Appendices

A Additional Details on Motivation in Introduction

A.1 Preprocssing all client vectors by the same random matrix does not improve performance

Consider n clients. Suppose client i holds a vector $\mathbf{x}_i \in \mathbb{R}^d$. We want to apply Rand-$k$ or Rand-k-Spatial, while also making the encoding process more flexible than just randomly choosing k out of d coordinates. One naïve way of doing this is for each client to pre-process its vector by applying an orthogonal matrix $\boldsymbol{G} \in \mathbb{R}^{d\times d}$ that is the *same* across all clients. Such a technique might be helpful in improving the performance of quantization because the MSE due to quantization often depends on how uniform the coordinates of $\mathbf{x}_i$'s are, i.e. whether the coordinates of $\mathbf{x}_i$ have values close to each other. $\boldsymbol{G}$ is designed to be the random matrix (e.g. SRHT) that rotates $\mathbf{x}_i$ and makes its coordinates uniform.

Each client sends the server $\widehat{\mathbf{x}}_i = \boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i$, where $\boldsymbol{E}_i \in \mathbb{R}^{k\times d}$ is the subsamaping matrix. If we use Rand-k, the server can decode each client vector by first applying the decoding procedure of Rand-k and then rotating it back to the original space, i.e., $\widehat{\mathbf{x}}_i^{(\text{Naïve})} = \frac{d}{k}\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i$. Note that

$$\begin{aligned}\mathbb{E}[\widehat{\mathbf{x}}_i^{(\text{Naïve})}] &= \frac{d}{k}\mathbb{E}[\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i]\\ &= \frac{d}{k}\boldsymbol{G}^T\frac{k}{d}\boldsymbol{I}_d\boldsymbol{G}\mathbf{x}_i\\ &= \mathbf{x}_i.\end{aligned}$$

Hence, $\widehat{\mathbf{x}}_i^{(\text{Naïve})}$ is unbiased. The MSE of $\widehat{\mathbf{x}}^{(\text{Naïve})} = \frac{1}{n}\sum_{i=1}^n \widehat{\mathbf{x}}_i^{(\text{Naïve})}$ is given as

$$\begin{aligned}&\mathbb{E}\left\|\bar{\mathbf{x}} - \widehat{\mathbf{x}}^{(\text{Naïve})}\right\|_2^2 = \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n\mathbf{x}_i - \frac{1}{n}\frac{d}{k}\sum_{i=1}^n\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i\right\|_2^2\\ &= \frac{1}{n^2}\mathbb{E}\left\|\sum_{i=1}^n\mathbf{x}_i - \frac{d}{k}\sum_{i=1}^n\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i\right\|_2^2\\ &= \frac{1}{n^2}\left\{\frac{d^2}{k^2}\mathbb{E}\left\|\sum_{i=1}^n\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i\right\|^2 - \left\|\sum_{i=1}^n\mathbf{x}_i\right\|^2\right\}\\ &= \frac{1}{n^2}\left\{\frac{d^2}{k^2}\left(\sum_{i=1}^n\mathbb{E}\|\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i\|_2^2 + \sum_{i\neq j}\mathbb{E}\left\langle\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i, \boldsymbol{G}\boldsymbol{E}_l^T\boldsymbol{E}_l\boldsymbol{G}\mathbf{x}_l\right\rangle\right) - \left\|\sum_{i=1}^n\mathbf{x}_i\right\|^2\right\}.\end{aligned} \tag{12}$$

Next, we bound the first term in Eq. 12.

$$\begin{aligned}&\mathbb{E}\|\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i\|_2^2 = \mathbb{E}[\mathbf{x}_i^T\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i] = \mathbb{E}[\mathbf{x}_i^T\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i]\\ &= \mathbf{x}_i^T\boldsymbol{G}^T\mathbb{E}[(\boldsymbol{E}_i^T\boldsymbol{E}_i)^2]\boldsymbol{G}\mathbf{x}_i\\ &= \mathbf{x}_i^T\frac{k}{d}\boldsymbol{I}_d\mathbf{x}_i \qquad (\because (\boldsymbol{E}_i^T\boldsymbol{E}_i)^2 = \boldsymbol{E}_i^T\boldsymbol{E}_i)\\ &= \frac{k}{d}\|\mathbf{x}_i\|_2^2\end{aligned} \tag{13}$$

The second term in Eq. 12 can also be simplified as follows.

$$\begin{aligned}&\mathbb{E}[\langle\boldsymbol{G}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{G}\mathbf{x}_i, \boldsymbol{G}^T\boldsymbol{E}_l^T\boldsymbol{E}_l\boldsymbol{G}\mathbf{x}_l\rangle]\\ &= \langle\boldsymbol{G}^T\mathbb{E}[\boldsymbol{E}_i^T\boldsymbol{E}_i]\boldsymbol{G}\mathbf{x}_i, \boldsymbol{G}^T\mathbb{E}[\boldsymbol{E}_l^T\boldsymbol{E}_l]\boldsymbol{G}\mathbf{x}_l\rangle\\ &= \langle\boldsymbol{G}^T\frac{k}{d}\boldsymbol{I}_d\boldsymbol{G}\mathbf{x}_i, \boldsymbol{G}^T\frac{k}{d}\boldsymbol{I}_d\boldsymbol{G}\mathbf{x}_l\rangle\end{aligned}$$

$$= \frac{k^2}{d^2}\langle \mathbf{x}_i, \mathbf{x}_l \rangle. \tag{14}$$

Plugging Eq. 13 and Eq. 14 into Eq. 12, we get the MSE is

$$\begin{aligned}
&\mathbb{E}\|\bar{\mathbf{x}} - \widehat{\mathbf{x}}^{(\text{Naïve})}\|_2^2 \\
&= \frac{1}{n^2}\Big\{\frac{d^2}{k^2}\Big(\sum_{i=1}^n \frac{k}{d}\|\mathbf{x}_i\|_2^2 + 2\sum_{i=1}^n\sum_{l=i+1}^n \frac{k^2}{d^2}\langle \mathbf{x}_i, \mathbf{x}_l\rangle\Big) - \Big\|\sum_{i=1}^n \mathbf{x}_i\Big\|^2\Big\} \\
&= \frac{1}{n^2}(\frac{d}{k} - 1)\sum_{i=1}^n \|\mathbf{x}_i\|_2^2,
\end{aligned}$$

which has exactly the same MSE as that of Rand-k. The problem is that if each client applies the same rotational matrix $\boldsymbol{G}$, simply rotating the vectors will not change the ℓ_2 norm of the decoded vector, and hence the MSE. Similarly, if one applies Rand-k-Spatial, one ends up having exactly the same MSE as that of Rand-k-Spatial as well. Hence, we need to design a new decoding procedure when the encoding procedure at the clients are more flexible.

A.2 $nk \gg d$ is not interesting

One can rewrite $\sum_{i=1}^n \boldsymbol{G}_i^T\boldsymbol{G}_i$ in the Rand-Proj-Spatial estimator (Eq. 5) as $\sum_{i=1}^n \boldsymbol{G}_i^T\boldsymbol{G}_i = \sum_{j=1}^{nk} \mathbf{g}_i\mathbf{g}_i^T$, where $\mathbf{g}_j \in \mathbb{R}^d$ and $\mathbf{g}_{ik}, \mathbf{g}_{ik+1}, \ldots, \mathbf{g}_{(i+1)k}$ are the rows of $\boldsymbol{G}_i$. Since when $nk \gg d$, $\sum_{j=1}^{nk} \mathbf{g}_j\mathbf{g}_j^T \to \mathbb{E}[\sum_{j=1}^n \mathbf{g}_j\mathbf{g}_j^T]$ due to Law of Large Numbers, one way to see the limiting MSE of Rand-Proj-Spatial when nk is large is to approximate $\sum_{i=1}^n\sum_{j=1}^{nk} \mathbf{g}_i\mathbf{g}_i^T$ by its expectation.

By Lemma 4.1, when $\boldsymbol{G}_i = \boldsymbol{E}_i$, Rand-Proj-Spatial recovers Rand-k-Spatial. We now discuss the limiting behavior of Rand-k-Spatial when $nk \gg d$ by leveraging our proposed Rand-Proj-Spatial. In this case, each $\mathbf{g}_j$ can be viewed as a random based vector $\mathbf{e}_w$ for w randomly chosen in $[d]$. $\sum_{i=1}^{nk} \mathbf{g}_j\mathbf{g}_j^T \to \mathbb{E}[\sum_{i=1}^{nk} \mathbf{g}_j\mathbf{g}_j^T] = \sum_{i=1}^{nk} \frac{1}{d}\boldsymbol{I}_d = \frac{nk}{d}\boldsymbol{I}_d$. And so the scalar $\bar{\beta}$ in Eq. 5 to ensure an unbiased estimator is computed as

$$\begin{aligned}
&\bar{\beta}\mathbb{E}[(\frac{nk}{d}\boldsymbol{I}_d)^\dagger \boldsymbol{G}_i^T\boldsymbol{G}_i] = \boldsymbol{I}_d \\
&\bar{\beta}\frac{d}{nk}\boldsymbol{I}_d\mathbb{E}[\boldsymbol{G}_i^T\boldsymbol{G}_i] = \boldsymbol{I}_d \\
&\bar{\beta}\frac{d}{nk}\frac{k}{d} = \boldsymbol{I}_d \\
&\bar{\beta} = n
\end{aligned}$$

And the MSE is now

$$\begin{aligned}
&\mathbb{E}\Big[\|\bar{\mathbf{x}} - \hat{\mathbf{x}}\|\Big] = \mathbb{E}\Big[\|\frac{1}{n}\sum_{i=1}^n \mathbf{x}_i - \frac{1}{n}\bar{\beta}\frac{d}{nk}\boldsymbol{I}_d\sum_{i=1}^n \boldsymbol{E}_i^T\boldsymbol{E}_i\mathbf{x}_i\|_2^2\Big] \\
&= \frac{1}{n^2}\Big\{\bar{\beta}^2\frac{d^2}{n^2k^2}\mathbb{E}\Big[\|\sum_{i=1}^n \boldsymbol{E}_i^T\boldsymbol{E}_i\mathbf{x}_i\|_2^2\Big] - \|\sum_{i=1}^n \mathbf{x}_i\|_2^2\Big\} \\
&= \frac{1}{n^2}\Big\{n^2\frac{d^2}{n^2k^2}\Big(\sum_{i=1}^n \mathbb{E}\Big[\|\boldsymbol{E}_i^T\boldsymbol{E}_i\mathbf{x}_i\|_2^2\Big] + 2\sum_{i=1}^n\sum_{l=i+1}^n \langle \boldsymbol{E}_i^T\boldsymbol{E}_i\mathbf{x}_i, \boldsymbol{E}_l^T\boldsymbol{E}_l\mathbf{x}_l\rangle\Big]\Big) - \|\sum_{i=1}^n \mathbf{x}_i\|_2^2\Big\} \\
&= \frac{1}{n^2}\Big\{\frac{d^2}{k^2}\Big(\sum_{i=1}^n \mathbb{E}\Big[\mathbf{x}_i^T(\boldsymbol{E}_i^T\boldsymbol{E}_i)^2\mathbf{x}_i\Big] + 2\sum_{i=1}^n\sum_{l=i+1}^n \frac{k^2}{d^2}\langle \mathbf{x}_i, \mathbf{x}_l\rangle\Big) - \|\sum_{i=1}^n \mathbf{x}_i\|_2^2\Big\} \\
&= \frac{1}{n^2}\Big\{\frac{d^2}{k^2}\Big(\sum_{i=1}^n \frac{k}{d}\|\mathbf{x}_i\|_2^2 + 2\sum_{i=1}^n\sum_{l=i+1}^n \frac{k^2}{d^2}\langle \mathbf{x}_i, \mathbf{x}_l\rangle\Big) - \sum_{i=1}^n \|\mathbf{x}_i\|_2^2 - 2\sum_{i=1}^n\sum_{l=i+1}^n \langle \mathbf{x}_i, \mathbf{x}_l\rangle\Big\} \\
&= \frac{1}{n^2}(\frac{d}{k} - 1)\sum_{i=1}^n \|\mathbf{x}_i\|_2^2
\end{aligned}$$

which is exactly the same MSE as Rand-k. This implies when nk is large, the MSE of Rand-k-Spatial does not get improved compared to Rand-k with correlation information. Intuitively, this implies when $nk \gg d$, the server gets enough amount of information from the client, and does not need correlation to improve its estimator. Hence, we focus on the more interesting case when $nk < d$ — that is, when the server does not have enough information from the clients, and thus wants to use additional information, i.e. cross-client correlation, to improve its estimator.

