# Tensor Random Projection for Low Memory Dimension Reduction

August 24, 2019

**Abstract**

andom projections reduce the dimension of data in form of vectors or matrix while preserving structural information, such as distances between vectors in the set. This paper proposes a novel use of row-product random matrices Rudelson [2012] in random projections which we call Tensor Random Projection (TRP). It requires substantially less memory than existing dimension reduction maps. The TRP map, formed as the Khatri-Rao product of several smaller random projections, is compatible with any basic random projection including sparse maps, which enable dimension reduction to realize at very low query costs without performing floating point operations. Theories of application of vector dimension reduction, matrix sketching, tensor regression have been built and tested through comprehensive simulation studies. We also experiment our methods in real scenarios with computer vision data.

## 1 Introduction

Linear random projection is operating a random matrix onto the data which could be either high dimension vector or matrix to reduce the dimension while preserving the useful information residing in the data. Then a linear random projection can be represented as a random matrix $\mathbf{\Omega} \in \mathbb{R}^{d\times k}$, operating on a vector $\mathbf{x} \in \mathbb{R}^d$ or a matrix $\mathbf{X} \in \mathbb{R}^{m\times d}$ to reduce the dimension:

$$\begin{aligned} &\mathbf{x} \in^n \rightarrow \mathbf{\Omega}^\top \mathbf{x} \in \mathbb{R}^k \\ &\mathbf{X} \in \mathbb{R}^{m\times d} \rightarrow \mathbf{X}\mathbf{\Omega} \in \mathbb{R}^{m\times k}. \end{aligned} \tag{1.1}$$

Random projection application to vector has a very long history which enables a broad range of modern applications from bio-informatics, informational retrieval to computer vision like Wright et al. [2009], Buhler and Tompa [2002], Allen-Zhu et al. [2014], Bingham and Mannila [2001], Fradkin and Madigan [2003], Halko et al. [2011], Wang et al. [2012], Jegou et al. [2008]. In the context of large-scale relational databases, these maps enable applications like information retrieval Papadimitriou et al. [2000], similarity search Sahin et al. [2005], Kaski [1998], and privacy preserving distributed data mining Liu et al. [2006]. Later, along with

the fast development in randomized algorithm, linear random projections are widely employed in constructing fast randomized algorithms in fields like matrix and tensor decomposition Woolfe et al. [2008], Tropp et al. [2017], optimization Yurtsever et al. [2017], streaming data compression Tropp et al. [2019], Sun et al. [2019]. The much smaller matrix after random projection in the second line in (1.1) is named *sketch*. The term 'sketch' describes the fact that the matrix after random projection captures most of the action of the original matrix.

The effectiveness of random projection is measured by whether the information inside data is well preserved in the low dimensional embedding after random projection. For the case where we operates random matrix $\mathbf{\Omega}$ onto vector $\mathbf{x}$, we require

$$\|\mathbf{\Omega}^\top \mathbf{x}_1 - \mathbf{\Omega}^\top \mathbf{x}_2\| \approx \|\mathbf{x}_1 - \mathbf{x}_2\|. \tag{1.2}$$

For the case, where we operating $\mathbf{\Omega}$ onto matrix $\mathbf{X}$, it requires that

$$\|\mathbf{Q}\mathbf{Q}^\top \mathbf{X} - \mathbf{X}\| \text{ is small,} \tag{1.3}$$

where $\mathbf{Q}$ is the ortho-normal matrix got from QR factorization from $\mathbf{X\Omega}$. For the case where $\mathbf{\Omega}$ has i.i.d. elements, there are many literature in how these two properties are preserved. We list two well known results in literature for those two cases separately.

**Lemma 1.1** (Arriaga and Vempala [2006])**.** *Let $\mathbf{x} \in \mathbb{R}^d$, assume that the entries in $\mathbf{\Omega} \in \mathbb{R}^{d\times k}$ are sampled independently from $\mathcal{N}(0,1)$. Then*

$$\mathbf{Prob}\left((1-\epsilon)\|\mathbf{x}\|^2 \leqslant \left\|\frac{1}{\sqrt{k}}\mathbf{\Omega}^\top \mathbf{x}\right\| \leqslant (1+\epsilon)\|\mathbf{x}\|^2\right) \leqslant 1 - 2e^{-(\epsilon^2-\epsilon^3)k/4}. \tag{1.4}$$

**Lemma 1.2** (Halko et al. [2011])**.** *Let $\mathbf{X} \in \mathbb{R}^{m\times d}$, assume that the entries in $\mathbf{\Omega} \in \mathbb{R}^{d\times(k+p)}$ are sampled independently from $\mathcal{N}(0,1)$. Then let $\mathbf{Q}$ be the orthonormal matrix from QR factorization $\mathbf{X\Omega} = \mathbf{QR}$, then*

$$\mathbf{E}\,\|\mathbf{X} - \mathbf{Q}\mathbf{Q}^\top \mathbf{X}\|_F \leqslant \left(1 + \frac{k}{p-1}\right)^{1/2} \left(\sum_{j>k} \sigma_j^2\right)^{1/2}. \tag{1.5}$$

**Memory Efficient Random Projection** Sparse random maps for low memory dimension reduction were first proposed by Achlioptas [2003], and further work has improved the memory requirements and guarantees of these methods Li et al. [2006], Ailon and Chazelle [2006], Bourgain et al. [2015]. Usually they propose the random projection to be sparse. But under modern 'big data' setting, their cost in storage/memory cost is still to big to be practical. Consider

Most closely related to our work is Rudelson's foundational study Rudelson [2012], which considers how the spectral and geometric properties of the random maps we use in this paper resemble a random map with iid entries, and shows that their largest and smallest singular values are of the same order. These results have been widely used to obtain guarantees for algorithmic privacy, but not for random projection. Battaglino et al. Battaglino et al. [2018] use random projections

of Khatri-Rao products to develop a randomized least squares algorithm for tensor factorization; in contrast, our method uses the (full) Khatri-Rao product to enable random projection. Sparse random projections to solve least squares problems were also explored in Wang et al. [2015] and Woodruff et al. [2014]. To our knowledge, this paper is the first to consider using the Khatri-Rao product for low memory random projection. However, if the dimension of vectors before reduction (here, the size of the lexicon) is too big, the storage cost of the random map is not negligible. Furthermore, even generating the pseudo-random numbers used to produce the random projection is expensive Matsumoto and Nishimura [1998].

To reduce the storage burden, we propose a novel use of the row-product random matrices in random projection, and call it the *Tensor Random Projection* (TRP), formed as the Khatri-Rao product of a list of smaller dimension reduction maps. We show this map is an approximate isometry, with tunable accuracy, and hence can serve as a useful dimension reduction primitive. Furthermore, the storage required to compress $d$ dimension vectors scales as $\sqrt[N]{d}$ where $N$ is the number of smaller maps used to form the TRP. We also develop a reduced variance version of the TRP that allows separate control of the dimension of the range and the quality of the isometry.

### 1.1 Notation

We denote *scalar*, *vector*, and *matrix* variables, respectively, by lowercase letters ($x$), boldface lowercase letters ($\mathbf{x}$), and boldface capital letters ($\mathbf{X}$). Let $[N] = \{1, \ldots, N\}$. For matrix $\mathbf{X}$, we denote its $i^{th}$ row, $j^{th}$ column, and the $(i,j)^{th}$ element as $\mathbf{X}_{(i,.)}, \mathbf{X}_{(.,j)}, \mathbf{X}_{(i,j)}$. The *Kronecker product* of two matrices $\mathbf{A} \in \mathbb{R}^{m\times n}, \mathbf{B} \in \mathbb{R}^{p\times q}$, denoted as $\mathbf{A} \otimes \mathbf{B} \in \mathbb{R}^{mp\times nq}$, is defined as

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} A_{11}\mathbf{B} & \cdots & A_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ A_{m1}\mathbf{B} & \cdots & A_{mn}\mathbf{B} \end{bmatrix}.$$

We let $\mathbf{X} \odot \mathbf{Y}$ denotes the *Khatri-Rao product*, $\mathbf{A} \in \mathbb{R}^{I\times K}, \mathbf{B} \in \mathbb{R}^{J\times K}$, i.e. the "matching column-wise" Kronecker product. The resulting matrix of size $(IJ) \times K$ is given by:

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{A}_{(1,.)} \otimes \mathbf{B}_{(1,.)}, \ldots, \mathbf{A}_{(K,.)} \otimes \mathbf{B}_{(K,.)}]. \tag{1.6}$$

## 2 Tensor Random Projection

We seek a random projection map to embed a collection of vectors $\mathcal{X} \subseteq \mathbb{R}^d$ into $\mathbb{R}^k$ with $k \ll d$. Let us take $d = \prod_{n=1}^{N} d_n$, motivated by the problem of compressing (the vectorization of) an order $N$ tensor with dimensions $d_1, \ldots, d_N$. Conventional random projections use $O(kd)$ random variables. Generating so many random numbers is costly; and storing them can be costly when $d$ is large. Is so much randomness truly necessary for a random projection map?

To reduce randomness and storage requirements, we propose the *tensor random projection* (TRP):

$$f_{\text{TRP}}(\mathbf{x}) := (\mathbf{A}_1 \odot \cdots \odot \mathbf{A}_N)^\top \mathbf{x}, \tag{2.1}$$

where each $\mathbf{A}_i \in \mathbb{R}^{d_i \times k}$, for $i \in [N]$, can be an arbitrary RP map and $\mathbf{A} := (\mathbf{A}_1 \odot \cdots \odot \mathbf{A}_N)^\top$. We call $N$ the *order* of the TRP. We show in this paper that the TRP is an expected isometry, has vanishing variance, and supports database-friendly operations.

The TRP requires only $k \sum_{i=1}^{N} d_i$ random variables (or $k \sqrt[N]{d}$ by choosing each $d_i$ to be equal), rather than the $kd$ random variables needed by conventional methods. Hence the TRP is database friendly: it significantly reduces storage costs and randomness requirements compared to its constituent DRMs.

