MVCAPA with correlations — meeting 1

September 24, 2019

Overview

I have:

- Run MVCAPA simulations with correlated data.
- Looked at the MVCAPA cost function when observations are multivariate normal with a single change in mean and a constant but general covariance matrix. How can it be optimised over (s, e, J)?

Next steps?

Simulation setup

Studied $P(\hat{K} = k)$ under different models for Σ in time-independent $N(\mu, \Sigma)$ data. Either $\Sigma_{i,j} = \rho > 0$ for $i \neq j$, or spatial AR correlations $\Sigma_{i,j} = \phi^{|i-j|}$

- $p = 4$:
  - $n = 10^3$, $K = 0$.
  - $n = 10^4$, $K = 1$, $|J| = 2$ and duration 610.
- $p = 100$
  - $n = 10^3$, $K = 0$.
  - $n = 3 \cdot 10^3$, $K = 1$, $|J| = 3$ and duration 280.

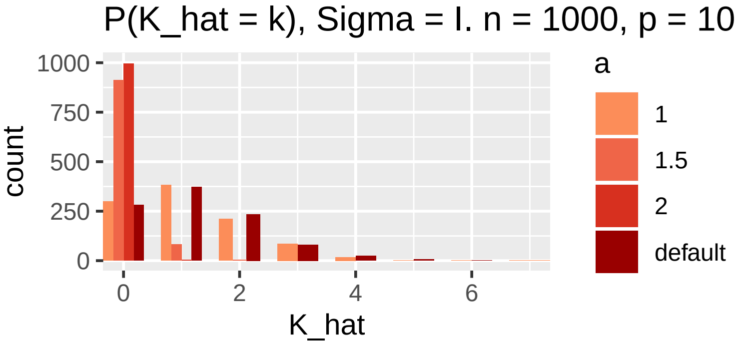
Durations are close to the minimum length dictated by Theorem 1 in the MVCAPA paper.

Penalty trouble

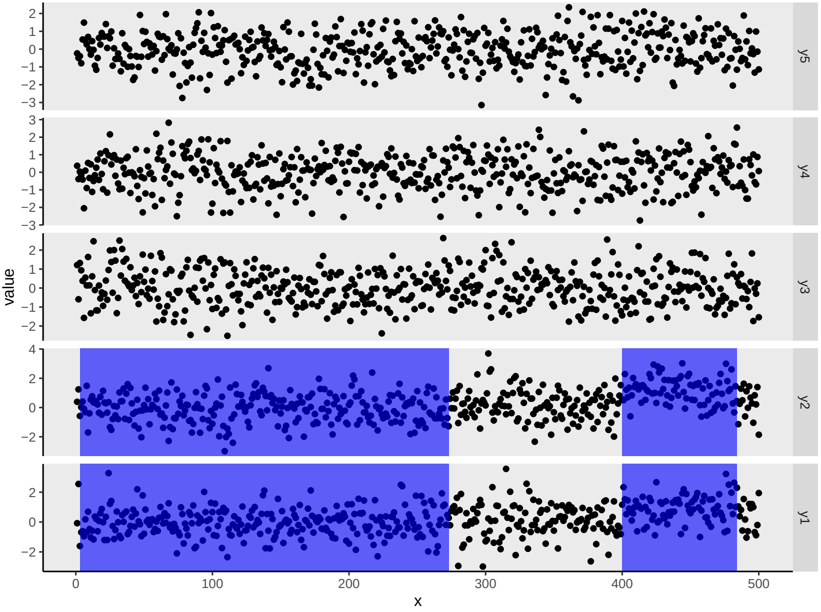
Penalty in package: Pointwise minimum between the three penalties, setting $a = 1$ in $\psi = a \log n$ in these penalties. However, theory dictates that $a > 2$ $(K = 0)$ or $a > 3$ $(K > 0)$.

So, scale penalties by a in ψ or all $\alpha, \beta_1, \ldots, \beta_p$ by b (as suggested in the paper)?

