Large Scale Matrix Factorization

Fei Wang

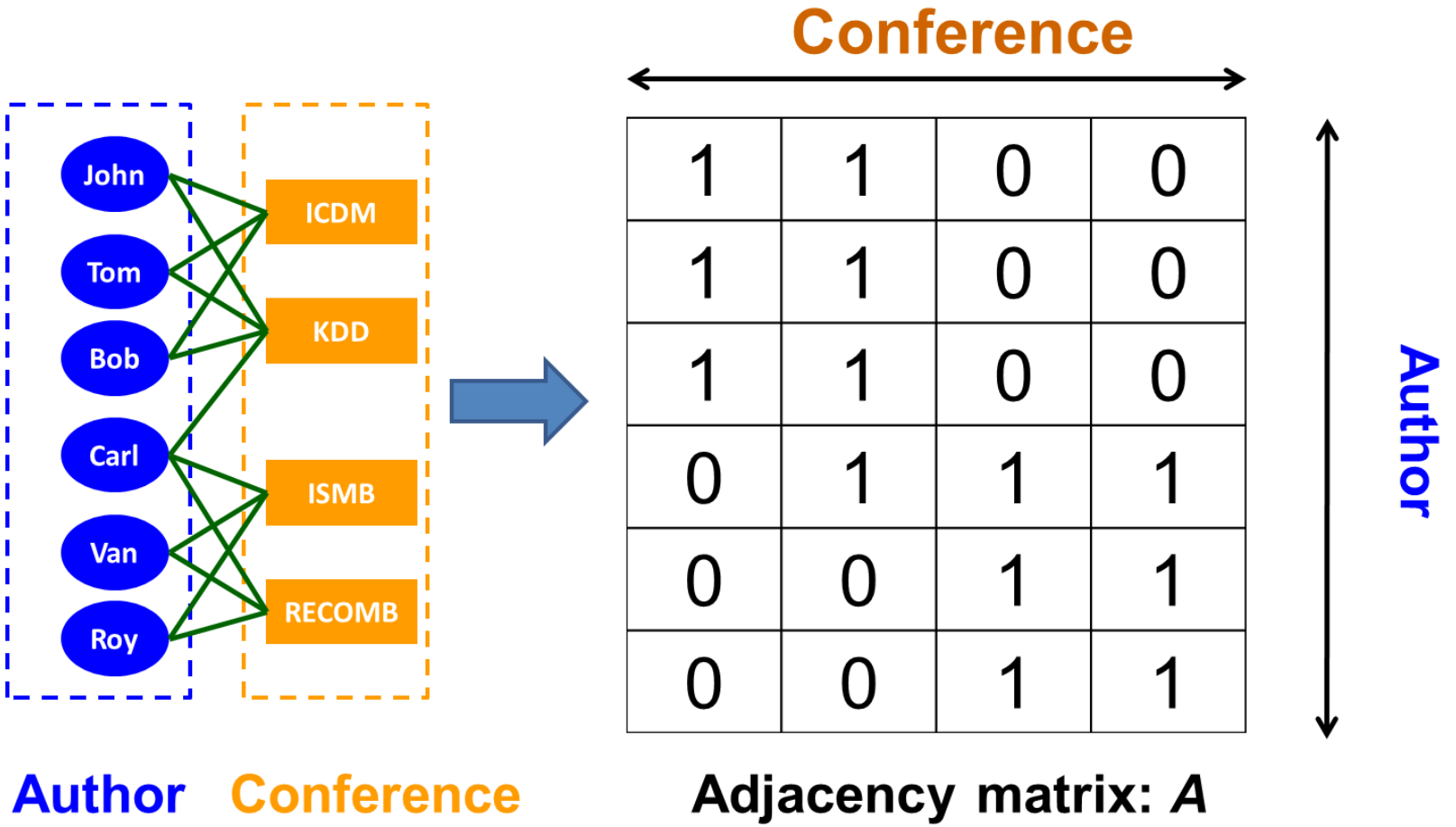
Division of Health Informatics
Department of Healthcare Policy and Research
Weill Cornell Medical College
Cornell University

Outline

- Introduction
- Matrix Factorization Technologies
- Conclusions and Discussions

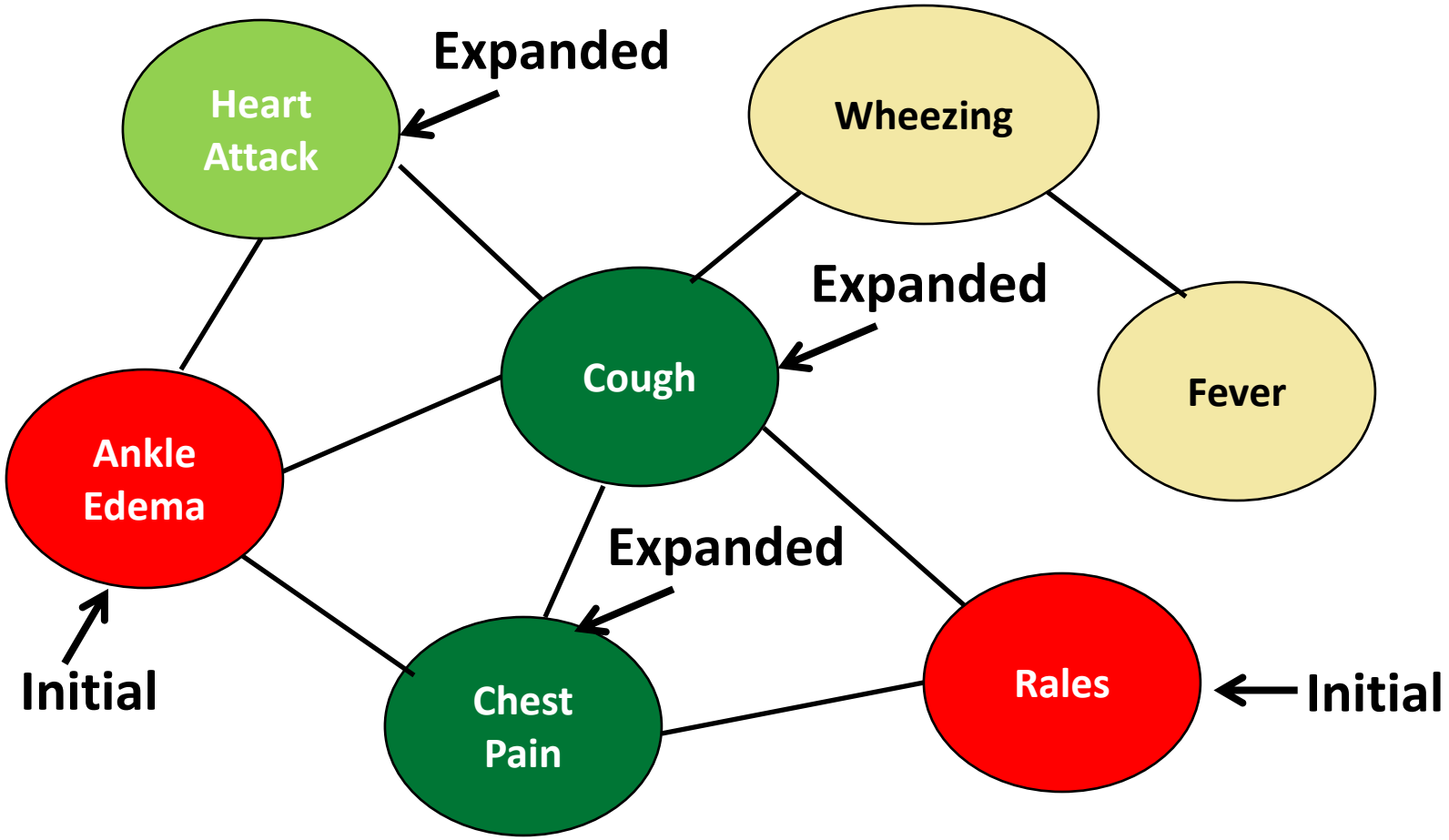
What is a matrix?

