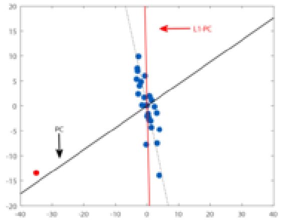
WIKIPEDIA

L1-norm principal component analysis

L1-norm principal component analysis (L1-PCA) is a general method for multivariate data analysis.[1] L1-PCA is often preferred over standard L2-norm principal component analysis (PCA) when the analyzed data may contain outliers (faulty values or corruptions).[2][3][4]



L1-PCA compared with PCA. Nominal data (blue points); outlier (red point); PC (black line); L1-PC (red line); nominal maximum-variance line (dotted line).

Both L1-PCA and standard PCA seek a collection of orthogonal directions (principal components) that define a subspace wherein data representation is maximized according to the selected criterion.[5][6][7] Standard PCA quantifies data representation as the aggregate of the L2-norm of the data point projections into the subspace, or equivalently the aggregate Euclidean distance of the original points from their subspace-projected representations. L1-PCA uses instead the aggregate of the L1-norm of the data point projections into the subspace.[8] In PCA and L1-PCA, the number of principal components (PCs) is lower than the rank of the analyzed matrix, which coincides with the dimensionality of the space defined by the original data points. Therefore, PCA or L1-PCA are commonly employed for dimensionality reduction for the purpose of data denoising or compression. Among the advantages of standard PCA that contributed to its high popularity are low-cost computational implementation by means of singular-value decomposition (SVD)[9] and statistical optimality when the data set is generated by a true multivariate Normal data source.

However, in modern big data sets, data often include corrupted, faulty points, commonly referred to as outliers.[10] Standard PCA is known to be sensitive to outliers, even when they appear as a small fraction of the processed data.[11] The reason is that the L2-norm formulation of L2-PCA places squared emphasis on the magnitude of each coordinate of each data point, ultimately overemphasizing peripheral points, such as outliers. On the other hand, following an L1-norm formulation, L1-PCA places linear emphasis on the coordinates of each data point, effectively restraining outliers.[12]

Contents

Formulation

Consider any matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$ consisting of N D-dimensional data points. Define $r = rank(\mathbf{X})$. For integer K such that $1 \leq K < r$, L1-PCA is formulated as:[1]

$$\begin{aligned} &\max_{\mathbf{Q}=[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_K] \in \mathbb{R}^{D \times K}} \quad \|\mathbf{X}^\top \mathbf{Q}\|_1 \\ &\text{subject to} \quad \mathbf{Q}^\top \mathbf{Q} = \mathbf{I}_K. \end{aligned} \qquad (1)$$

For $K = 1$, (1) simplifies to finding the L1-norm principal component (L1-PC) of $\mathbf{X}$ by

$$\begin{aligned} &\max_{\mathbf{q} \in \mathbb{R}^{D \times 1}} \quad \|\mathbf{X}^\top \mathbf{q}\|_1 \\ &\text{subject to} \quad \|\mathbf{q}\|_2 = 1. \end{aligned} \qquad (2)$$

In (1)-(2), L1-norm $\|\cdot\|_1$ returns the sum of the absolute entries of its argument and L2-norm $\|\cdot\|_2$ returns the sum of the squared entries of its argument. If one substitutes $\|\cdot\|_1$ in (2) by the Frobenius/L2-norm $\|\cdot\|_F$, then the problem becomes standard PCA and it is solved by the matrix $\mathbf{Q}$ that contains the K dominant singular vectors of $\mathbf{X}$ (i.e., the singular vectors that correspond to the K highest singular values).

