Case Study 3: fMRI Prediction

Coping with Large Covariances: Graphical Models, Graphical LASSO

Machine Learning for Big Data
CSE547/STAT548, University of Washington
Sham Kakade
May 10, 2016
So far, we looked at univariate multiple regression

If one has a multivariate response $y^i \in \mathbb{R}^d$
- Assuming independence between dimensions
Multivariate Normal Models

- If one has a multivariate response $y^i \in \mathbb{R}^d$
  - Assuming correlation between the output dimensions

- Assume linear (or other mean regression) is removed and focus on the correlation structure

- Matrix valued parameter!
Low-Rank Approximations

• In pictures...

\[ \Sigma = \Lambda \Lambda' + \Sigma_0 \]

\[ \Sigma_0 = \text{diag}(\sigma_1^2, \ldots, \sigma_d^2) \]

• Number of parameters:
Latent Factor Models

• Original multivariate regression
\[ y^i = B^T x^i + \epsilon^i, \quad \epsilon^i \sim N(0, \Sigma) \]

• Latent factor model assumption: \[ \Sigma = \Lambda \Lambda' + \Sigma_0 \]
• Low-rank approximation arises from a latent factor model

• Proof:
Lower-dim Embeddings

Sharing information in \textit{low-dim subspace}
Sparsity Assumptions

- What if we assume $\Sigma$ is sparse?

- More often, we can reasonably make statements about conditional independence.
Motivations for considering “information form” of multivariate normal

- Easier to read off conditional densities
- Has log-linear form in terms of “information parameters”
Assume a model with

and divide the dimensions into two sets

Then,
Let $A = \{s, t\}$

$$p(y_A \mid y_{\bar{A}}) = \mathcal{N}^{-1}(\eta_A - \Omega_{A\bar{A}} y_{\bar{A}}, \Omega_{AA})$$

Therefore,
Connection with Graphical Models

- Undirected graphical model or Markov random field (MRF)

\[
p(y | \eta, \Omega) \propto \prod_t \psi_t(y_t) \prod_{(s,t) \in E} \psi_{st}(y_s, y_t)
\]

\[
\psi_t(y_t) \propto e^{\eta_t y_t}
\]

\[
\psi_{st}(y_s, y_t) \propto e^{-\frac{1}{2} y_s \Omega_{st} y_t}
\]

©Sham Kakade 2016
Sparse Precision vs. Covariance

- For a sparse precision matrix, the covariance need not be

```
>> Omega
Omega =
  5.0000  0   1.3731   0   0.7988   0.9681   0  -0.8558   0   0
  0   3.3483  1.5783  -1.6742   0  -0.5654   0  -1.1826   0  0
-1.3731  1.5783  2.9305   0.9951   0   0  -0.6900  -1.2806   0.7026
  0  -1.6742  0.9951  6.0197   0   0  -0.8074   0  -0.5798
0.7988   0   0   0  4.0541   0  0.8074   0   0  0
0.9681  -0.5654   0   0   0   0   5.0000   0   0  0.1253
0  -0.6900   0   0   0   0   5.6526   0.8674  0  0
-0.8558  -1.1826  -1.2806   0.8074   0  0   0.8674  5.0000  -1.5433
0  0.7026   0   0   0   0  -1.1253   0  -1.129
0  0  -0.5798   0   0   0  -1.1253   0  -1.129
0  0   0   0   0   0  5.0000

>> Sigma = inv(Omega)
Sigma =
  0.3730 -0.2560  0.4290 -0.1448 -0.0947 -0.1125  0.0360  0.1056 -0.0505 -0.0280
-0.2560  0.9071 -0.7903  0.3906  0.453  0.1866 -0.1004  0.0258  0.1533  0.0754
0.4290 -0.7903  1.2528 -0.4354 -0.1147 -0.2103  0.1297  0.1514 -0.1682 -0.0879
-0.1448  0.3906 -0.4354  0.3523  0.0319  0.0894 -0.0506 -0.0167  0.0764  0.0578
-0.0947  0.453  0.1866 -0.2103  0.0229  0.2609  0.0251 -0.0026  0.0026  0.0311
-0.1125  0.1866 -0.2103  0.0894  0.0229  0.2609 -0.0251  0.0026 -0.0302  0.0251
0.0360 -0.1004  0.1297 -0.0506 -0.0016 -0.251  0.1970 -0.0276  0.0305  0.0630  0.0121
0.1066  0.0258  0.1514 -0.0167 -0.0808 -0.0035 -0.0276  0.3005  0.0630  0.0121
-0.0505  0.1533 -0.1682  0.0784 -0.0026 -0.0302  0.0630  0.2357  0.0613  0.2204
-0.0280  0.0794 -0.0879  0.0578  0.0031  0.0282 -0.0126  0.0121  0.0613  0.2204
```
ML Estimation for Given Graph

