Johnson-Lindenstrauss Transforms with Best Confidence

Maciej Skorski   MACIEJ.SKORSKI@GMAIL.COM
University of Luxembourg

Editors: Mikhail Belkin and Samory Kpotufe

Abstract

The seminal result of Johnson and Lindenstrauss on random embeddings has been intensively studied in applied and theoretical computer science. Despite that vast body of literature, we still lack of complete understanding of statistical properties of random projections; a particularly intriguing question is: why are the theoretical bounds that far behind the empirically observed performance?

Motivated by this question, this work develops Johnson-Lindenstrauss distributions with optimal, data-oblivious, statistical confidence bounds. These bounds are numerically best possible, for any given data dimension, embedding dimension, and distortion tolerance. They improve upon prior works in terms of statistical accuracy, as well as exactly determine the no-go regimes for data-oblivious approaches. Furthermore, the projection matrices are efficiently samplable.

The construction relies on orthogonal matrices, and the proof uses certain elegant properties of the unit sphere. In particular, the following techniques introduced in this work are of independent interest: a) a compact expression for the projection distortion in terms of singular eigenvalues of the projection matrix, b) a parametrization linking the unit sphere and the Dirichlet distribution and c) anti-concentration bounds for the Dirichlet distribution.

Besides the technical contribution, the paper presents applications and numerical evaluation along with working implementation in Python (shared as a GitHub repository).

Keywords: Random Projections, Johnson-Lindenstrauss Lemma, Minimax Risk

1. Introduction

The seminal result of Johnson and Lindenstrauss (1984) on random embeddings is the cornerstone tool in dimension reduction. It rigorously shows that euclidean distances are nearly preserved (low distortion) when high-dimensional data are projected into a lower-dimensional space using a random (appropriately sampled) matrix. What makes the random projections preferable in applications, over other dimension reductions techniques such as the principal component analysis or the singular value decomposition, are the speed, data-independence, and much stronger statistical guarantees (see Menon (2007); Akselrod-Ballin et al. (2011); Vu (2016); Bandeira et al. (2017)).

The low-distortion property of random projections is very appealing and makes them popular across many research areas. Among many applications one finds topics as diverse as functional analysis (Johnson and Naor (2010)), combinatorics (Frankl and Maehara (1988)), signal processing (Haupt and Nowak (2006)), proximity search Ailon and Chazelle (2006); Indyk and Motwani (1998), low-rank matrix approximations (Nguyen et al. (2009); Clarkson and Woodruff (2017); Ubaru et al. (2017)), learning theory (Arriaga and Vempala (2006)), feature hashing (Weinberger et al. (2009)), image hashing (Lv and Wang (2008)), classification (Rahimi et al. (2007); Ghalib et al. (2020)), regression (Maillard and Munos (2012)), face recognition (Goel et al. (2005)), text mining (Bingham and Mannila (2001); Lin and Gunopulos (2003); Ben-David et al. (2007)), clustering (Boutsidis et al. (2010); Tasoulis et al. (2014); Boutsidis et al. (2014); Makarychev et al.

(2019); Becchetti et al. (2019)), data storage (Candes (2008); Cormode and Indyk (2016)) and privacy (Blocki et al. (2012); Kenthapadi et al. (2013)).

Our focus is on the *statistical guarantees* of the *Distributional Johnson-Lindenstrauss Lemma*. The result states that for every data dimension m and the embedding dimension n there exists a random matrix A of shape $n \times m$ (explicitly samplable), such that for every non-zero data input $x \in \mathbb{R}^m$, and relative tolerance $0 < \epsilon < \frac{1}{2}$, the euclidean norm is ϵ-preserved with high confidence, provided that n is sufficiently big. Formally:

$$\mathbb{P}_A \left[\left| \|Ax\|_2^2 - \|x\|_2^2 \right| > \epsilon \|x\|_2^2 \right] \leqslant \mathrm{e}^{-\Omega(n\epsilon^2)}. \tag{1}$$

This work tackles the challenge of constructing random projections with the *smallest possible distortion probability* defined as above, given data dimension m, embedding dimension n, and distortion ϵ.

1.1. Related Work

There have been several works on simplifying the proof and improving the provable confidence, namely: Johnson and Lindenstrauss (1984); Frankl and Maehara (1988); Indyk and Motwani (1998); Achlioptas (2003); Dasgupta and Gupta (2003); Matoušek (2008). The best, up to date, upper bound for the distortion probability above is $2 \exp \left(-\frac{n\epsilon^2}{4} \cdot \left(1 - \frac{2\epsilon}{3} \right) \right)$ Indyk and Motwani (1998); Achlioptas (2003) achieved for scaled Gaussian or Rademacher matrices. ; in other words the exponent is nearly $\frac{1}{4}$ for small distortions ϵ. As for the impossibility results, we know that no distribution with $n < m/2$ can achieve distortion probability smaller than $\exp(-O(n\epsilon^2 + 1))$ for some unspecified constant (Alon (2003); Kane et al. (2011); Jayram and Woodruff (2013)). Regarding this hidden constant, it has been recently shown in Burr et al. (2018) that it cannot be better, for any construction, than $\frac{1}{4}$ in some restricted asymptotic regimes, namely when $\epsilon \to 0, n\epsilon^2 \to +\infty$ and when $\frac{n}{m} \to 0$. Interestingly, the evaluation experiments (see for example Venkatasubramanian and Wang (2011); Fedoruk et al. (2018)), found theoretical guarantees far behind the observed performance.

The above discussion summarizes the state-of-art on confidence bounds, which is the subject of this paper. However, for readers interested in a broader scope of research on random projections, we would like to briefly discuss other lines of research. There are many works on trading the statistical accuracy for certain algorithmic properties, such as sparsity and faster sampling (Dasgupta et al. (2010); Ailon and Liberty (2013); Kane and Nelson (2014); Cohen et al. (2018)) or specific matrix patterns (Allen-Zhu et al. (2014); Freksen and Larsen (2020)); these properties can be somewhat improved under certain structural properties of datasets, if known in advance (Bourgain et al. (2015)). Another trade-off is to extend the class of sampling distributions as much as possible (e.g. sub-gaussian matrices) as done by Matoušek (2008); Boucheron et al. (2003).

We also note that for certain datasets and for some parameter regimes, it is possible to slightly improve upon the DJL Lemma using *non-random* embeddings constructed combinatorially (Nelson et al. (2014); Larsen and Nelson (2016)); these however are more of theoretical interests, and generally in theory and practice DJL constructions are preferred, because of their *data oblivious* properties (particularly useful for streaming, distributed and parallel computing).

