Statistical Learning with Sparsity

Graphs, Signal Approximation and Compressed Sensing

Boen Jiang

Fudan University

June 17, 2025

Graphical Model with Sparsity

- Basics of graphical models.
  - Factorization property
  - Markov property
- Gaussian graphical models.
- Graph selection
  - Graphical lasso algorithm
  - Theoretical guarantees for graphical lasso
  - Neighborhood selection algorithm

Undirected Graphs

- Do not focus on DAG.
- Graph $G = (V, E)$ consists of a set of vertices V and a set of edges E.
- We focus exclusively on undirected graphs.
- We can associate a collection of random variables $X = (X_1, X_2, \ldots, X_p)$ with the vertex set $V = \{1, 2 \ldots, p\}$ of some underlying graph.
- Idea: see the structure of the underlying graph as a visual representation of the joint distribution of the random variables.

Factorization Property

- Let $\mathcal{C}$ be the set of all cliques in the graph G.
- For a clique $C \in \mathcal{C}$ a compatibility function ψ_C is a function of the subvector $x_C := (x_s, s \in C)$ taking positive real values.
- Given a collection of compatibility functions we say that a probability distribution P factorizes over G if and only if

$$P(x_1, \ldots, x_p) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C)$$

  where $Z = \sum_{x \in \chi^p} \prod_{c \in \mathcal{C}} \psi_C(x_C)$ ensures that P is properly normalized.
- Such a factorization can lead to savings in storage and computation if the clique sizes are not too large.

Factorization Property

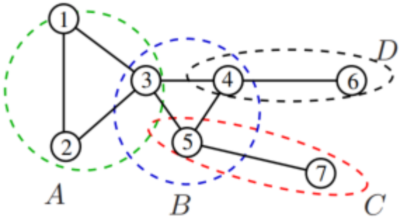


Figure: Source: fig 9.1(a) from SLS.

- P factorizes over this graph if it has the form

$$P(x_1, \ldots, x_7) = \frac{1}{Z}\psi_A(x_1, x_2, x_3)\,\psi_B(x_3, x_4, x_5)\,\psi_D(x_4, x_6)\,\psi_C(x_5, x_7)$$

For some choice of compatility functions $\{\psi_A, \psi_B, \psi_C, \psi_D\}$.

Markov Property

- Let S denote a cut set disconnectong the graph into components A and B.
- We say that a random vector X is Markov with respect to the graph G if

$$X_A \perp\!\!\!\perp X_B \mid X_S \text{ for all cut sets } S \subset V$$

- The same as the stochastic process version of the Markov property, which states that the future is independent of the past given the present.

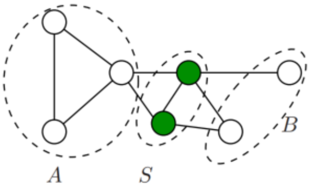


Figure: Source: fig 9.1(b) from SLS.

Hammersley-Clifford Theorem

This is the fundamental theorem of random fields and gives necessary and sufficient conditions under which a strictly positive probability distribution can be represented as a Markov network.

Hammersley-Clifford Theorem

For a strictly positive probability distribution P of a random vector X the two characterizations are equivalent; the distribution of X factorizes according to the graph G if and only if it is Markov with respect to G.

Gaussian Graphical Models

- Given a p dimensional Gaussian distribution with mean vector μ and covariance matrix $\mathbf{\Sigma}$:

$$P_{\mu,\Sigma}(x) = \frac{1}{(2\pi)^{\frac{p}{2}} \det(\mathbf{\Sigma})^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu)^T \mathbf{\Sigma}^{-1}(x-\mu)}.$$

- This can equivalently be formulated as

$$P_{\gamma,\mathbf{\Theta}}(x) = \exp\left\{ \sum_{s=1}^{p} \gamma_s x_s - \frac{1}{2} \sum_{s,t=1}^{p} \boldsymbol{\theta}_{st} x_s x_t - A(\mathbf{\Theta}) \right\}$$

where $\mathbf{\Theta} = \mathbf{\Sigma}^{-1}$ the precision matrix, $\gamma = \mathbf{\Theta}\mu$ and $A(\mathbf{\Theta}) = -\frac{1}{2} \log\det(\mathbf{\Theta}/(2\pi))$.

Gaussian Graphical Models

- This new representation allows us to discuss factorization properties in terms of the sparsity pattern of Θ.
- If X factorizes according to the graph G then for $(s, t) \notin E$ we must have that $\boldsymbol{\theta}_{st} = 0$. (Property of multivariate Gaussian distributions and HC theorem)

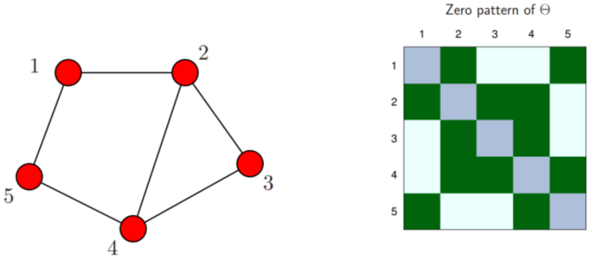


Figure: Source: fig 9.3 from SLS.

Graph Selection

Given a collection of samples X from a graphical model, where the underlying graph structure is unknown. How can we find the correct graph with high probability? $\rightsquigarrow \ell_1$ regularization.

- Suppose $\mathbf{X}$ represents samples from a zero-mean multivariate Gaussian distribution with unknown precision matrix Θ.
- Log-likelihood of this distribution takes the form

$$\mathcal{L}(\boldsymbol{\Theta}, \mathbf{X}) \propto \frac{1}{N} \sum_{i=1}^{N} \log P_{\boldsymbol{\Theta}}(x_i) \propto \log \det \boldsymbol{\Theta} - \operatorname{trace}(\mathbf{S}\boldsymbol{\Theta})$$

where $\mathbf{S} = \frac{1}{N} \sum_{i=1}^{N} x_i x_i^T$ the empirical covariance matrix and

$$\log \det(\boldsymbol{\Theta}) = \begin{cases} \sum_{j=1}^{p} \log(\lambda_j(\boldsymbol{\Theta})) & \text{if } \boldsymbol{\Theta} \succ 0, \lambda_j(\boldsymbol{\Theta}) \text{ is the j-th eval.} \\ -\infty & \text{otherwise.} \end{cases}$$

Graph Selection for Gaussian Graphical Models

- $\log\det$ function is strictly concave, so that if the maximum is achieved it must be unique and defines the MLE $\hat{\Theta}$.
- If we let $N \to \infty$, $\hat{\Theta}$ converges to the true precision matrix.
- But if $N < p$, no maximum likelihood estimator exists and we need to consider suitably constrained or regularized forms.
- If we are seeking Graphical models based on sparse graphs we could consider the following convex optimization problem

$$\widehat{\boldsymbol{\Theta}} \in \arg\max_{\substack{\boldsymbol{\Theta} \succeq \mathbf{0} \\ \rho_0(\boldsymbol{\Theta}) \leq k}} \{\log\det(\boldsymbol{\Theta}) - \mathrm{trace}(\mathbf{S}\ )\}$$

where $\rho_0(\Theta) = \sum_{s \neq t} \mathbb{I}\left[\theta_{st} \neq 0\right]$.

From ℓ_0 to ℓ_1

- Unfortunately, the ℓ_0-based constraint defines a highly nonconvex constraint set, essentially formed as the union over all $\binom{\binom{p}{2}}{k}$ possible subsets of k edges.
- It is natural to consider the convex relaxation obtained by replacing the ℓ_0 constraint with the corresponding ℓ_1-based constraint. Doing so leads to the following convex program

$$\hat{\mathbf{\Theta}} \in \arg\max_{\mathbf{\Theta} \succeq 0} \{\log\det \mathbf{\Theta} - \mathrm{trace}(\mathbf{S}\mathbf{\Theta}) - \lambda \rho_1(\mathbf{\Theta})\}$$

  where $\rho_1(\mathbf{\Theta}) = \sum_{s \neq t} |\boldsymbol{\theta}_{st}|$.
- This is often referred to as the **graphical lasso** problem.

