# Testing $k$-Wise Independence over Streaming Data

Kai-Min Chung Zhenming Liu Michael Mitzenmacher

**Abstract**

Following on the work of Indyk and McGregor [5], we consider the problem of identifying correlations in data streams. They consider a model where a stream of pairs $(i, j) \in [n]^2$ arrive, giving a joint distribution $(X, Y)$. They find approximation algorithms for how close the joint distribution is to the product of the marginal distributions under various metrics, which naturally corresponds to how close $X$ and $Y$ are to being independent.

We extend their main result to higher dimensions, where a stream of $m$ $k$-dimensional vectors in $[n]^k$ arrive, and we wish to approximate the $\ell_2$ distance between the joint distribution and the product of the marginal distributions in a single pass. Our analysis gives a randomized algorithm that is a $(1 \pm \epsilon)$ approximation (with probability $1 - \delta$) that requires space logarithmic in $n$ and $m$ and proportional to $3^k$.

# 1 Introduction

Following on the work of Indyk and McGregor [5], we consider the problem of identifying correlations in data streams. In their work, they consider a model where a stream of pairs $(i, j) \in [n]^2$ arrive, giving a joint distribution $(X, Y)$. They find approximation algorithms for how close the joint distribution is to the product of the marginal distributions under various metrics, which naturally corresponds to how close $X$ and $Y$ are to being independent. They leave the problem of higher-dimensional systems, such as when one obtains a stream of triples $(X, Y, Z) \in [n]^3$, open.

These questions have been considered in follow-up work by Braverman and Ostrovsky [3, 4]. Let us refer to the number of variables in the joint distribution as $k$. In [3], Braverman and Ostrovsky consider the $\ell_2$ metric, extending the ideas of [5] to $k > 2$, with a single-pass small-space algorithm where the space grows proportionally to $2^{O(k^2)}$. In [4], they consider the problem of the $\ell_1$ metric, where Indyk and McGregor obtain a small-space single-pass $O(\log n)$-approximation for $k = 2$. (Indyk and McGregor also obtain a linear space $(1 \pm \epsilon)$-approximation for this metric.) Here Braverman and Ostrovsky obtain a $(1 \pm \epsilon)$-approximation for all $k$, although the space required is doubly exponential in $k$.

In this paper, we provide an improvement on the results for the $\ell_2$ distance. Specifically, our analysis allows us to reduce the dependence on the space as a function of $k$ to an expression proportional to $3^k$, instead of $2^{O(k^2)}$ as in [3]. (The space used is asymptotically the same in all other parameters.) Besides yielding a technical but non-trivial improvement on the amount of space required, we believe our proof is simpler and more natural than the extension presented in [3], and hence might prove useful for further developments on this class of problems. Indeed, in [3] Braverman and Ostrovsky explicitly express the potential difficulties in generalizing the approach of Indyk and McGregor, and develop a different approach for their upper bound. In contrast, our argument develops more naturally from the original argument of Indyk and McGregor [5]. In particular, we demonstrate the existence of a useful geometric partitioning that extends their main idea to higher dimensions. More discussion on the limitation of Indyk and McGregor's approach, and the comparison of our approach to the technique of Braverman and Ostrovsky can be found in Section 3.1.

For motivation, we note that testing for independence has been an important subject in both the study of statistics and the design of database system. Further discussions appear in several previous works, including [3, 4, 5, 6, 8]. Traditional non-parametric methods of testing independence over empirical data usually require space complexity that is either linear in the support size or input size. The scale of contemporary data sets often prohibits such space complexity. It is therefore natural to ask whether we will be able to design algorithms to test for independence in streaming model. Interestingly, this specific problem appears not to have been introduced until the work of Indyk and McGregor. While arguably results for the $\ell_1$ norm would be stronger than for the $\ell_2$ norm in this setting, the problem for $\ell_2$ norms is interesting in its own right; see the discussion of [3] for further elaboration on this point. Further, given the current complexity of the best results for the $\ell_1$ metric given in [4], it may be that our result for the $\ell_2$ may be more appropriate for practical implementation.

Our specific theoretical contribution can be summarized as follows:

**Theorem 1.1.** *For every $\epsilon > 0$ and $\delta > 0$, there exists a randomized algorithm that computes, given a sequence $a_1, ..., a_m$ of $k$-tuples, in one pass and using $O(3^k \epsilon^{-2} \log \frac{1}{\delta}(\log m + \log n))$ memory bits, a number $Y$ so that the probability $Y$ deviates from the $\ell_2$ distance between product and joint distribution by more than a factor of $(1 + \epsilon)$ is at most $\delta$.*

# 2 Review of the Algorithm for $k = 2$

We begin by reviewing the approximation algorithm and associated proof for the $\ell_2$ norm given in [5]. Reviewing this result will allow us to provide the necessary notation and frame the setting for our extension to general $k$. Moreover, in our proof, we find that a constant in Lemma 3.1 from [5] that we subsequently generalize appears incorrect. (Because of this, our proof is slightly different and more detailed than the original.) Although the error is minor in the context of their paper (it only affects

the constant factor in the order notation), it becomes more important when considering the proper generalization to larger $k$, and hence it is useful to correct here.

### 2.1 The Model

We provide the general underlying model. Here we mostly follow the notation of [3, 5].

Let $S$ be a stream of size $m$ with elements $a_1, ..., a_m$, where $a_i \equiv (a_i^1, ..., a_i^k) \in [n]^k$. (When we have a sequence of elements that are themselves vectors, we denote the sequence number by a subscript and the vector entry by a superscript when both are needed.) The stream $S$ defines an empirical distribution over $[n]^k$ as follows: the frequency $f(\omega)$ of an element $\omega \in [n]^k$ is defined as the number of times it appears in $S$, and the empirical distribution is

$$\Pr[\omega] = \frac{f(\omega)}{m} \quad \text{for any } \omega \in [n]^k.$$

Since $\omega = (\omega_1, ..., \omega_k)$ is a vector of size $k$, we may also view the streaming data as defining a joint distribution over the random variables $X_1, ..., X_k$ corresponding to the values in each dimension. (In the case of $k = 2$, we write the random variables as $X$ and $Y$ rather than $X_1$ and $X_2$.) There is a natural way of defining marginal distribution for the random variable $X_i$: for $\omega_i \in [n]$, let $f_i(\omega_i)$ be the number of times $\omega_i$ appears in the $i$th coordinate of an element of $S$, or

$$f_i(\omega_i) = \left|\{a_j \in S : a_j^i = \omega_i\}\right|.$$

The empirical marginal distribution $\Pr_i[\cdot]$ for the $i$th coordinate is defined as

$$\Pr_i[\omega_i] = \frac{f_i(\omega_i)}{m} \quad \text{for any } \omega_i \in [n].$$

For $\omega \in [n]^k$, let $v_\omega = \Pr[\omega] - \prod_{1 \le i \le k} \Pr_i[\omega_i]$, and let $v$ be the vector in $\mathrm{R}^{[n]^k}$ of values $v_\omega$ in some order. Our goal is to approximate the value

$$\|v\| \equiv \left( \sum_{\omega \in [n]^k} \left| \Pr[\omega] - \prod_{1 \le i \le k} \Pr_i[\omega_i] \right|^2 \right)^{\frac{1}{2}}. \tag{1}$$

This represent the $\ell_2$ norm between the product of the marginal distributions and the joint distribution, which we would expect to be close to zero in the case where the $X_i$ were truly independent.

Finally, our algorithms will assume the availability of 4-wise independent binary vectors. For more on 4-wise independence, including efficient implementations, see [1, 9]. For the purposes of this paper, the following simple definition will suffice.

**Definition 2.1.** (4-wise independence) *A random variable $X$ over $\{-1, 1\}^n$ is* 4-wise independent *if for any distinct values $i_1, i_2, i_3, i_4 \in [n]$ and any $b_1, b_2, b_3, b_4 \in \{-1, 1\}$, the following equality holds,*

$$\Pr_{x \leftarrow X}[x_{i_1} = b_1, x_{i_2} = b_2, x_{i_3} = b_3, x_{i_4} = b_4] = 1/16.$$

### 2.2 The Algorithm and its Analysis for $k = 2$

In this case, we assume that the sequence $(a_1^1, a_1^2), (a_2^1, a_2^2), ..., (a_m^1, a_m^2)$ arrives an item by an item. Each $(a_i^1, a_i^2)$ (for $1 \le i \le m$) is an element in $[n]^2$. The random variables $X$ and $Y$ over $[n]$ can be expressed as follows:

$$\begin{cases} \Pr[i,j] = \Pr[X = i, Y = j] = |\{\ell : (a_\ell^1, a_\ell^2) = (i,j)\}|/m \\ \Pr_1[i] = \Pr[X = i] = |\{\ell : (a_\ell^1, a_\ell^2) = (i,\cdot)\}|/m \\ \Pr_2[j] = \Pr[Y = j] = |\{\ell : (a_\ell^1, a_\ell^2) = (\cdot,j)\}|/m. \end{cases}$$

We simplify the notation and use $p_i \equiv \Pr[X = i]$, $q_j \equiv \Pr[Y = j]$, $r_{i,j} = \Pr[X = i, Y = j]$. and $s_{i,j} = \Pr[X = i] \Pr[Y = j]$.