B Additional Details on the Rand-Proj-Spatial Family Estimator

B.1 $\bar{\beta}$ is a scalar

From Eq. 20 in the proof of Theorem 4.3 and Eq. 25 in the proof of Theorem 4.4, it is evident that the unbiasedness of the mean estimator $\widehat{\mathbf{x}}^{\text{Rand-Proj-Spatial}}$ is ensured collectively by

- The random sampling matrices $\{\boldsymbol{E}_i\}$.
- The orthogonality of scaled Hadamard matrices $\boldsymbol{H}^T\boldsymbol{H} = d\boldsymbol{I}_d = \boldsymbol{H}\boldsymbol{H}^T$.
- The rademacher diagonal matrices, with the property $(\boldsymbol{D}_i)^2 = \boldsymbol{I}_d$.

B.2 Alternative motivating regression problems

Alternative motivating regression problem 1.

Let $\boldsymbol{G}_i \in \mathbb{R}^{k\times d}$ and $\boldsymbol{W}_i \in \mathbb{R}^{d\times k}$ be the encoding and decoding matrix for client i. One possible alternative estimator that translates the intuition that the decoded vector should be close to the client's original vector, for all clients, is by solving the following regression problem,

$$\begin{aligned}\hat{\mathbf{x}} &= \arg\min_{\boldsymbol{W}} f(\boldsymbol{W}) = \mathbb{E}[\|\bar{\mathbf{x}} - \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{W}_i\boldsymbol{G}_i\mathbf{x}_i\|_2^2] \\ &\text{subject to } \bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\boldsymbol{W}_i\boldsymbol{G}_i\mathbf{x}_i] \end{aligned} \tag{15}$$

where $\boldsymbol{W} = (\boldsymbol{W}_1, \boldsymbol{W}_2, \ldots, \boldsymbol{W}_n)$ and the constraint enforces unbiasedness of the estimator. The estimator is then the solution of the above problem. However, we note that optimizing a decoding matrix $\boldsymbol{W}_i$ for each client leads to performing individual decoding of each client's compressed vector instead of a joint decoding process that considers all clients' compressed vectors. Only a joint decoding process can achieve the goal of leveraging cross-client information to reduce the estimation error. Indeed, we show as follows that solving the above optimization problem in Eq. 15 recovers the MSE of our baseline Rand-k. Note

$$\begin{aligned} f(\boldsymbol{W}) &= \mathbb{E}[\|\frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{W}_i\boldsymbol{G}_i\mathbf{x}_i)\|_2^2] = \mathbb{E}[\|\frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{I}_d - \boldsymbol{W}_i\boldsymbol{G}_i)\mathbf{x}_i\|_2^2] \\ &= \mathbb{E}\Big[\frac{1}{n^2}\Big(\sum_{i=1}^{n}\|(\boldsymbol{I}_d - \boldsymbol{W}_i\boldsymbol{G}_i\mathbf{x}_i)\|_2^2 + \sum_{i\neq j}\Big\langle(\boldsymbol{I}_d - \boldsymbol{W}_i\boldsymbol{G}_i)\mathbf{x}_i, (\boldsymbol{I}_d - \boldsymbol{W}_j\boldsymbol{G}_j)\mathbf{x}_j\Big\rangle\Big)\Big] \\ &= \frac{1}{n^2}\Big(\sum_{i=1}^{n}\mathbb{E}\Big[\|(\boldsymbol{I}_d - \boldsymbol{W}_i\boldsymbol{G}_i)\mathbf{x}_i\|_2^2\Big] + \sum_{i\neq j}\mathbb{E}\Big[\Big\langle(\boldsymbol{I}_d - \boldsymbol{W}_i\boldsymbol{G}_i)\mathbf{x}_i, (\boldsymbol{I}_d - \boldsymbol{W}_j\boldsymbol{G}_j)\mathbf{x}_j\Big\rangle\Big]\Big). \end{aligned} \tag{16}$$

By the constraint of unbiasedness, i.e., $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\boldsymbol{W}_i\boldsymbol{G}_i\mathbf{x}_i]$, there is

$$\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i - \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\boldsymbol{W}_i\boldsymbol{G}_i\mathbf{x}_i] = 0 \Leftrightarrow \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[(\boldsymbol{I}_d - \boldsymbol{W}_i\boldsymbol{G}_i)\mathbf{x}_i] = 0.$$

We now show that a sufficient and necessary condition to satisfy the above unbiasedness constraint is that for all $i \in [n]$, $\mathbb{E}[\boldsymbol{W}_i\boldsymbol{G}_i] = \boldsymbol{I}_d$.

Sufficiency. It is obvious that if for all $i \in [n]$, $\mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i] = \boldsymbol{I}_d$, then we have $\frac{1}{n}\mathbb{E}[(\boldsymbol{I}_d - \boldsymbol{W}_i \boldsymbol{G}_i)\mathbf{x}_i] = 0$.

Necessity. Consider the special case that for some $i \in [n]$ and $\lambda \in [d]$, $\mathbf{x}_i = n\mathbf{e}_\lambda$, where $\mathbf{e}_\lambda$ is the λ-th canonical basis vector, and $\mathbf{x}_j = 0$, and for all $j \in [n] \setminus \{i\}$. Then,

$$\mathbf{e}_\lambda = \bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i \mathbf{x}_i] = \frac{1}{n}\mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i]\mathbf{e}_\lambda = [\mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i]]_\lambda,$$

where $[\cdot]_\lambda$ denotes the λ-th column of matrix $\mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i]$.

Since our approach is agnostic to the choice of vectors, we need this choice of decoder matrices, by varying λ over $[d]$, we see that we need $\mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i] = \boldsymbol{I}_d$. And by varying i over $[n]$, we see that we need $\mathbb{E}[\boldsymbol{W}_j \boldsymbol{G}_j] = \boldsymbol{I}_d$ for all $j \in [n]$.

Therefore, $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i \mathbf{x}_i] \Leftrightarrow \forall i \in [n], \mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i] = \boldsymbol{I}_d$.

This implies the second term of $f(\boldsymbol{W})$ in Eq. 16 is 0, that is,

$$\sum_{i \neq j} \mathbb{E}\Big[\Big\langle (\boldsymbol{I}_d - \boldsymbol{W}_i \boldsymbol{G}_i)\mathbf{x}_i, (\boldsymbol{I}_d - \boldsymbol{W}_j \boldsymbol{G}_j)\mathbf{x}_j \Big\rangle = 0.$$

Hence, we only need to solve

$$\hat{\mathbf{x}} = \arg\min_{\boldsymbol{W}} f_2(\boldsymbol{W}) = \sum_{i=1}^{n} \mathbb{E}\Big[\|(\boldsymbol{I}_d - \boldsymbol{W}_i \boldsymbol{G}_i)\mathbf{x}_i\|_2^2\Big] \tag{17}$$

Since each $\boldsymbol{W}_i$ appears in $f_2(\boldsymbol{W})$ separately, each $\boldsymbol{W}_i$ can be optimized separately, via solving

$$\min_{\boldsymbol{W}_i} \mathbb{E}\Big[\|(\boldsymbol{I}_d - \boldsymbol{W}_i \boldsymbol{G}_i)\mathbf{x}_i\|_2^2\Big] \quad \text{subject to } \mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i] = \boldsymbol{I}_d.$$

One natural solution is to take $\boldsymbol{W}_i = \frac{d}{k}\boldsymbol{G}_i^\dagger$, $\forall i \in [n]$. For $i \in [n]$, let $\boldsymbol{G}_i = \boldsymbol{V}_i \Lambda_i \boldsymbol{U}_i^T$ be its SVD, where $\boldsymbol{V}_i \in \mathbb{R}^{k \times d}$ and $\boldsymbol{U}_i \in \mathbb{R}^{d \times d}$ are orthogonal matrices. Then,

$$\boldsymbol{W}_i \boldsymbol{G}_i = \frac{d}{k}\boldsymbol{U}_i \Lambda_i^\dagger \boldsymbol{V}_i^T \boldsymbol{V}_i \Lambda \boldsymbol{U}^T = \frac{d}{k}\boldsymbol{U}_i \Lambda_i^\dagger \Lambda \boldsymbol{U}^T = \frac{d}{k}\boldsymbol{U}_i \Sigma \boldsymbol{U}_i^T,$$

where Σ is a diagonal matrix with 0s and 1s on the diagonal.