Graphical Lasso Algorithm

- By taking subgradients of the objective function, the subgradient equation corresponding to this problem is given by

$$\mathbf{\Theta}^{-1} - \mathbf{S} - \lambda \mathbf{\Psi} = \mathbf{0}$$

  where $\mathbf{\Psi}$ has diagonal entries $0, \psi_{jk} = \operatorname{sign}\left(\theta_{jk}\right)$ if $\theta_{jk} \neq 0$ and $\psi_{jk} \in [-1, 1]$ if $\theta_{jk} = 0$.
- To solve this problem via blockwise coordinate descent we partition the matrices into (last) one column versus the rest:

$$\mathbf{\Theta} = \left[\begin{array}{cc} \mathbf{\Theta}_{11} & \boldsymbol{\theta}_{12} \\ \boldsymbol{\theta}_{12}^{T} & \theta_{22} \end{array} \right], \mathbf{S} = \left[\begin{array}{cc} \mathbf{S}_{11} & \mathbf{s}_{12} \\ \mathbf{s}_{12}^{T} & s_{22} \end{array} \right].$$

Graphical Lasso Algorithm

- By the formula of inverse of a block matrix, $\mathbf{W} = \boldsymbol{\Theta}^{-1} =$
$$\begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & w_{22} \end{bmatrix} = \begin{bmatrix} \left(\boldsymbol{\Theta}_{11} - \frac{\boldsymbol{\theta}_{12}\boldsymbol{\theta}_{21}}{\theta_{22}}\right)^{-1} & -\mathbf{W}_{11}\frac{\boldsymbol{\theta}_{12}}{\theta_{22}} \\ \cdot & \cdot \end{bmatrix}.$$
- So for the last column (without the last row) of our subgradient equation we get:

$$\mathbf{w}_{12} - \mathbf{s}_{12} + \lambda\boldsymbol{\psi}_{12} = \mathbf{W}_{11}\boldsymbol{\beta} - \mathbf{s}_{12} + \lambda\boldsymbol{\psi}_{12} = 0$$

where $\boldsymbol{\beta} = -\boldsymbol{\theta}_{12}/\theta_{22}$.
- It can be seen (next slide) that this is equivalent to a modified version of the estimating equations for a lasso regression.

Graphical Lasso Algorithm

- Recall that in the usual regression setup with outcome y and predictor matrix $\mathbf{Z}$ the lasso minimizes $\frac{1}{N}\|\mathbf{y}-\mathbf{Z}\beta\|_2^2+\lambda\|\boldsymbol{\beta}\|_1$.
- This has the subgradient equations $\frac{1}{N}\mathbf{Z}^T\mathbf{Z}\boldsymbol{\beta}-\frac{1}{N}\mathbf{Z}^T\mathbf{y}+\lambda\,\mathrm{sign}(\boldsymbol{\beta})=\mathbf{0}$.
- $\mathbf{W}_{11}\boldsymbol{\beta}-\mathbf{s}_{12}+\lambda\boldsymbol{\psi}_{12}=0$.
- Comparing to the last column of our subgradient equation shows that $\frac{1}{N}\mathbf{Z}^T\mathbf{y}$ corresponds to $\mathbf{s}_{12}$ and $\frac{1}{N}\mathbf{Z}^T\mathbf{Z}$ corresponds to $\mathbf{W}_{11}$.
- If W_{11} full-rank, then we want to minimize $\frac{1}{2}\left\|\mathbf{W}_{11}^{\frac{1}{2}}\boldsymbol{\beta}-\mathbf{W}_{11}^{-\frac{1}{2}}\mathbf{s}_{12}\right\|_2^2+\lambda\|\boldsymbol{\beta}\|_1$.

Graphical Lasso Algorithm

Algorithm 9.1 Graphical Lasso.

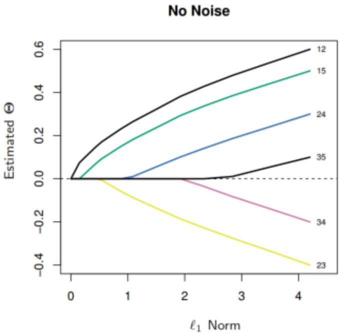
1. Initialize $\mathbf{W} = \mathbf{S}$. Note that the diagonal of $\mathbf{W}$ is unchanged in what follows.
2. Repeat for $j = 1, 2, \ldots p, 1, 2, \ldots p, \ldots$ until convergence:
   (a) Partition the matrix $\mathbf{W}$ into part 1: all but the j^{th} row and column, and part 2: the j^{th} row and column.
   (b) Solve the estimating equations $\mathbf{W}_{11}\boldsymbol{\beta} - \mathbf{s}_{12} + \lambda \cdot \text{sign}(\boldsymbol{\beta}) = 0$ using a cyclical coordinate-descent algorithm for the modified lasso.
   (c) Update $\mathbf{w}_{12} = \mathbf{W}_{11}\hat{\boldsymbol{\beta}}$
3. In the final cycle (for each j) solve for $\hat{\boldsymbol{\theta}}_{12} = -\hat{\boldsymbol{\beta}} \cdot \hat{\theta}_{22}$, with $1/\hat{\theta}_{22} = w_{22} - \mathbf{w}_{12}^T\hat{\boldsymbol{\beta}}$.

Graphical Lasso Algorithm

- If we repeat the algorithm for a range of different values for λ we can plot the estimates for the entires of the precision matrix against $\rho_1(\Theta)$.

- Example: Here the true precision matrix is

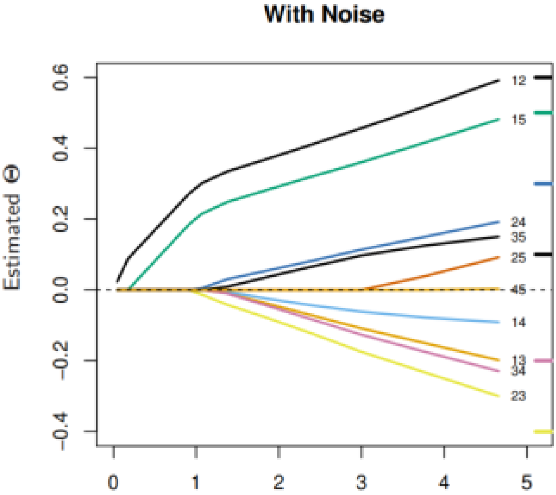
$$\boldsymbol{\Theta} = \begin{bmatrix} 2 & 0.6 & 0 & 0 & 0.5 \\ 0.6 & 2 & -0.4 & 0.3 & 0 \\ 0 & -0.4 & 2 & -0.2 & 0 \\ 0 & 0.3 & -0.2 & 2 & 0 \\ 0.5 & 0 & 0 & 0 & 2 \end{bmatrix}$$



- If we simulate data from the multivariate gaussian with Θ the true values are achieved at the right side of the plot.

Graphical Lasso Algorithm

- However if we add some standard Gaussian noise to each column the true edge set is not recovered for any value of λ.



Theoretical Guarantees for Graphical Lasso

- Plot of the operator norm $\|\hat{\mathbf{\Theta}} - \mathbf{\Theta}\|_2$ versus the sample size N for three different graph sizes where $\lambda_N = 2\sqrt{\frac{\log p}{N}}$ was used as the regularization parameter.
- We see that larger graphs require more samples for a consistent estimation.

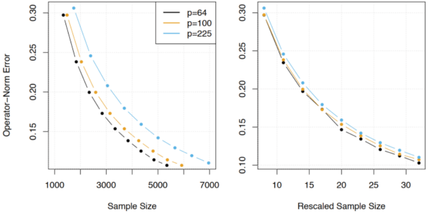
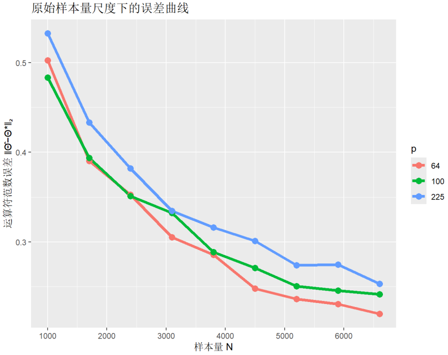
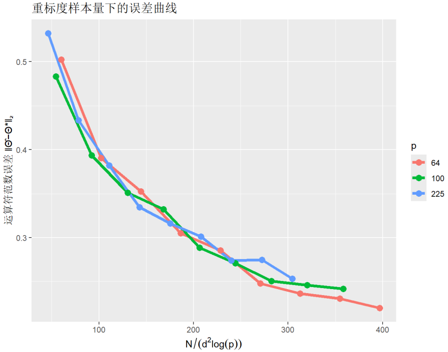


Figure 9.5 *Plots of the operator-norm error $\|\widehat{\Theta} - \Theta^*\|_2$ between the graphical lasso estimate $\widehat{\Theta}$ and the true inverse covariance matrix. Left: plotted versus the raw sample size N, for three different graph sizes $p \in \{64, 100, 225\}$. Note how the curves shift to the right as the graph size p increases, reflecting the fact that larger graphs require more samples for consistent estimation. Right: the same operator-norm error curves plotted versus the rescaled sample size $\frac{N}{d^2 \log p}$ for three different graph sizes $p \in \{64, 100, 225\}$. As predicted by theory, the curves are now quite well-aligned.*

Theoretical Guarantees for Graphical Lasso





This figure illustrates the theoretical guarantees

$$\left\|\widehat{\Theta} - \Theta^*\right\|_2 \lesssim \sqrt{\frac{d^2 \log p}{N}},$$

where d is the maximum degree of the graph, Θ^* is the true precision matrix, and $\widehat{\Theta}$ is the estimated precision matrix from the graphical lasso.

