Sparse Matrix Factorization

Behnam Neyshabur [1]   Rina Panigrahy [2]

[1]Toyota Technological Institute at Chicago

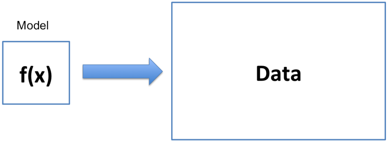
[2]Microsoft Research

November 19, 2013

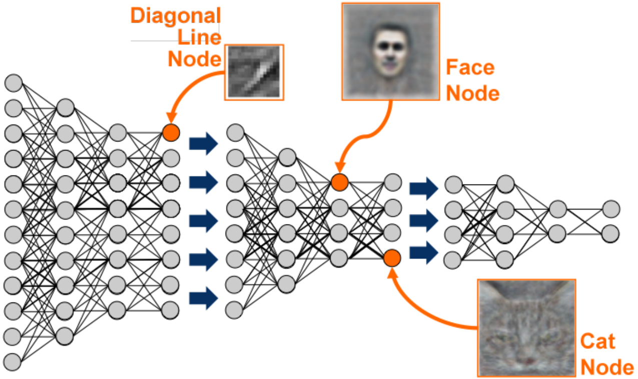
Unsupervised Learning

Unsupervised learning is about describing the data in the best informative way

- Why unsupervised learning?
  - Enormous amount of unlabeled data
    - Facebook: 300 million new photos are added per day
    - Youtube: 150,000 hours of video are uploaded per day
  - Multi-task learning
- How to capture properties of data?

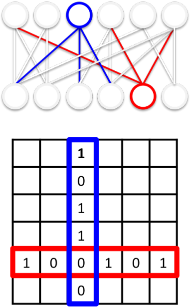


High Level Representation[1]

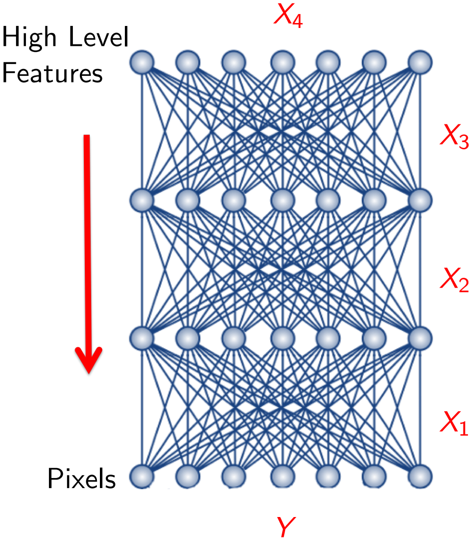


[1] Image from: http://theanalyticsstore.com/deep-learning

Deep Networks

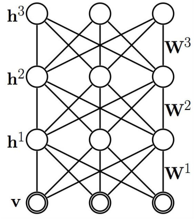


$$Y = \sigma(X_1 \sigma(X_2 \sigma(\ldots X_s)))$$



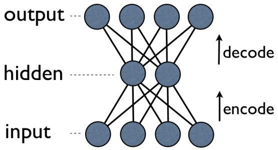
RBMs, Auto-Encoders

RBM



$$v = \sigma(W^1 h^1)$$
$$h^1 = \sigma({W^1}^\top v)$$

Auto-Encoder



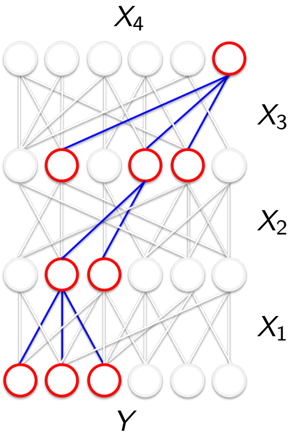
$$v = \sigma(Wh)$$
$$h = \sigma(\tilde{W}^\top v)$$

Sparsity Assumption

Main Observations:

- Each high level feature is composed of a few low level features
- Each image contains a few high level features.
- We are looking for a compact representation of the data:

$$Y = \sigma(X_1\sigma(X_2\sigma(\ldots X_s)))$$



Sparse Matrix Factorization

Problem

Given a matrix Y*, minimize the total sparsity* $\sum_{i=1}^{s} \pi(X_i)$
s.t $Y = \sigma(X_1.\sigma(X_2.\sigma(\ldots X_s)))$ [a]

[a] σ is a term-wise sign function

$$\begin{bmatrix} 4 & 6 & -6 & 3 & 2 & 2 & -4 & 2 \\ 3 & 2 & -6 & 6 & 0 & 1 & 3 & 2 \\ 1 & 2 & 5 & -2 & -6 & -1 & 0 & 3 \\ 3 & 2 & -5 & 4 & 0 & 1 & 4 & 0 \\ 1 & 1 & 3 & 1 & 2 & -1 & -3 & 3 \\ -3 & 4 & 1 & 7 & 1 & 3 & 6 & -4 \\ -4 & -3 & -1 & 3 & 2 & 1 & -5 & 1 \\ 4 & 2 & 3 & -1 & -3 & 2 & 1 & 6 \end{bmatrix} = \begin{bmatrix} 0 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\ +1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\ -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & +1 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\ 0 & 0 & 0 & 0 & 0 & +1 & 0 & +1 \\ +1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ +1 & 0 & +1 & 0 & 0 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\ 0 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & +1 & 0 & -1 & 0 \end{bmatrix} \cdots \begin{bmatrix} 0 & 0 & +1 & 0 & 0 & -1 & 0 & 0 \\ +1 & 0 & 0 & 0 & 0 & 0 & +1 & 0 \\ 0 & 0 & 0 & 0 & +1 & 0 & +1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & +1 \\ 0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\ +1 & 0 & 0 & 0 & 0 & 0 & +1 & 0 \\ 0 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \end{bmatrix}$$

Problem

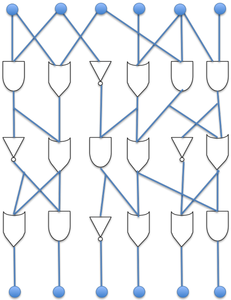
Given a matrix Y*, minimize the total sparsity* $\sum_{i=1}^{s} \pi(X_i)$
s.t. $Y = X_1 X_2 \ldots X_s$

Smallest Circuit

Sparse Matrix Factorization (non-linear) of Y

$$\equiv$$

Smallest circuit that generates Y.



$$a|b : \sigma(a + b - 0.5)$$

$$a\&b : \sigma(a + b - 1.5)$$

$$!a : \sigma(-a)$$

Problem Settings

- Each X_i is a d-sparse random matrix
- Each non-zero entry in X_i is either $+1$ or -1 with equal probability

Problem

Let $Y = X_1 X_2 \ldots X_s$.
Given Y, *find* $X_1, \ldots, X_s$.

