Scalable Hash-Based Estimation of Divergence Measures

Morteza Noshad and Alfred O. Hero III

University of Michigan, Electrical Engineering and Computer Science, Ann Arbor, Michigan, U.S.A

Abstract

We propose a scalable divergence estimation method based on hashing. Consider two continuous random variables $X$ and $Y$ whose densities have bounded support. We consider a particular locality sensitive random hashing, and consider the ratio of samples in each hash bin having non-zero numbers of $Y$ samples. We prove that the weighted average of these ratios over all of the hash bins converges to f-divergences between the two samples sets. We show that the proposed estimator is optimal in terms of both MSE rate and computational complexity. We derive the MSE rates for two families of smooth functions; the Hölder smoothness class and differentiable functions. In particular, it is proved that if the density functions have bounded derivatives up to the order $d/2$, where $d$ is the dimension of samples, the optimal parametric MSE rate of $O(1/N)$ can be achieved. The computational complexity is shown to be $O(N)$, which is optimal. To the best of our knowledge, this is the first empirical divergence estimator that has optimal computational complexity and achieves the optimal parametric MSE estimation rate.

1 Introduction

Information theoretic measures such as Shannon entropy, mutual information, and the Kullback-Leibler (KL) divergence have a broad range of applications in information theory, statistics and machine learning [1,2]. When we have two or more data sets and we are interested in finding the correlation or dissimilarity between them, Shannon mutual information or KL-divergence is often used. Rényi and f-divergence measures are two well studied generalizations of KL-divergence which comprise many important divergence measures such as KL-divergence, total variation distance, and $\alpha$-divergence [4,5].

Non-parametric estimators are a major class of divergence estimators, for which minimal assumptions on the density functions are considered. Some of the non-parametric divergence estimators are based on density plug-in estimators such as $k$-NN [6], KDE [7], and histogram [8]. A few researchers, on the other hand, have proposed direct estimation methods such as graph theoretic nearest neighbor ratio (NNR) [9]. In general, plug-in estimation methods suffer from high computational complexity, which make them unsuitable for large scale applications.

Recent advances on non-parametric divergence estimation have been focused on the MSE convergence rates of the estimator. Singh et al in [7] proposed a plug-in KDE estimator for Rényi divergence that achieves the MSE rate of $O(1/N)$ when the densities are at least $d$ times differentiable, and the support boundaries are sufficiently smooth. Kandasamy et al proposed a similar plug-in KDE estimator and extend the optimal MSE rate to densities that are at least $d/2$ differentiable [10]. However, they ignore a major source of error due to the boundaries. Moon et al proposed a weighted ensemble method to improve the MSE rate of plug-in KDE estimators [11]. The proposed estimator for f-divergence achieves the optimal MSE rate when the densities are at least $(d+1)/2$ times differentiable. They also assume stringent smoothness conditions at the support set boundary.

Noshad et al proposed a graph theoretic direct estimation method based on nearest neighbor ratios (NNR) [9]. Their estimator is simple and computationally more tractable than other competing estimators, and can achieve the optimal MSE rate of $O(1/N)$ for densities that are at least $d$ times differentiable. Although their basic estimator does not require any smoothness assumptions on the support set boundary, the ensemble estimator variant of their estimator does.

In spite of achieving the optimal theoretical MSE rate
by aforementioned estimators, there remain serious. The first challenge is the high computational complexity of the estimator. Most KDE based estimators require runtime complexity of $O(N^2)$, which is not suitable for large scale applications. The NNR estimator proposed in [2] has the runtime complexity of $O(kN \log N)$, which is faster than the previous estimators. However, in [2] they require $k$ to grow sub-linearly with $N$, which results in much higher complexity than linear runtime complexity. The other issue is the smoothness assumptions made on the support set boundary. Almost all previously proposed estimators assume extra smoothness conditions on the boundaries, which may not hold practical applications. For example, the method proposed in [2] assumes that the density derivatives up to order $d$ vanish at the boundary. Also it requires numerous computations at the support boundary, which become complicated when the dimension increases. The Ensemble NNR estimator in [9] assumes that the density derivatives vanish at the boundary. To circumvent this issue, Moon et al [11] assumed smoothness conditions at the support set boundary. However, these conditions may not hold in practice.

In this paper we propose a low complexity divergence estimator that can achieve the optimal MSE rate of $O(1/N)$ for the densities with bounded derivatives of up to $d/2$. Our estimator has optimal runtime complexity of $O(N)$, which makes it an appropriate tool for large scale applications. Also in contrast to other competing estimators, our estimator does not require stringent smoothness assumptions on the support set boundary.

