# *Learning* Statistical Property Testers

Sreeram Kannan
University of Washington Seattle

# Collaborators

Sudipto Mukherjee

Himanshu Asnani

Arman Rahimzamani

University of Washington, Seattle

Rajat Sen

UT, Austin

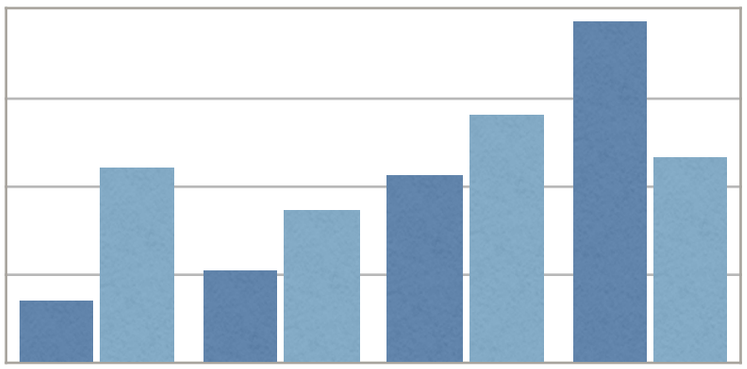
Karthikeyan Shanmugan

IBM Research

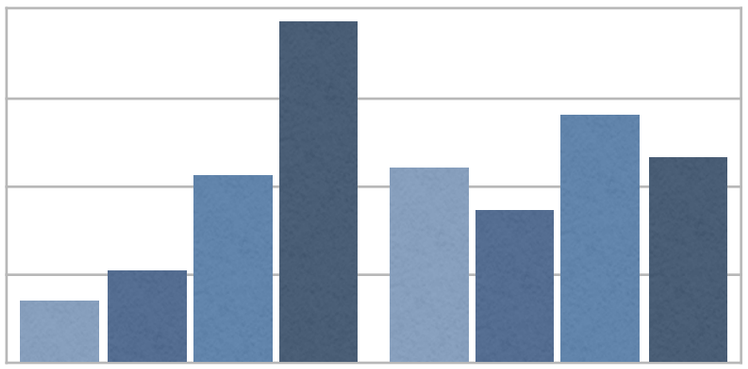
# Statistical Property Testing

- Closeness testing
- Independence testing
- Conditional Independence testing
- Information estimation

# Testing Total Variation Distance

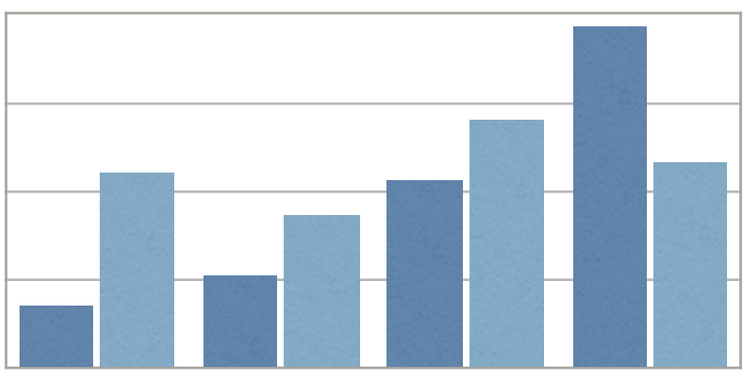


$P$

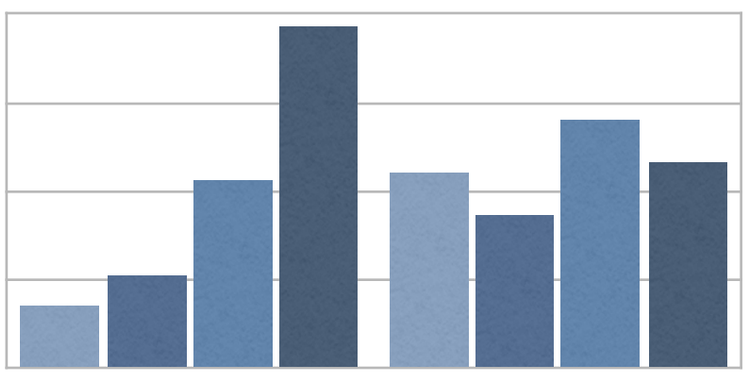


$Q$

# Testing Total Variation Distance



$P$

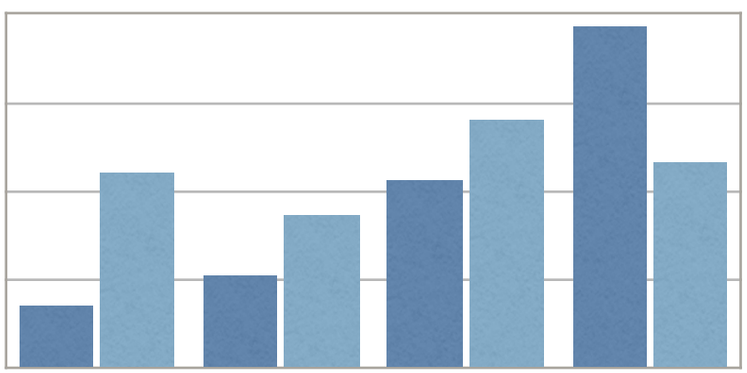


$Q$

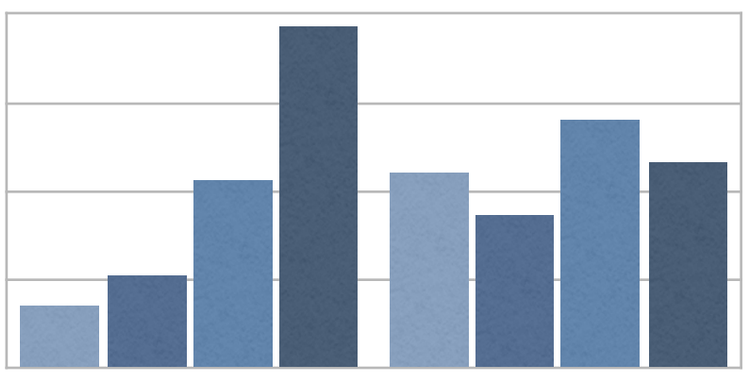
n samples

n samples

# Testing Total Variation Distance



$P$



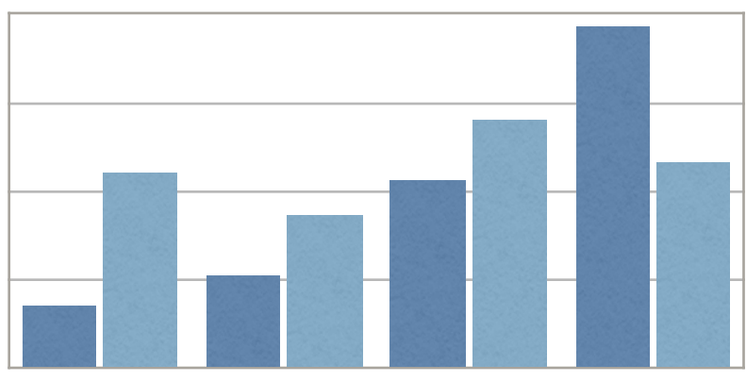
$Q$

n samples

n samples

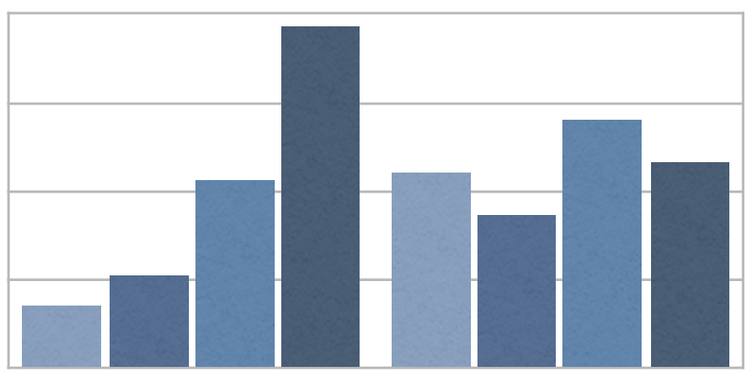
Estimate $D_{TV}(P, Q)$ ?

# Testing Total Variation Distance



$P$

n samples



$Q$

n samples

Estimate $D_{TV}(P, Q)$ ?

$P$ and $Q$ can be arbitrary.

Search beyond Traditional Density Estimation Methods

# Testing Total Variation: Prior Art

- Lots of work in CS theory on $D_{TV}$ testing
- Based on closeness testing between P and Q
- Sample complexity = $O(n^a)$, where n = alphabet size
- Curse of dimensionality if $n = 2^d$ Complexity is $O(2^{ad})$

* Chan et al, **Optimal Algorithms for testing closeness of discrete distributions**, *SODA 2014*

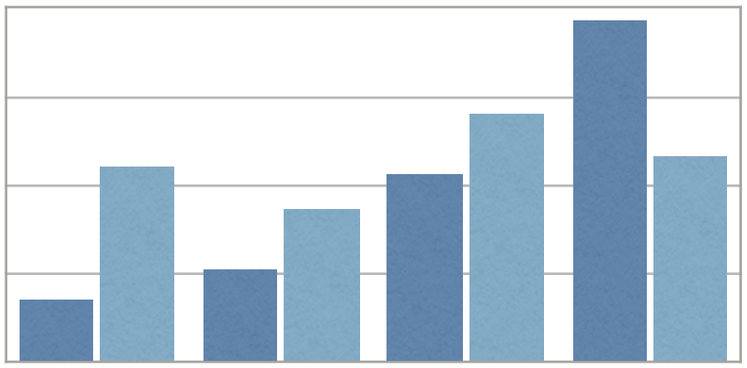
* Sriperumbudur et al, **Kernel choice and classifiability for RKHS embeddings of probability distributions**, *NIPS 2009*

# Classifiers beat curse-of-dimensionality

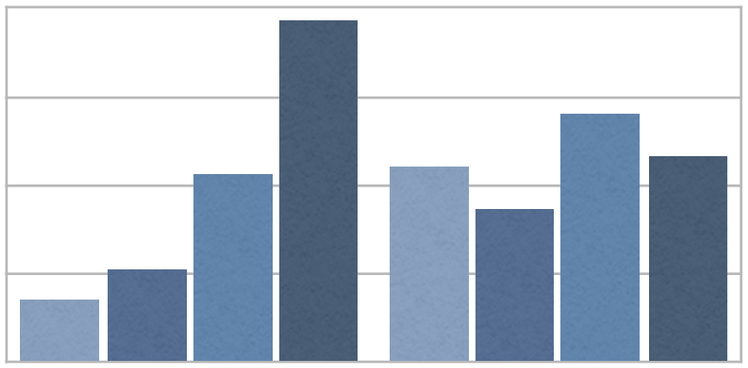
- Deep NN and boosted random forests achieve state-of-the-art performance
- Works very well even in practice when X is high dimensional.
- Exploits generic inductive bias:
  - Invariance
  - Hierarchical Structure
  - Symmetry


Theoretical guarantees lag severely behind practice!

# Distance Estimation via Classification

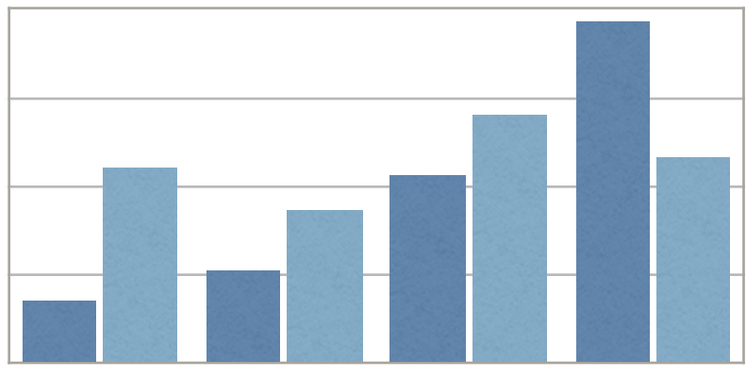


n samples $\sim$ P

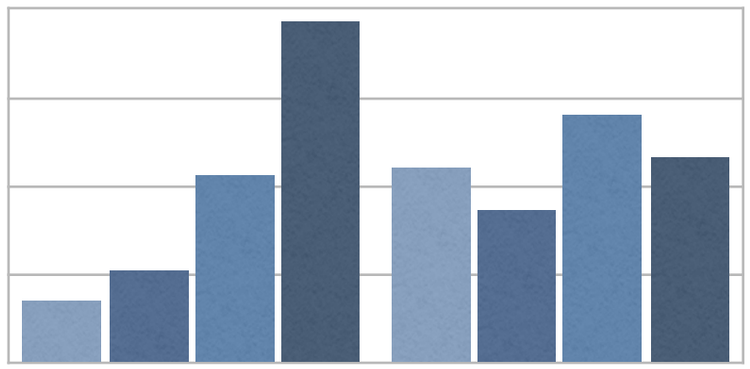


n samples $\sim$ Q

# Distance Estimation via Classification



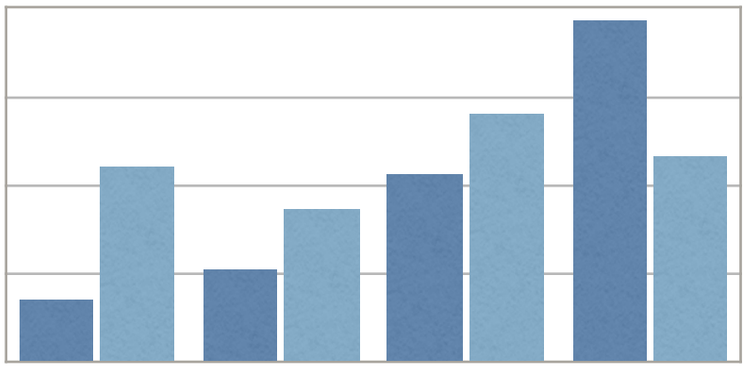
n samples $\sim$ P
(Label 0)



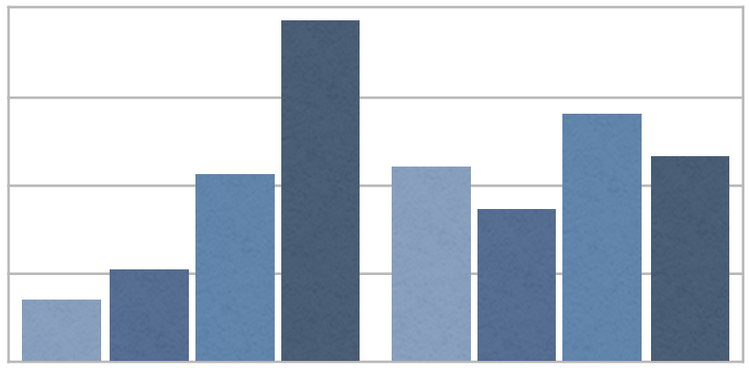
n samples $\sim$ Q
(Label 1)

Classifier

# Distance Estimation via Classification



n samples $\sim$ P
(Label 0)

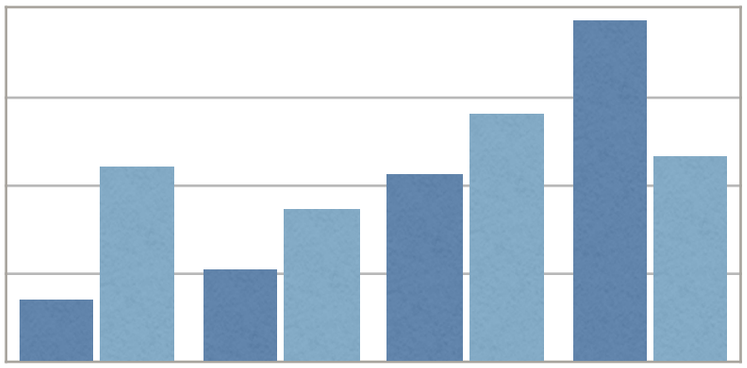


n samples $\sim$ Q
(Label 1)

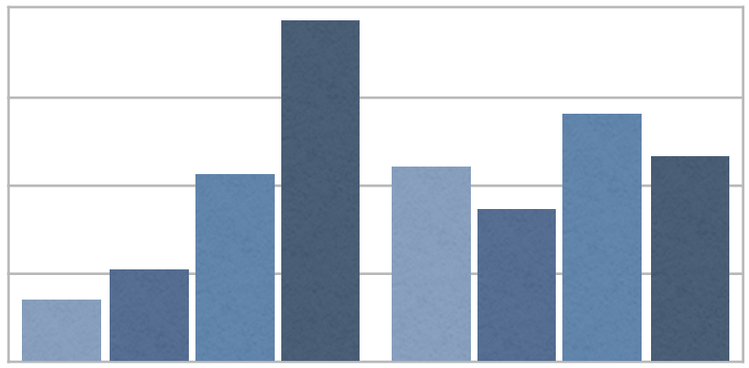
Classifier

Classification Error of Optimal Bayes Classifier $= \frac{1}{2} - \frac{1}{2} D_{\mathrm{TV}}(P, Q).$

# Distance Estimation via Classification



n samples $\sim$ P
(Label 0)



n samples $\sim$ Q
(Label 1)

Deep NN, Boosted Trees etc.