Problem

Let $Y = \sigma(X_1.\sigma(X_2.\sigma(\ldots X_s)))$ [a].
Given Y, *find* $X_1, \ldots, X_s$

[a] σ is a term-wise sign function

A Simple Case: $Y = X_1 X_2$

Sparse Decomposition:

$$Y = \begin{bmatrix} +1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & -1 & 0 & +1 \\ +1 & 0 & +1 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & +1 & 0 & 0 & +1 \\ -1 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & +1 & 0 & -1 & 0 \\ 0 & 0 & +1 & 0 & 0 & +1 & 0 & +1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & -1 & 0 & +1 \\ 0 & 0 & +1 & 0 & 0 & +1 & 0 & +1 \\ +1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ +1 & 0 & +1 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & +1 & 0 & 0 & +1 \\ -1 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & +1 & 0 & -1 & 0 \end{bmatrix}$$

A Simple Case: $Y = X_1 X_2$

Sparse Decomposition:

$$Y = \begin{bmatrix} +1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & -1 & 0 & +1 \\ +1 & 0 & +1 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & +1 & 0 & 0 & +1 \\ -1 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & +1 & 0 & -1 & 0 \\ 0 & 0 & +1 & 0 & 0 & +1 & 0 & +1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & -1 & 0 & +1 \\ 0 & 0 & +1 & 0 & 0 & +1 & 0 & +1 \\ +1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ +1 & 0 & +1 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & +1 & 0 & 0 & +1 \\ -1 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & +1 & 0 & -1 & 0 \end{bmatrix}$$

Low Rank Decomposition:

$$Y = \begin{bmatrix} +1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ +1 & -1 & +1 & 0 & 0 & 0 & 0 & 0 \\ +1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ +1 & +1 & +1 & 0 & 0 & 0 & 0 & 0 \\ -1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ +1 & +1 & +1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} -1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ +1 & -1 & +1 & 0 & 0 & 0 & 0 & 0 \\ +1 & +1 & +1 & 0 & 0 & 0 & 0 & 0 \\ +1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ +1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ +1 & +1 & +1 & 0 & 0 & 0 & 0 & 0 \\ -1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & +1 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^{\top}$$

A Simple Case: $Y = X_1 X_2$

Sparse Decomposition:

$$Y = \begin{bmatrix} +1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & -1 & 0 & +1 \\ +1 & 0 & +1 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & +1 & 0 & 0 & +1 \\ -1 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & +1 & 0 & -1 & 0 \\ 0 & 0 & +1 & 0 & 0 & +1 & 0 & +1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & -1 & 0 & +1 \\ 0 & 0 & +1 & 0 & 0 & +1 & 0 & +1 \\ +1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ +1 & 0 & +1 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & +1 & 0 & 0 & +1 \\ -1 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & +1 & 0 & -1 & 0 \end{bmatrix}$$

Low Rank Decomposition:

$$Y = \begin{bmatrix} +1 & +1 & -1 \\ -1 & +1 & -1 \\ +1 & -1 & +1 \\ +1 & +1 & -1 \\ +1 & +1 & +1 \\ -1 & +1 & -1 \\ -1 & +1 & -1 \\ +1 & +1 & +1 \end{bmatrix} \begin{bmatrix} -1 & +1 & -1 \\ +1 & -1 & +1 \\ +1 & +1 & +1 \\ +1 & +1 & -1 \\ +1 & +1 & -1 \\ +1 & +1 & +1 \\ -1 & +1 & -1 \\ -1 & +1 & -1 \end{bmatrix}^\top$$

A Simple Case: $Y = X_1 X_2$

Sparse Decomposition:

$$Y = \begin{bmatrix} +1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & -1 & 0 & +1 \\ +1 & 0 & +1 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & +1 & 0 & 0 & +1 \\ -1 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & +1 & 0 & -1 & 0 \\ 0 & 0 & +1 & 0 & 0 & +1 & 0 & +1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & -1 & 0 & +1 \\ 0 & 0 & +1 & 0 & 0 & +1 & 0 & +1 \\ +1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ +1 & 0 & +1 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & +1 & 0 & 0 & +1 \\ -1 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & +1 & 0 & -1 & 0 \end{bmatrix}$$

Low Rank Decomposition:

$$Y = \begin{bmatrix} +1 & +1 & -1 \\ -1 & +1 & -1 \\ +1 & -1 & +1 \\ +1 & +1 & -1 \\ +1 & +1 & +1 \\ -1 & +1 & -1 \\ -1 & +1 & -1 \\ +1 & +1 & +1 \end{bmatrix} \begin{bmatrix} -1 & +1 & -1 \\ +1 & -1 & +1 \\ +1 & +1 & +1 \\ +1 & +1 & -1 \\ +1 & +1 & -1 \\ +1 & +1 & +1 \\ -1 & +1 & -1 \\ -1 & +1 & -1 \end{bmatrix}^\top$$

A Special Case: $Y = AX$

Dictionary Learning: $A = [\, A_1 \; A_2 \; A_3 \; A_4 \; A_5 \; A_6 \; A_7 \; A_8 \,]$

$$Y = A \begin{bmatrix} 0 & -1 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & -1 & 0 & +1 \\ 0 & 0 & +1 & 0 & 0 & +1 & 0 & +1 \\ +1 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\ +1 & 0 & +1 & 0 & 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & +1 & 0 & 0 & +1 \\ -1 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & +1 & 0 & -1 & 0 \end{bmatrix}$$

PCA: $A = [\, A_1 \; A_2 \; A_3 \,]$

$$Y = A \begin{bmatrix} -1 & +1 & -1 \\ +1 & -1 & +1 \\ +1 & +1 & +1 \\ +1 & +1 & -1 \\ +1 & +1 & -1 \\ +1 & +1 & +1 \\ -1 & +1 & -1 \\ -1 & +1 & -1 \end{bmatrix}^{\top}$$

What do we already know?

What do we already know?

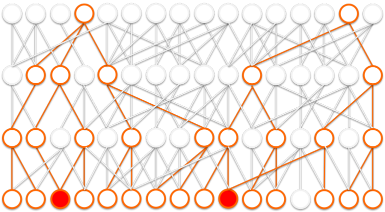
A very recent work by Arora et al:

- Polynomial time algorithm for non-linear factorizations of random d-sparse networks: $Y = \sigma(X_1.\sigma(X_2.\sigma(\ldots X_s)))$
- Sparsity $d \leq n^{1/5}$
- Depth $s \leq \log_d n$.
- When depth $s \leq \log_d n$, linear and nonlinear cases are almost the same.

What do we already know?

A very recent work by Arora et al:

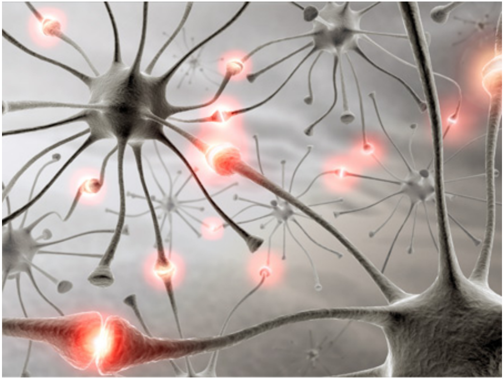
- Polynomial time algorithm for non-linear factorizations of random d-sparse networks: $Y = \sigma(X_1.\sigma(X_2.\sigma(\ldots X_s)))$
- Sparsity $d \leq n^{1/5}$
- Depth $s \leq \log_d n$.
- When depth $s \leq \log_d n$, linear and nonlinear cases are almost the same.



Our Goal

Develope a natural algorithm based on the following fundamental rule:

Neurons that fire together, wire together!



