MVCAPA with correlations — meeting 2

October 02, 2019

Overview

I have:

- Proved that optimising $\mathbf{J}$ is NP-hard.
- Reformulated problem to a binary quadratic program. Opens up wide range of literature.
- Found a DP solution to optimising $\mathbf{J}$ when the correlation matrix is AR(1).
- Written everything down in Latex.
- Looked briefly on the case where $\boldsymbol{\Sigma}$ is block diagonal.

Next steps?

Why is optimising J NP-hard?

Savings:

$$S(s, e, \mathbf{J}) = (e - s) \left(\sum_{j \in \mathbf{J}} a_{jj} \bar{x}_j^2 + 2 \sum_{j \in \mathbf{J}} \sum_{i < j \in \mathbf{J}} a_{ij} \bar{x}_i \bar{x}_j \right)$$

Binary quadratic programs (BQP):

$$\max_{\mathbf{u} \in \{0,1\}^p} \mathbf{u}^T \mathbf{Q} \mathbf{u} + \mathbf{c}^T \mathbf{u}$$

Reformulation to a BQP:

$$\begin{aligned} &\max_{\mathbf{J}} \left[S(s, e, \mathbf{J}) - P(|\mathbf{J}|) \right] \\ =&(e - s) \max_{\mathbf{u}} \left[\mathbf{u}^T (\mathbf{A} \circ \bar{\mathbf{x}} \bar{\mathbf{x}}^T) \mathbf{u} - P(\|\mathbf{u}\|_2^2) \right], \end{aligned}$$

where $P(\cdot)$ has to be linear in $\mathbf{u}$.

BQPs are NP-hard, even when $\mathbf{Q}$ is positive definite.

Look at BQP solvers?

- A very general solution.
- Alex emailed Adam Letchford, got some useful tips on BQPs.
- Hundreds/thousands of articles on the topic.
- Commercial solvers exist (CPLEX and Gurobi) together with R distributions -> Easy to implement if licence (Lancaster should have CPLEX, how to I access?).
- Implement and test?

Dynamic program for spatial AR(1) structure

Let $B(d,k) = \max_{1 \le i_1 < \ldots < i_k \le d} S(\{i_1, \ldots, i_k\})$

Then

$B(d,k) =$
$\max \left[B(d-1,k), \max_{1 \le j \le k} B(d-j-1, k-j) + S((d-j+1):d) \right]$,
for $0 \le k \le p$ and $k < d \le p$, where $B(\cdot, 0) = 0$, and
$B(k,k) = S((d-j+1):d)$.

I.e.,

$\max_{\mathbf{J}:|\mathbf{J}|=k} S(\mathbf{J}) = B(p,k)$.

Unfortunately it is $O(p^3)$, but restrict $|\mathbf{J}| \le l$ and it becomes $O(l^2 p)$.

Moreover. . .

- Two penalty regimes: Sparse, dense.
- In sparse, components are added one at a time.
- In dense, all components will be added because penalty is constant over $|\mathbf{J}|$.
- So in general it should be sufficient to optimise $\mathbf{J}$ over the sparse alternatives, i.e., $|\mathbf{J}| \leq$ the boundary between the two regimes?
- Theoretically, when $|\mathbf{J}|/p \leq p^{-1/2} = a$ we are in sparse world, $>$ in dense.
- In practice, the transition between penalties occurs at $|\mathbf{J}| \geq \frac{p+2\sqrt{p\psi}}{2\log p} = b > a$ (given the i.i.d. penalties).

Tried to find pruning strategy

Does not look promising so far. Tips?

Reasoning: If $B(d-2,2)+S(\{d\}) \leq B(d-3,1)+S(\{d,d-1\})$, do I know that
$B(d-2,l)-B(d-3,l-1) \leq S(\{d,d-1\})-S(\{d\})$ for all $l \geq 2$?

Block diagonal covariance matrix

$$(e-s)\max_{\mathbf{J}_1,\ldots,\mathbf{J}_m}\left[\sum_{i=1}^{m} x_i(\mathbf{J}_i)A_i^{-1}x_i(\mathbf{J}_i) - P\left(\sum_{i=1}^{m}|\mathbf{J}_i|\right)\right]$$

- Optimise jointly.
- Optimise in parallel, then new penalty per block.
- Only assume sparsity between blocks, but not within.

What now?

- Implement and test BQP?
- Try harder to find a pruning strategy for AR(1)?
- Try to generalise DP to AR(p)?
- Should I look at the equicorrelated case?
- Block diagonal covariance matrix?
- What about L1-penalised cost function?
- ...