For simplicity, we assume the random matrix $\boldsymbol{U}_i$ follows a continuous distribution. $\boldsymbol{U}_i$ being discrete follows a similar analysis. Let $\mu(\boldsymbol{U}_i)$ be the measure of $\boldsymbol{U}_i$.

$$\begin{aligned}
\mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i] &= \frac{d}{k}\mathbb{E}[\boldsymbol{U}_i \Sigma \boldsymbol{U}_i^T] = \frac{d}{k}\int_{\boldsymbol{U}_i} \mathbb{E}[\boldsymbol{U}_i \Sigma_i \boldsymbol{U}_i^T \mid \boldsymbol{U}_i] \cdot d\mu(\boldsymbol{U}_i) \\
&= \frac{d}{k}\int_{\boldsymbol{U}_i} \boldsymbol{U}_i \mathbb{E}[\Sigma_i \mid \boldsymbol{U}_i]\boldsymbol{U}_i^T \cdot \mu(\boldsymbol{U}_i) \\
&= \frac{d}{k}\int_{\boldsymbol{U}_i} \boldsymbol{U}_i \frac{k}{d}\boldsymbol{I}_d \boldsymbol{U}_i^T \cdot d\mu(\boldsymbol{U}_i) \\
&= \frac{d}{k}\frac{k}{d}\boldsymbol{I}_d = \boldsymbol{I}_d,
\end{aligned}$$

which means the estimator $\frac{1}{n}\sum_{i=1}^{n} \frac{k}{d}\boldsymbol{G}_i^\dagger \boldsymbol{G}_i$ satisfies unbiasedness. The MSE is now

$$\begin{aligned}
MSE &= \mathbb{E}\Big[\|\bar{\mathbf{x}} - \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{W}_i \boldsymbol{G}_i \mathbf{x}_i\|_2^2\Big] = \frac{1}{n^2}\sum_{i=1}^{n} \mathbb{E}\Big[\|(\boldsymbol{I}_d - \boldsymbol{W}_i \boldsymbol{G}_i)\mathbf{x}_i\|_2^2\Big] \\
&= \frac{1}{n^2}\sum_{i=1}^{n} \Big(\|\mathbf{x}_i\|_2^2 + \mathbb{E}[\|\boldsymbol{W}_i \boldsymbol{G}_i \mathbf{x}_i\|_2^2] - 2\langle \mathbf{x}_i, \mathbb{E}[\boldsymbol{W}_i \boldsymbol{G}_i]\mathbf{x}_i \rangle\Big) \\
&= \frac{1}{n^2}\sum_{i=1}^{n} \Big(\|\mathbf{x}_i\|_2^2 + \mathbb{E}[\|\boldsymbol{W}_i \boldsymbol{G}_i \mathbf{x}_i\|_2^2] - 2\langle \mathbf{x}_i, \mathbf{x}_i \rangle\Big)
\end{aligned}$$

$$= \frac{1}{n^2}\sum_{i=1}^{n}\Big(\mathbb{E}[\|\boldsymbol{W}_i\boldsymbol{G}_i\mathbf{x}_i\|_2^2 - \|\mathbf{x}_i\|_2^2]\Big)$$
$$= \frac{1}{n^2}\sum_{i=1}^{n}\Big(\mathbf{x}_i\mathbb{E}[(\boldsymbol{W}_i\boldsymbol{G}_i)^T(\boldsymbol{W}_i\boldsymbol{G}_i)]\mathbf{x}_i - \|\mathbf{x}_i\|_2^2\Big).$$

Again, let $\boldsymbol{G}_i = \boldsymbol{V}_i\Lambda_i\boldsymbol{U}_i^T$ be its SVD and consider $\boldsymbol{W}_i\boldsymbol{G}_i = \frac{d}{k}\boldsymbol{U}_i\Sigma_i\boldsymbol{U}_i^T$, where Σ_i is a diagonal matrix with 0s and 1s. Then,

$$MSE = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{i=1}^{n}\Big(\mathbf{x}_i^T\frac{d^2}{k^2}\mathbb{E}[\boldsymbol{U}_i\Sigma_i\boldsymbol{U}_i^T\boldsymbol{U}_i\Sigma_i\boldsymbol{U}_i^T]\mathbf{x}_i - \|\mathbf{x}_i\|_2^2\Big)$$
$$= \frac{1}{n^2}\sum_{i=1}^{n}\Big(\frac{d^2}{k^2}\mathbf{x}_i^T\mathbb{E}[\boldsymbol{U}_i\Sigma^2\boldsymbol{U}_i^T]\mathbf{x}_i - \|\mathbf{x}_i\|_2^2\Big).$$

Since $\boldsymbol{G}_i$ has rank k, Σ_i is a diagonal matrix with k out of d entries being 1 and the rest being 0. Let $\mu(\boldsymbol{U}_i)$ be the measure of $\boldsymbol{U}_i$. Hence, for $i \in [n]$,

$$\begin{aligned}
\mathbb{E}[\boldsymbol{U}_i\Sigma_i^2\boldsymbol{U}_i^T] &= \int_{\boldsymbol{U}_i}\mathbb{E}[\boldsymbol{U}_i\Sigma_i^2\boldsymbol{U}_i^T \mid \boldsymbol{U}_i]d\mu(\boldsymbol{U}_i)\\
&= \int_{\boldsymbol{U}_i}\boldsymbol{U}_i\mathbb{E}[\Sigma_i^2 \mid \boldsymbol{U}_i]\boldsymbol{U}_i^T d\mu(\boldsymbol{U}_i)\\
&= \int_{\boldsymbol{U}_i}\frac{k}{d}\boldsymbol{U}_i\boldsymbol{I}_d\boldsymbol{U}_i^T d\mu(\boldsymbol{U}_i)\\
&= \frac{k}{d}\int_{\boldsymbol{U}_i}\boldsymbol{I}_d d\mu(\boldsymbol{U}_i)\\
&= \frac{k}{d}\boldsymbol{I}_d.
\end{aligned}$$

Therefore, the MSE of the estimator, which is the solution of the optimization problem in Eq. 15, is

$$MSE = \frac{1}{n^2}\sum_{i=1}^{n}\Big(\frac{d^2}{k^2}\mathbf{x}_i^T\frac{k}{d}\boldsymbol{I}_d\mathbf{x}_i - \|\mathbf{x}_i\|_2^2\Big) = \frac{1}{n^2}(\frac{d}{k}-1)\sum_{i=1}^{n}\|\mathbf{x}_i\|_2^2,$$

which is the same MSE as that of Rand-k.

Alternative motivating regression problem 2.

Another motivating regression problem based on which we can design our estimator is

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}}\|\frac{1}{n}\sum_{i=1}^{n}\mathbf{G}_i\mathbf{x} - \frac{1}{n}\sum_{i=1}^{n}\mathbf{G}_i\mathbf{x}_i\|_2^2 \tag{18}$$

Note that $\boldsymbol{G}_i \in \mathbb{R}^{k\times d}, \forall i \in [n]$, and so the solution to the above problem is

$$\widehat{\mathbf{x}}^{(\text{solution})} = \Big(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{G}_i\Big)^{\dagger}\Big(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{G}_i\mathbf{x}_i\Big),$$

and to ensure unbiasedness of the estimator, we can set $\bar{\beta} \in \mathbb{R}$ and have the estimator as

$$\widehat{\mathbf{x}}^{(\text{estimator})} = \bar{\beta}\Big(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{G}_i\Big)^{\dagger}\Big(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{G}_i\mathbf{x}_i\Big).$$

It is not hard to see this estimator does not lead to an MSE as low as Rand-Proj-Spatial does. Consider the full correlation case, i.e., $\mathbf{x}_i = \mathbf{x}, \forall i \in [n]$, for example, the estimator is now

$$\widehat{\mathbf{x}}^{(\text{estimator})} = \bar{\beta}\Big(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{G}_i\Big)^{\dagger}\Big(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{G}_i\Big)\mathbf{x}.$$

Note that $\text{rank}(\frac{1}{n}\sum_{i=1}^n \boldsymbol{G}_i)$ is at most k, since $\boldsymbol{G}_i \in \mathbb{R}^{k\times d}$, $\forall i \in [k]$. This limits the amount of information of $\mathbf{x}$ the server can recover.

While recall that in this case, the Rand-Proj-Spatial estimator is

$$\widehat{\mathbf{x}}^{(\text{Rand-Proj-Spatial})} = \bar{\beta}\Big(\sum_{i=1}^n \boldsymbol{G}_i^T\boldsymbol{G}_i\Big)^\dagger \sum_{i=1}^n \boldsymbol{G}_i^T\boldsymbol{G}_i\mathbf{x} = \bar{\beta}\boldsymbol{S}^\dagger\boldsymbol{S}\mathbf{x},$$

where $\boldsymbol{S}$ can have rank at most nk.

B.3 Why deriving the MSE of Rand-Proj-Spatial with SRHT is hard

To analyze Eq. 11, one needs to compute the distribution of eigendecomposition of $\boldsymbol{S} = \sum_{i=1}^n \boldsymbol{G}_i^T\boldsymbol{G}_i$, i.e. the sum of the covariance of SRHT. To the best of our knowledge, there is no non-trivial closed form expression of the distribution of eigen-decomposition of even a single $\boldsymbol{G}_i^T\boldsymbol{G}_i$, when $\boldsymbol{G}_i$ is SRHT, or other commonly used random matrices, e.g. Gaussian. When $\boldsymbol{G}_i$ is SRHT, since $\boldsymbol{G}_i^T\boldsymbol{G}_i = \boldsymbol{D}_i\boldsymbol{H}\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i$ and the eigenvalues of $\boldsymbol{E}_i^T\boldsymbol{E}_i$ are just diagonal entries, one might attempt to analyze $\boldsymbol{H}\boldsymbol{D}_i$. While the hardmard matrix $\boldsymbol{H}$'s eigenvalues and eigenvectors are known[3], the result can hardly be applied to analyze the distribution of singular values or singular vectors of $\boldsymbol{H}\boldsymbol{D}_i$.

Even if one knows the eigen-decomposition of a single $\boldsymbol{G}_i^T\boldsymbol{G}_i$, it is still hard to get the eigen-decomposition of $\boldsymbol{S}$. The eigenvalues of a matrix $\boldsymbol{A}$ can be viewed as a non-linear function in the $\boldsymbol{A}$, and hence it is in general hard to derive closed form expressions for the eigenvalues of $\boldsymbol{A}+\boldsymbol{B}$, given the eigenvalues of $\boldsymbol{A}$ and that of $\boldsymbol{B}$. One exception is when $\boldsymbol{A}$ and $\boldsymbol{B}$ have the same eigenvector and the eigenvalues of $\boldsymbol{A}+\boldsymbol{B}$ becomes a sum of the eigenvalues of $\boldsymbol{A}$ and $\boldsymbol{B}$. Recall when $\boldsymbol{G}_i = \boldsymbol{E}_i$, Rand-Proj-Spatial recovers Rand-k-Spatial. Since $\boldsymbol{E}_i^T\boldsymbol{E}_i$'s all have the same eigenvectors (i.e. same as $\boldsymbol{I}_d$), the eigenvalues of $\boldsymbol{S} = \sum_{i=1}^n \boldsymbol{E}_i^T\boldsymbol{E}_i$ are just the sum of diagonal entries of $\boldsymbol{E}_i^T\boldsymbol{E}_i$'s. Hence, deriving the MSE for Rand-k-Spatial is not hard compared to the more general case when $\boldsymbol{G}_i^T\boldsymbol{G}_i$'s can have different eigenvectors.

Since one can also view $\boldsymbol{S} = \sum_{i=1}^{nk} \mathbf{g}_i\mathbf{g}_i^T$, i.e. the sum of nk rank-one matrices, one might attempt to recursively analyze the eigen-decomposition of $\sum_{i=1}^{n'} \mathbf{g}_i\mathbf{g}_i^T + \mathbf{g}_{n'+1}\mathbf{g}_{n'+1}^T$ for $n' \leq n$. One related problem is eigen-decomposition of a low-rank updated matrix in perturbation analysis: Given the eigen-decomposition of a matrix $\boldsymbol{A}$, what is the eigen-decomposition of $\boldsymbol{A}+\boldsymbol{V}\boldsymbol{V}^T$, where $\boldsymbol{V}$ is low-rank matrix (or more commonly rank-one)? To compute the eigenvalues of $\boldsymbol{A}+\boldsymbol{V}\boldsymbol{V}^T$ directly from that of $\boldsymbol{A}$, the most effective and widely applied solution is to solve the so-called secular equation, e.g. [59, 60, 61]. While this can be done computationally efficiently, it is hard to get a closed form expression for the eigenvalues of $\boldsymbol{A}+\boldsymbol{V}\boldsymbol{V}^T$ from the secular equation.

