minimizes projection error (E, 1st interpretation).
  - Instead of solving minimization problem, **maximize the dispersion of projection (D, 2nd interpretation).**
- Problem formulation

$$W^* = \underset{W}{\operatorname{argmax}}\, D_1(W) \quad \text{subject to} \quad W^T W = I_m \qquad (9)$$

$$D_1(W) = \sum_{i=1}^{n} ||W^T \boldsymbol{x}_i||_1 = \sum_{i=1}^{n} \sum_{k=1}^{m} |\boldsymbol{w}_k^T \boldsymbol{x}_i| \qquad (10)$$

  - $W^T W = I_m$: to ensure orthonormality of the projection vectors. (not necessary)
  - (9) and (5) are not equivalent!

Formulation of PCA-L1 [6] II

- Pros and Cons of (9)
  - (9) are invariant to rotations.
  - As R1-PCA, the solution depends on m.
- Modified problem: $m = 1$

$$\boldsymbol{w}^* = \underset{\boldsymbol{w}}{\operatorname{argmax}} ||\boldsymbol{w}^T X||_1 = \underset{\boldsymbol{w}}{\operatorname{argmax}} \sum_{i=1}^{n} |\boldsymbol{w}^T \boldsymbol{x}_i| \qquad (11)$$

$$\text{subject to} \quad ||\boldsymbol{w}||_2 = 1.$$

Algorithm: PCA-L1

1. Initialization: Pick any $\boldsymbol{w}(0)$. Set $\boldsymbol{w}(0) \leftarrow \boldsymbol{w}(0)/||\boldsymbol{w}(0)||_2$ and $t = 0$.
2. Polarity check: For all $i \in \{1, \cdots, n\}$, if $\boldsymbol{w}^T(t)\boldsymbol{x}_i < 0$, $p_i(t) = -1$, otherwise $p_i(t) = 1$.
3. Flipping and maximization: Set $t \leftarrow t + 1$ and $\boldsymbol{w}(t) = \sum_{i=1}^{n} p_i(t-1)\boldsymbol{x}_i$. Set $\boldsymbol{w}(t) \leftarrow \boldsymbol{w}(t)/||\boldsymbol{w}(t)||_2$.
4. Convergence check:
   a. If $\boldsymbol{w}(t) \neq \boldsymbol{w}(t-1)$, go to Step 2.
   b. Else if there exists i such that $\boldsymbol{w}^T(t)\boldsymbol{x}_i = 0$, set $\boldsymbol{w}(t) \leftarrow (\boldsymbol{w}(t) + \Delta\boldsymbol{w})/||\boldsymbol{w}(t) + \Delta\boldsymbol{w}||_2$ and go to Step 2. Here, $\Delta\boldsymbol{w}$ is a small nonzero random vector.
   c. Otherwise, set $\boldsymbol{w}^\star = \boldsymbol{w}(t)$ and stop.

Optimality of the Algorithm I

Theorem

With the above PCA-L1 procedure, the projection vector $\boldsymbol{w}$ *converges to* $\boldsymbol{w}^\star$*, which is a local maximum point of* $\sum_{i=1}^n |\boldsymbol{w}^T\boldsymbol{x}_i|$.

Proof.

Firstly, we can show that $\sum_{i=1}^n |\boldsymbol{w}^T(t)\boldsymbol{x}_i|$ is a non-decreasing function of t as the following:

$$\sum_{i=1}^n |\boldsymbol{w}^T(t)\boldsymbol{x}_i| = \boldsymbol{w}^T(t)(\sum_{i=1}^n p_i(t)\boldsymbol{x}_i) \geq \boldsymbol{w}^T(t)(\sum_{i=1}^n p_i(t-1)\boldsymbol{x}_i)$$
$$\geq \boldsymbol{w}^T(t-1)(\sum_{i=1}^n p_i(t-1)\boldsymbol{x}_i) = \sum_{i=1}^n |\boldsymbol{w}^T(t-1)\boldsymbol{x}_i|. \quad (12)$$

$\therefore \boldsymbol{w}(t) \rightarrow \boldsymbol{w}^*$ in finite number of iterations. □

