LA-UR-17-27781
Approved for public release; distribution is unlimited.

Title: Estimating Mutual Information for High-to-Low Calibration

Author(s): Michaud, Isaac James
Williams, Brian J.
Weaver, Brian Phillip

Intended for: Report

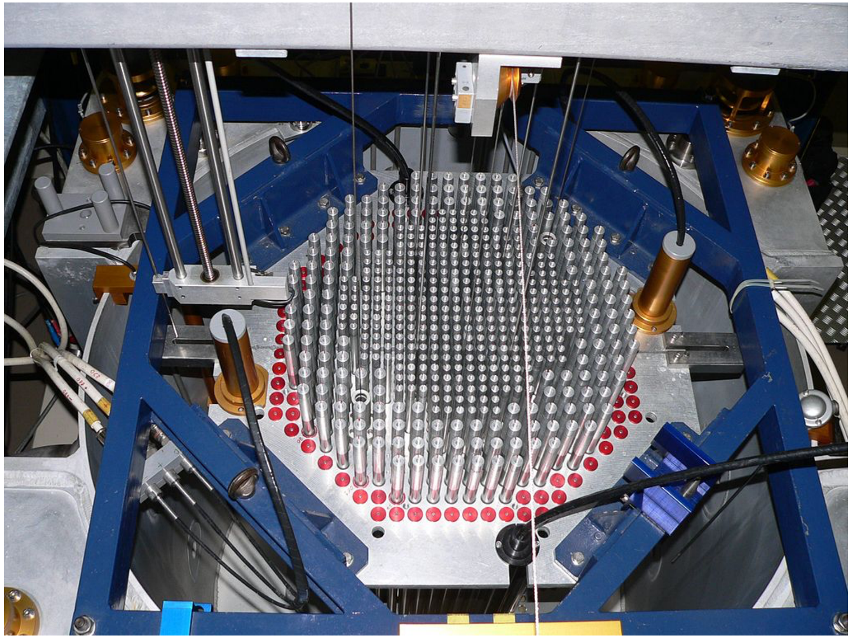
Issued: 2017-10-05 (rev.1)

Estimating Mutual Information for High-to-Low Calibration

Isaac J. Michaud
Brian J. Williams
Brian P. Weaver

High Fidelity Computer Codes

- **Simulations are an integral part of modern scientific research**
- **Monte Carlo N-Particle Transport Code (MCNP)**
  - LANL developed over 60 years
  - Simulates the movement and interactions of particles
  - User specified geometry and material cross-sections
  - Used to study reactor designs
  - Slow but accurate
- **How can we use these slow codes to study a process?**



High Fidelity (HiFi) to Low Fidelity (LoFi) Calibration

- **LoFi Mathematical Model:** $d_l(\theta, \xi)$
  - $\xi \in \Xi$, control variables
  - $\theta \in \Theta$, calibration parameters
- **Calibration involves finding θ so d_l matches observations**
- **Here the observation are from HiFi code**
- **Problem: Calibrate LoFi while minimizing HiFi evaluations**

$$\underbrace{\tilde{d}_n}_{\text{high-fidelity observation}} = \underbrace{d_l(\theta, \xi_n)}_{\text{low-fidelity model}} + \underbrace{\delta(\xi_n)}_{\text{systematic bias}} + \underbrace{\tilde{\epsilon}_n(\xi_n)}_{\text{random error}}$$

- **Challenge: Optimal experimental design exploits the mathematical structure of the statistical model being fit or calibrated**

High-to-Low Calibration with Mutual Information (MI)

- **Lewis et al. 2016 proposed sequentially optimizing MI between parameters and data,**

$$I(\theta; d_n | D_{n-1}, \xi_n) = \int_{\mathcal{D}} \int_{\Omega} p(\theta, d_n | D_{n-1}, \xi_n) \log \frac{p(\theta, d_n | D_{n-1}, \xi_n)}{p(\theta | D_{n-1}) p(d_n | D_{n-1}, \xi_n)} d\theta dd_n$$

$$\xi_n^* = \arg\max_{\xi_n \in \Xi} I(\theta; d_n | D_{n-1}, \xi_n)$$

  - MI is the expected Kullback–Leibler divergence between prior and posterior distributions of $\boldsymbol{\theta}$ if $\boldsymbol{d_n}$ is collected at ξ_n
  - Measures expected reduction in uncertainty (entropy) in parameters
  - Special cases: maximum entropy sampling and D-optimality
- **Requires integrating a known joint density (difficult)**
- **Lewis et al. estimated MI using samples instead**

Estimating Mutual Information

Method	Assumptions
Monte Carlo, MLE, Parametric	Known joint density
Binning, KDE	Small dimension
K^{th} Nearest Neighbor (kNN)	Locally uniform joint density

- **kNN Pros:**
  - Require fewer samples than brute force methods
  - Faster than KDE $(\mathcal{O}(n \log n) \text{ vs. } \mathcal{O}(n^2))$
- **kNN Cons:**
  - Asymptotic theory is not fully developed
  - Better at estimating independence than dependence

Estimating Shannon Entropy

MI can be decomposed into marginal and joint entropies:

$$I(X,Y) = H(X) + H(Y) - H(X,Y)$$

where

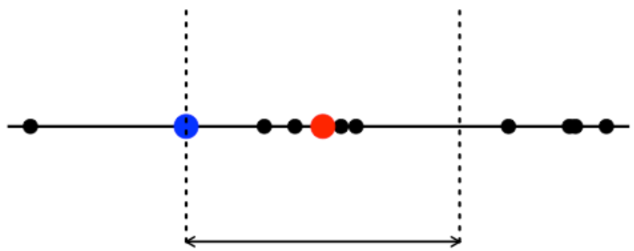
$$H(X) = -\int f(x)\log(f(x))dx$$

Kozachenko-Leonenko entropy (KLE) estimator, let $\epsilon_x(i)/2$ be the distance to kth nearest neighbor of point i,

$$\hat{H}(X) = -\psi(k) + \psi(N) + \log c_{d_x} + \frac{d_x}{N}\sum_{i=1}^{N}\log\epsilon_x(i)$$

where ψ is the digamma function, d_x is the dimension of X, and c_{d_x} is the volume of max-norm unit ball in $\mathbb{R}^{d_x}$