The previous analysis of SRHT in e.g. [37, 38, 39, 45, 55] is based on asymptotic properties of SRHT, such as the limiting eigen-spectrum, or concentration bounds that bounds the singular values. To analyze the MSE of Rand-Proj-Spatial, however, we need an exact, non-asymptotic analysis of the distribution of SRHT. Concentration bounds does not apply, since computing the pseudo-inverse in Eq. 5 naturally bounds the eigenvalues, and applying concentration bounds will only lead to a loose upper bound on MSE.

[3] See this note https://core.ac.uk/download/pdf/81967428.pdf

B.4 More simulation results on incorporating various degrees of correlation

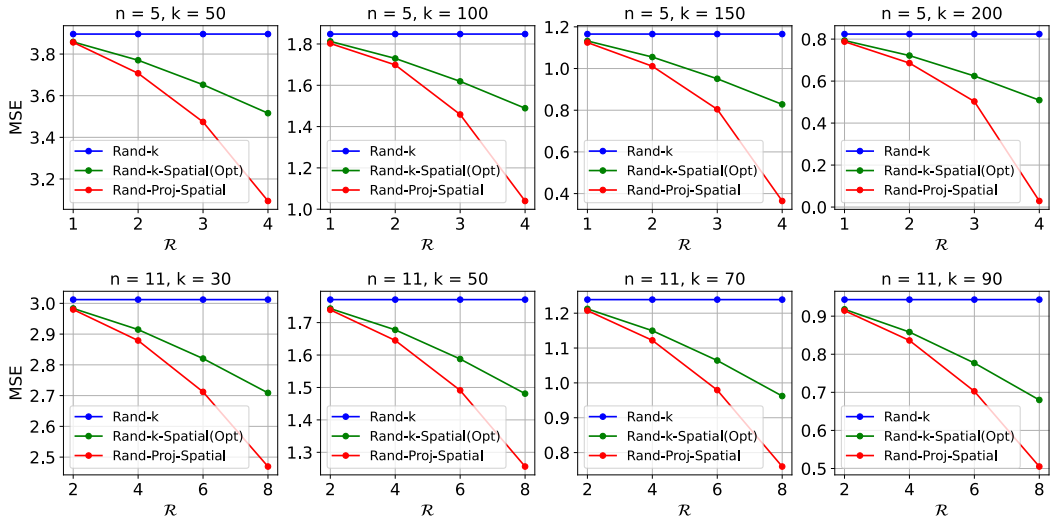

Figure 6: MSE comparison of estimators Rand-k, Rand-k-Spatial(Opt), Rand-Proj-Spatial, given the degree of correlation $\mathcal{R}$. Rand-k-Spatial(Opt) denotes the estimator that gives the lowest possible MSE from the Rand-k-Spatial family. We consider $d = 1024$, a smaller number of clients $n \in \{5, 11\}$, and k values such that $nk < d$. In each plot, we fix n, k, d and vary the degree of positive correlation $\mathcal{R}$. Note the range of $\mathcal{R}$ is $\mathcal{R} \in [0, n-1]$. We choose $\mathcal{R}$ with equal space in this range.

C All Proof Details

C.1 Proof of Theorem 4.3

Theorem 4.3 (MSE under Full Correlation)**.** *Consider n clients, each holding the same vector $\mathbf{x} \in \mathbb{R}^d$. Suppose we set $T(\lambda) = \lambda$, $\bar{\beta} = \frac{d}{k}$ in Eq. 5, and the random linear map $\boldsymbol{G}_i$ at each client to be an SRHT matrix. Let δ be the probability that $\boldsymbol{S} = \sum_{i=1}^n \boldsymbol{G}_i^T \boldsymbol{G}_i$ does not have full rank. Then, for $nk \leq d$,*

$$\mathbb{E}\left[\|\widehat{\mathbf{x}}^{(\textit{Rand-Proj-Spatial(Max)})} - \bar{\mathbf{x}}\|_2^2\right] \leq \left[\frac{d}{(1-\delta)nk + \delta k} - 1\right]\|\mathbf{x}\|_2^2 \tag{19}$$

Proof. All clients have the same vector $\mathbf{x}_1 = \mathbf{x}_2 = \cdots = \mathbf{x}_n = \mathbf{x} \in \mathbb{R}^d$. Hence, $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i = \mathbf{x}$, and the decoding scheme is

$$\widehat{\mathbf{x}}^{(\text{Rand-Proj-Spatial(Max)})} = \bar{\beta}\Big(\sum_{i=1}^n \boldsymbol{G}_i^T \boldsymbol{G}_i\Big)^\dagger \sum_{i=1}^n \boldsymbol{G}_i^T \boldsymbol{G}_i \mathbf{x} = \bar{\beta}\boldsymbol{S}^\dagger \boldsymbol{S}\mathbf{x},$$

where $\boldsymbol{S} = \sum_{i=1}^n \boldsymbol{G}_i^T \boldsymbol{G}_i$. Let $\boldsymbol{S} = \boldsymbol{U}\Lambda\boldsymbol{U}^T$ be its eigendecomposition. Since $\boldsymbol{S}$ is a real symmetric matrix, $\boldsymbol{U}$ is orthogonal, i.e., $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}_d = \boldsymbol{U}\boldsymbol{U}^T$. Also, $\boldsymbol{S}^\dagger = \boldsymbol{U}\Lambda^\dagger\boldsymbol{U}^T$, where $\Lambda^\dagger$ is a diagonal matrix, such that

$$[\Lambda^\dagger]_{ii} = \begin{cases} 1/[\Lambda]_{ii} & \text{if } \Lambda_{ii} \neq 0, \\ 0 & \text{else.} \end{cases}$$

Let δ_c be the probability that $\boldsymbol{S}$ has rank c, for $c \in \{k, k+1, \ldots, nk-1\}$. Note that $\delta = \sum_{c=k}^{nk-1} \delta_c$. For vector $\mathbf{m} \in \mathbb{R}^d$, we use $\text{diag}(\mathbf{m}) \in \mathbb{R}^{d\times d}$ to denote the matrix whose diagonal entries correspond to the coordinates of $\mathbf{m}$ and the rest of the entries are zeros.

Computing $\bar{\beta}$. First, we compute $\bar{\beta}$. To ensure that our estimator $\widehat{\mathbf{x}}^{(\text{Rand-Proj-Spatial(Max)})}$ is unbiased, we need $\bar{\beta}\mathbb{E}[\boldsymbol{S}^\dagger\boldsymbol{S}\mathbf{x}] = \mathbf{x}$. Consequently,

$$\mathbf{x} = \bar{\beta}\mathbb{E}[\boldsymbol{U}\Lambda^\dagger\boldsymbol{U}^T\boldsymbol{U}\Lambda\boldsymbol{U}^T]\mathbf{x}$$

$$
\begin{aligned}
&= \bar{\beta}\left[\sum_{\boldsymbol{U}=\Phi} \Pr[\boldsymbol{U}=\Phi]\mathbb{E}[\boldsymbol{U}\Lambda^{\dagger}\Lambda\boldsymbol{U}^T \mid \boldsymbol{U}=\Phi]\right]\mathbf{x} \\
&= \bar{\beta}\left[\sum_{\boldsymbol{U}=\Phi} \Pr[\boldsymbol{U}=\Phi]\boldsymbol{U}\mathbb{E}[\Lambda^{\dagger}\Lambda \mid \boldsymbol{U}=\Phi]\boldsymbol{U}^T\right]\mathbf{x} \\
&\overset{(a)}{=} \bar{\beta}\left[\sum_{\boldsymbol{U}=\Phi} \Pr[\boldsymbol{U}=\Phi]\boldsymbol{U}\mathbb{E}[\text{diag}(\mathbf{m}) \mid \boldsymbol{U}=\Phi]\boldsymbol{U}^T\right]\mathbf{x} \\
&\overset{(b)}{=} \bar{\beta}\sum_{\boldsymbol{U}=\Phi} \Pr[\boldsymbol{U}=\Phi]\left[\boldsymbol{U}\Big((1-\delta)\frac{nk}{d}\boldsymbol{I}_d + \sum_{c=k}^{nk-1}\delta_c\frac{c}{d}\boldsymbol{I}_d\Big)\boldsymbol{U}^T\right]\mathbf{x} \\
&= \bar{\beta}\Big[(1-\delta)\frac{nk}{d} + \sum_{c=k}^{nk-1}\delta_c\frac{c}{d}\Big]\mathbf{x} \\
\Rightarrow \bar{\beta} &= \frac{d}{(1-\delta)nk + \sum_{c=k}^{nk-1}\delta_c c}
\end{aligned} \tag{20}
$$

where in (a), $\mathbf{m} \in \mathbb{R}^d$ such that

$$
\mathbf{m}_i = \begin{cases} 1 & \text{if } \Lambda_{jj} > 0 \\ 0 & \text{else.} \end{cases}
$$

Also, by construction of $\boldsymbol{S}$, $\text{rank}(\text{diag}(\mathbf{m})) \leq nk$. Further, (b) follows by symmetry across the d dimensions.

Since $\delta k \leq \sum_{c=k}^{nk-1}\delta_c c \leq \delta(nk-1)$, there is

$$
\frac{d}{(1-\delta)nk + \delta(nk-1)} \leq \bar{\beta} \leq \frac{d}{(1-\delta)nk + \delta k} \tag{21}
$$

Computing the MSE. Next, we use the value of $\bar{\beta}$ in Eq. 20 to compute MSE.

$$
\begin{aligned}
&MSE(\text{Rand-Proj-Spatial(Max)}) = \mathbb{E}[\|\widehat{\mathbf{x}}^{(\text{Rand-Proj-Spatial(Max)})} - \bar{\mathbf{x}}\|_2^2] = \mathbb{E}[\|\bar{\beta}\boldsymbol{S}^{\dagger}\boldsymbol{S}\mathbf{x} - \mathbf{x}\|_2^2] \\
&= \bar{\beta}^2\mathbb{E}[\|\boldsymbol{S}^{\dagger}\boldsymbol{S}\mathbf{x}\|_2^2] + \|\mathbf{x}\|_2^2 - 2\Big\langle\bar{\beta}\mathbb{E}[\boldsymbol{S}^{\dagger}\boldsymbol{S}\mathbf{x}], \mathbf{x}\Big\rangle \\
&= \bar{\beta}^2\mathbb{E}[\|\boldsymbol{S}^{\dagger}\boldsymbol{S}\mathbf{x}\|_2^2] - \|\mathbf{x}\|_2^2 \qquad \text{(Using unbiasedness of } \widehat{\mathbf{x}}^{(\text{Rand-Proj-Spatial(Max)})}\text{)} \\
&= \bar{\beta}^2\mathbf{x}^T\mathbb{E}[\boldsymbol{S}^T(\boldsymbol{S}^{\dagger})^T\boldsymbol{S}^{\dagger}\boldsymbol{S}]\mathbf{x} - \|\mathbf{x}\|_2^2.
\end{aligned} \tag{22}
$$