Main Result

Theorem

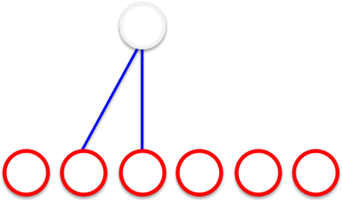
If

- $Y = X_1 X_2 \dots X_s$
- *sparsity* $d \leq n^{1/6}$
- $s \leq \frac{\sqrt{n}}{d}$

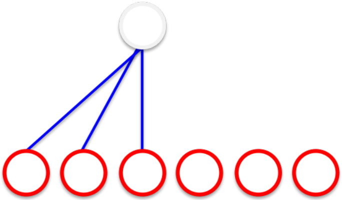
w.h.p. if X_i *is invertible, there is a natural algorithm that computes* $X_1, \dots, X_s$ *in a polynomial time.*

Intuition

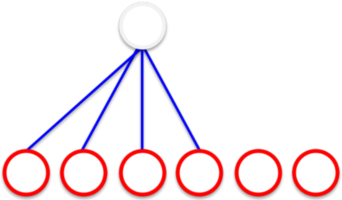
Intuition



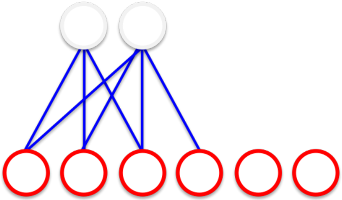
Intuition



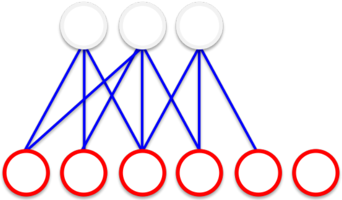
Intuition



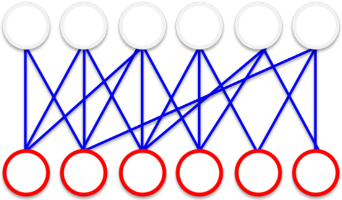
Intuition



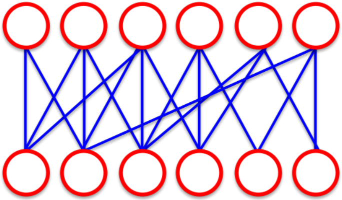
Intuition



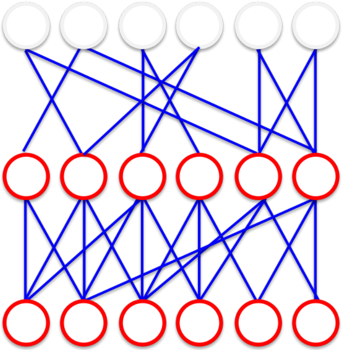
Intuition



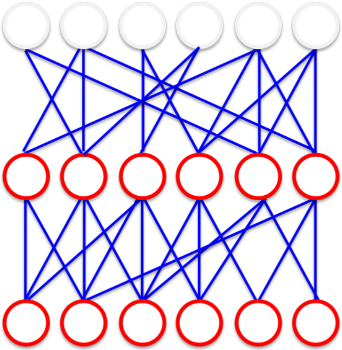
Intuition



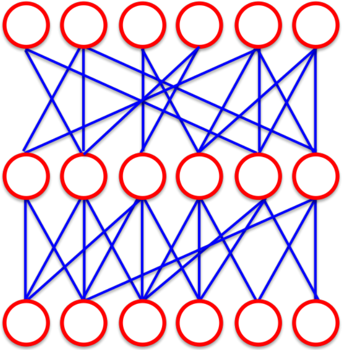
Intuition



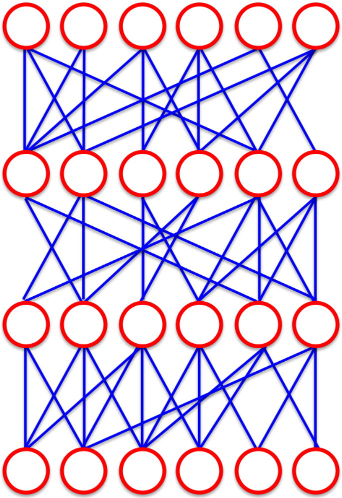
Intuition



Intuition



Intuition



Proof Sketch

$$Y = X_1 X_2 \ldots X_s$$

Proof Sketch

$$Y = \underbrace{X_1}_{X} \underbrace{X_2 \ldots X_s}_{Z}$$

- $YY^\top \sim XX^\top$
- $XX^\top = \text{round}(YY^\top)$
- Getting X from $XX^\top$.
- Recovering Z.

Distribution of Y

We want to show that :

$$x_i.x_j \sim (x_i Z).(x_j Z)$$

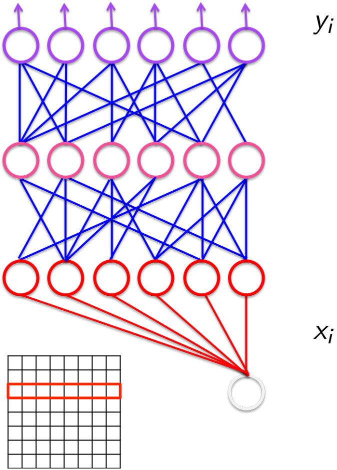
Lemma

If

- $q = x_i Z$
- $x_i \sim \mathcal{N}(0,1)^n$

w.h.p. $\Phi_q(t) \leq e^{\frac{t^2}{2} \pm t^2 \tilde{O}(\frac{\ell}{\sqrt{n}})}$.

Characteristic function is defined as: $\Phi_X(t) = E[e^{tX}]$ and for a gaussian variable X we have that $\Phi_X(t) = e^{t^2/2}$



Distribution of $YY^\top$

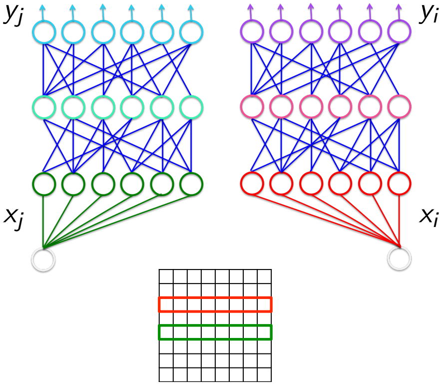
Lemma

If

- $q = x_i Z$ *and* $w = x_j Z$.
- $x_i, x_j \sim \mathcal{N}(0,1)^n$

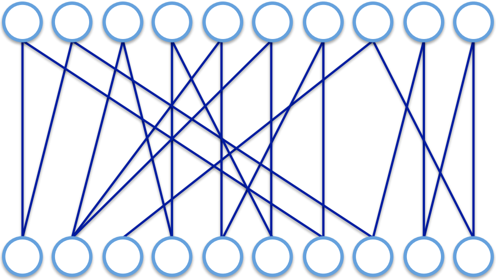
w.h.p.

$$\Phi_{q,w}(s,t) \le e^{\frac{t^2+s^2}{2} \pm (s+t)^2 \tilde{O}(\frac{\ell}{\sqrt{n}})}$$