In large scale database settings, where computational efficiency is critical and queries of vector elements are costly, practitioners often use sparse RPs. Let $\delta$ be the proportion of non-zero elements in the RP map. To achieve a $\delta$-sparse RP, a common construction is the scaled sign random map: each element is distributed as $(-1/\sqrt{\delta}, 0, 1/\sqrt{\delta})$ with probability $(\delta/2, 1-\delta, \delta/2)$. Achlioptas [2003] proposed $\delta = 1/3$, while Li et al. [2006] further suggests a sparser scheme with $\delta = 1/\sqrt{d}$ that he calls the *Very Sparse* RP.

To further reduce memory requirements of random projection, we can form a TRP whose constituent submatrices are generated each with sparsity factor $\delta$, which leads to a $\delta^N$-sparse TRP. Under sparse setting, it is a $(1/3)^N$ sparse TRP while under very sparse setting, it is a $1/\sqrt{d}$ sparse TRP. Both TRPs can be applied to a vector using very few queries to vector elements and no multiplications. Below, we show both sparse and very sparse TRP are low-variance approximate isometry empirically.

# 3 Main Theory

In this section, we discuss the properties of tensor random projection with application to length preservation and column space preservation.

## 3.1 Bias and Variance

In this section, we will show the TRP and $\text{TRP}_T$ are expected isometries with vanishing variance. We provide a rate for the decrease in variance with $k$. We also prove a non-asymptotic concentration bound on the quality of the isometry when $N = 2$. We begin by showing the TRP is an approximate isometry.

**Theorem 3.1.** *Fix $\mathbf{x} \in \mathbb{R}^{\prod_{n=1}^{N} d_n}$. Form a TRP and $\text{TRP}_T$ of order $N$ with range $k$ composed of independent matrices with independent columns whose entries are mean zero, variance one, and within each column every pair of elements has covariance zero. Then*

$$\mathbb{E}\,\|\text{TRP}(\mathbf{x})\|^2 = \|\mathbf{x}\|^2 \qquad \textit{and} \qquad \mathbb{E}\|\text{TRP}_T(\mathbf{x})\|^2 = \|\mathbf{x}\|^2.$$

Interestingly, Theorem Theorem 3.1 does not require elements of $\mathbf{A}_n$ to be iid. Now we present an explicit form for the variance of the isometry.

**Theorem 3.2.** *Fix $\mathbf{x} \in \mathbb{R}^{\prod_{n=1}^{N} d_n}$. Form a* TRP *and* $\text{TRP}_T$ *of order $N$ with range $k$ independent matrices whose entries are i.i.d. with mean zero, variance one, and fourth moment $\Delta$. Then*

$$\begin{aligned}\mathbf{var}(\|\text{TRP}(\mathbf{x})\|^2) &= \frac{1}{k}(\Delta^N - 3)\|\mathbf{x}\|_4^4 + \frac{2}{k}\|\mathbf{x}\|_2^4 \\ \mathbf{var}(\|\text{TRP}_T(\mathbf{x})\|^2) &= \frac{1}{Tk}(\Delta^N - 3)\|\mathbf{x}\|_4^4 + \frac{2}{k}\|\mathbf{x}\|_2^4.\end{aligned}$$

We can see the variance increases with $N$. In the $N = 1$ Gaussian case, this formula shows a variance of $2/k\|\mathbf{x}\|_2^4$, which agrees with the classic result. Notice the $\text{TRP}_T$ only reduces the first term in the variance bound: as $T \to \infty$, the variance converges to that of a Gaussian random map.

Next, since $\text{TRP}_T$ is a linear operator, treat $\mathbf{x} - \mathbf{y}$ as a vector, with above argument, we have the following lemma for pair-wise distance. Proof is omitted for the sake of brevity.

**Corollary 3.3.** *Fix $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{\prod_{n=1}^{N} d_n}$. Form a $\text{TRP}_T$ of order $N$ with range $k$ independent matrices whose entries are i.i.d with mean zero, variance one, and fourth moment $\Delta$. We have*

$$\begin{aligned}&\mathbb{E}(\|\text{TRP}_T(\mathbf{x}) - \text{TRP}_T(\mathbf{y})\|^2) = \|\mathbf{x} - \mathbf{y}\|^2, \\ &\mathbf{var}(\|\text{TRP}_T(\mathbf{x}) - \text{TRP}_T(\mathbf{y})\|^2) = \frac{1}{Tk}(\Delta^N - 3)\|\mathbf{x} - \mathbf{y}\|_4^4 + \frac{2}{k}\|\mathbf{x} - \mathbf{y}\|_2^4.\end{aligned} \tag{3.1}$$

For completeness, we also present the analysis for bias and variance for inner product.

**Lemma 3.4.** *Fix $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{\prod_{n=1}^{N} d_n}$. For* TRP *and* $\text{TRP}_T$ *of order $N$ with range $k$ independent matrices whose entries are i.i.d with mean zero, variance one, and fourth moment $\Delta$, we have*

$$\begin{aligned}\mathbb{E}(\langle \text{TRP}(\mathbf{x}), \text{TRP}(\mathbf{y})\rangle) &= \mathbb{E}(\langle \text{TRP}_T(\mathbf{x}), \text{TRP}_T(\mathbf{y})\rangle) = \langle \mathbf{x}, \mathbf{y}\rangle \\ \mathbf{var}(\langle \text{TRP}(\mathbf{x}), \text{TRP}(\mathbf{y})\rangle) &= \frac{1}{k}[(\Delta^N - 3)\sum_{\mathbf{r}} x_{\mathbf{r}}^2 y_{\mathbf{r}}^2 + \|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2 + \langle \mathbf{x}, \mathbf{y}\rangle^2]. \\ \mathbf{var}(\langle \text{TRP}_T(\mathbf{x}), \text{TRP}_T(\mathbf{y})\rangle) &= \frac{1}{kT}(\Delta^N - 3)\sum_{\mathbf{r}} x_{\mathbf{r}}^2 y_{\mathbf{r}}^2 + (\frac{2}{k} - \frac{1}{kT})\|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2 + \frac{1}{kT}\langle \mathbf{x}, \mathbf{y}\rangle^2.\end{aligned} \tag{3.2}$$

We can see as $T \to \infty$, $\mathbf{var}(\langle \text{TRP}_T(\mathbf{x}), \text{TRP}_T(\mathbf{y})\rangle) \to \frac{2}{k}\|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2$, same as the variance in the Gaussian Random map case.

**Remark.** *If we further assume each entry of $\mathbf{x}, \mathbf{y}$ to be a random variable with their second and fourth moment bounded by constants. We can see as $d \to \infty$, $\|\mathbf{x}\|_2^4$, $\|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2$, $\langle \mathbf{x}, \mathbf{y}\rangle^2$ are $\mathcal{O}(d^2)$, and $\|\mathbf{x}\|_4^4$, $\sum_{\mathbf{r}} x_{\mathbf{r}}^2 y_{\mathbf{r}}^2$ are $\mathcal{O}(d)$ respectively. Thus, $\frac{2}{k}\|\mathbf{x}\|_2^4$, i.e. the term same as in the Gaussian RP, dominates $\mathbf{var}(\|\text{TRP}(\mathbf{x})\|^2)$, and $\frac{1}{k}(\|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2 + \langle \mathbf{x}, \mathbf{y}\rangle^2)$ dominates $\mathbf{var}(\langle \text{TRP}(\mathbf{x}), \text{TRP}(\mathbf{y})\rangle)$.*

## 3.2 Asymptotic Behavior

## 3.3 Finite Sample Bound?

Finally we show a non-asymptotic concentration bound for $N = 2$. We leave the parallel result for $N \geqslant 3$ open for future exploration.

**Proposition 3.5.** *Fix $\mathbf{x} \in \mathbb{R}^{d_1 d_2}$ with sub-Gaussian norm $\varphi_2$. Form a* TRP*(T) of order 2 with range $k$ composed of two independent matrices whose entries are drawn i.i.d. from a sub-Gaussian distribution with mean zero and variance one. Then there exists a constant $C$ depending on $\varphi_2$ and a universal constant $c_1$ so that*

$$\mathbb{P}\left(\left|\|f_{\mathrm{TRP}}(\mathbf{x})\|^2 - \|\mathbf{x}\|_2^2\right| \geqslant \epsilon\|x\|^2\right) \leqslant C\exp\left[-c_1\left(\sqrt{k}\epsilon\right)^{1/4}\right],$$

Here $\varphi_2$ is the sub-Gaussian norm defined in D.1 in Appendix A. 3.5 shows that for a TRP to form an $\epsilon$-JL DRM with substantial probability on a dataset with $n$ points, our method requires $k = \mathcal{O}(\epsilon^{-2}\log^8 n)$ while conventional random projections require $k = \mathcal{O}(\epsilon^{-2}\log n)$. Numerical experiments suggest this bound is pessimistic.

## 3.4 Column Space Preservation

(1.5) in Lemma 1.2 shows that the random projection preserve the information in column space of a matrix well and provide the error bound compared with the tail energy. It is hard to derive similar result for general matrix with tensor random projection. But if the matrix is in form of kroneck product, we can get a similar result based as following proposition:

**Proposition 3.6.** *Let $\mathbf{X}_n \in \mathbb{R}^{m_i \times d_n}$ be a series of matrix and $\mathbf{\Omega}_n \in \mathbb{R}^{d_n \times (k+p_n)}$ with eact element sampled from standard Gaussian distribution, let $\tau_n(k) = \sum_{j>k}\sigma_j^2((x))$ be the tail energy for $\mathbf{X}_i$. Let $\mathbf{Q} \in \mathbb{R}^{d\times k}$ be the orthonormal matrix from QR factorization:*

$$\mathbf{Q}, - = \mathrm{QR}[(\mathbf{X}_1 \otimes \cdots \otimes \mathbf{X}_{\mathrm{N}})(\mathbf{\Omega}_1 \odot \cdots \odot \mathbf{\Omega}_{\mathrm{N}})]$$

*we have*

$$\begin{aligned}&\|(\mathbf{X}_1 \otimes \cdots \otimes \mathbf{X}_N) - \mathbf{Q}\mathbf{Q}^\top(\mathbf{X}_1 \otimes \cdots \otimes \mathbf{X}_N)\|_F \\ &\leqslant \prod_{i=1}^{N}\sqrt{\left(1 + \frac{k}{p_n - 1}\right)\tau_n(k)}.\end{aligned} \tag{3.3}$$

*Proof.* Schäcke [2013] has a detailed description on property for kroneck product. Following those properties, we have

$$\begin{aligned}&(\mathbf{X}_1 \otimes \cdots \otimes \mathbf{X}_N)(\mathbf{\Omega_1} \odot \cdots \odot \mathbf{\Omega_N}) \\ &= (\mathbf{X}_1\mathbf{\Omega_1} \otimes \cdots \otimes \mathbf{X}_N\mathbf{\Omega_N})\end{aligned} \tag{3.4}$$

Let $\mathbf{Q}_n \in \mathbb{R}^{d_n \times k}$ be the orthonormal matrix from QR factorization: $\mathbf{Q}_i, \mathrm{QR}(\mathbf{X_i}\mathbf{\Omega_i})$. The key observation is that $\mathbf{Q} = \mathbf{Q}_1 \otimes \cdots \otimes \mathbf{Q}_N$ then we finish the proof with

some association rule for kroneck product in Schäcke [2013] and result in Lemma 1.2. Also this proposition, indicate in practice, we should sketch each small matrix $\mathbf{X}_n$ then combine them together which is equivalent to do sketch the whole matrix after kroneck product.