Matrix: A Natural Representation for Networks/Graphs/Relational Data

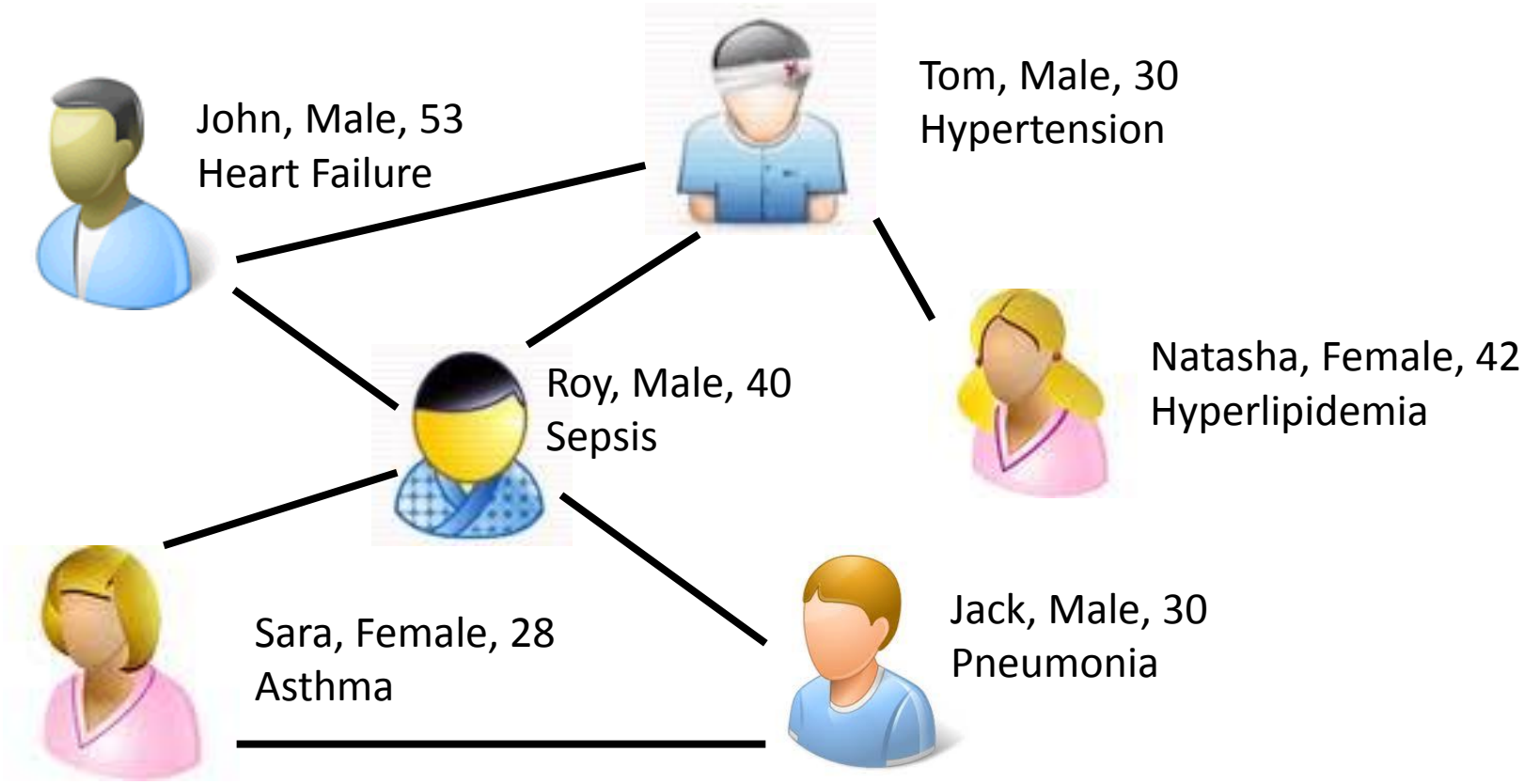


Matrices in Social Networks

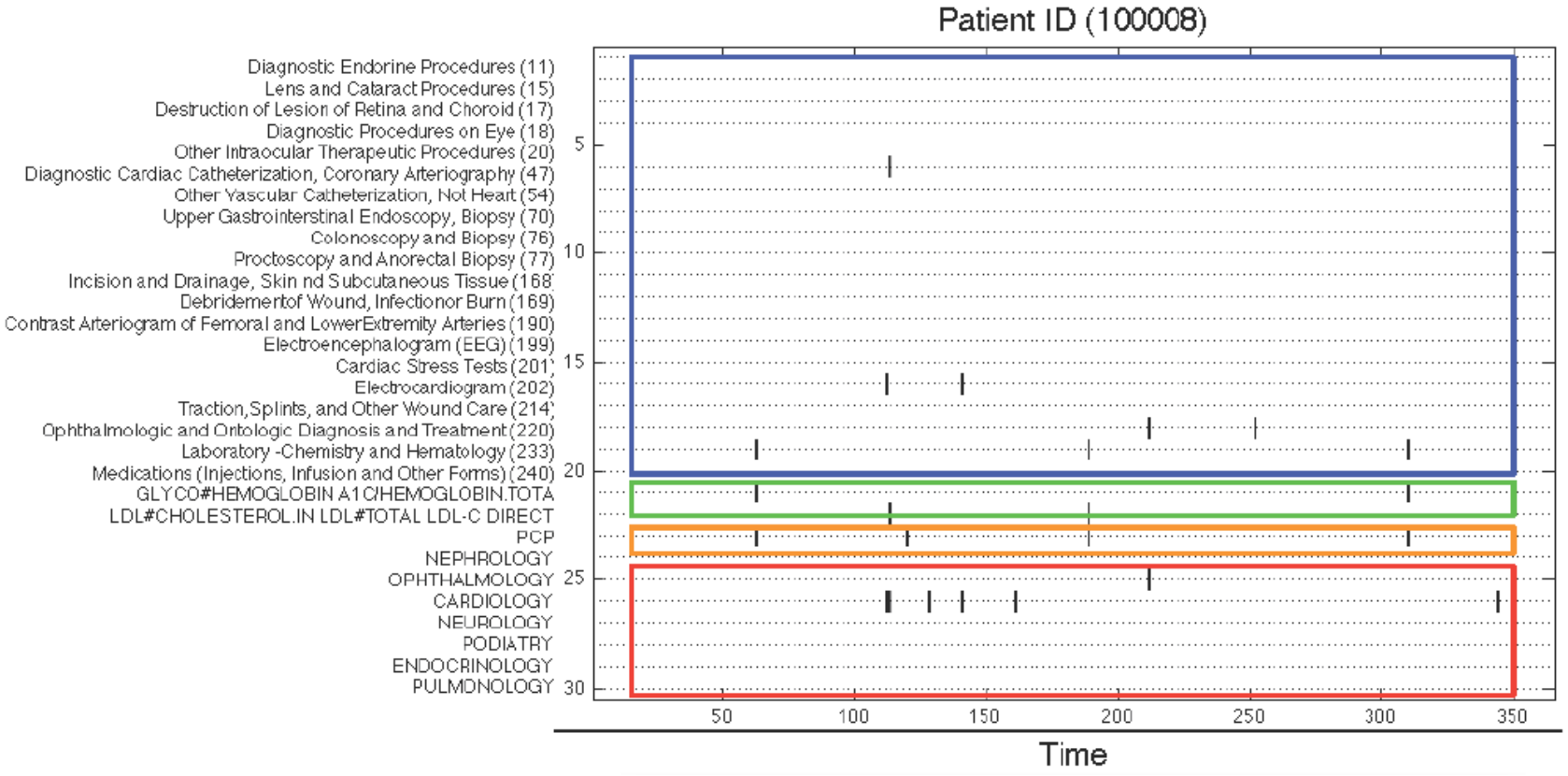
Matrices in Healthcare



Matrices in Healthcare



Matrices in Healthcare



Outline

- Introduction
- Matrix Factorization Technologies
  - Principal Component Analysis
  - Singular Value Decomposition
  - Nonnegative Matrix Factorization
  - Convolutional Matrix Factorization
  - Regularized Matrix Factorization
  - Inductive Matrix Factorization
- Conclusions and Discussions

Matrix Factorization

different types of product

different loss function

different regularizations

$$\mathcal{L}\left(\mathbf{X}, \prod_{i=1}^{K} \mathbf{A}_i\right) + \Omega\left(\bigcup_{i=1}^{K} \mathbf{A}_i\right)$$

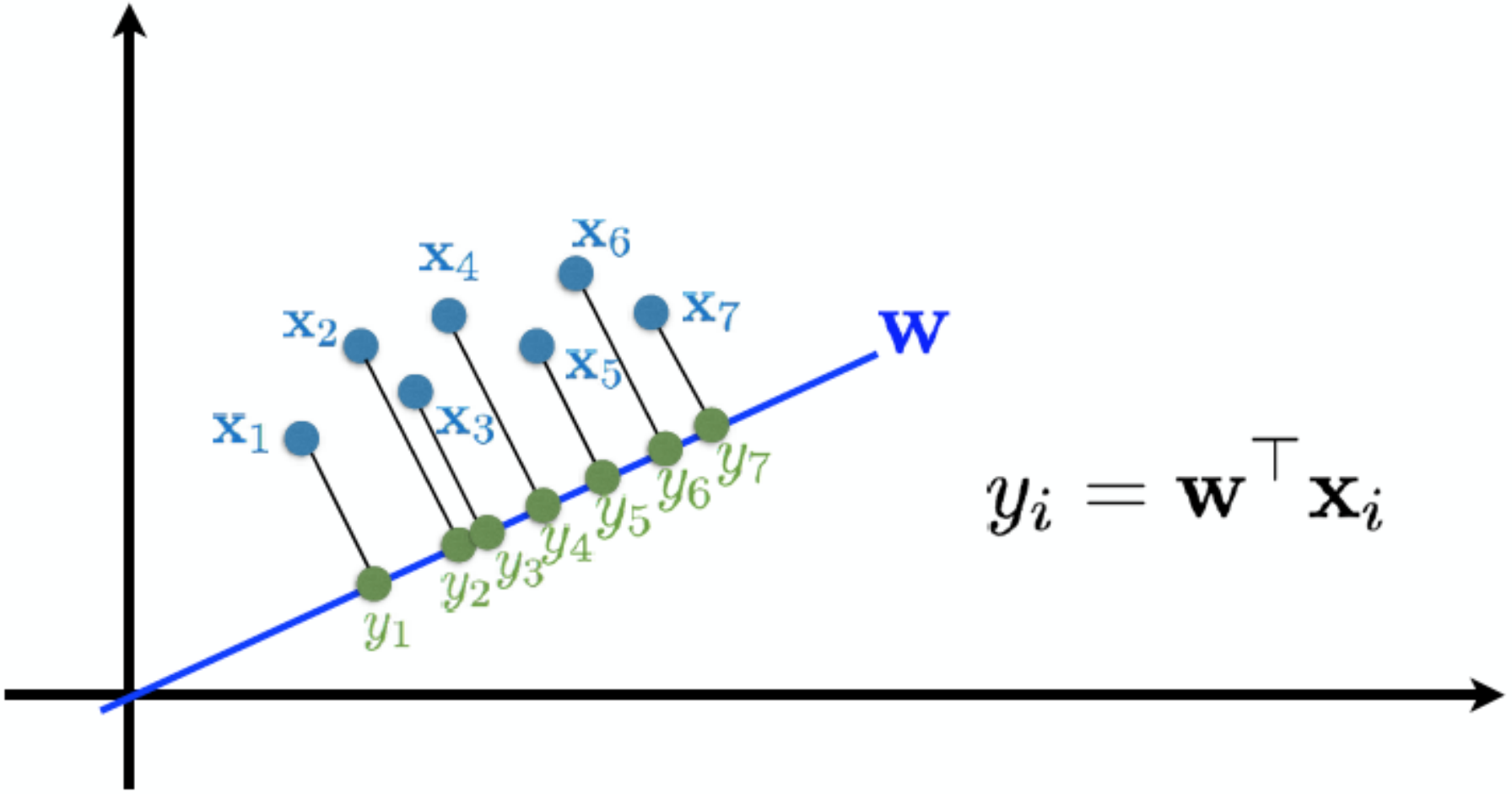
$$s.t. \quad \mathcal{C}\left(\bigcup_{i=1}^{K} \mathbf{A}_i\right)$$

different constraints

Outline

- Introduction
- Matrix Factorization Technologies
  - Principal Component Analysis
  - Singular Value Decomposition
  - Nonnegative Matrix Factorization
  - Convolutional Matrix Factorization
  - Regularized Matrix Factorization
  - Inductive Matrix Factorization
- Conclusions and Discussions

Orthogonal Projection

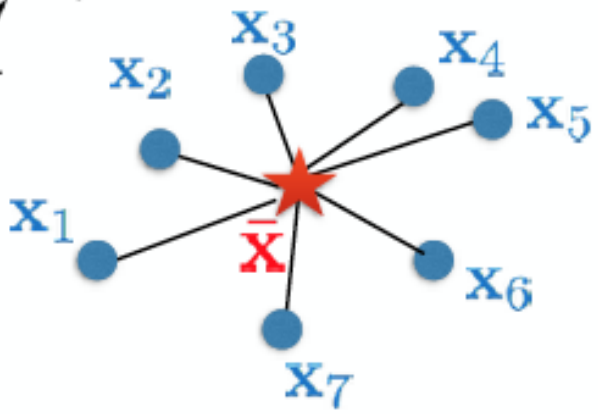


$$y_i = \mathbf{w}^\top \mathbf{x}_i$$

Principal Component Analysis

Find a direction where the data have the largest variance

$$\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$$



$$\frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})^\top(\mathbf{x}_i - \bar{\mathbf{x}})$$

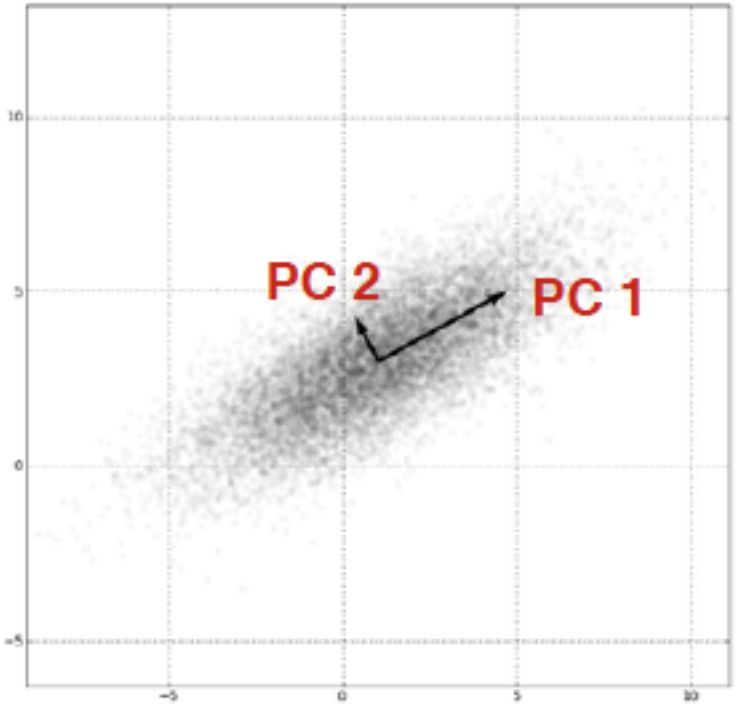
$$\frac{1}{n}\sum_{i=1}^{n}(\mathbf{w}^\top\mathbf{x}_i - \mathbf{w}^\top\bar{\mathbf{x}})^2$$

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i = \frac{1}{n}\sum_{i=1}^{n}\mathbf{w}^\top\mathbf{x}_i = \mathbf{w}^\top\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\right) = \mathbf{w}^\top\bar{\mathbf{x}}$$

Principal Component Analysis

1. Compute data covariance matrix
2. Do eigenvalue decomposition on the covariance matrix
3. Sort the eigenvalues from large to small

$$\mathbf{C} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top$$



Principal Component Analysis

- Fact: Any symmetric matrix M can be represented as $M = Q\Lambda Q^T$, where Q is orthonormal and Λ is diagonal.
- Columns of Q are the eigenvectors of X and the diagonal entries of Λ are associated eigenvalues

- Since $C = XX^T$ is symmetric, we can express $\frac{1}{n}YY^T$ as

$$\begin{aligned}\frac{1}{n}YY^T &= \frac{1}{n}(WX)(WX)^T \\ &= \frac{1}{n}WXX^TW^T \\ &= WCW^T \\ &= WQ\Lambda QW^T\end{aligned}$$

- Setting $W = Q^T$ gives $\frac{1}{n}YY^T = Q^TQ\Lambda Q^TQ = I\Lambda I = \Lambda$

