# Model-Powered Conditional Independence Test

**Rajat Sen*, Ananda Theertha Suresh§, Karthikeyan Shanmugam¶, Alexandros G. Dimakis* and Sanjay Shakkottai***

**University of Texas at Austin, §Google, New York ,¶IBM Research, New York*

## Conditional Independence Testing

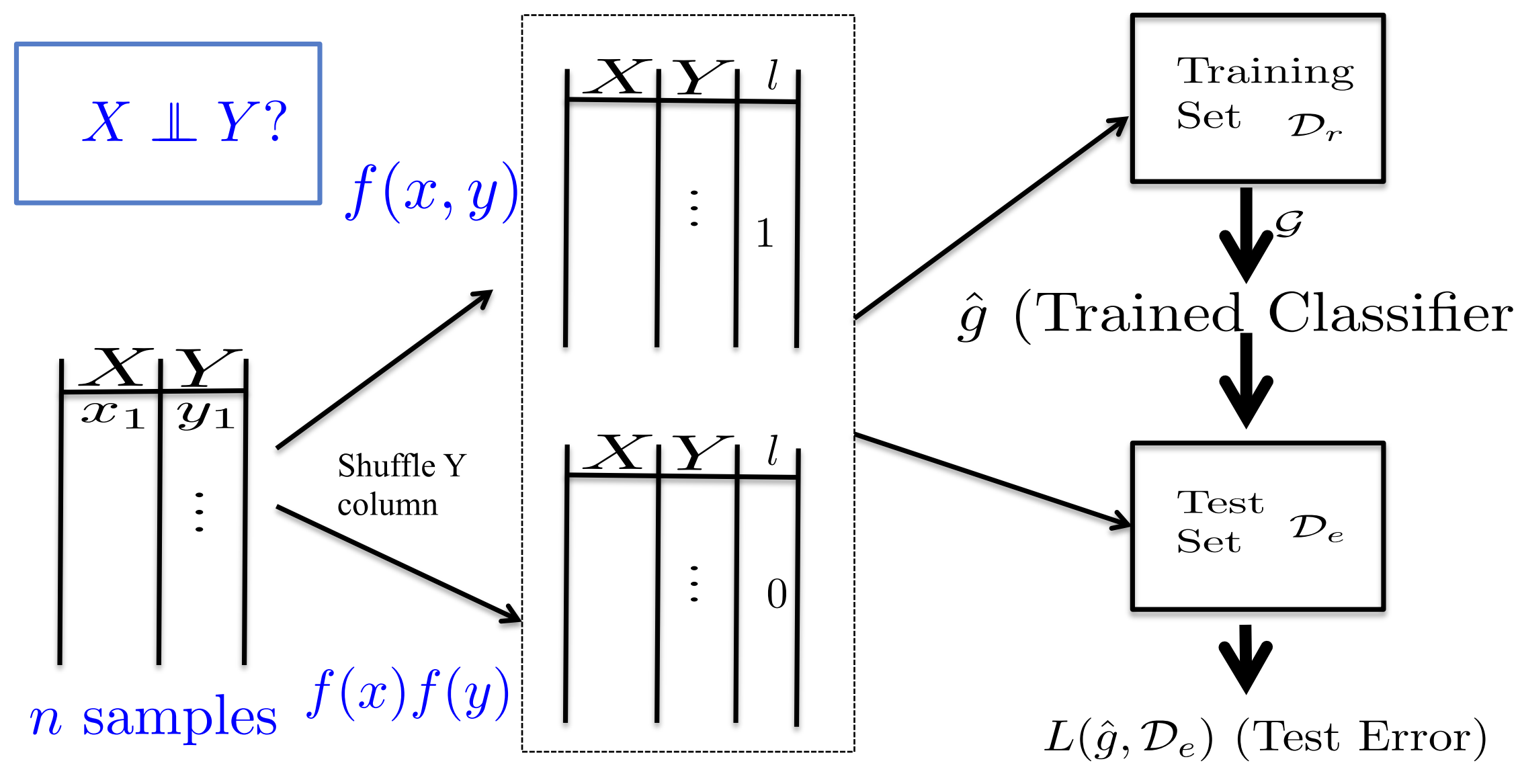
- Given $n$ samples i.i.d from $f_{X,Y,Z}(x,y,z)$ distinguish between:
  - $\mathcal{H}_0$: $X \perp\!\!\!\perp Y | Z \Leftrightarrow f_{X,Y,Z}(x,y,z) = f^{CI}(x,y,z)$
  - $\mathcal{H}_1$: $X \not\perp\!\!\!\perp Y | Z \Leftrightarrow f_{X,Y,Z}(x,y,z) \neq f^{CI}(x,y,z)$