The structure of the proposed estimator borrows ideas from hash based methods for KNN search and graph constructions problems [12,13], as well as from the NNR estimator proposed in [9]. The advantage of hash based methods is that they can be used to find the approximate nearest neighbor points with lower complexity as compared to the exact k-NN search methods. This suggests that fast and accurate algorithms for divergence estimation may be derived from hashing approximations of k-NN search. Noshad et al [9] consider the k-NN graph of Y in the joint data set $(X,Y)$, and show that the average exponentiated ratio of the number of X points to the number of Y points among all k-NN points is proportional to the Rényi divergence between the X and Y densities. It turns out that for estimation of the density ratio around each point we really do not need to find the exact k-NN points, but only need sufficient local samples from X and Y around each point. By using a randomized locality sensitive hashing (LSH), we find the closest points in Euclidean space. In this manner, applying ideas from the NNR estimation and hashing techniques to KNN search problem, we obtain a more efficient divergence estimator. Consider two sample sets $X$ and $Y$ with a bounded density support. We use a particular two-level locality sensitive random hashing, and consider the ratio of samples in each bin with a number of Y samples. We prove that the weighted average of these ratios over all of the bins can be made to converge almost surely to f-divergences between the two samples populations. We also argue that using the ensemble estimation technique provided in [2], we can achieve the optimal parametric rate of $O(1/N)$. Furthermore, using a simple algorithm for online estimation method has $O(N)$ complexity and $O(1/N)$ convergence rate, which is the first optimal online estimator of its type.

## 2 Hash-Based Estimation

In this section, we first introduce the f-divergence measure and propose a hash-based estimator. We outline the main theoretical results which will be proven in section 4.

Consider two density functions $f_1$ and $f_2$ with common bounded support set $\mathcal{X} \subseteq \mathbb{R}^d$.

The f-divergence is defined as follows [5].

$$D_g (f_1(x)\|f_2(x)) := \int g \left( \frac{f_1(x)}{f_2(x)} \right) f_2(x) dx = \mathbb{E}_{f_2} \left[ g \left( \frac{f_1(x)}{f_2(x)} \right) \right],$$  \hspace{1cm} (1)

where $g$ is a smooth and convex function such that $g(1) = 0$. KL-divergence, Hellinger distance and total variation distance are particular cases of this family. Note that for estimation, we don’t need convexity of $g$ and $g(1) = 0$. conditions. Assume that the densities are lower bounded by $C_L > 0$ and upper bounded by $C_U$. Assume $f_1$ and $f_2$ belong to the Hölder smoothness class with parameter $\gamma$.

**Definition** Given a support $\mathcal{X} \subseteq \mathbb{R}^d$, a function $f : \mathcal{X} \rightarrow \mathbb{R}$ is called Hölder continuous with parameter $0 < \gamma \leq 1$, if there exists a positive constant $G_f$, possibly depending on $f$, such that

$$|f(y) - f(x)| \leq G_f \|y - x\|^\gamma,$$

for every $x \neq y \in \mathcal{X}$.

The function $g$ in (1) is also assumed to be Lipschitz continuous; i.e. $g$ is Hölder continuous with $\gamma = 1$.

**Remark 1** The $\gamma$-Hölder smoothness family comprises a large class of continuous functions including continuously differentiable functions and Lipschitz continuous functions. Also note that for $\gamma > 1$, any $\gamma$-Hölder
continuous function on any bounded and continuous support is constant.

Hash-Based Divergence Estimator: Consider the i.i.d samples \( X = \{X_1, ..., X_N \} \) drawn from \( f_1 \) and \( Y = \{Y_1, ..., Y_M \} \) drawn from \( f_2 \). Define the fraction \( \eta := M/N \). We define the set \( Z := X \cup Y \). We define a positive real valued constant \( \epsilon \) as a user-selectable parameter of the estimator to be defined in \( \gamma \). We define the hash function \( H_1 : \mathbb{R}^d \rightarrow \mathbb{Z}^d \) as

\[
H_1(x) = [h_1(x_1), h_1(x_2), ..., h_1(x_d)],
\]

where \( x_i \) is the projection of \( x \) on the \( i \)th coordinate, and \( h_1(x) : \mathbb{R} \rightarrow \mathbb{Z} \) is defined as

\[
h_1(x) = \left\lfloor \frac{x + b}{\epsilon} \right\rfloor,
\]

for fixed \( b \). Let \( F := \{1, 2, ..., F\} \), where \( F := c_H N \) and \( c_H \) is a fixed real number. We define a random hash function \( H_2 : \mathbb{Z}^d \rightarrow F \) with a uniform density on the output and consider the combined hashing \( H(x) := H_2(H_1(x)) \), which maps the points in \( \mathbb{R}^d \) to \( F \).

Consider the mappings of the sets \( X \) and \( Y \) using the hash function \( H(x) \), and define the vectors \( \mathcal{N} \) and \( \mathcal{M} \) to respectively contain the number of collisions for each output bucket from the set \( F \). We represent the bins of the vectors \( \mathcal{N} \) and \( \mathcal{M} \) respectively by \( N_i \) and \( M_i \), \( 1 \leq i \leq F \).

The hash based f-divergence estimator is defined as

\[
\hat{D}_g(X,Y) := \max \left\{ \frac{1}{M} \sum_{i \in F} M_i \bar{g} \left( \frac{n N_i}{M_i} \right), 0 \right\},
\]

where \( \bar{g}(x) := \max \{ g(x), g(C_L/C_U) \} \).

Remark 2 In order for the estimator to be asymptotically unbiased, \( \epsilon \) needs to be a function of \( N \). The optimum bias rate of \( O\left( \frac{1}{N^{\gamma/(\gamma+d)}} \right) \) can be achieved for \( \epsilon = \left( \frac{1}{N} \right)^{\gamma/(\gamma+d)} \).

Theorem 2.2 Let \( \eta = M/N \) be fixed. The variance of the estimator \( \gamma \) can be bounded as

\[
\forall \left[ \hat{D}_g(X,Y) \right] \leq O \left( \frac{1}{N} \right).
\]

Remark 3 The same variance bound holds for the random variable \( \rho_1 := \frac{N}{M} \). The bias and variance results easily extend to Rényi divergence estimation.