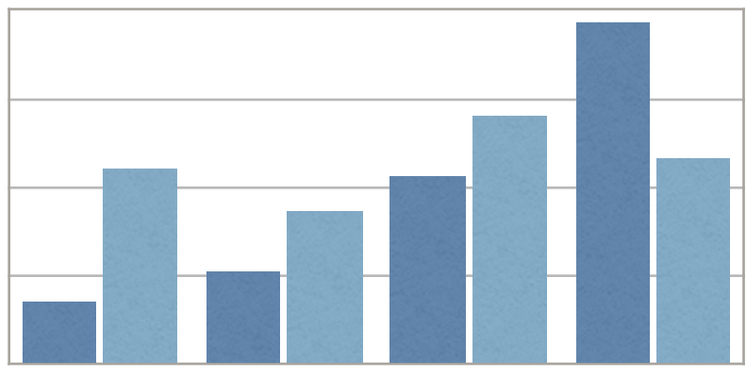
Classification Error of Optimal Classifier

$$= \frac{1}{2} - \frac{1}{2} D_{\mathrm{TV}}(P, Q).$$

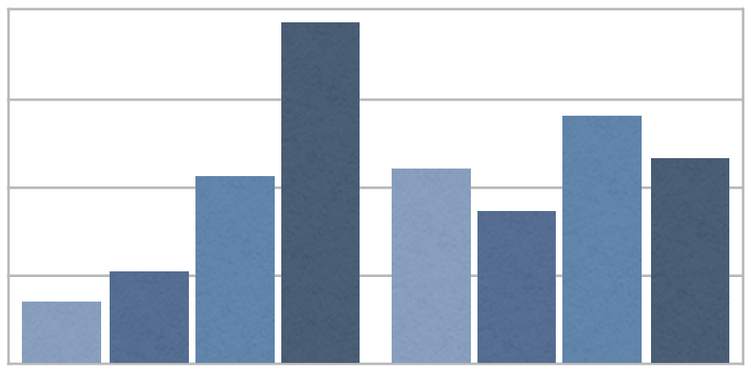
* Lopez-Paz et al, **Revisiting Classifier two-sample tests**, *ICLR 2017*

* Sriperumbudur et al, **Kernel choice and classifiability for RKHS embeddings of probability distributions**, *NIPS 2009*

# Distance Estimation via Classification



n samples $\sim$ P
(Label 0)



n samples $\sim$ Q
(Label 1)

Deep NN, Boosted Trees etc.

Classification Error of **Any** Classifier

$$>= \frac{1}{2} - \frac{1}{2} D_{\mathrm{TV}}(P, Q).$$

**Can get P-value control**

* Lopez-Paz et al, **Revisiting Classifier two-sample tests**, *ICLR 2017*

* Sriperumbudur et al, **Kernel choice and classifiability for RKHS embeddings of probability distributions**, *NIPS 2009*

# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

* Lopez-Paz et al, **Revisiting Classifier two-sample tests**, *ICLR 2017*

* Sriperumbudur et al, **Kernel choice and classifiability for RKHS embeddings of probability distributions**, *NIPS 2009*

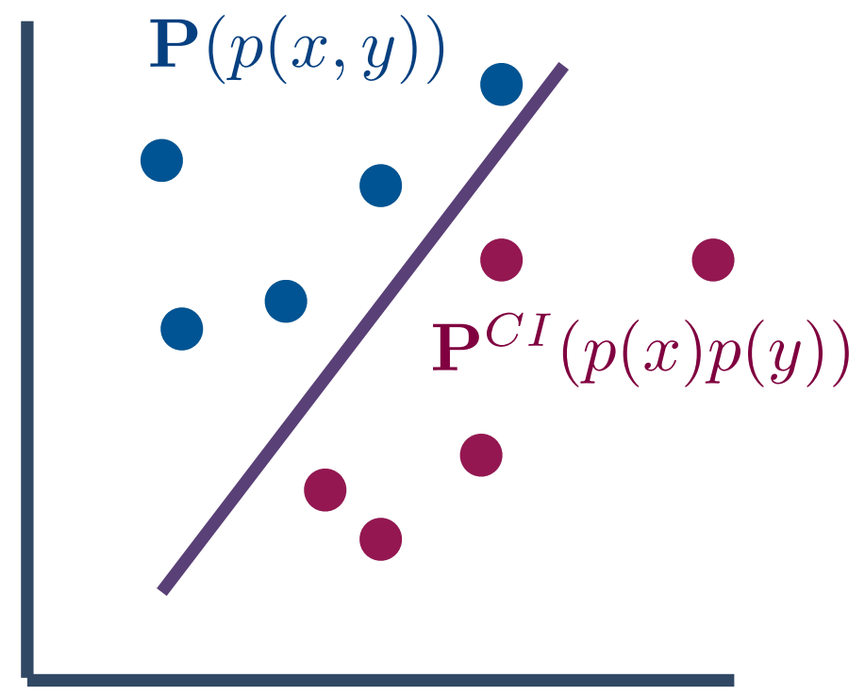
# Independence Testing

$$\text{n samples } \{x_i, y_i\}_{i=1}^{n} \begin{cases} \mathcal{H}_0 : X \perp\!\!\!\perp Y(\mathbf{P}^{CI}) \\ \\ \mathcal{H}_1 : X \not\perp\!\!\!\perp Y(\mathbf{P}) \end{cases}$$

# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

$$\begin{cases} \mathcal{H}_0 : X \perp\!\!\!\perp Y(\mathbf{P}^{CI}) \\ \mathcal{H}_1 : X \not\perp\!\!\!\perp Y(\mathbf{P}) \end{cases}$$




Classify

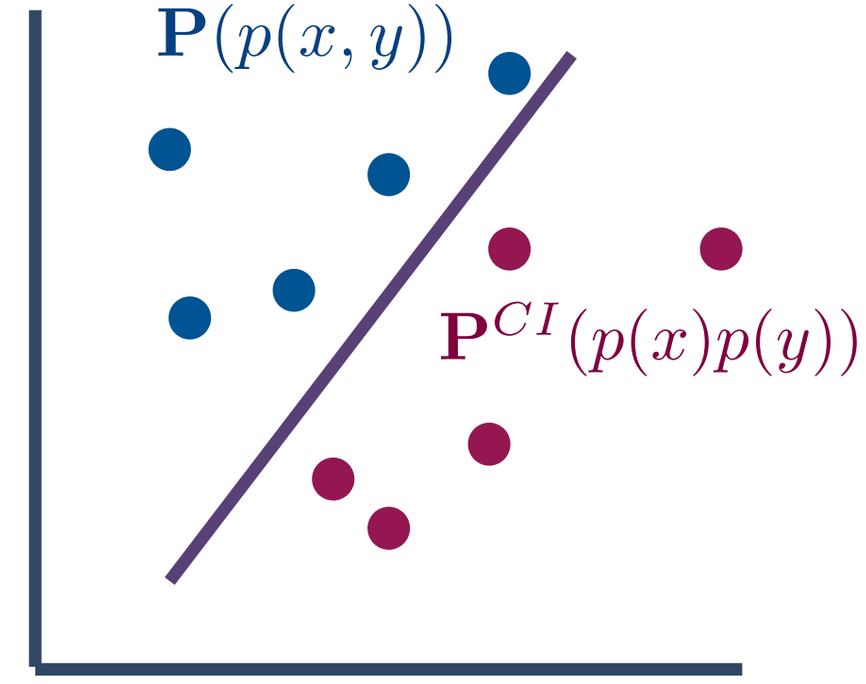
# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

