Enhancing Nearest Neighbor Based Entropy Estimator for High Dimensional Distributions via Bootstrapping Local Ellipsoid

Chien Lu, Jaakko Peltonen
Tampere University
Kalevantie 4
33100 Tampere, Finland
{chien.lu, jaakko.peltonen}@tuni.fi

Abstract

An ellipsoid-based, improved kNN entropy estimator based on random samples of distribution for high dimensionality is developed. We argue that the inaccuracy of the classical kNN estimator in high dimensional spaces results from the local uniformity assumption and the proposed method mitigates the local uniformity assumption by two crucial extensions, a local ellipsoid-based volume correction and a correction acceptance testing procedure. Relevant theoretical contributions are provided and several experiments from simple to complicated cases have shown that the proposed estimator can effectively reduce the bias especially in high dimensionalities, outperforming current state of the art alternative estimators.

Introduction

The differential entropy of a continuous-valued random variable $\boldsymbol{X}$ is defined as

$$H(\boldsymbol{X}) = -\int p(\boldsymbol{x}) \log p(\boldsymbol{x}) d\boldsymbol{x}. \tag{1}$$

where $p(\boldsymbol{x})$ is the probability density function of $\boldsymbol{X}$. Entropy has been an important numerical quantity in Statistics, Machine Learning and other disciplines such as Physics. It provides a summary measurement of the degree of uncertainty of a system. Entropy is also related to other important information theoretic measures, including Kullback-Leibler divergence (Kullback and Leibler 1951) and mutual information (Cover and Thomas 2006) , which is defined for random variables $\boldsymbol{X}$ and $\boldsymbol{Y}$ as

$$I(\boldsymbol{X}, \boldsymbol{Y}) = \int_Y \int_X \log \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} d\boldsymbol{x} d\boldsymbol{y} = -H(\boldsymbol{Y}|\boldsymbol{X}) + H(\boldsymbol{Y}) \ . \tag{2}$$

Classical kNN Estimator

In theory, to obtain the entropy of a system, the underlying probability distribution must be known, that is, the analytical form of the probability density function (PDF) needs to be available. However, in real-world cases, it is common that the underlying PDF is not available but only a set of samples are observed. This raises the research problem of estimating the entropy from the observed data only; in particular, flexible estimation approaches that do not assume the distribution to lie in a particular parametric model family are known as non-parametric entropy estimation. Estimators such as the k-nearest neighbor estimator (kNN; Kozachenko and Leonenko 1987 and Goria et al. 2005) and the kernel density estimator (KDE; Silverman 2018) or hybrid methods (Orava 2011) have been proposed. Note that there is another important group of approaches, called ensemble estimators (Sricharan, Wei, and Hero 2013; Moon, Sricharan, and Hero 2017; Moon et al. 2018), which use weighted combinations of different estimators. This work focuses on the kNN estimator and approaches related to it.

Assume there are N i.i.d. D-dimensional samples $\boldsymbol{x}_1, ..., \boldsymbol{x}_N \sim P$, where the probability density function p is unknown. The classical kNN estimator is written as

$$\begin{aligned} H(\boldsymbol{X}) &\approx -\frac{1}{N} \sum_{i=1}^{N} \log p(\boldsymbol{x}_i) \\ &\approx \psi(N) - \psi(k) + \log(c_D) + \frac{D}{N} \sum_{i=1}^{N} \log \varepsilon_i \end{aligned} \tag{3}$$

where ψ denotes the digamma function, ε_i denotes the Euclidean distance from $\mathbf{x}_i$ to its nearest neighbor, $c_D = \frac{\pi^{\frac{D}{2}}}{\Gamma(1+\frac{D}{2})}$, and $\psi(N) - \psi(k)$ is the correction term.

The classical estimator has been widely applied in many different research problems. It has been also adopted to non-parametrically estimate the mutual information (Kraskov, Stögbauer, and Grassberger 2004) of Equation (2) and KL divergence (Pérez-Cruz 2008; Wang, Kulkarni, and Verdú 2009).

Bias of the classical kNN

Although the classical kNN estimator has shown promising applicability, it has been found biased especially in higher dimensional cases (Noh et al. 2014; Chauveau and Vandekerkhove 2014). Theoretical bounds relating the convergence to the data dimension include (Gao, Oh, and Viswanath

2018; Singh and Póczos 2016b; 2014; 2016a), for example (Gao, Oh, and Viswanath 2018) showed the the bias of the classical kNN estimator is $\tilde{O}(N^{-1/D})$, where $\tilde{O}$ denotes limiting behavior up to polylogarithmic factors in N.

One explanation of the bias can be that the local uniform assumption of the classical kNN cannot always hold (Lombardi and Pant 2016; Gao, Ver Steeg, and Galstyan 2015; Gao, Steeg, and Galstyan 2015; Gao, Oh, and Viswanath 2016; Lord, Sun, and Bollt 2018) especially when the dimension is high. In other words, the hyperspherical structure (ε-ball) is not capable of capturing the twisted shape of the underlying distribution of the random variable around the sample; such a situation can have increasingly strong effect on the bias when dimensionality becomes higher.

The rest of the paper is organized as follows. In Section we describe our proposed method and bias analysis of the proposed method including the corresponding correction term and the bias bound. In Section we review related work based on local approximations. In Section we describe comparison experiments both in estimation of entropy and estimation of mutual information. Lastly, Section provides conclusions.

Method

We now introduce our novel entropy estimation method, called the kNN estimator with Ellipsoidal correction (EC-kNN); the resulting method will be directly applicable both to entropy estimation and to mutual information estimation. The idea is to construct a local ellipsoid instead of a ε-ball, so that inside of the local ellipsoid samples can be assumed more uniformly distributed and hence will better fit the uniformity assumption of kNN based entropy estimation. The proposed method comprises two parts: the first part is the local Ellipsoidal correction where a local ellipsoid is learned via performing a local principal component analysis (PCA) algorithm, the second part is an acceptance testing procedure which is performed in a boot-strapping manner. Note that the local-PCA approach has been adopted by other similar works but the bootstrap step is an significant novelty in our approach.