Algorithm 1: Histogram Estimator of f-Divergence

**Input**: Data sets \( X = \{X_1, ..., X_N\} \), \( Y = \{Y_1, ..., Y_M\} \)

1. \( X' \leftarrow H(X) \)
2. \( Y' \leftarrow H(Y) \)
3. for each \( i \in F \) do
   4. \( N_i \leftarrow \lvert X' = i \rvert \)
   5. \( M_i \leftarrow \lvert Y' = i \rvert \)
6. \( \hat{D} \leftarrow \max \left\{ \frac{1}{M} \sum_{M_i > 0} M_i \bar{g}(n N_i/M_i), 0 \right\} \)

**Output**: \( \hat{D} \)

We next show that, when \( f_1 \) and \( f_2 \) belong to the family of differentiable densities, we can improve the bias rate by applying the ensemble estimation approach in [3,11]. The Ensemble Hash-based (EHB) estimator is defined as follows.

Definition Assume that the density functions are in the Hölder space \( \Sigma(\gamma, L) \), which consists of functions on \( X \) continuous derivatives up to order \( q = \lceil \gamma \rceil \geq d \) and the \( q \)th partial derivatives are Hölder continuous with exponent \( \gamma' = \gamma - q \). Let \( T := \{t_1, ..., t_T\} \) be a set of index values with \( t_i < c \), where \( c > 0 \) is a constant. Let \( \epsilon(t) := \left( t N^{-1/2d} \right) \). The weighted ensemble estimator is defined as

\[
\hat{D}_w := \sum_{t \in T} w(t) \hat{D}_{\epsilon(t)},
\]

where \( \hat{D}_{\epsilon(t)} \) is the hash based estimator of f-divergence, with the hashing parameter of \( \epsilon(t) \).
Theorem 2.3 Let $T > d$ and $w_0$ be the solution to:

$$
\min_w \|w\|_2 \quad \text{subject to} \quad \sum_{t \in T} w(t) = 1,
\sum_{t \in T} w(t)t^{i/d} = 0, \quad i \in \mathbb{N}, \quad i \leq d.
$$

Then the MSE rate of the ensemble estimator $\hat{D}_{w_0}$ is $O(1/N)$.

3 Online Divergence Estimation

In this section we study the problem of online divergence estimation. In this setting we consider two data streams $X = \{X_1, X_2, ..., X_N\}$ and $Y = \{Y_1, Y_2, ..., Y_N\}$ with i.i.d samples, and we are interested in estimating the divergence between two data sets. The number of samples increase over time and an efficient update of the divergence estimate is desired. The time complexity of a batch update, which uses the entire update runtime complexity of order $O(1)$. Define the sets $X^N := \{X_i\}_{i=1}^N$, $Y^N := \{Y_i\}_{i=1}^N$, the number of $X$ and $Y$ samples in each partition, and the divergence estimate between $X^N$ and $Y^N$. Consider updating the estimator with new samples $X_{N+1}$ and $Y_{N+1}$. In the first and second lines of algorithm 2, the new samples are added to the datasets and the values of $N_i$ and $M_i$ of the bins in which the new samples fall. We can find these bins in $O(1)$ using a simple hashing. Note that once $N_i$ and $M_i$ are updated, the divergence measure can be updated, but the number of bins is not increased, by Theorem 2.1, it is clear that the bias will not be reduced. Since increasing the number of bins requires recomputing the bin partitions, a brute force rebinning approach would have order $O(N)$ complexity, and it were updated $N$ times, the total complexity would be $O(N^2)$. Here we use a trick and update the hash function only when $N + 1$ is a power of 2. In the following theorem, which is proved in appendix, we show that the MSE rate of this algorithm is order $O(1/N)$ and the total rebinning computational complexity is order $O(N)$.

**Theorem 3.1** MSE rate of the online divergence estimator shown in Algorithm 2 is order $O(1/N)$ and the total computational complexity is order $O(N)$.

Algorithm 2: Online Divergence Estimation

**Input**: $X^N := \{X_i\}_{i=1}^N$, $Y^N := \{Y_i\}_{i=1}^N$

**Output**: $\hat{D}$

1. Add $X_{N+1}$ and update $N_k$ s.t $H(X_{N+1}) = k$.
2. Add $Y_{N+1}$ and update $M_l$ s.t $H(Y_{N+1}) = l$.
3. If $N + 1 = 2^i$ for some $i$, Then
   4. Update $\epsilon$ to the optimum value
   5. Re-hash $X$ and $Y$
   6. Re-compute $N_i$ and $M_i$ for $0 \leq i \leq F$
   7. Update $\hat{D}$

4 Proofs

In this section we derive the bias bound for the densities in Hölder smoothness class, stated in Theorem 2.1. For the proofs of variance bound in Theorem 2.2, convergence rate of EHB estimator in Theorem 2.3 and online divergence estimator in Theorem 3.1 we refer the reader to the Appendix, provided as a supplementary pdf file.

Consider the mapping of the $X$ and $Y$ points by the hash function $H_1$, and let the vectors $\{V_i\}_{i=1}^L$ represent the distinct mappings of $X$ and $Y$ points under $H_1$. Here $L$ is the number of distinct outputs of $H_1$. In the following lemma we prove an upper bound on $L$.