Indyk and McGregor's algorithm proceeds in a similar fashion to the streaming algorithm presented in [2]. Specifically let $s_1 = 72\epsilon^{-2}$ and $s_2 = 2\log(1/\delta)$. The algorithm computes $s_2$ random variables $D_1, D_2, ..., D_{s_2}$ and outputs their median. The output is the algorithm's estimate on the norm of $v$ defined in Equation 1. Each $D_i$ is the average of $s_1$ random variables $D_{ij}$: $1 \le j \le s_1$, where $D_{ij}$ are independent, identically distributed random variables. Each of the variables $D = D_{ij}$ can be computed from the algorithmic routine shown in Figure 1.

2-D Approximation $\left((a_1^1, a_1^2), ..., (a_m^1, a_m^2)\right)$
1 Independently generate 4-wise independent random vectors $x, y$ from $\{-1, 1\}^n$.
2 $t_1 \leftarrow 0, t_2 \leftarrow 0, t_3 \leftarrow 0$.
3 **for** $c \leftarrow 1$ **to** $m$
4 **do** Let the $c$th item $(a_c^1, a_c^2) = (i, j)$
5 $t_1 \leftarrow t_1 + x_i y_j$, $t_2 \leftarrow t_2 + x_i$, $t_3 \leftarrow t_3 + y_j$.
6 Return $D = (t_1/m - t_2 t_3/m^2)^2$.

Figure 1: The procedure for generating random variable $D$ for $k = 2$.

Notice that by the end of the process 2-D Approximation, we have $t_1/m = \sum_{i,j\in[n]} x_i y_j r_{i,j}$, $t_2/m = \sum_{i\in[n]} x_i p_i$, and $t_3/m = \sum_{i\in[n]} y_i q_i$. Also, when a vector is in $\mathrm{R}^{(n^2)}$, its indices can be represented by $(i_1, i_2) \in [n]^2$. In what follows, we will use a bold letter to represent the index of a high dimensional vector, e.g., $v_{\mathbf{i}} \equiv v_{i_1,i_2}$. The following Lemma shows that the expectation of $D$ is $\|v\|$ and the variance of $D$ is at most $9(\mathrm{E}[D])^2$.

**Lemma 2.2.** *[5] Consider two independent vectors $x = (x_1, ..., x_n), y = (y_1, ..., y_n) \in \{-1, 1\}^n$, where each vector is 4-wise independent. Let $v \in \mathrm{R}^{n^2}$ and $z_{\mathbf{i}} (\equiv z_{(i_1,i_2)}) = x_{i_1} y_{i_2}$. Let us define $D = \left(\sum_{\mathbf{i}\in[n]^2} z_{\mathbf{i}} v_{\mathbf{i}}\right)^2$. Then $\mathrm{E}[D] = \sum_{\mathbf{i}\in[n]^2} v_{\mathbf{i}}^2$ and $\mathrm{Var}[D] \le 9(\mathrm{E}[D])^2$.*

*Proof.* We have $\mathrm{E}[D] = \mathrm{E}[(\sum_{\mathbf{i}} z_{\mathbf{i}} v_{\mathbf{i}})^2] = \sum_{\mathbf{i}} v_{\mathbf{i}}^2 \mathrm{E}[z_{\mathbf{i}}^2] + \sum_{\mathbf{i}\neq\mathbf{j}} v_{\mathbf{i}} v_{\mathbf{j}} \mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}}]$. For all $\mathbf{i} \in [n]^2$, we know $z_{\mathbf{i}}^2 = 1$. On the other hand, $z_{\mathbf{i}} z_{\mathbf{j}} \in \{-1, 1\}$. The probability that $z_{\mathbf{i}} z_{\mathbf{j}} = 1$ is $\Pr[z_{\mathbf{i}} z_{\mathbf{j}} = 1] = \Pr[x_{i_1} x_{j_1} y_{i_2} y_{j_2} = 1] = 1/16 + \binom{4}{2} 1/16 + 1/16 = 1/2$. The last equality holds is because $x_{i_1} x_{j_1} y_{i_2} y_{j_2} = 1$ is equivalent to saying either all these variables are 1, or exactly 2 of these variables are -1, or all these variables are -1. Therefore, $\mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}}] = 0$. Consequently, $\mathrm{E}[D] = \sum_{\mathbf{i}\in[n]^2} v_{\mathbf{i}}^2$.

Now we bound the variance. We have

$$\mathrm{Var}[D] \le \mathrm{E}[D^2] = \sum_{\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l}\in[n]^2} \mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}} z_{\mathbf{k}} z_{\mathbf{l}}] v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}} \le \sum_{\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l}\in[n]^2} |\mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}} z_{\mathbf{k}} z_{\mathbf{l}}]| \cdot |v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}|$$

Also $|\mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}} z_{\mathbf{k}} z_{\mathbf{l}}]| \in \{0, 1\}$. The quantity $E[z_{\mathbf{i}} z_{\mathbf{j}} z_{\mathbf{k}} z_{\mathbf{l}}] \neq 0$ if and only if the following relation holds,

$$\forall s \in [2] : ((i_s = j_s) \wedge (k_s = l_s)) \vee ((i_s = k_s) \wedge (j_s = l_s)) \vee ((i_s = l_s) \wedge (k_s = j_s)) . \tag{2}$$

Denote the set of 4-tuples $(\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l})$ that satisfy the above relation by $\mathcal{D}$. We may also view each 4-tuple as an ordered set that consists of 4 points in $[n]^2$. Consider the unique smallest axes-parallel rectangle in $[n]^2$ that contains a given 4-tuple in $\mathcal{D}$ (i.e. contains the four ordered points). Note this could either be a (degenerate) line segment or a (non-degenerate) rectangle, as we discuss below. Let $M : \mathcal{D} \to \{A, B, C, D\}$ be the function that maps an element $\sigma \in \mathcal{D}$ to the smallest rectangle $ABCD$ defined by $\sigma$. Since a rectangle can be uniquely determined by its diagonals, we may write $M : \mathcal{D} \to (\chi_1, \chi_2, \varphi_1, \varphi_2)$, where

$\chi_1 \le \chi_2 \in [n]$, $\varphi_1 \le \varphi_2 \in [n]$ and the corresponding rectangle is understood to be the one with diagonal $\{(\chi_1, \varphi_1), (\chi_2, \varphi_2)\}$. Also, the inverse function $M^{-1}(\chi_1, \chi_2, \varphi_1, \varphi_2)$ represents the pre-images of $(\chi_1, \chi_2, \varphi_1, \varphi_2)$ in $\mathcal{D}$. $(\chi_1, \chi_2, \varphi_1, \varphi_2)$ is degenerate if either $\chi_1 = \chi_2$ or $\varphi_1 = \varphi_2$, in which case the rectangle (and its diagonals) correspond to the segment itself, or $\chi_1 = \chi_2$ and $\varphi_1 = \varphi_2$, and the rectangle is just a single point.

**Example 2.3.** *Let* $\mathbf{i} = (1,2)$, $\mathbf{j} = (3,2)$, $\mathbf{k} = (1,5)$, *and* $\mathbf{l} = (3,5)$. *The tuple is in* $\mathcal{D}$ *and its corresponding bounding rectangle is a non-degenerate rectangle. The function* $M(\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l}) = (1, 3, 2, 5)$.

**Example 2.4.** *Let* $\mathbf{i} = \mathbf{j} = (1,4)$ *and* $\mathbf{k} = \mathbf{l} = (3,7)$. *The tuple is also in* $\mathcal{D}$ *and minimal bounding rectangle formed by these points is an interval* $\{(1,4),(3,7)\}$. *The function* $M(\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l}) = (1, 3, 4, 7)$.

To start we consider the non-degenerate cases. Fix any $(\chi_1, \chi_2, \varphi_1, \varphi_2)$ with $\chi_1 < \chi_2$ and $\phi_1 < \phi_2$. There are in total $\binom{4}{2}^2 = 36$ tuples $(\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l})$ in $\mathcal{D}$ with $M(\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l}) = (\chi_1, \chi_2, \varphi_1, \varphi_2)$. Twenty-four of these tuples correspond to the setting where none of $\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l}$ are equal, as there are twenty-four permutations of the assignment of the labels $\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l}$ to the four points. (This corresponds to the first example.) In this case the four points form a rectangle, and we have $|v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}| \le \frac{1}{2}((v_{\chi_1,\varphi_1} v_{\chi_2,\varphi_2})^2 + (v_{\chi_1,\varphi_2} v_{\chi_2,\varphi_1})^2)$. Intuitively, in these cases, we assign the "weight" of the tuple to the diagonals.