The classical kNN estimator of Equation (3) can be rewritten as

$$H(\boldsymbol{X}) \approx \frac{1}{N} H_{kNN}(\boldsymbol{x_i}) \tag{4}$$

where $H_{kNN}(\boldsymbol{x_i})$ is an approximation of $-\log p(\boldsymbol{x_i})$ based on a D-dimensional ball of radius ε_i encompassing k neighbors, where

$$H_{kNN}(\boldsymbol{x_i}) = \psi(N) - \psi(k) + \log(V_i) \tag{5}$$

and the $\log(V_i)$ term denotes the logarithm of the volume V_i of the ball, computed as:

$$\log(V_i) = \log(c_D) + D \log \varepsilon_i \tag{6}$$

where $c_D = \frac{\pi^{\frac{D}{2}}}{\Gamma(1+\frac{D}{2})}$ as before.

Our aim is to generalize the above by a logarithmic volume correction term $\log(\Delta \tilde{V}_i)$, corresponding to the difference between logarithmic volume of a local ellipsoid versus the epsilon-ball. Furthermore, as a result of the local ellipsoid the number of neighbors k may slightly change around each point $\boldsymbol{x_i}$, yielding an individual k_i for each point.

Theorem 1. *The correction term of the ellipsoid-based estimator is $\psi(N) - \psi(k)$, which is the same as the classical estimator.*

Proof. Let $\mathcal{E}(\mathbf{x}_i, \mathbf{r}(\mathbf{x}_i))$ denote the ellipsoid around $\mathbf{x}_i$ defined by axes $\mathbf{r}(\mathbf{x}_i) = [r_1(\mathbf{x}_i), r_2(\mathbf{x}_i), \ldots, r_D(\mathbf{x}_i)]$ where $r_1(\mathbf{x}_i)$ and $r_D(\mathbf{x}_i)$ are the longest and shortest axes respectively. For any choice of $\mathbf{r}(\mathbf{x}_i)$ the $\mathcal{E}(\mathbf{x}_i, \mathbf{r}(\mathbf{x}_i))$ can also be defined by introducing a Mahalanobis matrix $\mathcal{M}_{\mathbf{x}_i}$ such that $\forall \boldsymbol{\xi} \in \mathcal{E}(\mathbf{x}_i, \mathbf{r}(\mathbf{x}_i))$

$$\sqrt{(\boldsymbol{\xi} - \mathbf{x}_i)^\top \mathcal{M}_{\mathbf{x}_i} (\boldsymbol{\xi} - \mathbf{x}_i)} \leq r_1(\mathbf{x}_i). \tag{7}$$

In this definition the matrix controls the shape of the ellipsoid and $r_1(\mathbf{x}_i)$ controls its overall size.

Let $p(r_1(\mathbf{x}_i)$ denote the probability mass inside the Ellipsoid controlled by $r_1(\mathbf{x}_i)$ and $\mathcal{M}_{\mathbf{x}_i}$; for brevity the notation $p(r_1(\mathbf{x}_i)$ shows the dependence on $r_1(\mathbf{x}_i)$ only. The probability mass is

$$\begin{aligned} p(r_1(\mathbf{x}_i)) &= \int_{\boldsymbol{\xi} \in \mathcal{E}(\mathbf{x}, \mathbf{r}(\mathbf{x}))} p(\boldsymbol{\xi}) d\boldsymbol{\xi} \\ &= \int_{\sqrt{(\xi - \mathbf{x})^\top \mathcal{M}_{\mathbf{x}_i} (\xi - \mathbf{x})} \leq r_1(\mathbf{x}_i)} p(\boldsymbol{\xi}) d\boldsymbol{\xi} \end{aligned} \tag{8}$$

Similar to the work of (Kraskov, Stögbauer, and Grassberger 2004), let $\mu(r_1(\mathbf{x}_i))$ be the probability density of $r_1(\mathbf{x}_i)$, $\mu(r_1(\mathbf{x}_i)) dr_1(\mathbf{x}_i)$ is the probability that the Mahalanobis distance of the kth nearest neighbor and the $\mathbf{x}_i$ is in the interval $[r_1(\mathbf{x}_i), r_1(\mathbf{x}_i) + dr_1(\mathbf{x}_i)]$. It can be written as

$$\begin{aligned} \mu\left(r_1\left(\mathbf{x}_i\right)\right) dr_1(\mathbf{x}_i) = & \frac{(N-1)!}{1!(k-1)!(N-k-1)!} \\ & \times \frac{dp(r_1(\mathbf{x}_i))}{dr_1(\mathbf{x}_i)} dr_1(\mathbf{x}_i) \\ & \times p(r_1(\mathbf{x}_i))^{k-1} \times \left(1 - p(r_1(\mathbf{x}_i))\right)^{N-k-1} \end{aligned} \tag{9}$$

by using the trinomial formula. The correction term can be obtained via taking the expectation of $-\log p((r_1(\mathbf{x}_i))$

$$\begin{aligned} E\left[-\log p(r_1(\mathbf{x}_i))\right] &= \int_0^\infty \mu(r_1(\mathbf{x}_i)) dr_1(\mathbf{x}_i) \left(-\log p(r_1(\mathbf{x}_i))\right) \\ &= \psi(N) - \psi(k). \end{aligned} \tag{10}$$

□

Under the assumption that inside of $\mathcal{E}(\mathbf{x}_i, \mathbf{r}(\mathbf{x}_i))$, the probability density is $p(\mathbf{x}_i)$ a constant such that $p(\mathbf{x}_i) \times V_{\mathcal{E}_i} \approx p(r_1(\mathbf{x}_i))$ and $\log V_{\mathcal{E}_i} = \log(V_i) + \log(\Delta \tilde{V}_i)$,

$$\begin{aligned} E\left[-\log p(\mathbf{x}_i)\right] &\approx \psi(N) - \psi(k) + \log V_{\mathcal{E}_i} \\ &= \psi(N) - \psi(k) + \log(V_i) + \log(\Delta \tilde{V}_i) \end{aligned} \tag{11}$$

, the resulting entropy estimator EC-kNN will be of the form

$$H(\boldsymbol{X}) \approx \frac{1}{N} H_{EC-kNN}(\mathbf{x}_i) \tag{12}$$

where

$$H_{EC-kNN}(\mathbf{x}_i) = \psi(N) - \psi(k_i) + \log(V_i) + \log(\Delta\tilde{V}_i) \tag{13}$$

and $\log(V_i)$ is defined by Equation (6). However, since the local ellipsoid is learned from a small amount of local data points, we must guard against potential over-correction to the shape learned from this limited amount of data; we do this by an acceptance testing procedure, and apply the logarithmic correction term $\log(\Delta\tilde{V}_i)$ only if the test is successful.