2. Results

2.1. Main Result: Characterizing Best Confidence

We study the optimal error probability in (1), given as the min-max program:

$$\delta^{\text{Best}}(m, n, \epsilon) = \inf_{\mathcal{A} \in \mathbb{P}(\mathbb{R}^{n \times m})} \sup_{x \in \mathbb{R}^m} \mathbb{P}_{A \sim \mathcal{A}} \left[|\|Ax\|_2^2 - \|x\|_2^2| > \epsilon \|x\|_2^2 \right]. \tag{2}$$

Since we maximize over the m-dimensional data inputs x, and minimize over all the possible sampling distributions $\mathcal{A}$ for a projection from the dimension m to n, this gives the best possible confidence bounds for the (oblivious) DJL Lemma. It may be convenient to think of this program as a two-player game: we seek for the best projection (minimizing over the distribution of the matrix), while the adversary controls the data and is seeking for the malicious input.

Our main contribution shows that the ideal bound above is achievable, characterizes it, and develops an explicit sampler. Note that even the existence of a distribution achieving exactly (not approximately) the best bound is not trivial, because we deal with doubled optimization including distributions with unbounded support. As for the significance, our optimal confidence bound completes the line of research on improving the data-oblivious Distributional JL Lemma, establishing the numerically (not asymptotically) sharp no-go result.

We explain the notation, before stating our result. By $\text{Beta}(a, b)$ we denote the Beta distribution with shape parameters a, b; it has the cumulative distribution $B(z; a, b)/B(1; a, b)$ where the incomplete Beta function is defined as $B(z; a, b) \triangleq \int_0^z z^{a-1}(1-z)^{b-1} \mathrm{d}z$ ((DLMF, 8.17)). By $\mathcal{O}(d)$ we denote the set of orthogonal matrices of shape $d \times d$ and $I_{n,m}$ denotes the matrix of shape $n \times m$ with ones on the principal diagonal and zeros elsewhere (generalizing the identity matrix).

Theorem 1 (Best Oblivious DJL Confidence) *Let $1 \leqslant n < m$ be integers, and $0 < \epsilon < \frac{1}{2}$. Then the best value in (2) is achievable and equals:*

$$\delta^{\text{Best}}(m, n, \epsilon) = 1 - \max_{\lambda} \mathbb{P} \left[(1 - \epsilon)\lambda \leqslant \text{Beta} \left(\frac{n}{2}, \frac{m-n}{2} \right) \leqslant (1 + \epsilon)\lambda \right]. \tag{3}$$

Furthermore, let λ be the maximizer of the right-hand side, $U \sim \mathcal{O}(n)$, and $V \sim \mathcal{O}(m)$ be sampled uniformly and independently. Then

$$A^{\text{Best}} = \lambda^{-1/2} \cdot U \cdot I_{n,m} \cdot V^T \tag{4}$$

is the random matrix which achieves the best value $\delta^{\text{Best}}(m, n, \epsilon)$.

The prior results on confidence and our formula are summarized in Table 1. The step-by-step sampler construction is presented in Algorithm 1.

Remark 2 (Optimality under Worst vs Average Choice) *We define the best confidence in terms of a min-max problem, so that the construction is optimal under the* worst choice *of the input. However, the proof actually establishes more, namely that the confidence cannot be improved even under the* average choice *from the unit sphere (related to the use of Yao's Min-Max Principle). The uniform distribution on the sphere can be seen as the* non-informative prior*, perfectly suited to model the oblivious setup (only the input scale is known).*

Author	Error Upper Bound	Error Lower Bound	Restriction
Johnson and Lindenstrauss (1984)	$2\exp(-\Omega(n\epsilon^2))$	?	
Frankl and Maehara (1988)	$2\exp\left(-\frac{n\epsilon^2}{9}(1+O(\epsilon))\right)$	?	
Indyk and Motwani (1998); Achlioptas (2003)	$2\exp\left(-\frac{n\epsilon^2}{4}(1+O(\epsilon))\right)$	?	
Alon (2003)		$\exp(-O(n\epsilon^2\log(\frac{1}{\epsilon})))$	$n < \frac{m}{2}$
Kane et al. (2011)		$\exp(-O(n\epsilon^2))$	$n < \frac{m}{2}$
Jayram and Woodruff (2013)		$\exp(-O(n\epsilon^2))$	$n < \frac{m}{2}$
Burr et al. (2018)		$\exp\left(-\frac{n\epsilon^2}{4}(1+o(1))\right)$	$n\epsilon^2\to\infty$ $\frac{n}{m}\to 0$
this work	$1-\max_\lambda \mathbb{P}\left[(1-\epsilon)\lambda \leqslant \mathsf{Beta}\left(\frac{n}{2},\frac{m-n}{2}\right)\leqslant (1+\epsilon)\lambda\right]$		None

Table 1: Confidence error $\delta^{\mathrm{Best}}(m,n,\epsilon)$ (Distortion Probability) in DJL.

Input: Data dimension m, Embedding dimension n, Tolerance ϵ
Result: Best Oblivious DJL Matrix A^{Best}

```
   /* find the scaling factor                                        */
```
$\lambda \leftarrow \mathrm{argmax}_\lambda \mathbb{P}\left[(1-\epsilon)\lambda \leqslant \mathsf{Beta}\left(\frac{n}{2},\frac{m-n}{2}\right)\leqslant (1+\epsilon)\lambda\right]$
```
   /* sample orthogonal matrices, uniformly & independently          */
```
$V \sim \mathcal{O}(m), U \sim \mathcal{O}(n)$
```
   /* build the projection                                           */
```
$A \leftarrow \lambda^{-\frac{1}{2}} U I_{n,m} V^T$
return A

Algorithm 1: Best Sampler for Oblivious DJL Lemma.

Remark 3 (Parameter regimes) *Note that $\delta^{\mathrm{Best}}(m,n,\epsilon) = 0$ for $n \geqslant m$ trivially (take the identity matrix as A); this regime is not interesting, as there is no dimension reduction. The interesting cases $n < m$ are fully addressed by the result; note that the optimal distribution depends on m, n, ϵ.*

Remark 4 (Construction) *The sampler is built on appropriately scaled orthogonal matrices; there exist algorithms for efficiently sampling such matrices, see for example Stewart (1980); Genz (2000). The scaling factor is chosen carefully as a solution to the one-dimensional numerical optimization problem (3); since the objective derivative can be explicitly calculated and the optimal point lies in the interval $\left(0, \frac{1}{1-\epsilon}\right)$, the program can be readily solved by modern data-science software, for example using* R *or* Python*. Below in Listing 1 we demonstrate the implementation in the* SciPy *library for Python[1]. The optimization task can be best explained and interpreted geometrically, as visualized in Figure 1.*

Remark 5 (Closed-form Approximation) *We have exactly characterized best confidence. However, we mention the following, more readable and nearly sharp, convenient bounds:*

$$\min\left\{\mathbb{P}[B > (1+\epsilon)\mathbb{E}[B]],\ \mathbb{P}[B < (1-\epsilon)\mathbb{E}[B]]\right\} \leqslant \delta^{\mathrm{Best}}(m,n,\epsilon)$$
$$\delta^{\mathrm{Best}}(m,n,\epsilon) \leqslant \mathbb{P}[B > (1+\epsilon)\mathbb{E}[B]] + \mathbb{P}[B < (1-\epsilon)\mathbb{E}[B]],$$
(5)