The remaining twelve tuples in $M^{-1}(\chi_1, \chi_2, \varphi_1, \varphi_2)$ correspond to intervals. (This corresponds to the second example.) In this case two of $\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l}$ correspond to one endpoint of the interval, and the other two labels correspond to the other endpoint. Hence we have either $|v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}| = (v_{\chi_1,\varphi_1} v_{\chi_2,\varphi_2})^2$ or $|v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}| = (v_{\chi_1,\varphi_2} v_{\chi_2,\varphi_1})^2$, and there are six tuples for each case.

Therefore for any $\chi_1 < \chi_2 \in [n]$ and $\varphi_1 < \varphi_2 \in [n]$ we have:

$$\sum_{\substack{(\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l}) \in \\ M^{-1}(\chi_1,\chi_2,\varphi_1,\varphi_2)}} |v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}| \le 18((v_{\chi_1,\varphi_1} v_{\chi_2,\varphi_2})^2 + (v_{\chi_1,\varphi_2}, v_{\chi_2,\varphi_1})^2).$$

The analysis is similar for the degenerate cases, where the constant 18 in the bound above is now quite loose. When exactly one of $\chi_1 = \chi_2$ or $\varphi_1 = \varphi_2$ holds, the size of $M^{-1}(\chi_1, \chi_2, \varphi_1, \varphi_2)$ is $\binom{4}{2} = 6$, and the resulting intervals correspond to vertical or horizontal lines. When both $\chi_1 = \chi_2$ and $\varphi_1 = \varphi_2$, then $|M^{-1}(\chi_1, \chi_2, \varphi_1, \varphi_2)| = 1$. In sum, we have

$$\begin{aligned}
\sum_{\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l} \in \mathcal{D}} |v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}| &= \sum_{\substack{\chi_1 \le \chi_2 \\ \varphi_1 \le \varphi_2}} \sum_{\substack{(\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l}) \in \\ M^{-1}(\chi_1,\chi_2,\varphi_1,\varphi_2)}} |v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}| \\
&\le \sum_{\substack{\chi_1 < \chi_2 \\ \varphi_1 < \varphi_2}} 18((v_{\chi_1,\varphi_1} v_{\chi_2,\varphi_2})^2 + (v_{\chi_1,\varphi_2} v_{\chi_2,\varphi_1})^2) + \sum_{\substack{\chi_1 = \chi_2 \\ \varphi_1 < \varphi_2}} 6((v_{\chi_1,\varphi_1} v_{\chi_2,\varphi_2})^2 + (v_{\chi_1,\varphi_2} v_{\chi_2,\varphi_1})^2) \\
&\quad + \sum_{\substack{\chi_1 < \chi_2 \\ \varphi_1 = \varphi_2}} 6((v_{\chi_1,\varphi_1} v_{\chi_2,\varphi_2})^2 + (v_{\chi_1,\varphi_2} v_{\chi_2,\varphi_1})^2) + \sum_{\substack{\chi_1 = \chi_2 \\ \varphi_1 = \varphi_2}} (v_{\chi_1,\varphi_1} v_{\chi_2,\varphi_2})^2 \\
&\le 9 \sum_{\substack{\mathbf{i} \in [n]^2 \\ \mathbf{j} \in [n]^2}} (v_{\mathbf{i}} v_{\mathbf{j}})^2 = 9E^2[D].
\end{aligned}$$

Finally, we have $\sum_{\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l} \in [n]^2} |\mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}} z_{\mathbf{k}} z_{\mathbf{l}}]| \cdot |v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}| \le \sum_{\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l} \in \mathcal{D}} |v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}| \le 9\mathrm{E}^2[D]$ and $\mathrm{Var}[D] \le 9\mathrm{E}^2[D]$. □

We emphasize the geometric interpretation of the above proof as follows. The goal is to bound the variance by a constant times $\mathrm{E}^2[D] = \sum_{\mathbf{i},\mathbf{j} \in [n]^2} (v_{\mathbf{i}} v_{\mathbf{j}})^2$, where the index set is the set of all possible lines in plane $[n]^2$ (each line appears twice). We first show that $\mathrm{Var}[D] \le \sum_{\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l} \in \mathcal{D}} |v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}|$, where the 4-tuple index set corresponds to a set of rectangles in a natural way. The main idea of Indyk and McGregor is to use inequalities of the form $|v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}| \le \frac{1}{2}((v_{\chi_1,\varphi_1} v_{\chi_2,\varphi_2})^2 + (v_{\chi_1,\varphi_2} v_{\chi_2,\varphi_1})^2)$ to assign the "weight" of each

4-tuple to the diagonals of the corresponding rectangle. The above analysis shows that 18 copies of all lines are sufficient to accommodate all 4-tuples. While similar inequalities could also assign the weight of a 4-tuple to the vertical or horizontal edges of the corresponding rectangle, using vertical or horizontal edges is problematic. The reason is that there are $\Omega(n^4)$ 4-tuples but only $O(n^3)$ vertical or horizontal edges, so some lines would receive $\Omega(n)$ weight, requiring $\Omega(n)$ copies. This problem is also noted in [3].

Our bound here is $\mathrm{Var}[D] \leq 9\mathrm{E}^2[D]$, while in [5] the bound is given as $\mathrm{Var}[D] \leq 3\mathrm{E}^2[D]$. There appears to have been an error in the derivation in [5]; some intuition comes from the following example. We note that $|\mathcal{D}|$ is at least $\binom{4}{2}^2 \cdot \binom{n}{2}^2 = 9n^4 - 9n^2$ (This counts the number of non-degenerate 4-tuples.) Now if we set $v_i = 1$ for all $1 \leq i \leq n^2$, we have $\mathrm{E}[D^2] \geq |\mathcal{D}| = 9n^4 - 9n^2 \sim 9\mathrm{E}^2(D)$, which suggests $\mathrm{Var}[D] > 3\mathrm{E}^2[D]$. Again, we emphasize this discrepancy is of little importance to [5]; the point there is that the variance is bounded by a constant factor times the square of the expectation. It is here, where we are generalizing to higher dimensions, that the exact constant factor is of some importance.

Given the bounds on the expectation and variance for the $D_{i,j}$, standard techniques yield a bound on the performance of our algorithm.

**Theorem 2.5.** *For every $\epsilon > 0$ and $\delta > 0$, there exists a randomized algorithm that computes, given a sequence $(a_1^1, a_1^2), ..., (a_m^1, a_m^2)$, in one pass and using $O(\epsilon^{-2} \log \frac{1}{\delta}(\log m + \log n))$ memory bits, a number* Med *so that the probability* Med *deviates from $\|v\|$ by more than $\epsilon$ is at most $\delta$.*

*Proof.* Recall the algorithm described in the beginning of Section 2.2: let $s_1 = 72\epsilon^{-2}$ and $s_2 = 2\log\delta$. We first computes $s_2$ random variables $D_1, D_2, ..., D_{s_2}$ and outputs their median Med, where each $D_i$ is the average of $s_1$ random variables $D_{ij}$: $1 \leq j \leq s_1$ and $D_{ij}$ are independent, identically distributed random variables computed by Figure 1. By Chebyshev's inequality, we know that for any fixed $i$,

$$\Pr\left(\left|D_i - \|v\|\right|\right) \geq \epsilon\|v\|] \leq \frac{\mathrm{Var}(D_i)}{\epsilon^2\|v\|^2} = \frac{(1/s_1)\mathrm{Var}[D]}{\epsilon^2\|v\|^2} = \frac{(9\epsilon^2/72)\|v\|^2}{\epsilon^2\|v\|^2} = \frac{1}{8}.$$

Finally, by standard Chernoff bound arguments (see for example Chapter 4 of [7]), the probability that more than $s_2/2$ of the variables $Y_i$ deviate by more than $\epsilon\|v\|$ from $\|v\|$ is at most $\delta$. In case this does not happen, the median Med supplies a good estimate to the required quantity $\|v\|$ as needed. □

# 3 The Algorithm and its Analysis for $k > 2$

## 3.1 Gaining Intuition: The Case $k = 3$

For the case of general $k$, the input to the algorithm is $a_1, a_2, ..., a_m$, where each $a_i = (a_i^1, a_i^2, ..., a_i^k)$ is an element in $[n]^k$. Let $s_1 = 8 \cdot 3^k \cdot \epsilon^{-2}$ and $s_2 = 2\log\delta$. As before our algorithm computes $s_2$ random variables $D_1, ..., D_{s_2}$ and outputs their median. Each $D_i$ is the average of $s_1$ random variables $D_{ij}$: $1 \leq j \leq s_1$, where $D_{ij}$ are independent, identically distributed random variables. Each of the variables $D = D_{ij}$ can be computed from the algorithmic routine shown in Figure 2.

k-D Approximation $(a_1, ..., a_m)$

1 Independently generate 4-wise independent random vectors $x_1, ..., x_k$ from $\{-1, 1\}^n$.
2 $s \leftarrow 0$, and $t_i \leftarrow 0$ for $1 \leq i \leq k$.
3 **for** $c \leftarrow 1$ **to** $m$
4 **do** $s \leftarrow s + \prod_{1 \leq j \leq k} x_j^{a_c^j}$, $t_j \leftarrow t_j + x_j^{a_c^j}$ for $1 \leq j \leq k$.
5 Return $D = (s/m - \prod_{1 \leq j \leq k}(t_j/m))^2$.