$$f^{CI}(x,y,z) = f_{X|Z}(x|z) f_{Y|Z}(y|z) f_Z(z)$$

$x \in \mathbb{R}^{d_x}$
$y \in \mathbb{R}^{d_y}$
$z \in \mathbb{R}^{d_z}$

- Non-parametric conditional independence testing for continuous r.v's
- Applications in Causal Inference [23,14], Bayesian Networks [15,27], Feature Selection [16,31]…..

## Warm-up (Model-Powered Independence Test)



$X \perp\!\!\!\perp Y?$

$f(x,y)$

$X$ $Y$ $x_1$ $y_1$

$n$ samples

Shuffle Y column

$f(x)f(y)$

$X$ $Y$ $l$ 1

$X$ $Y$ $l$ 0

Training Set $\mathcal{D}_r$

$\mathcal{G}$

$\hat{g}$ (Trained Classifier)

Test Set $\mathcal{D}_e$

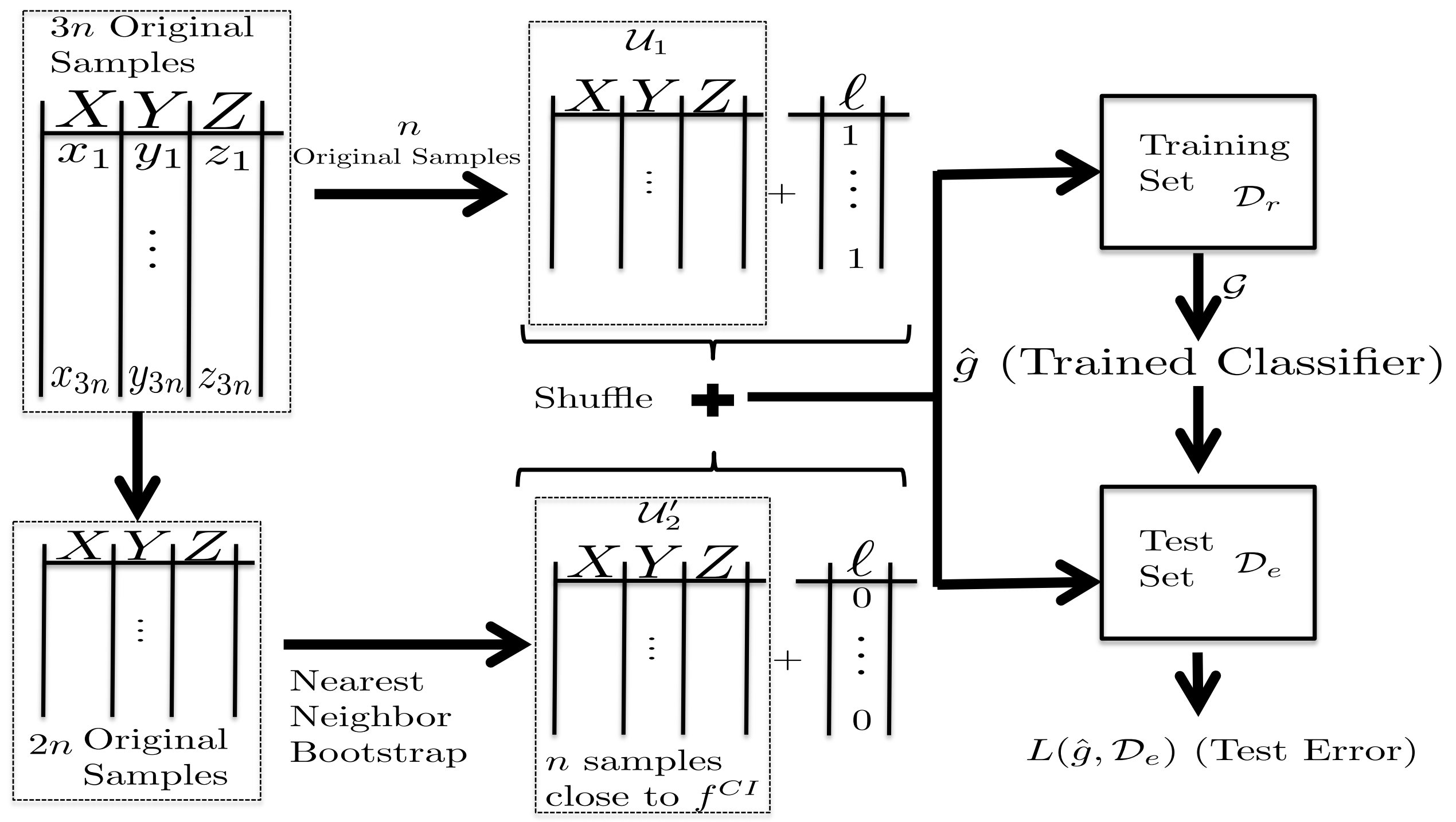
$L(\hat{g}, \mathcal{D}_e)$ (Test Error)