- Assume a known graph $G = \{V,E\}$
- Rewrite log likelihood:

$$
\frac{1}{\sqrt{2\pi|\Sigma|}} e^{-\frac{1}{2} (y-\mu)^T \Sigma^{-1} (y-\mu)}
$$
ML Estimation for Given Graph

\[ L(\Omega) = \log |\Omega| - \text{tr}(S\Omega) \]

- Take gradient:

- Many approaches to solving:
  - Barrier method – add penalty discouraging \( \Omega \) from leaving the positive definite cone (Dahl et al. 2008)
  - Coordinate descent method (cf., Hastie et al. 2009)
  - ...

©Sham Kakade 2016
Can show that the optimal solution satisfies

Example:

\[
G = \begin{pmatrix}
0 & 1 & 0 & 1 \\
1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
1 & 0 & 1 & 0
\end{pmatrix}, \quad
S = \begin{pmatrix}
10 & 1 & 5 & 4 \\
1 & 10 & 2 & 6 \\
5 & 2 & 10 & 3 \\
4 & 6 & 3 & 10
\end{pmatrix}
\]

\[
\Omega = \begin{pmatrix}
0 & 0 \\
0 & 0
\end{pmatrix}, \quad
\Sigma = \begin{pmatrix}
10 & 1 & 4 \\
1 & 10 & 2 \\
2 & 10 & 3 \\
4 & 3 & 10
\end{pmatrix}
\]
To learn the structure of the Gaussian graphical model, we want to trade off fit and sparsity

- Measure of fit:

- Encouraging sparsity:

Overall objective = “graphical LASSO” or “Glasso”
Solving the Graphical LASSO

- Objective is convex, but non-smooth as in LASSO
- Also, positive definite constraint!

- There are many approaches to optimizing the objective
  - Most common = coordinate descent akin to shooting algorithm (Friedman et al. 2008)

- Some issues...
  - Ballpark: several minutes for a 1000-variable problem
  - Algorithms scale as $O(d^3)$

- Other approach = ADMM
From Daniela Witten’s talk at JSM 2012:

1. The $j$th variable is unconnected from all others in the graphical lasso solution if and only if $|S_{ij}| \leq \lambda$ for all $i = 1, \ldots, j - 1, j + 1, \ldots, p$.

2. Let $A$ denote the $p \times p$ matrix whose elements take the form $A_{ii} = 1$, $A_{ij} = 1 |S_{ij}| > \lambda$. Then the connected components of $A$ are the same as the connected components of the graphical lasso solution.

We can obtain the exact right answer by solving the graphical lasso on each connected component separately!

**Citations:** Witten et al. JCGS 2011, Mazumder and Hastie JMLR 2012
Covariance Screening for Glasso

From Daniela Witten’s talk at JSM 2012:

- The solution to the graphical lasso problem with $\lambda = 0.7$ has five connected components (why 5?!)
- Perform graphical lasso on each component separately!
- **Reduction in computational time:** From $O(50^3)$ to $O(24^3)$. 