Figure 2: The procedure for generating random variable $X$ for general $k$.

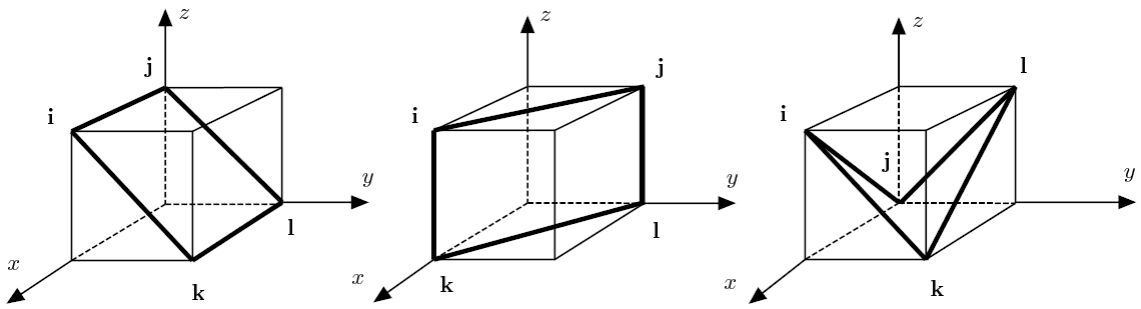


Figure 3: Examples of possible 4-tuples in $\mathcal{D}_3$. The leftmost instance: $\mathbf{i} = (1, 0, 1)$, $\mathbf{j} = (0, 0, 1)$, $\mathbf{k} = (1, 1, 0)$, and $\mathbf{l} = (0, 1, 0)$; the middle instance: $\mathbf{i} = (1, 0, 1)$, $\mathbf{j} = (0, 1, 1)$, $\mathbf{k} = (1, 0, 0)$, and $\mathbf{l} = (0, 1, 0)$; the rightmost instance: $\mathbf{i} = (1, 0, 1)$, $\mathbf{j} = (0, 0, 0)$, $\mathbf{k} = (1, 1, 0)$, and $\mathbf{l} = (0, 1, 1)$. In the first two instances, the four points $\mathbf{i}$, $\mathbf{j}$, $\mathbf{k}$, and $\mathbf{l}$ spans 2-dimensional subspace. The diagonals in the first two cases are intuitive to define. In the last instance, the four points $\mathbf{i}$, $\mathbf{j}$, $\mathbf{k}$, and $\mathbf{l}$ do not share the same plane. It is not clear how diagonals shall be defined in this case.

Like the case for $k = 2$, we shall show the algorithm $k$-D Approximation outputs a correct estimator and the variance of the estimator is well-controlled. We will soon see that finding $\mathrm{E}[D]$ is less challenging; the more difficult question is to bound $\mathrm{Var}[D]$, which we will focus on below. By the same argument, we have

$$\mathrm{Var}[D] \leq \mathrm{E}[D^2] = \sum_{\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l}} \mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}} z_{\mathbf{k}} z_{\mathbf{l}}] v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}} \leq \sum_{\mathbf{i},\mathbf{j},\mathbf{k},\mathbf{l}} |\mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}} z_{\mathbf{k}} z_{\mathbf{l}}]| \cdot |v_{\mathbf{i}} v_{\mathbf{j}} v_{\mathbf{k}} v_{\mathbf{l}}|$$

and the quantity $\mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}} z_{\mathbf{k}} z_{\mathbf{l}}] \neq 0$ if and only if the following relation holds:

$$\forall s \in [k] : ((i_s = j_s) \wedge (k_s = l_s)) \vee ((i_s = k_s) \wedge (j_s = l_s)) \vee ((i_s = l_s) \wedge (k_s = j_s)) \,. \tag{3}$$

Let $\mathcal{D}_k$ be the set of all 4-tuples $(\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l})$ that satisfy the above relation. There appears to be no natural way to translate a 4-tuple in $\mathcal{D}_k$ to two diagonal pairs by using only the techniques for the case for $k = 2$ regardless of how careful we are to define diagonals for a 4-tuple in $\mathcal{D}_k$. To clarify this, let us take a closer look at the case for $k = 3$. The quantity $\mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}} z_{\mathbf{k}} z_{\mathbf{l}}]$ is non-zero if and only if the 4-tuple $(\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l})$ are in $\mathcal{D}_3$ . Mimicking what Indyk and McGregor did, we interpret 4-tuples in $\mathcal{D}_3$ geometrically. Figure 3 illustrates possible configurations (up to rotations and permutations) of the tuples in $\mathcal{D}_3$.

An immediate problem in applying Indyk and McGregor's technique is that the "diagonals" now are no longer well-defined. A possible fix is to redefine the diagonals for the 4-tuples in $\mathcal{D}_3$. While there is a natural way of defining diagonals for the first two configurations in Figure 3 in the Appendix, defining diagonals for the rightmost configuration is more subtle. Let us call a the rightmost configuration a *bad* 4-tuple. Formally speaking, a 4-tuple $(A, B, C, D) \in \mathcal{D}_3$ is a bad tuple if the four points $A$, $B$, $C$, and $D$ do not lie on the same plane. We say this type of 4-tuples are bad because there is no natural way to define the "diagonals" over these tuples.

We shall argue that we would not be able to obtain any useful bound if we apply Indyk and McGregor's technique in a straightforward way. Indeed, there are only three possible ways of applying the arithmetic-geometric mean inequality (henceforth, referred to as the A-G inequality) over the bad tuples: $2|v_A v_B v_C v_D| \leq v_A^2 v_B^2 + v_C^2 v_D^2$, $2|v_A v_B v_C v_D| \leq v_A^2 v_C^2 + v_B^2 v_D^2$, or $2|v_A v_B v_C v_D| \leq v_A^2 v_D^2 + v_B^2 v_C^2$. Accordingly, there could be three ways of defining diagonals for the rightmost 4-tuple, which are all visualized in Figure 4.

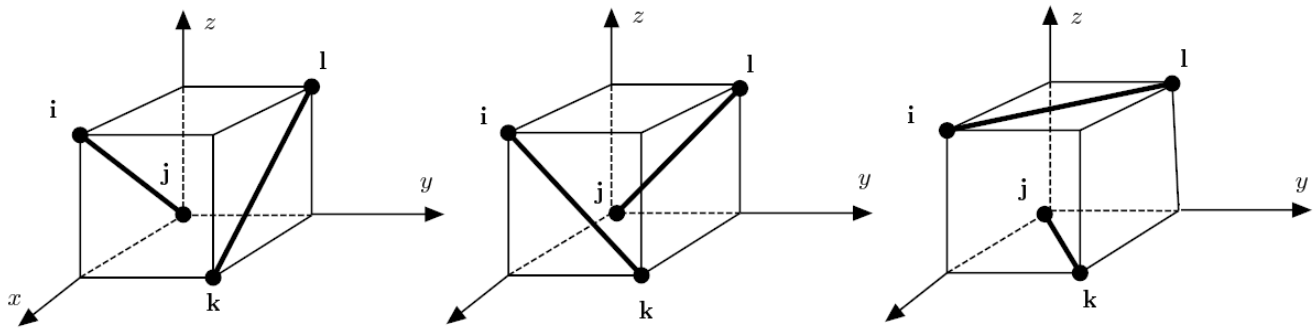


Figure 4: Possible ways of applying arithmetic-geometric inequality over a bad 4-tuple. The geometric interpretation of arithmetic-geometric inequality is to translate a product of four points into the sum of two lines. Leftmost: $v_{\mathbf{i}}v_{\mathbf{j}}v_{\mathbf{k}}v_{\mathbf{l}} \leq \frac{1}{2}(v_{\mathbf{i}}^2v_{\mathbf{j}}^2 + v_{\mathbf{k}}^2v_{\mathbf{l}}^2)$; middle: $v_{\mathbf{i}}v_{\mathbf{j}}v_{\mathbf{k}}v_{\mathbf{l}} \leq \frac{1}{2}(v_{\mathbf{i}}^2v_{\mathbf{k}}^2 + v_{\mathbf{j}}^2v_{\mathbf{l}}^2)$; rightmost: $v_{\mathbf{i}}v_{\mathbf{j}}v_{\mathbf{k}}v_{\mathbf{l}} \leq \frac{1}{2}(v_{\mathbf{i}}^2v_{\mathbf{l}}^2 + v_{\mathbf{j}}^2v_{\mathbf{k}}^2)$.