Neighborhood Selection

- High-dimensional Graphs and Variable Selection with the Lasso; (Meinshausen and Buhlmann, 2006)
- It is an alternative method for graph selection that is computationally efficient and consistent for high dimensional graphs.
- For a random vector $X = (X_1, \ldots, X_p)$ consider the conditional distribution of X_s given the random vector $X_{\setminus\{s\}} = (X_1, \ldots, X_{s-1}, X_{s+1}, \ldots, X_p)$.
- By the properties of a graphical model the only relevant variables are those in the neighborhood set $\mathcal{N}(s)$.

$$\mathcal{N}(s) = \{t \in V \mid (s, t) \in E\}$$

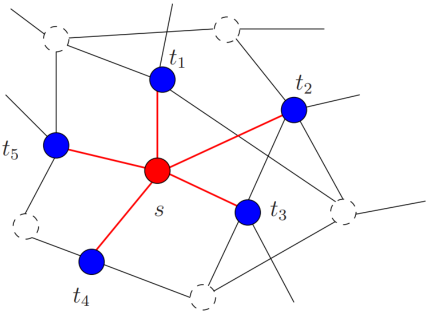


Figure 9.6 *The dark blue vertices form the neighborhood set $\mathcal{N}(s)$ of vertex s (drawn in red); the set $\mathcal{N}^+(s)$ is given by the union $\mathcal{N}(s) \cup \{s\}$. Note that $\mathcal{N}(s)$ is a cut set in the graph that separates $\{s\}$ from $V \backslash \mathcal{N}^+(s)$. Consequently, the variable X_s is conditionally independent of $X_{V \backslash \mathcal{N}^+(s)}$ given the variables $X_{\mathcal{N}(s)}$ in the neighborhood set. This conditional independence implies that the optimal predictor of X_s based on all other variables in the graph depends only on $X_{\mathcal{N}(s)}$.*

Remarks on multivariate Gaussian

- Partitioning $\{X_1, ..., X_p\}$ into (X_1, X_T) where $T = \{2, 3, \ldots, p\}$ and X_T is the vector of all variables except X_1.
- By the distribution of a conditional Gaussian,

$$\mathbb{E}\left[X_1 \mid X_T\right] = \Sigma_{1T}\Sigma_{TT}^{-1}X_T, \quad \mathrm{Var}\left(X_1 \mid X_T\right) = \Sigma_{11} - \Sigma_{1T}\Sigma_{TT}^{-1}\Sigma_{T1}.$$

- By letting

$$\theta = \Sigma_{TT}^{-1}\Sigma_{T1}, \quad W = X_1 - \theta^T X_T$$

- Then for the BLUP $Z = \theta^\top X_T + W$, we have

$$\theta = \Sigma_{TT}^{-1}\Sigma_{T1}, \quad W \sim N_1(0, \Sigma_{11} - \Sigma_{1T}\Sigma_{TT}^{-1}\Sigma_{T1}).$$

- It can be shown that: $\theta_j = 0$ (the entry of coefficient) $\iff$ $j \notin \mathcal{N}(1) \iff \Theta_{1,k} = 0, \quad$ for $k \in T = \{2, 3, \ldots, p\}$.

Remarks on multivariate Gaussian

- By the definition of the precision matrix Θ we have that

$$\begin{pmatrix} \Sigma_{11} & \mathbf{\Sigma}_{1T} \\ \mathbf{\Sigma}_{T1} & \mathbf{\Sigma}_{TT} \end{pmatrix} \begin{pmatrix} \Theta_{11} & \mathbf{\Theta}_{1T} \\ \mathbf{\Theta}_{T1} & \mathbf{\Theta}_{TT} \end{pmatrix} = \begin{pmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{I} \end{pmatrix}$$

- First consider the top right block of the product:

$$\Sigma_{11}\mathbf{\Theta}_{1T} + \mathbf{\Sigma}_{1T}\mathbf{\Theta}_{TT} = \mathbf{0}^T$$

- Bottom right block of the product gives us the following equation:

$$\mathbf{\Sigma}_{T1}\mathbf{\Theta}_{1T} + \mathbf{\Sigma}_{TT}\mathbf{\Theta}_{TT} = \mathbf{I} \quad \Longrightarrow \quad \mathbf{\Theta}_{TT} = \mathbf{\Sigma}_{TT}^{-1}(\mathbf{I} - \mathbf{\Sigma}_{T1}\mathbf{\Theta}_{1T})$$

Remarks on multivariate Gaussian

- Take the first equation and substitute the second equation into it, we get

$$\Sigma_{11}\Theta_{1T} + \mathbf{\Sigma}_{1T}\left[\mathbf{\Sigma}_{TT}^{-1}(\mathbf{I} - \mathbf{\Sigma}_{T1}\Theta_{1T})\right] = \mathbf{0}^{T}$$

- Rearranging gives us the following equation:

$$\underbrace{\left(\Sigma_{11} - \mathbf{\Sigma}_{1T}\mathbf{\Sigma}_{TT}^{-1}\mathbf{\Sigma}_{T1}\right)}_{\frac{1}{\Theta_{11}}}\mathbf{\Theta}_{1T} = -\underbrace{\mathbf{\Sigma}_{1T}\mathbf{\Sigma}_{TT}^{-1}}_{\theta^{\top}}$$

- Then for any $j \in T = \{2, ..., p\}$,

$$\Theta_{1j} = -\Theta_{11}\theta_j$$

- The regression coefficient vector θ satisfies the property that $\mathrm{supp}(\theta) = \mathcal{N}(1)$, i.e., the support of the regression coefficient vector is equal to the neighborhood set of the variable X_1.

Neighborhood Selection for Gaussians

- In the case of a multivarite Gaussian the conditional distribution of X_s given $X_{\setminus\{s\}}$ is (BLUP)

$$X_s = X_{\setminus\{s\}}\beta^s + W_{\setminus\{s\}}$$

  where $W_{\setminus\{s\}}$ corresponds to a prediction error independent of $X_{\setminus\{s\}}$.
- $\mathrm{Var}(W_{\setminus\{s\}}) = \mathrm{Var}(X_s | X_{\setminus\{s\}})$.
- The key property is that the regression vector β^s satisfies $\mathrm{supp}\,(\beta^s) = \mathcal{N}(s)$.
- It is natural to estimate β via the lasso.

Neighborhood Selection

Algorithm 9.2 Neighborhood-based graph selection for Gaussian graphical models.

1. For each vertex $s = 1, 2, \dots, p$:
   (a) Apply the lasso to solve the neighborhood prediction problem:
$$\widehat{\beta}^s \in \arg\min_{\beta^s \in \mathbb{R}^{p-1}} \left\{ \frac{1}{2N} \sum_{i=1}^{N} (x_{is} - x_{i,V\setminus\{s\}}^T \beta^s)^2 + \lambda \|\beta^s\|_1 \right\}. \tag{9.25}$$
   (b) Compute the estimate $\widehat{\mathcal{N}}(s) = \text{supp}(\widehat{\beta}^s)$ of the neighborhood set $\mathcal{N}(s)$.
2. Combine the neighborhood estimates $\{\widehat{\mathcal{N}}(s), s \in V\}$ via the AND or OR rule to form a graph estimate $\widehat{G} = (V, \widehat{E})$.