**Lemma 4.1** Let $f(x)$ be a density function with bounded support $X \subseteq \mathbb{R}^d$. Then if $L$ denotes the number of distinct outputs of the hash function $H_1$ (defined in 3) of i.i.d points with density $f(x)$, we have

$$L \leq O\left(\frac{1}{\epsilon^d}\right).$$

**Proof** Let $x = [x_1, x_2, ..., x_d]$ and define $X_l$ as the region defined as

$$X_l := \{x | -c_X \leq x_i \leq c_X, 1 \leq i \leq d\},$$

where $c_X$ is a constant such that $X \subseteq X_l$.

$L$ is clearly not greater than the total number of bins created by splitting the region $X$ into partitions of volume $\epsilon^d$. So we have

$$L \leq \frac{(2c_X)^d}{\epsilon^d}.$$

**Proof of Theorem 2.1** Let $\{N_i\}_{i=1}^L$ and $\{M_j\}_{j=1}^L$ respectively denote the number of collisions of $X$ and
We represent the second term in (13) by $P(E_i) = \left(1 - \frac{1}{F}\right)^L + \left(\frac{1}{F}\right) \left(\frac{F-1}{F}\right)^{L-1}$
\[
= 1 - O\left(\frac{L}{F}\right). \tag{12}
\]

By definition,
\[
\hat{D}_g(X,Y) := \frac{1}{M} \sum_{i : \text{i \in \hat{F}}} M_i \tilde{g} \left(\frac{\eta N_i}{M_i}\right).
\]

Therefore
\[
\mathbb{E} \left[ \hat{D}_g(X,Y) \right] = \frac{1}{M} \mathbb{E} \left[ \sum_{i : \text{i \in \hat{F}}} M_i \tilde{g} \left(\frac{\eta N_i}{M_i}\right) \right]
= \frac{1}{M} \sum_{i : \text{i \in \hat{F}}} P(E_i) \mathbb{E} \left[ M_i \tilde{g} \left(\frac{\eta N_i}{M_i}\right) \right] E_i
+ \frac{1}{M} \sum_{i : \text{i \in \hat{F}}} P(\bar{E}_i) \mathbb{E} \left[ M_i \tilde{g} \left(\frac{\eta N_i}{M_i}\right) \right] \bar{E}_i.
\]
\[
\tag{13}
\]

We represent the second term in (13) by $B_H$. $B_H$ has the interpretation as the bias error due to collisions in hashing. Remember that $\bar{E}_i$ is defined as the event that there is no collision at bin $i$ for the hash function $H_2$ with inputs $\{V_i\}_{i=1}^L$. For proving as upper bound on $B_H$, we first need to compute an upper bound on $\sum_{i=1}^L \mathbb{E} \left[ M_i | \bar{E}_i \right]$. This is stated in the following lemma.

\textbf{Lemma 4.2} We have
\[
\sum_{i \in \hat{F}} \mathbb{E} \left[ M_i | \bar{E}_i \right] \leq O(L) \tag{14}
\]

\textbf{Proof} Define $A_i := \{ j : H_2(V_j) = i \}$. For each $i$ we can rewrite $M_i$ as
\[
M_i = \sum_{j=1}^L \mathbb{I}_{A_i}(j) M'_j. \tag{15}
\]

Thus,
\[
\sum_{i \in \hat{F}} \mathbb{E} \left[ M_i | \bar{E}_i \right] = \sum_{i \in \hat{F}} \sum_{j=1}^L \mathbb{I}_{A_i}(j) M'_j \left[ \mathbb{E} \left[ E_i \right] \right]
 = \sum_{i \in \hat{F}} \sum_{j=1}^L M'_j \mathbb{E} \left[ \mathbb{I}_{A_i}(j) | \bar{E}_i \right] \mathbb{I}_{A_i}(j)
 = \sum_{i \in \hat{F}} \sum_{j=1}^L M'_j P(j \in A_i, \bar{E}_i)
 = \sum_{i \in \hat{F}} \sum_{j=1}^L M'_j \frac{P(j \in A_i, \bar{E}_i)}{P(E_i)} \tag{16}
\]

where $P(j \in A_i, \bar{E}_i)$ and $P(E_i)$ can be derived as
\[
P(j \in A_i, \bar{E}_i) = \frac{1}{F} \left[ 1 - \left(\frac{F-1}{F}\right)^{L-1} \right] \quad \text{and} \quad \tag{17}
P(E_i) = 1 - P(E_i) = O\left(\frac{L}{F}\right). \tag{18}
\]

Plugging in (17) and (18) in (16) results in
\[
\sum_{i \in \hat{F}} \mathbb{E} \left[ M_i | \bar{E}_i \right] = \sum_{i \in \hat{F}} \sum_{j=1}^L M'_j O\left(\frac{1}{F}\right)
 = \sum_{i \in \hat{F}} \sum_{j=1}^L O\left(\frac{1}{F}\right)
 = O\left(\frac{M}{F}\right)
 = O(L), \tag{19}
\]

where in the third line we use $\eta = M/N$ and $F = c_H N$. Now in the following lemma we prove a bound on $B_H$.

