# $k$ Nearest Neighbor Information Estimators

Weihao Gao

June 28, 2016

## 1 Introduction

This document is a description of the Python package for Fixed $k$-Nearest Neighbor Information estimator (knnie). You can download the code from http://github.com/liverlover/knnie. The related paper can be found in [2]. The repository contains the following files:

- *knnie.py*: Main code.
- *demo.py*: An example of usage.
- *readme.pdf*: Readme document.

## 2 Functions

1. **kraskov_mi(x,y)**: Estimate the mutual information $I(X;Y)$ of $X \in \mathbb{R}^{d_x}$ and $Y \in \mathbb{R}^{d_y}$ from samples $\{x_i, y_i\}_{i=1}^N$ using KSG estimator [1].
   - **x**: A 2D list of dimension $N \times d_x$, where each row is one sample $x_i \in \mathbb{R}^{d_x}$.
   - **y**: A 2D list of dimension $N \times d_y$, where each row is one sample $y_i \in \mathbb{R}^{d_y}$.
   - Output: Scalar $\widehat{I}(X;Y)$.
2. **revised_mi(x,y)**: Estimate the mutual information $I(X;Y)$ of $X \in \mathbb{R}^{d_x}$ and $Y \in \mathbb{R}^{d_y}$ from samples $\{x_i, y_i\}_{i=1}^N$ using revised KSG estimator [2].
   - **x**: A 2D list of dimension $N \times d_x$, where each row is one sample $x_i \in \mathbb{R}^{d_x}$.
   - **y**: A 2D list of dimension $N \times d_y$, where each row is one sample $y_i \in \mathbb{R}^{d_y}$.
   - Output: Scalar $\widehat{I}(X;Y)$.
3. **kraskov_multi_mi(x,y)**: Estimate the mutual information $I(X;Y;Z) \triangleq H(X) + H(Y) + H(Z) - H(X,Y,Z)$ of $X \in \mathbb{R}^{d_x}$, $Y \in \mathbb{R}^{d_y}$ and $Z \in \mathbb{R}^{d_z}$ from samples $\{x_i, y_i, z_i\}_{i=1}^N$ using simple generalization of KSG estimator [1].
   - **x**: A 2D list of dimension $N \times d_x$, where each row is one sample $x_i \in \mathbb{R}^{d_x}$.
   - **y**: A 2D list of dimension $N \times d_y$, where each row is one sample $y_i \in \mathbb{R}^{d_y}$.
   - **z**: A 2D list of dimension $N \times d_z$, where each row is one sample $z_i \in \mathbb{R}^{d_y}$.
   - Output: Scalar $\widehat{I}(X;Y;Z)$.
4. **revised_multi_mi(x,y)**: Estimate the mutual information $I(X;Y;Z) \triangleq H(X) + H(Y) + H(Z) - H(X,Y,Z)$ of $X \in \mathbb{R}^{d_x}$, $Y \in \mathbb{R}^{d_y}$ and $Z \in \mathbb{R}^{d_z}$ from samples $\{x_i, y_i, z_i\}_{i=1}^N$ using revised KSG estimator [2].
   - **x**: A 2D list of dimension $N \times d_x$, where each row is one sample $x_i \in \mathbb{R}^{d_x}$.
   - **y**: A 2D list of dimension $N \times d_y$, where each row is one sample $y_i \in \mathbb{R}^{d_y}$.
   - **z**: A 2D list of dimension $N \times d_z$, where each row is one sample $z_i \in \mathbb{R}^{d_y}$.
   - Output: Scalar $\widehat{I}(X;Y;Z)$.

# 3 Usage

Here we provide a simple sample of usage of the package. Here $X$, $Y$ and $Z$ are independent standard Gaussian random variable, so $I(X;Y) = I(X;Y;Z) = 0$.

```
>> import numpy.random as nr
>> import knnie
>> x = nr.normal(0,1,[1000,1])
>> y = nr.normal(0,1,[1000,1])
>> z = nr.normal(0,1,[1000,1])
>> print "I(X;Y) = ", knnie.kraskov_mi(x,y)
I(X;Y) = -0.0291136513027
>> print "I(X;Y) = ", knnie.revised_mi(x,y)
I(X;Y) = -0.0289167000205
>> print "I(X;Y;Z) = ", knnie.kraskov_multi_mi(x,y,z)
I(X;Y) = 0.00332720859842
>> print "I(X;Y;Z) = ", knnie.revised_multi_mi(x,y,z)
I(X;Y) = 0.00723561378027
```

You can find the code in *demo.py*.

# References

[1] Kraskov A, Stögbauer H, Grassberger P. Estimating mutual information[J]. Physical review E, 2004, 69(6): 066138.

[2] Gao, W., Oh, S. and Viswanath, P. Demystifying Fixed $k$-Nearest Neighbor Information Estimators, http://arxiv.org/abs/1604.03006/.