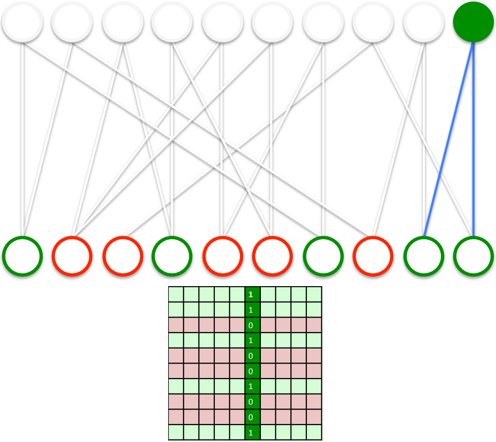


$$x_i x_j^\top = y_i y_j^\top \pm \frac{1}{2}$$

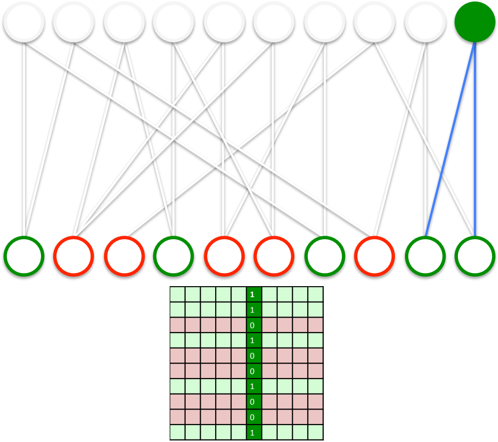
Recovering X from $XX^\top$



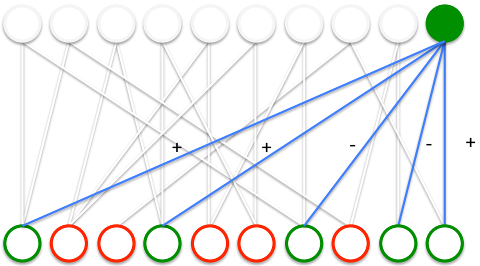
Recovering X from $XX^\top$



Recovering X from $XX^\top$



Recovering X from $XX^\top$



Recovering Z

- By calculating the inverse of X directly : $Z = X^{-1}Y$

Recovering Z

- By calculating the inverse of X directly : $Z = X^{-1}Y$
- By iterative corrections:

$$\mathrm{err}_n = Y(I - \frac{1}{\alpha}XX^\top)^n$$

Conclusion and Future Works

- There exists a simple natural algorithm that is guaranteed to recover random sparse networks of depth $O(\sqrt{n}/d)$.
- Future works:
  - Extending the algorithm and proves to non-linear case with $O(\sqrt{n}/d)$ layers.
  - Showing that we can recover denser matrices by more careful analysis
  - Develop a practical learning method based on the proposed algorithm and evaluate it on the real data

Thank You!

Dictionary Learning under Group Sparsity Assumption

Detection

Do we always need to detect different parts of human body before detecting a human?

Detection

Do we always need to detect different parts of human body before detecting a human?

Detection

Do we always need to detect different parts of human body before detecting a human?

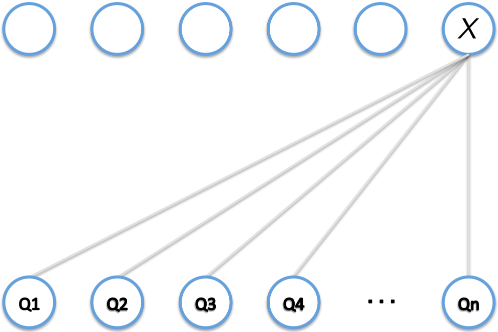
Group Sparsity

- Each group is a high level feature that includes several low level features (parts).
- Each part can be seen in several groups.
- Given matrix Y such that $Y = AX$, find A and X under group sparsity assumption for X.

The general idea

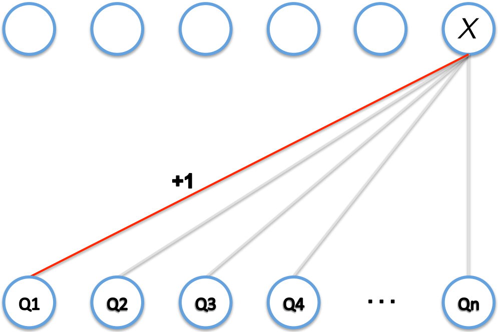
- Detect all groups.
- Find a correlation clustering of groups.
- Show that each cluster corresponds to a part.
- Set A_i to be the average over all Y_j where Y_j has the part i.

Distribution of $YY^\top$



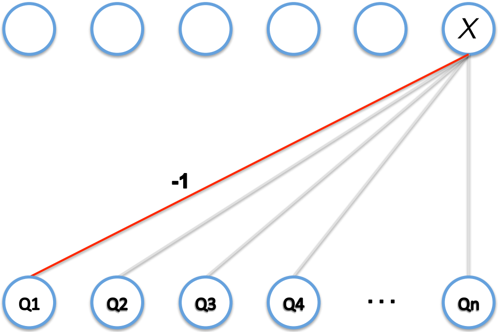
$E[X^k]$

Distribution of $YY^\top$



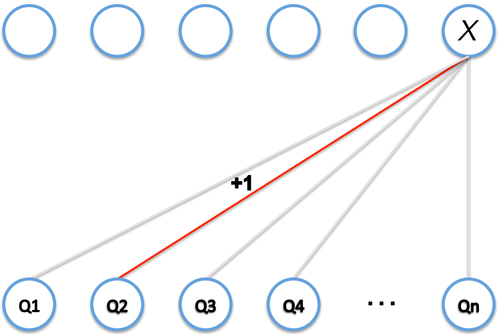
$$E[X^k] = \frac{Q_1^k}{2n}$$

Distribution of $YY^\top$



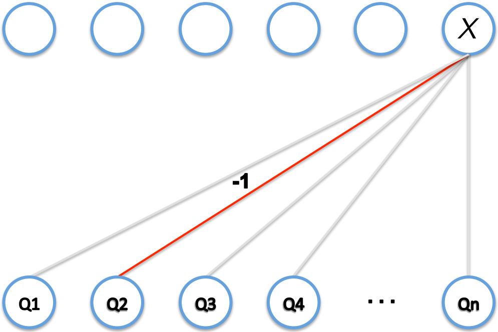
$$E[X^k] = \frac{Q_1^k}{2n} + \frac{(-Q_1)^k}{2n}$$

Distribution of $YY^\top$



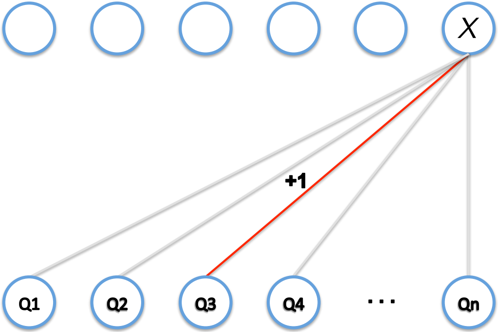
$$E[X^k] = \frac{Q_1^k}{2n} + \frac{(-Q_1)^k}{2n} + \frac{Q_2^k}{2n}$$

Distribution of $YY^\top$



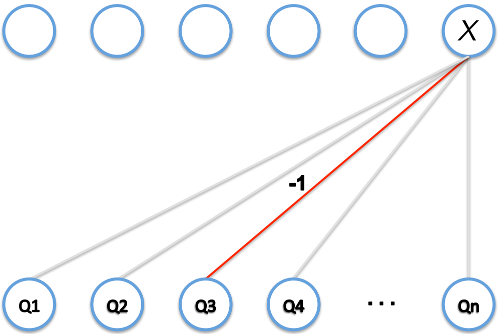
$$E[X^k] = \frac{Q_1^k}{2n} + \frac{(-Q_1)^k}{2n} + \frac{Q_2^k}{2n} + \frac{(-Q_2)^k}{2n}$$