Optimality of the Algorithm II

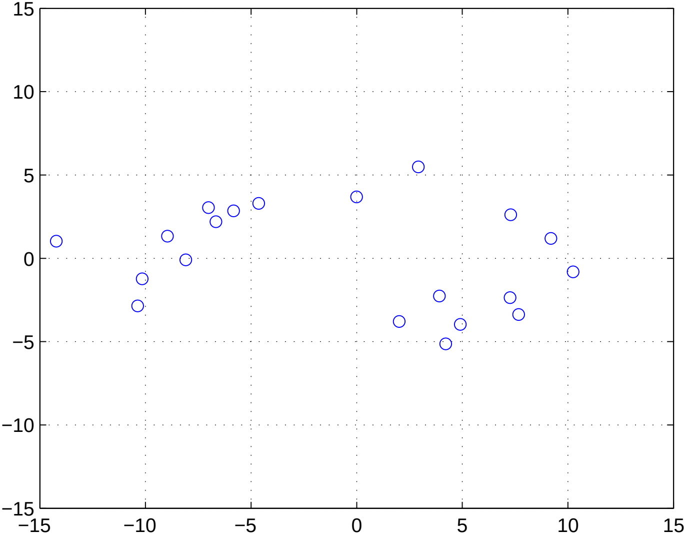
Proof.

Local maximality of $\boldsymbol{w}^\star$:

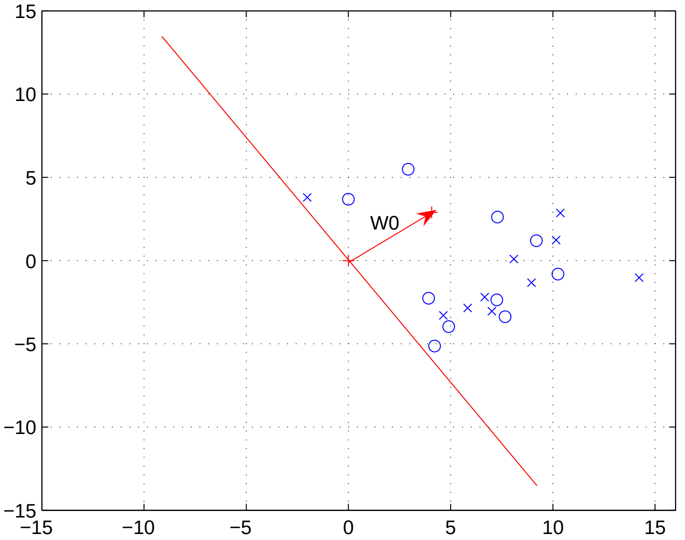
1. $\boldsymbol{w}^{\star T} p_i(t)\boldsymbol{x}_i \geq 0 \ \forall i.$ $\because \boldsymbol{w}(t)$ converges to $\boldsymbol{w}^\star$
2. $\exists$ a small neighbor $N(\boldsymbol{w}^\star)$ of $\boldsymbol{w}^\star \ni$ if $\boldsymbol{w} \in N(\boldsymbol{w}^\star)$, then $\boldsymbol{w}^T p_i(t)\boldsymbol{x}_i \geq 0 \ \forall i.$
   - Finite number of data points.
   - $\boldsymbol{w}^{\star T}\boldsymbol{x}_i \neq 0 \ \forall i.$ (Step 4b)
3. $\sum_{i=1}^n |\boldsymbol{w}^{\star T}\boldsymbol{x}_i| > \sum_{i=1}^n |\boldsymbol{w}^T\boldsymbol{x}_i| \ \forall \boldsymbol{w} \in N(\boldsymbol{w}^\star)$
   - $\because \boldsymbol{w}^\star \parallel \sum_{i=1}^n p_i(t)\boldsymbol{x}_i$
   - $\rightarrow \boldsymbol{w}^\star$ is a local maximum point.

Therefore, the PCA-L1 procedure finds a local maximum point $\boldsymbol{w}^\star$. □

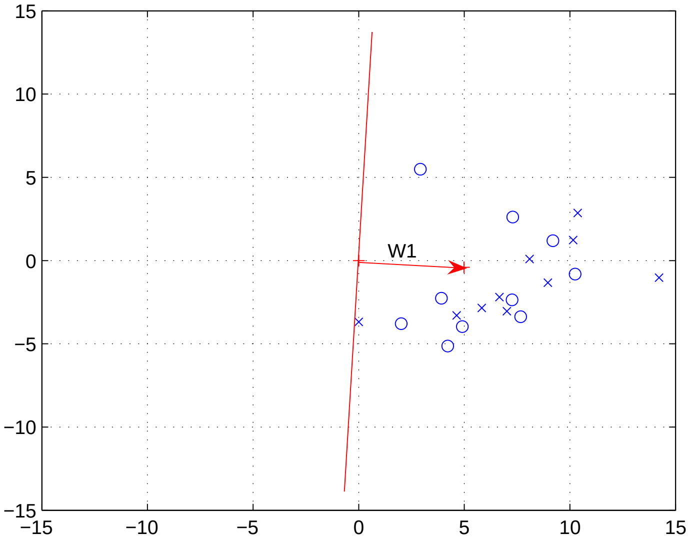
Simple Example (2D case)