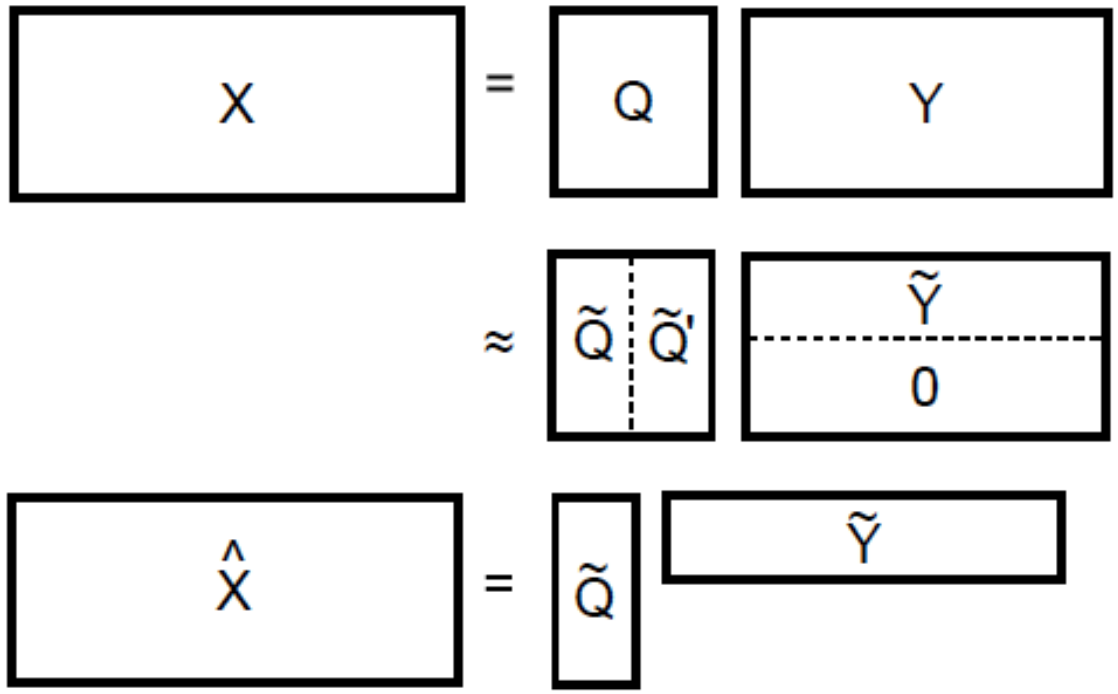
Principal Component Analysis

- So far, the PCA solution is $Y = Q^T X$, where $Q = [q_1, q_2 \cdots q_n]$ is a matrix of principal components, and Λ is a diagonal matrix of corresponding eigenvalues ordered from largest to smallest
- Let's consider a single data point $y = Q^T x$
- Then the ith coordinate $y(i) = q_i^T x$ is the orthogonal projection of x onto the ith principal component.
- To perform dimensionality reduction, discard the bottom features of y, as they correspond to directions of smallest variance

Principal Component Analysis

- If we retain all values of Y, We can recover X perfectly as $X = (Q^T)^{-1}Y = QY$ (property of orthonormal matrix Q)
- However, what if we removed the k features of Y corresponding to smallest variance
- Then, we can only recover $\hat{X} = Q\hat{Y}$ where $\hat{Y} = Y$ with the k features with smallest variance set to 0

Principal Component Analysis



- It turns out, approximation of X in this way by the top k components of Y minimizes $\|X - \hat{X}\|_F^2$ over *all* rank-k matrices $\hat{X}$

Outline

- Introduction
- Matrix Factorization Technologies
  - Principal Component Analysis
  - Singular Value Decomposition
  - Nonnegative Matrix Factorization
  - Convolutional Matrix Factorization
  - Regularized Matrix Factorization
  - Inductive Matrix Factorization
- Conclusions and Discussions

Singular Value Decomposition

- Any matrix $X \in R^{m \times n}$ can be represented as

$$X = USV^T$$

where $U \in R^{m \times m}$ and $V \in R^{n \times n}$ are orthonormal and $S \in R^{m \times n}$ is nonzero only on the diagonal

- We can obtain the PCA solution from SVD, without having to form the covariance matrix $\frac{1}{n}XX^T$ explicitly

Singular Value Decomposition

- Substitute $X = USV^T$ in the equation for the covariance matrix.
- Then, we obtain

$$XX^T = USV^T VS^T U = U(SS^T)U$$

  where (SS^T) is diagonal
- We can see immediately that $U(SS^T)U$ has the same structure as $Q\Lambda Q^T$
- Therefore, the U matrix in the SVD is exactly the Q matrix we are trying to obtain in the PCA problem (up to sign changes on eigenvectors)

Outline

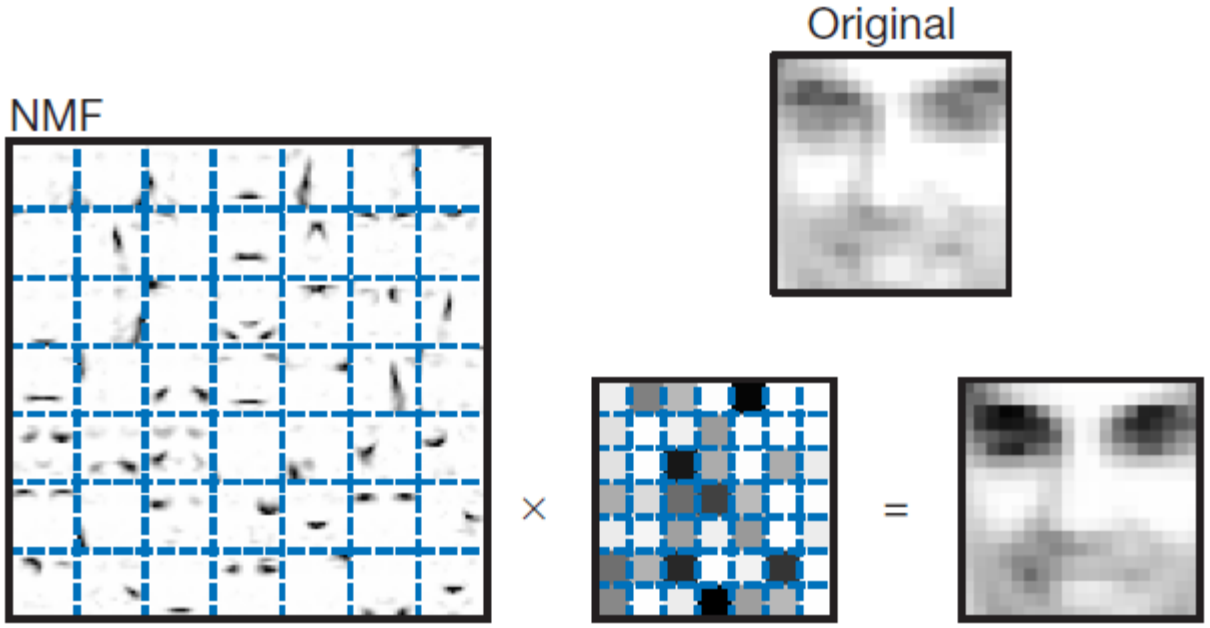
- Introduction
- Matrix Factorization Technologies
  - Principal Component Analysis
  - Singular Value Decomposition
  - Nonnegative Matrix Factorization
  - Convolutional Matrix Factorization
  - Regularized Matrix Factorization
  - Inductive Matrix Factorization
- Conclusions and Discussions

Nonnegative Matrix Factorization

- Consider the following problem
  - M = 2429 facial images
  - Each image of size n = 19 by 19 = 361
  - Matrix V = n by m is the original dataset
  - We want to approximate V by two lower rank matrix W (n by 49) and H (49 by m)
    - V ~ WH
    - Constraints
      - All entries of W and H are non-negative

Nonnegative Matrix Factorization

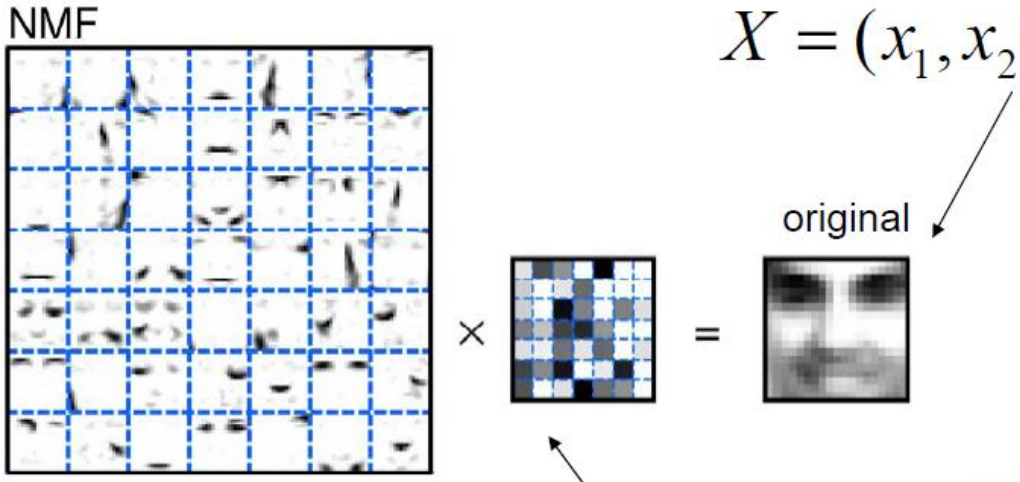
- How well can W and H approximate V
- How can we interpret the result



Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. Nature 401, 788-791 (21 October 1999)

Nonnegative Matrix Factorization

- Factorizing a nonnegative matrix to the product of two low-rank matrices



$$X = (x_1, x_2, \cdots, x_n)$$

$$\min_{\mathbf{F} \in \mathbb{R}_+^{d \times r}, \mathbf{G} \in \mathbb{R}_+^{n \times r}} \mathcal{L}(\mathbf{F}, \mathbf{G})$$

$$\mathcal{L}(\mathbf{F}, \mathbf{G}) = \left\| \mathbf{X} - \mathbf{F}\mathbf{G}^\top \right\|_F^2$$

$$FG^T \approx X$$

$$F = (f_1, f_2, \cdots, f_k) \quad G = (g_1, g_2, \cdots, g_k)$$

Nonnegative Matrix Factorization

- Multiplicative update method

$$\mathbf{F}_{ij} \longleftarrow \mathbf{F}_{ij} \frac{(\mathbf{XG})_{ij}}{(\mathbf{FG}^T\mathbf{G})_{ij}}$$

$$\mathbf{G}_{ij} \longleftarrow \mathbf{G}_{ij} \frac{(\mathbf{X}^T\mathbf{F})_{ij}}{(\mathbf{GF}^T\mathbf{F})_{ij}}$$

Daniel D. Lee and H. Sebastian Seung (2001). Algorithms for Non-negative Matrix Factorization. NIPS 2001.

Nonnegative Matrix Factorization

- Initialize F and G with nonnegative values
- Iterate the following procedure:
  - Fixing $\mathbf{G}^{(t)}$, Solve $\min_{\mathbf{F}} \quad J(\mathbf{F}, \mathbf{G}^{(t)}) = \left\| \mathbf{X} - \mathbf{F}(\mathbf{G}^{(t)})^T \right\|_F^2$
  - Fixing $\mathbf{F}^{(t)}$, Solve $\min_{\mathbf{G}} \quad J(\mathbf{F}^{(t)}, \mathbf{G}) = \left\| \mathbf{X} - \mathbf{F}^{(t)}\mathbf{G}^T \right\|_F$

(1) Projected Gradient: http://www.csie.ntu.edu.tw/~cjlin/nmf/
(2) Newtown Type of Method:
http://www.cs.utexas.edu/users/dmkim/Source/software/nnma/index.html
(3) Block Principal Pivoting: https://sites.google.com/site/jingukim/nmf_bpas.zip?attredirects=0

P. Paatero and U. Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. Environmetrics, 5(1):111–126, 1994

C.-J. Lin. Projected gradient methods for non-negative matrix factorization. *Neural Computation*,19(2007), 2756-2779.

D. Kim, S. Sra, I. S. Dhillon, Fast Newton-type Methods for the Least Squares Nonnegative Matrix Approximation Problem. SDM 2007.

J. Kim and H. Park. Toward Faster Nonnegative Matrix Factorization: A New Algorithm and Comparisons. ICDM 2008.

Nonnegative Matrix Factorization: An Online Algorithm

$$\mathcal{L}(\mathbf{F},\mathbf{G}) = \left\|\mathbf{X} - \mathbf{F}\mathbf{G}^\top\right\|_F^2 = \sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{F}\mathbf{g}_i\|_F^2$$

- Initialize the cluster representatives $\mathbf{F}$.
- Repeat until running out data points. At time t
  - Input a data point (or a chunk of data points) $\mathbf{x}^{(t)}$
  - Compute the optimal $\mathbf{g}^{(t)}$ by $\min_{\mathbf{g}\geqslant 0} \mathcal{L}(\mathbf{F},\mathbf{g})$
  - Update $\mathbf{F}$

Wang, Fei, Ping Li, and Arnd Christian König. "Efficient Document Clustering via Online Nonnegative Matrix Factorizations." In *SDM*, vol. 11, pp. 908-919. 2011.