\textbf{Lemma 4.3} Let $L$ denote the number of distinct outputs of the hash function $H_1$ of the $X$ and $Y$ sample points. The bias of estimator $\tilde{g}$ due to hashing collision can be upper bounded by
\[
B_H \leq O\left(\frac{L^2 N^2}{N^2}\right) \tag{20}
\]

\textbf{Proof} From the definition of $B_H$ we can write
\[ \mathbb{E}_H := \frac{1}{M} \sum_{i \leq F} P(E_i) \mathbb{E} \left[ M_i g \left( \frac{n_i}{M_i} \right) \left| E_i \right. \right] \]

\[ = \frac{P(E_1)}{M} \sum_{i \leq F} \mathbb{E} \left[ M_i g \left( \frac{n_i}{M_i} \right) \left| E_1 \right. \right] \leq \frac{P(E_1)\bar{g}(R_{\text{max}})}{M} \sum_{i \leq F} \mathbb{E} \left[ M_i \left| E_1 \right. \right] \]

\[ = \frac{P(E_1)\bar{g}(R_{\text{max}})}{M} O(L) = O\left( \frac{L^2}{N^2} \right), \quad (21) \]

where in the second line we used the fact that \( P(E_i) = P(E_1) \). In the third line we used the upper bound for \( \bar{g} \), and in the fourth line we used the result in equation (19).

Now we are ready to continue the proof of the bias bound in (13). Let \( E \) be defined as the event that there is no collision for the hash function \( H_2 \), and all of its outputs are distinct, that is, \( E = \cap_{i=1}^{F} E_i \) (13) can be written as

\[ \mathbb{E} \left[ \tilde{D}_\gamma(X,Y) \right] \]

\[ = \frac{1}{M} \sum_{i \leq F} P(E_i) \mathbb{E} \left[ M_i \bar{g} \left( \frac{n_i}{M_i} \right) \left| E_i \right. \right] + O \left( \frac{L}{F} \right) \]

\[ = \frac{P(E_1)}{M} \sum_{i \leq F} \mathbb{E} \left[ M_i \bar{g} \left( \frac{n_i}{M_i} \right) \left| E_1 \right. \right] + O \left( \frac{L}{F} \right) \]

\[ = \frac{P(E_1)}{M} \mathbb{E} \left[ \sum_{i \leq F} M_i \bar{g} \left( \frac{n_i}{M_i} \right) \left| E_1 \right. \right] + O \left( \frac{L}{F} \right) \]

\[ = \frac{P(E_1)}{M} \mathbb{E} \left[ \sum_{i=1}^{M} M_i \bar{g} \left( \frac{n_i}{M_i} \right) \left| E_1 \right. \right] + O \left( \frac{L}{F} \right) \]

\[ = \frac{1}{M} \sum_{i \leq F} P(E_i) \mathbb{E} \left[ M_i \bar{g} \left( \frac{n_i}{M_i} \right) \left| E_i \right. \right] + O \left( \frac{L}{F} \right) \]

\[ = \mathbb{E}_{Y_i \sim f_1(x)} \mathbb{E} \left[ \bar{g} \left( \frac{n_i}{M_i} \right) Y_i \right] + O \left( \frac{L}{F} \right), \quad (25) \]

where in (22) we have used the fact that conditioned on \( E_i \), \( N_i \) and \( M_i \) are independent of \( E_j \) for \( i \neq j \). In (23) since there is no collision in \( H_2 \), \( M_i' \) and \( N_i' \) are equal to \( M_j \) and \( N_j \) for some \( i \) and \( j \). Equation (24) is because the values \( M_i' \) and \( N_i' \) are independent of the hash function \( H_2 \) and its outputs, and finally in equation (25), we used the fact that each set \( N_i' \) and \( M_i' \) are i.i.d random variables.

At this point, assuming that the variance of \( \frac{N_i'}{M_i'} \) is upper bounded by \( O(1/N) \) and using (Lemma 3.2 in [9]), we only need to derive \( \mathbb{E} \left[ \frac{N_i'}{M_i'} \right] \), and then we can simply find the RHS in (25). Note that \( N_i' \) and \( M_i' \) are independent and have binomial distributions with the respective means of \( N \pi_i X \) and \( M \pi_i Y \), where \( P_i X \) and \( P_i Y \) are the probabilities of mapping \( X \) and \( Y \) points with the respective densities \( f_0 \) and \( f_1 \) into bin \( i \). Hence,

\[ \mathbb{E} \left[ \frac{N_i'}{M_i'} \right] Y_i = \mathbb{E} [N_i'|Y_i] \mathbb{E} \left[ M_i'^{-1} Y_i \right]. \quad (26) \]

Let \( B_i \) denote the area for which all the points map to the same vector \( V_i \). \( \mathbb{E} [N_i'] \) can be written as:

\[ \mathbb{E} [N_i'] = N \int_{x \in B_i} f_1(x) dx = N \int_{x \in B_i} f_1(Y_i) + O(||x - Y_i||^\gamma) dx \]

\[ = N \epsilon d f_1(Y_i) + N \int_{x \in B_i} O(||x - Y_i||^\gamma) dx \]

\[ = N \epsilon d f_1(Y_i) + N \int_{x \in B_i + Y_i} O(||x||^\gamma) dx, \quad (27) \]

where in the second equality we have used the definition in (2). Let define \( B_i' := \frac{1}{\epsilon} B_i + \frac{1}{\gamma} Y_i \) and

\[ C_\gamma(Y_i) := \int_{x \in B_i'} ||x||^\gamma dx. \quad (28) \]