Next, in Section we describe the local ellipsoidal correction for computing the correction term $\Delta\tilde{V}_i$ and in Section we describe the bootstrap based acceptance testing.

Local Ellipsoidal Correction

The ε-ball in the classical kNN entropy estimator treats all directions equally. In a high-dimensional space the underlying probability distribution around a data point $\boldsymbol{x_i}$ may be curved or stretched so that, due to the shape of the distribution, the density along some directions of the high-dimensional space may be larger than along other directions, and moreover, the density may experience larger changes along some directions than along others. In such a situation, it can still be a reasonable approximation to assume the density to be approximately uniform in a small hyper-region, but it is no longer a good assumption that the hyper-region would be an ε-ball. We instead aim to learn an ellipsoid as a better representation of a local hyper-region with approximately uniform density.

To learn a local ellipsoid structure, we propose to use a local PCA based approach as follows. In the local PCA approach, we first find the k nearest neighbors of $\mathbf{x}_i$ by smallest Euclidean distance as usual. We then compute the sample covariance matrix of the resulting neighborhood of $\boldsymbol{x_i}$ (including $\boldsymbol{x_i}$ itself) and rotate the neighborhood to a new coordinate system by projecting the data onto the eigenvectors of the obtained covariance matrix. The eigenvectors are used as the axes of the local ellipsoid. Next, the widths of the local ellipsoid along each axis are then computed via searching for the maximum distance from $\boldsymbol{x_i}$ to the neighbor points along each coordinate axis (see Figure 1).

After performing the local PCA, the sum of the log of ratios of the longest axis to other each of the axes of the estimated ellipsoid is then taken as the logarithmic volume correction term; additionally, the number of neighbors is recounted based on the ellipsoid.

In detail, since the volume of an ellipsoid with axes $r_1(\mathbf{x}_i), r_2(\mathbf{x}_i), ..., r_D(\mathbf{x}_i)$ is defined as

$$\frac{\pi^{\frac{D}{2}}}{\Gamma(1+\frac{D}{2})} \prod_{d=1}^{D} r_d(\mathbf{x}_i) \ , \tag{14}$$

we assume that the longest axis r_1, the distance from the origin to the farthest point alone the longest axis, represents the original radius, the correction term is obtained as

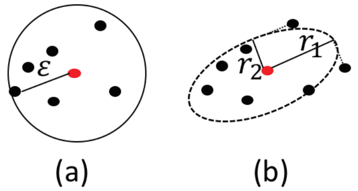

Figure 1: (a) The ε-ball of the classical kNN estimator. (b) Local ellipsoid learned via local PCA. Lines extending from the center denote local axes found by local PCA; the widths r_d of the lines denote distance from the central point to the furthest neighbor along each axis, and the width of the ellipsoid along the axis is defined based on that distance. Note that as a result, due to the curvature of the ellipsoid boundary, the furthest neighbor along an axis may not lie inside of the ellipsoid.

$$\Delta\hat{V}(\boldsymbol{x_i}, \boldsymbol{X}) = \prod_{d=1}^{D} \frac{r_d(\mathbf{x}_i)}{r_1(\mathbf{x}_i)} \ . \tag{15}$$

Algorithm 1 shows the computation of the correction term. In the complete entropy estimation algorithm described in the next section, we denote the correction term as $\Delta\tilde{V}_i$.

Algorithm 1 Local ellipsoid-based volume correction

```
Require: $\boldsymbol{x_i}$: sample point
         $\boldsymbol{X} = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_N}\}$: all sample points
         $D$: dimention
         $k$: number of neighbors
Ensure: $\Delta\hat{V}(\boldsymbol{x_i}, \boldsymbol{X})$: local ellipsoid-based volume correction
Find the $k$-th neighbor of the $\boldsymbol{x_i}$, compute the distance $\varepsilon_i$
Perform a PCA on the set of the $k+1$ points
Project the $k+1$ points to the new coordinate system, get the new center via averaging all the projected points
Find the lengths $r_1, ..., r_d$ of the axes of the projected ellipsoid through computing the maximum difference of the point from the center to the center along the PCA projection axis
    return $\Delta\hat{V}(\boldsymbol{x_i}, \boldsymbol{X}) = \prod_{d=1}^{D} \frac{r_d}{r_1}$
```

Bootstrap Testing for Correction Acceptance

A nonuniform empirical distribution of a finite set of data samples along different coordinate axes can also happen due to random sampling variation. That is, a nonuniform distribution of the data subset in an ε-ball around a data point can be observed even under the uniformity assumption of the underlying distribution. Therefore, it can happen that the volume corrections from some points are not necessary and an over-correction problem can occur if the correction is performed for every sample point.

Hence, we introduce a bootstrap style acceptance testing procedure to amend the potential over-correction issue. For each sample point, a reference correction $\Delta\hat{V}_u$ is generated, representing a rough estimate of the correction that can occur due to random sampling alone, under a uniformity assumption of the underlying density in the ε-ball around the sample point. The reference correction is then used in the acceptance testing procedure as an acceptance threshold.

The reference correction $\Delta\hat{V}_u$ is simply a volume correction obtained from a set of $k+1$ samples generated inside of a uniformly distributed ε-ball around the $\boldsymbol{x}_i$. The details of generating the $\Delta\hat{V}_u$ are shown in Algorithm 2.

Since the data point $\boldsymbol{x}_i$ has itself been randomly generated from an underlying distribution, the center point of the ε-ball is itself a random variable. Therefore, in order to simulate a true random configuration, the randomness of whether the $\boldsymbol{x}_i$ is the center of the ε-ball should also be taken into account. Therefore, after simulating the k + 1 points inside of the ε-ball, the center of the ball is not necessarily the original $\boldsymbol{x}_i$ but is instead redefined by choosing the point among the $k+1$ points which is closest to the original point.

The proposed kNN estimator with ellipsoidal correction (EC-kNN) is then developed. It combines the above-proposed Algorithm 1 and Algorithm 2. For each sample point, the volume correction is performed with a bootstrap acceptance test. The correction term $\Delta\tilde{V}_i$ and the referenced correction term ΔV_u are generated respectively. Then the bootstrap acceptance test is conducted via comparing the values of the two $\Delta\tilde{V}_i$ and ΔV_u. The correction is accepted if $\Delta\tilde{V}_i < \Delta V_u$, otherwise the algorithm uses the result of the classical kNN estimator.