E.g. $k = 5$



Kraskov et al. 2004 - Mutual Information Estimation

- **Naive mutual information estimator**

$$\hat{I}(X,Y) = \hat{H}(X) + \hat{H}(Y) - \hat{H}(X,Y)$$

$$= -\psi(k) + \psi(N) + \frac{d_x}{N}\sum_{i=1}^{N}\log\epsilon_x(i) + \frac{d_y}{N}\sum_{i=1}^{N}\log\epsilon_y(i) - \frac{d_x+d_y}{N}\sum_{i=1}^{N}\log\epsilon_{xy}(i)$$

- **Problem: Biases in each estimate are unlikely to cancel**
- **Solution: Force** $\epsilon_{xy}(i) = \epsilon_x(i) = \epsilon_y(i)$ **and varying *k* in the marginal estimates**

Kraskov et al. 2004 - Algorithm 1

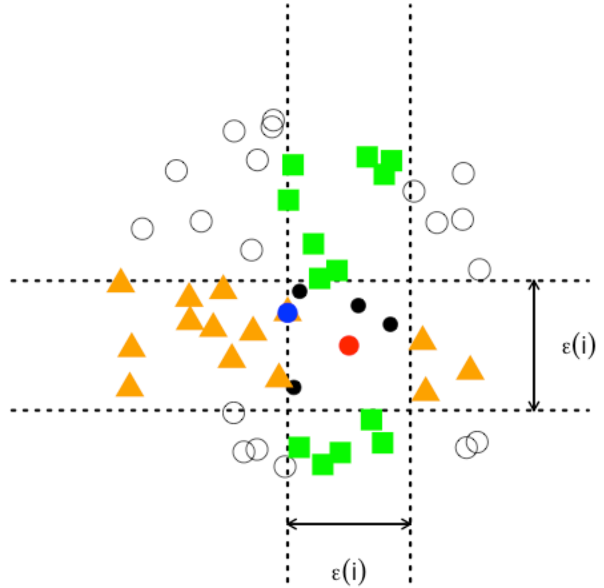
$$KSG_1(X,Y) = \psi(N) + \psi(k) - \frac{1}{N}\sum_{i=1}^{N}\psi(n_x(i)+1) - \frac{1}{N}\sum_{i=1}^{N}\psi(n_y(i)+1)$$

1. **For each point $z_i = (x_i, y_i)$, find its k^th^ nearest neighbor z_i^k in the joint space**
2. **Define:** $$\frac{\epsilon(i)}{2} = ||z_i^k - z_i||_\infty$$
3. **Compute:**

$$n_x(i) = \sum_{j \neq i} I\left(||x_j - x_i||_\infty < \frac{\epsilon(i)}{2}\right)$$

$$n_y(i) = \sum_{j \neq i} I\left(||y_j - y_i||_\infty < \frac{\epsilon(i)}{2}\right)$$

E.g. $k = 5$



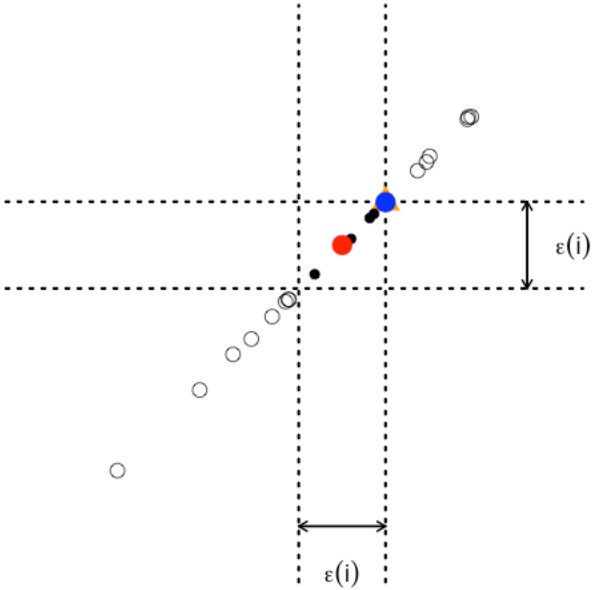
Kraskov et al. 2004 - Algorithm 1

Consider the linear model: $y_i = \beta_0 + \beta_1 x_i + \eta_i$

where $x_i \overset{iid}{\sim} \mathcal{U}(0,1)$ and $\eta_i \overset{iid}{\sim} \mathcal{N}(0,\sigma^2)$

- **Dependence between X and Y is govern by σ^2 and β_1**
- **Problem 1:**

$$\sigma^2 \to 0 \Rightarrow \begin{array}{l} n_x(i) \to k \\ n_y(i) \to k \end{array}$$



$$KSG_1(X,Y) \approx \psi(N) - \psi(k)$$

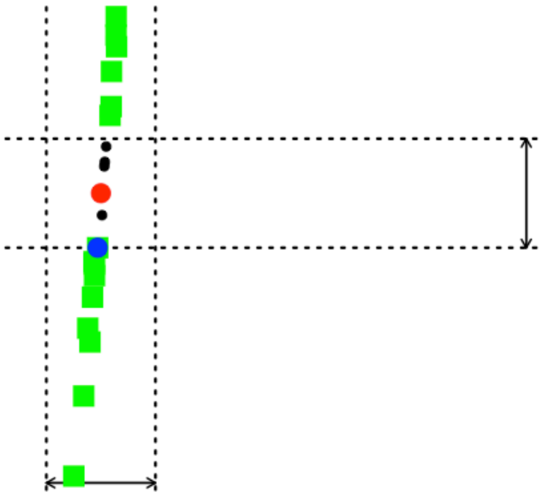
- **Maximum estimable information**

Kraskov et al. 2004 - Algorithm 1

- **Problem 2:**

$$\beta_1 \to \infty \Rightarrow \quad \begin{array}{l} n_x(i) \to N \\ n_y(i) \to k \end{array}$$