Online NMF: Updating g

$$\min_{\mathbf{g}^{(t)} \geqslant 0} \left\| \mathbf{x}^{(t)} - \mathbf{F}\mathbf{g}^{(t)} \right\|_F^2$$

Nonnegative Least Square (NLS)

- **Active Set**
- **Projected Gradient**
- **Principal Block Pivoting**

C. L. Lawson and R. J. Hanson. *Solving Least Squares Problems. Society for Industrial Mathematics, 1995.*

C. J. Lin. Projected gradient methods for non-negative matrix factorization. *Neural Computation,* 19(10):2756-2779.

J. Kim and H. Park. Toward faster nonnegative matrix factorization: A new algorithm and comparisons. In *Proceedings of the 8th International Conference on Data Mining (ICDM), pages 353-362, 2008.*

Online NMF: Updating F

loss

$$\mathcal{L}^{(t)}\left(\mathbf{F}^{(t)}\right)=\sum_{s=1}^{t}\left\|\mathbf{x}^{(s)}-\mathbf{F}^{(t)}\mathbf{g}^{(s)}\right\|_F^2=$$

$$\sum_{s=1}^{t}tr\left[\left(\mathbf{x}^{(s)}\right)^\top\mathbf{x}^{(s)}-2\left(\mathbf{x}^{(s)}\right)^\top\mathbf{F}^{(t)}\mathbf{g}^{(s)}+\left(\mathbf{F}^{(t)}\mathbf{g}^{(s)}\right)^\top\mathbf{F}^{(t)}\mathbf{g}^{(s)}\right]$$

gradient

$$\nabla_{\mathbf{F}^{(t)}}\mathcal{L}^{(t)}\left(\mathbf{F}^{(t)}\right)=-2\sum_{s=1}^{t}\left[\mathbf{x}^{(s)}\left(\mathbf{g}^{(s)}\right)^\top-\mathbf{F}^{(t)}\mathbf{g}^{(s)}\left(\mathbf{g}^{(s)}\right)^\top\right]$$

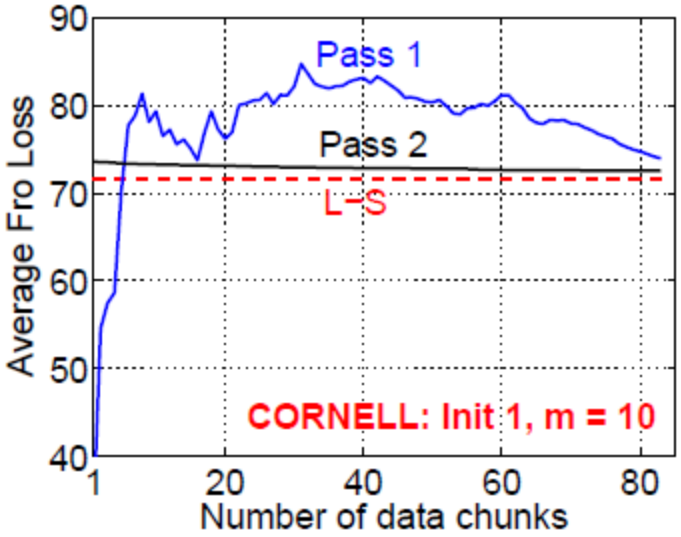
1st order projected gradient descent

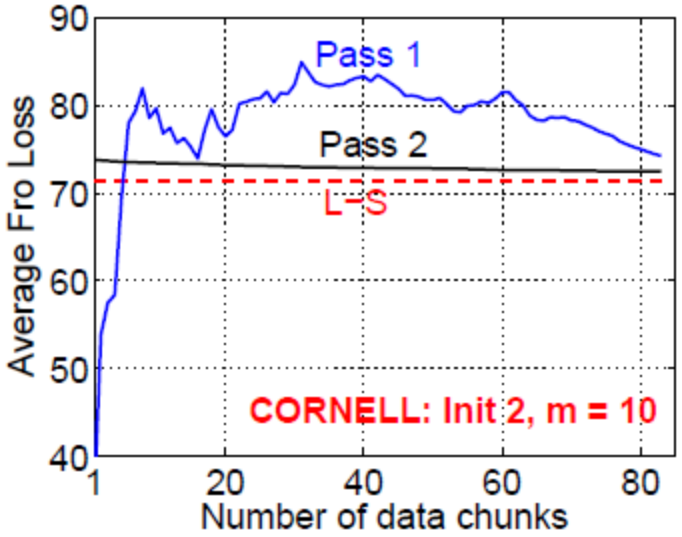
$$\mathbf{F}_{k+1}^{(t)}=P\left[\mathbf{F}_k^{(t)}+2\alpha_k\sum_{s=1}^{t}\left[\mathbf{x}^{(s)}\left(\mathbf{g}^{(s)}\right)^\top-\mathbf{F}_k^{(t)}\mathbf{g}^{(s)}\left(\mathbf{g}^{(s)}\right)^\top\right]\right]$$

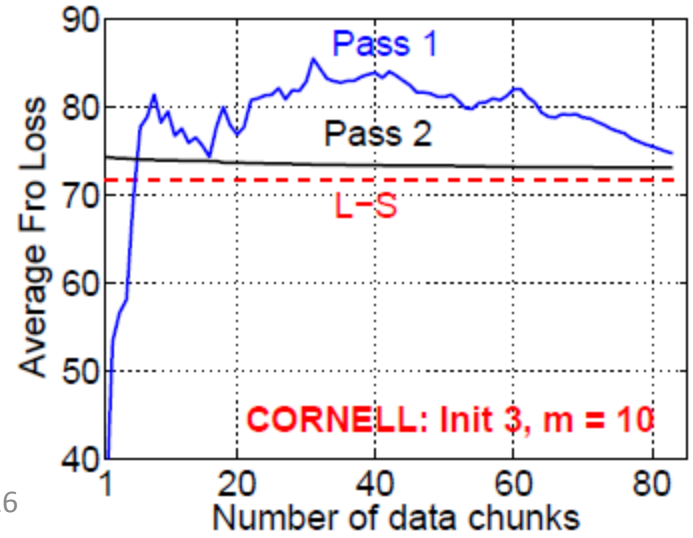
2nd order projected gradient descent

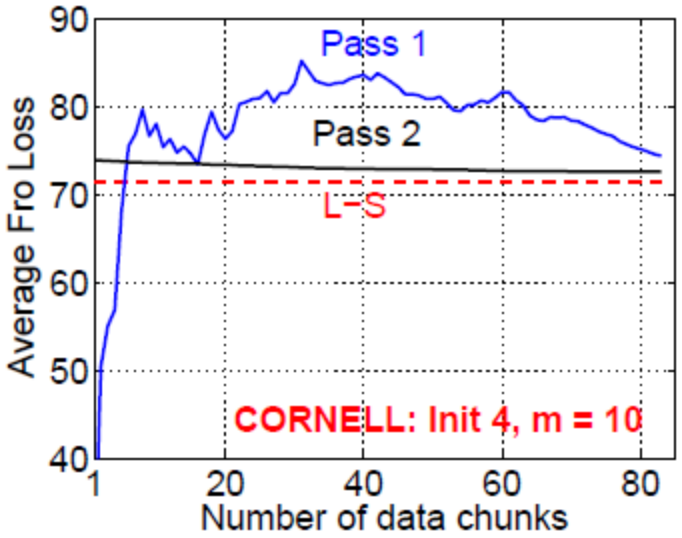
$$\mathbf{F}_{k+1}^{(t)}=P\left[\mathbf{F}_k^{(t)}-\nabla_{\mathbf{F}_k^{(t)}}\mathcal{L}^{(t)}\left(\mathbf{F}_k^{(t)}\right)\mathcal{H}^{-1}\left[\mathcal{L}^{(t)}\left(\mathbf{F}_k^{(t)}\right)\right]\right]$$

Online NMF: Experiments









NMF with Random Projections

Solution Procedure

- Initialize $\mathbf{G}$ with a nonnegative matrix $\mathbf{G}^{(0)}$; $t \leftarrow 0$
- Repeat until a stopping criterion is satisfied
  - Find $\mathbf{F}^{(t+1)}$: $J(\mathbf{F}^{(t+1)}, \mathbf{G}^{(t)}) \leqslant J(\mathbf{F}^{(t)}, \mathbf{G}^{(t)})$
  - Find $\mathbf{G}^{(t+1)}$: $J(\mathbf{F}^{(t+1)}, \mathbf{G}^{(t+1)}) \leqslant J(\mathbf{F}^{(t+1)}, \mathbf{G}^{(t)})$

Objectives

$$\min_{\mathbf{F}} \quad J(\mathbf{F}, \mathbf{G}^{(t)}) = \left\| \mathbf{X} - \mathbf{F}(\mathbf{G}^{(t)})^T \right\|_F^2$$

$$\min_{\mathbf{G}} \quad J(\mathbf{F}^{(t)}, \mathbf{G}) = \left\| \mathbf{X} - \mathbf{F}^{(t)}\mathbf{G}^T \right\|_F^2$$

Projected Objectives

$$\tilde{J}_G(\mathbf{F}, \mathbf{G}) = \left\| \tilde{\mathbf{R}}_d\mathbf{X} - \tilde{\mathbf{R}}_d\mathbf{F}\mathbf{G}^T \right\|_F^2$$

$$\tilde{J}_F(\mathbf{F}, \mathbf{G}) = \left\| \tilde{\mathbf{R}}_n\mathbf{X}^T - \tilde{\mathbf{R}}_n\mathbf{G}\mathbf{F}^T \right\|_F^2$$

$$\mathbf{G}^{(t)} = \underset{\mathbf{G} \geqslant 0}{\operatorname{argmin}} \tilde{J}_G(\mathbf{F}^{(t-1)}, \mathbf{G})$$

$$\mathbf{F}^{(t)} = \underset{\mathbf{F} \geqslant 0}{\operatorname{argmin}} \tilde{J}_F(\mathbf{F}, \mathbf{G}^{(t)})$$

$$\tilde{\mathbf{R}}_d \in \mathbb{R}^{k_1 \times d} = \frac{1}{\sqrt{k_1}}\mathbf{R}_{k_1 \times d} \quad (k_1 \ll d)$$

$$\tilde{\mathbf{R}}_n \in \mathbb{R}^{k_2 \times n} = \frac{1}{\sqrt{k_2}}\mathbf{R}_{k_2 \times n} \quad (k_2 \ll n)$$

Random NMF: Theoretical Analysis

THEOREM *Let* $\mathbf{R} = (r_{ij})$ *be a random* $k \times d$ *matrix, such that each entry* r_{ij} *is chosen independently according to* $\mathcal{N}(0,1)$. *For any vector fixed* $\mathbf{u} \in \mathbb{R}^d$ *and any* $0 < \epsilon < 1$, *let* $\tilde{\mathbf{u}} = \tilde{\mathbf{R}}\mathbf{u} = \frac{1}{\sqrt{k}}\mathbf{R}\mathbf{u}$. *Then,* $E(\|\tilde{\mathbf{u}}\|^2) = \|\mathbf{u}\|^2$ *and with the probability of at least* $1 - e^{-(\epsilon^2 - \epsilon^3)\frac{k}{4}}$

$$\left|\|\tilde{\mathbf{u}}\|^2 - \|\mathbf{u}\|^2\right| \leqslant \epsilon\|\mathbf{u}\|^2$$

COROLLARY *Let* $\tilde{\mathbf{R}} = \frac{1}{\sqrt{k}}\mathbf{R}$ *and* $\mathbf{R}$ *be a normal random matrix, then for any particular* $\mathbf{G}$, $E(\|\tilde{\mathbf{U}}^{(0)}\|_F^2) = \|\mathbf{U}^{(0)}\|_F^2$ *with* $\tilde{\mathbf{U}}^{(0)} = \tilde{\mathbf{R}}\mathbf{U}^{(0)}$, *i.e.,* $E\left[\tilde{J}\left(\tilde{\mathbf{F}}^{(0)}, \mathbf{G}\right)\right] = J\left(\mathbf{F}^{(0)}, \mathbf{G}\right)$. *Moreover, with the probability of at least* $1 - e^{-(\epsilon^2 - \epsilon^3)\frac{k}{4}}$ *with* $0 < \epsilon < 1$, *we have that*

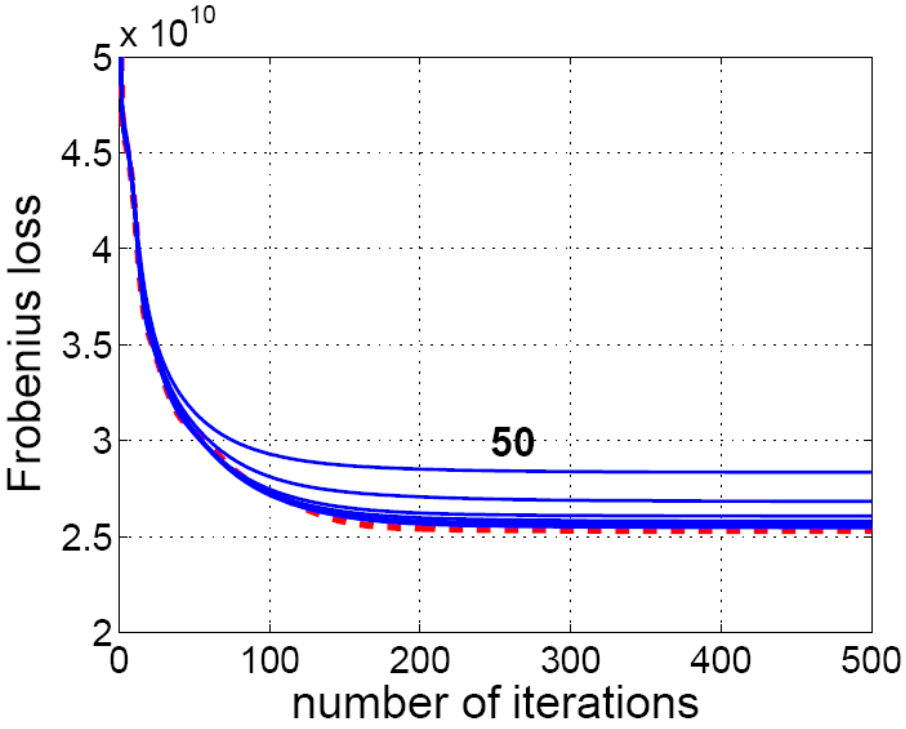
$$\left|\tilde{J}\left(\tilde{\mathbf{F}}^{(0)}, \mathbf{G}\right) - J\left(\mathbf{F}^{(0)}, \mathbf{G}\right)\right| \leqslant \epsilon J\left(\mathbf{F}^{(0)}, \mathbf{G}\right)$$