Using $\boldsymbol{S}^{\dagger} = \boldsymbol{U}\Lambda^{\dagger}\boldsymbol{U}^T$,

$$
\begin{aligned}
\mathbb{E}[\boldsymbol{S}^T(\boldsymbol{S}^{\dagger})^T\boldsymbol{S}^{\dagger}\boldsymbol{S}] &= \mathbb{E}[\boldsymbol{U}\Lambda\boldsymbol{U}^T\boldsymbol{U}\Lambda^{\dagger}\boldsymbol{U}^T\boldsymbol{U}\Lambda^{\dagger}\boldsymbol{U}^T\boldsymbol{U}\Lambda\boldsymbol{U}^T] \\
&= \mathbb{E}[\boldsymbol{U}\Lambda(\Lambda^{\dagger})^2\Lambda\boldsymbol{U}^T] \\
&= \sum_{\boldsymbol{U}=\Phi}\boldsymbol{U}\mathbb{E}[\Lambda(\Lambda^{\dagger})^2\Lambda]\boldsymbol{U}^T \cdot \Pr[\boldsymbol{U}=\Phi] \\
&= \sum_{\boldsymbol{U}=\Phi}\boldsymbol{U}\Big[(1-\delta)\frac{nk}{d}\boldsymbol{I}_d + \sum_{c=k}^{nk-1}\delta_c\frac{c}{d}\boldsymbol{I}_d\Big]\boldsymbol{U}^T \cdot \Pr[\boldsymbol{U}=\Phi] \\
&= \Big[(1-\delta)\frac{nk}{d} + \sum_{c=k}^{nk-1}\delta_c\frac{c}{d}\Big]\cdot\sum_{\boldsymbol{U}=\Phi}\boldsymbol{U}\boldsymbol{U}^T \cdot \Pr[\boldsymbol{U}=\Phi] \\
&= \Big[(1-\delta)\frac{nk}{d} + \sum_{c=k}^{nk-1}\delta_c\frac{c}{d}\Big]\boldsymbol{I}_d \\
&= \frac{1}{\bar{\beta}}\boldsymbol{I}_d
\end{aligned} \tag{23}
$$

Substituting Eq. 23 in Eq. 22, we get

$$MSE(\text{Rand-Proj-Spatial(Max)}) = \bar{\beta}^2 \mathbf{x}^T \frac{1}{\bar{\beta}} \boldsymbol{I}_d \mathbf{x} - \|\mathbf{x}\|_2^2 = (\bar{\beta} - 1)\|\mathbf{x}\|_2^2$$
$$\leq \Big[\frac{d}{(1-\delta)nk + \delta k} - 1\Big]\|\mathbf{x}\|_2^2,$$

where the inequality is by Eq 21. □

C.2 Comparing against Rand-k

Next, we compare the MSE of Rand-Proj-Spatial(Max) with the MSE of the baseline Rand-k analytically in the full-correlation case. Recall that in this case,

$$MSE(\text{Rand-}k) = \frac{1}{n}(\frac{d}{k} - 1)\|\mathbf{x}\|_2^2.$$

We have

$$\begin{aligned}
&MSE(\text{Rand-Proj-Spatial(Max)}) \leq MSE(\text{Rand-}k) \\
\Leftrightarrow\ &\frac{d}{(1-\delta)nk + \delta k} - 1 \leq \frac{1}{n}(\frac{d}{k} - 1) \\
\Leftrightarrow\ &\frac{d}{k}\frac{n - (1-\delta)n - \delta}{n((1-\delta)n + \delta)} \leq 1 - \frac{1}{n} \\
\Leftrightarrow\ &\frac{d}{k} \cdot \frac{\delta - \delta/n}{(1-\delta)n + \delta} \leq \frac{n-1}{n} \\
\Leftrightarrow\ &d\delta(1 - \frac{1}{n})n \leq k(n-1) \cdot ((1-\delta)n + \delta) \\
\Leftrightarrow\ &d\delta \leq k \cdot ((1-\delta)n + \delta) \\
\Leftrightarrow\ &d\delta + kn\delta - k\delta \leq kn \\
\Leftrightarrow\ &\delta \leq \frac{kn}{d + kn - k} \\
\Leftrightarrow\ &\delta \leq \frac{1}{\frac{d}{kn} + 1 - \frac{1}{n}}
\end{aligned}$$

Since $nk \leq d$, for $n \geq 2$, the above implies when

$$\delta \leq \frac{1}{1 + \frac{1}{2}} = \frac{2}{3},$$

the MSE of Rand-Proj-Spatial(Max) is always less than that of Rand-k.

C.3 $\boldsymbol{S}$ has full rank with high probability

We empirically verify that $\delta \approx 0$. With $d \in \{32, 64, 128, \ldots, 1024\}$ and 4 different nk value such that $nk \leq d$ for each d, we compute rank($\boldsymbol{S}$) for 10^5 trials for each pair of (nk, d) values, and plot the results for all trials. All results are presented in Figure 7. As one can observe from the plots, rank($\boldsymbol{S}$) $= nk$ with high probability, suggesting $\delta \approx 0$.

This implies the MSE of Rand-Proj-Spatial(Max) is

$$MSE(\text{Rand-Proj-Spatial(Max)}) \approx (\frac{d}{nk} - 1)\|\mathbf{x}\|_2^2,$$

in the full correlation case.

C.4 Proof of Theorem 4.4

Theorem 4.4 (MSE under No Correlation)**.** *Consider n clients, each holding a vector $\mathbf{x}_i \in \mathbb{R}^d$, $\forall i \in [n]$. Suppose we set $T \equiv 1$, $\bar{\beta} = \frac{d^2}{k}$ in Eq. 5 and the random linear map $\boldsymbol{G}_i$ at each client to be*

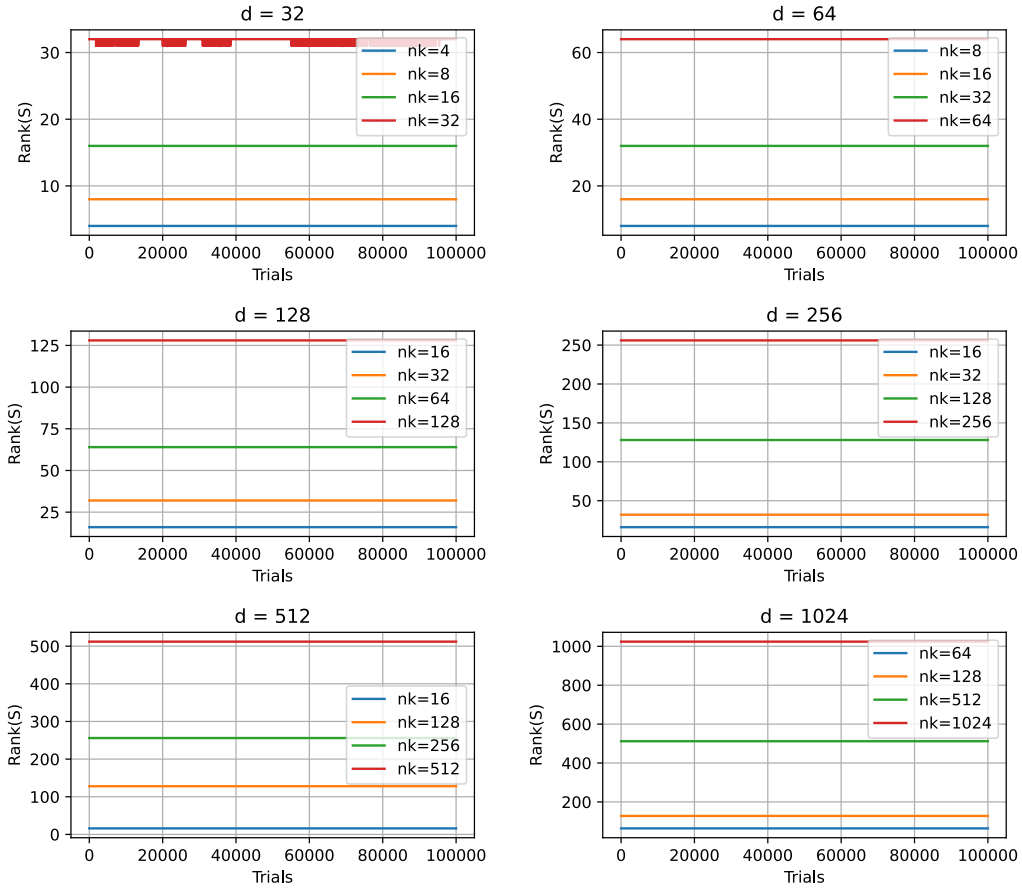

Figure 7: Simulation results of rank($\boldsymbol{S}$), where $\boldsymbol{S} = \sum_{i=1}^{n} \boldsymbol{G}_i^T \boldsymbol{G}_i$, with $\boldsymbol{G}_i$ being SRHT. With $d \in \{32, 64, 128, \ldots, 1024\}$ and 4 different nk values such that $nk \leq d$ for each d, we compute rank($\mathbf{S}$) for 10^5 trials for each pairs of (nk, d) values and plot the results for all trials. When $d = 32$ and $nk = 32$ in the first plot, rank($\boldsymbol{S}$) $= 31$ in 2100 trials, and rank($\boldsymbol{S}$) $= nk = 32$ in all the rest of the trials. For all other (nk, d) pairs, $\boldsymbol{S}$ always has rank nk in the 10^5 trials. This verifies that $\delta = \Pr[\text{rank}(\boldsymbol{S}) < nk] \approx 0$.

an SRHT matrix. Then, for $nk \leq d$,

$$\mathbb{E}\Big[\|\widehat{\mathbf{x}}^{(\textit{Rand-Proj-Spatial})} - \bar{\mathbf{x}}\|_2^2\Big] = \frac{1}{n^2}\Big(\frac{d}{k} - 1\Big)\sum_{i=1}^{n} \|\mathbf{x}_i\|_2^2.$$

Proof. When the client vectors are all orthogonal to each other, we define the transformation function on the eigenvalue to be $T(\lambda) = 1, \forall \lambda \geq 0$. We show that by considering the above constant T, SRHT becomes the same as rand k. Recall $\boldsymbol{S} = \sum_{i=1}^{n} \boldsymbol{G}_i^T \boldsymbol{G}_i$ and let $\boldsymbol{G}^T\boldsymbol{G} = \boldsymbol{U}\Lambda\boldsymbol{U}^T$ be its eigendecompostion. Then,

$$T(\boldsymbol{S}) = \boldsymbol{U}T(\Lambda)\boldsymbol{U}^T = \boldsymbol{U}\boldsymbol{I}_d\boldsymbol{U}^T = \boldsymbol{I}_d.$$

Hence, $(T(\boldsymbol{S}))^\dagger = \boldsymbol{I}_d$. And the decoded vector for client i becomes

$$\begin{aligned}
\widehat{\mathbf{x}}_i &= \bar{\beta}\Big(T(\boldsymbol{G}^T\boldsymbol{G})\Big)^\dagger \boldsymbol{G}_i^T\boldsymbol{G}_i\mathbf{x}_i = \bar{\beta}\boldsymbol{G}_i^T\boldsymbol{G}_i\mathbf{x}_i = \bar{\beta}\frac{1}{d}\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i, \\
\widehat{\mathbf{x}} &= \frac{1}{n}\sum_{i=1}^{n}\widehat{\mathbf{x}}_i = \frac{1}{n}\bar{\beta}\sum_{i=1}^{n}\frac{1}{d}\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i
\end{aligned} \tag{24}$$

$\boldsymbol{D}_i$ is a diagonal matrix. Also, $\boldsymbol{E}_i^T\boldsymbol{E}_i \in \mathbb{R}^{d\times d}$ is a diagonal matrix, where the i-th entry is 0 or 1.