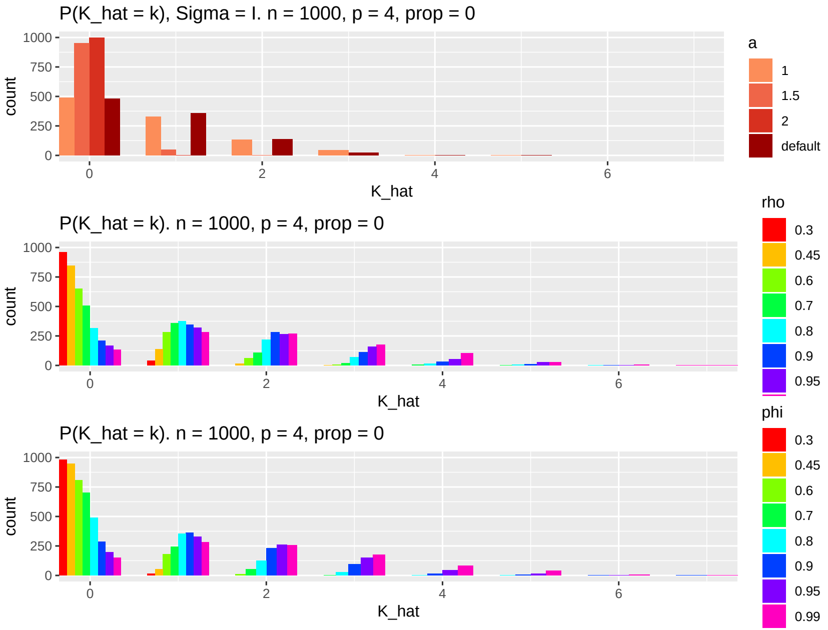
After speaking with Alex: Set $a = 3$, then scale all penalties to meet a certain false positive rate if that is desired.



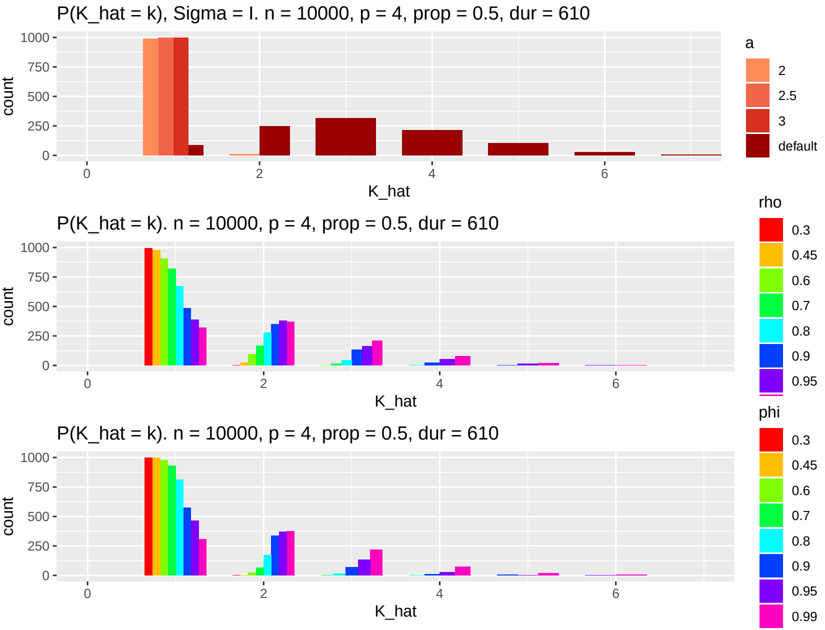
Penalty trouble 2 (from Dan)