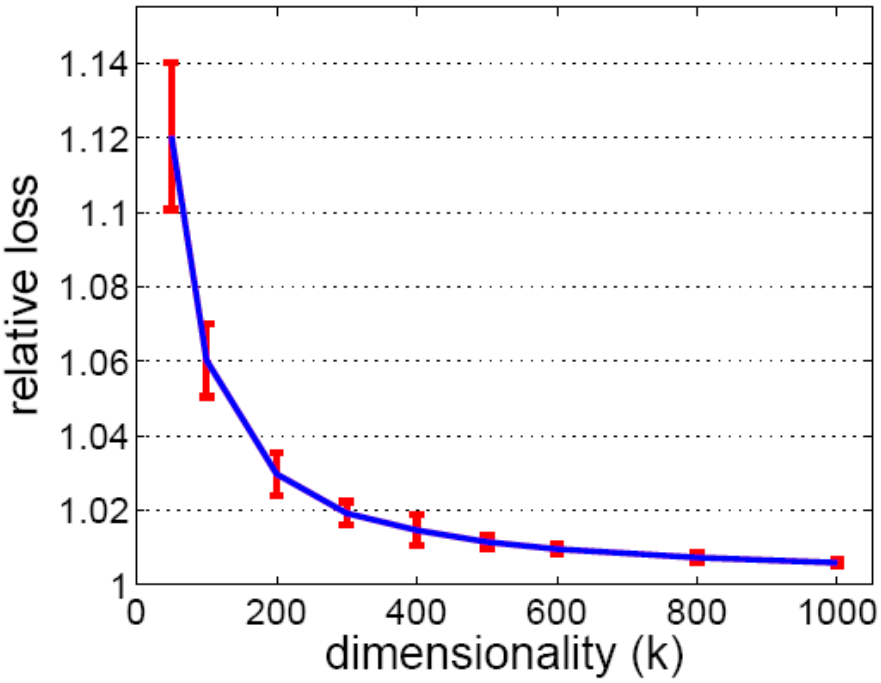
THEOREM *Let* $\mathbf{G}_{opt} = arg\min_{\mathbf{G} \geqslant 0} J(\mathbf{F}^{(0)}, \mathbf{G})$, *and* $\tilde{\mathbf{G}}_{opt} = arg\min_{\mathbf{G} \geqslant 0} \tilde{J}(\tilde{\mathbf{F}}^{(0)}, \mathbf{G})$. *Then with the probability of at least* $1 - e^{-(3\epsilon^2 - \epsilon^3)\frac{k}{108}}$ *with* $0 < \epsilon < 1$, *we have that*

$$J(\mathbf{F}^{(0)}, \tilde{\mathbf{G}}_{opt}) \leqslant (1+\epsilon) J(\mathbf{F}^{(0)}, \mathbf{G}_{opt})$$

R. I. Arriaga and S. Vempala. An algorithmic theory of learning: Robust concepts and random projection. In *FOCS, 1999.*

Random NMF: Experiments



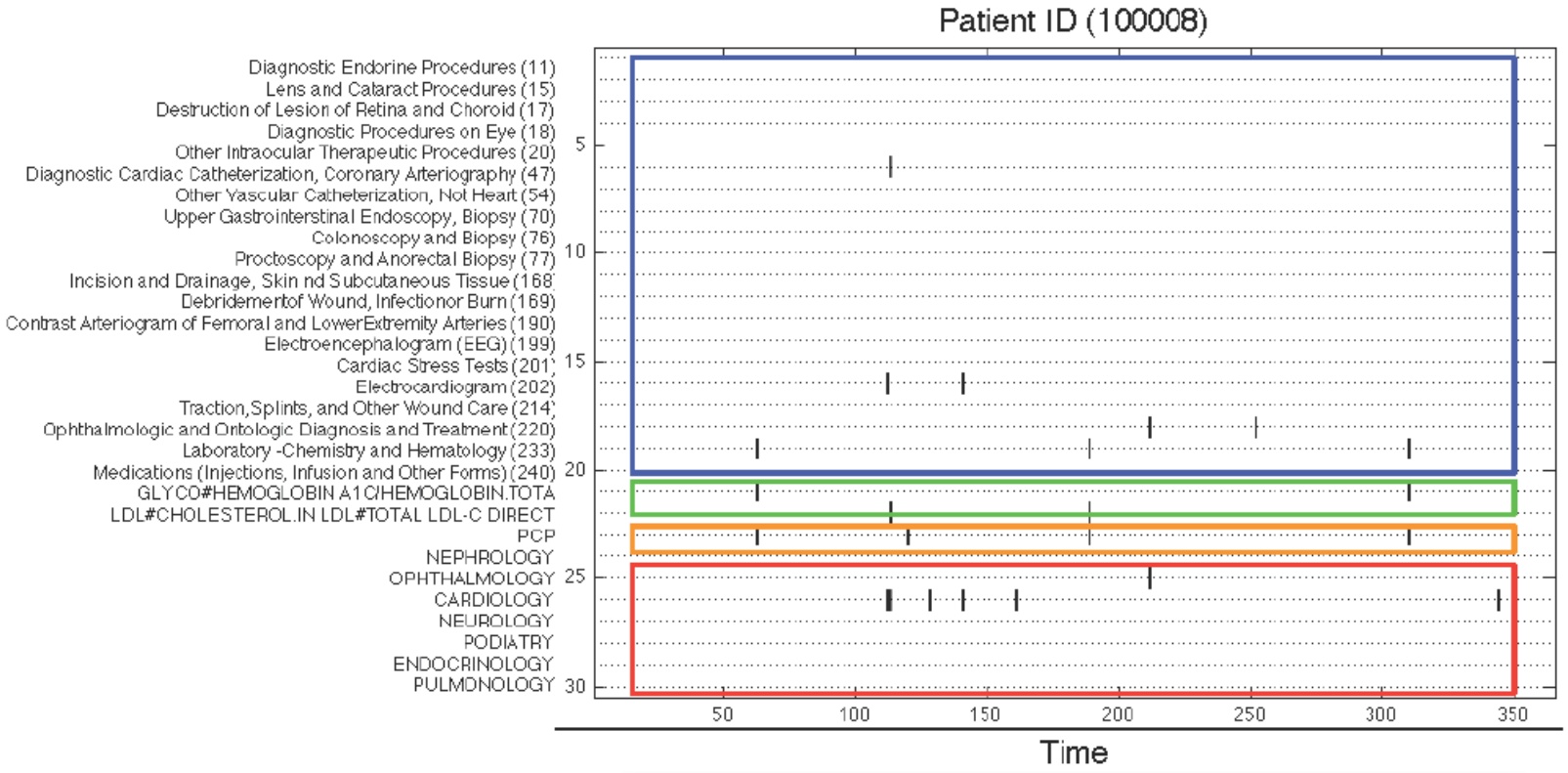


Data Dimension: 12600; Data Size: 203

Outline

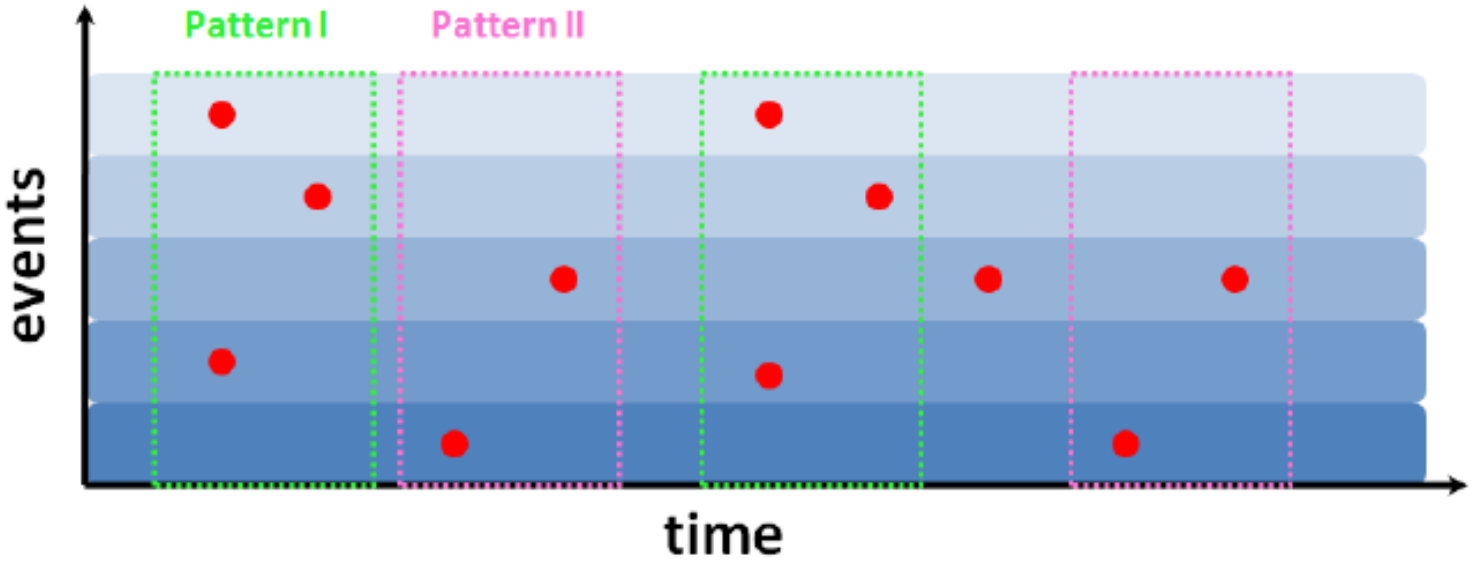
- Introduction
- Matrix Factorization Technologies
  - Principal Component Analysis
  - Singular Value Decomposition
  - Nonnegative Matrix Factorization
  - Convolutional Matrix Factorization
  - Regularized Matrix Factorization
  - Inductive Matrix Factorization
- Conclusions and Discussions

Patient EHR Matrix



Wang, Fei, Noah Lee, Jianying Hu, Jimeng Sun, and Shahram Ebadollahi. "Towards heterogeneous temporal clinical event pattern discovery: a convolutional approach." In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 453-461. ACM, 2012.

Temporal Patterns

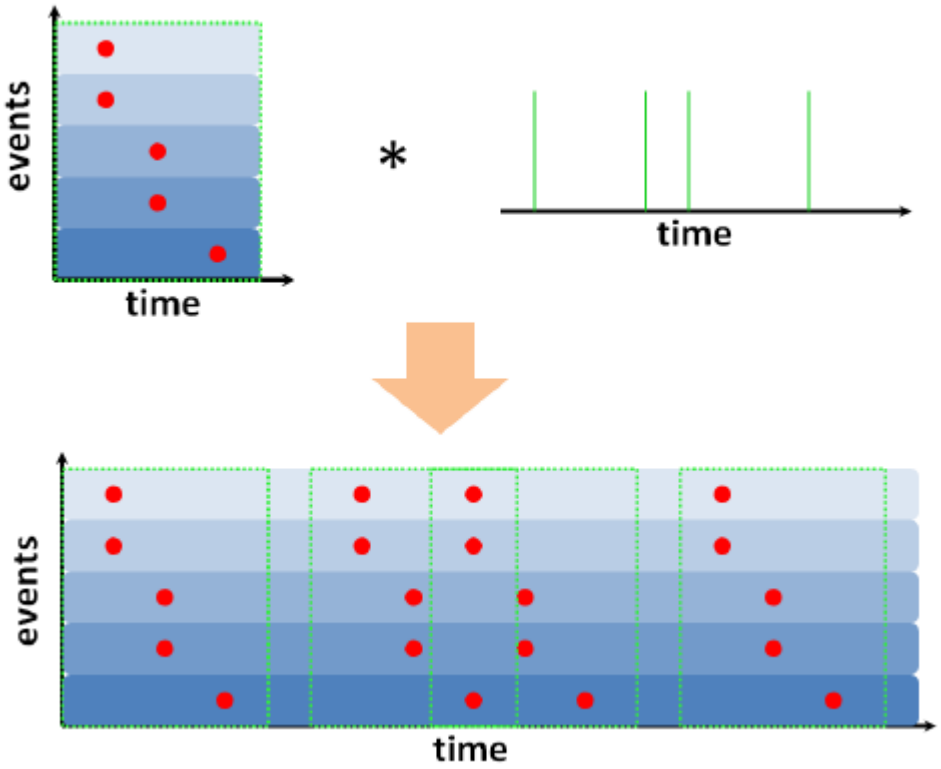


One-Side Convolution

Definition (One-Side Convolution). *The one-side convolution of* $\mathbf{F} \in \mathbb{R}^{n \times m}$ *and* $\mathbf{g} \in \mathbb{R}^{t \times 1}$ *is an* $n \times t$ *matrix with*

$$(\mathbf{F} * \mathbf{g})_{ij} = \sum_{k=1}^{t} g_{j-k+1} F_{ik}$$

Note that $g_j = 0$ *if* $j \leqslant 0$ *or* $j > t$, *and* $F_{ik} = 0$ *if* $k > m$.