Distribution of $YY^\top$



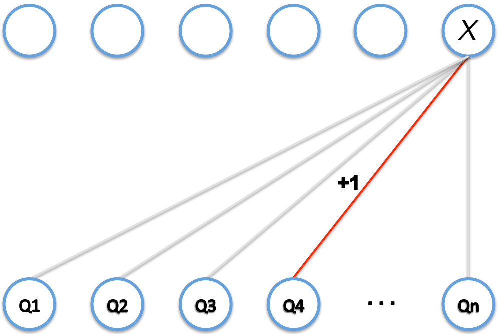
$$E[X^k] = \frac{Q_1^k}{2n} + \frac{(-Q_1)^k}{2n} + \frac{Q_2^k}{2n} + \frac{(-Q_2)^k}{2n} + \ldots$$

Distribution of $YY^\top$



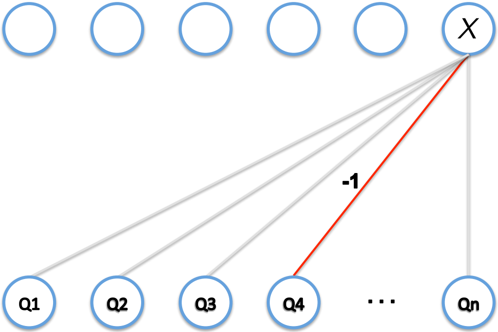
$$E[X^k] = \frac{Q_1^k}{2n} + \frac{(-Q_1)^k}{2n} + \frac{Q_2^k}{2n} + \frac{(-Q_2)^k}{2n} + \dots$$

Distribution of $YY^\top$



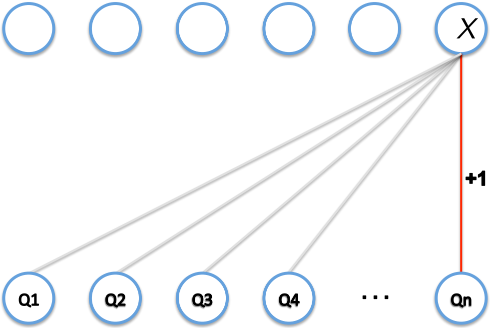
$$E[X^k] = \frac{Q_1^k}{2n} + \frac{(-Q_1)^k}{2n} + \frac{Q_2^k}{2n} + \frac{(-Q_2)^k}{2n} + \dots$$

Distribution of $YY^\top$



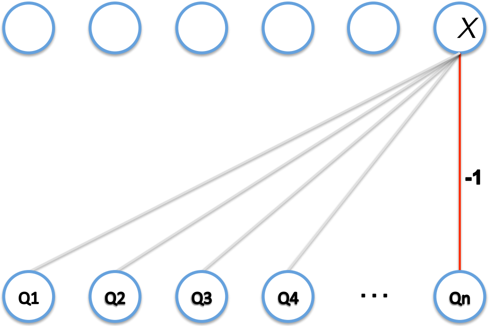
$$E[X^k] = \frac{Q_1^k}{2n} + \frac{(-Q_1)^k}{2n} + \frac{Q_2^k}{2n} + \frac{(-Q_2)^k}{2n} + \ldots$$

Distribution of $YY^\top$



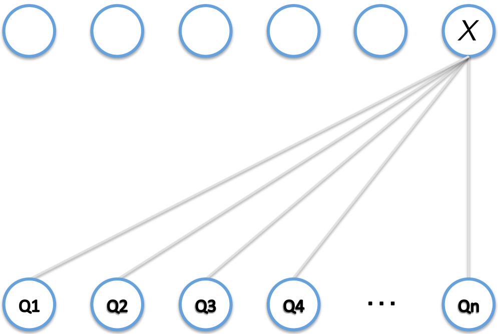
$$E[X^k] = \frac{Q_1^k}{2n} + \frac{(-Q_1)^k}{2n} + \frac{Q_2^k}{2n} + \frac{(-Q_2)^k}{2n} + \cdots + \frac{Q_n^k}{2n}$$

Distribution of $YY^\top$



$$E[X^k] = \frac{Q_1^k}{2n} + \frac{(-Q_1)^k}{2n} + \frac{Q_2^k}{2n} + \frac{(-Q_2)^k}{2n} + \cdots + \frac{Q_n^k}{2n} + \frac{(-Q_n)^k}{2n}$$

Distribution of $YY^\top$



$$E[X^k] = E[Q^k] + \frac{\sum_i Q_i^k - E[Q^k]}{n} \leq E[Q^k] + \frac{\log^k n}{\sqrt{n}}$$

Recovering $XX^\top$ from $YY^\top$

$$Y_i Y_j \top = X_i Z_1 \ldots Z_1 \top X_j$$

$$\begin{bmatrix} 1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ f_1 & f_2 & f_3 & f_4 \end{bmatrix} \ldots \begin{bmatrix} a_1 & b_1 & c_1 & f_1 \\ a_2 & b_2 & c_2 & f_2 \\ a_3 & b_3 & c_3 & f_3 \\ a_4 & b_4 & c_4 & f_4 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

Recovering $XX^\top$ from $YY^\top$

$$Y_i Y_j \top = X_i Z_1 \ldots Z_1 \top X_j$$

$$\begin{bmatrix} 1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ f_1 & f_2 & f_3 & f_4 \end{bmatrix} \ldots \begin{bmatrix} a_1 & b_1 & c_1 & f_1 \\ a_2 & b_2 & c_2 & f_2 \\ a_3 & b_3 & c_3 & f_3 \\ a_4 & b_4 & c_4 & f_4 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

$$Y_i Y_j \top = -1 \pm \epsilon_1$$

Recovering $XX^\top$ from $YY^\top$

$$Y_i Y_j \top = X_i Z_1 \dots Z_1 \top X_j$$

$$\begin{bmatrix} 1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ f_1 & f_2 & f_3 & f_4 \end{bmatrix} \dots \begin{bmatrix} a_1 & b_1 & c_1 & f_1 \\ a_2 & b_2 & c_2 & f_2 \\ a_3 & b_3 & c_3 & f_3 \\ a_4 & b_4 & c_4 & f_4 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

$$Y_i Y_j \top = -1 \pm (\epsilon_1 + \epsilon_2)$$

Recovering $XX^\top$ from $YY^\top$

$$Y_i Y_j \top = X_i Z_1 \dots Z_1 \top X_j$$

$$\begin{bmatrix} 1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ f_1 & f_2 & f_3 & f_4 \end{bmatrix} \dots \begin{bmatrix} a_1 & b_1 & c_1 & f_1 \\ a_2 & b_2 & c_2 & f_2 \\ a_3 & b_3 & c_3 & f_3 \\ a_4 & b_4 & c_4 & f_4 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