Ising Models

- Discrete graphical models
  - variables X_s at each vertex $s \in V$ take values in a discrete space $\mathcal{X}_s$. The simplest example is the binary case, say with $\mathcal{X}_s = \{-1, +1\}$.
- Given a graph $G = (V, E)$, one might consider the family of probability distributions

$$\mathbb{P}_\theta\left(x_1, \ldots, x_p\right) = \exp\left\{\sum_{s \in V} \theta_s x_s + \sum_{(s,t) \in E} \theta_{st} x_s x_t - A(\theta)\right\},$$

  parametrized by the vector $\theta \in \mathbb{R}^{|V|+|E|}$.
- Generate: Gibbs sampling.
- With the exception of some special cases, computing the value of $A(\theta)$ is computationally intractable in general.

$$A(\theta) = \log\left[\sum_{x \in \{-1,+1\}^p} \exp\left\{\sum_{s \in V} \theta_s x_s + \sum_{(s,t) \in E} \theta_{st} x_s x_t\right\}\right].$$

Extensions of a Ising Model

- Edge relations (clique $= 2$) $\rightsquigarrow$ larger cliques
  - $$\mathbb{P}_\theta(x) = \exp\left\{\sum_{s\in V}\theta_s x_s + \sum_{(s,t)\in E}\theta_{st}x_s x_t + \sum_{(s,t,u)\in E_3}\theta_{stu}x_s x_t x_u - A(\theta)\right\}.$$

    where E_3 is some subset of vertex triples.
- Binary outcomes $\rightsquigarrow$ multinomial outcomes $X_s \in \{0, 1, 2, \ldots, m-1\}$ for some $m > 2$.
  - $$\mathbb{P}_\theta\left(x_1, \ldots, x_p\right) = \exp\left\{\sum_{s\in V}\sum_{j=1}^{m-1}\theta_{s;j}\mathbb{I}\left[x_s = j\right] + \sum_{(s,t)\in E}\theta_{st}\mathbb{I}\left[x_s = x_t\right] - A(\theta)\right\}$$

    where the indicator function $\mathbb{I}\left[x_s = j\right]$ takes the value 1 when $x_s = j$, and 0 otherwise. When the weight $\theta_{st} > 0$, the edge-based indicator function

Neighborhood Selection for Ising Models

- Note that for Ising model,

$$\mathbb{P}_\theta(x) \propto \exp\left(\theta_s x_s + \sum_{t \neq s} \theta_t x_t + \sum_{t \neq s} \theta_{st} x_s x_t + \sum_{\substack{u<v \\ u,v \neq s}} \theta_{uv} x_u x_v\right).$$

- Thus

$$\mathbb{P}_\theta\left(X_s = x_s \mid X_{V \setminus \{s\}} = x_{\setminus s}\right) \propto \exp\left(\theta_s x_s + \sum_{t \neq s} \theta_{st} x_s x_t\right)$$

Neighborhood Selection for Ising Models

- Thus for $x_s \in \{-1, +1\}$,

$$\mathbb{P}\left(X_s = x_s \mid X_{\setminus s}\right) = \frac{\exp\left(x_s \eta^s\left(x_{\setminus s}\right)\right)}{\exp\left(+\eta^s\left(x_{\setminus s}\right)\right) + \exp\left(-\eta^s\left(x_{\setminus s}\right)\right)},$$

- $\eta^s\left(x_{\setminus s}\right) := \theta_s + \sum_{t \neq s} \theta_{st} x_t$.
- In particular, the log-odds are

$$\log \frac{\mathbb{P}\left(X_s = +1 \mid X_{\setminus s}\right)}{\mathbb{P}\left(X_s = -1 \mid X_{\setminus s}\right)} = 2\eta^s\left(x_{\setminus s}\right) = 2\theta_s + \sum_{t \neq s} 2\theta_{st} x_t$$

Connections to logistic lasso

Hence if we set

$$\beta_{s0} = 2\theta_s, \quad \beta_{st} = 2\theta_{st}$$

and write the observed pairs $\left\{ \left(x_{i,s}, x_{i,\setminus s} \right) \right\}_{i=1}^N$, we see that

$$\mathbb{P}\left(x_{i,s} = 1 \mid x_{i,\setminus s} \right) = \frac{\exp\left(\beta_{s0} + \sum_{t \neq s} \beta_{st} x_{i,t} \right)}{1 + \exp\left(\beta_{s0} + \sum_{t \neq s} \beta_{st} x_{i,t} \right)}$$

which is exactly the logistic-regression model. Consequently, fitting each node-wise neighborhood via $\widehat{\beta}^s =$

$$\arg\min_{\beta^s \in \mathbb{R}^p} \left\{ \frac{1}{N} \sum_{i=1}^N \left[-x_{i,s} \left(\beta_{s0} + x_{i,\setminus s}^T \beta_{s,-0} \right) + \log\left(1 + \exp\left(\beta_{s0} + x_{i,\setminus s}^T \beta_{s,-0} \right) \right) \right] + \lambda \sum_{t \neq s} |\beta_{st}| \right.$$

is precisely lasso-penalized logistic regression, with β_{st} encoding the edges.

Pseudo-likelihood for Mixed models

- Mixed models:
  - continuous and discrete variables
  - e.g., a mixture of Gaussian and Ising models
- Markov random field model:
  - X: p continuous variables,
  - Y: q discrete variables,
- $P_\Omega(x, y) \propto$

$$\exp\left\{\sum_{s=1}^{p}\gamma_s x_s - \frac{1}{2}\sum_{s=1}^{p}\sum_{t=1}^{p}\theta_{st}x_s x_t + \sum_{s=1}^{p}\sum_{j=1}^{q}\rho_{sj}\left[y_j\right]x_s + \sum_{j=1}^{q}\sum_{r=1}^{q}\psi_{jr}\left[y_j, y_r\right]\right\}.$$

  - ρ_{sj} represents an edge between continuous X_s and discrete Y_j.
  - If Y_j has L_j possible states or levels, then ρ_{sj} is a vector of L_j parameters, and $\rho_{sj}\left[y_j\right]$ references the y_j^{th} value.
  - Likewise ψ_{jr} will be an $L_j \times L_r$ matrix representing an edge between discrete Y_j and Y_r, and $\psi_{jr}\left[y_j, y_r\right]$ references the element in row y_j and column y_r.

Pseudo-likelihood for Mixed models

- The pseudo-log-likelihood is defined to be

$$\ell^p(\mathbf{\Omega}; \mathbf{X}, \mathbf{Y}) = \sum_{i=1}^{N} \left[\sum_{s=1}^{p} \log \mathbb{P}\left(x_{is} \mid x_{i\backslash\{s\}}, y_i; \mathbf{\Omega}\right) \sum_{j=1}^{q} \log \mathbb{P}\left(y_{ij} \mid x_i, y_{i\backslash\{j\}}; \mathbf{\Omega}\right) \right]$$

- Continuous: The conditional distribution for each of the p continuous variables is Gaussian, with mean linear in the conditioning variables.

$$\mathbb{P}\left(X_s \mid X_{\backslash\{s\}}, Y; \mathbf{\Omega}\right) = \left(\frac{\theta_{ss}}{2\pi}\right)^{\frac{1}{2}} e^{-\frac{\theta_{ss}}{2}\left(X_s - \frac{\gamma_s + \sum_j \rho_{sj}[Y_j] - \sum_{t \neq s} \theta_{st} X_t}{\theta_{ss}}\right)^2}$$

- Discrete: The conditional distribution for each of the q discrete variables is multinomial, with log-odds linear in the conditioning variables.

$$\mathbb{P}\left(Y_j \mid X, Y_{\backslash\{j\}}; \mathbf{\Omega}\right) = \frac{e^{\psi_{jj}[Y_j, Y_j] + \sum_s \rho_{sj}[Y_j] X_s + \sum_{r \neq j} \psi[Y_j, Y_r]}}{\sum_{\ell=1}^{L_j} e^{\psi_{jj}[\ell, \ell] + \sum_s \rho_{sj}[\ell] X_s + \sum_{r \neq j} \psi[\ell, Y_r]}}$$

Letting $\mathbf{K}_O = \mathbf{\Theta}$, the idea is to write

$$\tilde{\mathbf{K}}_O = \mathbf{\Theta} - \mathbf{L}$$

where $\mathbf{L}$ is assumed to be low rank, with the rank at most the number of hidden variables. We then solve the problem

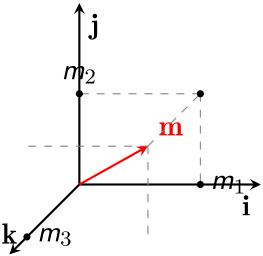
$$\underset{\mathbf{\Theta},\mathbf{L}}{\text{minimize}} \left\{ \text{trace}[\mathbf{S}(\mathbf{\Theta} - \mathbf{L})] - \log[\det(\mathbf{\Theta} - \mathbf{L})] + \lambda \|\mathbf{\Theta}\|_1 + \text{trace}(\mathbf{L}) \right\}$$

over the set $\{\mathbf{\Theta} - \mathbf{L} \succ \mathbf{0}, \mathbf{L} \succeq \mathbf{0}\}$.

Signal Approximation and Compressed Sensing

- Signals and sparse representation
  - orthogonal bases
  - approximation in orthogonal bases
  - reconstruction in overcomplete bases
- Random projection and approximation
  - Johnson-Lindenstrauss approximation
  - compress sensing
- Equivalence between ℓ_0 and ℓ_1 recovery
  - restricted nullspace property and geometric intuition
  - restricted isometry property

Toy example: Cartesian coordinates



- $\mathbf{m} = m_1\mathbf{i} + m_2\mathbf{j} + m_3\mathbf{k}$
- $m_1 = \langle \mathbf{m}, \mathbf{i} \rangle, \quad m_2 = \langle \mathbf{m}, \mathbf{j} \rangle, \quad m_3 = \langle \mathbf{m}, \mathbf{k} \rangle$

Orthogonal Bases

- Signal
  - data such as sea water levels, audio recordings, photographic images, video data, and financial data
  - vectorize signal if it is a matrix or tensor
- represent the signal by a vector $\theta^* \in R^p$.
- Orthogonal bases
  - A basis with finite dimension whose vectors are all unit vectors and orthogonal to each other.
- $\{\psi_j\}_{j=1}^p$ orthonormal basis of $R^p \rightsquigarrow \Psi := [\psi_1 \ \psi_2 \ \dots \ \psi_p]$ is a $p \times p$ matrix with orthonormality condition $\Psi^T \Psi = I_{p \times p}$.