One-Side Convolutional NMF

$$\begin{aligned} \min_{\mathcal{F},\mathcal{G}} \quad & \mathcal{J} \\ s.t. \quad & \forall r = 1, \cdots, R; c = 1, \cdots, C \\ & \mathbf{F}^{(r)} \geqslant 0, \mathbf{g}_c^{(r)} \geqslant 0 \end{aligned}$$

$$\mathcal{J} = \sum_{c=1}^{C} d_\beta \left(\mathbf{A}_c \odot \mathbf{X}_c, \mathbf{A}_c \odot \left(\sum_{r=1}^{R} \mathbf{F}^{(r)} * \mathbf{g}_c^{(r)} \right) \right) + \lambda_1 \sum_{r=1}^{R} \|\mathbf{F}^{(r)}\|_1 + \lambda_2 \sum_{c=1}^{C} \sum_{r=1}^{R} \|\mathbf{g}_c^{(r)}\|_1$$

Definition (β-divergence) *The β-divergence between two matrices $\mathbf{A}$ and $\mathbf{B}$ with the same size is*

$$d_\beta(\mathbf{A}, \mathbf{B}) = \frac{1}{\beta(\beta-1)} \sum\nolimits_{ij} \left(A_{ij}^\beta + (\beta-1) B_{ij}^\beta - \beta A_{ij} B_{ij}^{\beta-1} \right)$$

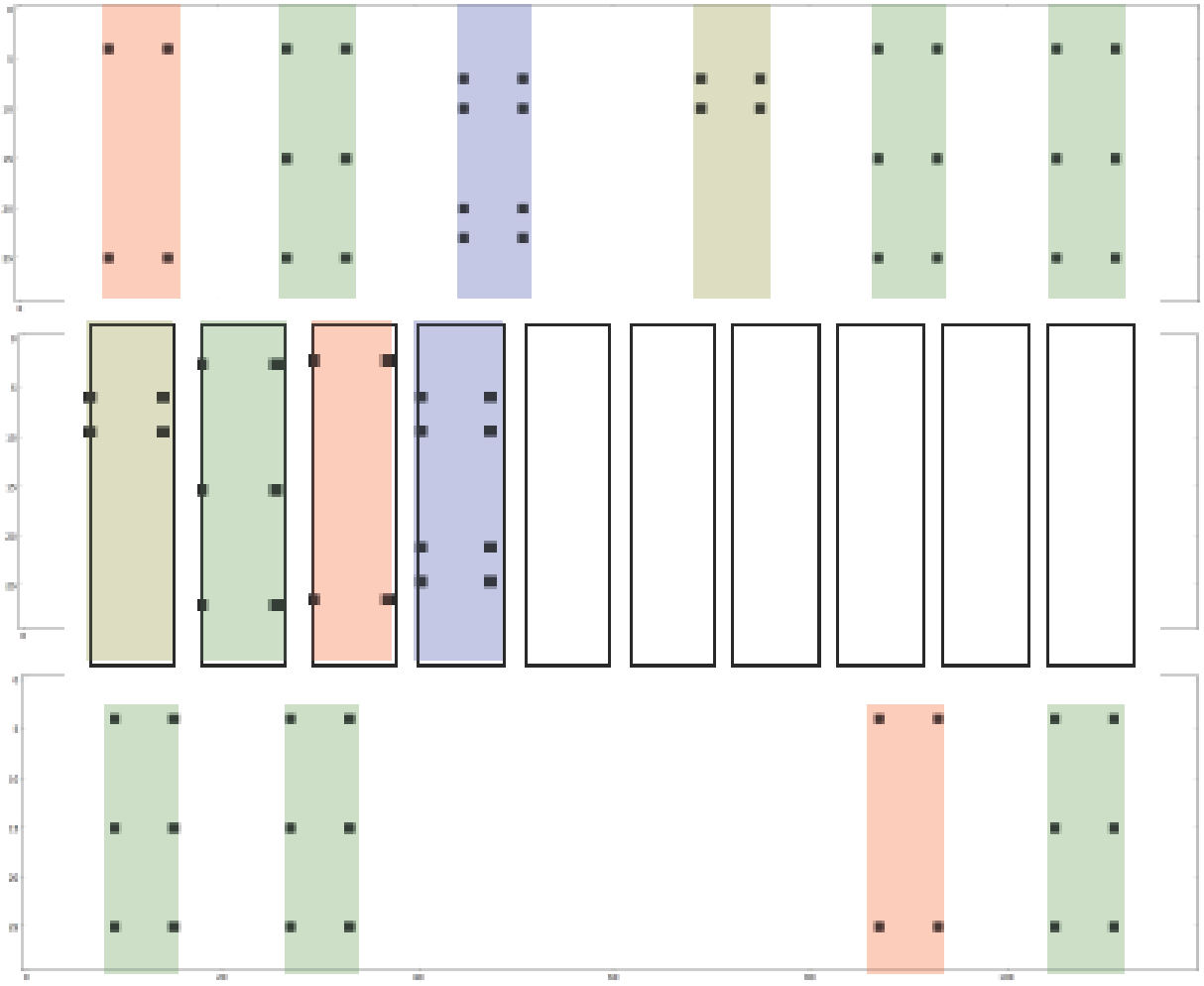
Multiplicative Updates

$$F_{ik}^{(r)} \leftarrow F_{ik}^{(r)} \left(\frac{\sum_{c=1}^{C} \sum_{j=1}^{t} A_{c_{ij}}^{\beta-1} X_{c_{ij}} Y_{c_{ij}}^{\beta-2} g_{c_{j-k+1}}^{(r)}}{\sum_{c=1}^{C} \sum_{j=1}^{t} A_{c_{ij}} Y_{c_{ij}}^{\beta-1} g_{c_{j-k+1}}^{(r)} + \lambda_1} \right)^{\eta(\beta)}$$

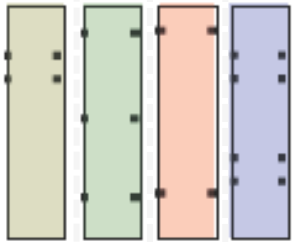
$$g_{c_k}^{(r)} \leftarrow g_c^{(r)} \left(\frac{\sum_{i=1}^{n} \sum_{j=1}^{t} A_{c_{ij}}^{\beta-1} X_{c_{ij}} Y_{c_{ij}}^{\beta-2} F_{i,j-k+1}^{(r)}}{\sum_{i=1}^{n} \sum_{j=1}^{t} A_{c_{ij}} Y_{c_{ij}}^{\beta-1} F_{i,j-k+1}^{(r)} + \lambda_2} \right)^{\eta(\beta)}$$

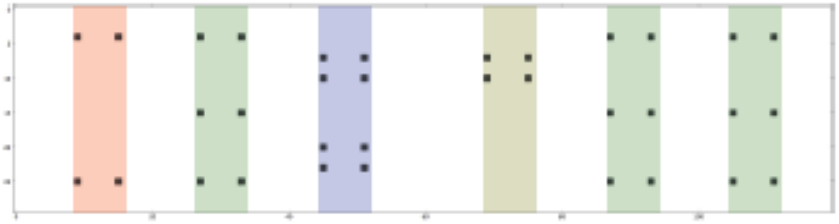
$$\eta(\beta) = \begin{cases} \frac{1}{2-\beta}, & \beta < 1 \\ 1, & 1 \leqslant \beta \leqslant 2 \\ \frac{1}{\beta-1}, & \beta > 2 \end{cases}$$

Synthetic Example

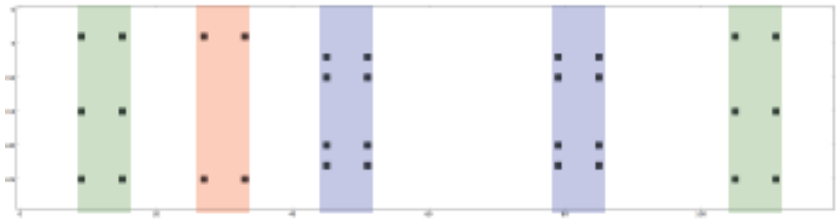


Bag-of-Pattern Representation





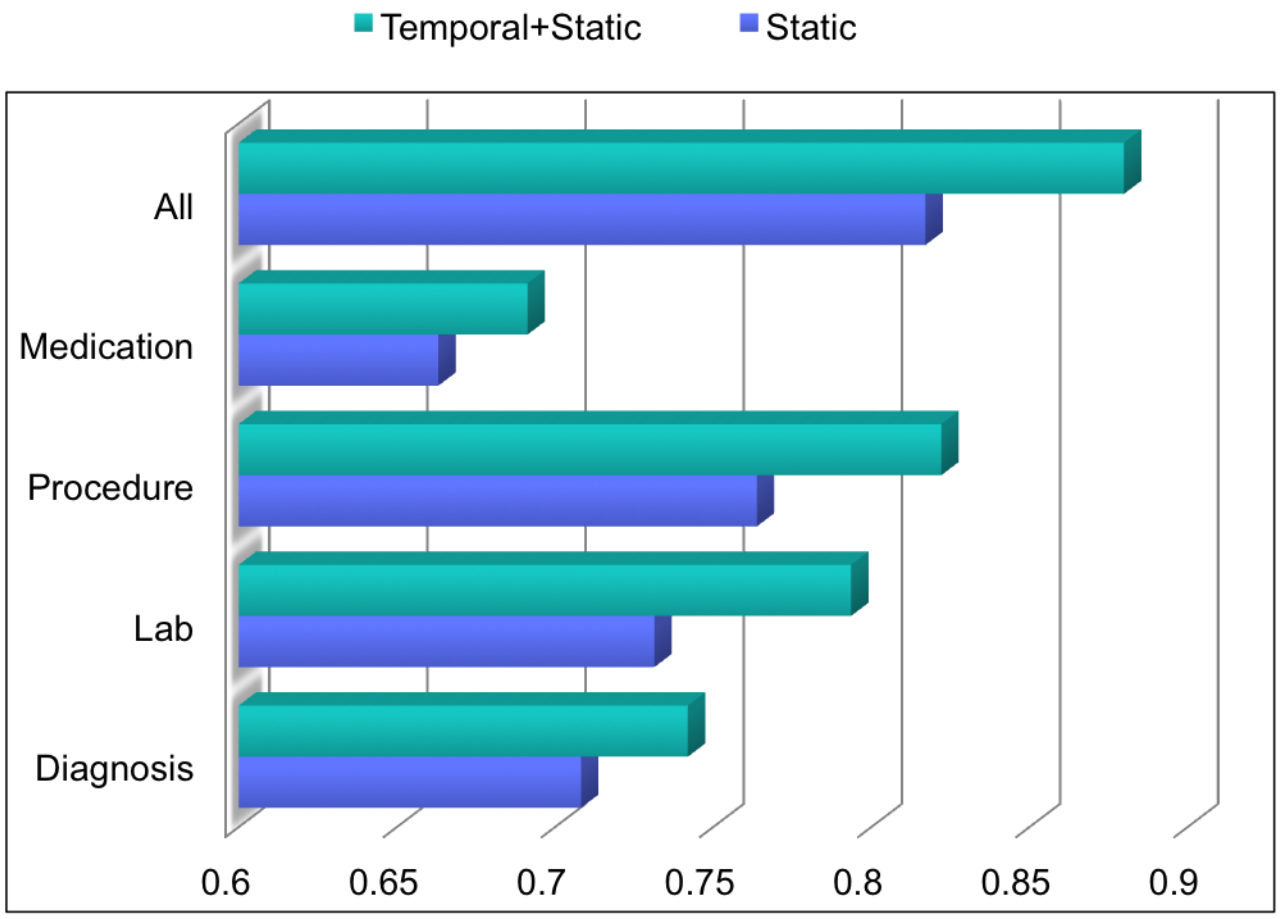
[1 3 1 1]



[0 2 1 2]

[0 3 1 0]