$$\mathcal{H}_0 : X \perp\!\!\!\perp Y (\mathbf{P}^{CI})$$

$$\mathcal{H}_1 : X \not\perp\!\!\!\perp Y (\mathbf{P})$$

$\mathbf{P}^{CI}(p(x)p(y))$




Classify

# Independence Testing

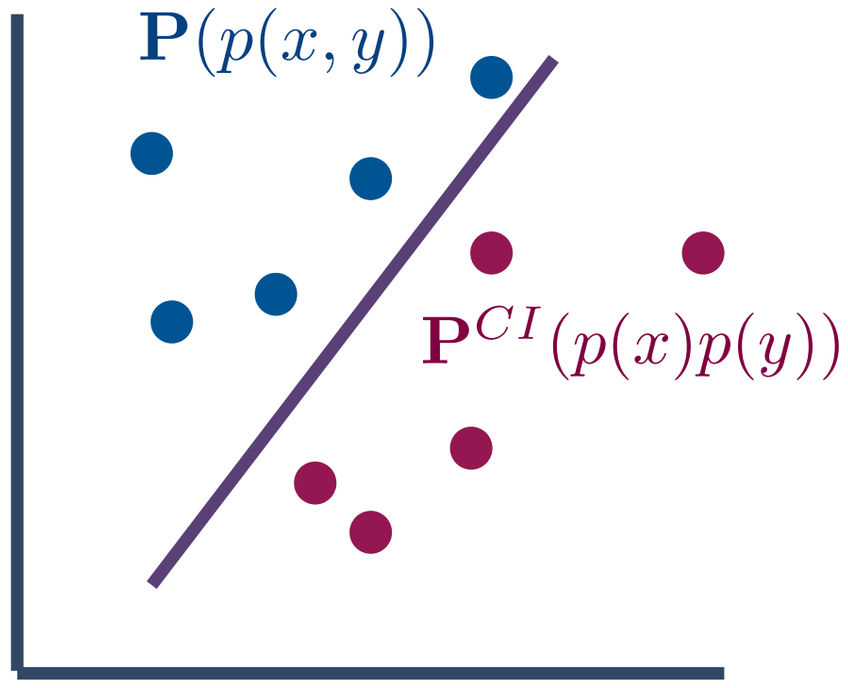
n samples $\{x_i, y_i\}_{i=1}^n$

$$\mathcal{H}_0 : X \perp\!\!\!\perp Y (\mathbf{P}^{CI})$$

$$\mathcal{H}_1 : X \not\perp\!\!\!\perp Y (\mathbf{P})$$

$\mathbf{P}^{CI}(p(x)p(y))$

Permutation




Classify

# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

Split

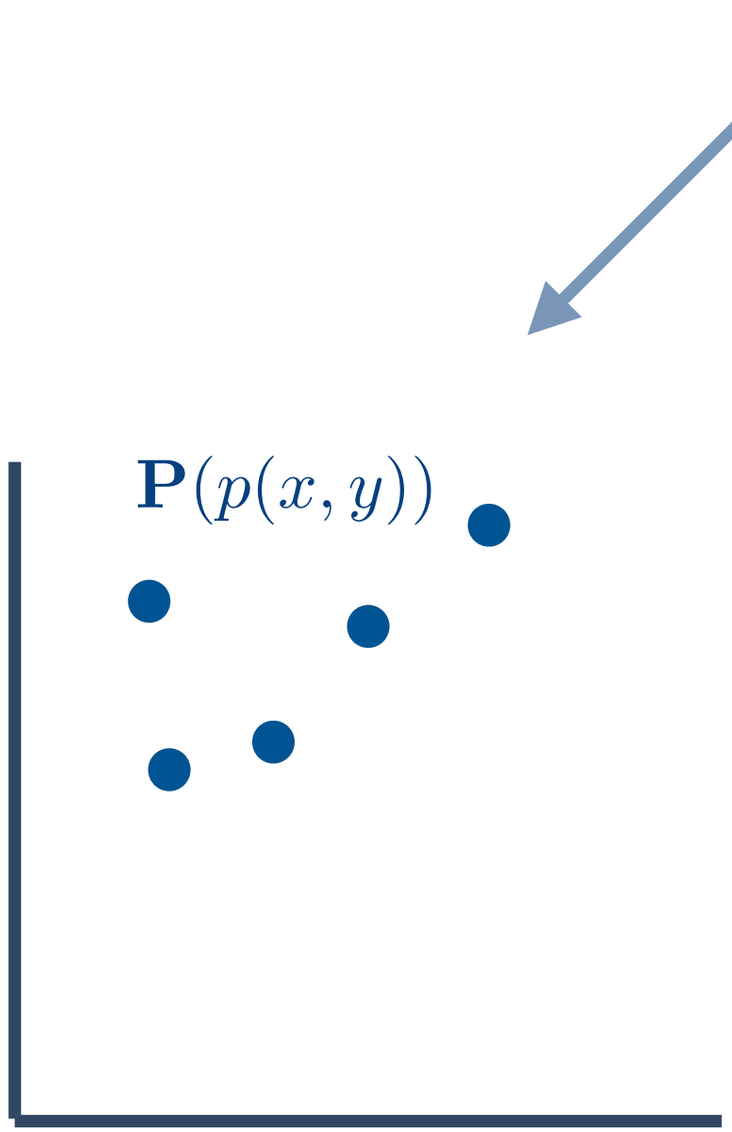
Equally

# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

Split

Equally

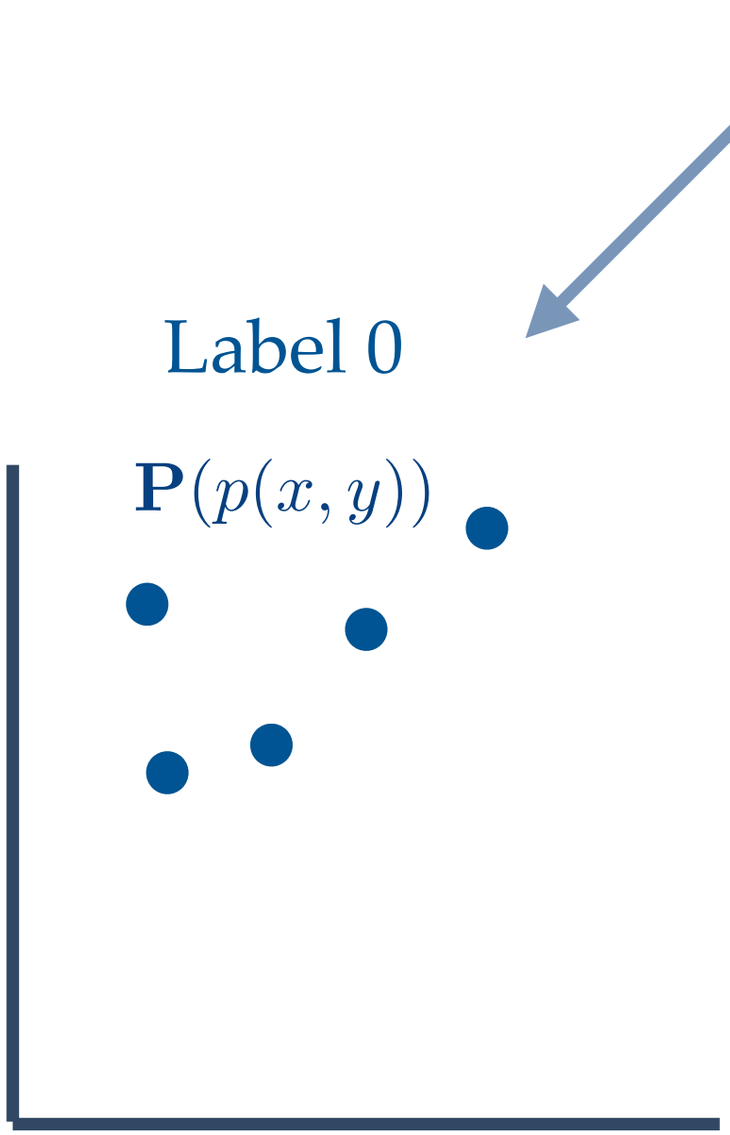


# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

Split

Equally

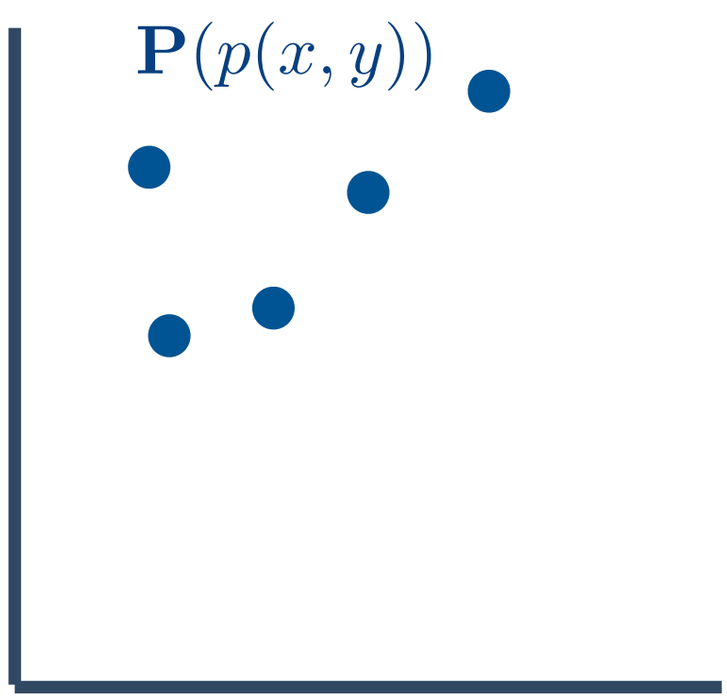


# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

Split Equally

Label 0




$y_i$'s are permuted

# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

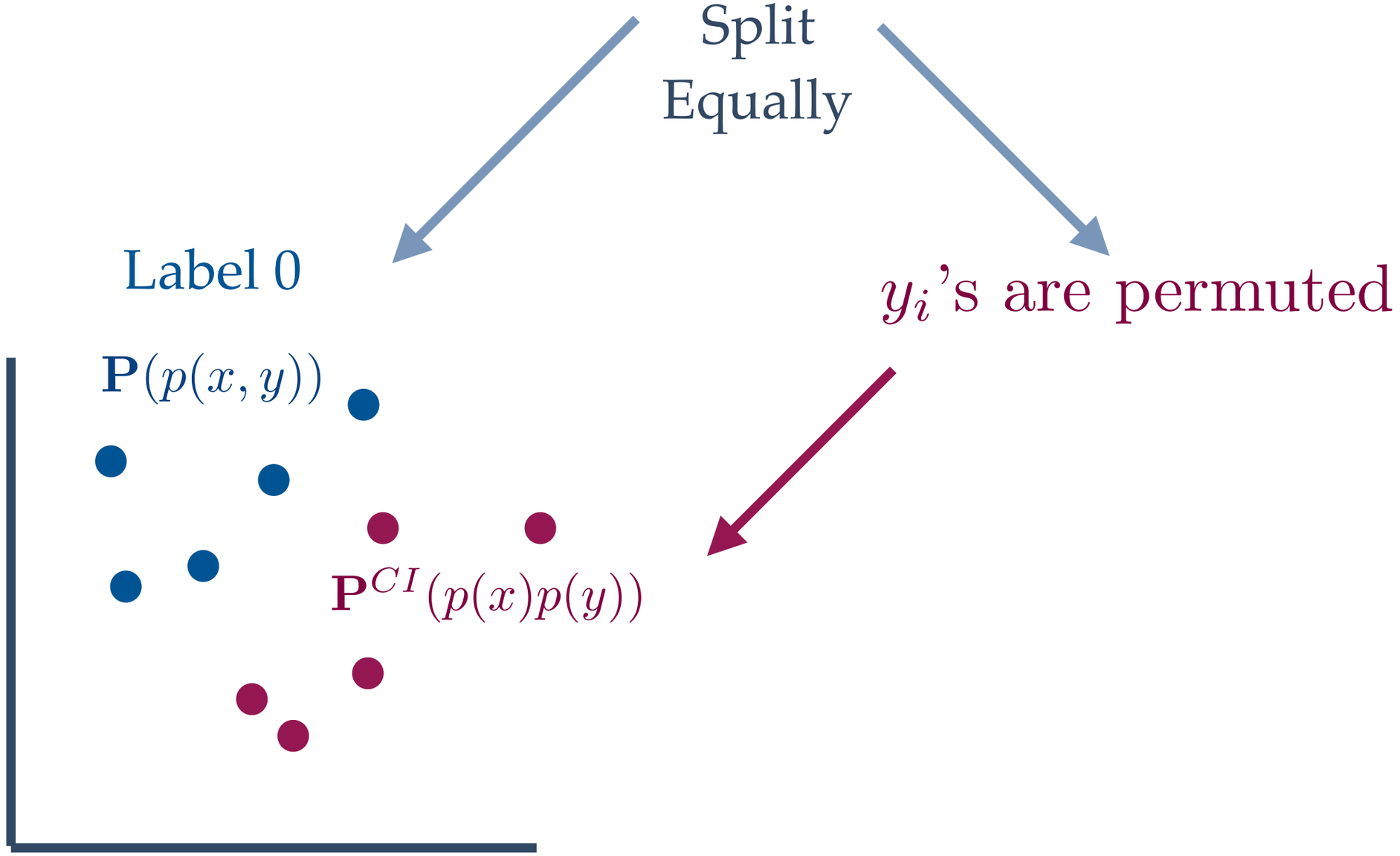
Split Equally

Label 0

$\mathbf{P}(p(x, y))$

$y_i$'s are permuted

$\mathbf{P}^{CI}(p(x)p(y))$



# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

Split Equally

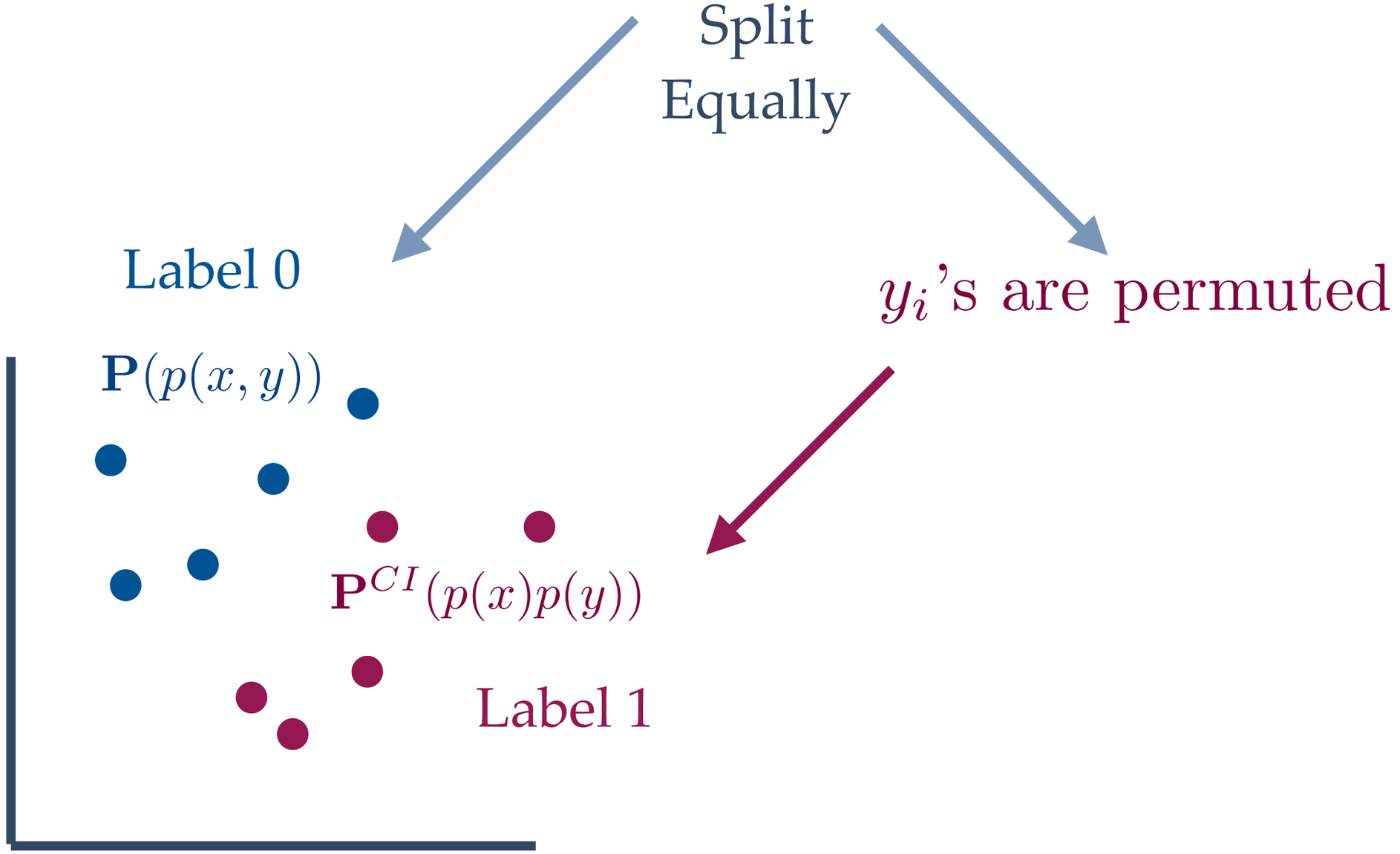
Label 0

$\mathbf{P}(p(x, y))$

$y_i$'s are permuted

$\mathbf{P}^{CI}(p(x)p(y))$

Label 1



# Independence Testing

n samples $\{x_i, y_i\}_{i=1}^n$

Split Equally

Label 0

$\mathbf{P}(p(x, y))$

$\mathbf{P}^{CI}(p(x)p(y))$

Label 1

$y_i$'s are permuted

P-value control

*Lopez-Paz et al, **Revisiting Classifier two-sample tests**, *ICLR 2017*

* Sriperumbudur et al, **Kernel choice and classifiability for RKHS embeddings of probability distributions,** *NIPS 2009*

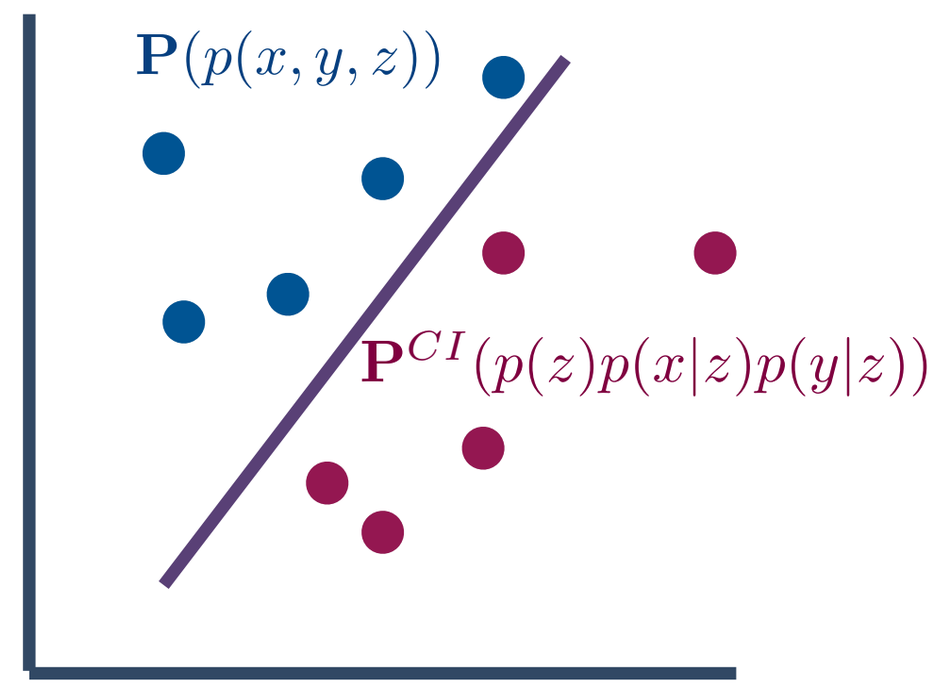
# Conditional Independence Testing

$$\text{n samples } \{x_i, y_i, z_i\}_{i=1}^{n} \begin{cases} \mathcal{H}_0 : X \perp\!\!\!\perp Y | Z \ (\mathbf{P}^{CI}) \\ \text{vs} \\ \mathcal{H}_1 : X \not\perp\!\!\!\perp Y | Z \ (\mathbf{P}) \end{cases}$$

# Conditional Independence Testing

n samples $\{x_i, y_i, z_i\}_{i=1}^n$

$$\begin{cases} \mathcal{H}_0 : X \perp\!\!\!\perp Y|Z \ (\mathbf{P}^{CI}) \\ \text{vs} \\ \mathcal{H}_1 : X \not\perp\!\!\!\perp Y|Z \ (\mathbf{P}) \end{cases}$$




Classify

# Conditional Independence Testing

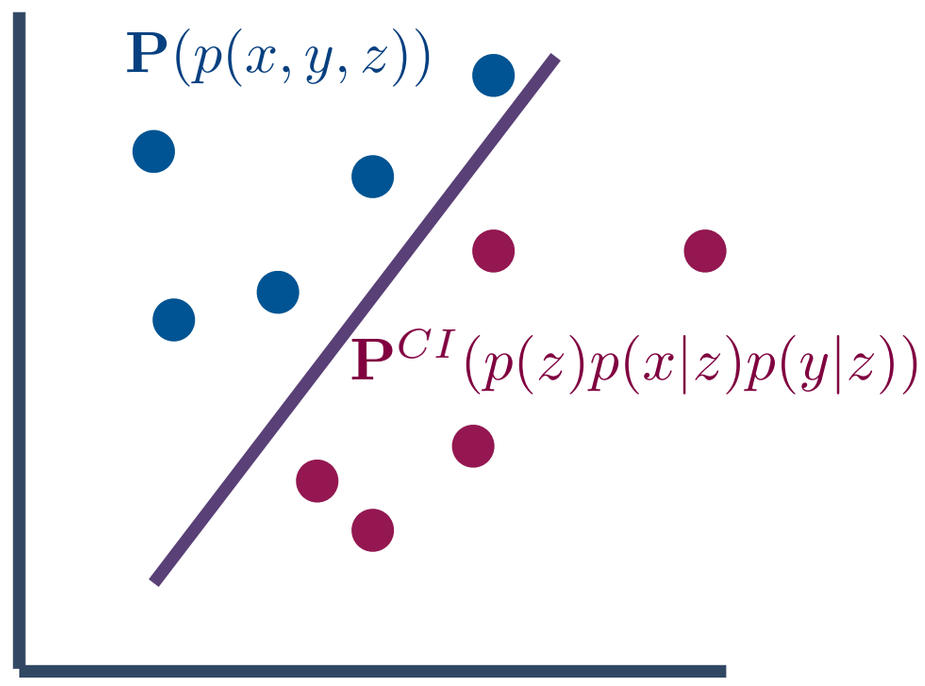
n samples $\{x_i, y_i, z_i\}_{i=1}^n$

$$\mathcal{H}_0 : X \perp\!\!\!\perp Y|Z \ (\mathbf{P}^{CI})$$

vs

$$\mathcal{H}_1 : X \not\perp\!\!\!\perp Y|Z \ (\mathbf{P})$$

How to get $\mathbf{P}^{CI}(p(z)p(x|z)p(y|z)$?




Classify

# Conditional Independence Testing

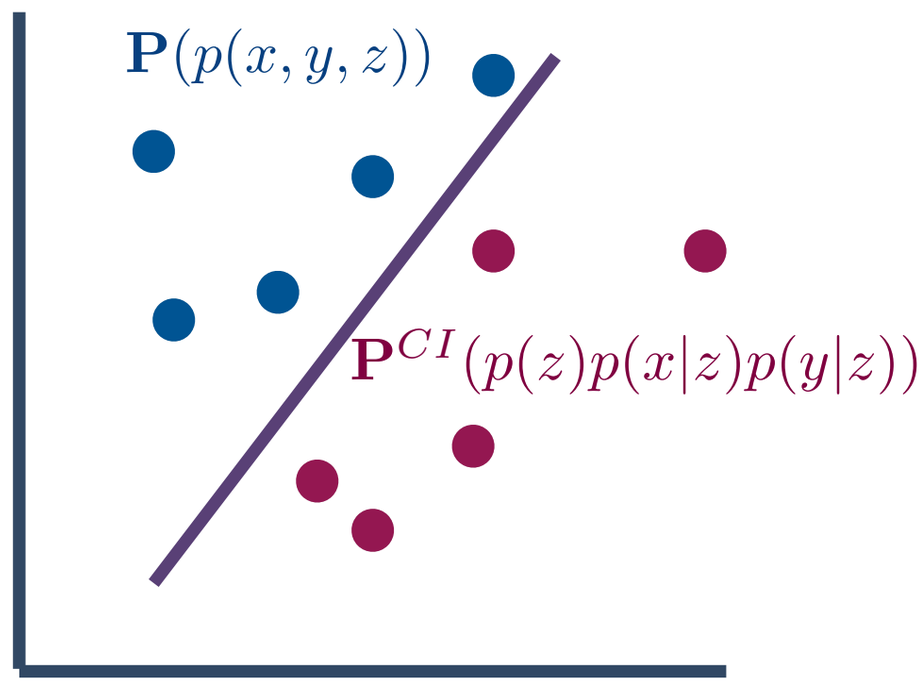
n samples $\{x_i, y_i, z_i\}_{i=1}^n$

$$\mathcal{H}_0 : X \perp\!\!\!\perp Y|Z \ (\mathbf{P}^{CI})$$

vs

$$\mathcal{H}_1 : X \not\perp\!\!\!\perp Y|Z \ (\mathbf{P})$$

Given samples $\sim p(x, z)$
How to emulate $p(y|z)$?