---

1. The full implementation and all examples is available at https://github.com/maciejskorski/confidence_optimal_random_embed

where we denote $B = \mathsf{Beta}\left(\frac{n}{2}, \frac{m-n}{2}\right)$. *The proof of this fact appears later, in the discussion of applications.*
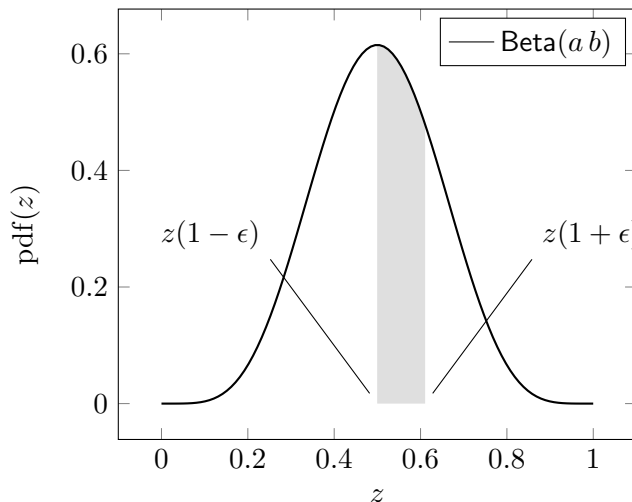

Figure 1: The maximal statistical error equals the maximal area under the beta probability density, captured by an interval of form $[\lambda(1-\epsilon), \lambda(1+\epsilon)]$. The beta shape is $a = \frac{n}{2}, b = \frac{m-n}{2}$. In the picture $n = 10, m = 20$, the optimal value is $\lambda \approx \frac{5}{9}$ for small ϵ.

Listing 1: Numerically Finding Optimal Confidence and Scaler (3)

```python
from scipy.optimize import minimize
from scipy.stats import beta

def optimal_djl(m,n,eps):
    ''' confidence-optimal sampler for DJL'''
    a = n/2
    b = (m-n)/2
    z0 = a/(a+b)
    dist = beta(a,b)
    fun = lambda z:-dist.cdf((1+eps)*z)+dist.cdf((1-eps)*z)
    betainc_jac = lambda z: dist.pdf(z)
    jac = lambda z:-(1+eps)*betainc_jac((1+eps)*z)+(1-eps)*betainc_jac((1-eps)*z)
    out = minimize(fun,x0=z0,jac=jac,method='Newton-CG')
    scale,delta = out.x,1+out.fun
    return scale,delta
```

2.2. Techniques of Independent Interest

The proof of Theorem 1 builds on three elegant facts of broader interest. Below we abstract them as independent results and discuss in more detail.

2.2.1. EXPLICIT DISTORTION WITH LATENT SINGULAR VALUES

Since the matrix product scales linearly with the input norm, the DJL Lemma reduces to the question about measure concentration on the unit sphere. In fact we know that random sphere points tend to

be "hardest" (giving the worst confidence) for the DJL Lemma, as shown by an application of the Yao (Kane et al. (2011); Burr et al. (2018)). The core of our approach is the observation that the distortion on the random sphere point can be very conveniently expressed (as a diagonal quadratic form) in terms of the matrix singular eigenvalues. This is formally stated below.

Theorem 6 *Let $X = (X_1, \ldots, X_m)$ be uniformly distributed on the unit sphere. Let A be any $n \times m$ random matrix independent of X, where $n \leqslant m$, and $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of AA^T. Then the following holds:*

$$\|AX\|_2^2 \sim \sum_{k=1}^n \lambda_k X_k^2. \tag{6}$$

Remark 7 *The matrix AA^T is positive semi-definite (it is the so called Grammian matrix Deza and Laurent (1997)), so the eigenvalues λ_k are non-negative.*

2.2.2. SPHERE SAMPLING WITH DIRICHLET DISTRIBUTION

To effectively handle calculations on the unit sphere, we develop the parametrization linking it to the *Dirichlet Distribution*. This is a novelty in the context of other works that used complicated sphere paramaterizations, formally justified by calculus on differential forms (Kane et al. (2011); Burr et al. (2018)), and in a wider context of sphere samplers, very important to Monte Carlo methods, as it is not addressed by extensive surveys (Roberts (2019)). In the theorem below, by Dirichlet(α) we denote the Dirichlet distribution with the vector parameter α.

Theorem 8 *Let $(X_1, \ldots, X_m)$ be uniform on the unit sphere in $\mathbb{R}^m$. Then:*

$$(X_1^2, \ldots, X_m^2) \sim \mathsf{Dirichlet}\left(\frac{1}{2}\mathbf{1}_m\right) \tag{7}$$

where $\mathbf{1}_m$ denotes the vector of m ones.

Remark 9 *Let $Z_k = X_k^2$ and $(\varepsilon_k)_k$ be independent Rademacher variables (that is ± 1 with equal probability). Then $(\varepsilon_k \sqrt{Z_k})_k$ is uniform on the unit sphere, as illustrated by the numerical simulation shown in Figure 2.*

2.2.3. ANTI-CONCENTRATION OF DIRICHLET DISTRIBUTION

The following result establishes *sharp anti-concentration bounds* for weighted sums of components of Dirichlet's distribution. This is of broader interest due to the popularity of Dirichlet distribution in statistics; such weighted sums appear in many applications (for a detailed discussion, see for example Provost and Cheong (2000)).

Theorem 10 *Let $Z = (Z_1, \ldots, Z_m)$ follow the Dirichlet distribution with parameters $(\alpha_1, \ldots, \alpha_m)$. Let $W = (W_1, \ldots, W_m)$ be a vector of any non-negative random variables independent of Z. Then for any non-empty and strict subset I of $\{1 \ldots m\}$ and real numbers $0 < p < q$ the following holds:*

$$\min_W \mathbb{P}\left[\sum_{k \in I} W_k Z_k \notin [p, q]\right] = 1 - \max_{z > 0}\left[pz \leqslant \mathsf{Beta}(a, b) \leqslant qz\right], \tag{8}$$

where $a = \sum_{i \in I} \alpha_i$ and $b = \sum_{i \notin I} \alpha_i$, and $\mathsf{Beta}(a, b)$ is the Beta distribution.
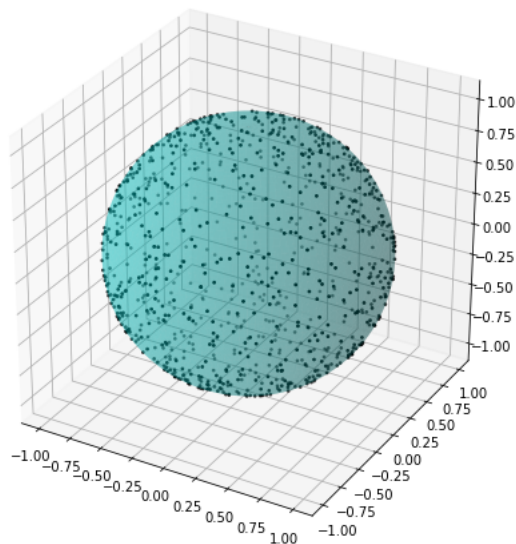
Figure 2: Sampling the unit sphere according to Theorem 8, in $m = 3$ dimensions.