□

# 4 Experiment

In this section, we compare the quality of the isometry of conventional RPs, TRP, and TRP(5), for Gaussian, Sparse Achlioptas [2003], and Very Sparse random maps Li et al. [2006] on both synthetic data and MNIST data. We also use TRP and TRP(5) to compute pairwise cosine similarity (Table 1 and Appendix B) and to sketch matrices and tensors (Appendix 5), although the theory still remains open.

Our first experiment evaluates the quality of the isometry for maps $\mathbb{R}^d \to \mathbb{R}^k$. We generate $n = 10$ independent vectors $\mathbf{x}_1, \ldots, \mathbf{x}_n$ of sizes $d = 2500, 10000, 40000$ from $\mathcal{N}(\mathbf{0}, \mathbf{I})$. We consider the following three RPs: 1. Gaussian RP; 2. Sparse RP Achlioptas [2003]; 3. Very Sparse RP Li et al. [2006]. For each, we compare the performance of RP, TRP, and TRP(5) with order 2 and $d_1 = d_2$. We evaluate the methods by repeatedly generating a RP and computing the reduced vector, and plot the ratio of the pairwise distance $\frac{1}{n(n-1)} \sum_{n \geqslant i \neq j \geqslant 1} \frac{\|\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_j\|_2}{\|\mathbf{x}_i - \mathbf{x}_j\|_2}$ and the average standard deviation for different $k$ averaged over 100 replications. In the MNIST example, we choose the first $n = 50$ vectors of size $d = 784$, normalize them, and perform the same experiment. Figure 1 shows results on simulated ($d = 2500$) and MNIST data for the Gaussian and Very Sparse RP. See B for additional experiments.

These experiments show that to preserve pairwise distance and cosine similarity, TRP performs nearly as well as RP for all three types of maps. With only five replicates, TRP(5) reduces the variance significantly in real data while not much in the simulation setting. The difference in accuracy between methods diminishes as $k$ increases. When $d = d_1 d_2 = 40000$, the storage for TRP(5) is still $\frac{1}{20}$ of the Gaussian RP. The variance reduction is effective especially in sparse and very sparse setting.

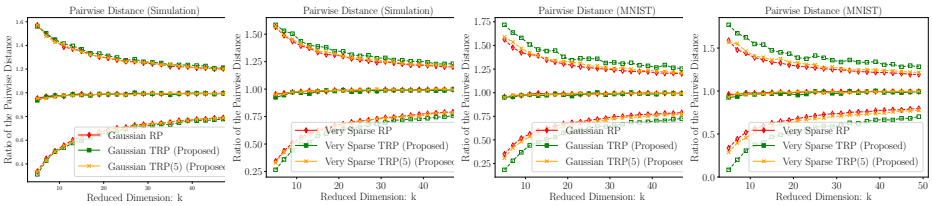


Figure 1: Isometry quality for simulated and MNIST data. The left two plots show results for Gaussian and Very Sparse RP, TRP, TRP(5) respectively applied to $n = 20$ standard normal data vectors in $\mathbb{R}^{2500}$. The right two plots show the same for 50 MNIST image vectors in $\mathbb{R}^{784}$. The dashed line shows the error two standard deviations from the average ratio.

| | Gaussian | Sparse | Very Sparse |
|---|---|---|---|
| RP | 0.1198 (0.0147) | 0.1198 (0.0150) | 0.1189 (0.0108) |
| TRP | 0.1540 (0.0290) | 0.1609 (0.0335) | 0.1662 (0.0307) |
| TRP(5) | 0.1262 (0.0166) | 0.1264 (0.0194) | 0.1276 (0.0164) |

Table 1: RMSE for the estimate of the pairwise inner product of the MNIST data, where standard error is in the parentheses.

# 5 Application: Sketching

Beyond random projection, our novel TRP also has an important application in sketching. Sketching is an important technique to accelerate expensive computations with widespread applications, such as regression, low-rank approximation, and graph sparsification, etc. Halko et al. [2011], Woodruff et al. [2014] The core idea behind sketching is to compress a large dataset, typically a matrix or tensor, into a smaller one by multiplying a random matrix. In this section, we will mainly focus on the low-rank matrix approximation problem. Consider a matrix $\mathbf{X} \in \mathbb{R}^{m\times d}$ with rank $r$, we want to find the best rank-$r$ approximation with the minimal amount of time. The most common method is the randomized singular value decomposition (SVD), whose underlying idea is sketching.

First, we compute the linear sketch $\mathbf{Z} \in \mathbb{R}^{m\times k}$ by $\mathbf{Z} = \mathbf{X}\mathbf{\Omega}$, where $\mathbf{\Omega} \in \mathbb{R}^{d\times r}$ is the random map. Then we compute the QR decomposition of $\mathbf{X}\mathbf{\Omega}$ by $\mathbf{QR} = \mathbf{Z}$, where $\mathbf{Q} \in \mathbb{R}^{m\times k}, \mathbf{R} \in \mathbb{R}^{r\times r}$. At the end, we project $\mathbf{X}$ onto the column space of $\mathbf{Q}$, and obtain the approximation $\hat{\mathbf{X}} = \mathbf{Q}\mathbf{Q}^\top\mathbf{X}$.

With our TRP, we can significantly reduce the storage of the random map, while achieving similar rate of convergence as demonstrated in Figure 2. With further variance reduction by taking the geometric-median over multiple runs, our TRP with variance reduction can achieve even better performance. The detailed implementation is given in Algorithm 1. And we will delay the theoretical analysis of this method for future works.

**Algorithm 1** Tensor Sketching with Variance Reduction

**Input:** $\mathbf{X} \in \mathbb{R}^{m \times d}$, where $d = \prod_{i=1}^{N} d_n$ and RMAP is a user-specified function that generates a random dimension reduction map. $T$ is the number of runs for variance reduction averaging.

**function** SSVR($\mathbf{X}, \{d_n\}, k, T,$ RMAP)
  **for** $t = 1 \dots T$ **do**
    **for** $i = 1 \dots N$ **do** $\mathbf{\Omega}_i^{(t)} = \mathrm{RMAP}(\mathrm{d_i}, \mathrm{k})$
    **end for**
    $\mathbf{\Omega}^{(t)} = \mathbf{\Omega}_1^{(t)} \odot \cdots \odot \mathbf{\Omega}_N^{(t)}$
    $(\mathbf{Q}^{(t)}, \sim) = \mathrm{QR}(\mathbf{X}\mathbf{\Omega}^{(\mathrm{t})})$
    $\hat{\mathbf{X}}^{(t)} = \mathbf{Q}^{(t)}\mathbf{Q}^{(t)T}\mathbf{X}$
  **end for**
  $\hat{\mathbf{X}} = \frac{1}{T}\sum_{t=1}^{T} \hat{\mathbf{X}}^{(t)}$
  **return G**
**end function**

Furthermore, the extension of TRP to tensor data is also natural. To be specific, the $n^{th}$ unfolding of a large tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, denoted as $\mathbf{X}^{(n)}$, has dimension $I_n \times I_{(-n)}$, where $I_{(-n)} = \prod_{i \neq n, i \in [N]} I_i$ . To construct a sketch for the unfolding, we need to create a random matrix of size $I_{(-n)} \times k$. Then, our TRP becomes a natural choice to avoid the otherwise extremely expensive storage cost. For many popular tensor approximation algorithms, it is even necessary to perform sketching for every dimension of the tensor De Lathauwer et al. [2000], Wang et al. [2015]. In the simulation section, we perform experiments for the unfolding of the higher-order order tensor with our structured sketching algorithms (Figure 2). For more details in tensor algebra, please refer to Kolda and Bader [2009].

**Experimental Setup** In sketching problems, considering a $N$-D tensor $\mathcal{X} \in \mathbb{R}^{I^N}$ with equal length along all dimensions, we want to compare the performance of the low rank approximation with different maps for its first unfolding $\mathbf{X}^{(1)} \in \mathbb{R}^{I \times I^{N-1}}$.

We construct the tensor $\mathcal{X}$ in the following way. Generate a core tensor $\mathcal{C} \in \mathbb{R}^{r^N}$, with each entry Unif($[0, 1]$). Independently generate $N$ orthogonal arm matrices by first creating $\mathbf{A}_1, \dots, \mathbf{A}_N \in \mathbb{R}^{r \times I}$ and then computing the arm matrices by $(\mathbf{Q}_n, \sim) = \mathrm{QR}(\mathbf{A}_\mathrm{n})$, for $1 \leqslant n \leqslant N$.

$$\mathcal{X} = \mathcal{C} \times_1 \mathbf{Q}_1 \cdots \times_N \mathbf{Q}_N + \sqrt{\frac{0.01 \cdot \|\mathcal{X}^\natural\|_F^2}{I^N}} \mathcal{N}(0, 1).$$

Then, we construct the mode-1 unfolding of $\mathbf{X} = \mathbf{X}^{(1)}$, which has a rank smaller than or equal to $r$.