These three ways of defining diagonals, however, all suffer from a serious drawback. The diagonals are *always* parallel to one of the $xy$, $yz$, or $xz$ planes.[1] The total number of intervals that are parallel to $xy$, $yz$, or $xz$ plane is $\Theta(n^5)$ while the total number of bad 4-tuples is $\Theta(n^6)$. Therefore, we can at best to get a bound that looks like $\mathrm{Var}[D] \leq \Theta(n)\mathrm{E}^2[D]$. This bound will be insufficient to establish polylogarithmic space algorithms. Again, Braverman and Ostrovsky also made similar observations on the limits of Indyk and McGregor's technique in [3].

Notice that Indyk and McGregor's way of using the A-G inequality in fact always translates a product of four points to a sum of two lines *locally*. The obstacle presented above suggests that only using local transformation will not suffice to give us a useful bound. We therefore provide a global transformation that can fix the problem. Specifically, we show how to handle the bad tuples by applying the A-G inequality twice. The geometric interpretation of our technique corresponds to global manipulations of the bad tuples.

Let us first develop a systematic way to generate all bad tuples in $\mathcal{D}_3$. Consider an arbitrary hyperrectangle $\mathcal{R}$ that is defined by its diagonal $\{(\alpha_1, \alpha_2, 1), (\beta_1, \beta_2, n)\}$. Consider two sets of point pairs which are along the edges of the hyperrectangle $\mathcal{R}$: $\mathbf{B} \equiv \{(B_{1i}, B_{2i}) \mid 1 \leq i \leq n, B_{1i} = (\alpha_1, \alpha_2, i), B_{2i} = (\beta_1, \beta_2, i)\}$ and $\mathbf{G} \equiv \{(G_{1i}, G_{2i}) \mid 1 \leq i \leq n, G_{1i} = (\alpha_1, \beta_2, i), G_{2i} = (\alpha_2, \beta_1, i)\}$. These two sets can be visualized in Figure 5. Notice that a pair in $\mathbf{B}$ and a pair in $\mathbf{G}$ with different $z$-coordinates form a bad 4-tuple. Directly applying the A-G inequality for an isolated quantity $v_{B_{1i}}v_{B_{2i}}v_{G_{1j}}v_{G_{2j}}$, where $(B_{1i}, B_{2i}, G_{1j}, G_{2j})$ is a bad 4-tuple, would not allow us to derive any useful bound. Instead, we may group *all* bad tuples (together with some degenerate tuples with the same $z$-coordinate) bounded by $\mathcal{R}$ and apply the A-G inequality two consecutive times as follows:

$$\sum_{1\leq i,j\leq n} |v_{G_{1i}}v_{G_{2i}}v_{B_{1j}}v_{B_{2j}}| = \left(\sum_{1\leq i\leq n} |v_{G_{1i}}v_{G_{2i}}|\right)\left(\sum_{1\leq j\leq n} |v_{B_{1j}}v_{B_{2j}}|\right) \leq \frac{1}{2}\left[\Big(\sum_i |v_{G_{1i}}v_{G_{2i}}|\Big)^2 + \Big(\sum_j |v_{B_{1j}}v_{B_{2j}}|\Big)^2\right],$$

where the last inequality holds because of the A-G inequality. Next, we see that

$$\Big(\sum_{1\leq i\leq n} |v_{G_{1i}G_{2i}}|\Big)^2 = \sum_{1\leq i,j\leq n} |v_{G_{1i}}v_{G_{2i}}v_{G_{1j}}v_{G_{2j}}| \leq \frac{1}{2}\sum_{i,j}\left(v_{G_{1i}}^2v_{G_{2j}}^2 + v_{G_{2i}}^2v_{G_{1j}}^2\right) = \sum_{i,j} v_{G_{1i}}^2v_{G_{2j}}^2.$$

[1]Here $x$, $y$, and $z$ shall be understood as 3 axes for the space $[n]^3$.

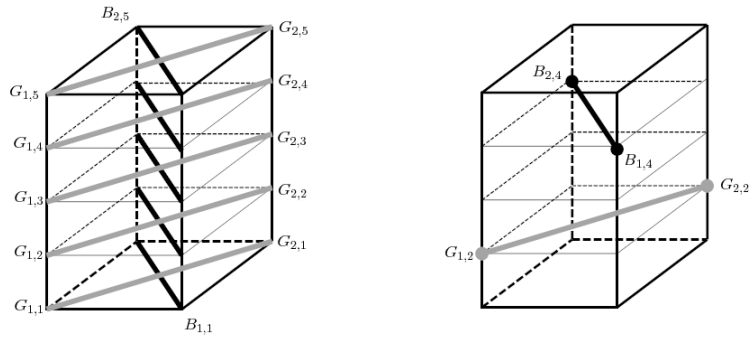


Figure 5: Defining the set of point pairs $\mathbf{B}$ and $\mathbf{G}$ in a hyperrectangle for $k = 3$. The rectangular parallelepiped to the left is an $\mathcal{R}$ for a fixed $\alpha_1, \alpha_2, \beta_1$, and $\beta_2$. The black lines drawn on the rectangular parallelepiped indicate the set $\mathbf{B}$ and the gray lines indicate the set $\mathbf{G}$. In particular a black (gray) line corresponds to a pair of points in $\mathbf{B}$ (in $\mathbf{G}$). An arbitrary black line and an arbitrary gray line corresponds with a bad 4-tuple in $\mathcal{D}_3$. For example, the rectangular parallelepiped to the right highlights a black line $\{B_{2,4}, B_{1,4}\}$ and a gray line $\{G_{1,2}, G_{2,2}\}$. The 4-tuples that consist of $\{B_{2,4}, B_{1,4}, G_{1,2}, G_{2,2}\}$ (e.g., $(B_{2,4}, B_{1,4}, G_{1,2}, G_{2,2})$) are all bad 4-tuples in $\mathcal{D}_3$.

The inequality in the middle holds because of the A-G inequality again. Similarly, $(\sum_j |v_{B_{1j}} v_{B_{2j}}|)^2 \leq \sum_{i,j} v_{B_{1i}}^2 v_{B_{2j}}^2$. Notice now that the set of all bad tuples generated by the sets $\mathbf{G}$ and $\mathbf{B}$ are charged to the diagonals in the form $v_{G_{1i}}^2 v_{G_{2j}}^2$ (or $v_{B_{1i}}^2 v_{B_{2j}}^2$). As opposed to having an insufficient number of diagonals being parallel to one of the $xy$, $yz$, or $xz$ planes, there are plenty of diagonals of the form $v_{B_{1i}}^2 v_{B_{2j}}^2$.

The above manipulation gives us a way to translate bad tuples to diagonals that are not parallel to the $xy, yz$, or $xz$ planes, which is sufficient to analyze the $k = 3$ case in a way similar to the $k = 2$ case. The manipulation allows us to fix bad tuples that reside on one stripe. Processing all 4-tuples in $\mathcal{D}_3$ requires a little additional work. We group bad tuples (and some of the good tuples) into multiple stripes and apply the above technique; for the rest of the good tuples, we use Indyk and McGregor's original technique. It can be shown using these two techniques that $\mathrm{Var}[D] \leq 27\mathrm{E}^2[D]$. We briefly sketch how to obtain this bound based on the technique in Section 3.1. We only give a sketch since our general analysis for $k > 2$ will also cover this case. Recall that we process all 4-tuples in $\mathcal{D}_3$ as follows: we group bad tuples (and some of the good tuples) into multiple stripes and apply the A-G inequality twice; for the rest of the good tuples, we use Indyk and McGregor's original technique.