2.3. Applications

2.3.1. DIMENSION ESTIMATION FOR DATA SCIENCE USAGE OF RANDOM PROJECTIONS

A good practice is to conservatively estimate the dimension before compressing the data with random projections. The DJL Lemma combined with a a union bound gives then provable guarantees. Such tests are implemented in modern data-science software, for example in the popular Scikit-learn library for Python (Pedregosa et al. (2011)). The problem with currently available bounds is that they are based on overly conservative estimates from prior works, which creates the false impression that random projections should not be used. Our bounds give the more accurate answer, as illustrated in Figure 3 (the best previous bounds used for comparison are from Indyk and Motwani (1998); Achlioptas (2003)). The Python code is available in Appendix A.

2.3.2. DEPENDENCY ON FEATURE DIMENSION

Prior works have studied versions of DJL Lemma that are *data-dimension* independent. However, even if we make no prior assumptions on the data structure, its dimension is known; is thus interesting to see the impact of the data dimension. This impact can be seen with the help of our optimal bounds; below in Figure 4 we show that knowing the data dimension helps improving the bound, with considerable impact when the dimension is of moderate magnitude.

2.3.3. CLOSED-FORM UPPER BOUNDS ON CONFIDENCE

Let $B = \text{Beta}\left(\frac{n}{2}, \frac{m-n}{2}\right)$. Instead of optimizing numerically the bound in Theorem 1, let us specialize $\lambda = \mathbb{E}[B]$. This way we obtain the following convenient upper-bound

$$\delta^{\text{Best}}(m, n, \epsilon) \leqslant \Pr[|B - \mathbb{E}[B]| > \epsilon \mathbb{E}[B]], \tag{9}$$
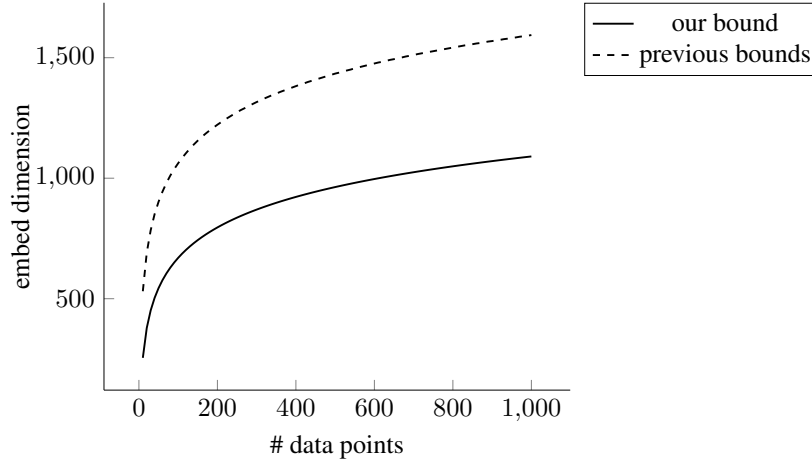
Figure 3: The minimal embedding dimension n which guarantees distortion $\epsilon \leqslant 0.2$ for all pairwise distances of the given number of data points x.
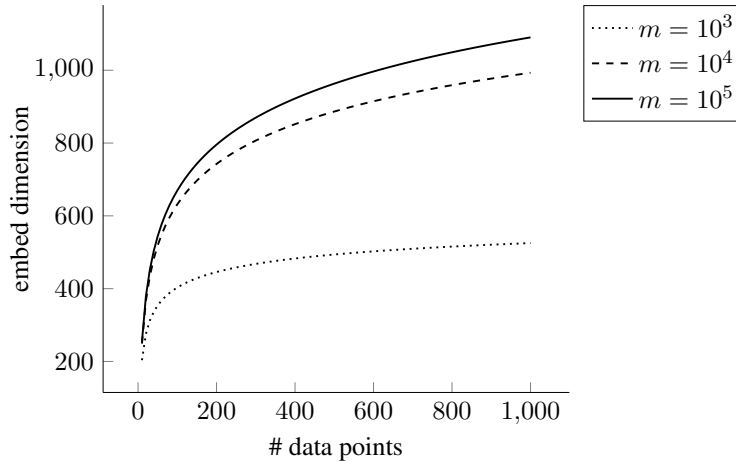

Figure 4: The minimal embedding dimension n which guarantees distortion $\epsilon \leqslant 0.2$ for all pairwise distances of the given number of data points x. The results are better (that is, the embedding dimension is smaller) when the data dimension m is smaller.

which is very close to the optimal value, but does not involve optimization. The comparison of the approximate and exact bounds is given in Figure 5.

2.3.4. LOWER BOUNDS (IMPOSSIBILITY RESULTS) ON CONFIDENCE

Again, let $B = \mathsf{Beta}\left(\frac{n}{2}, \frac{m-n}{2}\right)$. Then $\delta^{\mathrm{Best}}(m, n, \epsilon) = \min_\lambda \Pr[B \notin [\lambda(1-\epsilon), \lambda(1+\epsilon)]]$. Further, $\Pr[B \notin [\lambda(1-\epsilon), \lambda(1+\epsilon)]] = \Pr[B < (1-\epsilon)\lambda] + \Pr[B > (1+\epsilon)\lambda]$. Considering that for any λ
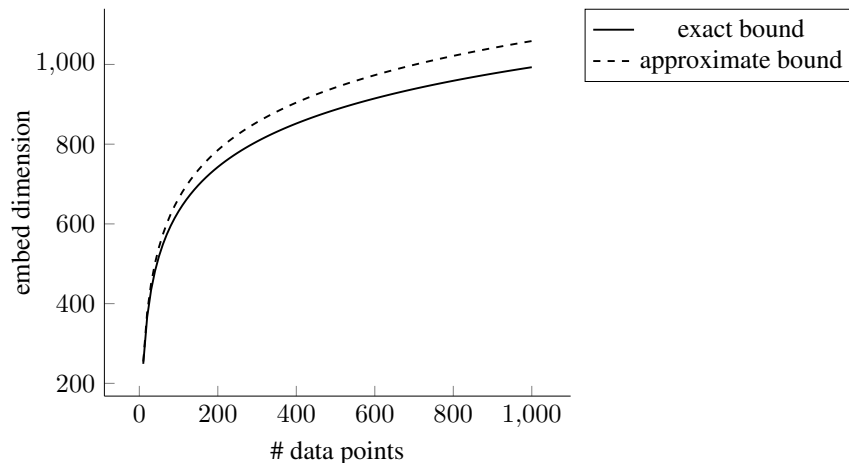
Figure 5: The minimal embedding dimension n with distortion $\epsilon \leqslant 0.2$ for all pairwise distances of the data points: exact (Theorem 1) and approximate (Theorem 5) bounds.