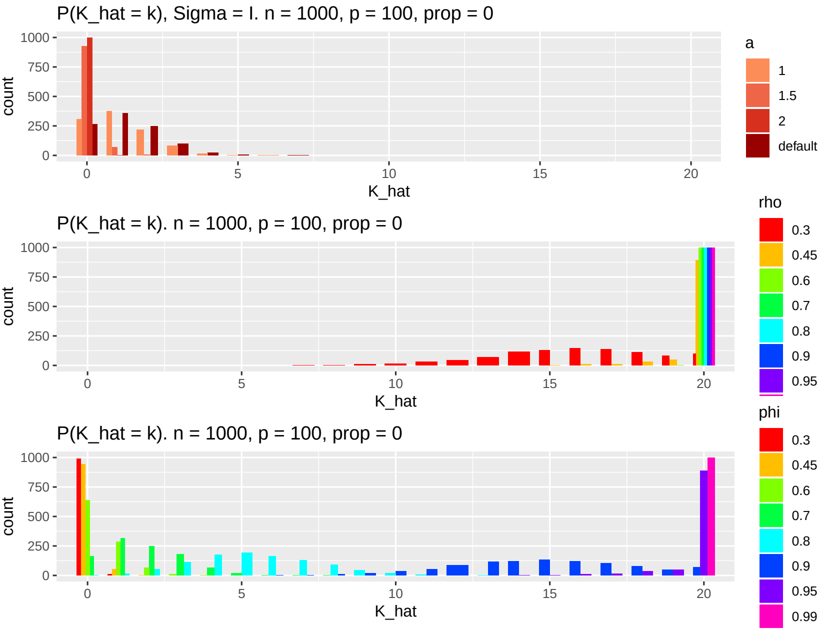
Results for $p = 4$, $K = 0$



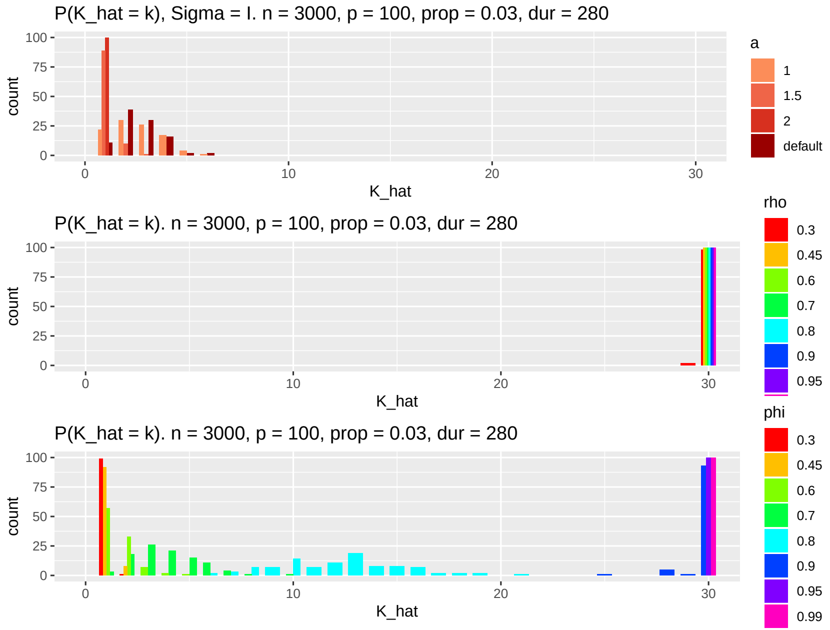
Results for $p = 4$, $K = 1$



Results for $p = 100$, $K = 0$



Results for $p = 100$, $K = 1$



MVCAPA with normal spatial dependence

$x_t \overset{i.i.d}{\sim} N(\mu_t, \Sigma_t)$, where $\Sigma_t = \Sigma$ is a known correlation matrix and $\mu_t^{(j)} = \mu^{(j)} \neq 0$ for $s < t \leq e$ and $j \in J$, and $\mu_t^{(j)} = 0$ otherwise.

Cost of introducing anomaly from s to e:
$C(x_{s+1:e}, \mu) = (e-s)\log|\Sigma| + \sum_{t=s+1}^{e}(x_t - \mu)^T\Sigma^{-1}(x_t - \mu)$

Savings: $S(s,e) = (e-s)\bar{x}_{s+1:e}^T\Sigma^{-1}\bar{x}_{s+1:e}$

To also include J, let $x(J) = (x_i I\{i \in J\})_{i=1}^p$. Then

$$\begin{aligned} S(s,e,J) &= (e-s)\bar{x}(J)_{s+1:e}^T\Sigma^{-1}\bar{x}(J)_{s+1:e} \\ &= (e-s)\left(\sum_{j\in J} a_{jj}\bar{x}_j^2 + 2\sum_{j\in J}\sum_{i<j\in J} a_{ij}\bar{x}_i\bar{x}_j\right), \end{aligned}$$

where $a_{ij} = (\Sigma^{-1})_{ij}$.

MVCAPA with normal spatial dependence

I.e., to proceed, we need an efficient solution to the combinatorial optimisation problem

$$\begin{aligned}\hat{J} &= \underset{J\in P([p])}{\operatorname{argmax}}\, x(J)^T A x(J) \\ &= \underset{J\in P([p])}{\operatorname{argmax}} \sum_{j\in J} a_{jj} x_j^2 + 2\sum_{j\in J}\sum_{i<j\in J} a_{ij} x_i x_j,\end{aligned}$$

for each $|J| = k$ and all $k = 1, \ldots, p$, where the x's and a's are given.

What have I tried?

- Structure on Σ (and therefore a's):
  - Spatial AR(1) (very simple precision matrix): Quicker to compute for each J, but the problem remains combinatorial.
- Written all out for $p = 3$ to look for recursive structure. I.e., $\hat{J}_k | \hat{J}_{k-1}$ like in the independent case.
- Is there an element-wise heuristic that can be made?
- Looks like sparse PCA, but it is not because both x and A are given.

What now?

- Try out discrete/combinatorial optimisation methods?
- Maybe PCA or some rotation is the only way to go?
- Time-dependency is easier to handle.