Prediction of CHF Onset Risk



Case: 1,127
Control: 3,850

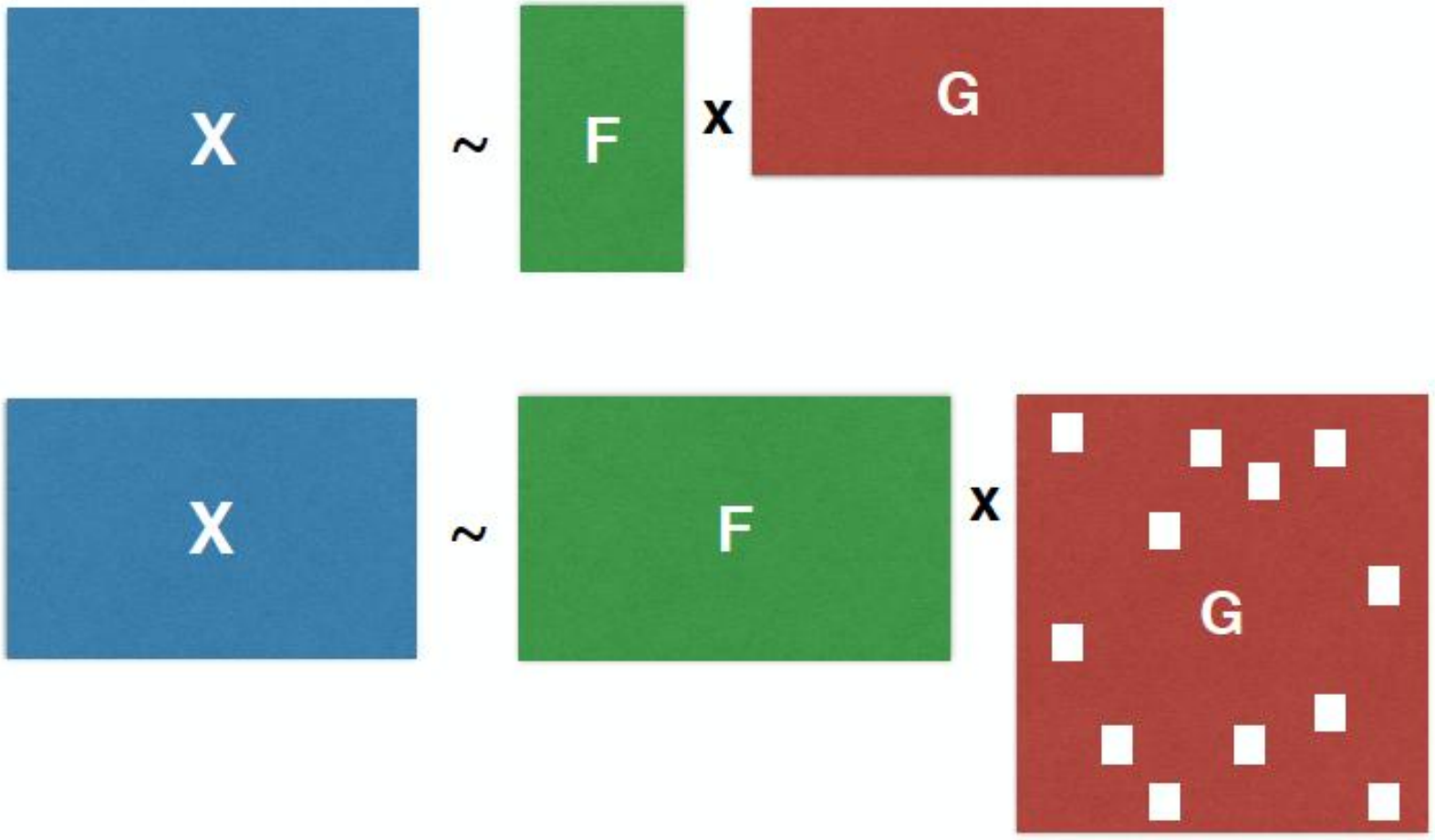
Prediction Window: 180 days
Observation Window: 360 days

Logistic Regression

Outline

- Introduction
- Matrix Factorization Technologies
  - Principal Component Analysis
  - Singular Value Decomposition
  - Nonnegative Matrix Factorization
  - Convolutional Matrix Factorization
  - Regularized Matrix Factorization
  - Inductive Matrix Factorization
- Conclusions and Discussions

Matrix Factorization



Sparsity

$$\begin{aligned} \min_{\mathbf{w}} \quad & \|\mathbf{w}\|_0 \\ s.t. \quad & \mathbf{w} \in \mathcal{C} \end{aligned}$$

Lower Bound

$$\begin{aligned} \min_{\mathbf{w},\mathbf{z}} \quad & \mathbf{1}^\top \mathbf{z} \\ s.t. \quad & |w_i| \leqslant R z_i \;\; \forall i = 1, 2, \cdots, d \\ & \mathbf{w} \in \mathcal{C}, \; z_i \in \{0, 1\} \;\; \forall i = 1, 2, \cdots, d \end{aligned}$$

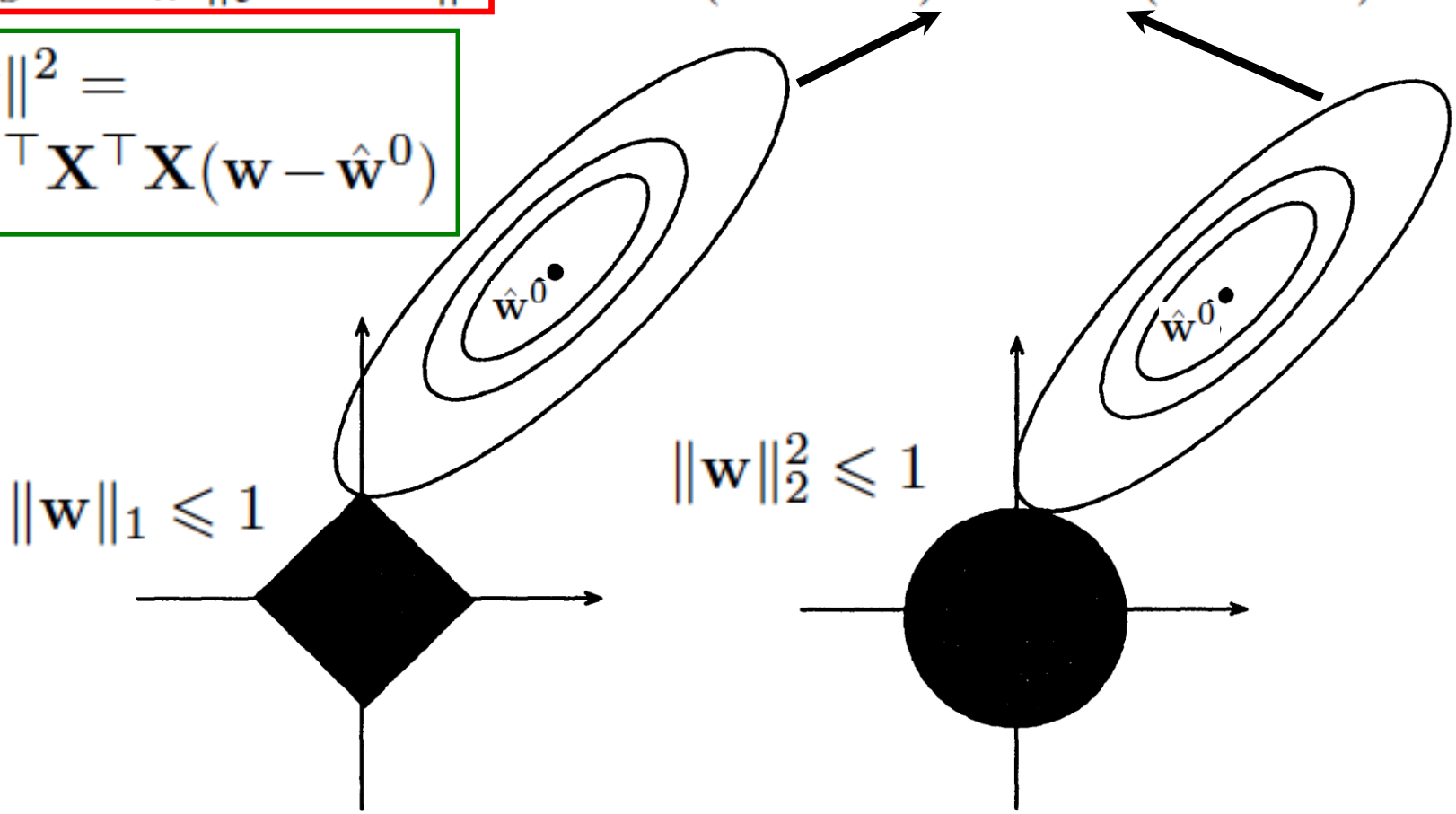
$$\begin{aligned} \min_{\mathbf{w},\mathbf{z}} \quad & \mathbf{1}^\top \mathbf{z} \\ s.t. \quad & |w_i| \leqslant R z_i \;\; \forall i = 1, 2, \cdots, d \\ & \mathbf{w} \in \mathcal{C} \\ & z_i \in [0, 1] \;\; \forall i = 1, 2, \cdots, d \end{aligned}$$

$$\begin{aligned} \min_{\mathbf{w}} \quad & 1/R \|\mathbf{w}\|_1 \\ s.t. \quad & \mathbf{w} \in \mathcal{C} \end{aligned}$$

Sparsity

$$\hat{\mathbf{w}}^0 = arg\min_{\mathbf{w}} \|\mathbf{y}-\mathbf{X}\mathbf{w}\|^2$$

$$\|\mathbf{y}-\mathbf{X}\mathbf{w}\|^2 = (\mathbf{w}-\hat{\mathbf{w}}^0)^\top \mathbf{X}^\top \mathbf{X}(\mathbf{w}-\hat{\mathbf{w}}^0)$$



Robert Tibshirani. Regression Shrinkage and Selection Via the Lasso. Journal of the Royal Statistical Society, Series B. 1994.

Sparsity

- Group Lasso: L1/2 norm
- Exclusive Lasso: L2/1 norm
- Elastic Net Regularization
- Fused Lasso
- Tree Structured Group Lasso

SLEP: A Sparse Learning Package
http://www.public.asu.edu/~jye02/Software/SLEP/

Lukas Meier, Sara Van De Geer, Peter Bühlmann. The group lasso for logistic regression. Journal of the Royal Statistical Society: Series B, 70(1), 53–71, 2008.

Y. Zhou, *R. Jin*, and S. C. H. Hoi. Exclusive Lasso for Multi-task Feature Selection. AISTATS 2010.

Zou, Hui; Hastie, Trevor. Regularization and Variable Selection via the Elastic Net. Journal of the Royal Statistical Society, Series B: 301–320. 2005.

R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, K. Knight. Sparsity and smoothness via the fused lasso. Journal of the Royal Statistical Society: Series B. 67(1), 91–108. 2005.

J. Liu, J. Ye. Moreau-Yosida Regularization for Grouped Tree Structure Learning. NIPS 2010.

Dictionary Learning

Seek for a set of basis which can sparsely represent the data set

$$\sum_{i=1}^{n} \left\| \mathbf{x}_i - \sum_{j=1}^{k} G_{ij} \mathbf{f}_j \right\|^2 \qquad \mathcal{J}_0 = \|\mathbf{X} - \mathbf{F}\mathbf{G}^\top\|_F^2 + \lambda \sum_{i=1}^{n} \|\mathbf{G}_{i\cdot}\|_1$$