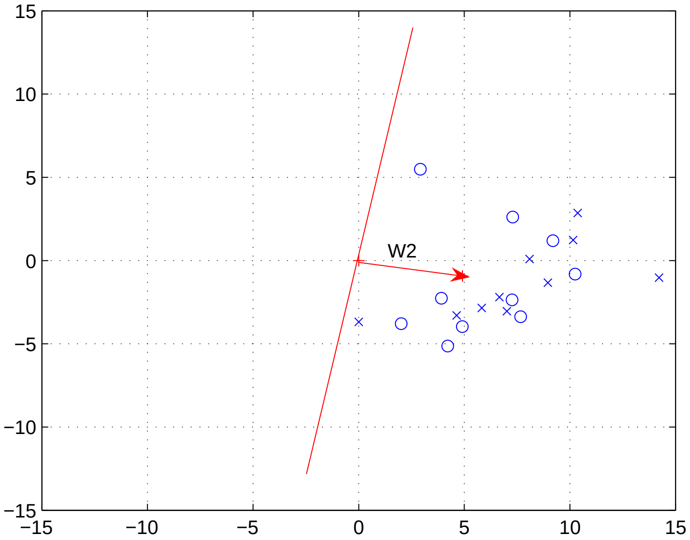
Simple Example (2D case)



Simple Example (2D case)



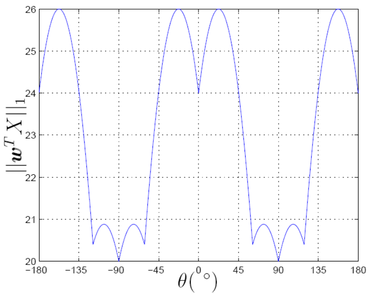
Simple Example (2D case)



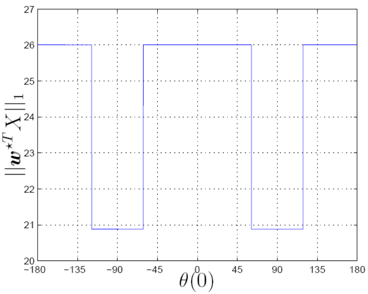
Dependency on Initial Projection Vector $\boldsymbol{w}(0)$

- An example with 5 data points (in 2D)

$$X = \begin{bmatrix} 0 & 9 & -9 & 3 & -3 \\ 10 & -5 & -5 & 0 & 0 \end{bmatrix}.$$



(a) Objective function



(b) Dependency on initial vector $\boldsymbol{w}(0)$ $(\theta(0))$

- Final vector $\boldsymbol{w}^\star$ depends on initial point $\boldsymbol{w}(0)$.
  - Start with various initial points.
  - Set $\boldsymbol{w}(0) = \boldsymbol{w}_{L2-PCA}$.

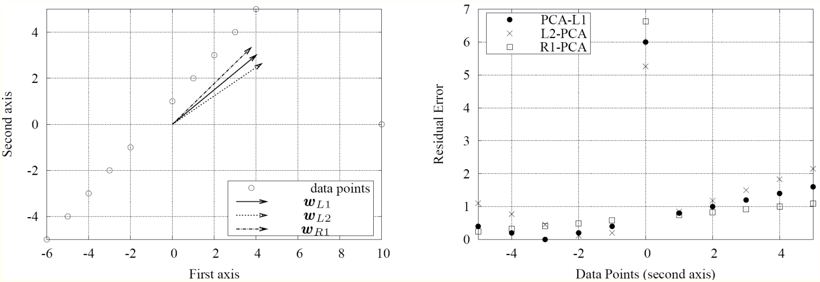
(Greedy search algorithm)

1. $\boldsymbol{w}_0 = \mathbf{0}, \{\boldsymbol{x}_i^0 = \boldsymbol{x}_i\}_{i=1}^n$.
2. For $j = 1$ to m,
   a. $\forall i \in \{1, \cdots, n\}, \boldsymbol{x}_i^j = \boldsymbol{x}_i^{j-1} - \boldsymbol{w}_{j-1}(\boldsymbol{w}_{j-1}^T \boldsymbol{x}_i^{j-1})$.
   b. In order to find $\boldsymbol{w}_j$, apply the PCA-L1 procedure to $X^j = [\boldsymbol{x}_1^j, \cdots, \boldsymbol{x}_n^j]$.

