$$H_{n^*} D^* X^* = \left[\underbrace{(H_{\frac{n^*}{K}})}_{K} \right] \begin{bmatrix} D_1 & & 0 \\ & D_2 & \\ 0 & & \ddots \\ & & & D_K \end{bmatrix} \begin{bmatrix} X_1^{loc} \\ X_2^{loc} \\ \vdots \\ X_K^{loc} \end{bmatrix} = H_K (H_{\frac{n^*}{K}}) \begin{pmatrix} D_1 X_1^{loc} \\ D_2 X_2^{loc} \\ \vdots \\ D_K X_K^{loc} \end{pmatrix} \leftarrow (H_K \otimes H_m)\, \mathrm{diag}(D_1, \dots, D_K)^{km \times km}\, \mathrm{col}_{i=1}^{K}(X_i) \in \mathbb{R}^{km \times p}$$

e.g. $n^* = 13$, $K = 4$, $\frac{n^*}{K} = 9 = m$

$$\begin{pmatrix} H_9 & H_9 & H_9 & H_9 \\ H_9 & -H_9 & H_9 & -H_9 \\ H_9 & H_9 & -H_9 & -H_9 \\ H_9 & -H_9 & -H_9 & H_9 \end{pmatrix}$$

$$H_K \otimes H_m \rightarrow H_4 \otimes H_9$$

or could try

Discrete Cosine Transformation

$$m = \frac{n^*}{K}$$

$$= (H_K \otimes H_m) \begin{pmatrix} D_1 X_1^{loc} \\ D_2 X_2^{loc} \\ \vdots \\ D_K X_K^{loc} \end{pmatrix}$$

$$= (\overset{A}{H_K} \overset{C}{I_K}) \otimes (\overset{B}{I_m} \overset{D}{H_m})\, \mathrm{col}_i (D_i X_i^{loc})$$

$$= (H_K \otimes I_m)(I_K \otimes H_m)\, \mathrm{col}_i (D_i X_i^{loc})$$

$$= (H_K \otimes I_m) \underbrace{\begin{pmatrix} H_m & & 0 \\ & H_m & \\ 0 & & \ddots \\ & & & H_m \end{pmatrix}}_{K} \mathrm{col}_i (D_i X_i^{loc})$$

$$= (H_K \otimes I_m) \begin{pmatrix} H_m D_1 X_1^{loc} \\ H_m D_2 X_2^{loc} \\ \vdots \\ H_m D_K X_K^{loc} \end{pmatrix}$$

① Kronecker produce basics + mixed-product theory.

$$(A \otimes B)(C \otimes D) = AC \otimes BD$$

② Hadamard / FWHT as a kronecker power. =

$$H_N = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \otimes H_{\frac{N}{2}} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes H_{\frac{N}{4}} \quad \dots \quad \text{induction.}$$

$$\Updownarrow$$

$$H_{2^m} = H_2^{\otimes m}$$

Fwht compute time

Hx in $O(N \log N)$ time

Now if on each client for privacy they do Sampling of $B_1, B_2, B_3, \dots, B_K$

then this is $$(H_K \otimes I_m) \begin{pmatrix} B_1 H_m D_1 X_1^{loc} \\ B_2 H_m D_2 X_2^{loc} \\ \vdots \\ B_K H_m D_K X_K^{loc} \end{pmatrix}$$

$$(H_K \otimes I_m)\begin{pmatrix} B_1 H_m & & 0 \\ & B_2 H_m & \\ 0 & & \ddots \; B_K H_m \end{pmatrix} \mathrm{col}_i(D_i X_i^{loc})$$

$\mathrm{diag}(B_1, \ldots, B_K)$

$$(H_K \otimes I_m) B (I_K \otimes H_m) \mathrm{col}_i(D_i X_i^{loc})$$

$$[(H_K \otimes I_m) B] \mathrm{col}_i(H_m D_i X_i^{loc})$$

$$(H_K \otimes I_m)\,\mathrm{diag}(B_1, \ldots, B_K)\,\mathrm{col}_i(H_m D_i X_i^{loc})$$

if $B_1 = B_2 = \cdots = B_K$

just : each client is required to sample fixed row index of B_i $1, 3, 7, \ldots$)

$$\mathrm{diag}(B_1, \ldots, B_1)(H_K \otimes I_m)\,\mathrm{col}_i(H_m D_i X_i^{loc})$$

and it is in skipping way I say paper

$B H_{n_k} D X^k$ calculated in above way.