In our simulation, we consider the scenarios of 2-D ($900 \times 900$), 3-D ($400 \times 400 \times 400$), 4-D ($100 \times 100 \times 100 \times 100$) tensor data, with corresponding mode-1 unfolding of size $900 \times 900$, $400 \times 160000$, $100 \times 1000000$ respectively and $r = 5$.

In each scenario, we compare the performance for Gaussian RP, TRP, and $\text{TRP}_5$ maps with varying $k$ from 5 to 25. The TRP map in these scenarios has 2, 4, 6 components of size $30 \times k$, $20 \times k$, $10 \times k$ respectively. And the number of runs variance reduction averaging is $T = 5$. In the end, we evaluate the performance by generating the random matrix 100 times and compute the relative error $\frac{\|\mathbf{X}-\hat{\mathbf{X}}\|}{\|\mathbf{X}\|}$, and constructing a 95% confidence interval for it.

**Result** From Figure 2, we can observe that the relative error decreases as $k$ increases as expected for all dimension reduction maps. The difference of the performance between the Khatri-Rao map and Gaussian map is small when $N = 2$, but increases when $N$ increases, whereas the Khatri-Rao variance reduced method is particularly effective producing strictly better performance than the other two.

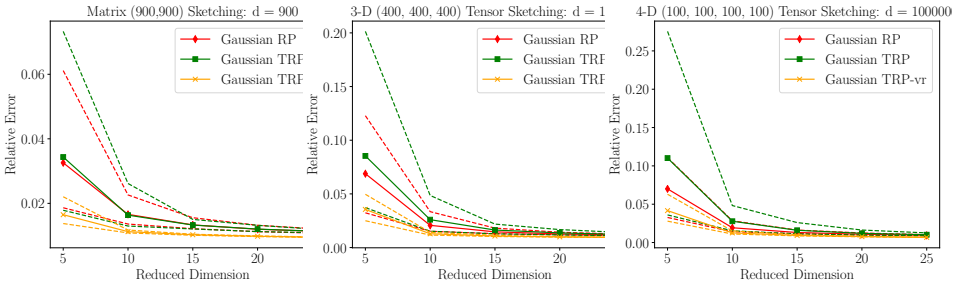


Figure 2: Relative Error for the low-rank tensor unfolding approximation: *we compare the relative errors for low-rank tensor approximation with different input size: 2-D (*$900 \times 900$*), 3-D (*$400 \times 400 \times 400$*), 4-D (*$100 \times 100 \times 100 \times 100$*). In each setting, we compare the performance of Gaussian RP, TRP, and* $\text{TRP}_5$*. The dashed line stands for the 95% confidence interval.*

# 6 Conclusion

The TRP is a novel dimension reduction map composed of smaller DRMs. Compared to its constituent DRMs, it significantly reduces the requirements for randomness and for storage. Numerically, the variance-reduced TRP(5) method with only five replicates achieves accuracy comparable to the conventional RPs for 1/20 of the original storage. We prove the TRP and TRP(T) are expected isometries with vanishing variance, and provide a non-asymptotic error bound for the order 2 TRP.

For the future work, we will provide a general non-asymptotic bound for the higher order TRP and develop the theory relevant for the application of the TRP in sketching low-rank approximation, given its practical effectiveness (shown in Appendix 5).

# References


Dimitris Achlioptas. Database-friendly random projections: Johnson-lindenstrauss with binary coins. *Journal of computer and System Sciences*, 66 (4):671–687, 2003.

Nir Ailon and Bernard Chazelle. Approximate nearest neighbors and the fast johnson-lindenstrauss transform. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 557–563. ACM, 2006.

Zeyuan Allen-Zhu, Rati Gelashvili, Silvio Micali, and Nir Shavit. Sparse sign-consistent johnson–lindenstrauss matrices: Compression with neuroscience-based constraints. *Proceedings of the National Academy of Sciences*, 111(47): 16872–16876, 2014.

Rosa I Arriaga and Santosh Vempala. An algorithmic theory of learning: Robust concepts and random projection. *Machine Learning*, 63(2):161–182, 2006.

Casey Battaglino, Grey Ballard, and Tamara G Kolda. A practical randomized cp tensor decomposition. *SIAM Journal on Matrix Analysis and Applications*, 39(2):876–901, 2018.

Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 245–250. ACM, 2001.

Jean Bourgain, Sjoerd Dirksen, and Jelani Nelson. Toward a unified theory of sparse dimensionality reduction in euclidean space. *Geometric and Functional Analysis*, 25(4):1009–1088, 2015.

Jeremy Buhler and Martin Tompa. Finding motifs using random projections. *Journal of computational biology*, 9(2):225–242, 2002.

Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications*, 21 (4):1253–1278, 2000.

Dmitriy Fradkin and David Madigan. Experiments with random projections for machine learning. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 517–522. ACM, 2003.

Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *European conference on computer vision*, pages 304–317. Springer, 2008.

Samuel Kaski. Dimensionality reduction by random mapping: Fast similarity computation for clustering. In *Neural networks proceedings, 1998. ieee world congress on computational intelligence. the 1998 ieee international joint conference on*, volume 1, pages 413–418. IEEE, 1998.

Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

Ping Li, Trevor J Hastie, and Kenneth W Church. Very sparse random projections. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 287–296. ACM, 2006.

Kun Liu, Hillol Kargupta, and Jessica Ryan. Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. *IEEE Transactions on knowledge and Data Engineering*, 18(1):92–106, 2006.

Makoto Matsumoto and Takuji Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 8(1):3–30, 1998.

Christos H Papadimitriou, Prabhakar Raghavan, Hisao Tamaki, and Santosh Vempala. Latent semantic indexing: A probabilistic analysis. *Journal of Computer and System Sciences*, 61(2):217–235, 2000.

Mark Rudelson. Row products of random matrices. *Advances in Mathematics*, 231(6):3199–3231, 2012.

Mark Rudelson, Roman Vershynin, et al. Hanson-wright inequality and sub-gaussian concentration. *Electronic Communications in Probability*, 18, 2013.

Ozgur D Sahin, Aziz Gulbeden, Fatih Emekçi, Divyakant Agrawal, and Amr El Abbadi. Prism: indexing multi-dimensional data in p2p networks using reference vectors. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 946–955. ACM, 2005.

Kathrin Schäcke. On the kronecker product. 2013.

Yiming Sun, Yang Guo, Charlene Luo, Joel Tropp, and Madeleine Udell. Low-rank tucker approximation of a tensor from streaming data. *arXiv preprint arXiv:1904.10951*, 2019.

J.A. Tropp, A. Yurtsever, M. Udell, and V. Cevher. Streaming low-rank matrix approximation with an application to scientific simulation. Technical report, 2019. URL `https://arxiv.org/pdf/1902.08651.pdf`.

Joel A Tropp, Alp Yurtsever, Madeleine Udell, and Volkan Cevher. Practical sketching algorithms for low-rank matrix approximation. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1454–1485, 2017.

Jun Wang, Sanjiv Kumar, and Shih-Fu Chang. Semi-supervised hashing for large-scale search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(12):2393–2406, 2012.

Yining Wang, Hsiao-Yu Tung, Alexander J Smola, and Anima Anandkumar. Fast and guaranteed tensor decomposition via sketching. In *Advances in Neural Information Processing Systems*, pages 991–999, 2015.

David P Woodruff et al. Sketching as a tool for numerical linear algebra. *Foundations and Trends® in Theoretical Computer Science*, 10(1–2):1–157, 2014.

Franco Woolfe, Edo Liberty, Vladimir Rokhlin, and Mark Tygert. A fast randomized algorithm for the approximation of matrices. *Applied and Computational Harmonic Analysis*, 25(3):335–366, 2008.

John Wright, Allen Y Yang, Arvind Ganesh, S Shankar Sastry, and Yi Ma. Robust face recognition via sparse representation. *IEEE transactions on pattern analysis and machine intelligence*, 31(2):210–227, 2009.

Alp Yurtsever, Madeleine Udell, Joel A Tropp, and Volkan Cevher. Sketchy decisions: Convex low-rank matrix optimization with optimal storage. *arXiv preprint arXiv:1702.06838*, 2017.

# Appendix A Proof for Bias and Variance Analysis

Before presenting the proof for the main theory, we first define some new notations. Since these notations will only be used in technical proofs, we do not include them in the main body.

## Extra Notations for Technical Proofs

For a vector $\mathbf{x}$ with length $\prod_{n=1}^{N} d_n$, for simplicity, we introduce the multi-index for it: let $\mathbf{x}_{r_1,\cdots,r_N}, \forall r_n \in [d_n]$, represent the $(1+\sum_{n=1}^{N}(r_n-1)s_n)^{th}$ element, where $s_n = \prod_{n+1}^{N} d_n$ for $n < N$, and 1 for $n = N$. For vector $\mathbf{r}_1$, $\mathbf{r}_2$, we say $\mathbf{r}_1 = \mathbf{r}_2$ if and only if all their elements are the same.

Also, we let $\mathbf{vec}(\mathbf{A})$ be the vectorization operator for any matrix $\mathbf{A} \in \mathbb{R}^{d\times k}$, which stacks all columns of matrix $\mathbf{A}$ and returns a vector of length $kd$, $[\mathbf{A}(\cdot,1);\cdots;\mathbf{A}(\cdot,k);]$. Here we use semi-colon to denote the vertical stack of vectors $\mathbf{x}$ and $\mathbf{y}$ as $[\mathbf{x};\mathbf{y}]$. As comparison, we use comma to mean stack row vector horizontally like $[\mathbf{x}^\top,\mathbf{y}^\top]$.

## Proof for Theorem 3.1

*Proof.* We first give a sufficient condition for general random matrix to let (3.1) be held, then we show that Khatri-Rao map with condition in Theorem 3.1 satisfies these two general sufficient conditions.