Note that \( C_\gamma(Y_i) \) is a constant independent of \( \epsilon \), since the volume of \( B_i' \) is independent of \( \epsilon \). By defining \( x' = x/\epsilon \) we can write

\[ \int_{x' \in B_i'} ||x||^\gamma dx = \int_{x' \in B_i'} \epsilon^\gamma ||x||^\gamma (\epsilon^d dx') = C_\gamma(Y_i) \epsilon^{\gamma + d} \quad (29) \]

Also note that since the number of \( X \) and \( Y \) points in each bin are independent we have \( \mathbb{E} [N_i'|Y_i] = \mathbb{E} [N_i'], \) and therefore

\[ \mathbb{E} [N_i'|Y_i] = N \epsilon d f_1(Y_i) + O(N \epsilon^{\gamma + d} C_\gamma(Y_i)). \quad (30) \]

Next, note that \( \mathbb{E} [M_i'|Y_i] \) has a non-zero binomial distribution, for which the first order inverse moment can be written as (14):
We use (Lemma 3.2 in [9]) and Remark 3, and obtain

\[ E \left[ M_i^{-1} | Y_i \right] = \left[ Me^d f_2(Y_i) + O(Me^{\gamma+d} C(Y_i)) \right]^{-1} \times \left( 1 + O\left( \frac{1}{Me^d f_2(Y_i)} \right) \right) \]

\[ = (Me^d f_2(Y_i))^{-1} \left[ 1 + O(\epsilon^\gamma) + O\left( \frac{1}{Me^d} \right) \right] \]

Thus, (26) can be simplified as

\[ E \left[ \frac{N'}{M'} | Y_i \right] = \frac{f_1(Y_i)}{\eta f_2(Y_i)} + O(\epsilon^\gamma) + O\left( \frac{1}{Me^d} \right). \]  

(32)

We use (Lemma 3.2 in [9]) and Remark 3 and obtain

\[ E \left[ g \left( \frac{\eta N'}{M'} \right) | Y_i \right] = g \left( \frac{f_1(Y_i)}{f_2(Y_i)} \right) + O(\epsilon^\gamma) \]

\[ + O\left( \frac{1}{Me^d} \right) + O(N^{-\frac{1}{2}}). \]  

(33)

Finally from (25) we get

\[ E \left[ \tilde{D}_g(X,Y) \right] = O(\epsilon^\gamma) + O\left( \frac{1}{Me^d} \right) + O(N^{-\frac{1}{2}}) + O\left( \frac{L}{F} \right) \]

\[ = O(\epsilon^\gamma) + O\left( \frac{1}{Me^d} \right), \]  

(34)

where in the second equation we have used the upper bound on \( L \) in Lemma 4.1 and the fact that \( M/N = \eta \).

Finally note that we can use a similar method with the same steps to prove the convergence of an estimator for Rényi divergence.

5 Discussion and Experiments

In this section we compare and contrast the advantages of the proposed estimator with competing estimators, and provide numerical results. These show the efficiency of our estimator in terms of MSE rate and computational complexity.

Table 1 summarizes the differences between the proposed optimum estimator (EHB) with other competing estimators: Ensemble NNR [9], Ensemble KDE [11] and Mirror KDE [12]. In terms of MSE rate, all of these estimators can achieve the optimal parametric MSE rate of \( O(1/N) \). In terms of computational complexity, our estimator has the best runtime compared to others. The smoothness parameter required for the optimum MSE rate is stated in terms of number of required derivatives of the density functions. The proposed estimator is the first divergence estimator that requires no extra smoothness at the boundaries. It is also the first divergence estimator that is directly applicable to online settings, retaining both the accuracy and linear total runtime. Finally, similar to NNR and Ensemble KDE estimators, the proposed estimator does not require any prior knowledge of the support of the densities.

We next compare the empirical performance of EHB to NNR, and the Ensemble KDE estimators. The experiments are done for two different types of f-divergence; KL-divergence and \( \alpha \)-divergence defined in [16]. Assume that \( X \) and \( Y \) are i.i.d. samples from independent truncated Gaussian densities. Figure 1 shows the MSE estimation rate of \( \alpha \)-divergence with \( \alpha = 0.5 \) of two Gaussian densities with the respective expectations of \([0,0]\) and \([0,1]\), and equal variances of \( \sigma_1^2 = \sigma_2^2 = I_2 \), versus different number of samples.

Figure 2 shows the MSE rates of the three \( \alpha \)-divergence estimators compared in dimension \( d = 4 \), \( \alpha = 2 \), for...
Table 1: Comparison of proposed estimator to Ensemble NNR [9], Ensemble KDE [11] and Mirror KDE [15]

| Estimator          | HB       | NNR      | Ensemble KDE | Mirror KDE |
|--------------------|----------|----------|--------------|------------|
| MSE Rate           | $O(1/N)$ | $O(1/N)$ | $O(1/N)$     | $O(1/N)$   |
| Computational Complexity | $O(N)$   | $O(kN \log N)$ | $O(N^2)$     | $O(N^2)$   |
| Required Smoothness ($\gamma$) | $d/2$    | $d$      | $(d + 1)/2$  | $d/2$      |
| Extra Smooth Boundaries | No       | Yes      | Yes          | Yes        |
| Online Estimation  | Yes      | No       | No           | No         |
| Knowledge about Boundary | No       | No       | No           | Yes        |

Figure 2: Runtime comparison of $\alpha$-divergence with $\alpha = 0.5$ between two independent truncated 2D Gaussian densities with the respective expectations of $[0,0]$ and $[0,1]$, and equal variances of $\sigma_1^2 = \sigma_2^2 = I_2$, versus different number of samples.