We want to count the number of times that an edge $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$ (for $\mathbf{i}, \mathbf{j} \in [n]^3$) will be assigned weight through the translation. Recall that a 4-tuple $(A, B, C, D)$ is good if $A, B, C, D$ are four distinct points lying on the same plane. Let us consider a fixed edge $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$, where $\mathbf{i}_1 \neq \mathbf{j}_1$, $\mathbf{i}_2 \neq \mathbf{j}_2$, and $\mathbf{i}_3 \neq \mathbf{j}_3$ and calculate the weight assigned to this edge (the analysis for the rest of edges $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$, where $\mathbf{i}$ and $\mathbf{j}$ share one or more coordinates will be similar). Fix an edge $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$. The number of good 4-tuples that use $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$ is $3 \cdot 4! = 72$. Each good 4-tuple only uses the A-G inequality once and assigns a weight of one half to $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$. Therefore, $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$ is assigned a total weight of $72/2 = 36$ from good tuples.

Using the above technique, it can be shown that the amortized number of bad 4-tuples that assign weight to $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$ is $2 \cdot 4! = 48$. Because we use the A-G inequality twice to deal with bad 4-tuples, each bad 4-tuple assigns a weight of one quarter to $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$. Therefore, $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$ is assigned weight $48/4 = 12$ to amortize for bad tuples.

Finally, there could be degenerate cases where the 4-tuples only form a line. Fixing the edge $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$, there could be $\binom{4}{2} = 6$ such 4-tuples. Each degenerate 4-tuple assigns a weight of one to $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$. Therefore, $v_{\mathbf{i}}^2 v_{\mathbf{j}}^2$ is assigned weight 6 from the degenerate cases. Summing up these three cases, we have $\mathrm{Var}[D] \leq ((36 + 12 + 6)/2) \cdot \mathrm{E}^2[D] = 27\mathrm{E}^2[D]$. We need to divide $36 + 12 + 6$ by 2 because each $v_{\mathbf{i}} v_{\mathbf{j}}$ appears twice

in $\mathrm{E}^2[D]$.

Before presenting our analysis for general $k$, we briefly discuss the technique of Braverman and Ostrovsky [3]. The above manipulation of applying A-G inequality twice also appears in their analysis to achieve global transformations. However, they do not give the aforementioned geometric interpretation, but instead use the approach to give an inductive argument on the number of coordinates $k$. They decompose the tuples according to certain combinatorial properties, and each global transformation is used to gain one "good" coordinate. They pay a multiplicative factor at each level of the induction that yields an overall bound with a factor of $2^{O(k^2)}$. In contrast, in the next section, we show that a global transformation can be used without an induction on all tuples (both good and bad tuples) to assign all diagonals equal weight, which gives our improved bound of a factor of $3^k$.

### 3.2 A General Analysis for $k \geq 3$

Now we consider general $k$. The above argument would become very complicated quickly as $k$ increases, due to the difficulty of dealing with various degenerate cases. We obtain a more elegant analysis by a slightly different global transformation again based on applying the A-G inequality twice. Instead of dividing tuples into good and bad tuples, we divide tuples into $3^k$ classes and apply a global transformation to show that each class of tuples can be bounded by a copy of $\mathrm{E}^2[D]$, which implies $\mathrm{Var}[D] \leq 3^k \cdot \mathrm{E}^2[D]$.

We say that 4-tuple $(A, B, C, D)$ is in $\mathcal{D}_k$ if and only if

$$\forall s \in [k] : ((A_s = B_s) \wedge (C_s = D_s)) \vee ((A_s = C_s) \wedge (B_s = D_s)) \vee ((A_s = D_s) \wedge (B_s = C_s)) .$$

For each coordinate $s \in [k]$, $(A_s, B_s, C_s, D_s)$ is in one of the following forms: $(\alpha, \alpha, \beta, \beta)$, $(\alpha, \beta, \alpha, \beta)$, $(\alpha, \beta, \beta, \alpha)$, or $(\alpha, \alpha, \alpha, \alpha)$. Let us refer to the first three cases as *Type* $1, 2, 3$, respectively. The last case can be viewed as a degenerate case of any three types, which we denote by *Type* $*$. Thus, every 4-tuple $(A, B, C, D) \in \mathcal{D}_k$ has certain type in $\{1, 2, 3, *\}^k$. Now, we can divide $\mathcal{D}_k$ into $3^k$ classes $C_v$ with $v \in [3]^k$ in a natural way. A class $C_v$ consists of all tuples that are consistent with type with $v$, where $*$ is consistent with any of $1, 2, 3$. Note that some tuples can belong to multiple classes and this is fine for our purpose. Example 3.1 illustrates our notation.

**Example 3.1.** *Let $k = 5$. Consider*

$$\begin{aligned} A &= (1, 2, 5, 4, 3) \\ B &= (1, 1, 3, 4, 1) \\ C &= (0, 2, 5, 3, 1) \\ D &= (0, 1, 3, 3, 3). \end{aligned}$$

*This 4-tuple $(A, B, C, D) \in \mathcal{D}_5$ and according to the classification rule above, the type of this 4-tuple $v$ corresponds to the vector $(1, 2, 2, 1, 3)$. Now consider another 4-tuple in $\mathcal{D}_5$:*

$$\begin{aligned} A' &= (2, 1, 6, 4, 3) \\ B' &= (2, 1, 3, 4, 1) \\ C' &= (0, 1, 6, 5, 3) \\ D' &= (0, 1, 3, 5, 1). \end{aligned}$$

*This 4-tuple $(A', B', C', D')$ has type $(1, *, 2, 1, 3)$ and is consistent with any of the following types: $(1, 1, 2, 1, 3)$, $(1, 2, 2, 1, 3)$, $(1, 3, 2, 1, 3)$.*

For notational convenience, we introduce the following alternative notation to describe the tuples and classes. Let $I$ be a subset of $k$ and let $A$ be a point in $[n]^k$. We denote $A \mid_I$ as a projection of the point $A$ in $[n]^k$ to a subspace $[n]^{|I|}$ specified by the coordinates in $I$. For example, let $k = 4$, $A = (1, 1, 2, 3)$, and $I = \{1, 2, 4\}$. Then we have $A \mid_I = (1, 1, 3)$.

Let $I_1, I_2, I_3$ be an arbitrary partition of $[k]$, where $|I_1| = k_1$, $|I_2| = k_2$, and $|I_3| = k_3$. Let $\alpha \in [n]^{k_1}$, $\beta \in [n]^{k_2}$, and $\gamma \in [n]^{k_3}$. Then let us define $\pi_{I_1,I_2,I_3}(\alpha, \beta, \gamma)$ be a point in $[n]^k$ such that $\pi_{I_1,I_2,I_3}(\alpha, \beta, \gamma) \mid_{I_1} = \alpha$, $\pi_{I_1,I_2,I_3}(\alpha, \beta, \gamma) \mid_{I_2} = \beta$, and $\pi_{I_1,I_2,I_3}(\alpha, \beta, \gamma) \mid_{I_3} = \gamma$. Note that a partition $(I_1, I_2, I_3)$ corresponds to an element in $[3]^k$.

**Example 3.2.** *Let $k = 4$ and $I_1 = \{1, 4\}$, $I_2 = \{2\}$, and $I_3 = \{3\}$. Let $\alpha = (3, 5)$, $\beta = 5$, and $\gamma = 2$. Then $\pi_{I_1,I_2,I_3}(\alpha, \beta, \gamma) = (3, 5, 2, 5)$ since $\pi_{I_1,I_2,I_3}(\alpha, \beta, \gamma) \mid_{I_1} = (3, 5) = \alpha$, $\pi_{I_1,I_2,I_3}(\alpha, \beta, \gamma) \mid_{I_2} = 5 = \beta$, and $\pi_{I_1,I_2,I_3}(\alpha, \beta, \gamma) \mid_{I_3} = 2 = \gamma$.*

We say a triple $(I_1, I_2, I_3)$ an *ordered 3-partition* for $[k]$ if $I_1 \cup I_2 \cup I_3 = [k]$ and $I_i \cap I_j = \emptyset$ for $i \neq j$. Let $\mathcal{I}$ be the set of all ordered 3-partition for $k$. Note that the set $\mathcal{I}$ corresponds to the set $[3]^k$ in a natural way. Define

$$\mathrm{V}(I_1, I_2, I_3) \equiv \sum_{\substack{\alpha_1,\beta_1 \in [n]^{|I_1|} \\ \alpha_2,\beta_2 \in [n]^{|I_2|} \\ \alpha_3,\beta_3 \in [n]^{|I_3|}}} |v_{\pi_{I_1,I_2,I_3}(\alpha_1,\alpha_2,\alpha_3)} v_{\pi_{I_1,I_2,I_3}(\alpha_1,\beta_2,\beta_3)} v_{\pi_{I_1,I_2,I_3}(\beta_1,\alpha_2,\beta_3)} v_{\pi_{I_1,I_2,I_3}(\beta_1,\beta_2,\alpha_3)}|.$$

