Understanding {pbdMPI} code MVN.R

X_i is a p-dim random vector following MVN

$X_i \sim N(\mu, \Sigma)$ μ: mean vector; Σ: (co)variance matrix

Our data is X_i, $i = 1, 2, \cdots, N$

Log likelihood of all the N data points is

$$ll = -\frac{Np}{2}\log(2\pi) - \frac{N}{2}\log|\Sigma| - \frac{1}{2}\sum_{i=1}^{N}\left[(x_i-\mu)^T\Sigma^{-1}(x_i-\mu)\right]$$

Cholesky decomposition on Σ: $\Sigma = U^TU$ (single process)

Given U, we calculate $\log|\Sigma| = 2\sum\log(\text{diag } U)$

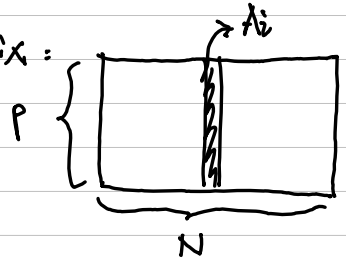
In order to calculate $(x_i-\mu)^T\Sigma^{-1}(x_i-\mu)$, we'll solve for A in $UA = B$.

Above, $B = X^T - \mu = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} - \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{pmatrix}$

X is data matrix of the entire N observations

Solving $UA = B$ can be done in parallel, using multiple processes. (MPI)

The A matrix:



For the i^{th} observation, we have $(x_i-\mu)^T\Sigma^{-1}(x_i-\mu) = A_i^TA_i$

So, each column in A correspond to one observation
A chunk of A correspond to a set of observations

To implement parallelism, we divide B into chunks, i.e. "B.spmd". Each B.spmd will produce a chunk of A, i.e. "A.spmd", via backsolve(···).

"A.spmd" is a matrix of $P \times N_s$. Now, instead of $A_i^T A_i$ like before, we would calcute

(N_s → size of the chunk)

Column sum of (A.spmd)

this is a vector of length N_s

We need to further sum over the N_s values; the sum is the LL of all the N_s observations in this chunk.

Finally, we use MPI's "reduce" function to combine results from each rank (i.e. chunk). Again, we'll use the "sum" operator to get a single value which represents all the N observations.