PCG for the RKHS CP-ALS mode-k subproblem (missing data)

Fix all CP factors except the RKHS mode k and solve for $W \in \mathbb{R}^{n\times r}$ in

$$\Big[(Z \otimes K)^\top SS^\top (Z \otimes K) + \lambda(I_r \otimes K)\Big]\operatorname{vec}(W) = (I_r \otimes K)\operatorname{vec}(B) = \operatorname{vec}(KB), \tag{1}$$

where $K \in \mathbb{R}^{n\times n}$ is a symmetric psd kernel matrix, $Z \in \mathbb{R}^{M\times r}$ is the Khatri–Rao product of the other factors, $B = TZ$ (sparse MTTKRP), and $S \in \mathbb{R}^{N\times q}$ selects the q observed entries of the mode-k unfolding $T \in \mathbb{R}^{n\times M}$ ($N = nM$). We assume $n, r < q \ll N$.

SPD structure. Let $P \equiv SS^\top$ (diagonal mask, $P^2 = P$). Define

$$A \equiv (Z \otimes K)^\top P(Z \otimes K) + \lambda(I_r \otimes K), \qquad b \equiv \operatorname{vec}(KB).$$

Then A is symmetric. If $K \succ 0$ and $\lambda > 0$, then for $x \neq 0$,

$$x^\top Ax = \|P^{1/2}(Z \otimes K)x\|_2^2 + \lambda\, x^\top (I_r \otimes K)x > 0,$$

so $A \succ 0$ and conjugate gradients applies. (If K is only psd, add a small nugget to make it SPD or reduce to the nonzero eigenspace of K.)

Matrix–vector products in $O(n^2r + qr)$ (no $O(N)$ work)

Write a vector $x \in \mathbb{R}^{nr}$ as $x = \operatorname{vec}(X)$ with $X \in \mathbb{R}^{n\times r}$ (column-stacked). The identity

$$(Z \otimes K)\operatorname{vec}(X) = \operatorname{vec}(KXZ^\top) \tag{2}$$

lets us interpret the data term as "predict on observed entries".

Observed index list. Store the q observed unfolding indices as pairs (i_t, j_t) with $i_t \in [n]$, $j_t \in [M]$. For any matrix $U \in \mathbb{R}^{n\times M}$, $S^\top \operatorname{vec}(U) = (U_{i_t,j_t})_{t=1}^q$ (gather), and Su corresponds to a sparse matrix with nonzeros u_t at the same locations (scatter).

Matvec algorithm. Given X:

1. Compute $G \leftarrow KX$ (cost $O(n^2r)$ for dense K).
2. For $t = 1, \dots, q$ let $z_t \equiv Z_{j_t,:} \in \mathbb{R}^r$ and form the scalar
$$u_t \leftarrow \langle G_{i_t,:}, z_t\rangle. \tag{3}$$
3. Accumulate $H \in \mathbb{R}^{n\times r}$ by
$$H_{i_t,:} \mathrel{+}= u_t\, z_t, \qquad t = 1, \dots, q. \tag{4}$$
This computes $H = \widetilde{U}Z$ where $\widetilde{U}$ is the sparse masked matrix with nonzeros $\widetilde{U}_{i_t,j_t} = u_t$.
4. Return $Y \leftarrow KH + \lambda G$ and output $\operatorname{vec}(Y)$.

Correctness follows from the transpose identity $(Z \otimes K)^\top \operatorname{vec}(\widetilde{U}) = \operatorname{vec}(K\widetilde{U}Z) = \operatorname{vec}(KH)$ and from $\lambda(I_r \otimes K)\operatorname{vec}(X) = \lambda \operatorname{vec}(KX)$.

Avoiding explicit Z. M can be huge, so we never form Z. For each observation we can compute z_t on the fly from the other factor rows as a Hadamard product; cost $O((d-1)r)$ per entry. If memory allows, cache all z_t once in $O(qr)$ memory so each matvec costs $O(qr)$ for the sparse loops.

RHS. Compute $B = TZ$ by the same sparse accumulation: for each observed value t_t at (i_t, j_t), do $B_{i_t,:} \mathrel{+}= t_t z_t$ (cost $O(qr)$), then set $b = \text{vec}(KB)$ (extra $O(n^2 r)$).

Preconditioning

A practical preconditioner replaces the mask by its mean under uniform sampling: $P \approx \alpha I_N$ with $\alpha \equiv q/N$. This gives

$$A_0 \equiv \alpha(Z \otimes K)^\top (Z \otimes K) + \lambda(I_r \otimes K) = \alpha(Z^\top Z) \otimes (K^2) + \lambda(I_r \otimes K). \tag{5}$$

Let $G \equiv Z^\top Z \in \mathbb{R}^{r \times r}$. In CP-ALS, G is obtained without forming Z via the Khatri–Rao Gram identity

$$G = \mathop{*}_{i \neq k} (A_i^\top A_i),$$

where $*$ denotes Hadamard product.

If $K = U \Lambda U^\top$ and $G = V \Sigma V^\top$, then A_0 diagonalizes in the Kronecker basis $(V \otimes U)$; applying A_0^{-1} to $x = \text{vec}(X)$ reduces to

$$\widehat{X} \leftarrow U^\top X V, \qquad \widehat{X}_{b,a} \leftarrow \widehat{X}_{b,a} / (\alpha \sigma_a \lambda_b^2 + \lambda \lambda_b), \qquad X \leftarrow U \widehat{X} V^\top.$$

Thus each preconditioner application costs $O(n^2 r + nr^2)$ after one-time eigendecompositions ($O(n^3 + r^3)$).

Complexity

Let m be the number of PCG iterations. Per iteration:

$$\text{matvec } Ax: \; O(n^2 r + qr), \qquad \text{preconditioner } A_0^{-1}: \; O(n^2 r + nr^2), \qquad \text{vector ops}: \; O(nr).$$

Total solve cost is $O\big(m(n^2 r + qr + nr^2)\big)$ time and $O(nr + q)$ memory (plus optional $O(qr)$ cache), with no $O(N)$ computation and no formation of the dense $(nr) \times (nr)$ matrix.