When $|I_i| = 0$ for some $i$, we shall view $\alpha_i = \beta_i = \emptyset$ representing a special symbol, instead of setting $\mathrm{V}(I_1, I_2, I_3) = 0$. For example, when $|I_1| = 0$,

$$\mathrm{V}(I_1, I_2, I_3) \equiv \sum_{\substack{\alpha_2,\beta_2 \in [n]^{|I_2|} \\ \alpha_3,\beta_3 \in [n]^{|I_3|}}} |v_{\pi_{I_1,I_2,I_3}(\emptyset,\alpha_2,\alpha_3)} v_{\pi_{I_1,I_2,I_3}(\emptyset,\beta_2,\beta_3)} v_{\pi_{I_1,I_2,I_3}(\emptyset,\alpha_2,\beta_3)} v_{\pi_{I_1,I_2,I_3}(\emptyset,\beta_2,\alpha_3)}|.$$

Notice that we have $|\mathcal{I}| = 3^k$. Also we shall see that $V(I_1, I_2, I_3)$ sums up all possible tuples that can be classified as $(c_1, c_2, ..., c_k)$, where $c_i = j$ if and only if $i \in I_j$. Furthermore, if a 4-tuple $(A, B, C, D) \in \mathcal{D}_k$ agrees on one or more coordinates, the term $v_A v_B v_C v_D$ may appear multiple times in different $V(I_1, I_2, I_3)$ for different ordered-3-partition. Therefore, the quantities $\mathrm{V}(I_1, I_2, I_3)$ indexed by $(I_1, I_2, I_3)$ for all possible ordered-3-partition $(I_1, I_2, I_3)$ cleanly cover all the tuples in $\mathcal{D}_k$. In other words,

$$\sum_{(A,B,C,D) \in \mathcal{D}_k} |v_A v_B v_C v_D| \leq \sum_{(I_1,I_2,I_3) \in \mathcal{I}} \mathrm{V}(I_1, I_2, I_3). \tag{4}$$

Finally, for $\alpha \in [n]^i$ and $\beta \in [n]^j$, define $\alpha \oplus \beta$ to be a point in $[n]^{i+j}$ by concatenating $\beta$ to the end of $\alpha$ in the natural way. Also, define $\alpha \oplus \beta \oplus \gamma = (\alpha \oplus \beta) \oplus \gamma$.

Now we are ready to state and prove our major lemma.

**Lemma 3.3.** *Let $x_1 = (x_1^1, ..., x_1^n), ..., x_k = (x_k^1, ..., x_k^n) \in \{-1, 1\}^n$ be $k$ independent vectors, where each vector is 4-wise independent. Let $v \in \mathrm{R}^{n^k}$ and for each $\mathbf{i} \in [n]^k$ let*

$$z_{\mathbf{i}} \equiv z_{i_1,...,i_k} \equiv \prod_{1 \leq j \leq k} x_j^{i_j}.$$

*Define $D = (\sum_{i \in [n]^k} z_{\mathbf{i}} v_{\mathbf{i}})^2$. Then $\mathrm{E}[D] = \sum_{i \in [n]^k} v_{\mathbf{i}}^2$ and $\mathrm{Var}[D] \leq 3^k \mathrm{E}^2[D]$.*

*Proof.* We have $\mathrm{E}[D] = \mathrm{E}[(\sum_{\mathbf{i}} z_{\mathbf{i}} v_{\mathbf{i}})^2] = \sum_{\mathbf{i}} v_{\mathbf{i}}^2 \mathrm{E}[z_{\mathbf{i}}^2] + \sum_{\mathbf{i} \neq \mathbf{j}} v_{\mathbf{i}} v_{\mathbf{j}} \mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}}]$. For all $\mathbf{i} \in [n]^k$, we know $z_{\mathbf{i}}^2 = 1$. On the other hand, $\mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}}] = \mathrm{E}[(\prod_l x_l^{i_l})(\prod_l x_l^{j_l})] = \prod_l \mathrm{E}[x_l^{i_l} x_l^{j_l}]$. Furthermore, since $x_l^{i_l} x_l^{j_l} \in \{-1, 1\}$ for any $l$ and $\Pr[x_l^{i_l} x_l^{j_l} = 1] = 1/2$, we have $\mathrm{E}[x_l^{i_l} x_l^{j_l}] = 0$. Therefore, $\mathrm{E}[z_{\mathbf{i}} z_{\mathbf{j}}] = 0$ for any $l$, and consequently $\mathrm{E}[D] = \sum_{\mathbf{i} \in [n]^k} v_{\mathbf{i}}^2$.

Next we bound for variance. The setup of the proof is similar to Lemma 2.2, except in a higher dimensional space. First we have $\mathrm{Var}[X] \leq \sum_{A,B,C,D \in [n]^k} |\mathrm{E}[z_A z_B z_C z_d]| \cdot |v_A v_B v_C v_D| \leq \sum_{A,B,C,D \in \mathcal{D}_k} |v_A v_B v_C v_D|$,

By Inequality 4, we have

$$\sum_{(A,B,C,D) \in \mathcal{D}_k} |v_A v_B v_C v_D| \leq \sum_{(I_1,I_2,I_3) \in \mathcal{I}} \mathrm{V}(I_1, I_2, I_3).$$

Now we bound each individual $\mathrm{V}(I_1, I_2, I_3)$ for an arbitrary ordered 3-partition $(I_1, I_2, I_3)$ by $\mathrm{E}^2[D]$. Let $|I_1| = k_1$, $|I_2| = k_2$, and $|I_3| = k_3$. To simplify the notation, in what follows, the operator $\pi$ means the operator $\pi_{I_1,I_2,I_3}$. In case $k_1 > 0$, we have

$$\begin{aligned}
& \mathrm{V}(I_1, I_2, I_3) \\
= & \sum_{\alpha_1\beta_1 \in [n]^{k_1}} \sum_{\alpha_2,\beta_2 \in [n]^{k_2}} \sum_{\alpha_3,\beta_3 \in [n]^{k_3}} |v_{\pi(\alpha_1,\alpha_2,\alpha_3)} v_{\pi(\alpha_1,\beta_2,\beta_3)} v_{\pi(\beta_1,\alpha_2,\beta_3)} v_{\pi(\beta_1,\beta_2,\alpha_3)}| \\
= & \sum_{\substack{\alpha_2,\beta_2 \in [n]^{k_2} \\ \alpha_3,\beta_3 \in [n]^{k_3}}} \left( \left( \sum_{\alpha_1 \in [n]^{k_1}} |v_{\pi(\alpha_1,\alpha_2,\alpha_3)} v_{\pi(\alpha_1,\beta_2,\beta_3)}| \right) \cdot \left( \sum_{\beta_1 \in [n]^{k_1}} |v_{\pi(\beta_1,\alpha_2,\beta_3)} v_{\pi(\beta_1,\beta_2,\alpha_3)}| \right) \right) \\
\leq & \frac{1}{2} \sum_{\substack{\alpha_2,\beta_2 \in [n]^{k_2} \\ \alpha_3,\beta_3 \in [n]^{k_3}}} \left( \left( \sum_{\alpha_1 \in [n]^{k_1}} |v_{\pi(\alpha_1,\alpha_2,\alpha_3)} v_{\pi(\alpha_1,\beta_2,\beta_3)}| \right)^2 + \left( \sum_{\beta_1 \in [n]^{k_1}} |v_{\pi(\beta_1,\alpha_2,\beta_3)} v_{\pi(\beta_1,\beta_2,\alpha_3)}| \right)^2 \right),
\end{aligned}$$

We may also be able to bound each term in the last line individually:

$$\begin{aligned}
& \left( \sum_{\alpha_1 \in [n]^{k_1}} |v_{\pi(\alpha_1,\alpha_2,\alpha_3)} v_{\pi(\alpha_1,\beta_2,\beta_3)}| \right)^2 \\
= & \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} |v_{\pi(\alpha_1,\alpha_2,\alpha_3)} v_{\pi(\alpha_1,\beta_2,\beta_3)} v_{\pi(\beta_1,\alpha_2,\alpha_3)} v_{\pi(\beta_1,\beta_2,\beta_3)}| \quad \text{(expand the square term)} \\
= & \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} |v_{\pi(\alpha_1,\alpha_2,\alpha_3)} v_{\pi(\beta_1,\beta_2,\beta_3)}| \cdot |v_{\pi(\beta_1,\alpha_2,\alpha_3)} v_{\pi(\alpha_1,\beta_2,\beta_3)}| \quad \text{(regroup the product)} \\
\leq & \frac{1}{2} \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} \left( v^2_{\pi(\alpha_1,\alpha_2,\alpha_3)} v^2_{\pi(\beta_1,\beta_2,\beta_3)} + v^2_{\pi(\beta_1,\alpha_2,\alpha_3)} v^2_{\pi(\alpha_1,\beta_2,\beta_3)} \right) \quad \text{(“translate” into diagonals)} \\
= & \frac{1}{2} \left( \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} v^2_{\pi(\alpha_1,\alpha_2,\alpha_3)} v^2_{\pi(\beta_1,\beta_2,\beta_3)} \right) + \frac{1}{2} \left( \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} v^2_{\pi(\beta_1,\alpha_2,\alpha_3)} v^2_{\pi(\alpha_1,\beta_2,\beta_3)} \right) \quad \text{(regroup)} \\
= & \frac{1}{2} \left( \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} v^2_{\pi(\alpha_1,\alpha_2,\alpha_3)} v^2_{\pi(\beta_1,\beta_2,\beta_3)} \right) + \frac{1}{2} \left( \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} v^2_{\pi(\alpha_1,\alpha_2,\alpha_3)} v^2_{\pi(\beta_1,\beta_2,\beta_3)} \right) \quad \text{(rename the indices)} \\
= & \left( \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} v^2_{\pi(\alpha_1,\alpha_2,\alpha_3)} v^2_{\pi(\beta_1,\beta_2,\beta_3)} \right)
\end{aligned}$$

