Dimensionality Reduction Algorithms for Medical Imaging: PCA, NMF, ICA, etc.

Ben Kandel

June 29, 2012

Introduction

Dimensionality reduction algorithms can be written as matrix factorization problems. The basic form of the factorization is something like

$$\mathbf{X} \approx \mathbf{W}\mathbf{V}^{\mathrm{T}} \tag{1}$$

Books and articles have different notations for the dimensionality of $\mathbf{X}$ and the different matrices (so sometimes you will see that $\mathbf{X}$ is the data matrix, and sometimes $\mathbf{X}^{\mathrm{T}}$), and the transposes are propagated along all the steps of the decomposition. We'll try to be consistent and use the following notation:

- The data matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ has n rows, with each row corresponding to an observation (one subject), and p columns, with each column correponding to a variable (voxel).
- The loading matrix $\mathbf{W} \in \mathbb{R}^{n \times r}$ corresponds to the reduced-dimensionality version of $\mathbf{X}$, replacing the p variables with $r \ll p$ variables.
- The basis matrix $\mathbf{V} \in \mathbb{R}^{p \times r}$ corresponds to the eigenvector matrix from classical PCA.

Using this notation, the loading matrix $\mathbf{W}$ corresponds to the projections of the data matrix on the basis matrix, $\mathbf{W} = \mathbf{X}\mathbf{V}$.

PCA

The most widely used dimensionality reduction algorithm is PCA. PCA finds an orthogonal rotation of the covariance matrix $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ that satisfies one of the following problems:

1. Maximize variance in projected space:

$$\begin{aligned} \underset{\mathbf{V}}{\text{maximize}} \quad & \|\mathbf{X}\mathbf{V}\|_2^2 \\ \text{subject to} \quad & \mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{I} \end{aligned} \tag{2}$$

2. Minimizing reconstruction error:

$$\begin{aligned} \underset{\mathbf{V}}{\text{minimize}} \quad & \|\mathbf{X} - \mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{X}\|_2^2 \\ \text{subject to} \quad & \mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{I} \end{aligned} \tag{3}$$

In either case, orthogonality in the projected space is enforced.

Sparse PCA

In sparse PCA, additional constraints are enforced on the eigenvectors of the covariance matrix. This gives us something like

$$\underset{\mathbf{V}}{\text{minimize}} \|\mathbf{X} - \mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{X}\|_2^2 + \lambda \sum_i |V_i|, \tag{4}$$

where each column in $\mathbf{V}$ is V_i. Enforcing sparsity, though, normally entails discarding orthogonality. Because the "eigenvectors" are not orthogonal, one component of the data matrix may project onto more than one "eigenvector," so total variance explained is not an appropriate measure of how good an approximation to the original matrix the sparse eigenvectors are. Most versions of SPCA still do have some sort of constraints, though:

1. Changing notation to standard SVD notation ($\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}}$), [6] enforces $\|\mathbf{u}\|_2^2 \leq 1$, $\|\mathbf{v}\|_2^2 \leq 1$ (but not orthogonality). This is sort of a lower bound on the kinds of restraints you have–each vector must at the very least not increase the size of the original matrix.

2. Jolliffe [3] *does* enforce orthogonality on the eigenvectors. The tradeoff this presents is that the output is not uncorrelated: $\mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{I}$, but $\mathbf{V}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{V}$ is not diagonal.

3. Zou [7] splits the eigenvector matrices ($\mathbf{V}\mathbf{V}^{\mathrm{T}}$) in two, $\mathbf{A}\mathbf{B}^{\mathrm{T}}$. He then enforces orthogonality on $\mathbf{A}$ and sparsity on $\mathbf{B}$:

$$\underset{\mathbf{A},\mathbf{B}}{\text{minimize}} \|\mathbf{X} - \mathbf{A}\mathbf{B}^{\mathrm{T}}\mathbf{X}\|_2^2 + p\left(\mathbf{B}\right), \tag{5}$$

   where $p\left(\cdot\right)$ is the sparsity penalty and $\mathbf{A}^{\mathrm{T}}\mathbf{A} = \mathbf{I}$.

4. Shen [5] only really gives results for a rank-one approximation of the data matrix. He basically iterates between projecting on $\mathbf{u}$ and $\mathbf{v}$, applying a thresholding operation, and re-scaling so that the vectors have unit norm. Except for the re-scaling (so that $\|\mathbf{v}\|_2^2 = 1$), there is no other constraint.