Orthogonal Bases

- Given an orthonormal basis, any signal $\theta^* \in \mathbb{R}^p$ can be expanded in the form

$$\theta^* := \sum_{j=1}^{p} \beta_j^* \psi_j$$

  where the j-th basis coefficient $\beta_j^* := \langle \theta^*, \psi_j \rangle = \sum_{i=1}^{p} \theta_i^* \psi_{ij}$ is obtained by projecting the signal onto the j-th basis vector ψ_j.
- Equivalently, we can write the transformation from signal $\theta^* \in \mathbb{R}^p$ to basis coefficient vector $\beta^* \in \mathbb{R}^p$ as the matrix-vector product $\beta^* = \mathbf{\Psi}^T \theta^*$.

Orthogonal Bases

Example: wavelet transform

Consider the following matrix

$$\boldsymbol{\Psi} := \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{2} & \frac{1}{2} & \frac{-1}{\sqrt{2}} & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{2} & -\frac{1}{2} & 0 & \frac{-1}{\sqrt{2}} \end{bmatrix}$$

The Haar transform matrix is real and orthogonal:

$$\boldsymbol{\Psi} = \boldsymbol{\Psi}^*, \quad \boldsymbol{\Psi}^{-1} = \boldsymbol{\Psi}^T, \quad \text{i.e.} \quad \boldsymbol{\Psi}^T \boldsymbol{\Psi} = \mathbf{I}_{4\times 4}$$

Remarks on Haar Transform

- In general, the Haar transform is defined recursively. Consider the following matrices:

$$\mathbf{H}_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad \mathbf{H}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

- General way to generate the Haar transform:
  $\mathbf{H}_{2N} = \begin{bmatrix} \mathbf{H}_{\mathrm{N}} \otimes [1, 1] \\ \mathbf{I}_{\mathrm{N}} \otimes [1, -1] \end{bmatrix}$ where $\otimes$ means the Kronecker product.
- Then, normalize the rows of $\mathbf{H}_{2N}$ to obtain the orthonormal Haar transform matrix $\mathbf{\Psi}_{2N}^{\top}$.

Illustration of sparsity in time series data

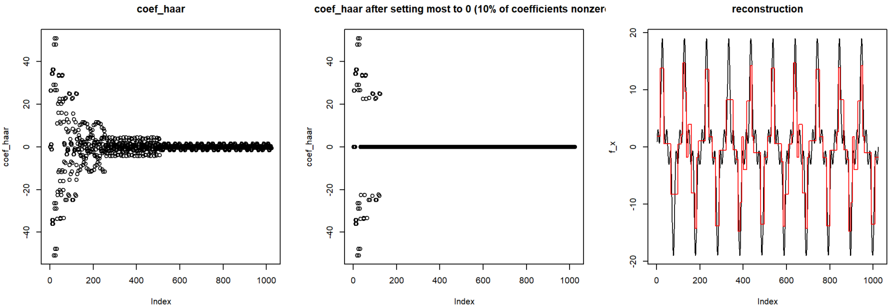


Figure: The signal (right panel, black) is generated as $x_t = 10\sin(20\pi t/N) - 5\sin(60\pi t/N) + 4\sin(100\pi t/N)$ for $t = 1, \ldots, N$ with $N = 1024$. The Haar transform (right panel, red) is applied to the "sparsed" signal. The 972 of 1024 Haar coefficients are set to be zero, hence sparse.

Approximation via Orthogonal Bases

- Goal of signal compression: represent signal $\theta^* \in R^p$ using $k \ll p$ coefficients.
- Use only sparse subset of the orthogonal vectors $\{\psi_j\}_{j=1}^{p}$ for $k \in \{1, \ldots, p\}$: consider reconstruction

$$\mathbf{\Psi}\beta = \sum_{j=1}^{p} \beta_j \psi_j, \quad \text{such that } \|\beta\|_0 := \sum_{j=1}^{p} \mathbb{I}\left[\beta_j \neq 0\right] \leq k$$

- Here, we introduce the ℓ_0 "norm", which counts the number of non-zero entries in the vector β. This is not a true norm, but it is useful for sparsity.
- Doesn't satisfy the positively homogeneous property.

Optimal k-sparse Approximation

- Compute: $\widehat{\beta}^k \in \arg\min_{\beta \in \mathbb{R}^p} \|\theta^* - \boldsymbol{\Psi}\beta\|_2^2 \quad$ such that $\|\beta\|_0 \leq k$.
- Reconstruction: $\theta^k := \sum_{j=1}^{p} \widehat{\beta}_j^k \psi_j$.
  - defines the best least-squares approximation to θ^* based on k terms
  - non-convex and combinatorial problem
- Solve by taking first k coefficients with largest absolute values:
  - we order the vector $\beta^* \in \mathbb{R}^p$ of basis coefficients in terms of their absolute values, thereby defining the order statistics

$$\left|\beta^*_{(1)}\right| \geq \left|\beta^*_{(2)}\right| \geq \ldots \geq \left|\beta^*_{(p)}\right|$$

  - For any given integer $k \in \{1, 2, \ldots, p\}$, it can be shown that the optimal k-term approximation is given by

$$\widehat{\theta}^k := \sum_{j=1}^{k} \beta^*_{(j)} \psi_{\sigma(j)}$$

where $\sigma(j)$ denotes the basis vector associated with the j-th basis.

(a) (b)

Figure 10.3 *Illustration of image compression based on wavelet thresholding. (a) Zoomed portion of the original "Boats" image from Figure 10.2(a). (b) Reconstruction based on retaining 5% of the wavelet coefficients largest in absolute magnitude. Note that the distortion is quite small, and concentrated mainly on the fine-scale features of the image.*

Approximation in Orthogonal Bases: Procedure

1. Compute basis coefficients $\beta_j^* = \langle \theta^*, \psi_j \rangle$ for $j = 1, ..., p$ or in matrix: $\beta^* = \Psi^\top \theta^* \rightsquigarrow O(p^2)$ complexity (matrix-vector products).
2. Sort coefficients in terms of absolute values $\rightsquigarrow O(p \log p)$
3. Extract the first k coefficients
4. Compute the best k-term approximation:

$$\widehat{\theta}^k := \sum_{j=1}^{k} \beta_{(j)}^* \psi_{\sigma(j)}$$

Reconstruction in Overcomplete Bases

- Shortcomings of orthonormal bases: only limited class of signals has sparse representations in ANY orthonormal bases.
  - Certain signals are sparse in one orthonormal basis, but not in another.
- Solution: combine different orthonormal bases $\rightsquigarrow$ use subsets of vectors from both bases simultaneously.

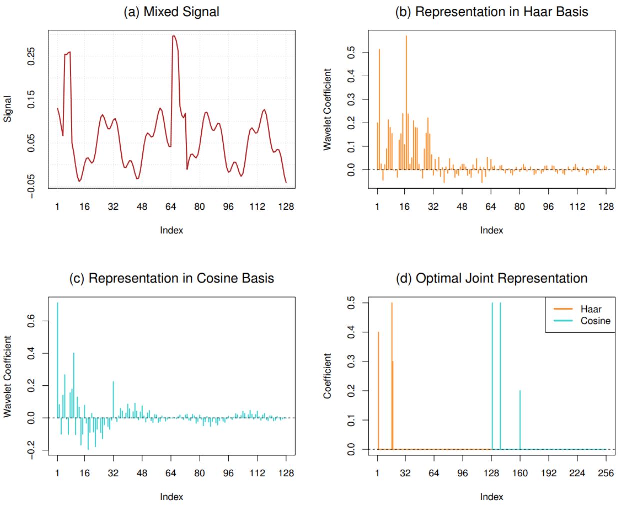


Figure 10.4 *(a) Original signal* $\theta^* \in \mathbb{R}^p$ *with* $p = 128$. *(b) Representation* $\mathbf{\Psi}^T\theta^*$ *in the Haar basis. (c) Representation* $\mathbf{\Phi}^T\theta^*$ *in the discrete cosine basis. (d) Coefficients* $(\widehat{\alpha}, \widehat{\beta}) \in \mathbb{R}^p \times \mathbb{R}^p$ *of the optimally sparse joint representation obtained by solving basis pursuit linear program* (10.11).