- Powerful classifiers like XGBoost, Deep Nets can be used
- Works well even for large dimensions
- More info at [19]

## Discussion and Prior Work

- For CI test using classifiers, we need to modify a part of the samples in order to emulate i.i.d samples coming from $f^{CI}(x,y,z)$.
- $Y$ column can no longer be shuffled randomly, instead the operation on $Y$ column must depend on the $Z$ column.
- **(Prior Work)** Permutation of $Y$ column dependent on $Z$ column has been explored before (KCIPT [10]). However, KCIPT requires solving expensive LP, lacks strong theoretical guarantees and uses a kernel based method for two-sample testing. Other state of the art CI testing methods like KCIT [32], RCIT [28] are kernel based.
- **(Our Work)** The key idea is to use a nearest-neighbor based bootstrap procedure on a part of the total samples to create a data-set that *approximately* simulates i.i.d samples from $f^{CI}(x,y,z)$.
- A classifier is then used to distinguish between the bootstrapped samples and the original samples, similar to independence testing above.
- If the classifier is able to distinguish well, then $f \neq f^{CI}$ and $\mathcal{H}_0$ is rejected. If the classifier fails to distinguish, then we fail to reject $\mathcal{H}_0$.

## Model-Powered CI Test



$3n$ Original Samples

$X$ $Y$ $Z$ $x_1$ $y_1$ $z_1$ $x_{3n}$ $y_{3n}$ $z_{3n}$

$n$ Original Samples

$\mathcal{U}_1$ $X$ $Y$ $Z$ + $\ell$ 1 1

Shuffle

$2n$ Original Samples

Nearest Neighbor Bootstrap

$\mathcal{U}_2'$ $X$ $Y$ $Z$ + $\ell$ 0 0

$n$ samples close to $f^{CI}$

Training Set $\mathcal{D}_r$

$\mathcal{G}$

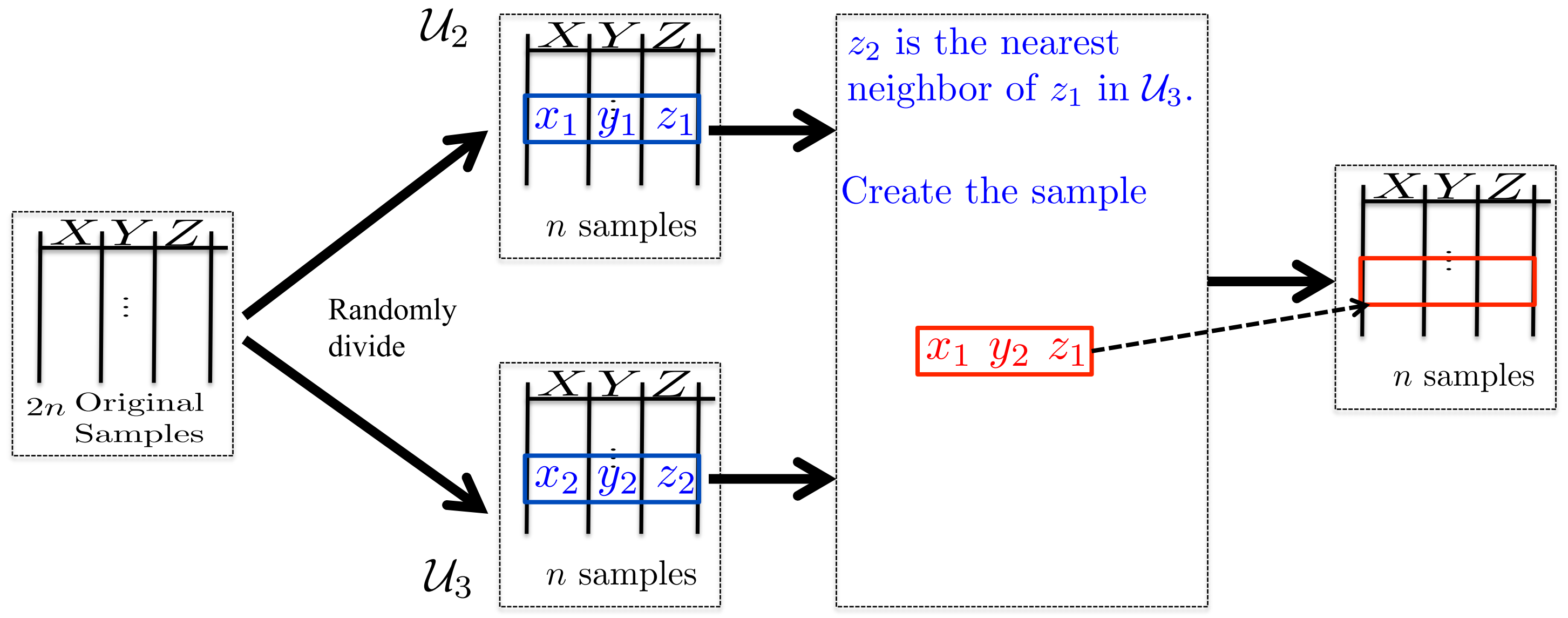
$\hat{g}$ (Trained Classifier)

Test Set $\mathcal{D}_e$

$L(\hat{g}, \mathcal{D}_e)$ (Test Error)

- If $L(\hat{g}, \mathcal{D}_e) > 0.5 + \tau$, then we reject $\mathcal{H}_0$ .
- Otherwise, we fail to reject $\mathcal{H}_0$ .