Consider a general random matrix $\mathbf{A} \in \mathbb{R}^{k\times d}$ and $\mathbf{x} \in \mathbb{R}^d$. we claim if $\mathbb{E}\mathbf{A}^2(r,s) = 1, \forall r,s$ and $\mathbb{E}\mathbf{A}(r,s_1)\mathbf{A}(r,s_2) = 0, \forall r \in [k], s_1 \neq s_2 \in [d]$, then $\mathbb{E}\|\frac{1}{\sqrt{k}}\mathbf{y}\|_2^2 = \|\mathbf{x}\|_2^2$, when $\mathbf{y} = \mathbf{A}\mathbf{x}$. To see why, it suffices to show that $\mathbb{E}y_r^2 = \|x\|_2^2$.

$$\begin{aligned}\mathbb{E}y_r^2 &= \mathbb{E}\sum_{s_1=1}^{d}\sum_{s_2=1}^{d}\mathbf{A}(r,s_1)\mathbf{A}(r,s_2)x_{s_1}x_{s_2}\\ &= \sum_{s=1}^{d}\mathbf{A}^2(r,s)x_s^2 = \|\mathbf{x}\|_2^2,\end{aligned}$$

where the first equation in the second line comes from the fact that $\mathbb{E}\mathbf{A}(r,s_1)\mathbf{A}(r,s_2) = 0$ for $s_1 \neq s_2$ and the second equation in the second line comes from that $\mathbb{E}\mathbf{A}^2(r,s) = 1$.

Then, we will prove Theorem 3.1 by induction. We first show that for two matrices $\mathbf{B}_1 \in \mathbb{R}^{d_1\times k}, \mathbf{B}_2 \in \mathbb{R}^{d_2\times k}$ whose entries satisfy the two conditions in Theorem 3.1: $\mathbb{E}\mathbf{B}_n^2(r,s) = 1$ and $\mathbb{E}[\mathbf{B}_n(r_1,s)\mathbf{B}_n(r_2,s)] = 0$ for $n = 1,2, s \in [d], r, r_1 \neq r_2 \in [d_n]$, we have $\mathbf{A} = (\mathbf{B}_1 \odot \mathbf{B}_2)^\top$ satifies the two sufficient conditions stated previously. It suffices to restrict our focus to the first row of $\mathbf{\Omega}$ and we apply the multi-index to it. For any $1 \leqslant r_1 \leqslant d_1, 1 \leqslant r_2 \leqslant d_2$,

$$\begin{aligned}&\mathbb{E}\mathbf{A}_1^2(k_1,k_2) = \mathbb{E}\mathbf{B}_1^2(k_1,1)\mathbf{B}_2^2(k_2,1)\\ &= \mathbb{E}\mathbf{B}_1^2(k_1,1)\mathbb{E}\mathbf{B}_2^2(k_1,1) = 1. \text{ (independence bewteen } \mathbf{B}_i, i = 1,2)\end{aligned}$$

To avoid confusion in notation, we argue that $\mathbf{A}(1,\cdot)$ is the first row vector of $\mathbf{A}$ of size $d_1 d_2$, and we apply the multi-index to it. Also, for two different elements in the first row of $\mathbf{A}$: $\mathbf{A}_1(k_1,k_2)\mathbf{A}_1(s_1,s_2)$ at least one of $k_1 \neq s_1$, $k_2 \neq s_2$ hold. Without losing generality, assuming $k_1 \neq s_1$,

$$\begin{aligned}
&\mathbb{E}\mathbf{A}_1(k_1,k_2)\mathbf{A}_1(s_1,s_2) = \mathbb{E}\mathbf{B}_1(k_1,1)\mathbf{B}_2(k_2,1)\mathbf{B}_1(s_1,1)\mathbf{B}_2(s_2,1)\\
&= \mathbb{E}\mathbb{E}\left[\mathbf{B}_1(k_1,1)\mathbf{B}_1(k_2,1)\mathbf{B}_2(k_2,1)\mathbf{B}_2(s_2,1) \mid \mathbf{B}_2(k_2,1)\mathbf{B}_2(s_2,1)\right]\\
&= \mathbb{E}\mathbf{B}_2(k_2,1)\mathbf{B}_2(s_2,1)\mathbb{E}\left[\mathbf{B}_1(k_1,1)\mathbf{B}_1(s_1,1)\right] = 0,
\end{aligned}$$

where we use the fact that entries within/across $B_i$ are independent with each other and have zero expectation.

Notice that two conditions for $\mathbf{A} = (\mathbf{B}_1 \odot \mathbf{B}_2)^\top$ directly show that $\mathbf{B}_1 \odot \mathbf{B}_2$ satisfies two conditions in Theorem 3.1, we could use a standard mathematical induction argument to finish the proof for TRP. For TRP(T),

$$\begin{aligned}
\mathbb{E}\|\mathrm{TRP}_T(\mathbf{x})\|_2^2 &= \frac{1}{T}\mathbb{E}\|\sum_{t=1}^{T}\mathrm{TRP}^{(t)}(\mathbf{x})\|_2^2\\
&= \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\mathrm{TRP}^{(t)}(\mathbf{x})\|_2^2 = \|\mathbf{x}\|_2^2,
\end{aligned}$$

where in the second line we use the fact that each $\mathrm{TRP}^{(t)}$ is independent with each other. □

Next we introduce a lemma which shows that by bounding the deviation for the norm square of each vector, we could also bound the deviation for inner product. Although it is commonly known in any random projection literature, for completeness, we still list the lemma with proof here.

## Proof for Theorem 3.2

*Proof.* Let $\mathbf{y} = \mathbf{A}\mathbf{x}$. We know from Theorem 3.1 that $\mathbb{E}\|\mathrm{TRP}(\mathbf{x})\|_2^2 = \frac{1}{k}\mathbb{E}\|\mathbf{A}\mathbf{x}\|^2 = \|\mathbf{x}\|_2^2$. Notice

$$\mathbb{E}(\|\mathrm{TRP}_T(\mathbf{x})\|_2^2) = \|\mathbf{x}\|_2^2,$$

and $\mathbb{E}y_1^2 = \|x\|_2^2$ as shown in the poof of Lemma 3.1. It is easy to see that

$$\mathbb{E}\|\mathbf{y}\|_2^4 = \sum_{i=1}^{k}\mathbb{E}y_i^4 + \sum_{i\neq j}\mathbb{E}y_i^2 y_j^2.$$

Again, as shown in Theorem 3.1, $\mathbb{E}y_i^2 y_j^2 = \mathbb{E}y_i^2\mathbb{E}y_j^2 = \|\mathbf{x}\|^4$. To find $\mathbb{E}\|\mathbf{y}\|_2^4$, it suffices to find $\mathbb{E}y_1^4$ by noticing that $y_i$ are iid random variables. Let $\Omega$ be the

set containing all corresponding multi-index vector for $\{1, \cdots, \prod_{n=1}^{N} d_n\}$.

$$y_1^4 = \left[\sum_{\mathbf{r}\in\Omega} \mathbf{A}(1,\mathbf{r})x_{\mathbf{r}}\right]^4 = \sum_{\mathbf{r}\in\Omega} \mathbf{A}^4(1,\mathbf{r})x_{\mathbf{r}}^4 + 3\sum_{\mathbf{r}_1\neq\mathbf{r}_2\in\Omega} \mathbf{A}^2(1,\mathbf{r}_1)x_{\mathbf{r}_1}^2\mathbf{A}^2(1,\mathbf{r}_2)x_{\mathbf{r}_2}^2$$
$$+ 6\sum_{\mathbf{r}_1\neq\mathbf{r}_2\neq\mathbf{r}_3\in\Omega} \mathbf{A}^2(1,\mathbf{r}_1)x_{\mathbf{r}_1}\mathbf{A}(1,\mathbf{r}_2)x_{\mathbf{r}_2}\mathbf{A}(1,\mathbf{r}_3)x_{\mathbf{r}_3} + 4\sum_{\mathbf{r}_1\neq\mathbf{r}_2\in\Omega} \mathbf{A}^3(1,\mathbf{r}_1)x_{\mathbf{r}_1}^3\mathbf{A}(1,\mathbf{r}_2)x_{\mathbf{r}_2}$$
$$+ \sum_{\mathbf{r}_1\neq\mathbf{r}_2\neq\mathbf{r}_3\neq\mathbf{r}_4\in\Omega} \mathbf{A}(1,\mathbf{r}_1)x_{\mathbf{r}_1}\mathbf{A}(1,\mathbf{r}_2)x_{\mathbf{r}_2}\mathbf{A}(1,\mathbf{r}_3)x_{\mathbf{r}_3}\mathbf{A}(1,\mathbf{r}_4)x_{\mathbf{r}_4}.$$

It is not hard to see that except for the first line, the expectation of second and third line is zero.

$$\mathbb{E}\mathbf{A}^4(1,\mathbf{r}) = \mathbb{E}\mathbf{A}_1^4(1,r_1)\cdots\mathbf{A}_N^4(1,r_N) = \Delta^N.$$

Also with proof in Theorem 3.1,

$$\mathbb{E}\mathbf{A}^2(1,\mathbf{r}_1)\mathbf{A}^2(1,\mathbf{r}_2) = \mathbb{E}\mathbf{A}^2(1,\mathbf{r}_1)\mathbb{E}\mathbf{A}^2(1,\mathbf{r}_2) = 1.$$

Combining these two together, we have

$$\begin{aligned}\mathbb{E}\|\mathrm{TRP}(\mathbf{x})\|^4 &= \frac{1}{k^2}\left[k(\Delta^N-3)\|\mathbf{x}\|_4^4 + 3k\|\mathbf{x}\|_2^4 + (k-1)k\|\mathbf{x}\|_2^4\right]\\ &= \frac{1}{k}\left[(\Delta^N-3)\|\mathbf{x}\|_4^4 + 2\|\mathbf{x}\|_2^4\right] + \|\mathbf{x}\|_2^4.\end{aligned} \tag{A.1}$$

Therefore,

$$\mathrm{Var}(\|\mathrm{TRP}(\mathbf{x})\|_2^2) = \mathbb{E}\|\mathrm{TRP}(\mathbf{x})\|_2^4 - (\mathbb{E}\|\mathrm{TRP}(\mathbf{x})\|_2^2)^2 = \frac{1}{k}\left[(\Delta^N-3)\|\mathbf{x}\|_4^4 + 2\|\mathbf{x}\|_2^4\right].$$

Now we switch to see how much variance could be reduced by the variance reduction method. With Theorem 3.1, we already know that $\mathbb{E}\|\mathrm{TRP}_T(\mathbf{x})\|_2^2 = \|\mathbf{x}\|_2^2$. The rest is to calculate $\mathbb{E}\|\mathrm{TRP}_T(\mathbf{x})\|_2^4$ out.