$$KSG_1(X,Y) \approx 0$$



- **This can be "fixed" by rescaling both variables by their standard deviations, giving:**

$$KSG_1(X/\sigma_X, Y/\sigma_Y) \approx \psi(N) - \psi(k)$$

Kraskov et al. 2004 - Algorithm 2

$$KSG_2(X,Y) = \psi(N) + \psi(k) - \frac{1}{N}\sum_{i=1}^{N}\psi(n_x(i)) - \frac{1}{N}\sum_{i=1}^{N}\psi(n_y(i)) - \frac{1}{k}$$

1. Let z_i^j be the j[th] nearest neighbor of $z_i = (x_i, y_i)$ in the joint space
2. Define:

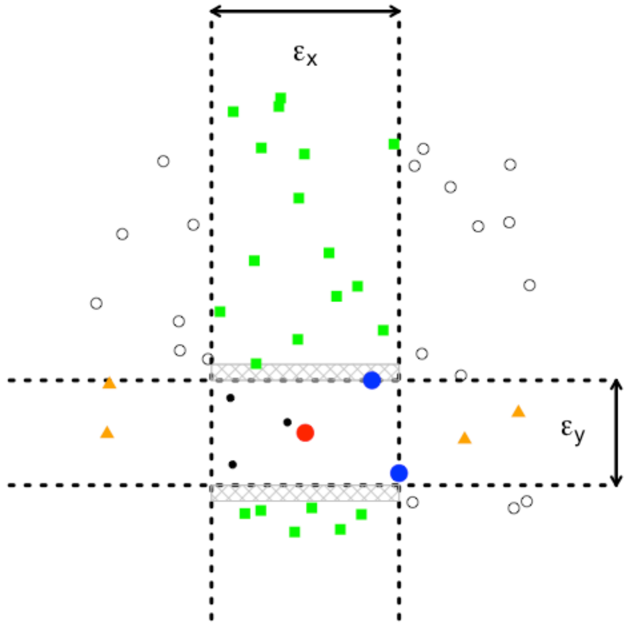
$$\frac{\epsilon_x(i)}{2} = \max_{1\le j\le k} ||x_i^j - x_i||_\infty$$

$$\frac{\epsilon_y(i)}{2} = \max_{1\le j\le k} ||y_i^j - y_i||_\infty$$

3. Compute:

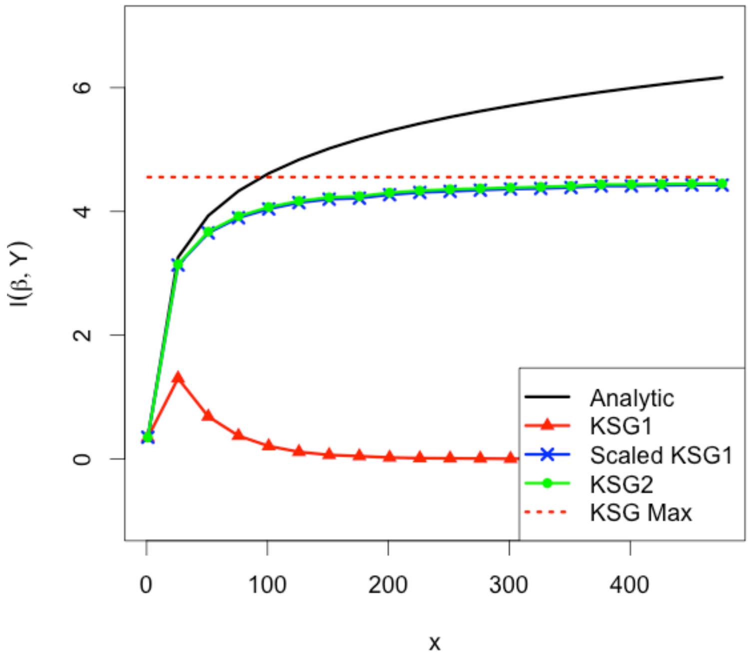
$$n_x(i) = \sum_{j\neq i} I\left(||x_j - x_i||_\infty \le \frac{\epsilon_x(i)}{2}\right)$$

$$n_y(i) = \sum_{j\neq i} I\left(||y_j - y_i||_\infty \le \frac{\epsilon_y(i)}{2}\right)$$