- Orthogonality is not necessary.
- **However**, to limit the search space, orthogonality is assumed.

Comparison with other versions of PCA

- **A toy problem with an outlier**



- At a glance, R1-PCA seems to remove the effect of outliers most effectively.
- However, the average residual error is

	L2-PCA	R1-PCA	PCA-L1
Average Residual Error	1.401	1.206	**1.200**

Classification Results (UCI datasets)

Table: Average Classification Rates on UCI datasets (%). The last column is the averages of the best classification rates among the cases where the number of extracted features was one to half the number of original inputs

No. of extracted features	1	2	3	4	Best performance
L2-PCA	62.49	68.59	73.36	76.79	76.46
R1-PCA	62.44	68.49	73.63	76.52	76.47
PCA-L1	63.94	71.88	74.90	77.43	**78.15**

Face Reconstruction Results

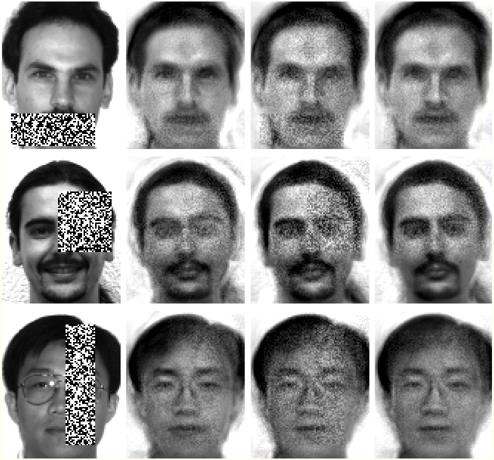


Figure: Face images with occlusion and the reconstructed faces: 1st column: original, 2nd column: PCA-L1, 3rd column: L2-PCA, 4th column: R1-PCA. (reconstructed with 20 projection vectors)

Application to BDA: L1-BDA [7]

- BDA (Biased Discriminant Analysis)
  - One-class classification problem
  - Maximizes the ratio between the positive scatter and negative scatter matrices.

$$W^* = \underset{W}{\operatorname{argmax}} \frac{tr(W^T S_y W)}{tr(W^T S_x W)}. \tag{13}$$

$$S_x = \sum_{i=1}^{n_x} (\boldsymbol{x}_i - \boldsymbol{m}_{\boldsymbol{x}})(\boldsymbol{x}_i - \boldsymbol{m}_{\boldsymbol{x}})^T \tag{14}$$

$$S_y = \sum_{i=1}^{n_y} (\boldsymbol{y}_i - \boldsymbol{m}_{\boldsymbol{x}})(\boldsymbol{y}_i - \boldsymbol{m}_{\boldsymbol{x}})^T, \tag{15}$$

- $\{\boldsymbol{x}_i\}_{i=1}^{n_x}$: positive samples
- $\{\boldsymbol{y}_i\}_{i=1}^{n_y}$: negative samples
- $\boldsymbol{m}_{\boldsymbol{x}}$: mean of the positive samples

Algorithm: L2-BDA

1. Sphering

   a. Solve the EVD problem $S_x U = U\Lambda_1$.

   b. Scale each column $\boldsymbol{u}_n$ of U to make $\{\hat{x}_{in} = \hat{\boldsymbol{u}}_n^T(\boldsymbol{x}_i - \boldsymbol{m}_{\boldsymbol{x}})\}_{i=1}^{n_x}$ have unit variance.
   $\hat{\boldsymbol{u}}_n$: a scaled version of $\boldsymbol{u}_n$
   If $||\boldsymbol{u}_n|| = 1$, this is equivalent to setting $\hat{\boldsymbol{u}}_n = \boldsymbol{u}_n/\sqrt{\lambda_{1n}}$, where λ_{1n} denotes the n-th eigenvalue of S_x.