Classify

# Conditional Independence Testing

n samples $\{x_i, y_i, z_i\}_{i=1}^n$

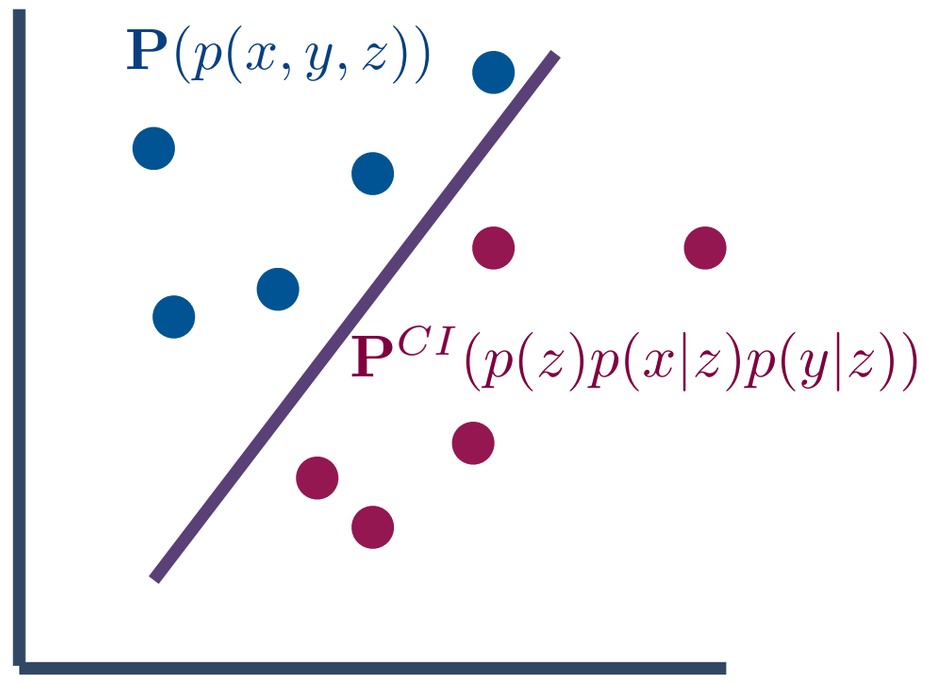
$$\mathcal{H}_0 : X \perp\!\!\!\perp Y|Z \ (\mathbf{P}^{CI})$$

vs

$$\mathcal{H}_1 : X \not\perp\!\!\!\perp Y|Z \ (\mathbf{P})$$

## Emulate $p(y|z)$ as $q(y|z)$

- KNN Based Methods
- Kernel Methods




Classify

# Conditional Independence Testing

n samples $\{x_i, y_i, z_i\}_{i=1}^n$

$$\mathcal{H}_0 : X \perp\!\!\!\perp Y|Z \ (\mathbf{P}^{CI})$$

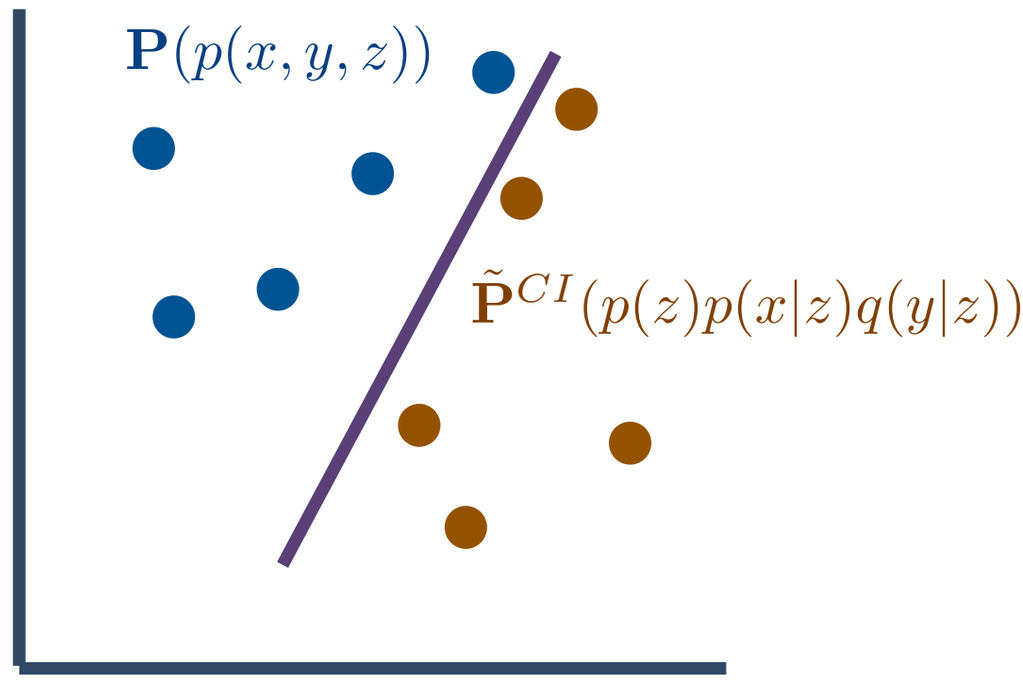
vs

$$\mathcal{H}_1 : X \not\perp\!\!\!\perp Y|Z \ (\mathbf{P})$$

Emulate $p(y|z)$ as $q(y|z)$

- KNN Based Methods
- Kernel Methods

$\tilde{\mathbf{P}}^{CI}(p(z)p(x|z)q(y|z))$




Classify

# Conditional Independence Testing

n samples $\{x_i, y_i, z_i\}_{i=1}^n$

$$\mathcal{H}_0 : X \perp\!\!\!\perp Y|Z \ (\mathbf{P}^{CI})$$

vs

$$\mathcal{H}_1 : X \not\perp\!\!\!\perp Y|Z \ (\mathbf{P})$$

- [KCIT] Gretton et al, **Kernel-based conditional independence test and application in causal discovery,** *NIPS 2008*
- [KCIPT] Doran et al, **A permutation-based kernel conditional independence test,** *UAI 2014*
- [CCIT] Sen et al, **Model-Powered Conditional Independence Test,** *NIPS 2017*
- [RCIT] Strobl et al, **Approximate Kernel-based Conditional Independence Tests for Fast Non-Parametric Causal Discovery,** *arXiv*

Emulate $p(y|z)$ as $q(y|z)$

- KNN Based Methods
- Kernel Methods

$\mathbf{P}(p(x,y,z))$

$\mathbf{P}^{CI}(p(z)p(x|z)q(y|z))$

$\mathbf{P}^{CI}(p(z)p(x|z)q(y|z))$

Classify

# Conditional Independence Testing

n samples $\{x_i, y_i, z_i\}_{i=1}^n$

$$\mathcal{H}_0 : X \perp\!\!\!\perp Y|Z \ (\mathbf{P}^{CI})$$

vs

$$\mathcal{H}_1 : X \not\perp\!\!\!\perp Y|Z \ (\mathbf{P})$$

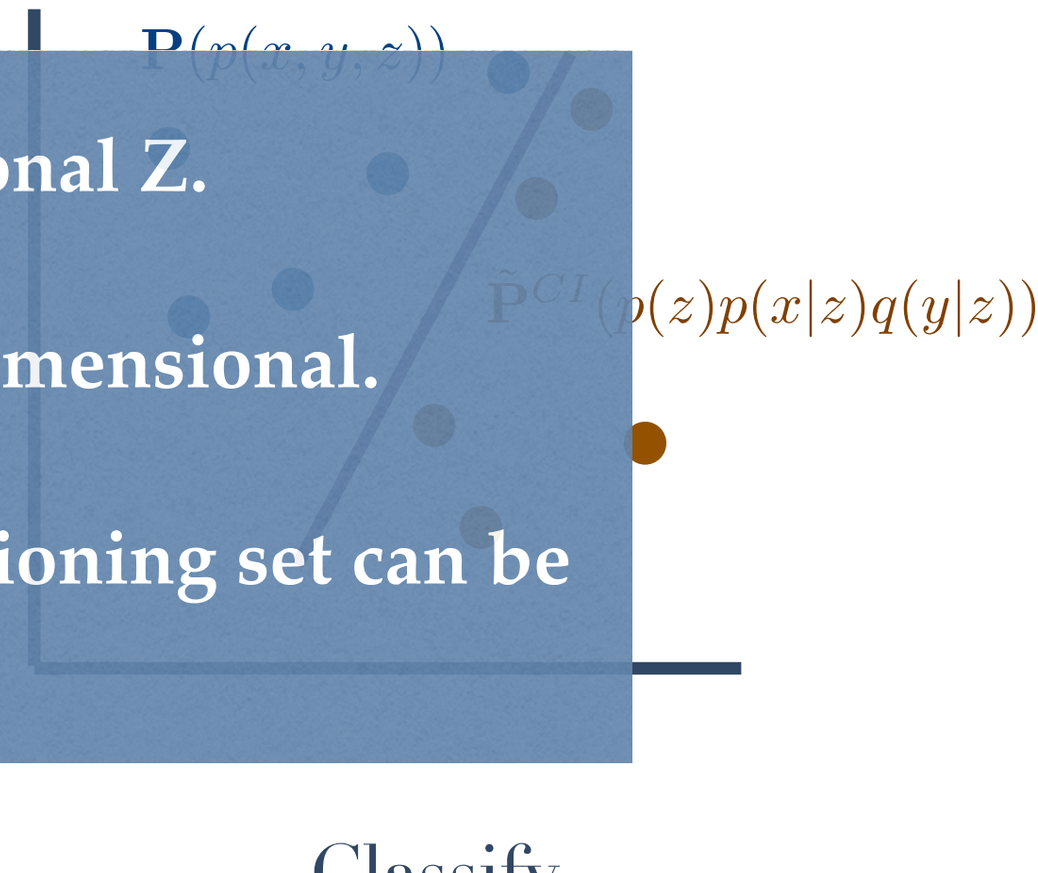
Emulate $p(y|z)$ as $q(y|z)$

- KNN Based Methods
- Kernel Based Methods

**Limited to low-dimensional Z.**

**In practice, Z is often high dimensional.**

**(Eg. In graphical model, conditioning set can be entire graph.)**




Classify

# Generative Models beat curse-of-dimensionality

# Generative Models beat curse-of-dimensionality


- Trained Real Samples of x
- Can generate any number of new samples

# Generative Models beat curse-of-dimensionality


- Trained Real Samples of x

- Can generate any number of new samples

How loose can the estimate be for $\tilde{\mathbf{P}}^{CI}$ or $q(y|z)$?

How loose can the estimate be for $\tilde{\mathbf{P}}^{CI}$ or $q(y|z)$?

As long as the density function $q(\mathbf{y}|\mathbf{z}) > 0$ whenever $p(\mathbf{y}, \mathbf{z}) > 0$.

***Mimic-and-Classify works***

How loose can the estimate be for $\tilde{\mathbf{P}}^{CI}$ or $q(y|z)$?

***Novel Bias Cancellation Method in Mimic-and-Classify works***

As long as the density function $q(\mathbf{y}|\mathbf{z}) > 0$ whenever $p(\mathbf{y}, \mathbf{z}) > 0$.

Mimic Functions : GANs, Regressors etc.