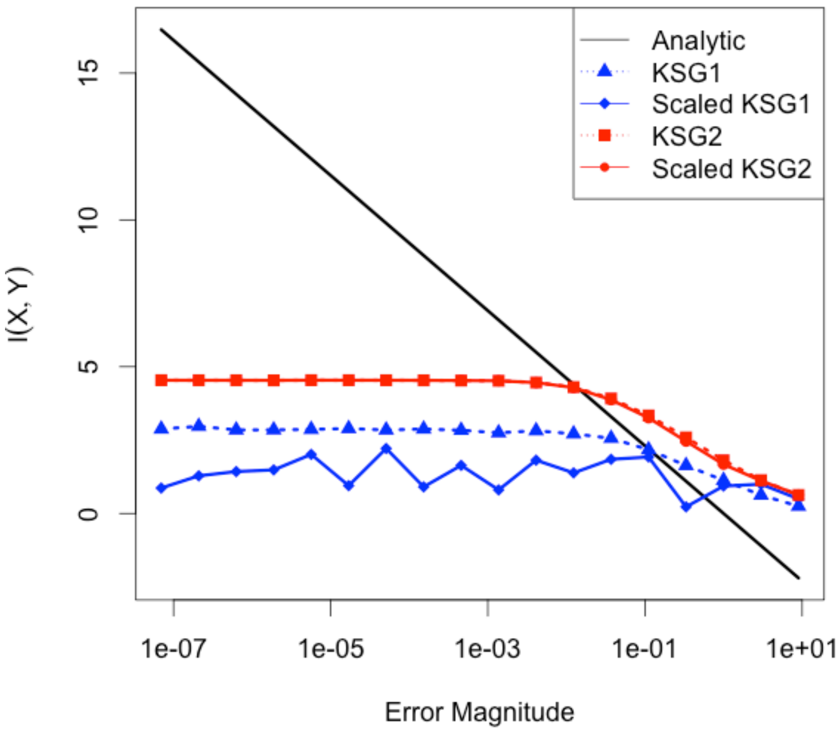
Simple Linear Scaling

- **Consider the linear model:** $y_i = \beta x_i + \eta_i, \ \beta \sim \mathcal{U}[0,1], \eta_i \sim \mathcal{N}(0,1)$
- **Estimating $I(\beta, Y)$ from 1000 samples**
- **KSG_2 and Scaled KSG_1 perform similarly**
- **Unscaled KSG_1 initially increases and then decays to zero**



Reciprocal: X~unif(0,1), U~unif(-ϵ/2, ϵ/2), $Y = \frac{1}{X} + U$

- Nonlinear relationships remain after scaling
- KSG_1 performs worse after scaling than without
- KSG_2 adapts locally to nonlinearities

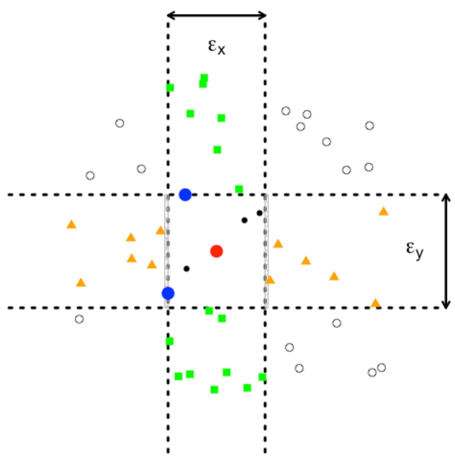


KSG_1 and KSG_2 Summary

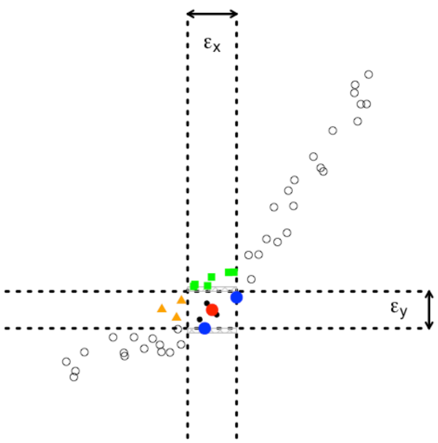
- **Most documentation does not specify used method**
- **Both are limited to a maximum estimable MI** $(\approx \boldsymbol{log}(\boldsymbol{N}))$
- **Both are best at estimating near zero MI (independence)**
- **Assumptions:**
  - Independent Samples
  - Continuous variables
  - Locally uniform joint density
- **KSG_1**
  - More common version found in software
  - Bias if variables have disparate scales
  - Scaling problem is fixed if the variables can be standardized (globally)
- **KSG_2**
  - Scaling issue is avoided by handling each variable separately

Improving KSG Estimators

- **Locally uniform joint density over the nearest neighbor rectangle**
- **Non-Uniformity indicates high MI**
  - **Option 1: Increase sample size exponentially**
  - **Option 2: Modify the estimator for non-uniformity**



Plausibly Uniform

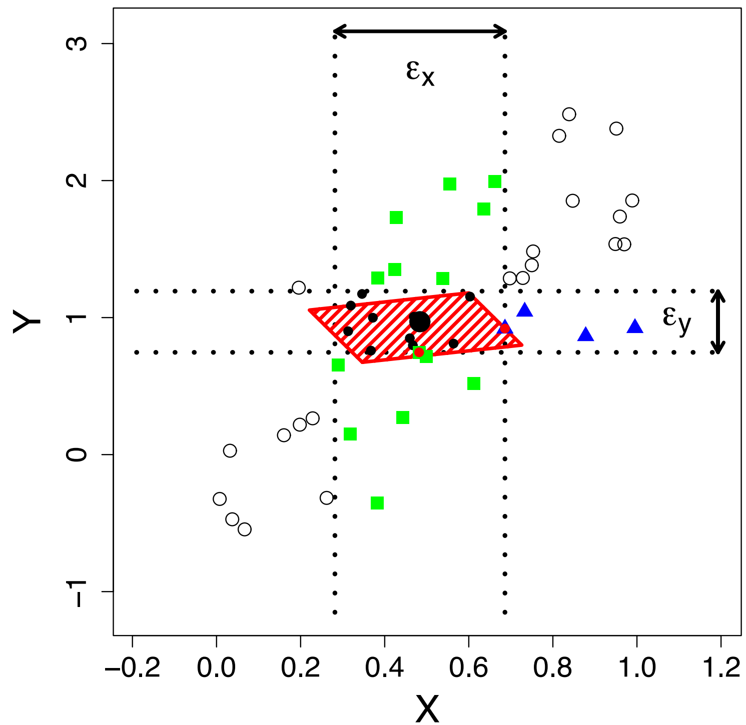


Not Uniform

Local Non-Uniformity Corrected KSG (LNC)

$$LNC(X,Y) = KSG_2(X,Y) - \frac{1}{N}\sum_{i=1}^{N} I\left(\frac{\bar{V}(i)}{V(i)} < \alpha_{k,d}\right)\log\frac{\bar{V}(i)}{V(i)}$$