$$Y_i Y_j \top = -1 \pm (\epsilon_1 + 2\epsilon_2)$$

Recovering $XX^\top$ from $YY^\top$

$$Y_i Y_j \top = X_i Z_1 \ldots Z_1 \top X_j$$

$$\begin{bmatrix} 1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ f_1 & f_2 & f_3 & f_4 \end{bmatrix} \ldots \begin{bmatrix} a_1 & b_1 & c_1 & f_1 \\ a_2 & b_2 & c_2 & f_2 \\ a_3 & b_3 & c_3 & f_3 \\ a_4 & b_4 & c_4 & f_4 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

$$Y_i Y_j \top = -2 \pm (2\epsilon_1 + 4\epsilon_2)$$

Recovering $XX^\top$ from $YY^\top$

$$Y_i Y_j \top = X_i Z_1 \dots Z_1 \top X_j$$

$$\begin{bmatrix} 1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ f_1 & f_2 & f_3 & f_4 \end{bmatrix} \dots \begin{bmatrix} a_1 & b_1 & c_1 & f_1 \\ a_2 & b_2 & c_2 & f_2 \\ a_3 & b_3 & c_3 & f_3 \\ a_4 & b_4 & c_4 & f_4 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

$$Y_i Y_j \top = -2 \pm (2\epsilon_1 + 7\epsilon_2)$$

Recovering $XX^\top$ from $YY^\top$

$$Y_i Y_j{}^\top = X_i Z_1 \dots Z_1{}^\top X_j$$

$$\begin{bmatrix} 1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ f_1 & f_2 & f_3 & f_4 \end{bmatrix} \dots \begin{bmatrix} a_1 & b_1 & c_1 & f_1 \\ a_2 & b_2 & c_2 & f_2 \\ a_3 & b_3 & c_3 & f_3 \\ a_4 & b_4 & c_4 & f_4 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

$$Y_i Y_j{}^\top = -2 \pm 2\epsilon_1 + 7\epsilon_2$$

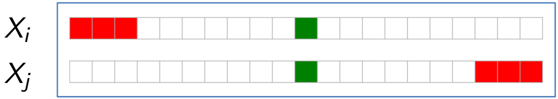
$$Y_i Y_j{}^\top = X_i X_j{}^\top \pm (d\epsilon_1 + d^2\epsilon_2) = X_i X_j{}^\top \pm \tfrac{1}{2} o(1)$$

A Simple Case: $Y = X_1 X_2$

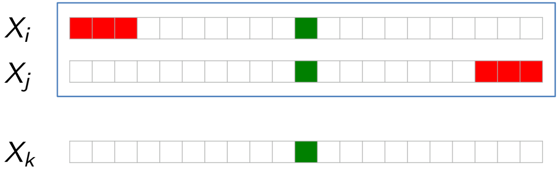
Sparse Decomposition:

$$
\begin{bmatrix}
4 & 6 & -6 & 3 & 2 & 2 & -4 & 2 \\
3 & 2 & -6 & 6 & 0 & 1 & 3 & 2 \\
1 & 2 & 5 & -2 & -6 & -1 & 0 & 3 \\
3 & 2 & -5 & 4 & 0 & 1 & 4 & 0 \\
1 & 1 & 3 & 1 & 2 & -1 & -3 & 3 \\
-3 & 4 & 1 & 7 & 1 & 3 & 6 & -4 \\
-4 & -3 & -1 & 3 & 2 & 1 & -5 & 1 \\
4 & 2 & 3 & -1 & -3 & 2 & 1 & 6
\end{bmatrix}
=
\begin{bmatrix}
0 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\
0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\
+1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\
-1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 \\
0 & 0 & +1 & 0 & -1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
0 & 0 & 0 & +1 & 0 & 0 & -1 & 0 \\
0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\
0 & 0 & 0 & 0 & 0 & +1 & 0 & +1 \\
+1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\
+1 & 0 & +1 & 0 & 0 & 0 & 0 & 0 \\
0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\
0 & 0 & 0 & +1 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & +1 & 0 & -1 & 0
\end{bmatrix}
\cdots
\begin{bmatrix}
0 & 0 & +1 & 0 & 0 & -1 & 0 & 0 \\
+1 & 0 & 0 & 0 & 0 & 0 & +1 & 0 \\
0 & 0 & 0 & 0 & +1 & 0 & +1 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & +1 \\
0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\
+1 & 0 & 0 & 0 & 0 & 0 & +1 & 0 \\
0 & 0 & +1 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & +1 & 0 & -1 & 0 & 0
\end{bmatrix}
$$