$$\|\mathrm{TRP}_T(\mathbf{x})\|_2^4 = \frac{1}{T^2}\left[\sum_{t=1}^{T}\|\mathrm{TRP}^{(t)}(\mathbf{x})\|_2^2 + \sum_{t_1\neq t_2}\langle\mathrm{TRP}^{(t_1)}(\mathbf{x}), \mathrm{TRP}^{(t_2)}(\mathbf{x})\rangle\right]^2$$
$$= \frac{1}{T^2}\left[\sum_{t=1}^{T}\|\mathrm{TRP}^{(t)}(\mathbf{x})\|_2^4 + \sum_{t_1\neq t_2}\|\mathrm{TRP}^{(t_1)}(\mathbf{x})\|_2^2\|\mathrm{TRP}^{(t_2)}(\mathbf{x})\|_2^2 + 2\sum_{t_1\neq t_2}\langle\mathrm{TRP}^{(t_1)}(\mathbf{x}), \mathrm{TRP}^{(t_2)}(\mathbf{x})\rangle^2 + \mathrm{rest}\right].$$

It is not hard to show that $\mathbb{E}(\mathrm{rest}) = 0$. Following the definition of $\mathbf{y}$,

$$\mathbb{E}\|\mathrm{TRP}^{(t_1)}(\mathbf{x})\|_2^2\|\mathrm{TRP}^{(t_2)}(\mathbf{x})\|_2^2 = \|\mathbf{x}\|_2^4,$$

and

$$\begin{aligned}&\mathbb{E}\langle\mathrm{TRP}^{(t_1)}(\mathbf{x}), \mathrm{TRP}^{(t_2)}(\mathbf{x})\rangle^2\\ &= \frac{1}{k^2}\mathbb{E}\left[\sum_{i=1}^{k} y_i^{(t_1)}y_i^{(t_2)}\right]^2\\ &= \frac{1}{k}\mathbb{E}(y_1^{(t_1)})^2\mathbb{E}(y_1^{(t_2)})^2 = \frac{1}{k}\|\mathbf{x}\|_2^4.\end{aligned}$$

Combining all these together, we could show that

$$\begin{aligned}\mathrm{Var}(\|\mathrm{TRP}_T(\mathbf{x})\|_2^2) &= \mathbb{E}\|\mathrm{TRP}_T(\mathbf{x})\|_2^4 - (\mathbb{E}\|\mathrm{TRP}_T(\mathbf{x})\|_2^2)^2 \\ &= \frac{1}{T^2}\left[\frac{T}{k}\left[(\Delta^N-3)\|\mathbf{x}\|_4^4 + 2\|\mathbf{x}\|_2^4\right]\right. \\ &\left. +T\|\mathbf{x}\|_2^4 + T(T-1)\|\mathbf{x}\|_2^4 + \frac{2T(T-1)}{k}\|\mathbf{x}\|_2^4\right] - \|\mathbf{x}\|_2^4 \\ &= \frac{1}{Tk}(\Delta^N-3)\|\mathbf{x}\|_4^4 + \frac{2}{k}\|\mathbf{x}\|_2^4.\end{aligned}$$

□

## Proof for Lemma 3.4

*Proof.* First, we show the unbiasedness of the inner product estimation:

$$\mathbb{E}(\langle \mathrm{TRP}(\mathbf{x}), \mathrm{TRP}(\mathbf{y})\rangle) = [\|\mathrm{TRP}(\mathbf{x}) + \mathrm{TRP}(\mathbf{y})\|_2^2 - \|\mathrm{TRP}(\mathbf{x})\|_2^2 - \|\mathrm{TRP}(\mathbf{y})\|_2^2]/2 = \langle \mathbf{x}, \mathbf{y}\rangle.$$

The equation above follows from Thm 3.1, the unbiasedness of norm estimation. We can apply the similar idea to get $\mathbb{E}(\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle) = \langle \mathbf{x}, \mathbf{y}\rangle$.

Now, let $\mathbf{u} = \mathbf{A}\mathbf{x}$, $\mathbf{v} = \mathbf{A}\mathbf{y}$. Then,

$$\begin{aligned}(u_1v_1)^2 &= [\sum_{\mathbf{r}\in\Omega}\mathbf{A}(1,\mathbf{r})x_{\mathbf{r}}]^2[\sum_{\mathbf{r}\in\Omega}\mathbf{A}(1,\mathbf{r})y_{\mathbf{r}}]^2 \\ &= \sum_{\mathbf{r}}\mathbf{A}(1,\mathbf{r})^4x_{\mathbf{r}}^2y_{\mathbf{r}}^2 + \sum_{\mathbf{r}_1\neq\mathbf{r}_2}\mathbf{A}(1,\mathbf{r}_1)^2\mathbf{A}(1,\mathbf{r}_2)^2x_{\mathbf{r}_1}^2y_{\mathbf{r}_2}^2 \\ &+ 2\sum_{\mathbf{r}_1\neq\mathbf{r}_2}\mathbf{A}(1,\mathbf{r}_1)^2\mathbf{A}(1,\mathbf{r}_2)^2x_{\mathbf{r}_1}x_{\mathbf{r}_2}y_{\mathbf{r}_1}y_{\mathbf{r}_2} + \text{rest.},\end{aligned}$$

Since $\mathbb{E}\mathbf{A}(1,\mathbf{r}) = 0,\ \forall\mathbf{r}$, $\mathbb{E}(\text{rest}) = 0$. Also with proof in Thm 3.1,

$$\mathbb{E}\mathbf{A}^2(1,\mathbf{r}_1)\mathbf{A}^2(1,\mathbf{r}_2) = \mathbb{E}\mathbf{A}^2(1,\mathbf{r}_1)\mathbb{E}\mathbf{A}^2(1,\mathbf{r}_2) = 1.$$

And,

$$\mathbb{E}\mathbf{A}^4(1,\mathbf{r}) = \mathbb{E}\mathbf{A}_1^4(1,r_1)\cdots\mathbf{A}_N^4(1,r_N) = \Delta^N.$$

Then, similar to (A.1), we can obtain:

$$\begin{aligned}\mathbb{E}(\langle \mathrm{TRP}(\mathbf{x}), \mathrm{TRP}(\mathbf{y})\rangle)^2 &= \frac{1}{k^2}\mathbb{E}[\sum_{i,j}u_iv_iu_jv_j]^2 = \frac{1}{k^2}\mathbb{E}[\sum_i(u_iv_i)^2] + \frac{1}{k^2}\mathbb{E}[\sum_{i\neq j}(u_iv_iu_jv_j)] \\ &= \frac{1}{k}\mathbb{E}(u_1v_1)^2 + \frac{k(k-1)}{k^2}\langle \mathbf{x}, \mathbf{y}\rangle^2 \\ &= \frac{1}{k}[(\Delta^N-3)\sum_{\mathbf{r}}x_{\mathbf{r}}^2y_{\mathbf{r}}^2 + \|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2 + \langle \mathbf{x}, \mathbf{y}\rangle^2] + \langle \mathbf{x}, \mathbf{y}\rangle^2.\end{aligned} \tag{A.2}$$

Then, with the unbiasedness of TRP map, we get

$$\begin{aligned}\mathrm{Var}(\langle \mathrm{TRP}(\mathbf{x}), \mathrm{TRP}(\mathbf{y})\rangle) &= \mathbb{E}(\langle \mathrm{TRP}(\mathbf{x}), \mathrm{TRP}(\mathbf{y})\rangle^2) - (\mathbb{E}\langle \mathrm{TRP}(\mathbf{x}), \mathrm{TRP}(\mathbf{y})\rangle)^2 \\ &= \frac{1}{k}[(\Delta^N - 3)\sum_{\mathbf{r}} x_{\mathbf{r}}^2 y_{\mathbf{r}}^2 + \|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2 + \langle \mathbf{x}, \mathbf{y}\rangle^2].\end{aligned}$$

Now, we proceed to find the variance for the inner product estimation with $\mathrm{TRP}_T$. Since $\mathbf{var}(\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle) = \mathbb{E}(\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle^2) - (\mathbb{E}\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle)^2$, we first compute:

$$\begin{aligned}\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle^2 &= \frac{1}{T^2}\langle \sum_{i=1}^{T} \mathrm{TRP}^{(i)}(\mathbf{x}), \sum_{j=1}^{T} \mathrm{TRP}^{(j)}(\mathbf{y})\rangle^2 \\ &= \frac{1}{T^2}\sum_{i=1}^{T}\langle \mathrm{TRP}^{(i)}(\mathbf{x}), \mathrm{TRP}^{(i)}(\mathbf{y})\rangle^2 \\ &+ \frac{1}{T^2}\sum_{i\neq j}(\langle \mathrm{TRP}^{(i)}(\mathbf{x}), \mathrm{TRP}^{(i)}(\mathbf{y})\rangle)(\langle \mathrm{TRP}^{(j)}(\mathbf{x}), \mathrm{TRP}^{(j)}(\mathbf{y})\rangle) \\ &+ \frac{2}{T^2}\sum_{i\neq j}\langle \mathrm{TRP}^{(i)}(\mathbf{x}), \mathrm{TRP}^{(j)}(\mathbf{y})\rangle^2 + \text{rest}.\end{aligned}$$

Following the definition of the TRP map, we can see:

$$\begin{aligned}&\mathbb{E}\langle \mathrm{TRP}^{(t_1)}(\mathbf{x}), \mathrm{TRP}^{(t_2)}(\mathbf{y})\rangle^2 \\ &= \frac{1}{k^2}\mathbb{E}\left[\sum_{i=1}^{k} u_i^{(t_1)} v_i^{(t_2)}\right]\left[\sum_{i=1}^{k} u_i^{(t_1)} v_i^{(t_2)}\right] \\ &= \frac{1}{k}\mathbb{E}[u_1^{(t_1)} u_1^{(t_1)}]\mathbb{E}[v_1^{(t_2)} v_1^{(t_2)}] = \frac{1}{k}\|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2.\end{aligned}$$