it holds that either $\lambda \leqslant \mathbb{E}[B]$ or $\lambda > \mathbb{E}[B]$, we obtain:

$$\delta^{\text{Best}}(m, n, \epsilon) \geqslant \min\left\{\mathbb{P}[B > (1 + \epsilon)\mathbb{E}[B]], \mathbb{P}[B < (1 - \epsilon)\mathbb{E}[B]]\right\}. \tag{10}$$

This lower bound implies impossibility results obtained in prior works by Kane et al. (2011) and Burr et al. (2018) (when combined with accurate approximations for tails of the beta distribution, such as those in (Zhang and Zhou (2020))).

Note that taking into account the previous upper bound, we prove Theorem 5.

2.3.5. USE AS BENCHMARK

The fact that our result is numerically optimal for data-oblivious setup, and also easy to explicitly compute, makes it a perfect reference tool. When comparing the theoretical and empirical performance (such as in works of Venkatasubramanian and Wang (2011); Fedoruk et al. (2018)), our Theorem 1 now clarifies how big that gap actually is. Similarly for theoretical research, quantifying the best possible oblivious bound serves as a reference point for non-oblivious approaches, and also determines the range of possible improvements.

3. Proofs

3.1. Proof of Theorem 1

Define

$$g(m, n, \epsilon) \triangleq 1 - \max_\lambda \mathbb{P}\left[\mathsf{Beta}\left(\frac{n}{2}, \frac{m - n}{2}\right) \in [\lambda(1 - \epsilon), \lambda(1 + \epsilon)]\right]. \tag{11}$$

It suffices to show that

$$\inf_{\mathcal{A} \in \mathbb{P}(\mathbb{R}^{n \times m})} \sup_{x \in \mathbb{R}^m} \mathbb{P}_{A \sim \mathcal{A}}\left[|\|Ax\|_2^2 - \|x\|_2^2| > \epsilon\|x\|_2^2\right] = g(m, n, \epsilon). \tag{12}$$

9

We first prove that for $A = A^{\text{Best}}$ sampled as described in Theorem 1, and every fixed m-dimensional non-zero vector x the following holds:

$$\mathbb{P}_A \left[|\|Ax\|_2^2 - \|x\|_2^2| > \epsilon \|x\|_2^2 \right] = g(m, n, \epsilon), \tag{13}$$

which in turn implies that the inequality "$\leqslant$" holds in (12).

We observe that the condition under the probability is homogeneous (quadratic in $\|x\|_2$), thus we can restrict $\|x\|_2 = 1$. Now x is on the unit sphere $\mathbb{S}^{m-1}$, and thus $X = V^T x$ is uniformly distributed on $\mathbb{S}^{m-1}$, because V is orthogonal. We can write $Ax = A'X$ where $A' = \lambda^{-\frac{1}{2}} \cdot UI_{m,n}$. Then $A'A'^T = \lambda^{-1} UI_{n,m}I_{n,m}^T U^T = \lambda^{-1} UI_{n,n}U^T = \lambda^{-1}UU^T = \lambda^{-1}I_{n,n}$, where we used the structure of $I_{n,m}$ and the orthogonality of U. Now by Theorem 6 we obtain:

$$\|Ax\|_2^2 \sim \lambda^{-1} \sum_{k=1}^{n} X_k^2. \tag{14}$$

Combining this with Theorem 8, for $Z \sim \text{Dirichlet}\left(\frac{1}{2}\mathbf{1}_m\right)$ we obtain that for every non-zero x:

$$\mathbb{P}_A \left[|\|Ax\|_2^2 - \|x\|_2^2| > \epsilon \|x\|_2^2 \right] = 1 - \mathbb{P}\left[1 - \epsilon \leqslant \lambda^{-1} \sum_{k=1}^{n} Z_k \leqslant 1 + \epsilon \right]. \tag{15}$$

By the properties of the Dirichlet distribution (Albert and Denis (2012)) we have that $\sum_{k=1}^{n} Z_k \sim \text{Beta}(a, b)$ with $a = \frac{n}{2}$ and $b = \frac{m-n}{2}$. Thus, by the definition of λ, we conclude that the right-side equals $g(m, m, \epsilon)$.

In the second part we show that for every *fixed* matrix A we have:

$$\mathbb{P}_{x \sim \mathbb{S}^{m-1}} \left[|\|Ax\|_2^2 - \|x\|_2^2| > \epsilon \|x\|_2^2 \right] \geqslant g(m, n, \epsilon), \tag{16}$$

where $\mathbb{S}^{m-1}$ denotes unit sphere in m-dimensions; this establishes the inequality "$\geqslant$" in (12) (by replacing the expectation over x with the maximum and taking the expectation over the distribution of A). Let X be uniform on $\mathbb{S}^{m-1}$. Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of AA^T (they are deterministic numbers). Then:

$$\|AX\|_2^2 \sim \sum_{k=1}^{n} \lambda_k X_k^2. \tag{17}$$

Combining this with Theorem 8, for $Z \sim \text{Dirichlet}\left(\frac{1}{2}\mathbf{1}_m\right)$ we obtain:

$$\begin{aligned}
\mathbb{P}_{x \sim \mathbb{S}^{m-1}} \left[|\|Ax\|_2^2 - \|x\|_2^2| > \epsilon \|x\|_2^2 \right] &= 1 - \mathbb{P}\left[1 - \epsilon \leqslant \sum_{k=1}^{n} \lambda_k Z_k \leqslant 1 + \epsilon \right] \\
&= \mathbb{P}\left[\sum_{k=1}^{n} \lambda_k Z_k \notin [1 - \epsilon, 1 + \epsilon] \right].
\end{aligned} \tag{18}$$

Regardless of the choice of λ_k, by Theorem 10 we get the lower bound:

$$\mathbb{P}_{x \sim \mathbb{S}^{m-1}} \left[|\|Ax\|_2^2 - \|x\|_2^2| > \epsilon \|x\|_2^2 \right] \geqslant 1 - \max_{\lambda} \mathbb{P}\left[\lambda(1 - \epsilon) \leqslant B \leqslant \lambda(1 + \epsilon) \right], \tag{19}$$

where $B = \text{Beta}\left(\frac{n}{2}, \frac{m-n}{2}\right)$. This completes the proof, since the expression on the right-hand side equals $g(m, n, \epsilon)$.