Reconstruction in Overcomplete Bases

- two pairs of orthonormal bases: $\{\psi_j\}_{j=1}^p, \{\phi_j\}_{j=1}^p$.
- reconstruction of the form:

$$\underbrace{\sum_{j=1}^{p} \alpha_j \phi_j}_{\mathbf{\Phi}\alpha} + \underbrace{\sum_{j=1}^{p} \beta_j \psi_j}_{\mathbf{\Psi}\beta} \quad \text{such that } \|\alpha\|_0 + \|\beta\|_0 \leq k$$

- optimization problem:

$$\underset{(\alpha,\beta)\in\mathbb{R}^p\times\mathbb{R}^p}{\text{minimize}} \|\theta^* - \mathbf{\Phi}\alpha - \mathbf{\Psi}\beta\|_2^2 \quad \text{such that } \|\alpha\|_0 + \|\beta\|_0 \leq k.$$

- Note that this is a non-convex and combinatorial problem, as it involves the ℓ_0 norm.
- Nonetheless, we can resort to our usual relaxation of the ℓ_0-"norm," and consider the following convex program

$$\underset{(\alpha,\beta)\in\mathbb{R}^p\times\mathbb{R}^p}{\text{minimize}} \|\theta^* - \mathbf{\Phi}\alpha - \mathbf{\Psi}\beta\|_2^2 \quad \text{such that } \|\alpha\|_1 + \|\beta\|_1 \leq R$$

where $R > 0$ is a user-defined radius.

Compressed Sensing with random basis projections

- We discussed approximating a signal by computing its projection onto each of a fixed set of basis functions.
- A random projection of a signal θ^* is a measurement of the form

$$y_i = \langle z_i, \theta^* \rangle = \sum_{j=1}^{p} z_{ij}\theta_j^*$$

where $z_i \in \mathbb{R}^p$ is a random vector.

Johnson-Lindenstrauss Approximation

Johnson-Lindenstrauss Lemma, 1984

Given $0 < \varepsilon < 1$, a set X of N points in $\mathbb{R}^n$, and an integer $k > 8(\ln N)/\varepsilon^2$, , there is a linear map $f \colon \mathbb{R}^n \to \mathbb{R}^k$ such that

$$(1-\varepsilon)\|u-v\|^2 \leq \|f(u)-f(v)\|^2 \leq (1+\varepsilon)\|u-v\|^2$$

for all $u, v \in X$. This map can be found in randomized polynomial time.

Remarks on JL's lemma (Exercise 10.1)

Let's start with the chernoff bound for the chi-square distribution:

- Let
$$Z = \sum_{i=1}^{N} Y_i^2$$
where $Y_i \sim \mathcal{N}(0,1)$ are independent.
-
$$\mathbb{E}\left[e^{\lambda Y_i^2}\right] = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-y^2/2} e^{\lambda y^2} dy = \frac{1}{\sqrt{2\pi}} \int e^{-\frac{1}{2}(1-2\lambda)y^2} dy = \frac{1}{\sqrt{1-2\lambda}}$$
valid for $1-2\lambda > 0$. Hence,
- $\mathbb{E}[\exp(\lambda(Z-d))] = \left[\frac{e^{-\lambda}}{\sqrt{1-2\lambda}}\right]^d$, for all $\lambda < 1/2$.

Remarks on JL's lemma

For any $0 < \lambda < 1/2$, we use the Markov inequality to obtain

$$\mathbb{P}[Z - N \geq tN] \leq \inf_{\lambda > 0} e^{-\lambda t N} \mathbb{E}\left[e^{\lambda(Z-N)}\right]$$

$$= \inf_{\lambda>0} e^{-\lambda t N} \left[\frac{e^{-\lambda}}{\sqrt{1-2\lambda}}\right]^N = \inf_{\lambda>0} \left[\frac{e^{-\lambda(1+t)}}{\sqrt{1-2\lambda}}\right]^N$$

Then we want to find the optimal λ that minimizes the right-hand side. We can do this by taking the logarithm and let

$$f(\lambda) = -\lambda(1+t) - \frac{1}{2}\ln(1-2\lambda)$$

Simply set $\lambda = \frac{t}{8}$ gives the desirable result. Or we can also set the derivative to be zero, this gives

$$\lambda = \frac{t}{2(1+t)}$$

Remarks on JL's lemma

Let $h(t) = -\frac{t}{2} + \frac{1}{2}\ln(1+t)$. We need to show that $h(t) \leq -\frac{t^2}{32}$ for $t \in (0, 1/2)$. Consider the function $k(t) = h(t) + \frac{t^2}{32}$. We have $k(0) = 0$. The derivative is:

$$k'(t) = h'(t) + \frac{2t}{32} = \frac{-t}{2(1+t)} + \frac{t}{16} = t\left(\frac{1}{16} - \frac{1}{2(1+t)}\right).$$

For $t \in (0, 1/2)$, we have $1 < 1 + t < 3/2$, so $1/3 < \frac{1}{2(1+t)} < 1/2$. Since $1/16 < 1/3$, the term in the parenthesis is negative. Thus, $k'(t) < 0$ for $t \in (0, 1/2)$. As $k(0) = 0$ and $k(t)$ is decreasing, $k(t) \leq 0$ on this interval. This proves $h(t) \leq -\frac{t^2}{32}$. Therefore:

$$\mathbb{P}[Z - N \geq tN] \leq e^{Nh(t)} \leq e^{-\frac{Nt^2}{32}}$$

Remarks on JL's lemma (Exercise 10.2)

For an RM $Z \in \mathbb{R}^{N \times p}$, where entries Z_{ij} are i.i.d. $N(0,1)$, define

$$f(u) := \frac{1}{\sqrt{N}} Zu$$

and we have the transformed data $\{f(u_1), f(u_2), ..., f(u_N)\}$. Then the random variable $N\|f(u)\|_2^2$ follows a chi-squared distribution with N degrees of freedom (sum of i.i.d. standard normal).

We want to find a lower bound for the probability of the event $\mathcal{E}(\delta)$, where

$$\mathcal{E}(\delta) := \left\{ \frac{\|f(u_i) - f(u_j)\|_2^2}{\|u_i - u_j\|_2^2} \in [1-\delta, 1+\delta] \quad \text{for all pairs } i \neq j \right\}.$$

Remark

The event in $\mathcal{E}(\delta)$ is the statement of the Johnson-Lindenstrauss lemma.

Remarks on JL's lemma

- We will bound the probability of the complement event, $\mathcal{E}(\delta)^c$, using the union bound. Let $\mathcal{E}_{ij}$ be the event that the distance is preserved for a single pair (i, j). Then $\mathcal{E}(\delta)^c = \bigcup_{i<j} \mathcal{E}_{ij}^c$.
- By the union bound, $\mathbb{P}\left[\mathcal{E}(\delta)^c\right] \leq \sum_{i<j} \mathbb{P}\left[\mathcal{E}_{ij}^c\right]$. The number of pairs is $\binom{M}{2} = \frac{M(M-1)}{2}$.
- Let's analyze the probability for a single pair, $\mathbb{P}\left[\mathcal{E}_{ij}^c\right]$. Let $u = \frac{u_i - u_j}{\|u_i - u_j\|_2}$. This is a unit vector. The ratio of squared norms can be written as:

$$\frac{\|f(u_i) - f(u_j)\|_2^2}{\|u_i - u_j\|_2^2} = \frac{\|f(u_i - u_j)\|_2^2}{\|u_i - u_j\|_2^2} = \left\| f\left(\frac{u_i - u_j}{\|u_i - u_j\|_2} \right) \right\|_2^2 = \|f(u)\|_2^2$$

Remarks on JL's lemma

- We know that $Z := N\|F(u)\|_2^2$ follows a χ_N^2 distribution. Thus, $\|F(u)\|_2^2 = Z/N$.
- Then

$$\mathcal{E}_{ij}^c = \left\{ \frac{Z}{N} \notin [1-\delta, 1+\delta] \right\} = \left\{ \left| \frac{Z}{N} - 1 \right| > \delta \right\} = \{|Z-N| > \delta N\}$$

- By the chernoff bound for the chi-square distribution, we have:

$$\mathbb{P}\left[\mathcal{E}_{ij}^c\right] = \mathbb{P}[|Z-N| \geq \delta N] \leq 2e^{-\frac{N\delta^2}{32}}$$

- Union bound:

$$\mathbb{P}\left[\mathcal{E}(\delta)^c\right] \leq \sum_{i<j} \mathbb{P}\left[\mathcal{E}_{ij}^c\right] = \binom{M}{2} \cdot 2e^{-\frac{N\delta^2}{32}} = M(M-1)e^{-\frac{N\delta^2}{32}}$$

Remarks on JL's lemma

- We can use the looser inequality $M(M-1) < M^2$:

$$\mathbb{P}\left[\mathcal{E}(\delta)^c\right] < M^2 e^{-\frac{N\delta^2}{32}}$$

- Thus

$$\Pr[\mathcal{E}(\delta)] = 1 - \Pr\left[\mathcal{E}(\delta)^c\right] \geq 1 - M^2 e^{-N\delta^2/32}$$

- $\Pr[\mathcal{E}(\delta)] = 1$ when $N > \dfrac{64}{\delta^2} \log M$.