The maximization metric in (2) can be expanded as

$$\|\mathbf{X}^\top \mathbf{Q}\|_1 = \sum_{k=1}^{K}\sum_{n=1}^{N} |\mathbf{x}_n^\top \mathbf{q}_k|. \quad (3)$$

Solution

For any matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ with $m \geq n$, define $\Phi(\mathbf{A})$ as the nearest (in the L2-norm sense) matrix to $\mathbf{A}$ that has orthonormal columns. That is, define

$$\begin{aligned}\Phi(\mathbf{A}) =& \underset{\mathbf{Q}\in\mathbb{R}^{m\times n}}{\operatorname{argmin}} \ \|\mathbf{A}-\mathbf{Q}\|_F \\ & \text{subject to } \ \mathbf{Q}^\top\mathbf{Q} = \mathbf{I}_n.\end{aligned} \quad (4)$$

Procrustes Theorem[13][14] states that if $\mathbf{A}$ has SVD $\mathbf{U}_{m\times n}\mathbf{\Sigma}_{n\times n}\mathbf{V}_{n\times n}^\top$, then $\Phi(\mathbf{A}) = \mathbf{U}\mathbf{V}^\top$.

Markopoulos, Karystinos, and Pados[1] showed that, if $\mathbf{B}_{\text{BNM}}$ is the exact solution to the binary nuclear-norm maximization (BNM) problem

$$\max_{\mathbf{B}\in\{\pm1\}^{N\times K}} \ \|\mathbf{X}\mathbf{B}\|_*^2, \quad (5)$$

then

$$\mathbf{Q}_{\text{L1}} = \Phi(\mathbf{X}\mathbf{B}_{\text{BNM}}) \quad (6)$$

is the exact solution to L1-PCA in (2). The nuclear-norm $\|\cdot\|_*$ in (2) returns the summation of the singular values of its matrix argument and can be calculated by means of standard SVD. Moreover, it holds that, given the solution to L1-PCA, $\mathbf{Q}_{\text{L1}}$, the solution to BNM can be obtained as

$$\mathbf{B}_{\text{BNM}} = \text{sgn}(\mathbf{X}^\top\mathbf{Q}_{\text{L1}}) \quad (7)$$

where $\text{sgn}(\cdot)$ returns the $\{\pm1\}$-sign matrix of its matrix argument (with no loss of generality, we can consider $\text{sgn}(0)=1$). In addition, it follows that $\|\mathbf{X}^\top\mathbf{Q}_{\text{L1}}\|_1 = \|\mathbf{X}\mathbf{B}_{\text{BNM}}\|_*$. BNM in (5) is a combinatorial problem over antipodal binary variables. Therefore, its exact solution can be found through exhaustive evaluation of all 2^{NK} elements of its feasibility set, with asymptotic cost $\mathcal{O}(2^{NK})$. Therefore, L1-PCA can also be solved, through BNM, with cost $\mathcal{O}(2^{NK})$ (exponential in the product of the number of data points with the number of the sought-after components). It turns out that L1-PCA can be solved optimally (exactly) with polynomial complexity in N for fixed data dimension D, $\mathcal{O}(N^{rK-K+1})$.[1]

For the special case of $K=1$ (single L1-PC of $\mathbf{X}$), BNM takes the binary-quadratic-maximization (BQM) form

$$\max_{\mathbf{b}\in\{\pm1\}^{N\times 1}} \ \mathbf{b}^\top\mathbf{X}^\top\mathbf{X}\mathbf{b}. \quad (8)$$

The transition from (5) to (8) for $K=1$ holds true, since the unique singular value of $\mathbf{X}\mathbf{b}$ is equal to $\|\mathbf{X}\mathbf{b}\|_2 = \sqrt{\mathbf{b}^\top\mathbf{X}^\top\mathbf{X}\mathbf{b}}$, for every $\mathbf{b}$. Then, if $\mathbf{b}_{\text{BNM}}$ is the solution to BQM in (7), it holds that

$$\mathbf{q}_{\text{L1}} = \Phi(\mathbf{X}\mathbf{b}_{\text{BNM}}) = \frac{\mathbf{X}\mathbf{b}_{\text{BNM}}}{\|\mathbf{X}\mathbf{b}_{\text{BNM}}\|_2} \quad (9)$$

is the exact L1-PC of $\mathbf{X}$, as defined in (1). In addition, it holds that $\mathbf{b}_{\text{BNM}} = \text{sgn}(\mathbf{X}^\top\mathbf{q}_{\text{L1}})$ and $\|\mathbf{X}^\top\mathbf{q}_{\text{L1}}\|_1 = \|\mathbf{X}\mathbf{b}_{\text{BNM}}\|_2$.