## Nearest-Neighbor Bootstrap



$2n$ Original Samples

Randomly divide

$\mathcal{U}_2$ $X$ $Y$ $Z$ $x_1$ $y_1$ $z_1$ $n$ samples

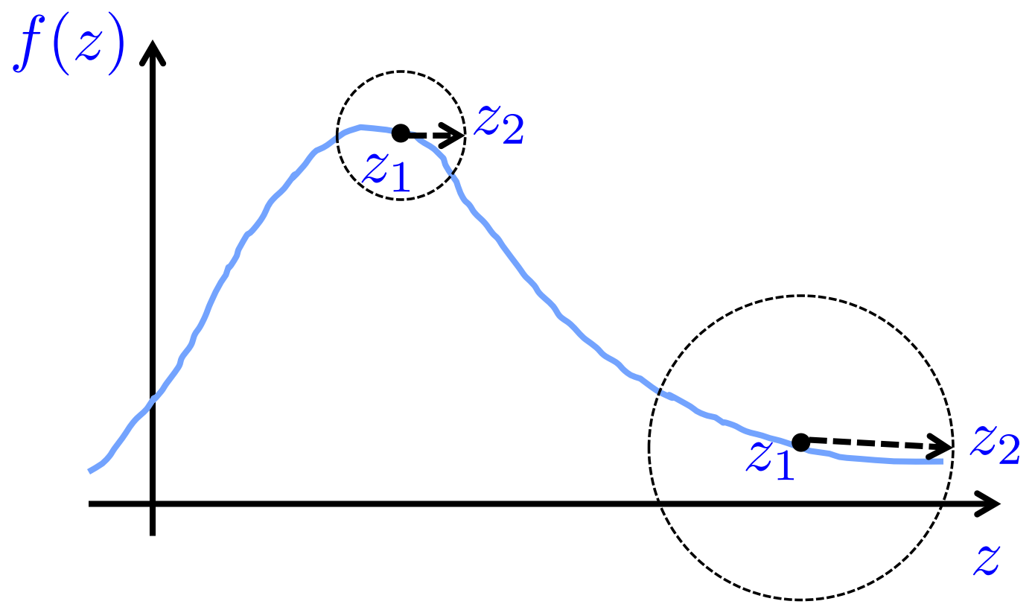
$\mathcal{U}_3$ $X$ $Y$ $Z$ $x_2$ $y_2$ $z_2$ $n$ samples

$z_2$ is the nearest neighbor of $z_1$ in $\mathcal{U}_3$.

Create the sample

$x_1$ $y_2$ $z_1$

$X$ $Y$ $Z$ $n$ samples



**Intuition:**

- If $z_2$ is close to $z_1$ then $f(y|z_1) \sim f(y|z_2)$ under smoothness assumptions.
- When $z_1$ is such that $f(z_1)$ is high, then $z_2$ is close to $z_1$. If $f(z_1)$ is low, then $z_1$ is a rare point and does not matter much.

**Assumptions and Result:**

- For $z \in \mathbb{R}^{d_z}$, $a$ such that $\|a - z\|_2 \leq \epsilon_1$, the generalized curvature matrix $\mathbf{I}_a(z)$ is,

$$\mathbf{I}_a(z)_{ij} = \left( \frac{\partial^2}{\partial z_i' \partial z_j'} \int \log \frac{f(y|z)}{f(y|z')} f(y|z) dy \right) \bigg|_{z'=a} = \mathbb{E}\left[ - \frac{\delta^2 \log f(y|z')}{\delta z_i' \delta z_j'} \bigg|_{z'=a} \bigg| Z = z \right]$$

  Assume for all $z \in \mathbb{R}^{d_z}$ and all $a$ such that $\|a - z\|_2 \leq \epsilon_1$, $\lambda_{max}\left(\mathbf{I}_a(z)\right) \leq \beta$.