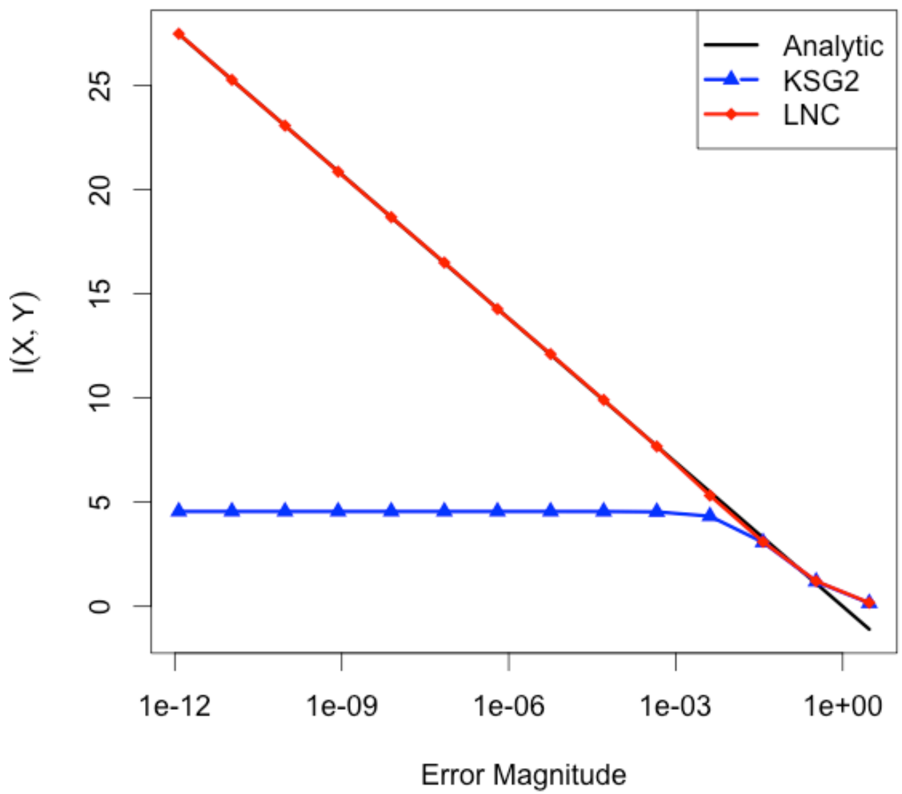
- **Developed by Gao et al. 2015**
- **kNN neighborhood may not be uniform, perhaps some volume within it is?**
- **Use PCA to uncover this volume**
- **Same as KSG_2 with adjustments for non-uniformity**
- $V(i)$ **- kNN neighborhood volume**
- $\bar{V}(i)$ **- PCA aligned volume**
- $\alpha_{k,d}$ **- correction threshold**
- d **- dimension of joint space**



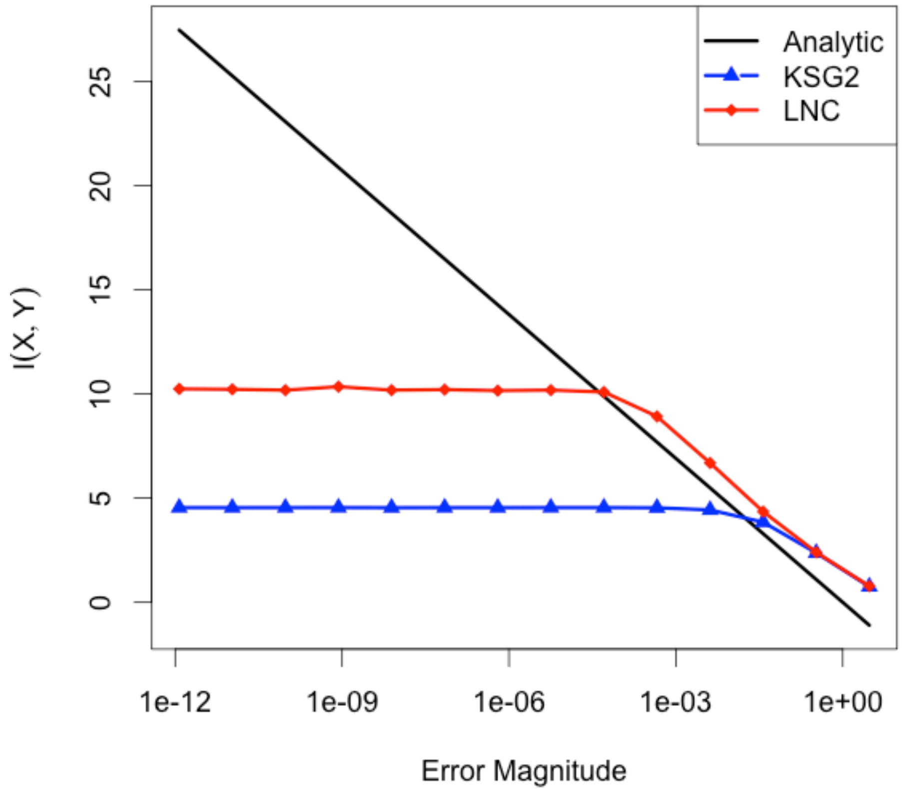
Selecting α for LNC

- **If $\bar{V}(i) < V(i)$ the correction is positive**
- **If $\bar{V}(i) > V(i)$ the correction is negative**
- **α defines a threshold for applying the correction**
- **Optimal α is selected using Monte Carlo algorithm:**
  1. Sample $\boldsymbol{k}$ points from multidimensional uniform distribution $\boldsymbol{N}$ times
  2. Compute $\bar{V}(i)/V(i)$
  3. Set $\boldsymbol{\alpha}$ to be p^{th} sample quantile $(p = 0.005)$
- **Under the assumption of local uniformity, using the α defined above will cause the correction to be applied 0.005 of the time**
- **N and p can be varied**
- **Low α filters out moderately dependent relationships**
- **High α inflates estimates**