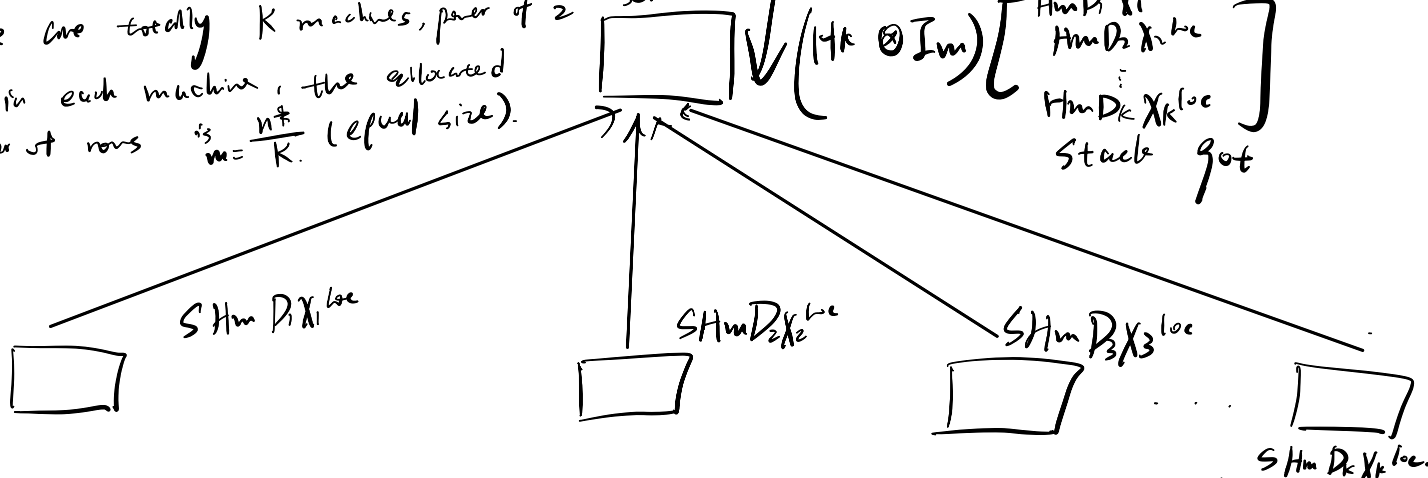
But AI says although this might ensures privacy. formal privacy. needs some JL transform "differential privacy". and no multiple round sampling of B_i.

$$B = \begin{pmatrix} s & & 0 \\ & s & \\ 0 & & \ddots \; s \end{pmatrix}$$

overall number of rows : n^*

there are totally K machines, power of 2

so in each machine, the allocated number of rows is $m = \frac{n^*}{K}$ (equal size).



scheme

$$(H_K \otimes I_m)\begin{bmatrix} H_mD_1X_1^{loc} \\ H_mD_2X_2^{loc} \\ \vdots \\ H_mD_KX_K^{loc} \end{bmatrix}$$

Stack got

K machines : $K = 2^{k'}$

local dataset.

$X_1^{loc} \in R^{m\times p}$ (client dataset) $X_2^{loc} \in R^{m\times p}$ (client dataset) $X_3^{loc} \in R^{m\times p}$ (client dataset) ... $X_K^{loc} \in R^{m\times p}$ (client dataset)

$D_1 \in R^{m\times m}$ (by client) $D_2 \in R^{m\times m}$ (by client) $D_3 \in R^{m\times m}$ (by client) $D_K \in R^{m\times m}$ (by client)

H_m is the m-order hadamard matrix. (known to both)

$H_mD_1X_1^{loc}$ $H_mD_2X_2^{loc}$ $H_mD_3X_3^{loc}$ ... $H_mD_KX_K^{loc}$

$B_1 = S \in R$ $B_2 = S$ $B_3 = S$ ... $B_K = S$

as required of sampling index matrix by the server.