Johnson-Lindenstrauss Approximation

- establish the existence of distance-preserving dimension reduction projection
  - $\|f(u_i) - f(u_j)\|_2 \approx \|u_i - u_j\|_2$
- provide an explicit bound on the dimension required for approximate distance preserving
  - $N > \frac{c}{\delta^2} \log M$
- provide an explicit construction of the random projection
  - $f(u) = \frac{1}{\sqrt{N}} Zu$

Compressed Sensing

- Motivation: In previous orthonormal bases algorithm we discard most β_j^* 's, so do we really need to calculate them all?
  - select only the k largest coefficients $\beta^*_{(1)}, \ldots, \beta^*_{(k)}$ and discard the rest.
- Oracle technique: if we know which subset of k coefficients will be retained for sparse approximation $\rightsquigarrow$ only need to compute this subset of basis coefficients
- Compressed sensing
  - Instead of precomputing all coefficients $\beta^* = \Psi^T \theta^*$, we compute N random projections $y = Z_{N \times p}\theta$ (with $N \ll p$), i.e., $y_i = \langle z_i, \theta \rangle, i = 1, ..., N, z_i \in \mathbb{R}^p$.
  - Z: design matrix
  - Mimics behaviour of the oracle technique with only little computational overhead.

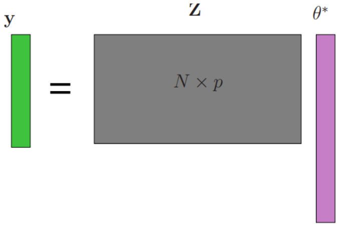
Compressed Sensing v.s. regression

- Given: $y \in \mathbb{R}^N$: Vector of random projections of signal θ^*
- Given: $Z \in \mathbb{R}^{N \times p}$: Design matrix used to compute random projections
- Goal: Recover signal $\theta^* \in \mathbb{R}^p$
- Problem: $y = Z\theta$ is highly underdetermined as $N \ll p$

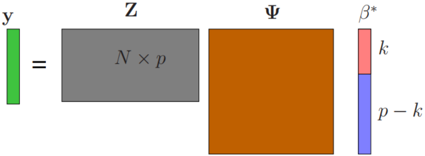
Example

$$y = x_1 + x_2 \text{ if } y = 1 :$$
$$x_1 = 1 \text{ and } x_2 = 0$$
$$x_1 = 0.5 \text{ and } x_2 = 0.5$$
etc...

Compressed Sensing



(a)



(b)

$$\underset{\theta \in \mathbb{R}^p}{\text{minimize}} \left\|\Psi^T \theta\right\|_0 \ \text{such that } y = Z\theta$$

$$\downarrow \quad l_1 \text{ -relaxation}$$

$$\underset{\theta \in \mathbb{R}^p}{\text{minimize}} \left\|\Psi^T \theta\right\|_1 \ \text{such that } y = Z\theta$$

Equivalently we can write:

$$\underset{\beta \in R^p}{\text{minimize}} \|\beta\|_1 \ \text{such that } y = \widetilde{Z}\beta$$

where $\widetilde{Z} = Z\Psi$ $(\beta = \Psi^\top \theta)$.

Compressed Sensing

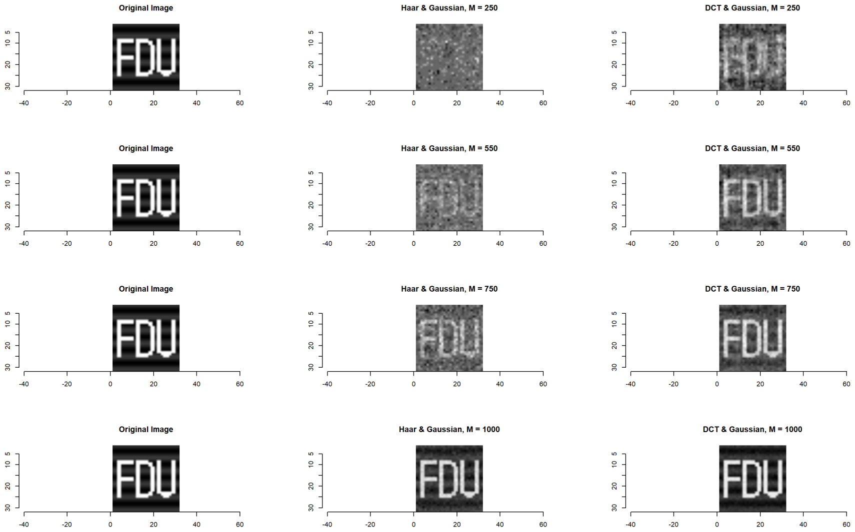
The method of compressed sensing operates as follows:

1. For given sample size N, compute random projections $y_i = \langle z_i, \theta^* \rangle$ (Or ideally measure the random projections y instead of full signal θ^*)
2. Estimate θ^* by solving linear program: $\underset{\theta \in \mathbb{R}^p}{\text{minimize}} \left\| \Psi^T \theta \right\|_1$ such that $y = Z\theta$, to obtain $\hat{\theta}$.

Other random sensing matrices $Z \in \mathbb{R}^{N \times p}$

- Gaussian random matrix: $z_{ij} \in N(0, 1/N)$
- Bernoulli random matrix: $z_{ij} \in \{-1/\sqrt{N}, 1/\sqrt{N}\}$ with probability p.

Cases study: image reconstruction



Equivalence between ℓ_0 and ℓ_1 Recovery

Given $y \in \mathbb{R}^N$ and $X \in \mathbb{R}^{N \times p}$:

- ℓ_0 problem: $\underset{\beta \in \mathbb{R}^p}{\text{minimize}} \|\beta\|_0 \quad$ such that $\mathbf{X}\beta = \mathbf{y}$,
- ℓ_1 problem: $\underset{\beta \in \mathbb{R}^p}{\text{minimize}} \|\beta\|_1 \quad$ such that $\mathbf{X}\beta = \mathbf{y}$.

Example: compressed sensing: $X = \tilde{Z} = Z\Psi$.

Restricted Nullspace Property

An $N \times p$ matrix X satisfies the restricted nullspace property for a set $S \subseteq \{1, \ldots, p\}$ if

$$\|\beta_S\|_1 < \|\beta_{S^c}\|_1 \text{ for all } \beta \in \ker(\mathrm{X}) \backslash \{0\}$$

It is said to satisfy the null space property of order k if it satisfies the Nullspace property for any set S with $\mathrm{card}(S) \leq k$.

RN(S)

For a given subset $S \subseteq \{1, 2, \ldots, p\}$, it is stated in terms of the set

$$\mathbb{C}(S) := \{\beta \in \mathbb{R}^p \mid \|\beta_{S^c}\|_1 \leq \|\beta_S\|_1\}.$$

For a given subset $S \subseteq \{1, 2, \ldots, p\}$, we say that the design matrix $\mathbf{X} \in \mathbb{R}^{N \times p}$ satisfies the restricted nullspace property over S, denoted by $\mathrm{RN}(S)$, if

$$\ker(\mathbf{X}) \cap \mathbb{C}(S) = \{0\}$$

Equivalence between ℓ_0 and ℓ_1 Recovery

Theorem 10.1

If a given matrix $\mathrm{X} \in \mathbb{R}^{N \times p}$ satisfies the null space property for a set S (RN(S)), every vector $\beta^* \in \mathbb{R}^p$ supported on this set S is the unique solution of the ℓ_1-problem with $y = \mathrm{X}\beta^*$.

- ℓ_1 problem: $\underset{\beta \in \mathbb{R}^p}{\text{minimize}} \|\beta\|_1 \quad$ such that $\mathbf{X}\beta = \mathbf{y}$.
- First, suppose that $\mathbf{X}$ satisfies the $\mathrm{RN}(S)$ property.
- Let $\widehat{\beta} \in \mathbb{R}^p$ be any optimal solution to the basis pursuit LP (ℓ_1 optimization problem), and define the error vector $\Delta := \widehat{\beta} - \beta^*$.
- Our goal is to show that $\Delta = 0$.
- It suffices to show that $\Delta \in \ker(\mathbf{X}) \cap \mathbb{C}(S)$.