3.2. Proof of Theorem 8

The Dirichlet distribution $Z = (Z_1, \ldots, Z_m)$ with parameters $\frac{1}{2}\mathbf{1}_m$ can be sampled as $Z_k = \Gamma_k / \sum_{i=1}^n \Gamma_i$, where Γ_i are independent and follow the Gamma distribution with the parameters: shape $a = \frac{1}{2}$ and the rate $b = 1$ (Albert and Denis (2012)). We next observe that $\Gamma_i \sim \frac{1}{2}\chi_1$, where χ_1 is the chi-squared distribution with 1 degree of freedom (Thom (1958)). By definition, $\chi_1 \sim N^2$ where N is the standard normal random variable. Therefore, we obtain:

$$Z_k \sim \frac{\frac{1}{2}N_k^2}{\sum_{i=1}^m \frac{1}{2}N_i^2} = \frac{N_k^2}{\sum_{i=1}^m N_i^2}, \quad N_i \sim^{iid} \mathsf{Norm}(0,1). \tag{20}$$

We now recall that the normalized normal vector generates the uniform measure on the sphere (see Muller (1959); Marsaglia et al. (1972).); more precisely if $(X_1, \ldots, X_m)$ is the uniform distribution on the sphere, then

$$\left(\frac{N_k}{\sqrt{\sum_{i=1}^m N_i^2}} \right)_{k=1}^m \sim (X_k)_{k=1}^m, \tag{21}$$

and combining this with the previous equation we get

$$(Z_k)_{k=1}^m \sim (X_k^2)_{k=1}^m, \tag{22}$$

so the result follows.

3.3. Proof of Theorem 6

By the SVD decomposition (see Stewart (2001)) we have $A = U\Sigma V^T$ where U, V are orthogonal with shapes $n \times n$ and $m \times m$ respectively, and Σ is an $n \times m$ diagonal (rectangular) matrix with real values $\sigma_1, \ldots, \sigma_n$ on the principal diagonal (recall that $n \leqslant m$). Using the orthogonality of U and V we obtain:

$$\begin{aligned}
\|AX\|_2^2 &= \|U\Sigma V^T X\|_2^2 \\
&= \|\Sigma V^T X\|_2^2.
\end{aligned} \tag{23}$$

Since V is orthogonal, so is V^T. Since X is uniform on the unit sphere and independent of V and Σ, we see that $V^T X$ conditioned on the pair Σ, V is also uniform on the unit sphere and thus distributed as X:

$$V^T X | \Sigma, V \sim X, \tag{24}$$

as the sphere uniform measure is invariant under orthogonal transforms. Combining the two equations above we express the squared distance $\|AX\|_2$, conditioned on Σ and V, as follows:

$$\|AX\|_2^2 \big| \Sigma, V \sim \|\Sigma X\|_2^2, \tag{25}$$

and since the right-hand side does not depend on V, this gives:

$$\|AX\|_2^2 \sim \|\Sigma X\|_2^2 = \sum_{k=1}^n \sigma_k^2 X_k^2. \tag{26}$$

It remains to observe that

$$\text{diag}(\sigma_1, \ldots, \sigma_n)^2 = \Sigma\Sigma^T, \tag{27}$$

and, because V and U are orthogonal, that:

$$AA^T = U\Sigma\Sigma^T U^T = U\Sigma\Sigma^T U^{-1}. \tag{28}$$

Thus, we see that $\lambda_k = \sigma_k^2$ are eigenvalues of AA^T.

3.4. Proof of Theorem 10

Let $Z = (Z_1, \ldots, Z_m)$ follow the Dirichlet distribution with parameters $(\alpha_1, \ldots, \alpha_m)$. Let $I \subset \{1 \ldots m\}$ and $J = \{1 \ldots m\} \setminus I$; by the self-normalizing properties (see Albert and Denis (2012)):

$$\frac{1}{\sum_{i \in I} Z_i} (Z_i)_{i \in I} \bigg| (Z_i)_{i \in J} \sim \mathsf{Dirichlet}((\alpha_i)_{i \in I}). \tag{29}$$

Since $\sum_{i=1}^m Z_i = 1$, we have that $\sum_{i \in I} Z_i = 1 - \sum_{i \in J} Z_i$ depends only on the components $(Z_i)_{i \in J}$. Therefore, from the above identity we obtain:

$$\frac{1}{\sum_{i \in I} Z_i} (Z_i)_{i \in I} \bigg| \sum_{i \in I} Z_i \sim \mathsf{Dirichlet}((\alpha_i)_{i \in I}). \tag{30}$$

Further, by the proportion properties Albert and Denis (2012):

$$\sum_{i \in I} Z_i \sim \mathsf{Beta}\left(\sum_{i \in I} \alpha_i, \sum_{i \in J} \alpha_i\right). \tag{31}$$

Let B, D be such that $D \sim \mathsf{Dirichlet}((\alpha_i)_{i \in I})$, $B \sim \mathsf{Beta}\left(\sum_{i \in I} \alpha_i, \sum_{i \in J} \alpha_i\right)$ and that random variables B, D, Z, W where $W = (W_1, \ldots, W_m)$ are independent (this is possible, since Z, W are independent). By the two equations above :

$$(Z_i)_{i \in I} \sim B \cdot D. \tag{32}$$

Now, for any *deterministic* scalar vector $w = (w_i)_{i \in I}$ it holds that:

$$\sum_{i \in I} w_i Z_i \sim B \cdot \sum_{i \in I} w_i D_i. \tag{33}$$

Since $(W_i)_{i \in I}$ is independent of B, D, Z we obtain:

$$\sum_{i \in I} W_i Z_i \sim B \cdot \sum_{i \in I} W_i D_i. \tag{34}$$

Consider now any fixed numbers $0 < p < q$. Denote $U = \frac{1}{\sum_{i \in I} W_i D_i}$, then:

$$\mathbb{P}\left[\sum_{i \in I} W_i Z_i \in [p, q]\right] = \mathbb{P}_{B,U}\left[pU \leqslant B \leqslant qU\right]. \tag{35}$$

Since B, U are independent, we have $\mathbb{P}_{B,U}\left[pU \leqslant B \leqslant qU\right] = \mathbb{E}_{u \sim U}\mathbb{P}_B\left[pu \leqslant B \leqslant qu\right]$ and thus

$$\mathbb{P}\left[\sum_{i \in I} W_i Z_i \in [p, q]\right] \leqslant \max_{u>0} \mathbb{P}[pu \leqslant B \leqslant qu], \tag{36}$$

with the equality when $\mathbb{P}[U = u^*] = 1$ where

$$u^* = \operatorname{argmax}_u \mathbb{P}_B\left[pu \leqslant B \leqslant qu\right]. \tag{37}$$

The upper bound is indeed achieved with the following choice of W:

$$\forall i \in I: \ W_i = \frac{1}{\cdot u^*}, \tag{38}$$

because then $U = \frac{1}{\sum_{i \in I} D_i/u^*} = u^*$, we use $\sum_{i \in I} D_i = 1$. Thus, we have shown

$$\max_W \mathbb{P}\left[\sum_{i \in I} W_i Z_i \in [p, q]\right] = \max_{u>0} \mathbb{P}[pu \leqslant B \leqslant qu]. \tag{39}$$

Since $\min_W \mathbb{P}\left[\sum_{i \in I} W_i Z_i \notin [p, q]\right] = 1 - \max_W \mathbb{P}\left[\sum_{i \in I} W_i Z_i \in [p, q]\right]$, the result follows.