Computing $\bar{\beta}$. To ensure that $\widehat{\mathbf{x}}$ is an unbiased estimator, from Eq. 24

$$\begin{aligned}
\mathbf{x}_i &= \bar{\beta}\mathbb{E}[\boldsymbol{G}_i^T\boldsymbol{G}_i]\mathbf{x}_i \\
&= \frac{\bar{\beta}}{d}\mathbb{E}[\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i]\mathbf{x}_i \\
&= \frac{\bar{\beta}}{d}\mathbb{E}_{\boldsymbol{D}_i}\Big[\boldsymbol{D}_i\boldsymbol{H}^T\underbrace{\mathbb{E}[\boldsymbol{E}_i^T\boldsymbol{E}_i]}_{=(k/d)\boldsymbol{I}_d}\boldsymbol{H}\boldsymbol{D}_i\Big]\mathbf{x}_i && (\because \boldsymbol{E}_i \text{ is independent of } \boldsymbol{D}_i) \\
&= \frac{\bar{\beta}}{d}k\mathbb{E}_{\boldsymbol{D}_i}\left[\boldsymbol{D}_i^2\right]\mathbf{x}_i && (\because \boldsymbol{H}^T\boldsymbol{H} = d\boldsymbol{I}_d) \\
&= \frac{\bar{\beta}k}{d}\mathbf{x}_i && (\because \boldsymbol{D}_i^2 = \boldsymbol{I} \text{ is now deterministic.}) \\
\Rightarrow \bar{\beta} &= \frac{d}{k}. && (25)
\end{aligned}$$

Computing the MSE.

$$\begin{aligned}
MSE &= \mathbb{E}\Big\|\widehat{\mathbf{x}} - \bar{\mathbf{x}}\Big\|_2^2 \\
&= \mathbb{E}\Big\|\frac{1}{n}\bar{\beta}\sum_{i=1}^n\frac{1}{d}\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i - \frac{1}{n}\sum_{i=1}^n\mathbf{x}_i\Big\|_2^2 \\
&= \frac{1}{n^2}\left\{\mathbb{E}\Big\|\bar{\beta}\sum_{i=1}^n\frac{1}{d}\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i\Big\|_2^2 + \Big\|\sum_{i=1}^n\mathbf{x}_i\Big\|_2^2\right. \\
&\qquad\left. -2\Big\langle\bar{\beta}\mathbb{E}[\sum_{i=1}^n\frac{1}{d}\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i], \sum_{i=1}^n\mathbf{x}_i\Big\rangle\right\} \\
&= \frac{1}{n^2}\left\{\bar{\beta}^2\mathbb{E}\Big\|\sum_{i=1}^n\frac{1}{d}\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i\Big\|_2^2 - \Big\|\sum_{i=1}^n\mathbf{x}_i\Big\|_2^2\right\} && (\because \mathbb{E}[\widehat{\mathbf{x}}] = \bar{\mathbf{x}}) \\
&= \frac{1}{n^2}\left\{\sum_{i=1}^n\frac{\bar{\beta}^2}{d^2}\mathbb{E}\Big\|\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i\Big\|_2^2 - \sum_{i=1}^n\Big\|\mathbf{x}_i\Big\|_2^2\right. && (26) \\
&\quad\left. +2\sum_{i=1}^n\sum_{l=i+1}^n\frac{\bar{\beta}^2}{d^2}\Big\langle\mathbb{E}[\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i], \mathbb{E}[\boldsymbol{D}_l\boldsymbol{H}^T\boldsymbol{E}_l^T\boldsymbol{E}_l\boldsymbol{H}\boldsymbol{D}_l\mathbf{x}_l]\Big\rangle - 2\sum_{i=1}^n\sum_{l=i+1}^n\Big\langle\mathbf{x}_i, \mathbf{x}_l\Big\rangle\right\}.
\end{aligned}$$

Note that in Eq. 26

$$\begin{aligned}
\mathbb{E}\Big\|\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i\Big\|_2^2 &= \mathbb{E}[\mathbf{x}_i^T\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i] \\
&= d\mathbb{E}[\mathbf{x}_i^T\boldsymbol{D}_i\boldsymbol{H}^T(\boldsymbol{E}_i^T\boldsymbol{E}_i)^2\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i] && (\because \boldsymbol{D}_i^2 = \boldsymbol{I}_d; \boldsymbol{H}^T\boldsymbol{H} = \boldsymbol{H}\boldsymbol{H}^T = d\boldsymbol{I}_d) \\
&= d\mathbf{x}_i^T\mathbb{E}_{\boldsymbol{D}_i}\left[\boldsymbol{D}_i\boldsymbol{H}^T\mathbb{E}[\boldsymbol{E}_i^T\boldsymbol{E}_i]\boldsymbol{H}\boldsymbol{D}_i\right]\mathbf{x}_i && (\boldsymbol{E}_i, \boldsymbol{D}_i \text{ are independent; } (\boldsymbol{E}_i^T\boldsymbol{E}_i)^2 = \boldsymbol{E}_i^T\boldsymbol{E}_i) \\
&= kd\|\mathbf{x}_i\|_2^2, && (27)
\end{aligned}$$

since $\mathbb{E}[\boldsymbol{E}_i^T\boldsymbol{E}_i] = (k/d)\boldsymbol{I}_d$, $\boldsymbol{H}^T\boldsymbol{H} = d\boldsymbol{I}_d$ and for $i \neq l$

$$\Big\langle\mathbb{E}[\boldsymbol{D}_i\boldsymbol{H}^T\boldsymbol{E}_i^T\boldsymbol{E}_i\boldsymbol{H}\boldsymbol{D}_i\mathbf{x}_i], \mathbb{E}[\boldsymbol{D}_l\boldsymbol{H}^T\boldsymbol{E}_l^T\boldsymbol{E}_l\boldsymbol{H}\boldsymbol{D}_l\mathbf{x}_l]\Big\rangle = \Big\langle k\mathbf{x}_i, k\mathbf{x}_l\Big\rangle = k^2\Big\langle\mathbf{x}_i, \mathbf{x}_l\Big\rangle. \qquad (28)$$

Substituting Eq. 27, 28 in Eq. 26, we get

$$\begin{aligned}
MSE &= \frac{1}{n^2}\left\{\Big(\frac{\bar{\beta}^2}{d^2}\sum_{i=1}^n kd\|\mathbf{x}_i\|_2^2 + 2\sum_{i=1}^n\sum_{l=i+1}^n\frac{\bar{\beta}^2k^2}{d^2}\Big\langle\mathbf{x}_i, \mathbf{x}_l\Big\rangle\Big) - \sum_{i=1}^n\Big\|\mathbf{x}_i\Big\|_2^2 - 2\sum_{i=1}^n\sum_{l=i+1}^n\Big\langle\mathbf{x}_i, \mathbf{x}_l\Big\rangle\right\} \\
&= \frac{1}{n^2}\Big(\frac{d}{k} - 1\Big)\sum_{i=1}^n\|\mathbf{x}_i\|_2^2,
\end{aligned}$$

which is exactly the same as the MSE of rand k. □

C.5 Rand-Proj-Spatial recovers Rand-k-Spatial (Proof of Lemma 4.1)

Lemma 4.1 (Recovering Rand-k-Spatial)**.** *Suppose client i generates a subsampling matrix $\boldsymbol{E}_i = [\mathbf{e}_{i_1}, \quad \ldots, \quad \mathbf{e}_{i_k}]^\top$, where $\{\mathbf{e}_j\}_{j=1}^d$ are the canonical basis vectors, and $\{i_1, \ldots, i_k\}$ are sampled from $\{1, \ldots, d\}$ without replacement. The encoded vectors are given as $\widehat{\mathbf{x}}_i = \boldsymbol{E}_i \mathbf{x}_i$. Given a function T, $\widehat{\mathbf{x}}$ computed as in Eq. 5 recovers the Rand-k-Spatial estimator.*

Proof. If client i applies $\boldsymbol{E}_i \in \mathbb{R}^{k \times d}$ as the random matrix to encode $\mathbf{x}_i$ in Rand-Proj-Spatial, by Eq. 5, client i's encoded vector is now

$$\hat{\mathbf{x}}_i^{\text{(Rand-Proj-Spatial)}} = \bar{\beta}\Big(T(\sum_{i=1}^n \boldsymbol{E}_i^T \boldsymbol{E}_i)\Big)^\dagger \boldsymbol{E}_i^T \boldsymbol{E}_i \mathbf{x}_i \tag{29}$$

Notice $\boldsymbol{E}_i^T \boldsymbol{E}_i$ is a diagonal matrix, where the j-th diagonal entry is 1 if coordinate j of $\mathbf{x}_i$ is chosen. Hence, $\boldsymbol{E}_i^T \boldsymbol{E}_i \mathbf{x}_i$ can be viewed as choosing k coordinates of $\mathbf{x}_i$ without replacement, which is exactly the same as Rand-k-Spatial's (and Rand-k's) encoding procedure.

Notice $\sum_{i=1}^n \boldsymbol{E}_i^T \boldsymbol{E}_i$ is also a diagonal matrix, where the j-th diagonal entry is exactly M_j, i.e. the number of clients who selects the j-th coordinate as in Rand-k-Spatial [12]. Furthermore, notice $\Big(T(\sum_{i=1}^n \boldsymbol{E}_i^T \boldsymbol{E}_i)\Big)^\dagger$ is also a diagonal matrix, where the j-th diagonal entry is $\frac{1}{T(M_j)}$, which recovers the scaling factor used in Rand-k-Spatial's decoding procedure.

Rand-Proj-Spatial computes $\bar{\beta}$ as $\bar{\beta}\mathbb{E}\Big[\Big(T(\sum_{i=1}^n \boldsymbol{E}_i^T \boldsymbol{E}_i)\Big)^\dagger \boldsymbol{E}_i^T \boldsymbol{E}_i \mathbf{x}_i\Big] = \mathbf{x}_i$. Since $\Big(T(\sum_{i=1}^n \boldsymbol{E}_i^T \boldsymbol{E}_i)\Big)^\dagger$ and $\boldsymbol{E}_i^T \boldsymbol{E}_i \mathbf{x}_i$ recover the scaling factor and the encoding procedure of Rand-k-Spatial, and $\bar{\beta}$ is computed in exactly the same way as Rand-k-Spatial does, $\bar{\beta}$ will be exactly the same as in Rand-k-Spatial.