- The probability density function $f(z)$ satisfies the following:
  (1) $f(z)$ is twice continuously differentiable and the Hessian matrix $H_f$ satisfies $\|H_f(z)\|_2 \leq c_{d_z}$ almost everywhere, where $c_{d_z}$ is only dependent on the dimension.
  (2) $\int f(z)^{1-1/d} dz \leq c_3, \; \forall d \geq 2$ where $c_3$ is a constant.

$$\text{Main Result: } d_{TV}(f^{CI}, \Phi) \leq b(n) \triangleq \mathcal{O}\left(\frac{1}{n^{1/d_z}}\right) + G(2c_{d_z}\epsilon_1^2)$$

Here $\Phi(x,y,z)$ is the marginal distribution of one sample in the bootstrapped data-set, and $G(\delta) = \mathbb{P}(f(Z) \leq \delta)$.

**References:** The references in this poster follow the indexing in the arxiv version of our work [0] (accepted for publication in NIPS 2017).

[0] Sen, R., Suresh, A.T., Shanmugam, K., Dimakis, A. G., & Shakkottai, S. (2017). Model-Powered Conditional Independence Test. arXiv preprint arXiv:1709.06138.

Python Package: https://github.com/rajatsen91/CCIT [pip install CCIT]

## Classification Guarantees

- The marginal distributions of the classes in the classification problem that needs to be solved is given by,

$$q(x,y,z|\ell = 1) = f_{X,Y,Z}(x,y,z) = \begin{cases} f^{CI}(x,y,z) & \text{if } \mathcal{H}_0 \text{ holds} \\ \neq f^{CI}(x,y,z) & \text{if } \mathcal{H}_1 \text{ holds} \end{cases}$$
$$q(x,y,z|\ell = 0) = \phi_{X,Y,Z}(x,y,z)$$

- However, the samples with label 0 are not i.i.d because of the nearest-neighbor bootstrap. We show that the samples are near i.i.d in a spatial sense and classification guarantees still hold under this setting.
- Let $\mathcal{G}$ be the class of classifying functions with VC dimension $V$ . Let the risk of a classifier we defined as $R_q(g) = \mathbb{E}_{(u,l)\sim q}\left[\mathbf{1}_{g(u)\neq l}\right]$. The optimal classifier in the class is $g_q^* = \text{argmin}_{g \in \mathcal{G}} R_q(g)$.
- Main Result: Suppose the class of classifying functions is such that $R_q(g_q^*) \leq r_0 + \eta$. Here, $r_0 \triangleq 0.5(1 - d_{TV}(q(x,y,z|1), q(x,y,z|0)))$ is the risk of the Bayes optimal classifier when $q(\ell = 1) = q(\ell = 0)$. This is the best loss that any classifier can achieve for this classification problem. Under this setting, w.p at least $1 - 8\delta$ we have:

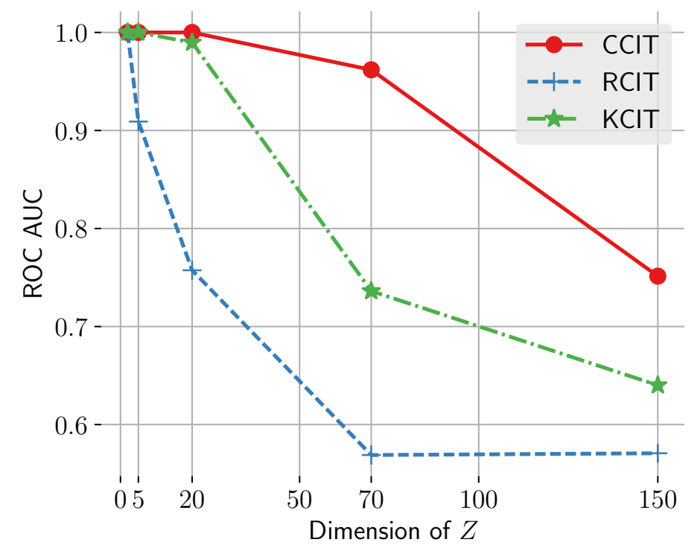
$$\frac{1}{2}\left(1 - d_{TV}(f, f^{CI})\right) - \frac{b(n)}{2} \leq R_q(\hat{g}) \leq \frac{1}{2}\left(1 - d_{TV}(f, f^{CI})\right) + \frac{b(n)}{2} + \eta + \gamma_n$$

$$\text{Here, } \gamma_n = \mathcal{O}\left(\sqrt{V}\left(n^{-1/3} + \sqrt{2^{d_z}/n}\right)\right)$$