4. Conclusion

This work constructed the *confidence-optimal* Distributional Johnson-Lindenstrauss distribution; the optimal bounds and the sampler are built based on the solution of a 1-dimensional optimization program involving the Beta distribution. With best bounds clearly established, the only way to improve further is by non-oblivious bounds.

In our approach the critical role play the techniques for handling the distortion probability on the unit sphere, the sphere parametrization using the Dirichlet distribution, and anticoncentration inequalities for the Dirichlet distribution. These techniques are of independent interest.

Acknowledgments

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, 2006.

Nir Ailon and Edo Liberty. An almost optimal unrestricted fast Johnson-Lindenstrauss transform. *ACM Transactions on Algorithms (TALG)*, 9(3):1–12, 2013.

Ayelet Akselrod-Ballin, Davi Bock, R Clay Reid, and Simon K Warfield. Accelerating image registration with the Johnson-Lindenstrauss lemma: Application to imaging 3-d neural ultrastructure with electron microscopy. *IEEE transactions on medical imaging*, 30(7):1427–1438, 2011.

Isabelle Albert and Jean-Baptiste Denis. Dirichlet and multinomial distributions: properties and uses in jags. *Unite Mathematiques et Informatique Applique*, pages 2012–5, 2012.

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.

Noga Alon. Problems and results in extremal combinatorics—i. *Discrete Mathematics*, 273(1-3): 31–53, 2003.

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

Afonso S Bandeira, Dustin G Mixon, and Benjamin Recht. Compressive classification and the rare eclipse problem. In *Compressed Sensing and its Applications*, pages 197–220. Springer, 2017.

Luca Becchetti, Marc Bury, Vincent Cohen-Addad, Fabrizio Grandoni, and Chris Schwiegelshohn. Oblivious dimension reduction for k-means: beyond subspaces and the Johnson-Lindenstrauss lemma. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1039–1050, 2019.

Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19:137, 2007.

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, 2001.

Jeremiah Blocki, Avrim Blum, Anupam Datta, and Or Sheffet. The Johnson-Lindenstrauss transform itself preserves differential privacy. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 410–419. IEEE, 2012.

Stéphane Boucheron, Gábor Lugosi, and Olivier Bousquet. Concentration inequalities. In *Summer School on Machine Learning*, pages 208–240. Springer, 2003.

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.

Christos Boutsidis, Anastasios Zouzias, and Petros Drineas. Random projections for k-means clustering. *Advances in Neural Information Processing Systems*, 23:298–306, 2010.

Christos Boutsidis, Anastasios Zouzias, Michael W Mahoney, and Petros Drineas. Randomized dimensionality reduction for k-means clustering. *IEEE Transactions on Information Theory*, 61 (2):1045–1062, 2014.

Michael Burr, Shuhong Gao, and Fiona Knoll. Optimal bounds for Johnson-Lindenstrauss transformations. *The Journal of Machine Learning Research*, 19(1):2920–2941, 2018.

Emmanuel J Candes. The restricted isometry property and its implications for compressed sensing. *Comptes rendus mathematique*, 346(9-10):589–592, 2008.

Kenneth L Clarkson and David P Woodruff. Low-rank approximation and regression in input sparsity time. *Journal of the ACM (JACM)*, 63(6):1–45, 2017.

Michael B Cohen, TS Jayram, and Jelani Nelson. Simple analyses of the sparse Johnson-Lindenstrauss transform. In *1st Symposium on Simplicity in Algorithms (SOSA 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

Graham Cormode and Piotr Indyk. Stable distributions in streaming computations. In *Data Stream Management*, pages 283–300. Springer, 2016.

Anirban Dasgupta, Ravi Kumar, and Tamás Sarlós. A sparse Johnson-Lindenstrauss transform. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 341–350, 2010.

Sanjoy Dasgupta and Anupam Gupta. An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003.

M.M. Deza and M. Laurent. *Geometry of Cuts and Metrics*. Algorithms and Combinatorics. Springer Berlin Heidelberg, 1997. ISBN 9783540616115. URL https://books.google. at/books?id=UQMXPi_luLIC.

DLMF. *NIST Digital Library of Mathematical Functions*. http://dlmf.nist.gov/, Release 1.1.0 of 2020-12-15. URL http://dlmf.nist.gov/. F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain, eds.

John Fedoruk, Byron Schmuland, Julia Johnson, and Giseon Heo. Dimensionality reduction via the Johnson-Lindenstrauss lemma: theoretical and empirical bounds on embedding dimension. *The Journal of Supercomputing*, 74(8):3933–3949, 2018.

Peter Frankl and Hiroshi Maehara. The Johnson-Lindenstrauss lemma and the sphericity of some graphs. *Journal of Combinatorial Theory, Series B*, 44(3):355–362, 1988.

Casper Benjamin Freksen and Kasper Green Larsen. On using toeplitz and circulant matrices for Johnson-Lindenstrauss transforms. *Algorithmica*, 82(2):338–354, 2020.

Alan Genz. Methods for generating random orthogonal matrices. In *Monte-Carlo and Quasi-Monte Carlo Methods 1998*, pages 199–213. Springer, 2000.

Abdulaziz Ghalib, Tyler D Jessup, Julia Johnson, and Seyedamin Monemian. Clustering and classification to evaluate data reduction via Johnson-Lindenstrauss transform. In *Future of Information and Communication Conference*, pages 190–209. Springer, 2020.

Navin Goel, George Bebis, and Ara Nefian. Face recognition experiments with random projection. In *Biometric Technology for Human Identification II*, volume 5779, pages 426–437. International Society for Optics and Photonics, 2005.

Jarvis Haupt and Robert Nowak. Signal reconstruction from noisy random projections. *IEEE Transactions on Information Theory*, 52(9):4036–4048, 2006.

Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 604–613, 1998.

Thathachar S Jayram and David P Woodruff. Optimal bounds for Johnson-Lindenstrauss transforms and streaming problems with subconstant error. *ACM Transactions on Algorithms (TALG)*, 9(3): 1–17, 2013.

William B Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary mathematics*, 26(189-206):1, 1984.

William B Johnson and Assaf Naor. The Johnson-Lindenstrauss lemma almost characterizes hilbert space, but not quite. *Discrete & Computational Geometry*, 43(3):542–553, 2010.

Daniel Kane, Raghu Meka, and Jelani Nelson. Almost optimal explicit Johnson-Lindenstrauss families. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 628–639. Springer, 2011.