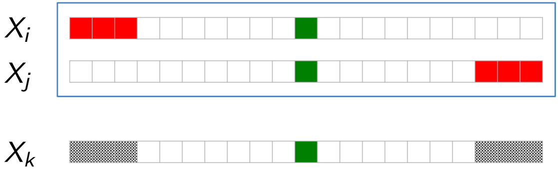
Recovering X from $XX^\top$



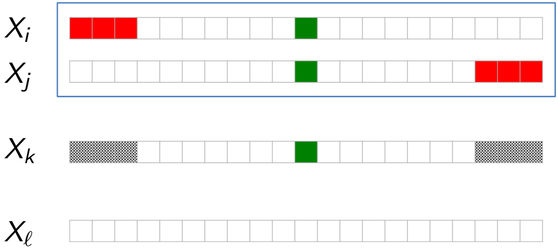
Recovering X from $XX^\top$



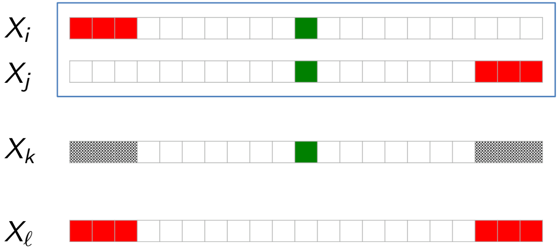
Recovering X from $XX^\top$



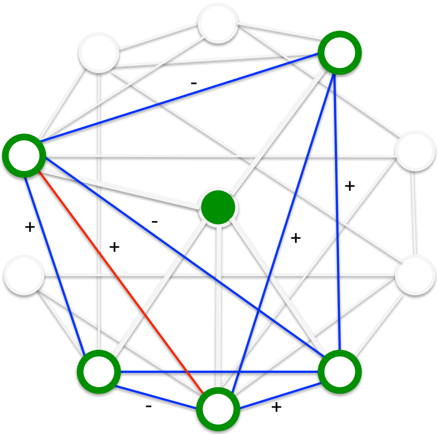
Recovering X from $XX^\top$



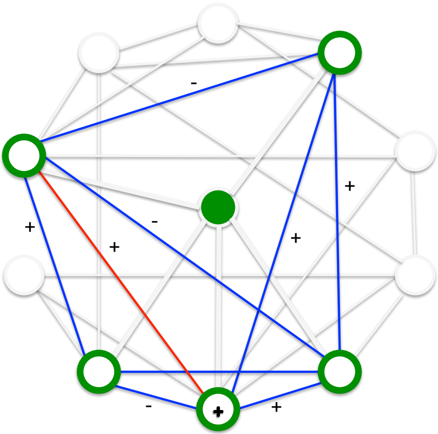
Recovering X from $XX^\top$



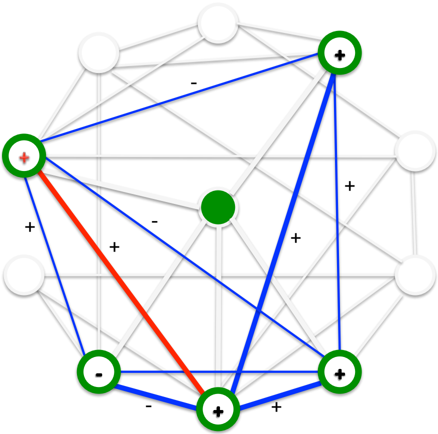
Recovering X from $XX^\top$



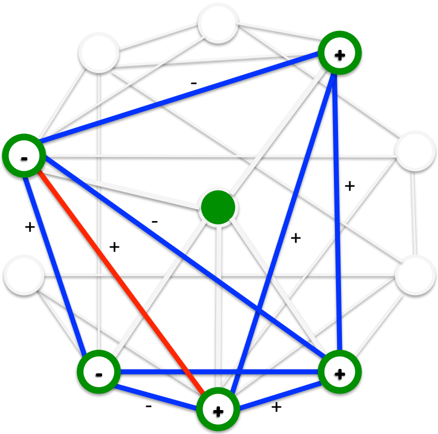
Recovering X from $XX^\top$



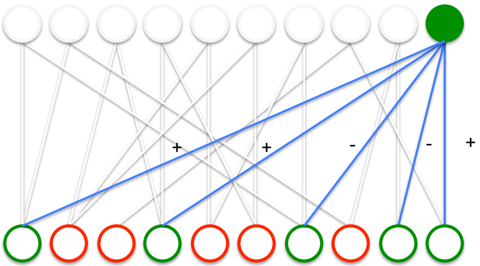
Recovering X from $XX^\top$



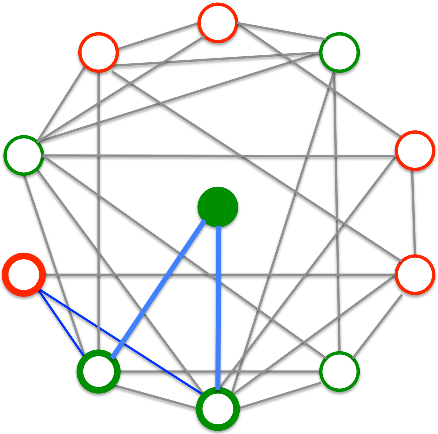
Recovering X from $XX^\top$



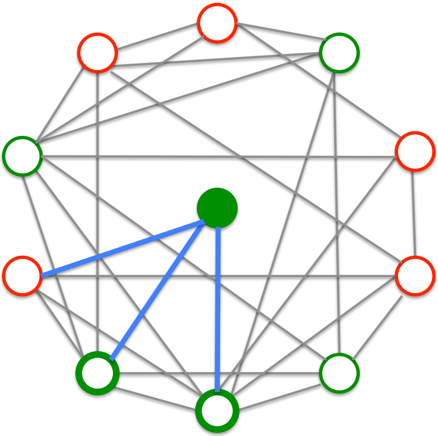
Recovering X from $XX^\top$



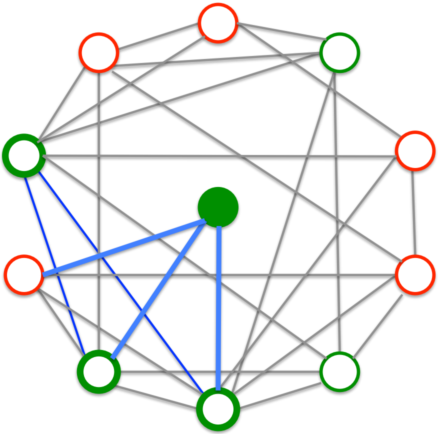
Recovering X from $XX^\top$



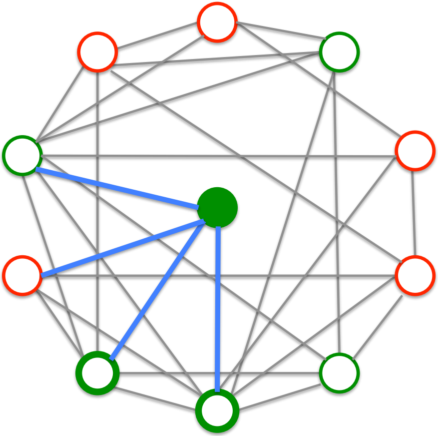
Recovering X from $XX^\top$



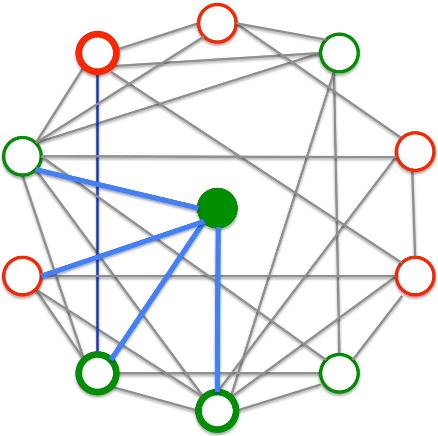
Recovering X from $XX^\top$



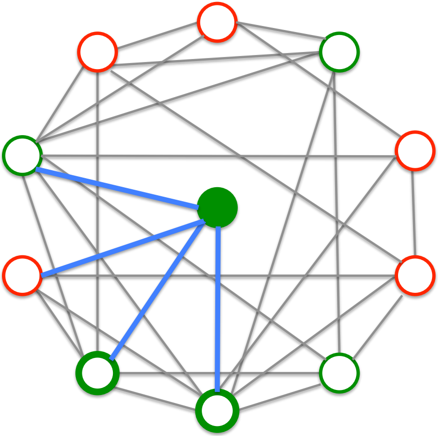
Recovering X from $XX^\top$