Proof of Theorem 10.1

- On the one hand, since β^* and $\widehat{\beta}$ are optimal (and hence feasible) solutions to the ℓ_0 and ℓ_1 problems, respectively, we are guaranteed that $\mathbf{X}\beta^* = \mathbf{y} = \mathbf{X}\widehat{\beta}$, showing that $\mathbf{X}\Delta = 0$, namely, $\Delta \in \ker(\mathbf{X})$.
- On the other hand, since β^* is also feasible for the ℓ_1-based problem, the optimality of $\widehat{\beta}$ implies that $\|\widehat{\beta}\|_1 \leq \|\beta^*\|_1 = \|\beta^*_S\|_1$ (β^* is supported on S, $\beta^*_{S^c} = 0$). Note that for any vector v,

$$\|v\|_1 = \sum_{i=1}^{p} |v_i| = \sum_{i \in S} |v_i| + \sum_{i \in S^c} |v_i| = \|v_S\|_1 + \|v_{S^c}\|_1 .$$

Writing $\widehat{\beta} = \beta^* + \Delta$, we have

$$\begin{aligned} \|\beta^*_S\|_1 \geq \|\widehat{\beta}\|_1 &= \|\beta^*_S + \Delta_S\|_1 + \|\Delta_{S^c}\|_1 \\ &\geq \|\beta^*_S\|_1 - \|\Delta_S\|_1 + \|\Delta_{S^c}\|_1 \end{aligned}$$

where the final bound follows by triangle inequality. Rearranging terms, we find that $\Delta \in \mathbb{C}(S)$.

Pairwise incoherence

$$\nu(\mathbf{X}) := \max_{\substack{j,j'=1,2,\ldots,p \\ j\neq j'}} \frac{\left|\left\langle \mathbf{x}_j, \mathbf{x}_{j'} \right\rangle\right|}{\left\|\mathbf{x}_j\right\|_2 \left\|\mathbf{x}_{j'}\right\|_2}.$$

- For centered $\mathbf{x}_j$ this is the maximal absolute pairwise correlation.

Pairwise Incoherence $\implies$ RN(S) (Proposition 10.1)

Suppose that for some integer $k \in \{1, 2, \ldots, p\}$, the pairwise incoherence satisfies the bound $\nu(\mathbf{X}) < \frac{1}{3k}$. Then $\mathbf{X}$ satisfies the uniform RN property of order k, and hence, the basis pursuit LP is exact for all vectors with support at most k.

- Easy to verify: time complexity $O(Np^2)$.

Proof of Proposition 10.1

- WLOG, $\|\mathbf{x}_j\|_2 = 1$ for all $j = 1, 2, \ldots, p$. To simplify notation, let us assume an incoherence condition of the form $\nu(\mathbf{X}) < \frac{\delta}{k}$ for some $\delta > 0$.
- Want to show: for an arbitrary subset S of cardinality k, suppose that $\beta \in \mathbb{C}(S) \backslash \{0\}$, then $\|\mathbf{X}\beta\|_2^2 > 0$, which means $\beta \notin \mathrm{Ker}\, X$.
- To begin with,

$$\|\mathbf{X}\beta\|_2^2 \geq \|\mathbf{X}_S\beta_S\|_2^2 + 2\beta_S^T\mathbf{X}_S^T\mathbf{X}_{S^c}\beta_{S^c}$$

- Then let's dive into the cross term.

Proof of Proposition 10.1

- By definition:

$$\beta_S^T \mathrm{X}_S^T \mathrm{X}_{S^c} \beta_{S^c} = \sum_{i \in S} \sum_{j \in S^c} \beta_i \beta_j \langle x_i, x_j \rangle . \qquad (1)$$

- Then by absolute value inequality, we have

$$2 \left| \beta_S^T \mathbf{X}_S^T \mathbf{X}_{S^c} \beta_{S^c} \right| \leq 2 \sum_{i \in S} \sum_{j \in S^c} |\beta_i| \cdot |\beta_j| \cdot |\langle \mathbf{x}_i, \mathbf{x}_j \rangle|$$

- Note that

$$\sum_{i \in S} \sum_{j \in S^c} |\beta_i| \, |\beta_j| = \left(\sum_{i \in S} |\beta_i| \right) \left(\sum_{j \in S^c} |\beta_j| \right) = \|\beta_S\|_1 \, \|\beta_{S^c}\|_1 .$$

Proof of Proposition 10.1

- Then by the definition of pairwise incoherence, we have $|\langle \mathbf{x}_i, \mathbf{x}_j \rangle| \le \nu(X) \le \frac{\delta}{k}, \forall i \neq j$.
- By Cauchy-Schwarz inequality, we have

$$\|\beta_S\|_1^2 \le |S| \, \|\beta_S\|_2^2 \le k \, \|\beta_S\|_2^2$$

- Together, we obtain

$$\|\mathbf{X}\beta\|_2^2 \ge \|\mathbf{X}_S\beta_S\|_2^2 - 2\delta \, \|\beta_S\|_2^2 = \beta_S^\top [I + (X_S^\top X_S - I)]\beta_S - 2\delta \, \|\beta_S\|_2^2 .$$

- Note that

$$\left\|\mathbf{X}_S^T\mathbf{X}_S - \mathbf{I}_{k\times k}\right\|_{\text{op}} \le \max_{i\in S} \sum_{j\in S\setminus\{i\}} |\langle x_i, x_j \rangle| \le k\frac{\delta}{k} = \delta$$

- Then $\|\mathbf{X}\beta\|_2^2 > (1 - 3\delta) \, \|\beta_S\|_2^2$, let $\delta = 1/3$ completes the proof.

Restricted Isometry Property

RIP

For tolerance $\delta \in (0,1)$ and $2k \in \{1,\dots,\mathrm{p}\}$ we say that $\mathrm{RIP}(2k,\delta)$ holds if $\frac{\|\mathrm{X}_S u\|_2^2}{\|u\|_2^2} \in [1-\delta, 1+\delta]$ for all $u \in \mathbb{R}^k \backslash \{0\}$ for all subsets $\mathrm{S} \subset \{1,\dots,\mathrm{p}\}$ of cardinality $2k$.

Intuition:

- RIP holds if X_S changes length of vectors very little, eigenvalues close to 1.
- Every set of columns of size at most $2k$ approximately behaves like orthonormal system.

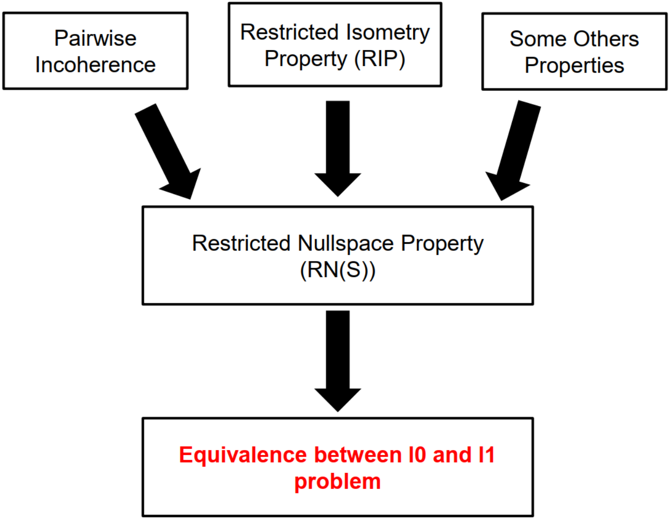
$$\left\|\mathbf{X}_S^T\mathbf{X}_S - \mathbf{I}_{k\times k}\right\|_{\mathrm{op}} = \sup_{\|u\|_2=1} \left|u^T\left(\mathbf{X}_S^T\mathbf{X}_S - \mathbf{I}_{k\times k}\right)u\right| = \sup_{\|u\|_2=1}\left|\|\mathbf{X}_S u\|_2^2 - 1\right|.$$

RIP implies RN(S)

Proposition 10.2

If $\mathrm{RIP}(2k, \delta)$ holds with $\delta < 1/3$, then the uniform RN property of order k holds, and hence the ℓ_1 relaxation is exact for all vectors supported on at most k elements.

- Advantage: RIP constant δ does not depend on k.
- Problem: constraint on a huge number of submatrices, $\binom{p}{2k}$ in total.
- Various choices of random projection matrix X satisfy RIP with high probability as long as $N \gtrsim k \log \dfrac{ep}{k}$.


Pairwise Incoherence
Restricted Isometry Property (RIP)
Some Others Properties
Restricted Nullspace Property (RN(S))
Equivalence between l0 and l1 problem

References

- ISLR, SLS and ESL,
- 36-708 Statistical Methods for Machine Learning, CMU
- Time-Frequency Analysis and Wavelet Transform, TFW_Write6, NTHU
- Sparse learning slides, ETHZ

Questions or comments?