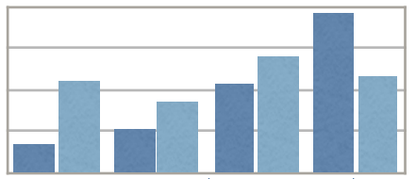
# Mimic and Classify

Mimic Step

Classify Step

# Mimic and Classify

Mimic Step

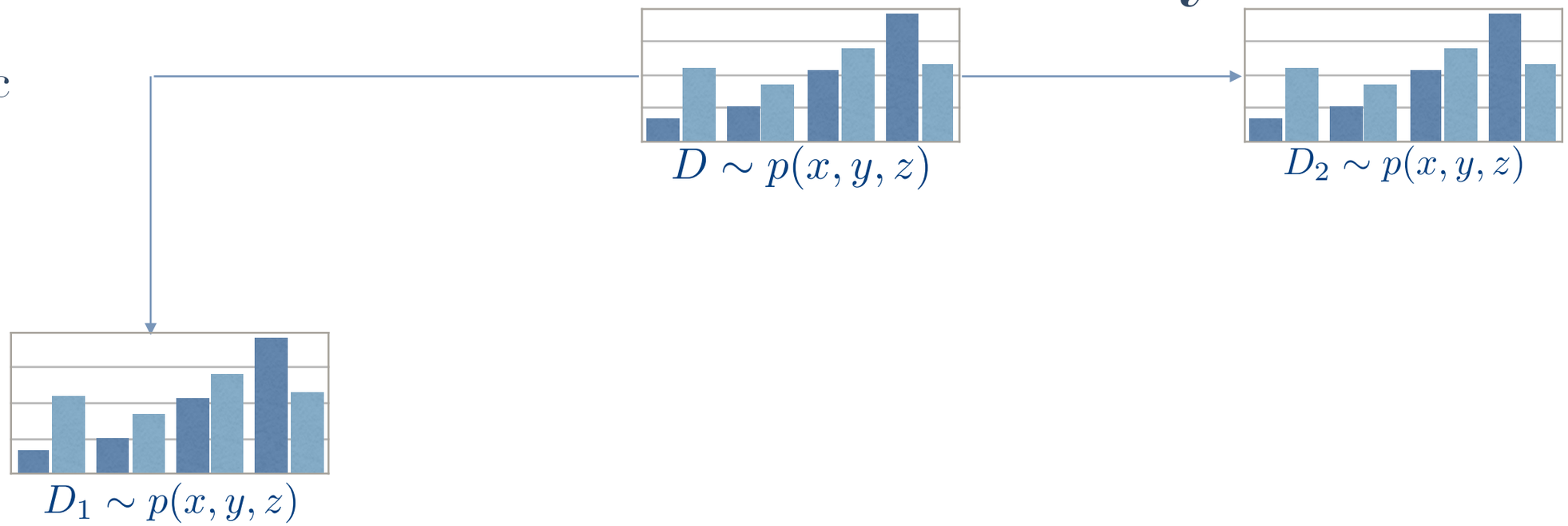


$D \sim p(x, y, z)$

Classify Step

# Mimic and Classify

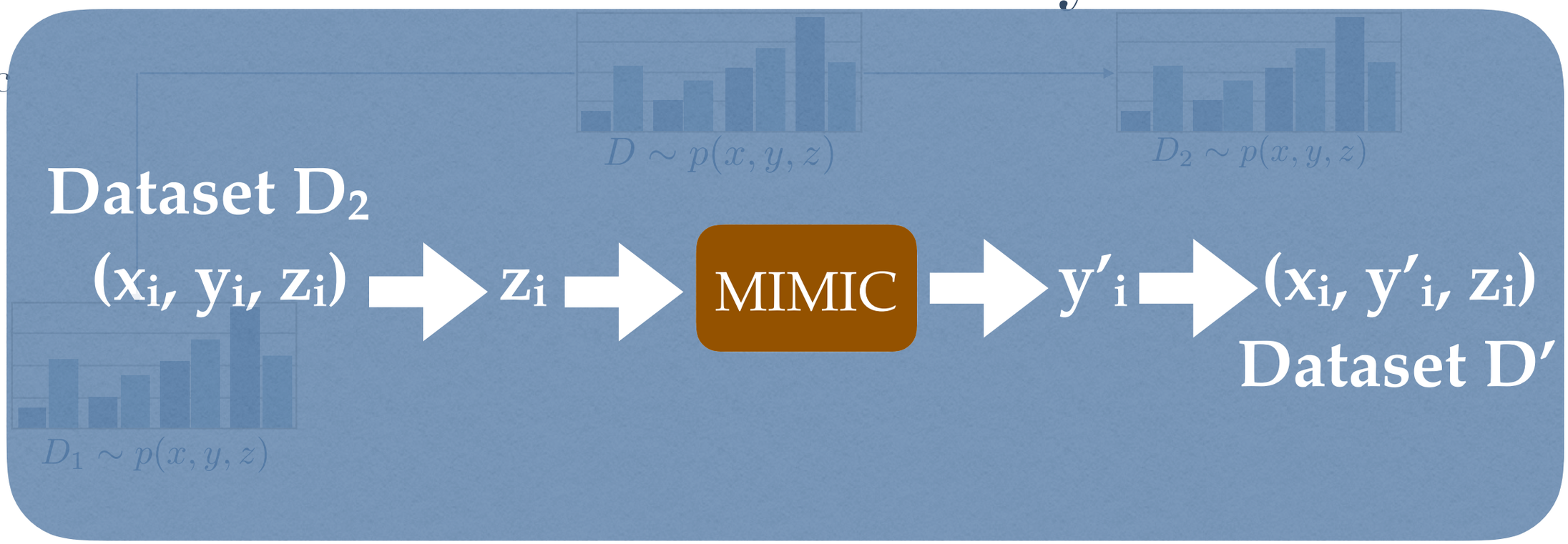
Mimic Step




Classify Step

# Mimic and Classify

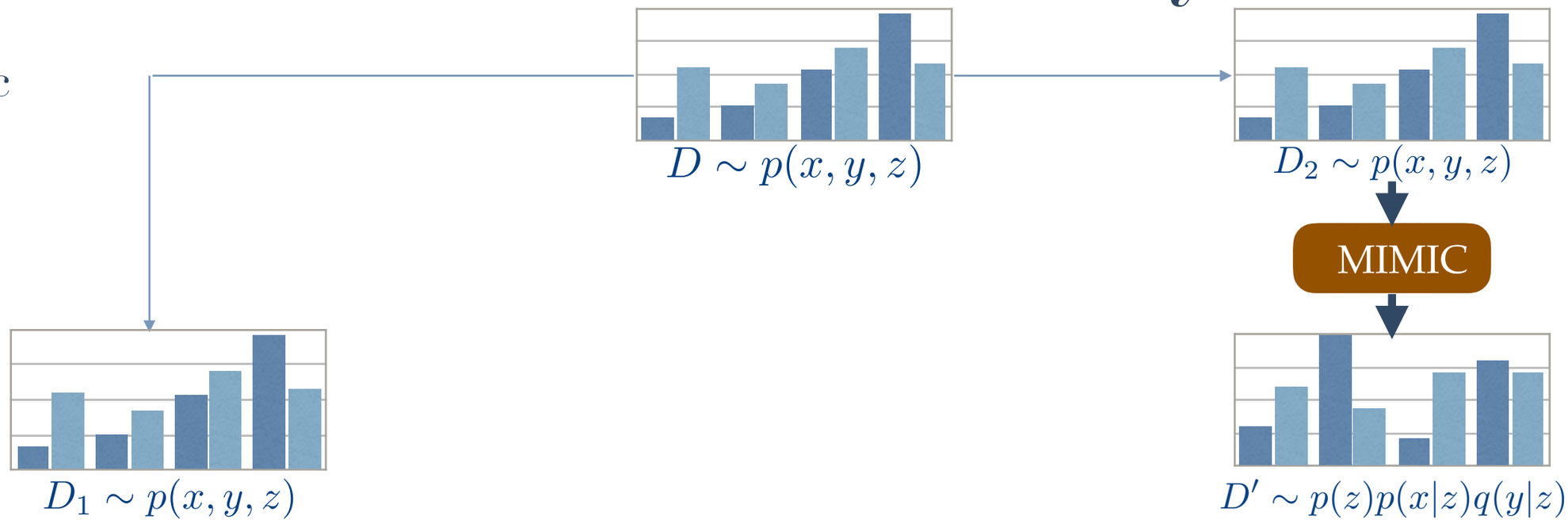
Mimic Step




Classify Step

# Mimic and Classify

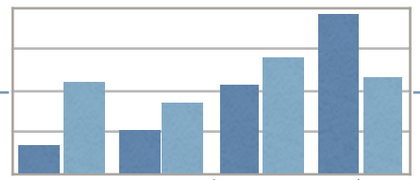
Mimic Step



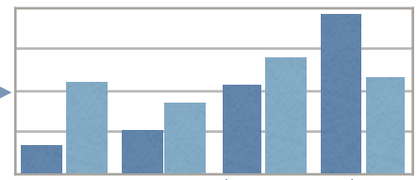

Classify Step

# Mimic and Classify

Mimic Step

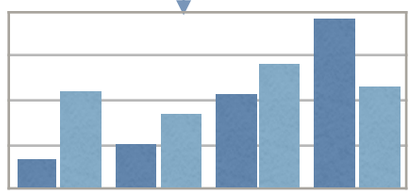


$D \sim p(x, y, z)$



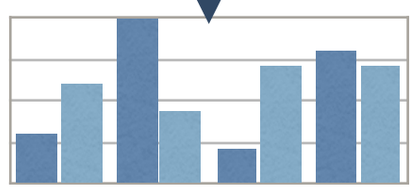
$D_2 \sim p(x, y, z)$

MIMIC



$D_1 \sim p(x, y, z)$

**(Label 0)**



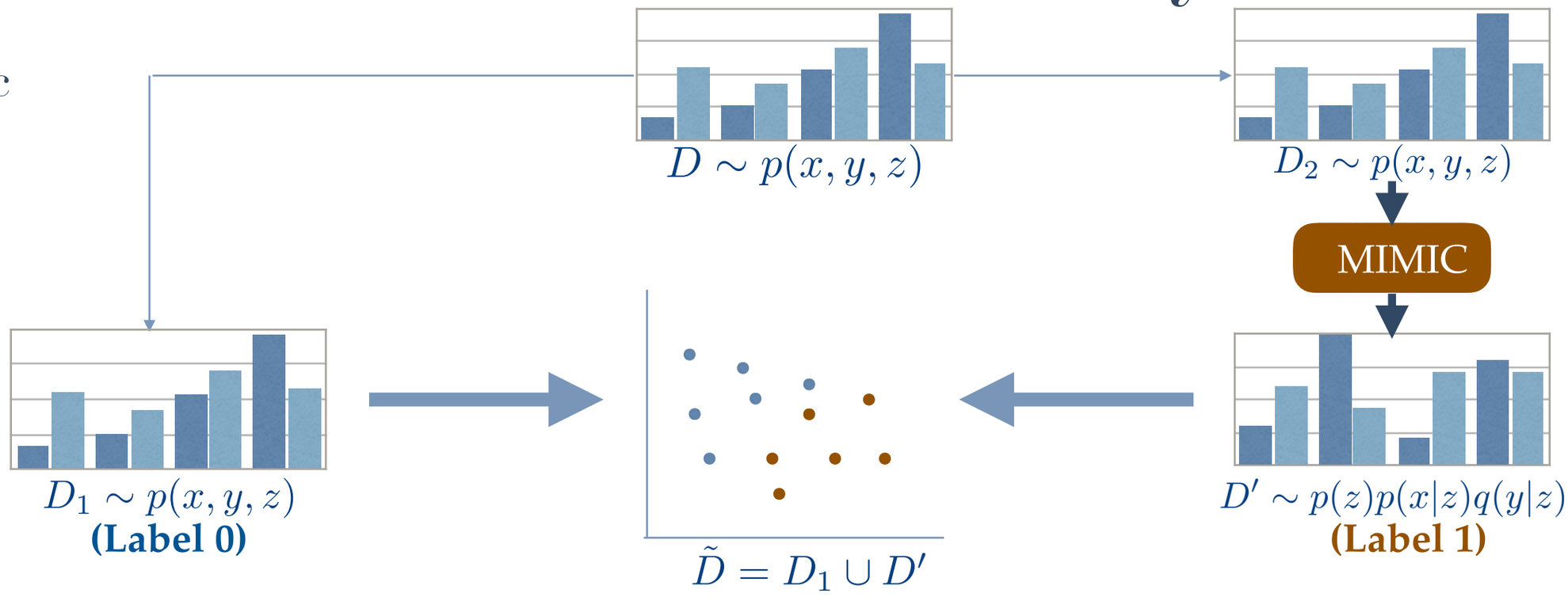
$D' \sim p(z)p(x|z)q(y|z)$

**(Label 1)**

Classify Step

# Mimic and Classify

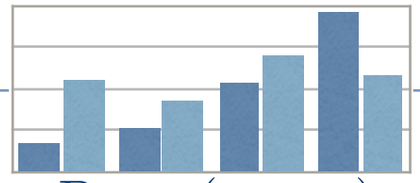
Mimic Step



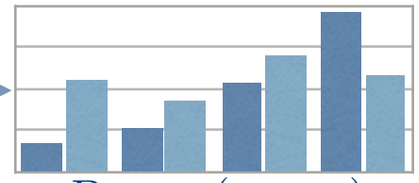

Classify Step

# Mimic and Classify

Mimic Step

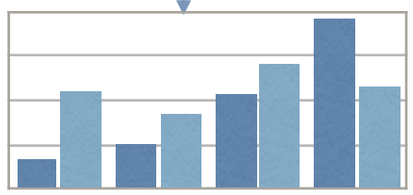


$D \sim p(x, y, z)$



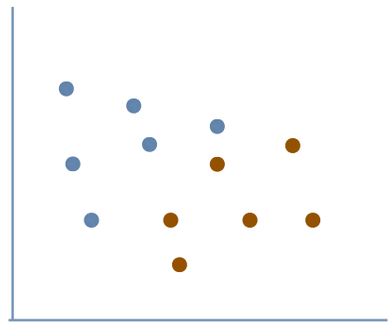
$D_2 \sim p(x, y, z)$

MIMIC

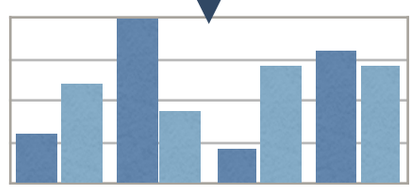


$D_1 \sim p(x, y, z)$

**(Label 0)**



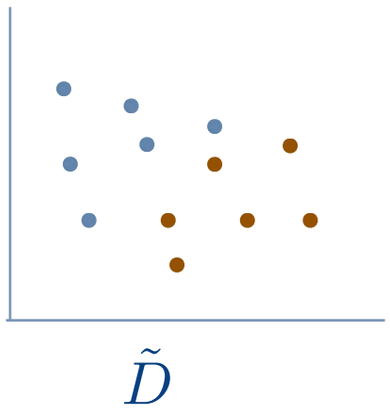
$\tilde{D} = D_1 \cup D'$



$D' \sim p(z)p(x|z)q(y|z)$

**(Label 1)**

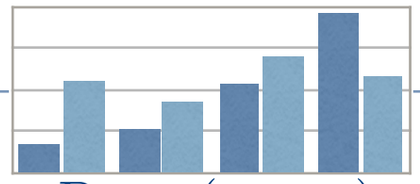
Classify Step



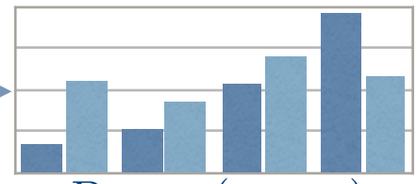
$\tilde{D}$

# Mimic and Classify

Mimic Step

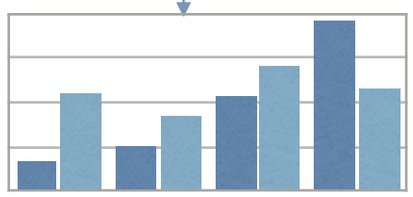


$D \sim p(x, y, z)$



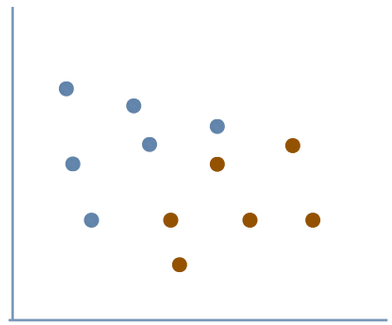
$D_2 \sim p(x, y, z)$

MIMIC

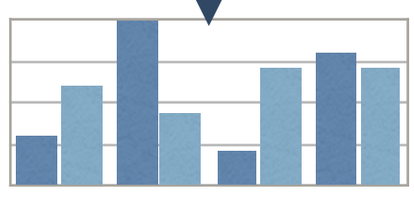


$D_1 \sim p(x, y, z)$

**(Label 0)**



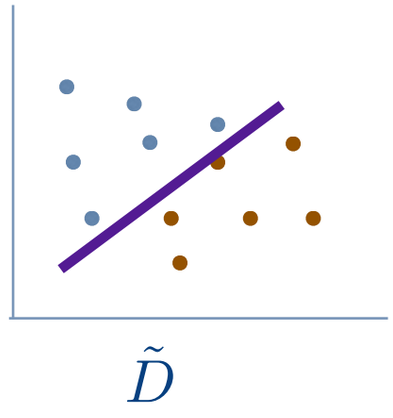
$\tilde{D} = D_1 \cup D'$



$D' \sim p(z)p(x|z)q(y|z)$

**(Label 1)**

Classify Step

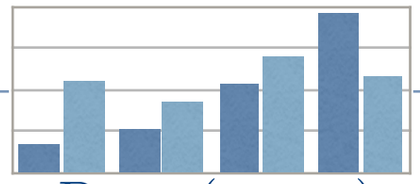


$\tilde{D}$

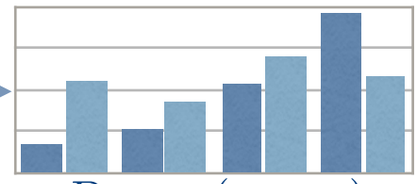
Classification Error : $\mathcal{E}_{xyz}$

# Mimic and Classify

Mimic Step

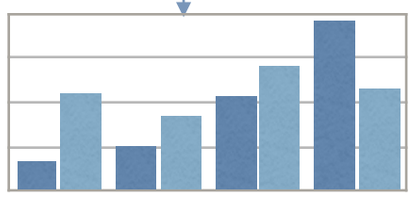


$D \sim p(x, y, z)$



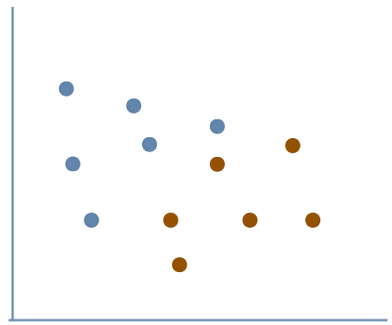
$D_2 \sim p(x, y, z)$

MIMIC

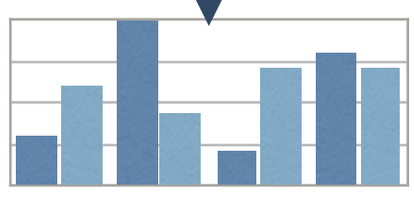


$D_1 \sim p(x, y, z)$

**(Label 0)**



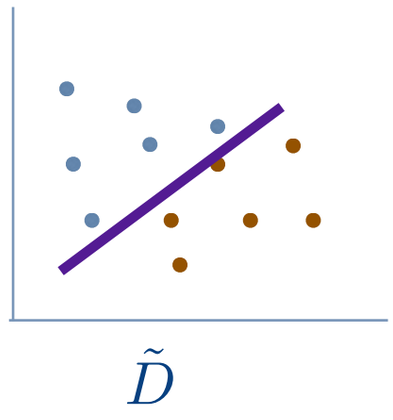
$\tilde{D} = D_1 \cup D'$



$D' \sim p(z)p(x|z)q(y|z)$

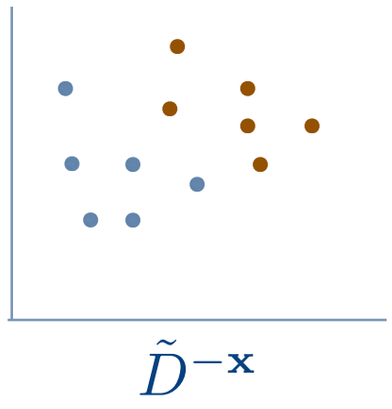
**(Label 1)**

Classify Step



$\tilde{D}$

Drop x

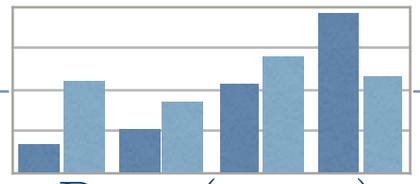


$\tilde{D}^{-\mathbf{x}}$

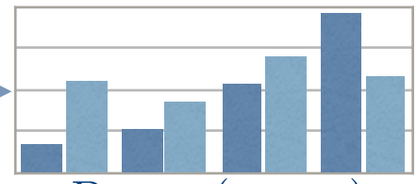
Classification Error : $\mathcal{E}_{xyz}$