Similarly,

$$\left( \sum_{\beta_1 \in [n]^{k_1}} |v_{\pi(\beta_1,\alpha_2,\beta_3)} v_{\pi(\beta_1,\beta_2,\alpha_3)}| \right)^2 \leq \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} v^2_{\pi(\alpha_1,\alpha_2,\beta_3)} v^2_{\pi(\beta_1,\beta_2,\alpha_3)}.$$

Therefore,

$$\begin{aligned}
& \frac{1}{2} \sum_{\substack{\alpha_2,\beta_2 \in [n]^{k_2} \\ \alpha_3,\beta_3 \in [n]^{k_3}}} \left( \left( \sum_{\alpha_1 \in [n]^{k_1}} |v_{\pi(\alpha_1,\alpha_2,\alpha_3)} v_{\pi(\alpha_1,\beta_2,\beta_3)}| \right)^2 + \left( \sum_{\beta_1 \in [n]^{k_1}} |v_{\pi(\beta_1,\alpha_2,\beta_3)} v_{\pi(\beta_1,\beta_2,\alpha_3)}| \right)^2 \right) \\
\leq & \frac{1}{2} \sum_{\substack{\alpha_2,\beta_2 \in [n]^{k_2} \\ \alpha_3,\beta_3 \in [n]^{k_3}}} \left( \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} v^2_{\pi(\alpha_1,\alpha_2,\alpha_3)} v^2_{\pi(\beta_1,\beta_2,\beta_3)} + \sum_{\alpha_1,\beta_1 \in [n]^{k_1}} v^2_{\pi(\alpha_1,\alpha_2,\beta_3)} v^2_{\pi(\beta_1,\beta_2,\alpha_3)} \right) \\
= & \sum_{\substack{\alpha_1 \oplus \alpha_2 \oplus \alpha_3 \in [n]^k \\ \beta_1 \oplus \beta_2 \oplus \beta_3 \in [n]^k}} v^2_{\pi(\alpha_1,\alpha_2,\alpha_3)} v^2_{\pi(\beta_1,\beta_2,\beta_3)} \\
= & \sum_{\substack{\alpha_1 \oplus \alpha_2 \oplus \alpha_3 \in [n]^k \\ \beta_1 \oplus \beta_2 \oplus \beta_3 \in [n]^k}} v^2_{\alpha_1 \oplus \alpha_2 \oplus \alpha_3} v^2_{\beta_1 \oplus \beta_2 \oplus \beta_3} \\
= & \mathrm{E}^2[D].
\end{aligned}$$

When $k_1 = 0$, we have

$$\begin{aligned}
\mathrm{V}(I_1, I_2, I_3) &= \sum_{\alpha_2,\beta_2 \in [n]^{k_2}} \sum_{\alpha_3,\beta_3 \in [n]^{k_3}} \left| v_{\pi(\emptyset,\alpha_2,\alpha_3)} v_{\pi(\emptyset,\beta_2,\beta_3)} v_{\pi(\emptyset,\alpha_2,\beta_3)} v_{\pi(\emptyset,\beta_2,\alpha_3)} \right| \\
&\leq \frac{1}{2} \sum_{\alpha_2,\beta_2} \sum_{\alpha_3,\beta_3} \left( v^2_{\pi(\emptyset,\alpha_2,\alpha_3)} v^2_{\pi(\emptyset,\beta_2,\beta_3)} + v^2_{\pi(\emptyset,\alpha_2,\beta_3)} v^2_{\pi(\emptyset,\alpha_2,\beta_3)} \right) \\
&= \sum_{\substack{\alpha_2,\beta_2 \\ \alpha_3,\beta_3}} v^2_{\pi(\emptyset,\alpha_2,\alpha_3)} v^2_{\pi(\emptyset,\beta_2,\beta_3)} \\
&= \sum_{\substack{\alpha_2 \oplus \alpha_3 \in [n]^k \\ \beta_2 \oplus \beta_3 \in [n]^k}} v^2_{\alpha_2 \oplus \alpha_3} v^2_{\beta_2 \oplus \beta_3} \\
&= \mathrm{E}^2[D].
\end{aligned}$$

Therefore, in any case, $\mathrm{V}(I_1, I_2, I_3) \leq \mathrm{E}^2[D]$ for any ordered 3-partition $(I_1, I_2, I_3)$ and

$$\sum_{(A,B,C,D) \in \mathcal{D}_k} |v_A v_B v_C v_D| \leq \sum_{(I_1,I_2,I_3) \in \mathcal{I}} \mathrm{V}(I_1, I_2, I_3) \leq |\mathcal{I}| \mathrm{E}^2[D] = 3^k \mathrm{E}^2[D].$$

□

We note that $|\mathcal{D}_k| > \binom{4}{2}^k \binom{n}{2}^k = 3^k n^{2k} - o(n^{2k})$. If we set $v \in \mathrm{R}^{n^k}$ to be a uniform vector, we have $\mathrm{E}(D^2) \sim 3^k \mathrm{E}^2(D)$. Therefore, we do not expect to be able to improve this result without a different approach or further additional techniques. Following exactly the approach of Theorem 2.5, we obtain our main result from Lemma 3.3.

**Theorem 3.4.** *For every $\epsilon > 0$ and $\delta > 0$, there exists a randomized algorithm that computes, given a sequence $a_1, ..., a_m$ of $k$-tuples, in one pass and using $O(3^k \epsilon^{-2} \log \frac{1}{\delta} (\log m + \log n))$ memory bits, a number* Med *so that the probability* Med *deviates from the $\ell_2$ distance between product and joint distribution by more than $\epsilon$ is at most $\delta$.*

# 4 Conclusion

There remain several open questions left in this space. Lower bounds, particularly bounds that depend non-trivially on the dimension $k$, would be useful. There may still be room for better algorithms for testing $k$-wise independence in this manner using the $\ell_2$ norm, and there certainly appears to be possible improvements in the harder case of the $\ell_1$ norm. A natural generalization would be to find a particularly efficient algorithm for testing $k$-out-of-$n$-wise independence (other than handling each set of $k$ variable separately). More generally, a question given in [5], to identify random variables whose correlation exceeds some threshold according to some measure, remains widely open.

# References


[1] N. Alon, L. Babai, A. Itai, "A Fast and simple randomized parallel algorithm for the maximal independent set problem," in *Journal of Algorithms* vol.7, issue 4, pp.567-583, 1986.

[2] N. Alon, Y. Matias, M. Szegedy, "The space complexity of approximating the frequency moments," in *Journal of Computer and System Sciences,* pp 137-147, 1999.

[3] V. Braverman, R. Ostrovsky, "Measuring $k$-wise independence of streaming data under $L_2$ norm," `http://arxiv.org/abs/0806.4790`.

[4] V.Braverman, R. Ostrovsky, "Measuring independence of datasets," `http://arxiv.org/abs/0903.0034`.

[5] P. Indyk, A. McGregor, "Declaring independence via the sketching of sketches", in *Proceedings of the 19th annual ACM-SIAM symposium on Discrete algorithms,* 2008.

[6] E. L. Lehmann, "Testing statistical hypotheses," *Wadsworth and Brooks/Cole*, 1986.

[7] M. Mitzenmacher, E. Upfal, "Probability and computing: randomized algorithms and probabilistic analysis," *Cambridge University Press*, 2005.

[8] V. Poosala, Y. E. Ioannidis, "Selectivity estimation without the attribute value independence assumption" *Proceedings of the 23rd International Conference on Very Large Data Bases,* pp.486-495, 1997.

[9] M. Thorup, Y. Zhang, "Tabulation based 4-universal hashing with applications to second moment estimation,"in *Proceedings of the 19th annual ACM-SIAM symposium on Discrete algorithms,* 2004.