Note that, in high dimensional cases, it can happen that when the number of neighbors inside the new ellipsoid are counted, it turns out that there are no points of the finite data set inside of the ellipsoid. When encountering this issue, the axes of the ellipsoid are increased slightly until there is at least one data point inside of the ellipsoid. Here, in the proposed algorithm, every one of the axes is lengthened by multiplying the r_d by a small ratio (e.g., 1.01). The volume correction is changed accordingly. By definition, $\boldsymbol{x}_i = [x_{i,1}, ..., x_{i,D}]^\top$ is inside of the ellipsoid if

$$\sum_{d=1}^{D} \frac{(x_{i,d} - c_d)^2}{r_d^2} \leq 1 \qquad (16)$$

where $\boldsymbol{c} = [c_1, ..., c_D]^\top$ is the origin of the ellipsoid.

After going through every data point in the data set, the algorithm produces the final corrected result by averaging the corrected entropy values from each data point from the data set. Note that, there is not hyper-parameter setting required and since the bootstrap acceptance test procedure is a result of a randomly generated value ΔV_u, therefore, the result of the computation can be slightly different every time.

The final algorithm including the local correction computation, reference correction computation, bootstrap testing, enlargement of ellipsoids, and computation of the entropy estimate is summarized as Algorithm 3.

Algorithm 2 Reference correction

```
Require: $D$: dimension
         $k$: number of neighbors
         $\varepsilon$: radius
Ensure: $\hat{V}_u$: Acceptance variable
  Generate random samples $\boldsymbol{U} = \{\boldsymbol{u}_1, ..., \boldsymbol{u}_{k+1}\}$ from a
    $D$-dimensional uniformly distributed $\varepsilon$-ball
  Select the point $\boldsymbol{u}_j$ which is the closest to the origin
    among the $k+1$ random samples
      return $\hat{V}_u = \Delta\hat{V}_k(\boldsymbol{u_j}, \boldsymbol{U})$ using Algorithm 1
```

Bias analysis

Theorem 2. *The bias* $E_{\mathbf{X}\sim p}\left[|\log p(\mathbf{x}) - \log \hat{p}_{\mathbf{r}(\mathbf{x})}(\mathbf{x})|\right]$ *is bounded.*

Proof. By the mean value theorem (Lebesgue 1910) for any $0 < a < b$ we have $(\log(b) - \log(a))/(b-a) = 1/c$ for some c in the open interval (a, b) Applying this inside the expectation $E_{\mathbf{X}\sim p}\left[|\log p(\mathbf{x}) - \log \hat{p}(\mathbf{x})|\right]$ we get the bound

$$E_{\mathbf{X}\sim p}\left[|\log p(\mathbf{x}) - \log \hat{p}(\mathbf{x})|\right] \leq M^* E_{\mathbf{X}\sim p}\left[|p(\mathbf{x}) - \hat{p}(\mathbf{x})|\right] \qquad (18)$$

where $\hat{p}(\mathbf{x})$ is an arbitrary estimator of $p(\mathbf{x})$, and $M^* = \sup_{\mathbf{X}\sim p} \max(1/p(\mathbf{x}), 1/\hat{p}(\mathbf{x}))$ which is finite if p and $\hat{p}$ are both bounded above zero inside the support of p . We make use of the result that $E_{\mathbf{X}\sim p}\left[|p(\mathbf{x}) - \hat{p}_{\varepsilon(\mathbf{x})}(\mathbf{x})|\right]$ is bounded for the classical estimator (Singh and Póczos 2016b).

Consider the classical estimator $\hat{p}_{\varepsilon(\mathbf{x})}(\mathbf{x})$ which assumes the density is uniform inside the ball. In the asymptotic case of increasing data the estimate becomes the integral over density inside the ball,

$$\hat{p}_{\varepsilon(\mathbf{x})}(\mathbf{x}) = \frac{1}{V_{\mathcal{B}}} \int_{\boldsymbol{\xi}\in E(x,\varepsilon(\mathbf{x}))} p(\boldsymbol{\xi}) d\boldsymbol{\xi}$$

where, by an abuse of notation, $V_{\mathcal{B}}$ denotes the volume of the Ball. The bias of the estimator then becomes

$$E_{\mathbf{X}\sim p}\left[|p(\mathbf{x}) - \hat{p}_{\varepsilon(\mathbf{x})}(\mathbf{x})|\right] = E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{B}}} \int_{\boldsymbol{\xi}\in E(x,\varepsilon(\mathbf{x}))} p(\mathbf{x}) - p(\boldsymbol{\xi}) d\boldsymbol{\xi}\right|\right] \qquad (19)$$

Similarly, for the case of the proposed estimator, let $r(\mathbf{x})$ denote the axes of the ellipsoid $\mathcal{E}(\mathbf{x}, \mathbf{r}(\mathbf{x}))$ around $\mathbf{x}$ so that

$$E_{\mathbf{X}\sim p}\left[|p(\mathbf{x}) - \hat{p}_{\mathbf{r}(\mathbf{x})}(\mathbf{x})|\right] = E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{E}}} \int_{\xi\in E(x,\mathbf{r}(x))} p(x) - p(\xi) d\boldsymbol{\xi}\right|\right] \qquad (20)$$

Since by construction the ellipsoid is contained within the

Algorithm 3 EC-kNN

```
Require: $\boldsymbol{X} = \{\boldsymbol{x_1}, \dots, \boldsymbol{x_N}\}$: all sample points
         $D$: dimension
         $k$: number of neighbors
Ensure: $\hat{H_{\boldsymbol{X}}}$: corrected entropy estimation
for each $\boldsymbol{x}_i$ do
    Find the $k$ nearest neighbors, compute the distance $\varepsilon_i$ and the volume of the $\varepsilon_i$-ball $V_i$
    Compute the empirical volume correction
        $\Delta\tilde{V}_i = \Delta\hat{V}(\boldsymbol{x_i}, \boldsymbol{X})$    (17)
    using Algorithm 1
    Compute the reference volume $\Delta\hat{V}_u$ correction using Algorithm 2
    if $\Delta\tilde{V}_i < \Delta\hat{V}_u$ then
        Find the number of samples inside of the ellipsoid
        while No samples inside of the ellipsoid do
            Lengthen every axis by multiplying each $r_d$ with a small ratio (1.01), adjust the $\Delta\tilde{V}_i$ accordingly
        end while
        $H_{EC-kNN}(\boldsymbol{x}_i) = \psi(N) - \psi(k_i) + \log(V_i) + \log(\Delta\tilde{V}_i)$
    else
        $H_{EC-kNN}(\boldsymbol{x}_i) = \psi(N) - \psi(k_i) + \log(V_i)$
    end if
end for
return $H(\boldsymbol{X}) = \frac{1}{N}\sum_{i=1}^{N} H_{EC-kNN}(\boldsymbol{x}_i)$
```