Algorithms

Exact solution of exponential complexity

As shown above, the exact solution to L1-PCA can be obtained by the following two-step process:

```
1. Solve the problem in ($5$) to obtain $\mathbf{B}_{\text{BNM}}$.
2. Apply SVD on $\mathbf{X}\mathbf{B}_{\text{BNM}}$ to obtain $\mathbf{Q}_{\text{L1}}$.
```

BNM in (5) can be solved by exhaustive search over the domain of $\mathbf{B}$ with cost $\mathcal{O}(2^{NK})$.

Exact solution of polynomial complexity

Also, L1-PCA can be solved optimally with cost $\mathcal{O}(N^{rK-K+1})$, when $r = rank(\mathbf{X})$ is constant with respect to N (always true for finite data dimension D).[1][15]

Approximate efficient solvers

In 2008, Kwak[12] proposed an iterative algorithm for the approximate solution of L1-PCA for $K = 1$. This iterative method was later generalized for $K > 1$ components.[16] Another approximate efficient solver was proposed by McCoy and Tropp[17] by means of semi-definite programming (SDP). Most recently, L1-PCA (and BNM in (5)) were solved efficiently by means of bit-flipping iterations (L1-BF algorithm).[8][18]

L1-BF algorithm

```
function L1BF($\mathbf{X}$, $K$):
    Initialize $\mathbf{B}^{(0)} \in \{\pm 1\}^{N \times K}$ and $\mathcal{L} \leftarrow \{1, 2, \ldots, NK\}$
    Set $t \leftarrow 0$ and $\omega \leftarrow \|\mathbf{X}\mathbf{B}^{(0)}\|_*$
    Until termination (or $T$ iterations)
        $\mathbf{B} \leftarrow \mathbf{B}^{(t)}$, $t' \leftarrow t$
        For $x \in \mathcal{L}$
            $k \leftarrow \lceil \frac{x}{N} \rceil$, $n \leftarrow x - N(k-1)$
            $[\mathbf{B}]_{n,k} \leftarrow -[\mathbf{B}]_{n,k}$                // flip bit
            $a(n,k) \leftarrow \|\mathbf{X}\mathbf{B}\|_*$                // calculated by SVD or faster (see[8])
            if $a(n,k) > \omega$
                $\mathbf{B}^{(t)} \leftarrow \mathbf{B}$, $t' \leftarrow t + 1$
                $\omega \leftarrow a(n,k)$
            end
            if $t' = t$                              // no bit was flipped
                if $\mathcal{L} = \{1, 2, \ldots, NK\}$
                    terminate
                else
                    $\mathcal{L} \leftarrow \{1, 2, \ldots, NK\}$
```

The computational cost of L1-BF is $\mathcal{O}(NDmin\{N, D\} + N^2K^2(K^2 + r))$.[8]

Complex data

L1-PCA has also been generalized to process complex data. For complex L1-PCA, two efficient algorithms were proposed in 2018.[19]

Tensor data

L1-PCA has also been extended for the analysis of tensor data, in the form of L1-Tucker, the L1-norm robust analogous of standard Tucker decomposition.[20] Two algorithms for the solution of L1-Tucker are L1-HOSVD and L1-HOOI.[20][21][22]

Code

MATLAB code for L1-PCA is available at MathWorks[23] and other repositories.[18]