Data matrix

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]$$

Coding coefficient matrix, n by k

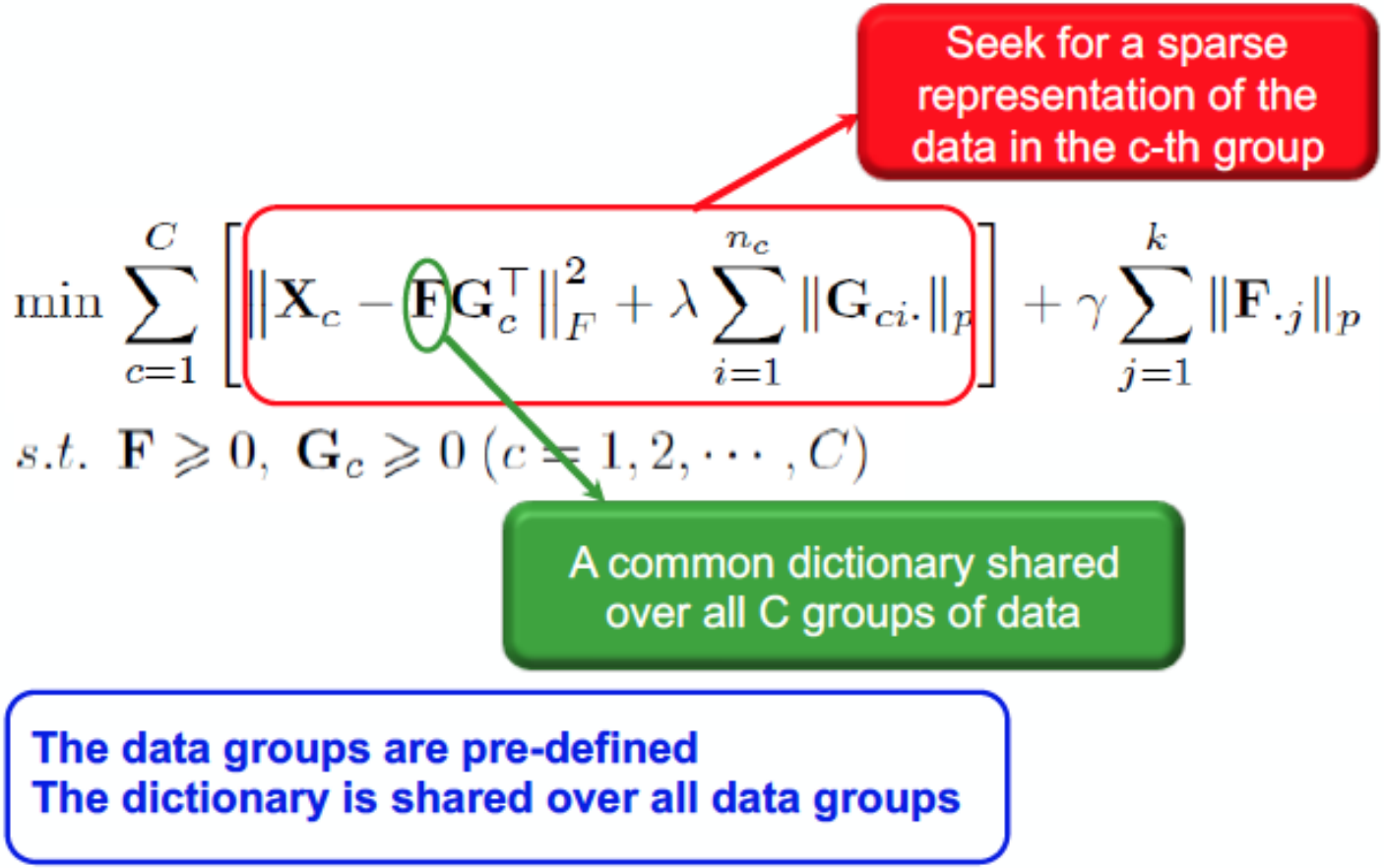
Enforce a sparse representation of the i-th data object

Basis matrix

$$\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_k]$$

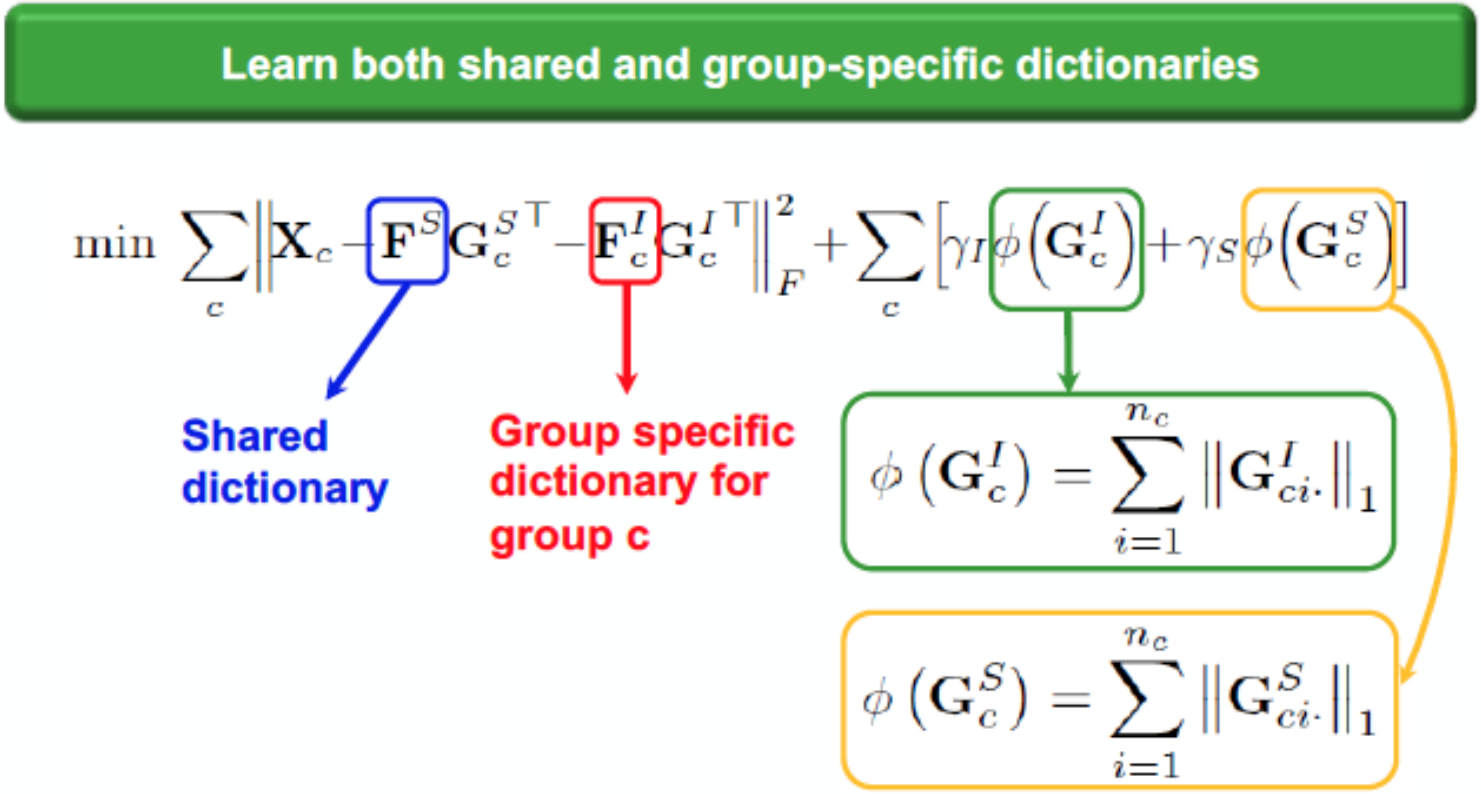
Non-negativity

Group Sparse Coding



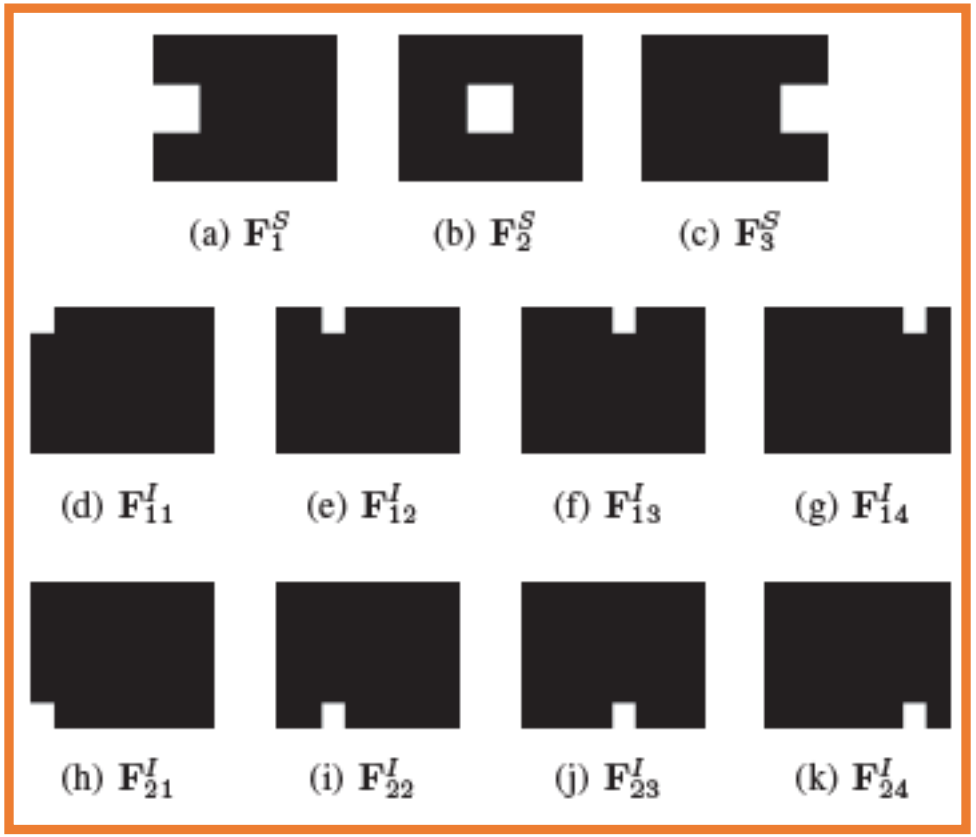
Bengio, Samy, et al. "Group sparse coding." Advances in neural information processing systems. 2009.

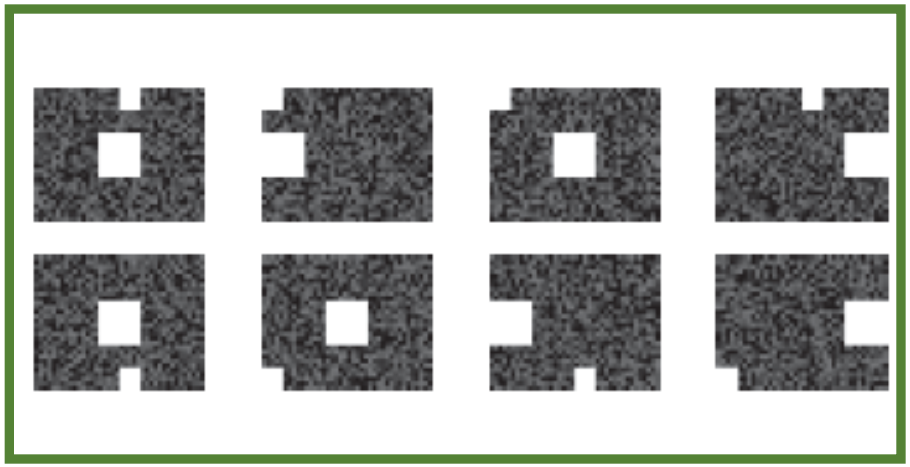
Automatic Group Sparse Coding



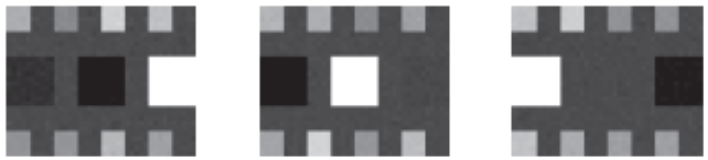
Wang, Fei, Noah Lee, Jimeng Sun, Jianying Hu, and Shahram Ebadollahi. "Automatic Group Sparse Coding." In *AAAI*. 2011.

Synthetic Example





Synthetic Example



(a) $\mathbf{F}_1$ (b) $\mathbf{F}_2$ (c) $\mathbf{F}_3$

(a) $\mathbf{F}_1$ (b) $\mathbf{F}_2$ (c) $\mathbf{F}_3$

(a) $\mathbf{F}_1^S$ (b) $\mathbf{F}_2^S$ (c) $\mathbf{F}_3^S$

(d) $\mathbf{F}_{11}^I$ (e) $\mathbf{F}_{12}^I$ (f) $\mathbf{F}_{13}^I$ (g) $\mathbf{F}_{14}^I$

(h) $\mathbf{F}_{21}^I$ (i) $\mathbf{F}_{22}^I$ (j) $\mathbf{F}_{23}^I$ (k) $\mathbf{F}_{24}^I$

Outline

- Introduction
- Matrix Factorization Technologies
  - Principal Component Analysis
  - Singular Value Decomposition
  - Nonnegative Matrix Factorization
  - Convolutional Matrix Factorization
  - Regularized Matrix Factorization
  - Inductive Matrix Factorization
- Conclusions and Discussions

Inductive Matrix Factorization

$$\min_{W \in \mathbb{R}^{N_g \times k}, H \in \mathbb{R}^{N_d \times k}} \sum_{(i,j) \in \Omega} (P_{ij} - W_i^T H_j)^2 + \frac{1}{2}\lambda(\|W\|_F^2 + \|H\|_F^2)$$

$$\min_{W \in \mathbb{R}^{f_g \times k}, H \in \mathbb{R}^{f_d \times k}} \sum_{(i,j) \in \Omega} \ell(P_{ij}, \mathbf{x}_i^T W H^T \mathbf{y}_j) + \frac{\lambda}{2}(\|W\|_F^2 + \|H\|_F^2)$$

Natarajan, Nagarajan, and Inderjit S. Dhillon. "Inductive matrix completion for predicting gene–disease associations." *Bioinformatics* 30, no. 12 (2014): i60-i68.

Inductive Matrix Factorization