Figure 4: MSE estimation rate of $\alpha$-divergence with $\alpha = 2$ between two independent truncated Gaussian densities with dimension $d = 4$ and equal expectations $\mu_1 = \mu_2$ and covariance matrices $\sigma_1^2 = \sigma_2^2 = I_4$, versus different number of samples.

6 Conclusion

In this paper we proposed a fast hash based estimation method for $f$-divergence. We obtained bias and variance convergence rates, and validated our results by numerical experiments. Extending the method to hash-based mutual information estimation is a worthwhile topic for future work.

References

[1] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.

[2] K. R. Moon and A. O. Hero, “Ensemble estimation of multivariate $f$-divergence,” in *Information Theory (ISIT), 2014 IEEE International Symposium on*, pp. 356–360, IEEE, 2014.

[3] K. R. Moon, M. Noshad, S. Y. Sekeh, and A. O. Hero III, “Information theoretic structure learning with confidence,” in *Proc IEEE Int Conf Acoust Speech Signal Process*, 2017.
[4] A. Rényi, “On measures of entropy and information,” in Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1, pp. 547–561, University of California Press, 1961.

[5] S. M. Ali and S. D. Silvey, “A general class of coefficients of divergence of one distribution from another,” Journal of the Royal Statistical Society. Series B (Methodological), pp. 131–142, 1966.

[6] B. Póczos and J. G. Schneider, “On the estimation of alpha-divergences,” in AISTATS, pp. 609–617, 2011.

[7] S. Singh and B. Póczos, “Exponential concentration of a density functional estimator,” in Advances in Neural Information Processing Systems, pp. 3032–3040, 2014.

[8] Q. Wang, S. R. Kulkarni, and S. Verdú, “Divergence estimation for multidimensional densities via-nearest-neighbor distances,” IEEE Transactions on Information Theory, vol. 55, no. 5, pp. 2392–2405, 2009.

[9] M. Noshad, K. R. Moon, S. Y. Sekeh, and A. O. Hero III, “Direct estimation of information divergence using nearest neighbor ratios,” arXiv preprint arXiv:1702.05222, 2017.

[10] K. Kandasamy, A. Krishnamurthy, B. Poczos, L. Wasserman, et al., “Nonparametric Von Mises estimators for entropies, divergences and mutual informations,” in NIPS, pp. 397–405, 2015.

[11] K. R. Moon, K. Sricharan, K. Greenewald, and A. O. Hero, “Improving convergence of divergence functional ensemble estimators,” in IEEE International Symposium Inf Theory, pp. 1133–1137, IEEE, 2016.

[12] Y.-m. Zhang, K. Huang, G. Geng, and C.-l. Liu, “Fast kNN graph construction with locality sensitive hashing,” in Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 660–674, Springer, 2013.

[13] Q. Lv, W. Josephson, Z. Wang, M. Charikar, and K. Li, “Multi-probe LSH: efficient indexing for high-dimensional similarity search,” in Proceedings of the 33rd international conference on Very large data bases, pp. 950–961, VLDB Endowment, 2007.

[14] M. Znidaric, “Asymptotic expansion for inverse moments of binomial and Poisson distributions,” arXiv preprint math/0511226, 2005.

[15] S. Singh and B. Póczos, “Generalized exponential concentration inequality for Renyi divergence estimation,” in ICML, pp. 333–341, 2014.

[16] A. Cichocki, H. Lee, Y.-D. Kim, and S. Choi, “Non-negative matrix factorization with alpha-divergence,” Pattern Recognition Letters, vol. 29, no. 9, pp. 1433–1440, 2008.
A. Variance Proof

Proof of Theorem 2.2: The proof is based on Efron-Stein inequality. We follow similar steps used to prove the variance of NNR estimator in [9]. Note that the proof for variance of \( \hat{\rho}_i = N_i/(M_i) \) is contained in the the variance proof for \( \hat{D}_g(X,Y) \). Assume that we have two sets of nodes \( X_i, 1 \leq i \leq N \) and \( Y_j, 1 \leq j \leq M \). Here for simplicity we assume that \( N = M \), however, the extension of the proof to the case when \( M \) and \( N \) are not equal, is straightforward, by considering a number of virtual points, as considered in [9]. Define \( Z_i := (X_i, Y_i) \). For using the Efron-Stein inequality on \( Z := (Z_1, ..., Z_N) \), we consider another independent copy of \( Z \) as \( Z' := (Z'_1, ..., Z'_N) \) and define \( Z^{(i)} := (Z_1, ..., Z_{i-1}, Z'_i, Z_{i+1}, ..., Z_N) \). Define \( \hat{D}_g(Z) := \hat{D}_g(X,Y) \). By applying Efron-Stein inequality we have