References

1. Markopoulos, Panos P.; Karystinos, George N.; Pados, Dimitris A. (October 2014). "Optimal Algorithms for L1-subspace Signal Processing". *IEEE Transactions on Signal Processing*. **62** (19): 5046–5058. arXiv:1405.6785 (https://arxiv.org/abs/1405.6785). Bibcode:2014ITSP...62.5046M (https://ui.adsabs.harvard.edu/abs/2014ITSP...62.5046M). doi:10.1109/TSP.2014.2338077 (https://doi.org/10.1109%2FTSP.2014.2338077).
2. Barrodale, I. (1968). "L1 Approximation and the Analysis of Data". *Applied Statistics*. **17** (1): 51–57. doi:10.2307/2985267 (https://doi.org/10.2307%2F2985267). JSTOR 2985267 (https://www.jstor.org/stable/2985267).
3. Barnett, Vic; Lewis, Toby (1994). *Outliers in statistical data* (3. ed.). Chichester [u.a.]: Wiley. ISBN 978-0471930945.
4. Kanade, T.; Ke, Qifa (June 2005). *Robust L1 Norm Factorization in the Presence of Outliers and Missing Data by Alternative Convex Programming*. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. **1**. IEEE. p. 739. CiteSeerX 10.1.1.63.4605 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.63.4605). doi:10.1109/CVPR.2005.309 (https://doi.org/10.1109%2FCVPR.2005.309). ISBN 978-0-7695-2372-9.