corresponding ball, (20) becomes

$$
\begin{aligned}
&E_{\mathbf{X}\sim p}\left[\left|\frac{V_{\mathcal{B}}}{V_{\mathcal{E}}}\frac{1}{V_{\mathcal{B}}}\int_{\xi\in E(x,\mathbf{r}(x))} p(x) - p(\xi)d\boldsymbol{\xi}\right|\right] \\
&\quad\le C\cdot E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{B}}}\int_{\xi\in E(x,\mathbf{r}(x))} p(x) - p(\xi)d\boldsymbol{\xi}\right|\right] \\
&\quad\le C\cdot\Bigg(E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{B}}}\int_{\xi\in E(x,\mathbf{r}(x))} p(x) - p(\xi)d\boldsymbol{\xi}\right|\right] \\
&+ E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{B}}}\int_{\xi\in E(x,\varepsilon(\mathbf{x}))\setminus E(x,\mathbf{r}(x))} p(x) - p(\xi)d\boldsymbol{\xi}\right|\right]\Bigg) \\
&\quad= C\cdot E_{\mathbf{X}\sim p}\left[\left|p(\mathbf{x}) - \hat{p}_{\varepsilon(\mathbf{x})}(\mathbf{x})\right|\right] \qquad (21)
\end{aligned}
$$

where $C = \sup_{\mathbf{X}\sim p}\frac{V_{\mathcal{B}}}{V_{\mathcal{E}}}$ is bounded above by any suitable regularization keeping the ellipsoids at non-zero volume, the right-hand term is for the classical estimator which is know to be bounded, thus the ellipsoid case is bounded as well. □

Theorem 3. *Bound of* $E_{\mathbf{X}\sim p}\left[|\log p(\mathbf{x}) - \log\hat{p}_{\mathbf{r}(\mathbf{x})}(\mathbf{x})|\right]$ *is less or equal to bound of* $E_{\mathbf{X}\sim p}\left[|\log p(\mathbf{x}) - \log\hat{p}_{\varepsilon(\mathbf{x})}(\mathbf{x})|\right]$.

Proof. We continue from (18). Assume that the probability density p is second order differentiable. Then, with Taylor expansion the density bias term of the classical estimator can be approximated as

$$
\begin{aligned}
&E_{\mathbf{X}\sim p}\left[|p(\mathbf{x}) - \hat{p}_{\varepsilon(\mathbf{x})}(\mathbf{x})|\right] = \\
&\qquad E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{B}}}\int_{\boldsymbol{\xi}\in E(x,\varepsilon(\mathbf{x}))} p(\mathbf{x}) - p(\boldsymbol{\xi})d\boldsymbol{\xi}\right|\right] \\
&= E_{\mathbf{X}\sim p}\left[\left|\frac{1}{2V_{\mathcal{B}}}\int_{\boldsymbol{\xi}\in\mathcal{B}(\mathbf{x},\varepsilon(\mathbf{x}))}(\boldsymbol{\xi}-\mathbf{x})^{\top}H(\mathbf{x})(\boldsymbol{\xi}-\mathbf{x})d\boldsymbol{\xi}\right|\right] + \epsilon_{\mathcal{B}}
\end{aligned}
\tag{22}
$$

where, again by an abuse of notation, $V_{\mathcal{B}}$ denotes the volume of the ball and $\epsilon_{\mathcal{B}}$ denotes the remainder term of the Taylor series; we assume $\epsilon_{\mathcal{B}}$ is small enough to be neglected. Similarly, for the case of the proposed estimator, let $r(\mathbf{x})$ again denote the axes of the ellipsoid $\mathcal{E}(x, \mathbf{r}(x))$ around $\mathbf{x}$,

$$
\begin{aligned}
&E_{\mathbf{X}\sim p}\left[|p(\mathbf{x}) - \hat{p}_{\mathbf{r}(\mathbf{x})}(\mathbf{x})||\right] \\
&\qquad = E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{E}}}\int_{\boldsymbol{\xi}\in\mathcal{E}(\mathbf{x},\mathbf{r}(\mathbf{x}))} p(\mathbf{x}) - p(\boldsymbol{\xi})d\boldsymbol{\xi}\right|\right] \\
&\quad = E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{E}}}\int_{\boldsymbol{\xi}\in\mathcal{E}(\mathbf{x},\mathbf{r}(\mathbf{x}))} \triangledown p(\mathbf{x})^{\top}(\boldsymbol{\xi}-\mathbf{x})d\boldsymbol{\xi}\right|\right] \\
&+ E_{\mathbf{X}\sim p}\left[\left|\frac{1}{2V_{\mathcal{E}}}\int_{\boldsymbol{\xi}\in\mathcal{E}(\mathbf{x},\mathbf{r}(\mathbf{x}))}(\boldsymbol{\xi}-\mathbf{x})^{\top}H(\mathbf{x})(\boldsymbol{\xi}-\mathbf{x})d\boldsymbol{\xi}\right|\right] + \epsilon_{\mathcal{E}}
\end{aligned}
\tag{23}
$$

where $V_{\mathcal{E}}$ denotes the volume of the ellipsoid, and again, $\epsilon_{\mathcal{E}}$ is assumed ignorable. Due to the symmetry of the ellipsoid, $E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{E}}}\int_{\boldsymbol{\xi}\in\mathcal{E}(\mathbf{x},\mathbf{r}(\mathbf{x}))}\triangledown p(\mathbf{x})^{\top}(\boldsymbol{\xi}-\mathbf{x})d\boldsymbol{\xi}\right|\right] = 0$. Hence,