\[
\mathbb{V}[\hat{D}_g(Z)] \leq \frac{1}{2} \sum_{i=1}^{N} \mathbb{E} \left( (\hat{D}_g(Z) - \hat{D}_g(Z^{(i)}))^2 \right)
= \frac{N}{2} \mathbb{E} \left( (\hat{D}_g(Z) - \hat{D}_g(Z^{(i)}))^2 \right)
\leq \frac{N}{2} \mathbb{E} \left( \left( \frac{1}{N} \sum_{i \in F, M_i > 0} M_i \hat{g} \left( \frac{\eta N_i}{M_i} \right) - \frac{1}{N} \sum_{i \in F, M_i > 0} M_i^{(1)} \hat{g} \left( \frac{\eta N_i^{(1)}}{M_i^{(1)}} \right) \right)^2 \right)
= \frac{1}{2N} \mathbb{E} \left( \sum_{i \in F, M_i > 0} \left( M_i \hat{g} \left( \frac{\eta N_i}{M_i} \right) - M_i^{(1)} \hat{g} \left( \frac{\eta N_i^{(1)}}{M_i^{(1)}} \right) \right)^2 \right)
= \frac{1}{2N} O(1) = O\left( \frac{1}{N} \right).
\]

(35)

where in the last line we used the fact that \( M_i \) and \( M_i^{(1)} \) can be different just for two of \( i \leq F \), and that difference is just \( O(1) \). So, the proof is complete.

B. Proof of Theorem 2.3

Assume that the densities have bounded derivatives up to the order \( q \). Then the Taylor expansion of \( f(y) \) around \( f(x) \) is as follows

\[
f(y) = f(x) + \sum_{|i| \leq q} \frac{D^i f(x)}{i!} \|y - x\|^i + O(\|y - x\|^q).
\]

(36)

Therefore, similar to [27] and using [29] we can write

\[
\mathbb{E} \left[ N_i' \right] = N \int_{x \in B_i} f_1(x) dx
= N \int_{x \in B_i} f(Y_i) + \sum_{|j| \leq q} \frac{D^j f(Y_i)}{j!} \|x - Y_i\|^j + O(\|x - Y_i\|^q) dx
= N e^d f_1(Y_i) + N \sum_{|j| \leq q} \frac{D^j f(Y_i)}{j!} C_j(Y_i) e^{j|d|} + O\left( N C_q(Y_i) e^{q|d|} \right)
= N e^d \left[ f_1(Y_i) + \sum_{i=1}^{q} C'_i(Y_i) e^i + O\left( C_q(Y_i) e^{q|d|} \right) \right],
\]

(37)

where

\[
C'_j(Y_i) := \sum_{|j| \leq q} \frac{D^j f(Y_i)}{j!} C_j(Y_i).
\]
Similarly we obtain
\begin{equation}
\mathbb{E} \left[ (M_i')^{-1} \right] = M^{-1} e^{-d} \left[ f_2(Y_i) + \sum_{i=1}^{q} C_i'(Y_i)e^i + O(C_q(Y_i)e^q) \right]^{-1} \left( 1 + O\left( \frac{1}{Me^d f_2(Y_i)} \right) \right).
\end{equation}

(38)

The rest of the proof follows by using the same steps as used in equations (32)-(34), and we get
\begin{equation}
\mathbb{E} \left[ \hat{D}_g(X,Y) \right] = \sum_{i=1}^{q} C''_i e^i + O\left( \frac{1}{Ne^d} \right).
\end{equation}

(39)

where $C''_1, ..., C''_2$ are constants.

Now are ready to apply the ensemble theorem (11, Theorem 4). Let $T := \{t_1, ..., t_T\}$ be a set of index values with $t_i < c$, where $c > 0$ is a constant. Let $\epsilon(t) := \left\lceil tN^{-1/2d} \right\rceil$. The proof completes by using the ensemble theorem in (11, Theorem 4) with the parameters $\psi_i(t) = t_i/d$ and $\phi'_i,d(N) = \phi_{i,k}(N)/Ni^d$. So the following weighted ensemble has the MSE convergence rate of $O(1/N)$:
\begin{equation}
\hat{D}_w := \sum_{t \in T} w(t)\hat{D}_{\epsilon(t)}.
\end{equation}

(40)

C. Proof of Theorem 3.1

We first argue that amortized runtime complexity of the online estimation algorithm is order $O(1)$ for each update after adding new samples. Note that when we add a new pair of samples $X_{N+1}$ and $Y_{N+1}$, if $N + 1 \neq 2^k$ for some integer $k$, we only find $H(X_{N+1})$ and $H(Y_{N+1})$ and update the corresponding $M_i$ and $N_i$, which take a constant time. But, only when $N + 1 = 2^k$ for some integer $k$, we need $O(N)$ time complexity to update $\epsilon$ and therefore the hash function. Thus, if we have $N = 2^k$ nodes added to the estimation algorithm, the total complexity due to rehashing, $T_H$, is as follows:
\begin{equation}
T_H = 1 + 2 + 2^2 + ... + 2^k = 2^{k+1} - 1 = 2N - 1.
\end{equation}

(41)

So, the amortized runtime complexity per each time step is $O(\frac{2N-1}{N}) = O(1)$. So overall, the amortized computational complexity is order $O(1)$. Finally, note that since we update $\epsilon$ when $N$ doubles, it is at most by a factor of 2 away from the optimum $\epsilon$. Since constant factor doesn’t affect the asymptotic order of the bias error, the bias bound always holds for online estimation algorithm.