## Bias-Correction and Choice of Threshold $\tau$

- In the case of finite samples and because of non-i.i.d'ness, the trained classifier $\hat{g}$ may be able to achieve a loss $L(\hat{g}, \mathcal{D}_e) = 0.5 - b$ , when null hypothesis holds. The bias $b$ can be corrected by training another classifier $\hat{g}'$ without using the X-coordinates. The loss of $\hat{g}'$ on the test set is expected to be $0.5 - b$ under both hypothesis, and therefore can be subtracted. More details in our paper [0].
- Under null hypothesis by VC theory, the risk of the classifier is a sub-gaussian random variable centered at 0.5 with variance $\mathcal{O}(1/\sqrt{n})$. Therefor, $\tau = 1/\sqrt{n}$ is a good choice of threshold. We also discuss a robust bootstrap method to choose the threshold in [0].

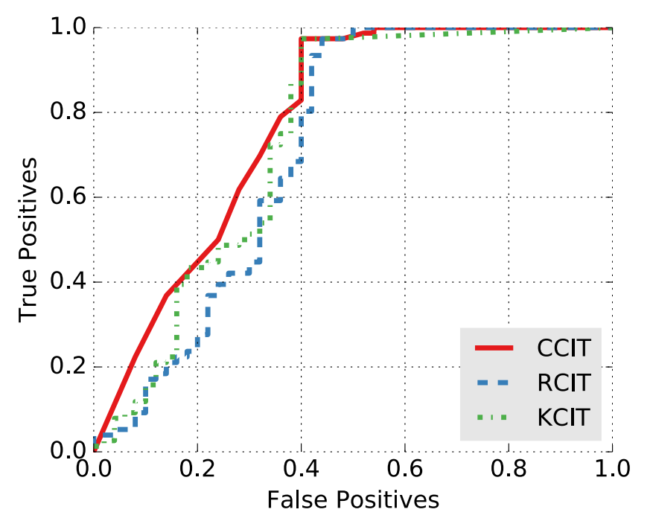
## Empirical Results



**Synthetic Experiments:** The experiments are performed in the post-nonlinear noise setting (popular in literature).

$$\mathcal{H}_0 : X = cos(a^T Z + \eta_1), \; Y = cos(b^T Z + \eta_2)$$
$$\mathcal{H}_1 : X = cos(a^T Z + \eta_1), \; Y = cos(b^T Z + cX + \eta_2)$$

The dimensions of X and Y are fixed at 1, while dimension of Z is varied. In this experiment, $n$ = *1000*. ROC AUC over 300 data-sets are used to generate each point in the plot, half of the data-sets being CI and vice-versa.



| Algo. | Graph $(i)$ | Graph $(ii)$ | Graph $(iii)$ |
|---|---|---|---|
| CCIT | 0.6848 | 0.7778 | 0.7156 |
| RCIT | 0.6448 | 0.7168 | 0.6928 |
| KCIT | 0.6528 | 0.7416 | 0.6610 |

**Flow-Cytometry Data [26]:** This data-set has observational and interventional data which gives expression levels of 11 proteins under various conditions.
The ground truth graph is not known with certainty. We use three graphs popularly accepted in literature [1(a,b,c) in [22] ], as the ground truth. Using these graphs as ground truth, we generate CI and non-CI relations that should hold between the variables, according to these graphs.

The ROC-AUC results are shown in the table on the left. We see that our algorithm outperforms KCIT and RCIT, on all three graphs as ground truth. ROC curve is shown on the left for the experiments considering one of the graphs [1(b) in [22] ] as ground-truth.