Daniel M Kane and Jelani Nelson. Sparser Johnson-Lindenstrauss transforms. *Journal of the ACM (JACM)*, 61(1):1–23, 2014.

Krishnaram Kenthapadi, Aleksandra Korolova, Ilya Mironov, and Nina Mishra. Privacy via the Johnson-Lindenstrauss transform. *Journal of Privacy and Confidentiality*, 5(1):39–71, 2013.

Kasper Green Larsen and Jelani Nelson. The Johnson-Lindenstrauss lemma is optimal for linear dimensionality reduction. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

Jessica Lin and Dimitrios Gunopulos. Dimensionality reduction by random projection and latent semantic indexing. In *proceedings of the Text Mining Workshop, at the 3rd SIAM International Conference on Data Mining*, 2003.

Xudong Lv and Z Jane Wang. Fast Johnson-Lindenstrauss transform for robust and secure image hashing. In *2008 IEEE 10th workshop on multimedia signal processing*, pages 725–729. IEEE, 2008.

Odalric Maillard and Rémi Munos. Linear regression with random projections. *Journal of Machine Learning Research*, 13:2735–2772, 2012.

Konstantin Makarychev, Yury Makarychev, and Ilya Razenshteyn. Performance of Johnson-Lindenstrauss transform for k-means and k-medians clustering. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1027–1038, 2019.

George Marsaglia et al. Choosing a point from the surface of a sphere. *The Annals of Mathematical Statistics*, 43(2):645–646, 1972.

Jiří Matoušek. On variants of the Johnson-Lindenstrauss lemma. *Random Structures & Algorithms*, 33(2):142–156, 2008.

Aditya Krishna Menon. Random projections and applications to dimensionality reduction. *School of Information Technologies, The University of Sydney*, 2007.

Mervin E Muller. A note on a method for generating points uniformly on n-dimensional spheres. *Communications of the ACM*, 2(4):19–20, 1959.

Jelani Nelson, Huy L Nguyen, and David P Woodruff. On deterministic sketching and streaming for sparse recovery and norm estimation. *Linear Algebra and its Applications*, 441:152–167, 2014.

Nam H Nguyen, Thong T Do, and Trac D Tran. A fast and efficient algorithm for low-rank approximation of a matrix. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 215–224, 2009.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Serge B Provost and Young-Ho Cheong. On the distribution of linear combinations of the components of a Dirichlet random vector. *Canadian Journal of Statistics*, 28(2):417–425, 2000.

Ali Rahimi, Benjamin Recht, et al. Random features for large-scale kernel machines. In *NIPS*, volume 3, page 5. Citeseer, 2007.

Martin Roberts. How to generate uniformly random points on n-spheres and in n-balls, 2019. http://extremelearning.com.au/how-to-generate-uniformly-random-points-on-n-spheres-and-n-balls/.

G. W. Stewart. The efficient generation of random orthogonal matrices with an application to condition estimators. *SIAM Journal on Numerical Analysis*, 17(3):403–409, 1980. ISSN 00361429. URL http://www.jstor.org/stable/2156882.

G.W. Stewart. *Matrix Algorithms Volume 2: Eigensystems*. Other Titles in Applied Mathematics. Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2001. ISBN 9780898718058. URL https://books.google.at/books?id=_ui9DwAAQBAJ.

Sotiris Tasoulis, Lu Cheng, Niko Välimäki, Nicholas J Croucher, Simon R Harris, William P Hanage, Teemu Roos, and Jukka Corander. Random projection based clustering for population genomics. In *2014 IEEE international conference on big data (big data)*, pages 675–682. IEEE, 2014.

Herbert CS Thom. A note on the gamma distribution. *Monthly weather review*, 86(4):117–122, 1958.

Shashanka Ubaru, Arya Mazumdar, and Yousef Saad. Low rank approximation and decomposition of large matrices using error correcting codes. *IEEE Transactions on Information Theory*, 63(9): 5544–5558, 2017.

Suresh Venkatasubramanian and Qiushi Wang. The Johnson-Lindenstrauss transform: an empirical study. In *2011 Proceedings of the Thirteenth Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 164–173. SIAM, 2011.

Khac Ky Vu. *Random projection for high-dimensional optimization*. PhD thesis, Université Paris-Saclay (ComUE), 2016.

Kilian Weinberger, Anirban Dasgupta, John Langford, Alex Smola, and Josh Attenberg. Feature hashing for large scale multitask learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 1113–1120, 2009.

Anru R Zhang and Yuchen Zhou. On the non-asymptotic and sharp lower tail bounds of random variables. *Stat*, 9(1):e314, 2020.

Appendix A.

Listing 2: Minimal Dimension Bounds for JL Lemma

```python
import numpy as np
from scipy.optimize import bisect
from sklearn.random_projection import johnson_lindenstrauss_min_dim
from matplotlib import pyplot as plt


def optimal_djl_dim(N,m,eps):
  ''' optimal dimension with many data points '''
  fun = lambda n: optimal_djl(m,n,eps)[1]-2/(N*(N-1))
  return bisect(fun,10,m-10)

m = 1e5
eps = 0.2
sample = np.linspace(10,1000,100)
my_dim = [optimal_djl_dim(N,m,eps) for N in sample]
plt.plot(sample,my_dim,label='our_bound')
their_dim = [johnson_lindenstrauss_min_dim(N,eps) for N in sample]
plt.plot(sample,their_dim,label='previous_bound')
plt.legend()
plt.show()
```

Listing 3: Impact of Data Dimension for JL Lemma

```python
eps = 0.2
sample = np.linspace(10,1000,100)

my_dims = []

for m in [1e3,1e4,1e5]:
  my_dim = [optimal_djl_dim(N,m,eps) for N in sample]
  my_dims.append(my_dim)
  plt.plot(sample,my_dim,label='%s'%m)
plt.legend()
plt.show()
```

Listing 4: Sampling Sphere with Dirichlet Distribution

```python
from matplotlib import pyplot as plt
```

```python
from mpl_toolkits.mplot3d import axes3d
from scipy import stats

def sample_dirichlet(size=1,dim=3):
  w = stats.dirichlet(0.5*np.ones(dim)).rvs(size=size)
  sign = np.random.choice([-1,1],size=(size,dim),p=[0.5,0.5])
  return sign*w**0.5

fig = plt.figure(figsize=(8,8))
ax = fig.add_subplot(111, projection='3d')

u = np.linspace(0, 2 * np.pi, 120)
v = np.linspace(0, np.pi, 60)
x = np.outer(np.cos(u), np.sin(v))
y = np.outer(np.sin(u), np.sin(v))
z = np.outer(np.ones(np.size(u)), np.cos(v))
ax.plot_surface(x, y, z,  rstride=1, cstride=1, color='c', alpha = 0.3, linewidth = 0)

xi, yi, zi = sample_dirichlet(1000).T
ax.scatter(xi, yi, zi,color="k",s=3)
```