Therefore, $\hat{\mathbf{x}}_i^{\text{(Rand-Proj-Spatial)}}$ in Eq. 29 with $\boldsymbol{E}_i$ as the random matrix at client i recovers $\hat{\mathbf{x}}_i^{\text{(Rand-}k\text{-Spatial)}}$. This implies Rand-Proj-Spatial recovers Rand-k-Spatial in this case. □

D Additional Experiment Details and Results

Implementation. All experiments are conducted in a cluster of 20 machines, each of which has 40 cores. The implementation is in `Python`, mainly based on `numpy` and `scipy`. All code used for the experiments can be found at `https://github.com/11hifish/Rand-Proj-Spatial`.

Data Split. For the non-IID dataset split across the clients, we follow [62] to split `Fashion-MNIST`, which is used in distributed power iteration and distributed k-means. Specifically, the data is first sorted by labels and then divided into $2n$ shards with each shard corresponding to the data of a particular label. Each client is then assigned 2 shards (i.e., data from 2 classes). However, this approach only works for datasets with discrete labels (i.e. datasets used in classification tasks). For the other dataset `UJIndoor`, which is used in distributed linear regression, we first sort the dataset by the ground truth prediction and then divides the sorted dataset across the clients.

D.1 Additional experimental results

For each one of the three tasks, distributed power iteration, distributed k-means, and distributed linear regression, we provide additional results when the data split is IID across the clients for smaller n, k values in Section D.1.1, and when the data split is Non-IID across the clients in Section D.1.2. For the Non-IID case, we use the same settings (i.e. n, k, d values) as in the IID case.

Discussion. For smaller n, k values compared to the data dimension d, there is less information or less correlation from the client vectors. Hence, both Rand-k-Spatial and Rand-Proj-Spatial perform better as nk increases. When n, k is small, one might notice Rand-Proj-Spatial performs worse than Rand-k-Wangni in some settings. However, Rand-k-Wangni is an *adaptive* estimator, which optimizes the sampling weights for choosing the client vector coordinates through an iterative process. That means Rand-k-Wangni requires more computation from the clients, while in practice, the clients often have limited computational power. In contrast, our Rand-Proj-Spatial estimator is *non-adaptive* and the server does more computation instead of the clients. This is more practical since the central server usually has more computational power than the clients in applications like FL. See the introduction for more discussion.

In most settings, we observe the proposed Rand-Proj-Spatial has a better performance compared to Rand-k-Spatial. Furthermore, as one would expect, both Rand-k-Spatial and Rand-Proj-Spatial perform better when the data split is IID across the clients since there is more correlation among the client vectors in the IID case than in the Non-IID case.

D.1.1 More results in the IID case

Distributed Power Iteration and Distribued K-Means. We use the `Fashion-MNIST` dataset for both distributed power iteration and distributed k-means, which has a dimension of $d = 1024$. We consider more settings for distributed power iteration and distributed k-means here: $n = 10, k \in \{5, 25, 51\}$, and $n = 50, k \in \{5, 10\}$.

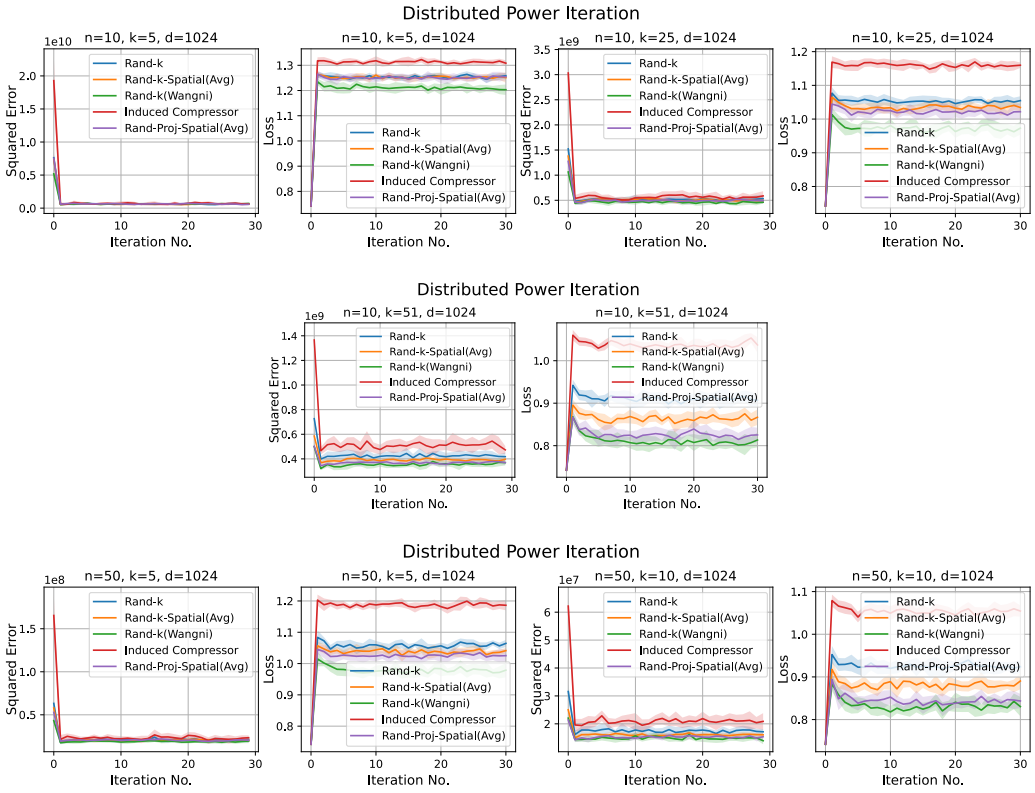

Figure 8: More results of distributed power iteration on `Fashion-MNIST` (IID data split) with $d = 1024$ when $n = 10$, $k \in \{5, 25, 51\}$ and when $n = 50$, $k \in \{5, 10\}$.

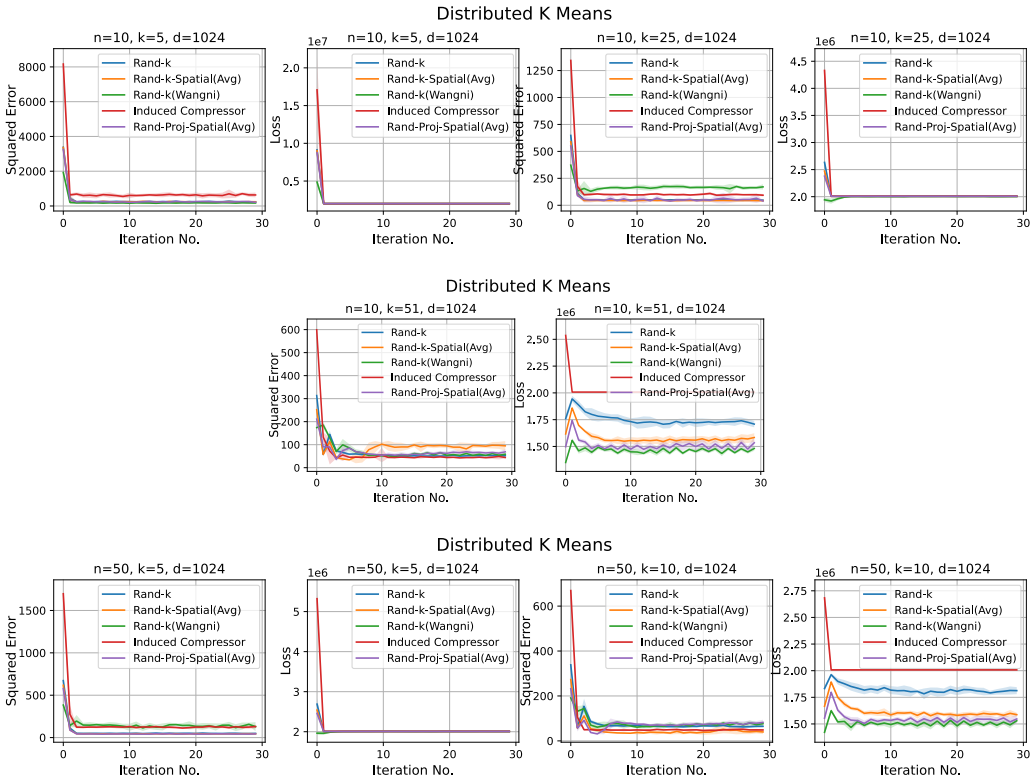

Figure 9: More results on distributed k-means on `Fashion-MNIST` (IID data split) with $d = 1024$ when $n = 10, k \in \{5, 25, 51\}$ and when $n = 50, k \in \{10, 51\}$.

Distributed Linear Regression. We use the `UJIndoor` dataset distributed linear regression, which has a dimension of $d = 512$. We consider more settings here: $n = 10, k \in \{5, 25\}$ and $n = 50, k \in \{1, 5\}$.

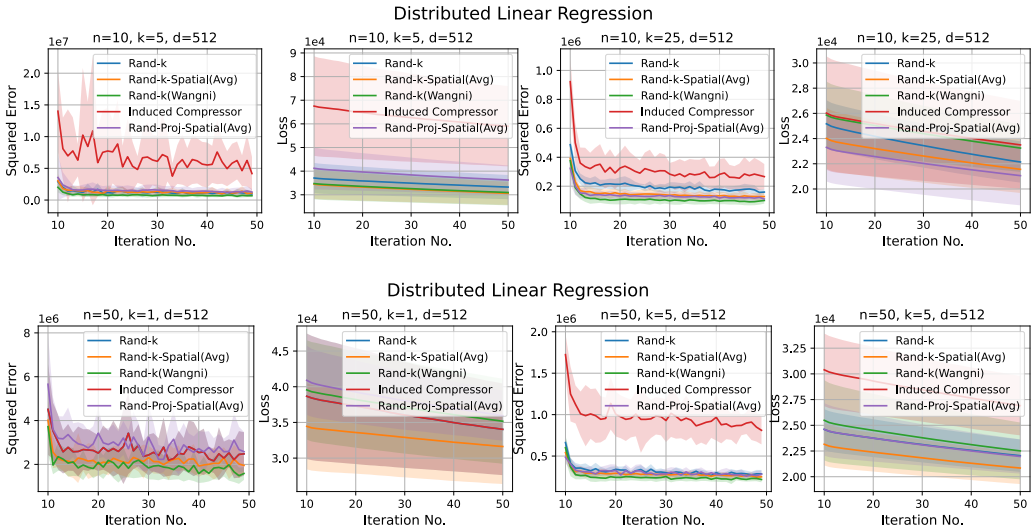

Figure 10: More results of distributed linear regression on `UJIndoor` (IID data split) with $d = 512$, when $n = 10, k \in \{5, 25\}$ and when $n = 50, k \in \{1, 5\}$. Note when $k = 1$, the Induced estimator is the same as Rand-k.

D.1.2 Additional results in the Non-IID case

In this section, we report results when the dataset split across the clients are Non-IID, using the same datasets as in the IID case. We choose exactly the same set of n, k values as in the IID case.

Distributed Power Iteration and Distributed K-Means. Again, both distributed power iteration and distributed k-means use the `Fashion-MNIST` dataset, with a dimension $d = 1024$. We consider the following settings for both tasks: $n = 10, k \in \{5, 25, 51, 102\}$ and $n = 50, k \in \{5, 10, 20\}$.