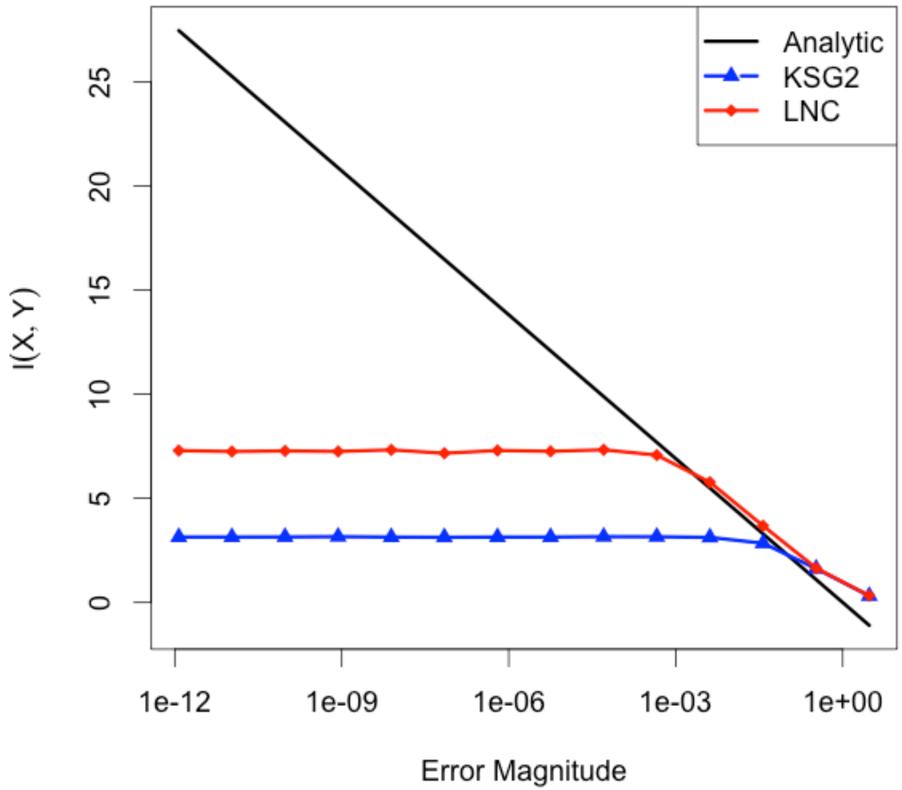
Linear: X~unif(0,1), U~unif(-ϵ/2, ϵ/2), $Y = X + U$



Quadratic: X~unif(0,1), U~unif(-ϵ/2, ϵ/2), $Y = 5X^2 + U$



Periodic: X~unif(0,1), U~unif(-ϵ/2, ϵ/2), $Y = \sin(4\pi X) + U$



Simple Linear Scaling

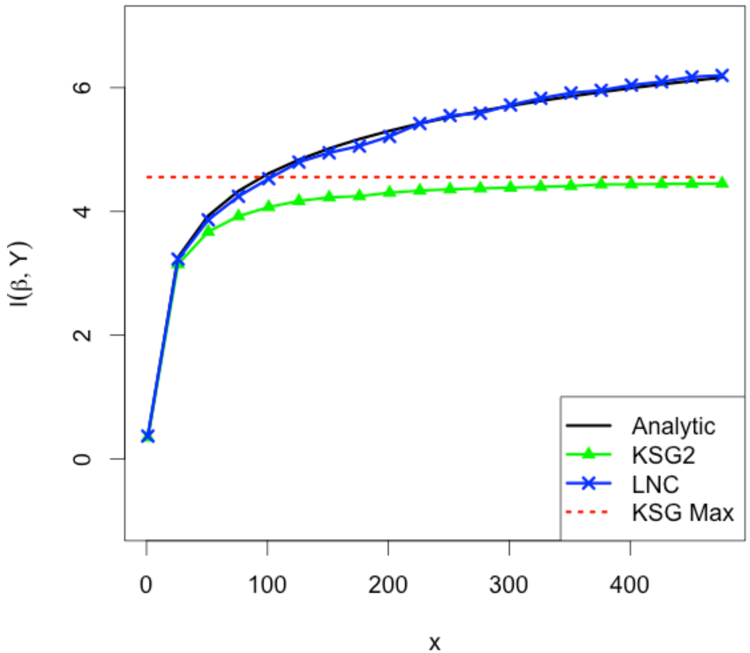
- **Consider the linear model:** $y_i = \beta x_i + \eta_i,\ \beta \sim \mathcal{U}[0,1], \eta_i \sim \mathcal{N}(0,1)$

- **Estimating $I(\beta, Y)$ from 1000 samples**

- **LNC reproduces the analytic mutual information over the entire range of X**

- **LNC is not limited to the same maximum mutual information as KSG**

- **LNC's performance is better for more extreme scalings (>1e5)**



Generalization to Multivariate Mutual Information

- Kraskov et al. generalized their estimators to compute high dimensional redundancy

$$I(x_1; x_2; x_3; x_4) = \int\int\int\int f(x_1, x_2, x_3, x_4) \log \frac{f(x_1, x_2, x_3, x_4)}{f(x_1)f(x_2)f(x_3)f(x_4)} dx_1 dx_2 dx_3 dx_4$$

- For high-to-low calibration we need information gain

$$I((x_1, x_2); (x_3, x_4)) = \int\int\int\int f(x_1, x_2, x_3, x_4) \log \frac{f(x_1, x_2, x_3, x_4)}{f(x_1, x_2)f(x_3, x_4)} dx_1 dx_2 dx_3 dx_4$$

Improved Local Non-Uniformity Corrected KSG (iLNC)

- Correlated parameters cause LNC to over correct
- Correction applied to the joint space, even when activated by dependence in the marginal spaces
- We modified LNC to correct for correlations within the parameter and predictive distributions:

$$iLNC(X_1, ..., X_\ell) = LNC(X_1, ..., X_\ell) + \frac{1}{n} \sum_{j=1}^{\ell} \sum_{i=1}^{n} I\left(\frac{\bar{V}_j(i)}{V_j(i)} < \alpha_{k,d_{X_j}}\right) \log\left(\frac{\bar{V}_j(i)}{V_j(i)}\right)$$