NMF

NMF has two objective functions (see "Algorithms for Non-Negative Matrix Factorization" in NIPS 2001):

1.
$$\begin{aligned} \underset{\mathbf{V},\mathbf{W}}{\text{minimize}} \quad & \|\mathbf{X} - \mathbf{V}\mathbf{W}^{\mathrm{T}}\|_2^2 \\ \text{subject to} \quad & \mathbf{V}, \mathbf{W} \succeq 0 \end{aligned} \tag{6}$$

2.
$$\begin{aligned} \underset{\mathbf{V},\mathbf{W}}{\text{minimize}} \quad & D\left(\mathbf{X} \| \mathbf{V}\mathbf{W}^{\mathrm{T}}\right) \\ \text{subject to} \quad & \mathbf{V}, \mathbf{W} \succeq 0, \end{aligned} \tag{7}$$

where $D(X\|Y)$ is the divergence (KL, if they're probabilities) of the two arguments.

There are no orthogonality, diagonalization, etc. constraints at all.

ICA

ICA uses a fundamentally different objective function from PCA. As explained before, PCA maximizes the variance of the projections onto the eigenvectors. This objective has a simple closed-form solution, but suffers from the drawback of assuming that the variance of the projections is a sufficient statistic to characterize the data. If the data follows a Gaussian distribution, the variance is a sufficient statistic. ICA, on the other hand, does not assume Gausssianity of the data. Because of this, it cannot use variance as a statistic. Instead, it first whitens the data so that it has unit variance, and then computes "eigenvectors" that give independent projections.

Instead of using the variance of the projections, ICA (at least as implemented in the FastICA algorithm, which is the most widely used algorithm for ICA [1, 2]) uses the negentropy, which is a measure of how far apart a given random variable is from Gaussianity:

$$J_G(\mathbf{v}) = \left[E\left\{G\left(\mathbf{v}^{\mathrm{T}}\mathbf{x}\right)\right\} - E\left\{G(\nu)\right\}\right]^2, \tag{8}$$

where $\mathbf{v}$ is a column of $\mathbf{V}$ and $\mathbf{x}$ is a row of $\mathbf{X}$, ν is a Gaussian rv with unit variance (same as $\mathbf{v}$) and G is some non-quadratic "contrast" function, such as $\log\cosh(x)$. One other major difference between PCA and ICA is that the orthogonality constraint is replaced by a uncorrelatedness constraint.[1] With these two modifications, we can write the objective function of ICA as follows:

$$\begin{aligned} \underset{\mathbf{v}}{\text{maximize}} \quad & \sum_{i=1}^{n} J_G(\mathbf{v_i}) \\ \text{subject to} \quad & E\left\{\left(\mathbf{v}_i^{\mathrm{T}}\mathbf{x}\right)\left(\mathbf{v}_j^{\mathrm{T}}\mathbf{x}\right)\right\} = \delta_{ij} \end{aligned} \tag{9}$$

The FastICA optimization procedure involves Gram–Schmidt "decorrelation" (which here seems to me the same as orthogonalization, unless I'm missing

[1] Just to be clear: The difference between orthogonal and uncorrelated is that uncorrelated utilizes to the dot product of *centered* vectors, and orthogonal uses the dot product of *uncentered* vectors. See [4].

something), the same as in power method iteration versions of PCA. So I'm not sure how different these really are in practice.

References

[1] A. Hyvarinen. Fast and robust fixed-point algorithms for independent component analysis. *Neural Networks, IEEE Transactions on*, 10(3):626 –634, May 1999.

[2] A. Hyvrinen and E. Oja. Independent component analysis: Algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.

[3] I.T. Jolliffe and M. Uddin. The simplified component technique: An alternative to rotated principal components. *Journal of Computational and Graphical Statistics*, 9(4):689–710, 2000.

[4] J. L. Rodgers, W. A. Nicewander, and L. Toothaker. Linearly independent, orthogonal, and uncorrelated variables. *The American Statistician*, 38(2):133–134, 1984.

[5] H. Shen and J.Z. Huang. Sparse principal component analysis via regularized low rank matrix approximation. *Journal of Multivariate Analysis*, 99(6):1015–1034, 2008.

[6] Daniela M Witten, Robert Tibshirani, and Trevor Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics (Oxford, England)*, 10(3):515–534, July 2009. PMID: 19377034.

[7] H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265—286, 2006.