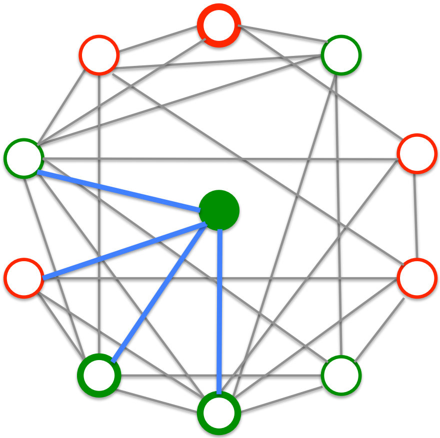
Recovering X from $XX^\top$



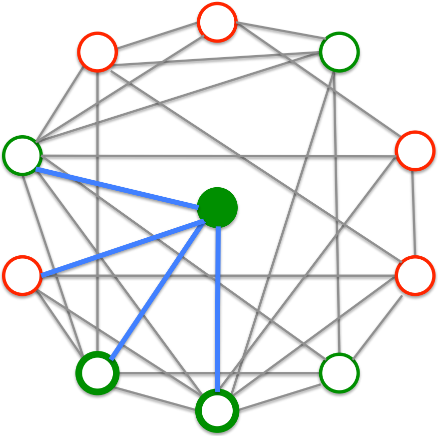
Recovering X from $XX^\top$



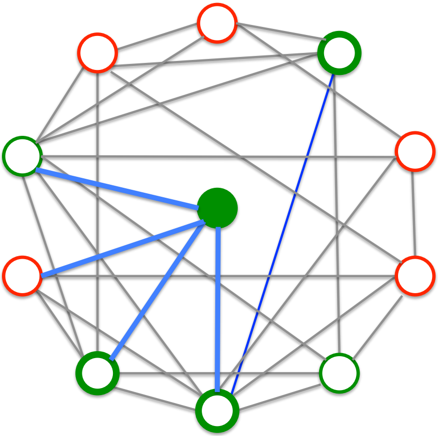
Recovering X from $XX^\top$



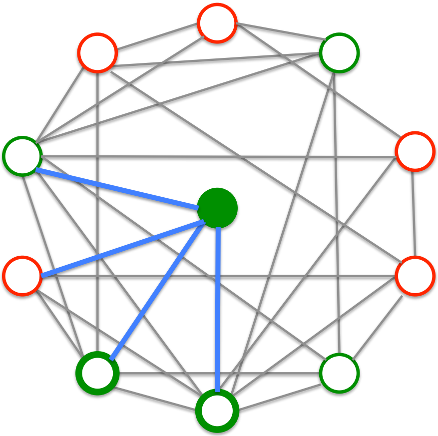
Recovering X from $XX^\top$



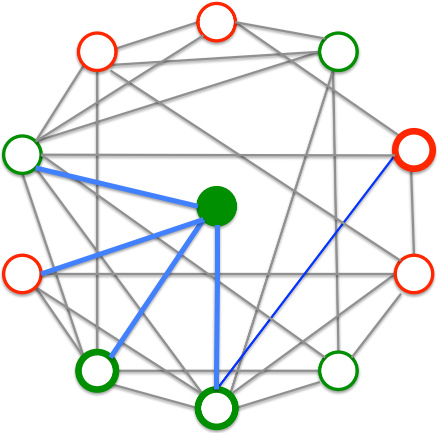
Recovering X from $XX^\top$



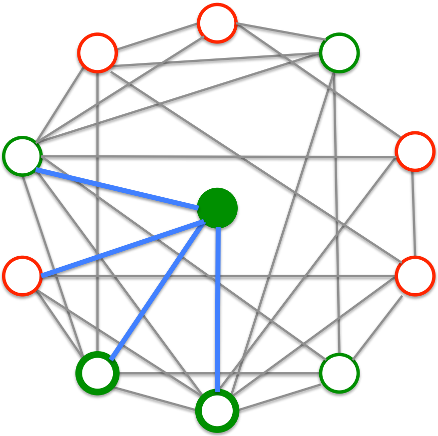
Recovering X from $XX^\top$



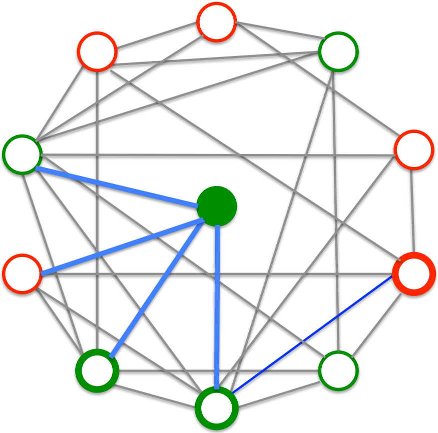
Recovering X from $XX^\top$



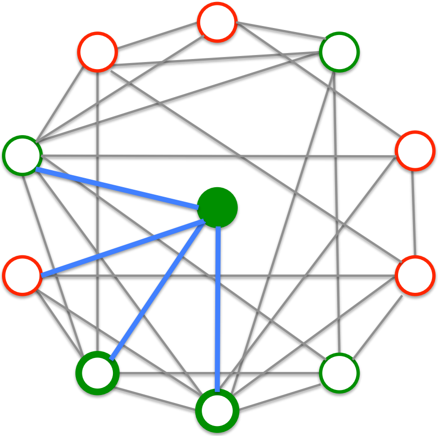
Recovering X from $XX^\top$



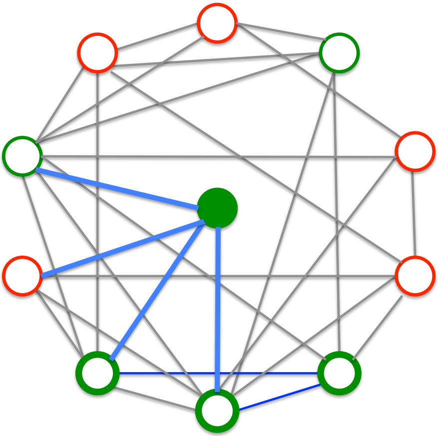
Recovering X from $XX^\top$



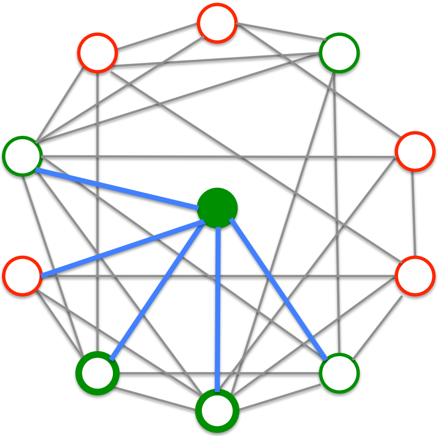
Recovering X from $XX^\top$



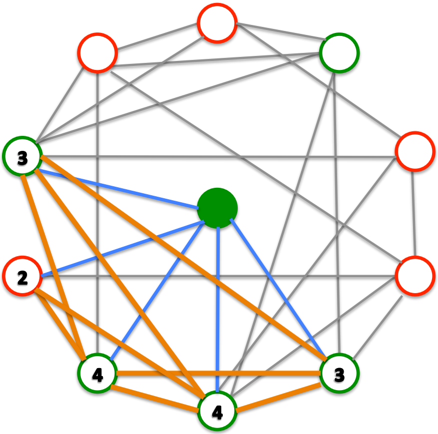
Recovering X from $XX^\top$



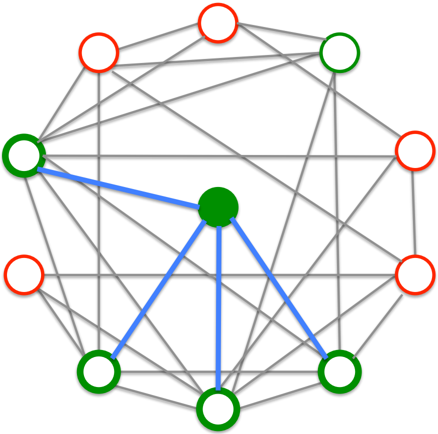
Recovering X from $XX^\top$



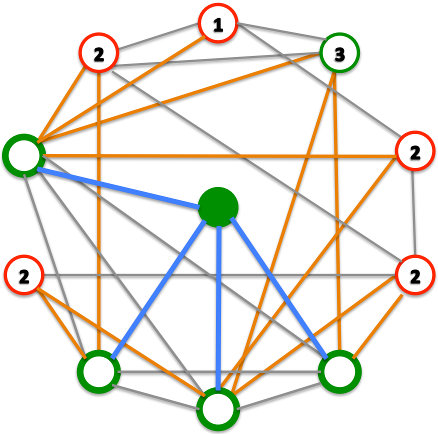
Recovering X from $XX^\top$



Recovering X from $XX^\top$



Recovering X from $XX^\top$



Recovering X from $XX^\top$