# Mimic and Classify

Mimic Step

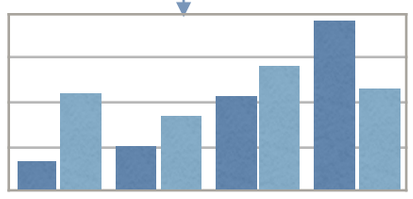


$D \sim p(x, y, z)$



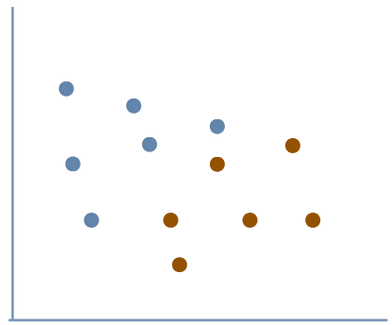
$D_2 \sim p(x, y, z)$

MIMIC

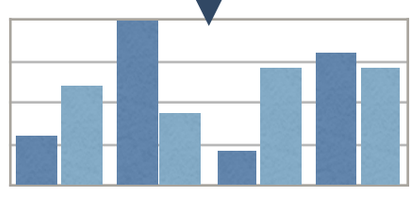


$D_1 \sim p(x, y, z)$

**(Label 0)**



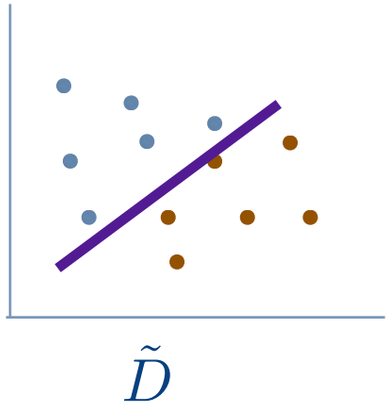
$\tilde{D} = D_1 \cup D'$



$D' \sim p(z)p(x|z)q(y|z)$

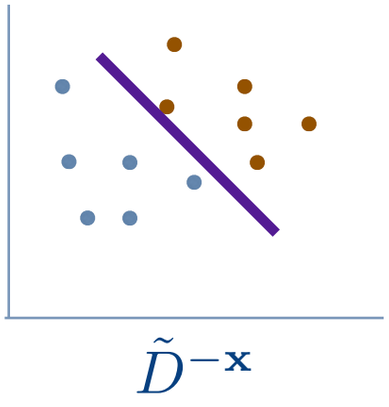
**(Label 1)**

Classify Step



$\tilde{D}$

Drop x



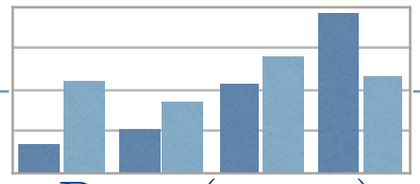
$\tilde{D}^{-\mathbf{x}}$

Classification Error : $\mathcal{E}_{xyz}$

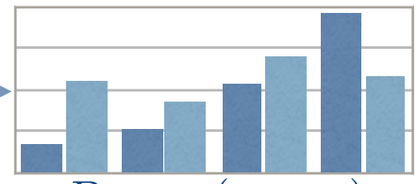
Classification Error : $\mathcal{E}_{yz}$

# Mimic and Classify

Mimic Step

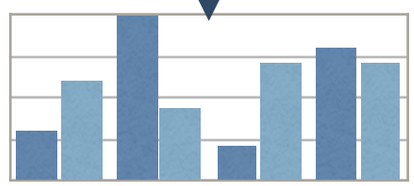


$D \sim p(x, y, z)$



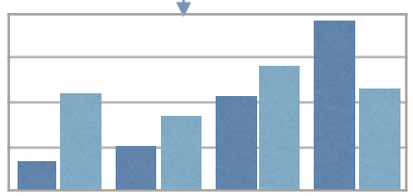
$D_2 \sim p(x, y, z)$

MIMIC



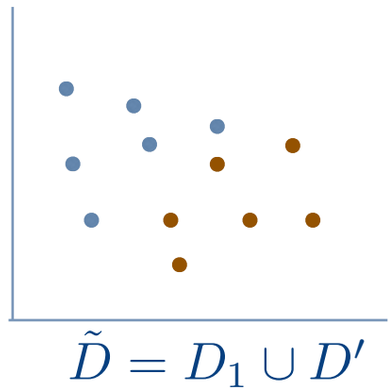
$D' \sim p(z)p(x|z)q(y|z)$

**(Label 1)**



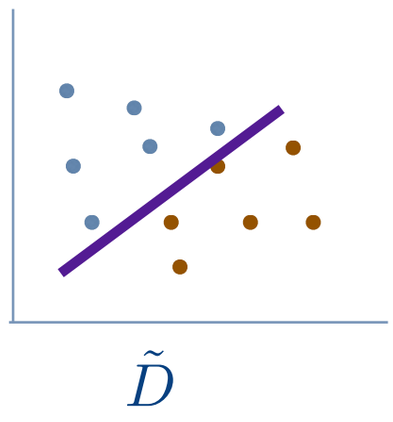
$D_1 \sim p(x, y, z)$

**(Label 0)**



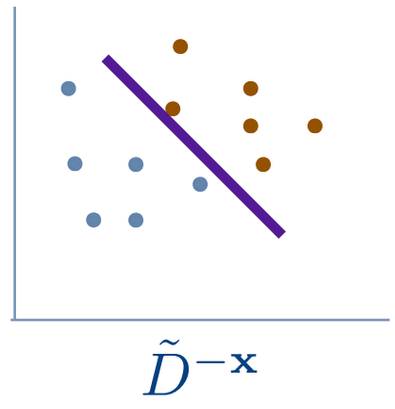
$\tilde{D} = D_1 \cup D'$

Classify Step



$\tilde{D}$

Drop x



$\tilde{D}^{-\mathbf{x}}$

Classification Error : $\mathcal{E}_{xyz}$

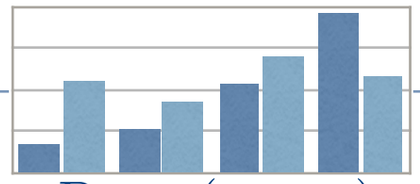
D(p(xyz) | p(xz)q(y | z))

Classification Error : $\mathcal{E}_{yz}$

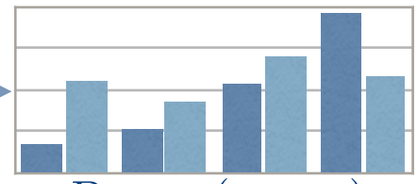
D(p(yz) | p(z)q(y | z))

# Mimic and Classify

Mimic Step

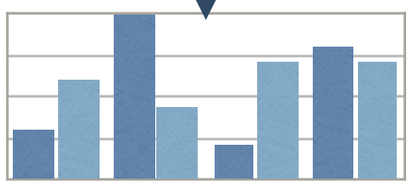


$D \sim p(x, y, z)$



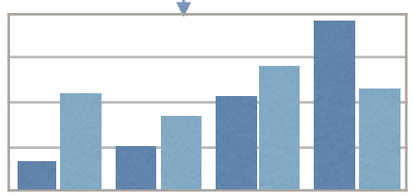
$D_2 \sim p(x, y, z)$

MIMIC



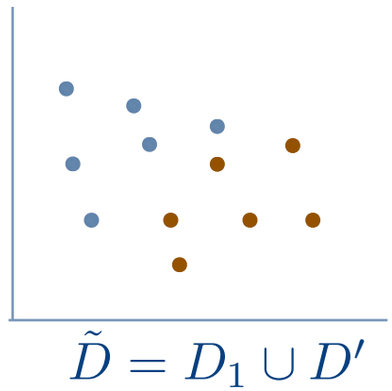
$D' \sim p(z)p(x|z)q(y|z)$

**(Label 1)**



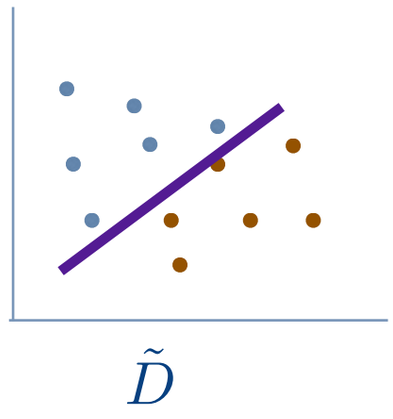
$D_1 \sim p(x, y, z)$

**(Label 0)**



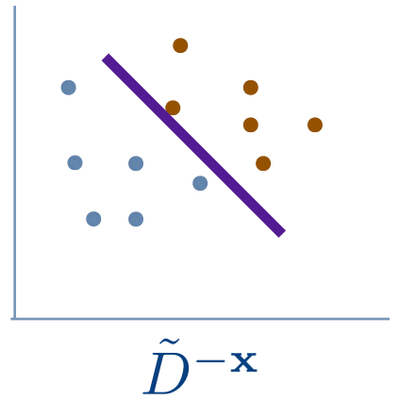
$\tilde{D} = D_1 \cup D'$

Classify Step



$\tilde{D}$

Drop x



$\tilde{D}^{-\mathbf{x}}$

Classification Error : $\mathcal{E}_{xyz}$

D(p(xyz) | p(xz)q(y | z))

Classification Error : $\mathcal{E}_{yz}$

D(p(yz) | p(z)q(y | z))

**Statistic = $E_{xyz}$-$E_{yz}$** Cancels bias due to q(y | z)

# Mimic and Classify

Mimic Step

As long as the density function $q(\mathbf{y}|\mathbf{z}) > 0$ whenever $p(\mathbf{y}, \mathbf{z}) > 0$.

Classify Step

# Mimic and Classify

Mimic Step

As long as the density function $q(\mathbf{y}|\mathbf{z}) > 0$ whenever $p(\mathbf{y}, \mathbf{z}) > 0$.

Classify Step

$$|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]| = 0 \leftrightarrow \mathcal{H}_0 \text{ is true}$$

$$2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]|$$
$$= D_{\mathrm{TV}}(p(\mathbf{z}, \mathbf{x}, \mathbf{y}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})) - D_{\mathrm{TV}}(p(\mathbf{y}, \mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

*The errors here are the corresponding optimal Bayes classifier errors.

# Mimic and Classify (Theory)

$$
\begin{aligned}
&2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]| \\
&= D_{\mathrm{TV}}(p(\mathbf{z},\mathbf{x},\mathbf{y}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})) - D_{\mathrm{TV}}(p(\mathbf{y},\mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))
\end{aligned}
$$

# Mimic and Classify (Theory)

$$
\begin{aligned}
&2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]| \\
&= D_{\mathrm{TV}}(p(\mathbf{z},\mathbf{x},\mathbf{y}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})) - D_{\mathrm{TV}}(p(\mathbf{y},\mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})) \\
&\geq \int_{\mathbf{y},\mathbf{z}} \min(p(\mathbf{z})q(\mathbf{y}|\mathbf{z}), p(\mathbf{z})p(\mathbf{y}|\mathbf{z}))(1 - \epsilon(\mathbf{y},\mathbf{z}))d(\mathbf{y},\mathbf{z})
\end{aligned}
$$

# Mimic and Classify (Theory)

$$
\begin{aligned}
&2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]| \\
&= D_{\mathrm{TV}}(p(\mathbf{z},\mathbf{x},\mathbf{y}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})) - D_{\mathrm{TV}}(p(\mathbf{y},\mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})) \\
&\geq \int_{\mathbf{y},\mathbf{z}} \min(p(\mathbf{z})q(\mathbf{y}|\mathbf{z}), p(\mathbf{z})p(\mathbf{y}|\mathbf{z}))(1 - \epsilon(\mathbf{y},\mathbf{z}))d(\mathbf{y},\mathbf{z})
\end{aligned}
$$

Where: $\epsilon(\mathbf{y},\mathbf{z}) = \max_{\pi \in \Pi(p(\mathbf{x}|\mathbf{z}), p(\mathbf{x}'|\mathbf{y},\mathbf{z}))} \mathbb{E}_\pi[\mathbf{1}_{\{\mathbf{x}=\mathbf{x}'\}}|\mathbf{y},\mathbf{z}]$

Conditional dependence $\leftrightarrow \epsilon(y,z) < 1$ with non-zero probability

# Mimic and Classify (Theory)

$$
\begin{aligned}
&2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]| \\
&= D_{\mathrm{TV}}(p(\mathbf{z},\mathbf{x},\mathbf{y}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})) - D_{\mathrm{TV}}(p(\mathbf{y},\mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})) \\
&\geq \int_{\mathbf{y},\mathbf{z}} \min(p(\mathbf{z})q(\mathbf{y}|\mathbf{z}), p(\mathbf{z})p(\mathbf{y}|\mathbf{z}))(1 - \epsilon(\mathbf{y},\mathbf{z}))d(\mathbf{y},\mathbf{z})
\end{aligned}
$$

Where: $\epsilon(\mathbf{y},\mathbf{z}) = \max_{\pi \in \Pi(p(\mathbf{x}|\mathbf{z}), p(\mathbf{x}'|\mathbf{y},\mathbf{z}))} \mathbb{E}_\pi[\mathbf{1}_{\{\mathbf{x}=\mathbf{x}'\}}|\mathbf{y},\mathbf{z}]$

Conditional dependence $\leftrightarrow \epsilon(y,z) < 1$ with non-zero probability

Theorem 1

As long as the density function $q(\mathbf{y}|\mathbf{z}) > 0$ whenever $p(\mathbf{y},\mathbf{z}) > 0$, then conditional dependence implies that $2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]| > 0$

# Mimic and Classify (Theory)

Conditional independence implies $p(\mathbf{x}, \mathbf{y}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})$.

$$D_{\mathrm{TV}}(p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})) = D_{\mathrm{TV}}(p(\mathbf{x}|\mathbf{z})p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

# Mimic and Classify (Theory)

Conditional independence implies $p(\mathbf{x}, \mathbf{y}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})$.

$$D_{\mathrm{TV}}(p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})) = D_{\mathrm{TV}}(p(\mathbf{x}|\mathbf{z})p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

$$2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]|$$

$$= D_{\mathrm{TV}}(p(\mathbf{z}, \mathbf{x}, \mathbf{y}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})) - D_{\mathrm{TV}}(p(\mathbf{y}, \mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

# Mimic and Classify (Theory)

Conditional independence implies $p(\mathbf{x}, \mathbf{y}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})$.

$$D_{\mathrm{TV}}(p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})) = D_{\mathrm{TV}}(p(\mathbf{x}|\mathbf{z})p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

$$2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]|$$

$$= D_{\mathrm{TV}}(p(\mathbf{z}, \mathbf{x}, \mathbf{y}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})) - D_{\mathrm{TV}}(p(\mathbf{y}, \mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

$$= D_{\mathrm{TV}}(p(\mathbf{x}|\mathbf{z})p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

# Mimic and Classify (Theory)

Conditional independence implies $p(\mathbf{x}, \mathbf{y}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})$.

$$D_{\mathrm{TV}}(p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})) = D_{\mathrm{TV}}(p(\mathbf{x}|\mathbf{z})p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

$$2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]|$$

$$= D_{\mathrm{TV}}(p(\mathbf{z}, \mathbf{x}, \mathbf{y}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})) - D_{\mathrm{TV}}(p(\mathbf{y}, \mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

$$= D_{\mathrm{TV}}(p(\mathbf{x}|\mathbf{z})p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

$$= D_{\mathrm{TV}}(p(\mathbf{x}, \mathbf{y}, \mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

# Mimic and Classify (Theory)

Conditional independence implies $p(\mathbf{x}, \mathbf{y}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})$.

$$D_{\mathrm{TV}}(p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})) = D_{\mathrm{TV}}(p(\mathbf{x}|\mathbf{z})p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

$$2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]|$$

$$= D_{\mathrm{TV}}(p(\mathbf{z}, \mathbf{x}, \mathbf{y}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z})p(\mathbf{x}|\mathbf{z})) - D_{\mathrm{TV}}(p(\mathbf{y}, \mathbf{z}), p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

$$= D_{\mathrm{TV}}(p(\mathbf{x}|\mathbf{z})p(\mathbf{z})p(\mathbf{y}|\mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

$$= D_{\mathrm{TV}}(p(\mathbf{x}, \mathbf{y}, \mathbf{z}), p(\mathbf{x}|\mathbf{z})p(\mathbf{z})q(\mathbf{y}|\mathbf{z}))$$

Theorem 2

Conditional independence implies that $2|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]| = 0$