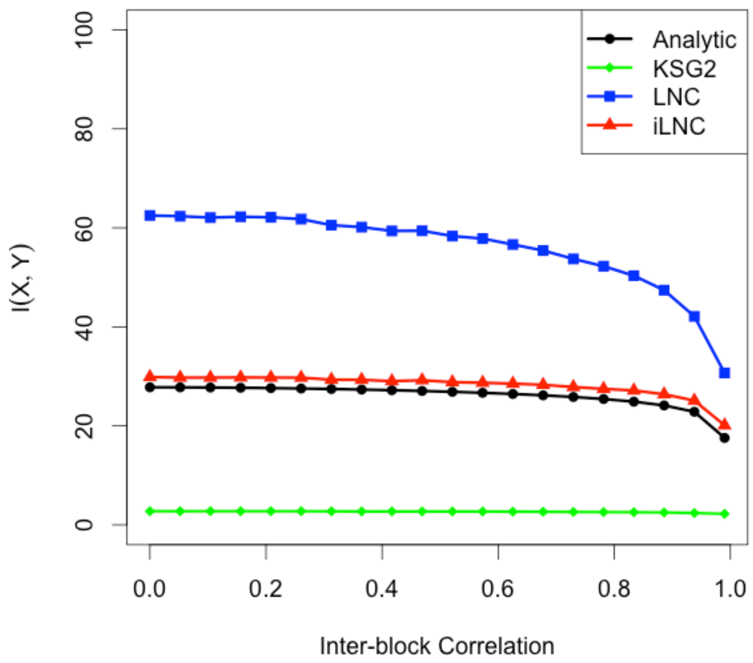
- Typically $\ell = 2$ for high-to-low calibration
- α terms are selected using the same algorithm as described for LNC

Multivariate Normal Simulation Study

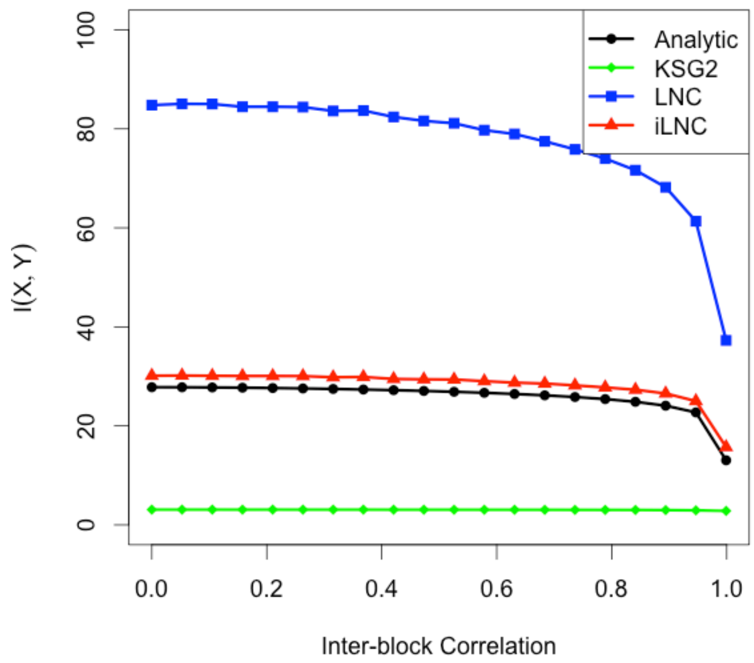
- **25 dimensional calibration parameter vector**
  - Normally distributed vector
  - 5 blocks of 5 parameters each
  - Block-compound symmetric covariance structure
  - intra-block correlation ρ_{intra} and inter-block correlation set to ρ_{inter}
- **5 dimensional prediction vector**
  - $Y = TX + \epsilon$, where $\epsilon \sim N(0, \sigma^2 I)$
  - $E[y_1] = 100(x_1)$
  - $E[y_2] = 100(x_6 + x_7)$
  - $E[y_3] = 100(x_{11} + x_{12} + x_{13})$
  - $E[y_4] = 100(x_{16} + x_{17} + x_{18} + x_{19})$
  - $E[y_5] = 100(x_{21} + x_{22} + x_{23} + x_{24} + x_{25})$

Multivariate Normal Simulation Study

$\rho_{\text{intra}} = 0.99$



$\rho_{\text{intra}} = 0.999$



Steady-State Heat Model (Lewis et al. 2016)

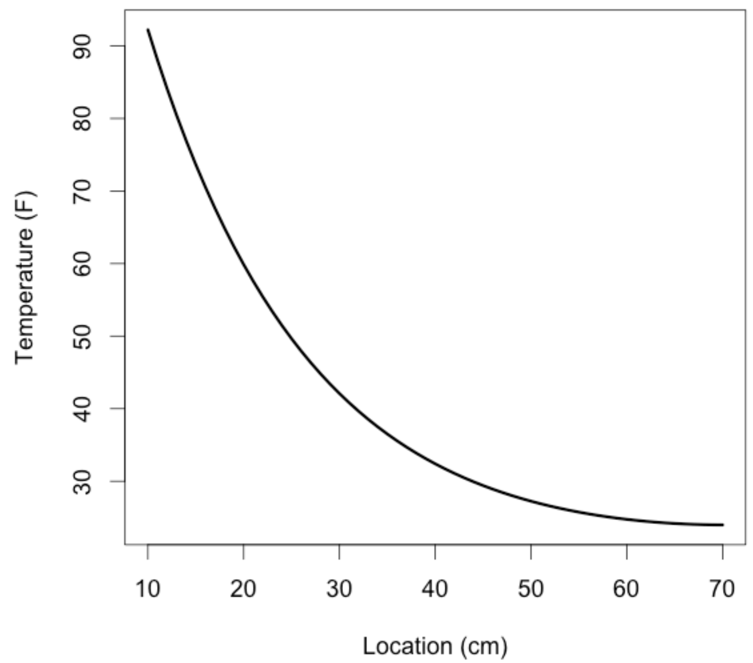
- **70 cm metal bar heated at endpoint**
- **Model equilibrium heat distribution using steady-state heat equation**

$$T_s(x;\phi) = c_1(\phi)e^{-\gamma x} + c_2(\phi)e^{\gamma x} + T_{amb}$$

$$c_1(\phi) = -\frac{\Phi}{K\gamma}\left[\frac{e^{\gamma L}(h+K\gamma)}{e^{-\gamma L}(h+K\gamma)+e^{\gamma L}(h+K\gamma)}\right]$$