2. Maximization

   a. Find M weight vectors $\{\boldsymbol{v}_n\}_{n=1}^{M}$ that maximizes the following objective function:

$$V = \underset{W}{\text{argmax}}\, |W^T \hat{S}_y W|, \quad \text{subject to} \quad W^T W = I, \qquad (16)$$

   where $\hat{S}_y = \hat{U}^T S_y \hat{U}$.

   b. Output $W = \hat{U}V$.
   Here, $\hat{U} = [\hat{\boldsymbol{u}}_1, \hat{\boldsymbol{u}}_2, \cdots]$.

Sphering Operation in BDA

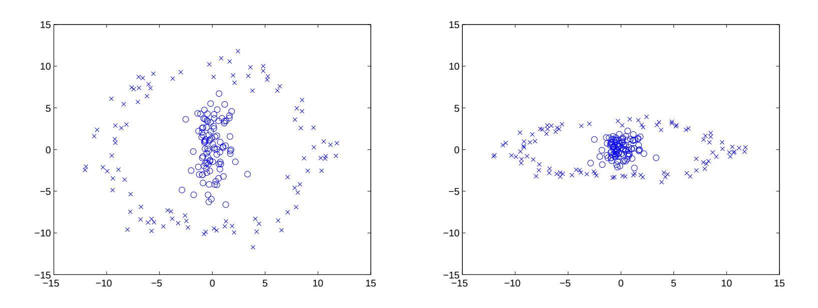


Figure: Data distribution before and after sphering process

L1-BDA

- Problem reformulation: L1-BDA

$$W^* = \underset{W}{\operatorname{argmax}} \frac{(\sum_{i=1}^{n_y} |W^T \boldsymbol{y}_i|)^2}{tr(W^T S_x W)}. \tag{17}$$

- Solution: 2-step (Sphering $\rightarrow$ Maximization)
  - Replace L2-norm with L1-norm in the Maximization part (after sphering)
  - The L1-norm maximization technique developed for L1-PCA can be directly utilized for maximizing the numerator.

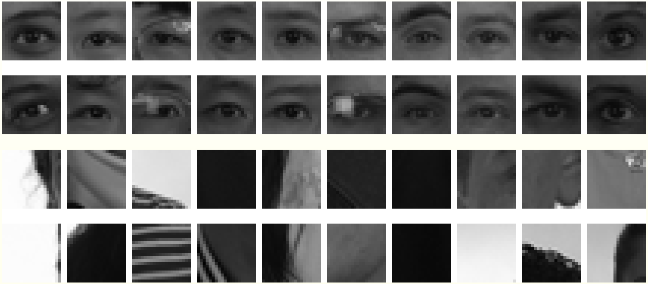


Figure: Eye and noneye samples for training

Table: Classification rates for FERET Eye dataset. (1-NN)

# of features	LDA	Cher. LDA	BDA	SBDA ($\gamma = 1$)	L1BDA	SL1BDA ($\gamma = 1$)
1	**86.875**	**84.625**	81.875	86.25	87.625	86.625
2	–	83.75	88	95	91.125	94.5
3	–	81.375	91.25	96.625	96.375	96.75
4	–	78.75	93	97.5	97.125	97.625
5	–	80.625	93	98.25	97.375	98.25
6	–	79.75	94.875	**98.625**	97.375	**98.625**
7	–	78	94.75	98.5	97.875	98.125
8	–	79.125	95	98.125	**98**	98.125
9	–	78.625	95	98.375	98	98.5
10	–	79.75	**95.875**	98	97.875	98.125

Experimental Result of L1-BDA (UCI)

Table: Experimental Results on UCI datasets (1-NN classifier)