# Mimic and Classify (Theory)

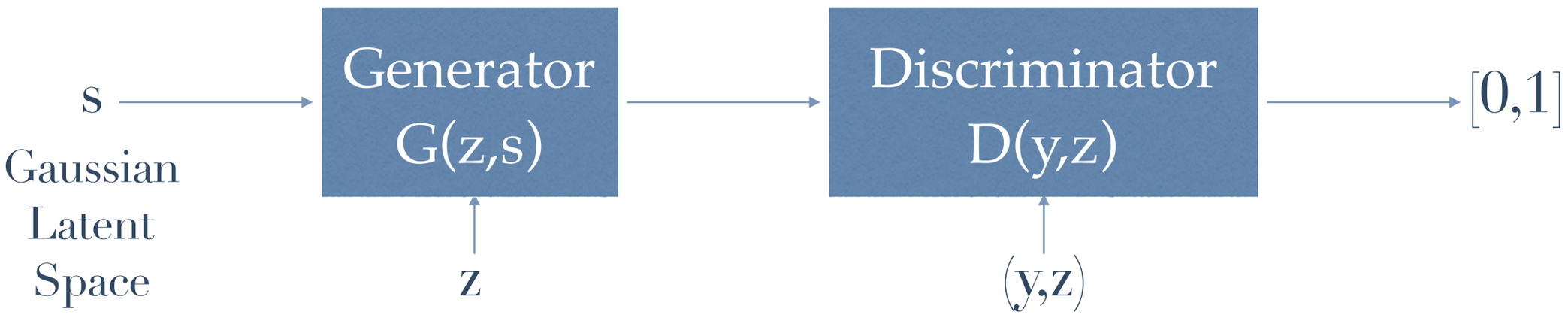
Combining Theorem 1 and Theorem 2

Theorem 3

As long as the density function $q(y|z) > 0$ when $p(y, z) > 0$
$|\mathbf{E}_D[\mathcal{E}_{xyz}] - \mathbf{E}_D[\mathcal{E}_{yz}]| = 0 \leftrightarrow \mathcal{H}_0$ is true

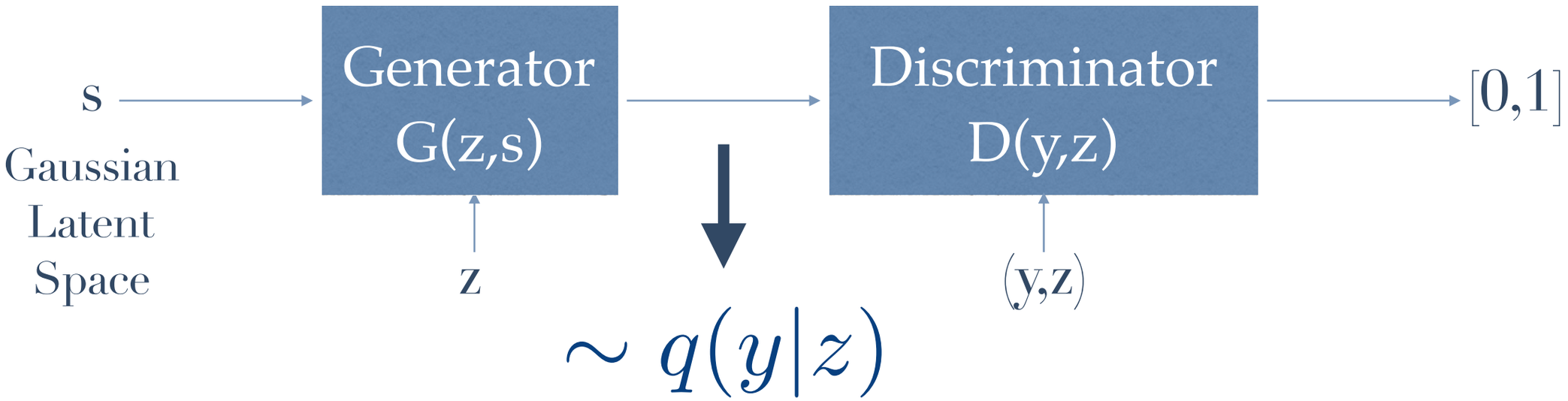
# Deep Learning based MIMIC Functions

## MIMIFY - CGAN



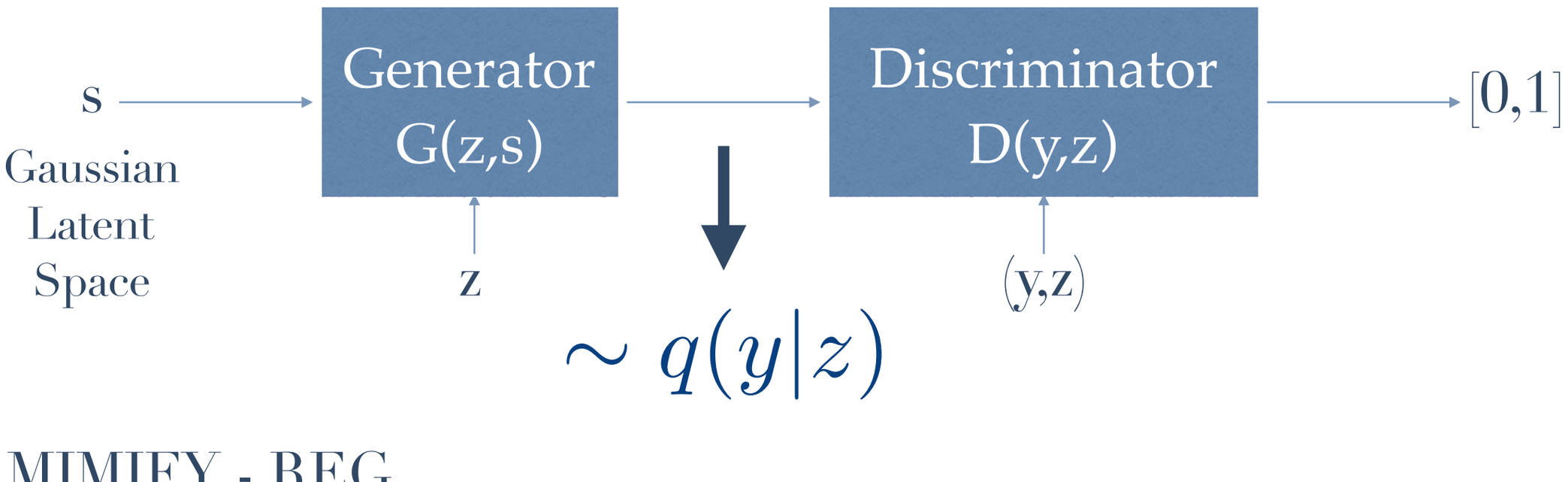
# Deep Learning based MIMIC Functions

## MIMIFY - CGAN



# Deep Learning based MIMIC Functions

## MIMIFY - CGAN

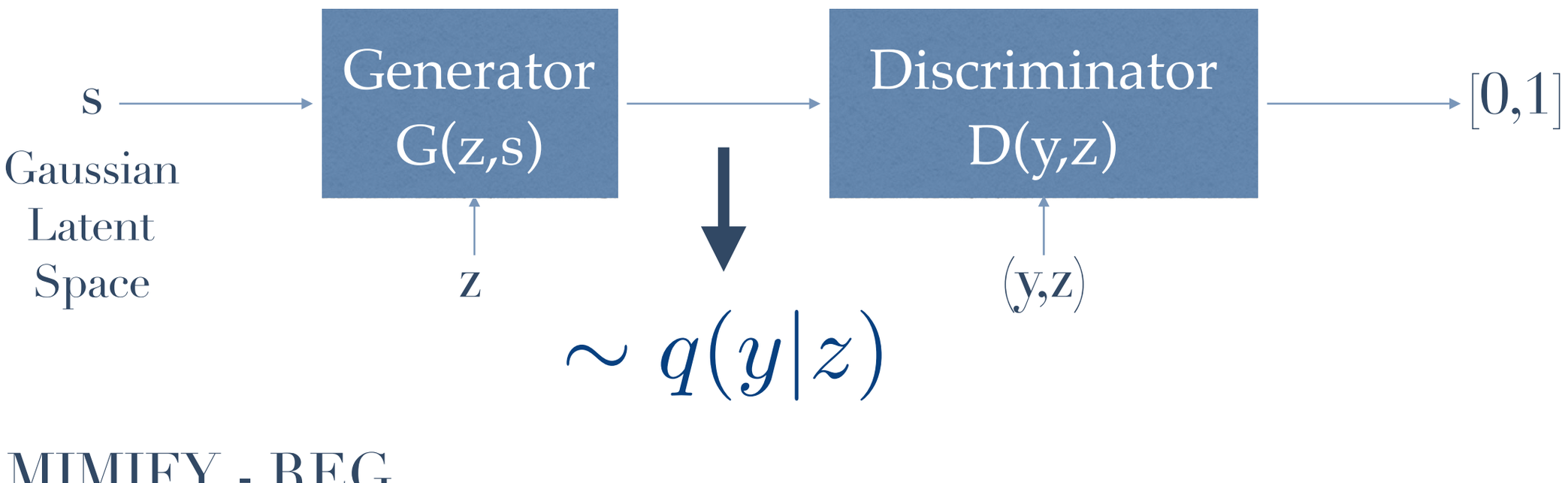



## MIMIFY - REG

$$\text{Regress to estimate } r(z) = \mathbf{E}[Y|Z = z]$$

# Deep Learning based MIMIC Functions

## MIMIFY - CGAN



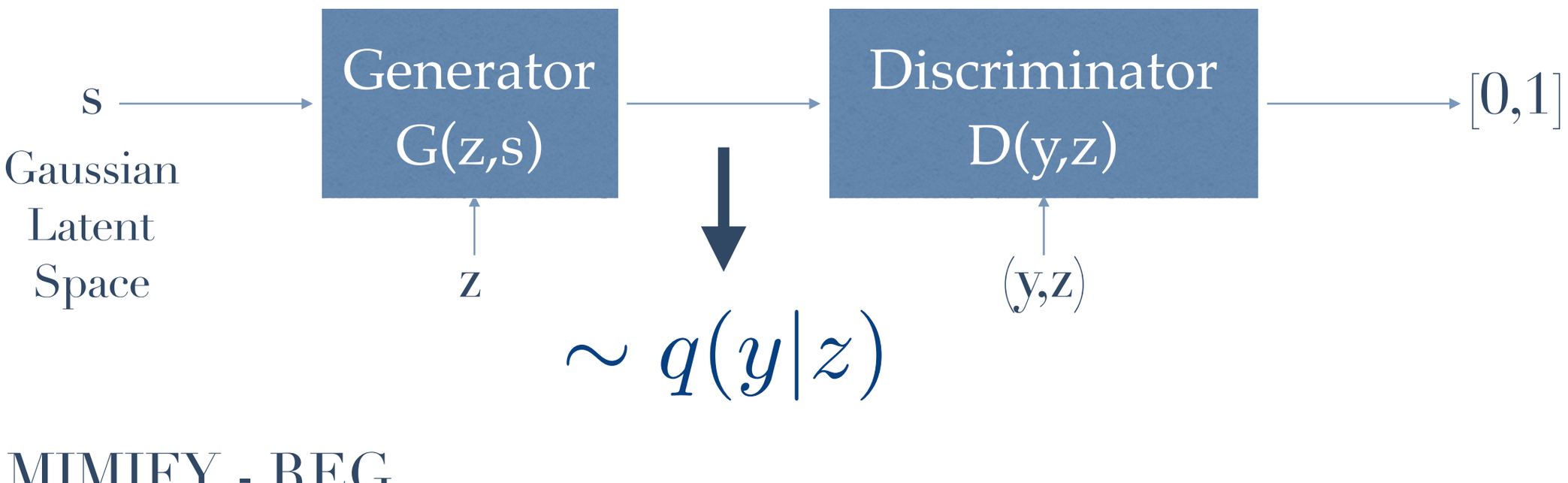

## MIMIFY - REG

$$\text{Regress to estimate } r(z) = \mathbf{E}[Y|Z = z]$$

$$\hat{y} = r(z) + \text{ Gaussian Noise } \sim q(y|z)$$

# Deep Learning based MIMIC Functions

## MIMIFY - CGAN




## MIMIFY - REG

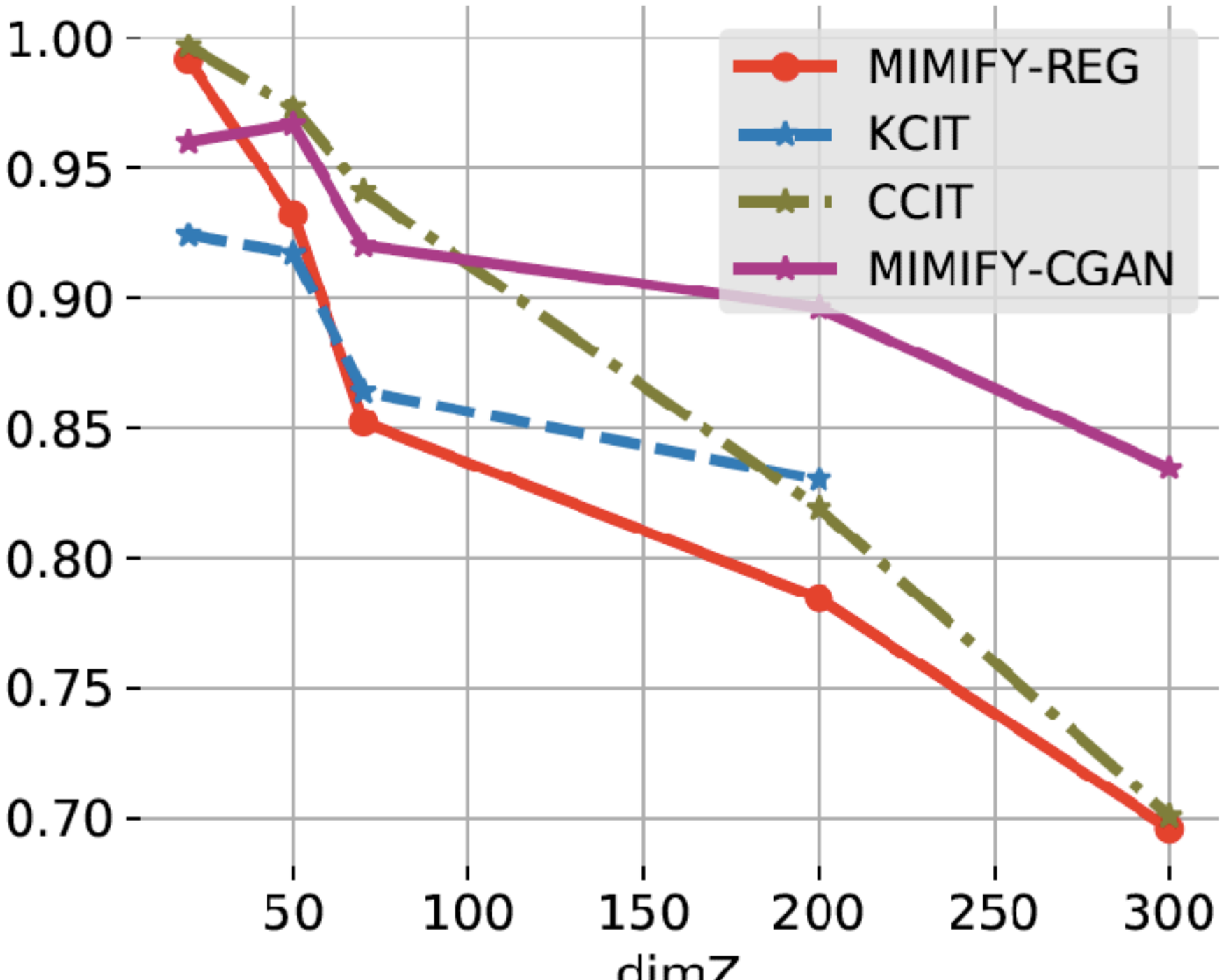
$$\text{Regress to estimate } r(z) = \mathbf{E}[Y|Z = z]$$

$$\hat{y} = r(z) + \text{ Gaussian Noise } \sim q(y|z)$$

(or, laplacian noise)