$$
\begin{aligned}
&E_{\mathbf{X}\sim p}\left[\left|p(\mathbf{x}) - \hat{p}_{\mathbf{r}(\mathbf{x})}(\mathbf{x})\right|\right] \\
&\qquad \approx E_{\mathbf{X}\sim p}\left[\left|\frac{1}{2V_{\mathcal{E}}}\int_{\boldsymbol{\xi}\in\mathcal{E}(\mathbf{x},\mathbf{r}(\mathbf{x}))}(\boldsymbol{\xi}-x)^{\top}H(\mathbf{x})(\boldsymbol{\xi}-\mathbf{x})d\boldsymbol{\xi}\right|\right] \\
&= E_{\mathbf{X}\sim p}\left[\left|\frac{1}{2V_{\mathcal{E}}}\int\cdots\int_{\sum_{d=1}^{D}\frac{(\xi_d-x_d)^2}{r_d(\mathbf{x})^2}\le 1}(\boldsymbol{\xi}-\mathbf{x})^{\top}H(\mathbf{x})(\boldsymbol{\xi}-\mathbf{x})d\boldsymbol{\xi}\right|\right] \\
&= E_{\mathbf{X}\sim p}\left[\left|\frac{\prod_d r_d(\mathbf{x})}{2V_{\mathcal{E}}}\int\cdots\int_{\mathbf{u}^{\top}\mathbf{u}\le 1}(\mathbf{r}(\mathbf{x})\odot\mathbf{u})^{\top}H(\mathbf{x})(\mathbf{r}(\mathbf{x})\odot\mathbf{u})d\mathbf{u}\right|\right] \\
&\quad \le E_{\mathbf{X}\sim p}\left[\left|\frac{\varepsilon(\mathbf{x})^{D}}{2V_{\mathcal{B}}}\int\cdots\int_{\mathbf{u}^{\top}\mathbf{u}\le 1}(\varepsilon(\mathbf{x})\mathbf{u})^{\top}H(\mathbf{x})\,(\varepsilon(\mathbf{x})\mathbf{u})\,d\mathbf{u}\right|\right] \\
&\qquad = E_{\mathbf{X}\sim p}\left[\left|\frac{1}{2V_{\mathcal{B}}}\int_{\xi\in\mathcal{B}(x,\varepsilon(\mathbf{x}))}(\boldsymbol{\xi}-\mathbf{x})^{\top}H(\mathbf{x})(\boldsymbol{\xi}-\mathbf{x})d\boldsymbol{\xi}\right|\right] \\
&\qquad\quad \approx E_{\mathbf{X}\sim p}\left[\left|\frac{1}{V_{\mathcal{B}}}\int_{\boldsymbol{\xi}\in\mathcal{B}(\mathbf{x},\varepsilon(\mathbf{x}))} p(\mathbf{x}) - p(\boldsymbol{\xi})d\boldsymbol{\xi}\right|\right] \\
&\qquad\qquad = E_{\mathbf{X}\sim p}\left[\left|p(\mathbf{x}) - \hat{p}_{\varepsilon(\mathbf{x})}(\mathbf{x})|\right|\right]
\end{aligned}
\tag{24}
$$

where the ≈ denotes the Taylor approximation and we used the knowledge that $r_d(\mathbf{x}) \le \varepsilon(\mathbf{x})\ \forall r_d(\mathbf{x}) \in \mathbf{r}(\mathbf{x})$. With the proven inequality, the mean value theorem indicates that the asymptotic bias bound of the proposed estimator is less or equal to the bias bound of the classical estimator. □

Related Works

Local PCA approximation

Several approaches based on local PCA approximation (Gao, Ver Steeg, and Galstyan 2015; Lord, Sun, and Bollt 2018) have been developed. Gao et al. have proposed LNC (Local Nonuniformity Correction) estimator which comprises a local PCA based correction and a test procedure. However, the testing procedure highly depends on an arbitrary ”small value” α and the way of how to determine the α is not addressed. Where as in our proposed method, no hyper-parameter for the testing procedure is required.

Likewise, Lord et al. have proposed a similar approach with a re-centralizing local points and applying SVD to estimate the local volume. Nonetheless, there is no testing or filtering procedure in the method which might lead to over-correction. Besides, in the their paper, only the estimation of mutual information is performed. It is possible that the over-correction resulted error has been cancelled out during the computation. Also note that, both of the above-mentioned works, in their papers, haven't applied their methods in realistically high dimensional cases. The highest dimension used to evaluate LNC is $D = 3$ whereas in Lord et al.'s work the highest dimension is $D = 4$.

Local Gaussian approximation

Approaches based on local Gaussian approximation have been proposed (Gao, Steeg, and Galstyan 2015; Lord, Sun, and Bollt 2018). One approach, kNN-bw (bandwidth) (Gao, Oh, and Viswanath 2016) utilizes a local Gaussian kernel to determine the bandwidth of the kNN estimator; as this method is the only one of this group having a public implementation by authors available, we chose it as the representative for this group of approaches.

Simulation Experiments

We carry out several experiments comparing our estimator EC-kNN to other methods in two tasks, entropy estimation and mutual information estimation[1]. For both tasks we compare to state of the art alternatives, using implementations from their respective authors with default values. We also compare to the baseline kNN estimator using several values of k, whereas for our method EC-kNN k simply fixed to 25 in every dimension, which already turns out to work well. We next describe the data distributions used in the experiments, and then describe the entropy estimation and mutual information tasks along with their respective results.

Designed Cases

To verify the usability of the proposed approach, three cases are set from simple to complicated, details are as follows.

1. **Symmetric Gaussian.** Samples are generated from a d-dimensional Gaussian distribution $\boldsymbol{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The mean vector $\boldsymbol{\mu}$ is set to a zero-vector

$$\boldsymbol{\mu} = [0, 0, \ldots, 0] \tag{25}$$

and the covariance matrix $\boldsymbol{\Sigma}$ is generated as follow

$$\boldsymbol{\Sigma} = \boldsymbol{s} \times \boldsymbol{s}^\top + diag(1, \ldots, 1) \tag{26}$$

where $\boldsymbol{s} \sim \boldsymbol{N}(\boldsymbol{0}, diag(3, 3, ...3))$. The generation of the $\boldsymbol{\Sigma}$ creates the random dependence between dimensions.