Data set	LDA	Chernoff LDA	BDA	SBDA ($\gamma = 1$)	L1BDA	SL1BDA ($\gamma = 1$)
Austrailian	80.55±1.02 (1)	81.35±0.82 (6)	80.68±1.28 (3)	81.29±0.74 (4)	81.57±1.19 (4)	**81.67±1.19 (3)**
Balance	88.00±0.49 (2)	87.68±0.79 (2)	91.25±0.99 (3)	95.73±0.62 (2)	**95.76±0.64 (2)**	94.38±0.87 (3)
Breast	96.03±0.34 (1)	**96.34±0.35 (2)**	95.94±0.44 (2)	96.28±0.29 (6)	96.05±0.62 (2)	96.09±0.20 (6)
Glass	62.85±1.73 (5)	63.55±1.43 (9)	66.19±2.21 (3)	**70.65±1.12 (6)**	67.66±2.32 (3)	68.74±1.37 (8)
Heart	76.43±1.19 (1)	76.60±1.78 (1)	76.73±1.41 (2)	76.73±1.41 (2)	**77.44±1.41 (4)**	**77.44±1.41 (4)**
Iris	96.93±0.72 (1)	97.13±0.77 (1)	**97.20±0.69 (1)**	96.87±0.63 (1)	97.13±0.77 (1)	96.60±0.21 (4)
Liver	61.01±3.28 (1)	**65.65±2.50 (4)**	65.04±2.20 (3)	65.42±2.24 (3)	65.13±1.39 (5)	65.13±1.39 (5)
Pima	69.13±1.32 (1)	69.78±0.51 (8)	69.79±0.74 (5)	70.01±0.56 (5)	**70.42±1.55 (4)**	**70.42±1.55 (4)**
Sonar	73.03±2.27 (1)	81.06±2.25 (54)	78.56±2.03 (5)	84.86±1.59 (18)	80.86±1.28 (5)	**84.90±1.51 (17)**
Vehicle	74.33±1.10 (3)	**81.55±0.51 (9)**	73.45±0.66 (5)	76.32±0.75 (15)	74.65±0.46 (2)	80.01±0.50 (3)
Average	77.83	80.07	79.48	81.41	80.67	**81.54**

Conclusion

- **PCA-L1**
  - finds projections that maximizes L1-norm in the projected space.
  - is proven to find a local maximum point.
  - is robust to outliers.
  - is simple and easy to implement.
  - is relatively fast with small number of iterations.
  - The number of iterations does not depend on the dimension of input space.
  - The same technique can be applied to other feature extraction algorithm.

Recent developments

- Non-greedy version extension
  - F. Nie, H. Huang, C. Ding, D. Luo, and H. Wang, "Robust principal component analysis with non-greedy L1-norm maximization", in Proc. 22nd International Conf. on Artificial Intelligence, 2011, pp. 1433 – 1438.
- 2D & tensor extension
  - X. Li, Y. Pang, and Y. Yuan, "L1-norm based 2DPCA", IEEE Trans. on SMC-B, vol. 38, no. 4, pp. 1170 – 1175, Aug. 2010.
  - Y. Pang, X. Li, and Y. Yuan, "Robust tensor analysis with L1-norm", IEEE Trans. Circuits Syst. Video Techn., vol. 20, no. 2, pp. 172.178, 2010.
- Mean or Median? - Generalized mean
  - Jiyong Oh, Nojun Kwak, Minsik Lee and Chong-Ho Choi, "Generalized mean for feature extraction in one-class classification problems", submitted to Pattern Recognition.
- Generalization to Lp-norm
  - Nojun Kwak, "Principal component analysis by Lp-norm maximization", submitted to IEEE Trans. on SMC-B.
- Applications to other subspace methods (e.g., LDA)
  - Jae Hyun Oh and Nojun Kwak, "Generalization of linear discriminant analysis using Lp-norm", submitted to Pattern Recognition Letters.

References

[1] I.T. Jolliffe,
Principal Component Analysis,
Springer-Verlag, 1986.

[2] A. Baccini, P. Besse, and A.D. Falguerolles,
"A L1-norm pca and a heuristic approach,"
in *Ordinal and Symbolic Data Analysis*, E. Diday, Y. Lechevalier, and P. Opitz, Eds. 1996, pp. 359–368, Springer.

[3] Q. Ke and T. Kanade,
"Robust subspace computation using l1 norm,"
Tech. Rep. CMU-CS-03-172, Carnegie Mellon University, Aug. 2003,
http://citeseer.ist.psu.edu/ ke03robust.html.

[4] Q. Ke and T. Kanade,
"Robust l1 norm factorization in the presence of outliers and missing data by alternative convex programming,"
in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, June 2005.

[5] C. Ding, D. Zhou, X. He, and H. Zha,
"R1-pca: rotational invariant l1-norm principal component analysis for fobust subspace factorization,"
in *Proc. International Conference on Machine Learning*, Pittsburgh, PA, June 2006.

[6] N. Kwak,
"Principal component analysis based on L1 norm maximization,"
IEEE TPAMI, vol. 30, no. 9, pp. 1672–1680, Sep. 2008.

[7] N. Kwak and J. Oh,
"Feature extraction for one-class classification problem: Enhancements to biased discriminant analysis,"
Pattern Recognition, vol. 42, no. 1, pp. 17–26, Jan. 2009.