First, $\mathbb{E}(\text{rest}) = 0$. Then, combining all the above results, we obtain:

$$
\begin{aligned}
\mathbf{var}(\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle) &= \mathbb{E}(\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle^2) - (\mathbb{E}\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle)^2 \\
&= \mathbb{E}(\langle \frac{1}{T^2}\sum_i \mathrm{TRP}^{(i)}(\mathbf{x}), \frac{1}{T^2}\sum_j \mathrm{TRP}^{(j)}(\mathbf{y})\rangle^2) - (\mathbb{E}\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle)^2 \\
&= \frac{1}{T^2}\mathbb{E}(\sum_i \langle \mathrm{TRP}^{(i)}(\mathbf{x}), \mathrm{TRP}^{(i)}(\mathbf{y})\rangle^2 \\
&+ \sum_{i\neq j} \langle \mathrm{TRP}^{(i)}(\mathbf{x}), \mathrm{TRP}^{(i)}(\mathbf{y})\rangle \langle \mathrm{TRP}^{(j)}(\mathbf{x}), \mathrm{TRP}^{(j)}(\mathbf{y})\rangle \\
&+ 2\sum_{i\neq j} \langle \mathrm{TRP}^{(i)}(\mathbf{x}), \mathrm{TRP}^{(j)}(\mathbf{y})\rangle^2) - (\mathbb{E}\langle \mathrm{TRP}_T(\mathbf{x}), \mathrm{TRP}_T(\mathbf{y})\rangle)^2 \\
&= \frac{1}{T^2}\left[\frac{T}{k}[(\Delta^N - 3)\sum_{\mathbf{r}} x_{\mathbf{r}}^2 y_{\mathbf{r}}^2 + \|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2 + \langle \mathbf{x}, \mathbf{y}\rangle^2] + T\langle \mathbf{x}, \mathbf{y}\rangle^2\right. \\
&\left. + \frac{2T(T-1)}{k}\|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2 + T(T-1)\langle \mathbf{x}, \mathbf{y}\rangle^2\right] - \langle \mathbf{x}, \mathbf{y}\rangle^2 \\
&= \frac{1}{kT}(\Delta^N - 3)\sum_{\mathbf{r}} x_{\mathbf{r}}^2 y_{\mathbf{r}}^2 + (\frac{2}{k} - \frac{1}{kT})\|\mathbf{x}\|_2^2\|\mathbf{y}\|_2^2 + \frac{1}{kT}\langle \mathbf{x}, \mathbf{y}\rangle^2.
\end{aligned}
$$

□

# Appendix B More Simulation Results

**Pairwise Distance Estimation** In Figure 3, 4, 5, we compare the performance of Gaussian, Sparse, Very Sparse random maps on the pairwise distance estimation problem with $d = 2500, 10000, 40000, N = 2$. Additionally, we compare their performance for $d = 125000, N = 3$ in Figure 6.

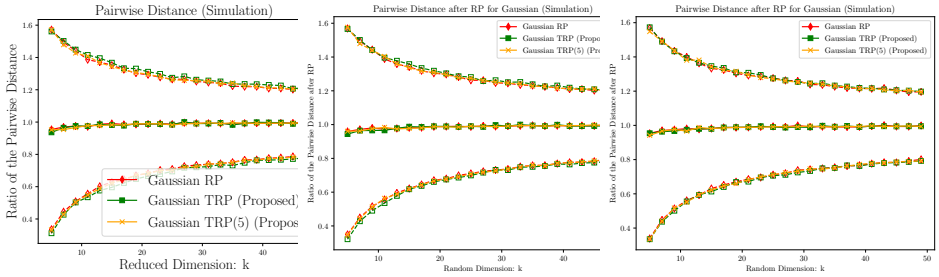


Figure 3: Average ratio of the pairwise distance for simulation data using Gaussian RP: *The plots correspond to the simulation for Gaussian RP, TRP,* $\text{TRP}_5$ *respectively with* $n = 20, d = 2500, 10000, 40000$ *and each data vector comes from* $N(\mathbf{0}, \mathbf{I})$. *The dashed line represents the error bar 2 standard deviation away from the average ratio.*

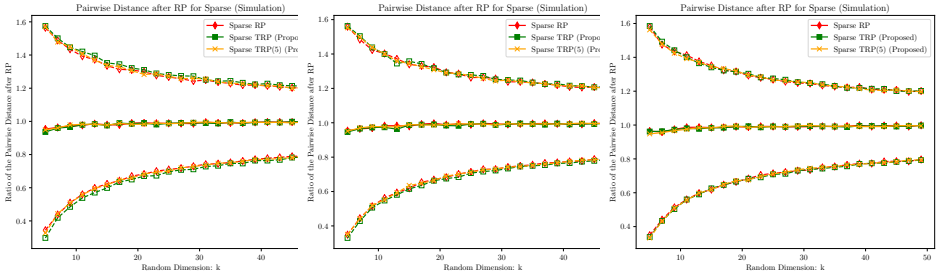


Figure 4: Average ratio of the pairwise distance for simulation data using Sparse RP: *The plots correspond to the simulation for Sparse RP, TRP,* $\text{TRP}_5$ *respectively with* $n = 20, d = 2500, 10000, 40000$ *and each data vector comes from* $N(\mathbf{0}, \mathbf{I})$. *The dashed line represents the error bar 2 standard deviation away from the average ratio.*

**Pairwise Cosine Similarity Estimation** The second experiment is to estimate the pairwise cosine similarity, i.e. $\frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\|_2 \|\mathbf{x}_j\|_2}$ for $\mathbf{x}_i, \mathbf{x}_j$. We use both the simulation data ($d = 10000$) and the MNIST data ($d = 784, n = 60000$). We experiment with Gaussian, Sparse, Very Sparse RP, TRP, and $\text{TRP}_5$ with the

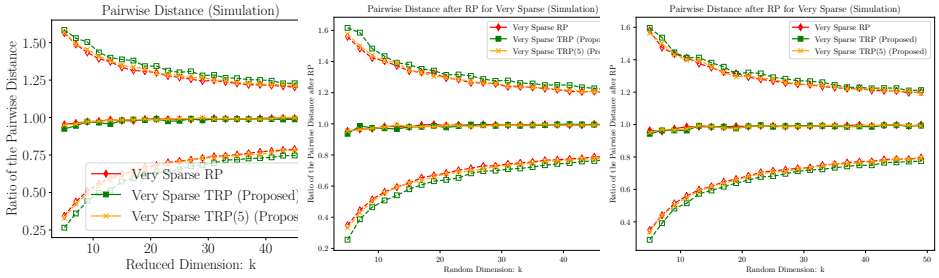


Figure 5: Average ratio of the pairwise distance for simulation data using Very Sparse RP: *The plots correspond to the simulation for Very Sparse RP, TRP,* $\text{TRP}_5$ *respectively with* $n = 20, d = 2500, 10000, 40000$ *and each data vector comes from* $N(\mathbf{0}, \mathbf{I})$*. The dashed line represents the error bar 2 standard deviation away from the average ratio.*

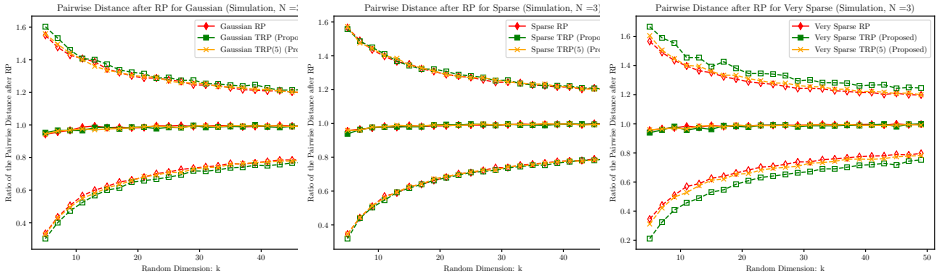


Figure 6: Average ratio of the pairwise distance for simulation data using: *The plots correspond to the simulation for Gaussian, Sparase, Very Sparse RP, TRP,* $\text{TRP}_5$ *respectively with* $n = 20, d = d_1 d_2 d_3 = 50 \times 50 \times 50 = 125000$ *and each data vector comes from* $N(\mathbf{0}, \mathbf{I})$*. The dashed line represents the error bar 2 standard deviation away from the average ratio.*

same setting as above ($k = 50$). We evaluate the performance by the average root mean square error (RMSE). The results is given in Table 1, 2.

| | Gaussian | Sparse | Very Sparse |
|---|---|---|---|
| RP | 0.1409 (0.0015) | 0.1407 (0.0013) | 0.1412 (0.0014) |
| TRP | 0.1431 (0.0016) | 0.1431 (0.0015) | 0.1520 (0.0033) |
| $\text{TRP}_5$ | 0.1412 (0.0012) | 0.1411 (0.0015) | 0.1427 (0.0014) |

Table 2: RMSE for the estimate of the pairwise inner product of the simulation data ($d = 10000, k = 50, n = 100$), where standard error is in the parentheses.