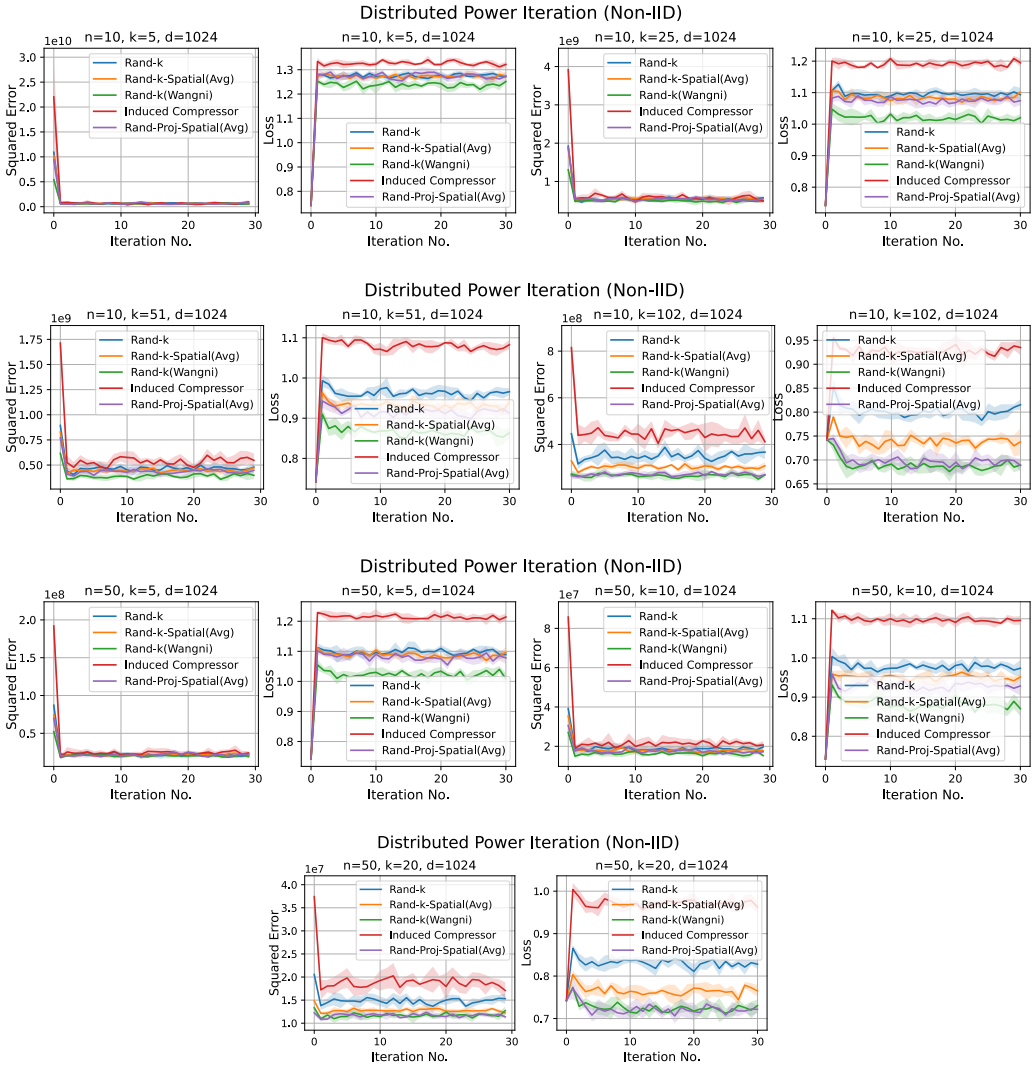

Figure 11: Results of distributed power iteration when the data split is Non-IID. $n = 10, k \in \{5, 25, 51, 102\}$ and $n = 50, k \in \{5, 10, 20\}$.

Distributed Linear Regression. Again, we use the `UJIndoor` dataset for distributed linear regression, which has a dimension $d = 512$. We consider the following settings: $n = 10, k \in \{5, 25, 50\}$ and $n = 50, k \in \{1, 5, 50\}$.

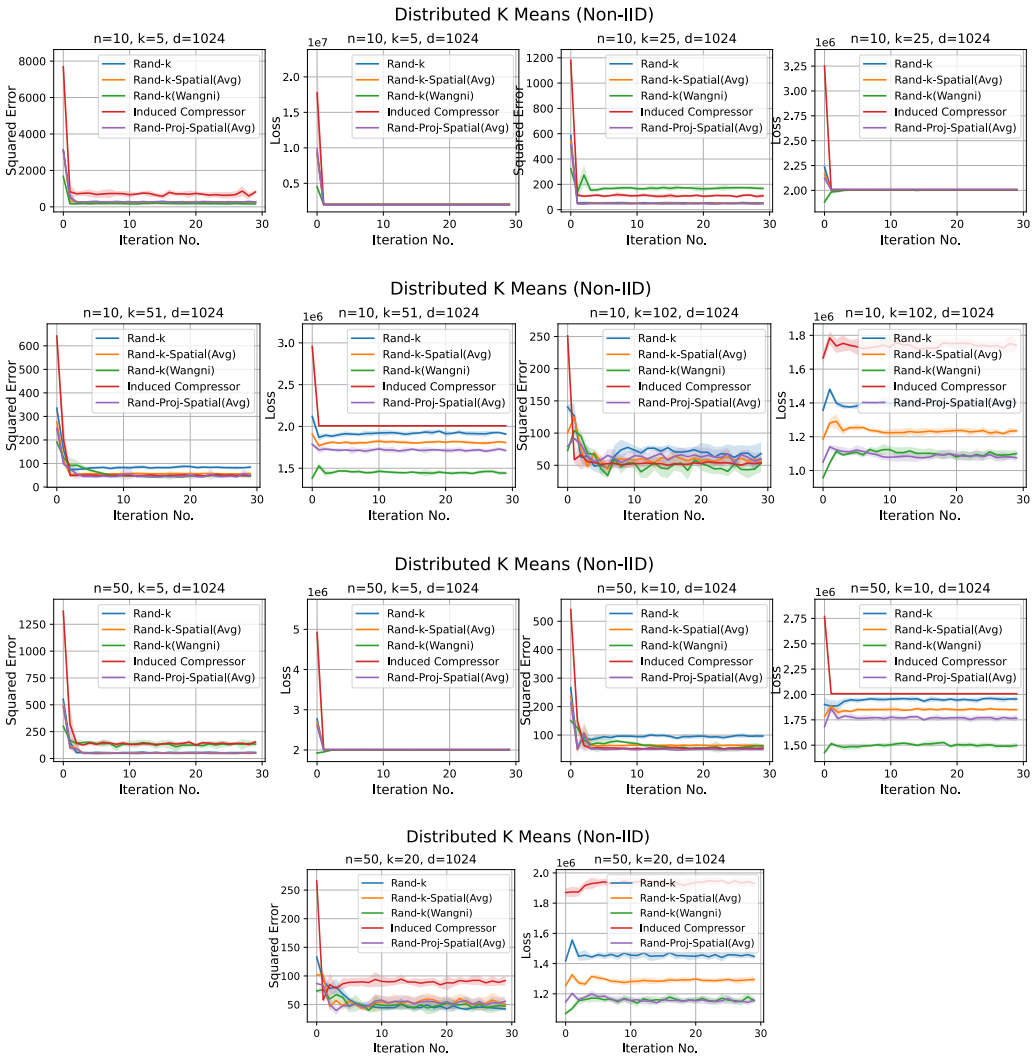

Figure 12: Results of distributed k-means when the data split is Non-IID. $n = 10, k \in \{5, 25, 51, 102\}$ and $n = 50, k \in \{5, 10, 20\}$.

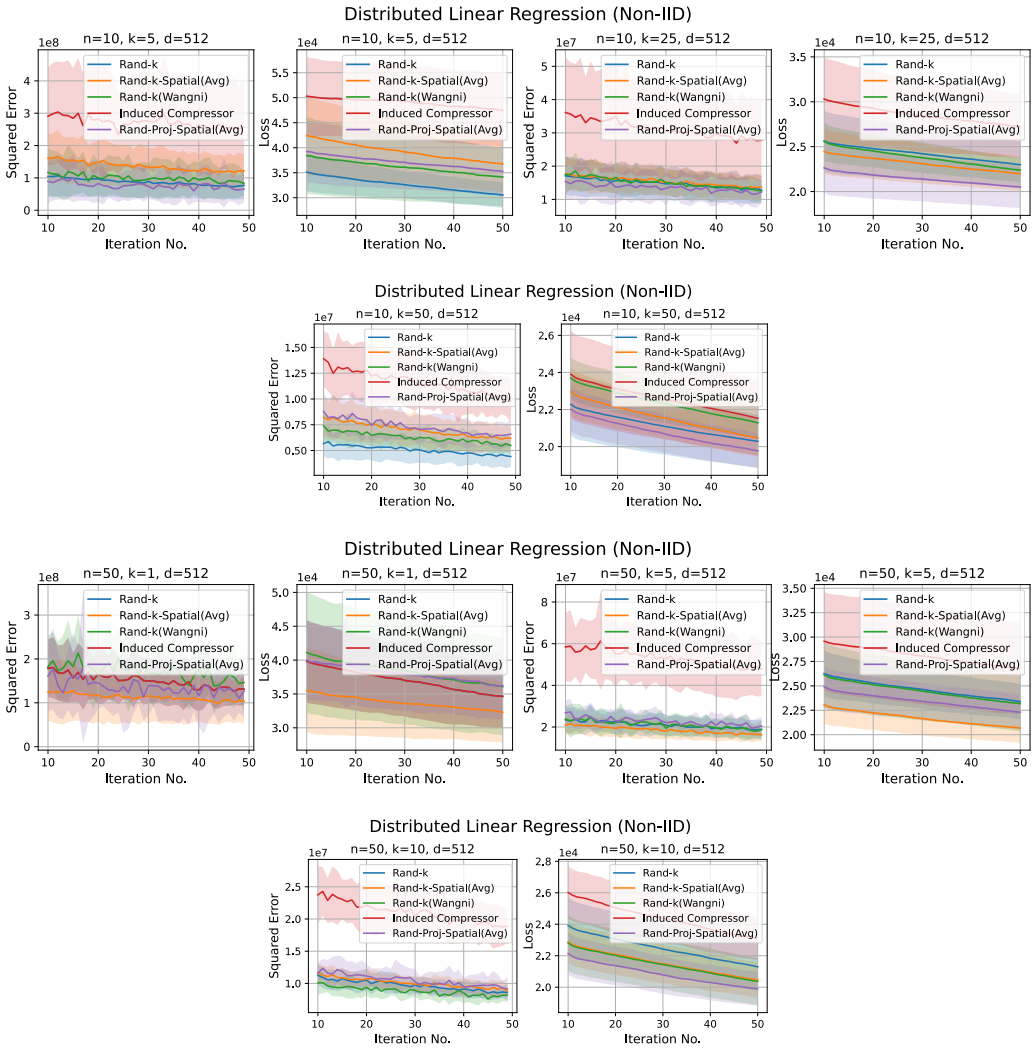

Figure 13: Results of distributed linear regression when the data split is Non-IID. $n = 10, k \in \{5, 25, 50\}$ and $n = 50, k \in \{1, 5, 50\}$.