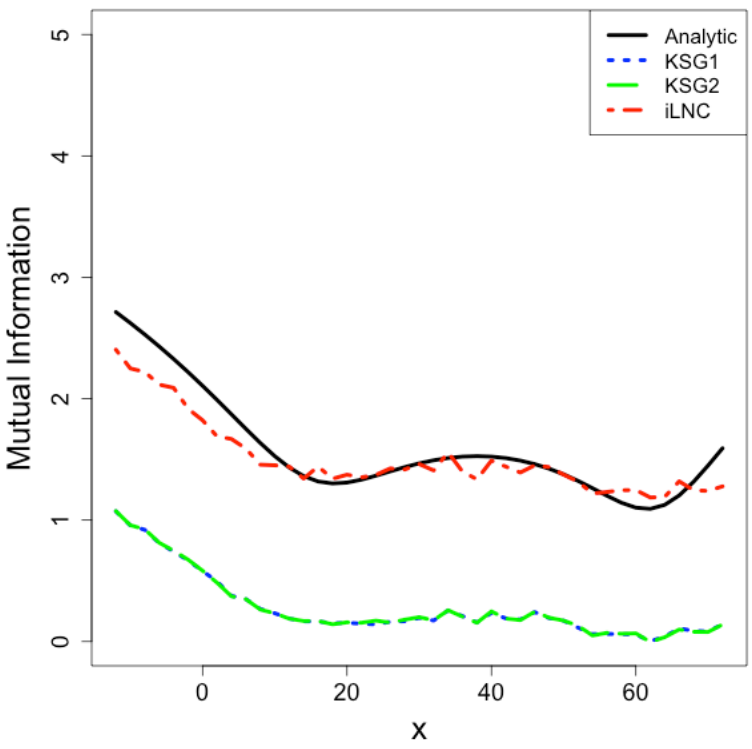
$$c_2(\phi) = \frac{\Phi}{K\gamma} + c_1(\phi)$$

- **HiFi:** $\tilde{d}_n = T_s(x_n;\phi) + \tilde{\epsilon}(x_n)$
- **LoFi:** $y = Ax^2 + Bx + C$
- **Design Space: [10,66]**



Steady-State Heat Model

- Initial Data at $x = 10, 38, 66, 66$
- Posteriors simulated using DRAM
- 1000 parameter samples used
- MI computed over design space
- Analytic mutual information criterion is the same as D-optimal criterion because LoFi is linear model with normal errors
- Maximizing mutual information tries to move design towards a balanced three-point design
- iLNC provides more fidelity of MI criterion than KSG



Steady-State Heat Model

- **Design points selected with replacement**
- **Mutual information optimization performed with GADGET (GP optimization)**

Stage	0	1	2	3	4	5	6
D-Optim	10,38,66	10	38	66	10	38	66
MI (iLNC)	10,38,66	10	42	10	66	38	10

- **iLNC produces design similar to D-optimal design as expected**
- **iLNC used 1000 sample points, larger samples should improve performance**

Conclusions

- KSG_2 is superior to KSG_1 because it scales locally automatically
- KSG estimators are limited to a maximum MI due to sample size
- LNC extends the capability of KSG without onerous assumptions
- iLNC allows LNC to estimate information gain

Recommendations:

1. Jitter the sample points to break possible ties (magnitude: 1e-10)
2. Center and scale each variable independently
3. Replace KSG_1 with KSG_2
4. Incorporate iLNC and α estimator

Future Work

- Develop selection method for optimal k for LNC and iLNC
- Simulation study of high dimensional nonlinear relationships
- Sensitivity to approximate nearest neighbor (ANN) algorithm
- Sensitivity to independence assumption
- Manifold learning methods for estimating MI

References:

Gao, Shuyang, Greg Ver Steeg, and Aram Galstyan. "Efficient estimation of mutual information for strongly dependent variables." Artificial Intelligence and Statistics. 2015.

Kraskov, Alexander, Harald Stögbauer, and Peter Grassberger. "Estimating mutual information." *Physical review E* 69.6 (2004): 066138.

Lewis, Allison, et al. "An information theoretic approach to use high-fidelity codes to calibrate low-fidelity codes." Journal of Computational Physics 324 (2016): 24-43.

Singh, Harshinder, et al. "Nearest neighbor estimates of entropy." American journal of mathematical and management sciences 23.3-4 (2003): 301-321.

Conditional Entropy Estimator (conEnt)

- **Mutual information can be decomposed as**

$$I(X,Y) = H(Y) - H(Y|X)$$

- **Entropy of Y can estimated easily using KLE, but conditional entropy is more difficult for small samples**
- **Inspired by Manifold Learning, we estimate conditional entropy using linear models fit to kNN defined neighborhoods**
- **Assuming normally distributed residuals:**

$$\hat{I}_{\mathrm{conEnt}}(X,Y) = \hat{H}(Y) - \frac{1}{N}\sum_{i=1}^{N}\frac{\log(2\pi e\hat{\sigma}_i^2)}{2}$$

where $\hat{\sigma}_i^2$ **is the MSE for the model fit locally around point** i