2. **Asymmetric Gaussian.** Similar to the previous case, the mean vector $\boldsymbol{\mu}$ is set to a zero-vector but the covariance matrix $\boldsymbol{\Sigma}$ is generated as follow

$$\boldsymbol{\Sigma} = \boldsymbol{s} \times \boldsymbol{s}^\top + diag(1, \ldots, D) \tag{27}$$

where $\boldsymbol{s} \sim \boldsymbol{N}(\boldsymbol{0}, diag(3, 3, ...3))$. When the dimensionality D grows, the variation increases accordingly.

3. **Mixture Gaussian.** A mixture of two Gaussian distributions is selected to evaluate the performance of the proposed method in a scenario which is more complicated than the previous two cases. The probability density function of a mixture of two Gaussian distributions is

$$p(\boldsymbol{x}) = \pi p(\boldsymbol{x}|\boldsymbol{\mu_1}, \boldsymbol{\Sigma_1}) + (1 - \pi) p(\boldsymbol{x}|\boldsymbol{\mu_2}, \boldsymbol{\Sigma_2}) \tag{28}$$

where $\boldsymbol{\mu_1}$ and $\boldsymbol{\mu_2}$ are mean vectors; $\boldsymbol{\Sigma_1}$ and $\boldsymbol{\Sigma_2}$ are covariance matrices of the two different Gaussian distributions and the π is the mixture weight. Here we set π to 0.4,

$$\boldsymbol{\mu_1} = [0, 0, \ldots, 0] \tag{29}$$

$$\boldsymbol{\mu_2} = [10, 10, \ldots, 10] \tag{30}$$

$$\boldsymbol{\Sigma_1} = \boldsymbol{s_1} \times \boldsymbol{s_1}^\top + diag(1, \ldots, 1) \tag{31}$$

$$\boldsymbol{\Sigma_2} = \boldsymbol{s_2} \times \boldsymbol{s_2}^\top + diag(1, \ldots, D) \tag{32}$$

where $\boldsymbol{s_1}$ and $\boldsymbol{s_2}$ are sampled from $\boldsymbol{N}(\boldsymbol{0}, diag(3, 3, ...3))$.

Entropy Estimation

We compare our approach (EC-kNN) with the classical kNN estimator (Kozachenko and Leonenko 1987) and kNN-bw estimator (Gao, Oh, and Viswanath 2016) [2] with three different cases. The ground truth value of entropy of the first two cases can be obtained analytically

$$H(\boldsymbol{X}) = \frac{1}{2} \log \det (2\pi e \boldsymbol{\Sigma}) \tag{33}$$

for the mixture Gaussian case, since the entropy value cannot be analytically obtained, Monte Carlo estimation with 100000 samples sampled from the distribution is employed.

In each case, each dimension and each method, we repeat 10 times simulating 1000 samples and computing the estimation to compare with the ground truth entropy and mutual information. The average of the root mean square error (RMSE) is taken as the performance measure. The result can be found in Figure 2. EC-kNN outperforms both the classical kNN and kNN-bw especially in high dimensionalities.

[1] R implementation can be found in online repository https://github.com/hummmblelu/eckNN

[2] We directly use the program from https://github.com/wgao9/lnn, input parameters are set to the default values

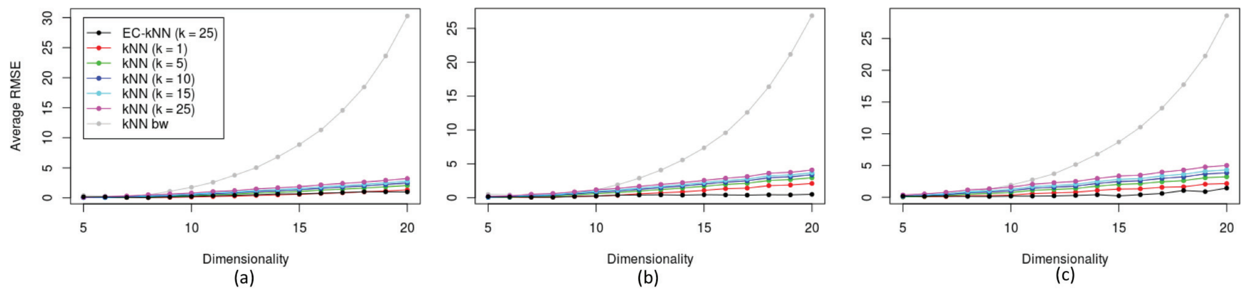

Figure 2: Performance comparisons for entropy estimation. (Average RMSE) (a): Symmetric Gaussian Case (b): Asymmetric Case (c) Mixture Gaussian Case. EC-kNN outperforms other approaches in all cases when dimensionality becomes higher.

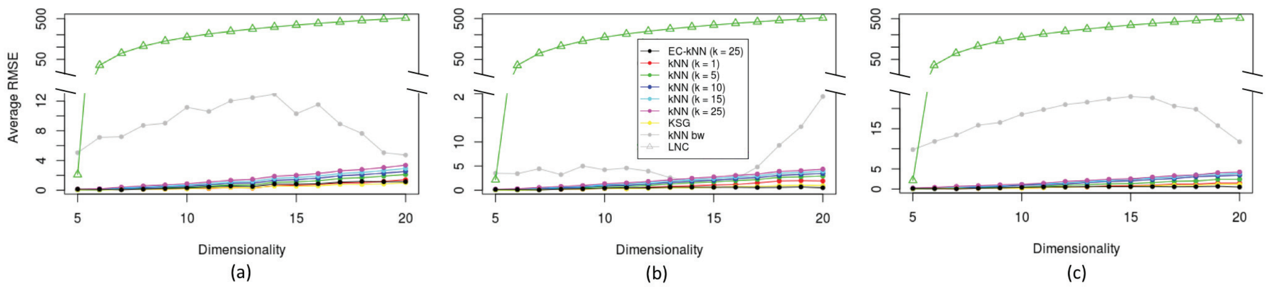

Figure 3: Performance comparisons for mutual information estimation. In order to fit the LNC dot line (grey) into the plot, we apply a logarithm scale on the y axis after around 50 (a): Symmetric Gaussian Case (b): Asymmetric Case (c) Mixture Gaussian Case. The proposed EC-kNN outperforms other approaches in all cases when dimensionality becomes higher.

Mutual Information Estimation

For multivariate extension of mutual information between random variables $\boldsymbol{X} = [X_1, \ldots, X_D]$, where X_d denote the component random variables of the vector-valued random variable $\boldsymbol{X}$, here we adopt the following definition

$$\sum_{d=1}^{D} H(X_d) - H(\boldsymbol{X}). \tag{34}$$

It is also called the total correlation or multi-information (Van de Cruys 2011) and it is also used in one of the above-mentioned works (Gao, Ver Steeg, and Galstyan 2015).