5. Jolliffe, I.T. (2004). *Principal component analysis* (https://archive.org/details/principalcompone00joll_0) (2nd ed.). New York: Springer. ISBN 978-0387954424.
6. Bishop, Christopher M. (2007). *Pattern recognition and machine learning* (Corr. printing. ed.). New York: Springer. ISBN 978-0-387-31073-2.
7. Pearson, Karl (8 June 2010). "On Lines and Planes of Closest Fit to Systems of Points in Space" (https://zenodo.org/record/1430636/files/article.pdf) (PDF). *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*. **2** (11): 559–572. doi:10.1080/14786440109462720 (https://doi.org/10.1080%2F14786440109462720).
8. Markopoulos, Panos P.; Kundu, Sandipan; Chamadia, Shubham; Pados, Dimitris A. (15 August 2017). "Efficient L1-Norm Principal-Component Analysis via Bit Flipping". *IEEE Transactions on Signal Processing*. **65** (16): 4252–4264. arXiv:1610.01959 (https://arxiv.org/abs/1610.01959). Bibcode:2017ITSP...65.4252M (https://ui.adsabs.harvard.edu/abs/2017ITSP...65.4252M). doi:10.1109/TSP.2017.2708023 (https://doi.org/10.1109%2FTSP.2017.2708023).
9. Golub, Gene H. (April 1973). "Some Modified Matrix Eigenvalue Problems". *SIAM Review*. **15** (2): 318–334. CiteSeerX 10.1.1.454.9868 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.454.9868). doi:10.1137/1015032 (https://doi.org/10.1137%2F1015032).
10. Barnett, Vic; Lewis, Toby (1994). *Outliers in statistical data* (3. ed.). Chichester [u.a.]: Wiley. ISBN 978-0471930945.
11. Candès, Emmanuel J.; Li, Xiaodong; Ma, Yi; Wright, John (1 May 2011). "Robust principal component analysis?". *Journal of the ACM*. **58** (3): 1–37. arXiv:0912.3599 (https://arxiv.org/abs/0912.3599). doi:10.1145/1970392.1970395 (https://doi.org/10.1145%2F1970392.1970395).
12. Kwak, N. (September 2008). "Principal Component Analysis Based on L1-Norm Maximization". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **30** (9): 1672–1680. CiteSeerX 10.1.1.333.1176 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.333.1176). doi:10.1109/TPAMI.2008.114 (https://doi.org/10.1109%2FTPAMI.2008.114). PMID 18617723 (https://pubmed.ncbi.nlm.nih.gov/18617723).
13. Eldén, Lars; Park, Haesun (1 June 1999). "A Procrustes problem on the Stiefel manifold". *Numerische Mathematik*. **82** (4): 599–619. CiteSeerX 10.1.1.54.3580 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.54.3580). doi:10.1007/s002110050432 (https://doi.org/10.1007%2Fs002110050432).
14. Schönemann, Peter H. (March 1966). "A generalized solution of the orthogonal procrustes problem". *Psychometrika*. **31** (1): 1–10. doi:10.1007/BF02289451 (https://doi.org/10.1007%2FBF02289451). hdl:10338.dmlcz/103138 (https://hdl.handle.net/10338.dmlcz%2F103138).
15. Markopoulos, PP; Kundu, S; Chamadia, S; Tsagkarakis, N; Pados, DA (2018). *Outlier-Resistant Data Processing with L1-Norm Principal Component Analysis*. *Advances in Principal Component Analysis*. p. 121. doi:10.1007/978-981-10-6704-4_6 (https://doi.org/10.1007%2F978-981-10-6704-4_6). ISBN 978-981-10-6703-7.
16. Nie, F; Huang, H; Ding, C; Luo, Dijun; Wang, H (July 2011). "Robust principal component analysis with non-greedy l1-norm maximization". *22nd International Joint Conference on Artificial Intelligence*: 1433–1438.
17. McCoy, Michael; Tropp, Joel A. (2011). "Two proposals for robust PCA using semidefinite programming". *Electronic Journal of Statistics*. **5**: 1123–1160. arXiv:1012.1086 (https://arxiv.org/abs/1012.1086). doi:10.1214/11-EJS636 (https://doi.org/10.1214%2F11-EJS636).
18. Markopoulos, PP. "Software Repository" (https://people.rit.edu/pxmeee/soft.html). Retrieved May 21, 2018.
19. Tsagkarakis, Nicholas; Markopoulos, Panos P.; Sklivanitis, George; Pados, Dimitris A. (15 June 2018). "L1-Norm Principal-Component Analysis of Complex Data". *IEEE Transactions on Signal Processing*. **66** (12): 3256–3267. arXiv:1708.01249 (https://arxiv.org/abs/1708.01249). Bibcode:2018ITSP...66.3256T (https://ui.adsabs.harvard.edu/abs/2018ITSP...66.3256T). doi:10.1109/TSP.2018.2821641 (https://doi.org/10.1109%2FTSP.2018.2821641).
20. Chachlakis, Dimitris G.; Prater-Bennette, Ashley; Markopoulos, Panos P. (22 November 2019). "L1-norm Tucker Tensor Decomposition". *IEEE Access*. **7**: 178454–178465. doi:10.1109/ACCESS.2019.2955134 (https://doi.org/10.1109%2FACCESS.2019.2955134).
21. Markopoulos, Panos P.; Chachlakis, Dimitris G.; Prater-Bennette, Ashley (21 February 2019). "L1-norm Higher-Order Singular-Value Decomposition". *IEEE Proc. 2018 IEEE Global Conference on Signal and Information Processing*. doi:10.1109/GlobalSIP.2018.8646385 (https://doi.org/10.1109%2FGlobalSIP.2018.8646385).
22. Markopoulos, Panos P.; Chachlakis, Dimitris G.; Papalexakis, Evangelos (April 2018). "The Exact Solution to Rank-1 L1-Norm TUCKER2 Decomposition". *IEEE Signal Processing Letters*. **25** (4). arXiv:1710.11306 (https://arxiv.org/abs/1710.11306). doi:10.1109/LSP.2018.2790901 (https://doi.org/10.1109%2FLSP.2018.2790901).
23. "L1-PCA TOOLBOX" (https://www.mathworks.com/matlabcentral/fileexchange/64855-l1-pca-toolbox). Retrieved May 21, 2018.

Retrieved from "https://en.wikipedia.org/w/index.php?title=L1-norm_principal_component_analysis&oldid=950455542"

This page was last edited on 12 April 2020, at 05:39 (UTC).