# Appendix C Appendix: Finite Sample Bound

**Definition C.1.** *A random variable $x$ is said to satisfy the generalized-sub-exponential moment condition with constant $\alpha$, if for general positive integer $k$, there exists a general constant $C$(not depending on $k$), s.t.*

$$\mathbb{E}|x|^k \leqslant (Ck)^{k\alpha} \tag{C.1}$$

## Proof for Proposition 3.5

*Proof.* From now on, with losing generality, we will assume $\|x\| = 1$. Let

$$\mathbf{y} = \frac{1}{\sqrt{k}}(\mathbf{A}_1 \odot \mathbf{A}_2)^\top \mathbf{x},$$

Lemma **??** asserts that $\mathbb{E}\|\mathbf{y}\|_2^2 = \|\mathbf{x}\|_2^2$ (conditions in lemma **??** naturally hold for iid random variables in our setting). The key observation is that $y_i, i \in [k]$ is quadratic form of elements of $\mathbf{A}_i, i = 1, 2$. Then as quadratic form of sub-Gaussian variables, $y_i$ are identically independently distributed generalized sub-exponential random variable. Then we could use Hanson-Wright inequality to determine the constants in moments condition C.1 which shall present tighter bound compared to directly citing results of linear combination of sub-exponential random variable defined in (C.1)

We aim to write $y_i$ as a quadratic form of $\mathbf{z}_i := [\mathbf{vec}(\mathbf{A}_1(\cdot, i)); \mathbf{vec}(\mathbf{A}_2(\cdot, i))]$. Also, for convenience, we partition $\mathbf{x}$ into $d_1$ sub-vectors with equal length $d_2$ i.e., $\mathbf{x} = [\mathbf{x}_1; \cdots; \mathbf{x}_{d_1}]$. To make it clear, we consider writing $y_1$ as quadratic form of $\mathbf{z}_1$ first.

$$y_1 = \langle [\mathbf{A}_1(1,1)\mathbf{A}_2(\cdot,1); \cdots; \mathbf{A}_1(d_1,1)\mathbf{A}_2(\cdot,1)], [\mathbf{x}_1; \cdots; \mathbf{x}_{d_1}] \rangle$$

which indicates that we could write

$$y_1 = \mathbf{z}_1^\top \mathbf{M} \mathbf{z}_1,$$

where

$$\mathbf{M} = \begin{bmatrix} \mathbf{0} & \mathbf{D} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad \mathbf{D} = \begin{bmatrix} \mathbf{x}_1^\top \\ \vdots \\ \mathbf{x}_{d_1}^\top. \end{bmatrix}$$

It is easy to see that $\|\mathbf{M}\| \leqslant \|\mathbf{D}\| \leqslant \|\mathbf{D}\|_F = \|\mathbf{M}\|_F = 1$ by assuming $\|\mathbf{x}\| = 1$. Then applying the Hanson Wright inequality in Lemma D.1, we could have for any positive number $\eta$, there exists a general constant $c_1$ s.t.

$$\begin{aligned} \mathbb{P}(|y_i| \geqslant \eta) &\leqslant 2\exp\left[-c_1 \min\left\{-\frac{\eta}{\varphi_2^2\|M\|}, \frac{\eta^2}{\varphi_2^4\|M\|_F^2}\right\}\right] \\ &\leqslant 2\exp\left[-c_1 \min\left\{-\frac{\eta}{\varphi_2^2}, \frac{\eta^2}{\varphi_2^4}\right\}\right]. \end{aligned}$$

Then by Lemma D.2, we could find a constant $C$ depending on sub-Gaussian norm and general constant $c_1$ s.t.

$$\mathbb{E}|y_i|^k \leqslant (Ck)^k,$$

where in fact we could give the explicit form of $C$ as

$$C = 1 + \frac{c_1}{\min\{\varphi_2^2, \varphi_2^4\}}. \tag{C.2}$$

Notice $y_i$ has mean zero and variance 1 (assuming $\|x\| = 1$), then apply Lemma D.3, we could assert that there exists a general constant $c_2$

$$\mathbb{P}\left(\left|\frac{1}{k}\mathbf{y}^\top \mathbf{I}_{k,k}\mathbf{y} - 1\right| \geqslant \epsilon\right) \leqslant C\exp\left(-c_2\left[\sqrt{k}\epsilon\right]^{1/4}\right),$$

where $C$ is defined in (C.2) and we use the fact $\alpha = 1$ in our case which is defined in moments condition.

□

**Lemma C.1.** *For a linear mapping from $\mathbb{R}^d \to \mathbb{R}^k$: $f(\mathbf{x}) = \frac{1}{\sqrt{k}}\mathbf{\Omega}\mathbf{x}$,*

$$\mathbb{P}(|\langle f(\mathbf{x}), f(\mathbf{y})\rangle - \langle \mathbf{x}, \mathbf{y}\rangle| \geqslant \epsilon|\langle \mathbf{x}, \mathbf{y}\rangle|) \leqslant 2\sup_{\mathbf{x}\in\mathbb{R}^d}\mathbb{P}(|\|f(\mathbf{x})\|^2 - \|\mathbf{x}\|^2| \geqslant \epsilon\|\mathbf{x}\|_2^2).$$

*Proof.* Since $f$ is a linear mapping, we have

$$4f(\mathbf{x})f(\mathbf{y}) = \|f(\mathbf{x}+\mathbf{y})\|_2^2 - \|f(\mathbf{x}-\mathbf{y})\|_2^2.$$

Consider the event

$$\mathcal{A}_1 = \left\{|\|f(\mathbf{x}+\mathbf{y})\|_2^2 - \|\mathbf{x}+\mathbf{y}\|_2^2| \geqslant \epsilon\|\mathbf{x}+\mathbf{y}\|_2^2\right\}$$
$$\mathcal{A}_2 = \left\{|\|f(\mathbf{x}-\mathbf{y})\|_2^2 - \|\mathbf{x}-\mathbf{y}\|_2^2| \geqslant \epsilon\|\mathbf{x}-\mathbf{y}\|_2^2\right\}$$

On the event $\mathcal{A}_1^\complement \cap \mathcal{A}_2^\complement$,

$$4f(\mathbf{x})f(\mathbf{y}) \geqslant (1-\epsilon)(\mathbf{x}+\mathbf{y})^2 - (1+\epsilon)(\mathbf{x}-\mathbf{y})^2 = 4\langle \mathbf{x}, \mathbf{y}\rangle - 2\epsilon(\|\mathbf{x}\|^2 + \|\mathbf{y}\|^2),$$

noticing $\|\mathbf{x}\|^2 + \|\mathbf{y}\|^2 \geqslant 2\langle \mathbf{x}, \mathbf{y}\rangle$, and by similar argument on the other side of the inequality, we could claim that

$$\{|\langle f(\mathbf{x}), f(\mathbf{y})\rangle - \langle \mathbf{x}, \mathbf{y}\rangle| \geqslant \epsilon|\langle \mathbf{x}, \mathbf{y}\rangle|\} \subseteq \mathcal{A}_1 \cup \mathcal{A}_2.$$

Then we finish the proof by simply applying an union bound of two events. □

**Remark.** *The key element of classic random projections is the dimension-free bound. Similarly, according to Prop. 3.5, our TRP has a norm preservation bound independent of the particular vector $\mathbf{x}$ and dimension $d$ and thus a dimension-free inner product preservation bound according to Lemma C.3.*

# Appendix D Technical Lemmas

In this section, we list some technical lemmas we use in this paper. All of them are about tail probability of sub-Gaussian or generalized sub-exponential variables.

**Definition D.1.** *A random variable $x$ is called sub-Gaussian if $\mathbb{E}|x|^p = \mathcal{O}(p^{p/2})$ when $p \to \infty$. With this, we define sub-Gaussian norm for $x$ (less than infinity) as*

$$\|x\|_{\varphi_2} = \sup_{p \geqslant 1} p^{-1/2} (\mathbb{E}|x|^p)^{1/p}. \tag{D.1}$$

Note that for Bernoulli random variable, i.e., $\{-1, 1\}$ with prob. $\{\frac{1}{2}, \frac{1}{2}\}$, $\varphi_2 = 1$; any bounded random variable with absolute value less than $M > 0$ has $\varphi_2 \leqslant M$. For standard Gaussian random variable, $\varphi_2 = 1$.

**Lemma D.1.** *(Hanson-Wright Inequality) Let $\mathbf{x} = (x_1, \cdots, x_n) \in \mathbb{R}^n$ be a random vector with independent components $X_i$ which satisfies $\mathbb{E}\mathbf{x}_i = 0$ and $\varphi_2(x_1) \leqslant K$. Let $A$ be an $n \times n$ matrix. Then, for every $\eta \geqslant 0$, there exists a general constant $c$ s.t.*

$$\mathbb{P}\left(|\mathbf{x}^\top \mathbf{A}\mathbf{x} - \mathbb{E}\mathbf{x}^\top \mathbf{A}\mathbf{x}| \geqslant \eta\right) \leqslant 2\exp\left[-c\min\left\{\frac{\eta}{K^2\|A\|}, \frac{\eta^2}{\|A\|_F^2 K^4}\right\}\right].$$

*Proof.* Please refer to Rudelson et al. [2013] □

**Lemma D.2.** *Let $\mathbf{x}$ be a random variable whose tail probability satisfies for every $\eta \geqslant 0$, there exists a constant $c_1$ s.t.*

$$\mathbb{P}\left(|x| \geqslant \eta\right) \leqslant 2\exp\left[-c_1 \min\left(\eta, \eta^2\right)\right].$$

*Then for any $k \geqslant 1$, $x$ satisfies generalized sub-exponential moment condition C.1 with $\alpha = 1$, i.e.,*

$$\mathbb{E}|x|^k \leqslant (Ck)^k,$$

*where $C = 1 + \frac{1}{c_1}$.*

*Proof.*

$$\begin{aligned}
\mathbb{E}|x|^k &= \int_0^1 k x^{k-1} 2\exp[-c_1 x^2] dx + \int_1^\infty k x^{k-1} 2\exp[-c_1 x] dx \\
&\leqslant 1 + + \frac{1}{c_1^k} \int_0^\infty k y^{k-1} 2\exp[-y] dy \\
&= 1 + \frac{1}{c_1^k} k\Gamma(k-1) \leqslant \left[1 + \frac{1}{c_1^k}\right] k^k.
\end{aligned} \tag{D.2}$$

Noticing $\left[1 + \frac{1}{c_1^k}\right]^{1/k} \leqslant 1 + \frac{1}{c_1}$, we finish the proof. □

**Lemma D.3.** *For a random vector* $\mathbf{x}$ *with each element independent and identically distributed with mean zero and variance 1, suppose each element of* $\mathbf{x}$ *satisfies generalized sub-exponential moment condition as in* (D.2)*, that there exists a general constant* $C$ *s.t.* $\mathbb{E}|x_1|^k \leqslant (Ck)^{\alpha k}$*. Then for any matrix* $\mathbf{A} \in \mathbb{R}^{n\times n}$*, there exists a general constant* $c_1$

$$\mathbb{P}\left(\left|\mathbf{x}^\top \mathbf{A}\mathbf{x} - \mathbb{E}\mathbf{x}^\top \mathbf{A}\mathbf{x}\right| \geqslant \eta\right) \leqslant C \exp\left(-c_1 \left[\frac{\eta}{\|A\|_F}\right]^{1/(2(1+\alpha))}\right).$$

*Proof.* The proof is directly from Lemma 8.3 in Buhler and Tompa [2002] and we change the statement on generalized sub-exponential R.V. directly to the statement on the moment condition. □