The ground truth value of mutual information for the first two cases can be obtained analytically using the Equation (33). For the third case (mixture Gaussian), the ground true value of mutual information for the mixture Gaussian case is obtained using Monte-Carlo integration with 100000 samples.We compare our approach with kNN estimator, KSG estimator, LNC [3] estimator and kNN-bw estimator.

Again, in each case, each dimension and each method, we repeat 10 times simulating 1000 samples and computing the estimation to compare with the ground truth entropy and mutual information. The average of the root mean square error (RMSE) is taken as the performance measure. The result can be found in Figure 3. EC-kNN again outperforms both the classical kNN and other alternatives especially in high dimensionalities.

Conclusions and Discussions

The contribution of this paper is the novel approach for entropy estimation called the EC-kNN estimator which reduces the bias of the kNN estimator. The proposed EC-kNN comprises the local PCA learning and the boot-strap style correction acceptance procedure which together address the bias resulting from the uniformity assumption.

The advantage of the EC-kNN has been shown to be prominent especially when the data set is high-dimensional and complicated. The experiments have implied that the local ellipsoidal correction and the boot-strap type acceptance procedure can properly capture the local uniformity region.

Our approach provides several interesting directions of future work. Our method using a fixed value of k already outperformed alternatives and performance with optimized k could be even better. Secondly, we proved here a bias bound and additional bounds of variance of the estimator would also be valuable.

Acknowledgments

The work was supported by Academy of Finland decisions 312395 and 313748.

[3] We use the program from https://github.com/BiuBiuBiLL/NPEET_LNC for the computations of KSG and LNC and the input parameters are set to the default values

References

Chauveau, D., and Vandekerkhove, P. 2014. The nearest neighbor entropy estimate: an adequate tool for adaptive mcmc evaluation. Technical report, HAL. https://hal.archives-ouvertes.fr/hal-0106808.

Cover, T. M., and Thomas, J. A. 2006. Elements of information theory 2nd edition. *Willey-Interscience: NJ*.

Gao, W.; Oh, S.; and Viswanath, P. 2016. Breaking the bandwidth barrier: Geometrical adaptive entropy estimation. In *Advances in Neural Information Processing Systems*, 2460–2468.

Gao, W.; Oh, S.; and Viswanath, P. 2018. Demystifying fixed k-nearest neighbor information estimators. *IEEE Transactions on Information Theory*.

Gao, S.; Steeg, G. V.; and Galstyan, A. 2015. Estimating mutual information by local gaussian approximation. In *Uncertainty in Artificial Intelligence*, 278–285.

Gao, S.; Ver Steeg, G.; and Galstyan, A. 2015. Efficient estimation of mutual information for strongly dependent variables. In *Artificial Intelligence and Statistics*, 277–286.

Goria, M. N.; Leonenko, N. N.; Mergel, V. V.; and Novi Inverardi, P. L. 2005. A new class of random vector entropy estimators and its applications in testing statistical hypotheses. *Journal of Nonparametric Statistics* 17(3):277–297.

Kozachenko, L., and Leonenko, N. N. 1987. Sample estimate of the entropy of a random vector. *Problemy Peredachi Informatsii* 23(2):9–16.

Kraskov, A.; Stögbauer, H.; and Grassberger, P. 2004. Estimating mutual information. *Physical review E* 69(6):066138.

Kullback, S., and Leibler, R. A. 1951. On information and sufficiency. *The annals of mathematical statistics* 22(1):79–86.

Lebesgue, H. 1910. Sur l'intégration des fonctions discontinues. In *Annales scientifiques de l'École normale supérieure*, volume 27, 361–450.

Lombardi, D., and Pant, S. 2016. Nonparametric k-nearest-neighbor entropy estimator. *Physical Review E* 93(1):013310.

Lord, W. M.; Sun, J.; and Bollt, E. M. 2018. Geometric k-nearest neighbor estimation of entropy and mutual information. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 28(3):033114.

Moon, K.; Sricharan, K.; Greenewald, K.; and Hero, A. 2018. Ensemble estimation of information divergence. *Entropy* 20(8):560.

Moon, K. R.; Sricharan, K.; and Hero, A. O. 2017. Ensemble estimation of mutual information. In *2017 IEEE International Symposium on Information Theory (ISIT)*, 3030–3034. IEEE.

Noh, Y.-K.; Sugiyama, M.; Liu, S.; Plessis, M. C.; Park, F. C.; and Lee, D. D. 2014. Bias reduction and metric learning for nearest-neighbor estimation of kullback-leibler divergence. In *Artificial Intelligence and Statistics*, 669–677.

Orava, J. 2011. K-nearest neighbour kernel density estimation, the choice of optimal k. *Tatra Mountains Mathematical Publications* 50(1):39–50.

Pérez-Cruz, F. 2008. Kullback-leibler divergence estimation of continuous distributions. In *Information Theory, 2008. ISIT 2008. IEEE International Symposium on*, 1666–1670. IEEE.

Silverman, B. W. 2018. *Density estimation for statistics and data analysis*. Routledge.

Singh, S., and Póczos, B. 2014. Exponential concentration of a density functional estimator. In *Advances in Neural Information Processing Systems*, 3032–3040.

Singh, S., and Póczos, B. 2016a. Analysis of k-nearest neighbor distances with application to entropy estimation. *arXiv preprint arXiv:1603.08578*.

Singh, S., and Póczos, B. 2016b. Finite-sample analysis of fixed-k nearest neighbor density functional estimators. In *Advances in neural information processing systems*, 1217–1225.

Sricharan, K.; Wei, D.; and Hero, A. O. 2013. Ensemble estimators for multivariate entropy estimation. *IEEE transactions on information theory* 59(7):4374–4388.

Van de Cruys, T. 2011. Two multivariate generalizations of pointwise mutual information. In *Proceedings of the Workshop on Distributional Semantics and Compositionality*, 16–20. Association for Computational Linguistics.

Wang, Q.; Kulkarni, S. R.; and Verdú, S. 2009. Divergence estimation for multidimensional densities via k-nearest-neighbor distances. *IEEE Transactions on Information Theory* 55(5):2392–2405.