

# PCG for the RKHS mode- $k$ CP subproblem with missing entries

We solve for  $W \in \mathbb{R}^{n \times r}$  in

$$\left[ (Z \otimes K)^\top S S^\top (Z \otimes K) + \lambda (I_r \otimes K) \right] \text{vec}(W) = (I_r \otimes K) \text{vec}(B),$$

with  $K \in \mathbb{R}^{n \times n}$  a symmetric kernel matrix,  $Z \in \mathbb{R}^{M \times r}$  the Khatri–Rao product of the other CP factors, and  $S \in \mathbb{R}^{N \times q}$  selecting the  $q$  observed entries of the mode- $k$  unfolding ( $N = nM$ ). We assume  $n, r < q \ll N$ .

## 1. Why (P)CG applies

Let  $P \equiv S S^\top$  (a diagonal mask). Define

$$A = (Z \otimes K)^\top P (Z \otimes K) + \lambda (I_r \otimes K), \quad b = (I_r \otimes K) \text{vec}(B) = \text{vec}(KB).$$

$A$  is symmetric. If  $K \succ 0$  and  $\lambda > 0$ , then  $A \succ 0$ , so CG applies; preconditioning reduces iterations. Direct dense solve costs  $O((nr)^3) = O(n^3 r^3)$  and requires forming  $A$ , while PCG only needs (i) matvecs  $x \mapsto Ax$  and (ii) applying a cheap approximation  $M^{-1} \approx A^{-1}$ .

## 2. Matvec in $O(n^2 r + qr)$ without forming $A$

Represent  $x = \text{vec}(X)$  with  $X \in \mathbb{R}^{n \times r}$ . Use

$$(Z \otimes K) \text{vec}(X) = \text{vec}(K X Z^\top).$$

Store the observed index list in unfolding coordinates as pairs  $(i_t, j_t)$  for  $t = 1, \dots, q$  with  $i_t \in [n]$ ,  $j_t \in [M]$ . Then  $S^\top \text{vec}(U) = (U_{i_t, j_t})_{t=1}^q$  (gather) and  $S$  scatters a length- $q$  vector back into an  $n \times M$  sparse matrix.

Given  $X$ :

1. Compute  $G \leftarrow KX$  ( $O(n^2 r)$ ).
2. For each observed entry  $t$ : compute (or fetch) the row  $z_t \equiv Z_{j_t, :}$  and the scalar  $u_t \leftarrow G_{i_t, :} z_t^\top$  ( $O(r)$ ).
3. Accumulate  $H \in \mathbb{R}^{n \times r}$  via  $H_{i_t, :} += u_t z_t$  for  $t = 1, \dots, q$  ( $O(qr)$ ).
4. Return  $Y \leftarrow KH + \lambda G$  and output  $\text{vec}(Y)$  ( $O(n^2 r)$ ).

This equals  $Ax$  because the first term is  $(Z \otimes K)^\top \text{vec}(\tilde{U})$  with  $\tilde{U}$  the masked matrix whose nonzeros are  $u_t$  at  $(i_t, j_t)$ , and  $(Z \otimes K)^\top \text{vec}(\tilde{U}) = \text{vec}(K \tilde{U} Z) = \text{vec}(KH)$ . Equivalently, define  $\mathcal{L}(X) \equiv S^\top \text{vec}(K X Z^\top)$  so that  $A = \mathcal{L}^\top \mathcal{L} + \lambda (I \otimes K)$ ; its adjoint is  $\mathcal{L}^\top u = \text{vec}(K U Z)$  where  $U$  is the sparse matrix with  $\text{vec}(U) = Su$ , which is implemented by the same scatter/accumulate loop.

**Avoiding explicit  $Z$ .**  $M$  can be huge; instead compute each needed  $z_t$  on the fly as a Hadamard product of the other factor rows using the observed multi-index. Optionally cache all  $z_t$  once in  $O(qr)$  memory.

## 3. RHS in $O(n^2 r + qr)$

Compute  $B = TZ$  by a sparse MTTKRP over the  $q$  observed entries: for each observed value  $t_t$  at  $(i_t, j_t)$ , do  $B_{i_t, :} += t_t z_t$  ( $O(qr)$ ), then compute  $KB$  ( $O(n^2 r)$ ).

#### 4. Preconditioner exploiting Kronecker structure

A standard preconditioner replaces the mask by a scaled identity ( $P \approx \alpha I$  with  $\alpha = q/N$ ), motivated by  $\mathbb{E}[(Z \otimes K)^\top P(Z \otimes K)] = \alpha(Z^\top Z \otimes K^2)$  under uniform sampling, yielding

$$A_0 = \alpha[(Z \otimes K)^\top (Z \otimes K)] + \lambda(I \otimes K) = \alpha(Z^\top Z \otimes K^2) + \lambda(I \otimes K).$$

Let  $G \equiv Z^\top Z$  (computable without forming  $Z$  via Hadamard products of the factor Gram matrices). With eigendecompositions  $K = U\Lambda U^\top$  and  $G = V\Sigma V^\top$ ,  $A_0$  diagonalizes in the Kronecker basis ( $V \otimes U$ ) and applying  $A_0^{-1}$  to  $\text{vec}(X)$  reduces to:

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

Cost per application:  $O(n^2 r + nr^2)$  after one-time setup  $O(n^3 + r^3)$ . In matrix form,  $A_0 \text{vec}(X) = \text{vec}(K(\alpha K X G + \lambda X))$ , so applying  $A_0^{-1}$  amounts to solving the Sylvester-type equation  $\alpha K X G + \lambda X = K^{-1} R$ .

#### 5. Complexity

Let  $m$  be the PCG iteration count (typically  $m \ll nr$  with a good preconditioner). Per iteration:

$$\text{matvec } Ax : O(n^2 r + qr), \quad \text{preconditioner } M^{-1} : O(n^2 r + nr^2), \quad \text{BLAS-1} : O(nr).$$

Total solve:  $O(m(n^2 r + qr + nr^2))$  time and  $O(nr + q)$  memory (plus optional  $O(qr)$  cache), with no  $O(N)$  computation.