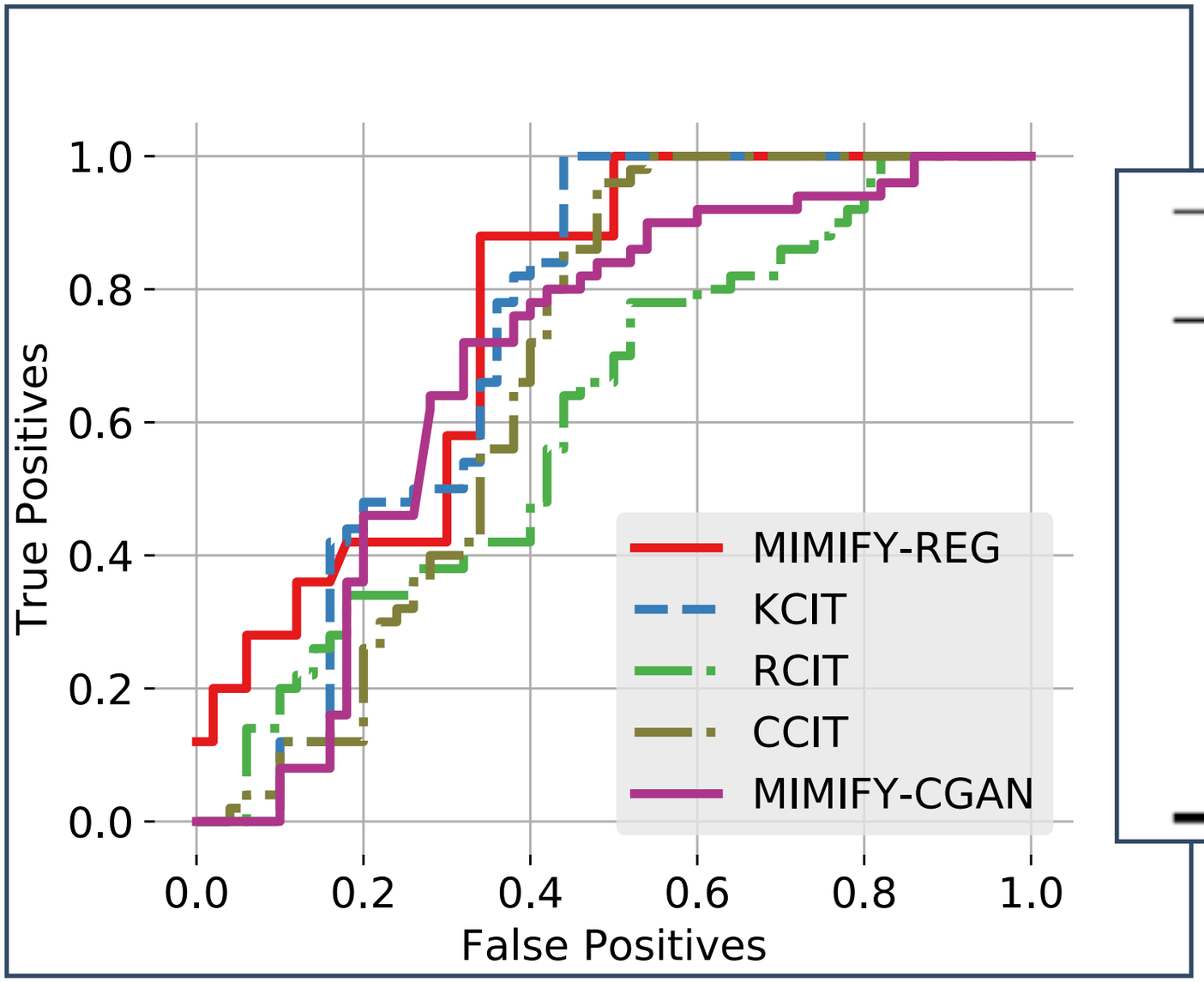
# Experiments

## Post-Nonlinear Noise Synthetic Experiments: AUROC



# Experiments

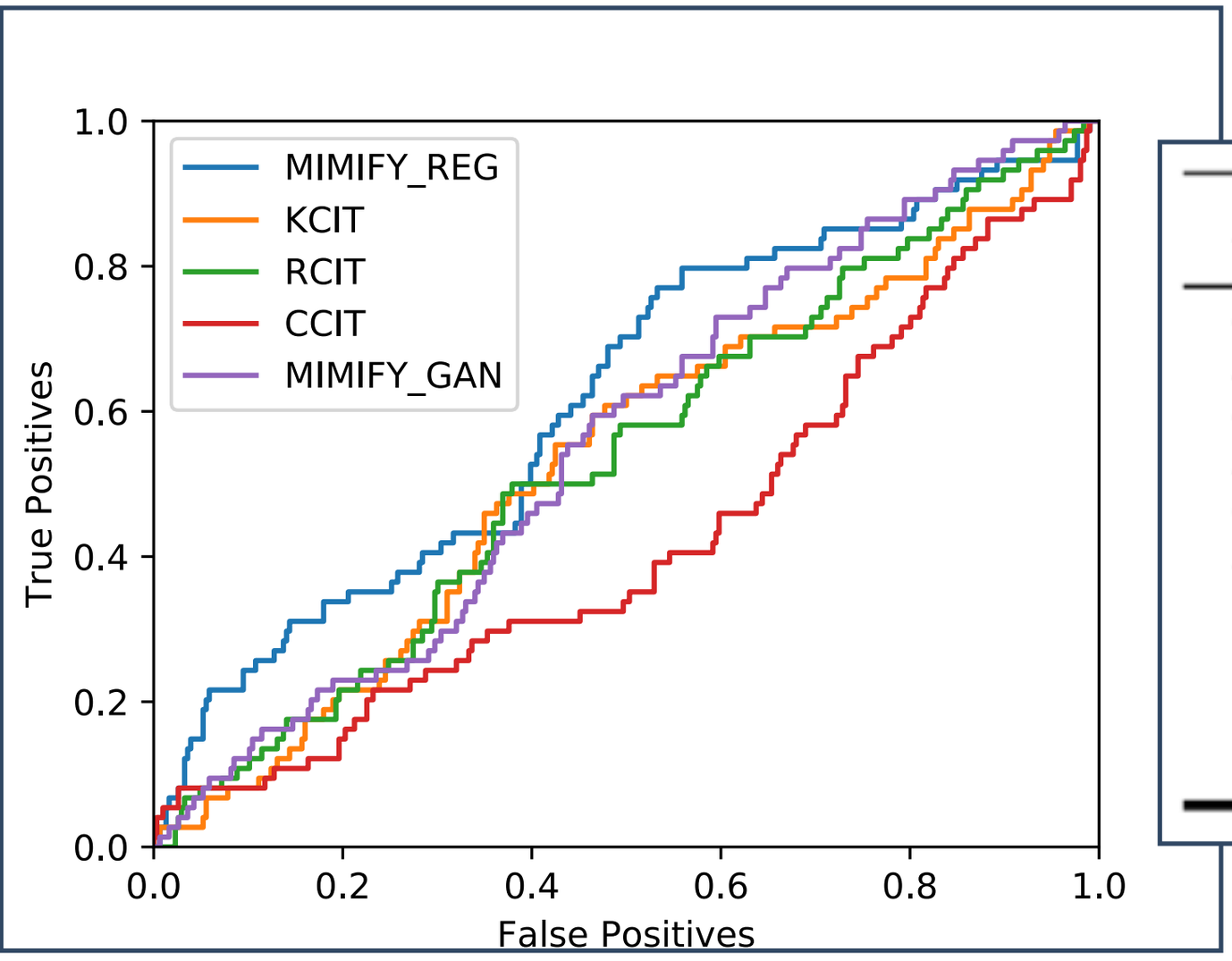
## Flow-cytometry Data




| Algo. | ROC-AUC |
|---|---|
| MIMIFY-REG | 0.7638 |
| KCIT | 0.7328 |
| MIMIFY-GAN | 0.6891 |
| CCIT | 0.6816 |
| RCIT | 0.6135 |

# Experiments

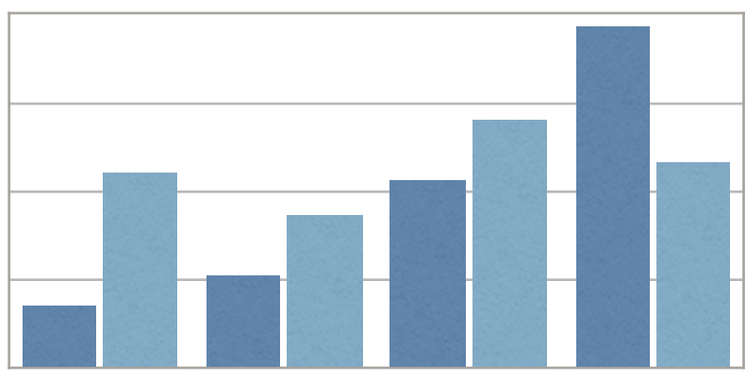
## Gene Regulatory Network Inference (DREAM)



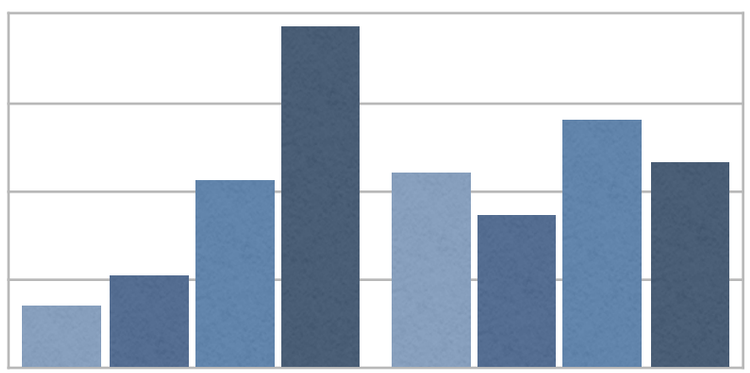

| Algo. | ROC-AUC |
|---|---|
| MIMIFY-REG | 0.61645 |
| MIMIFY-GAN | 0.55679 |
| RCIT | 0.53511 |
| KCIT | 0.53175 |
| CCIT | 0.42439 |

# Estimating Information Measures

# Estimating Kullback-Leibler Distance



$P$



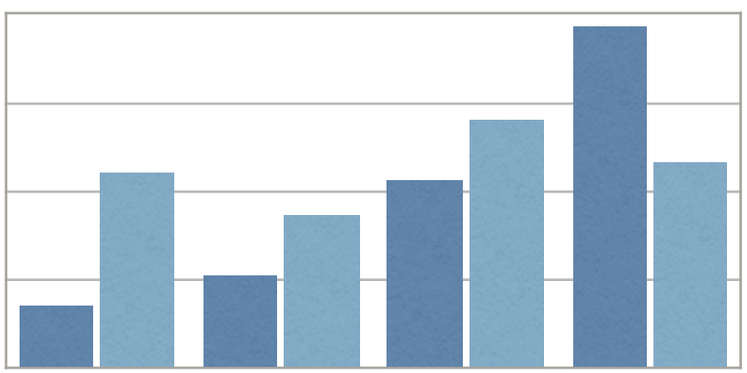
$Q$

n samples

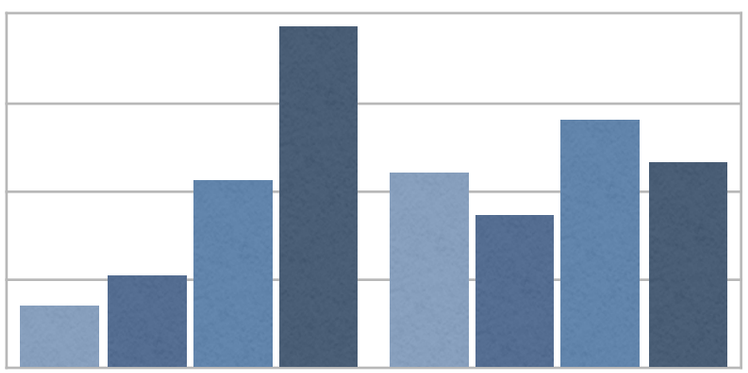
n samples

Estimate $D_{KL}(P \parallel Q)$ ?

# Estimating Kullback-Leibler Distance



$P$



$Q$

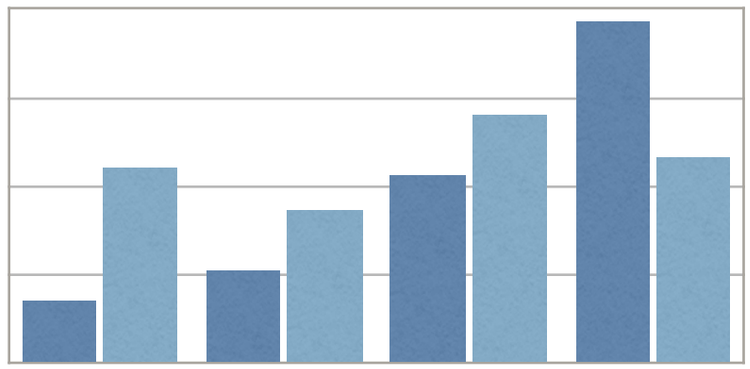
n samples

n samples

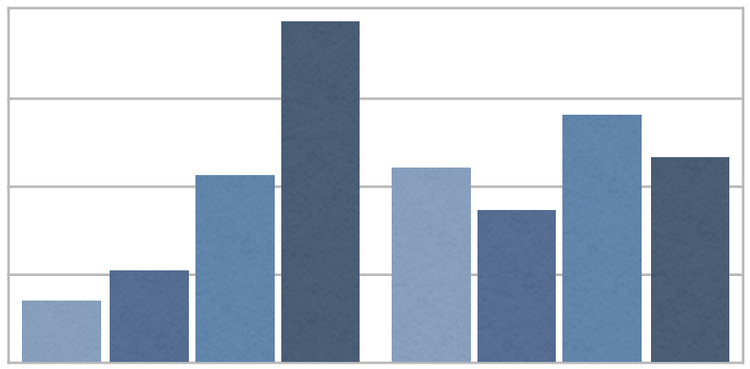
Estimate $D_{KL}(P \parallel Q)$ ?

Curse of dimensionality: Sample complexity $O(n/\log n)$ with $n = 2^d$

# Neural Network Approximation

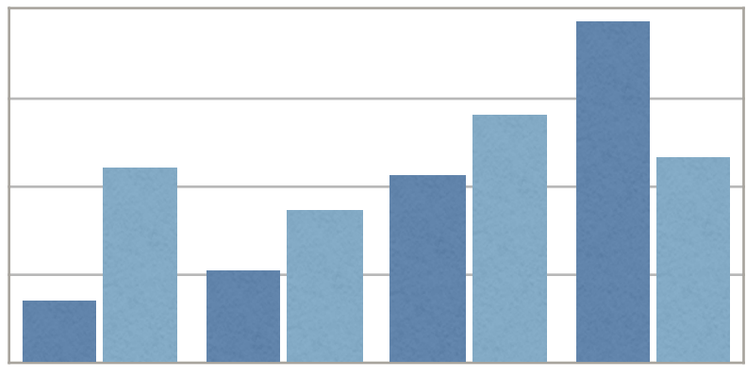


n samples ~ P

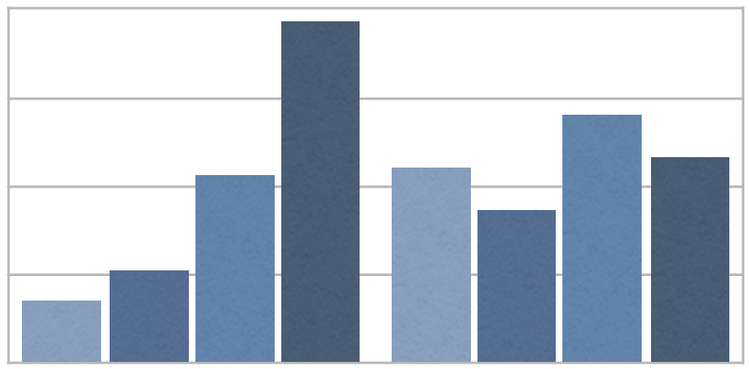


n samples ~ Q

# Neural Network Approximation



n samples $\sim$ P

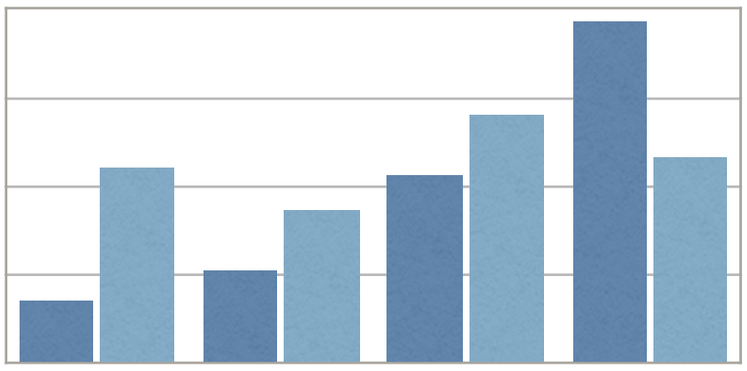


n samples $\sim$ Q

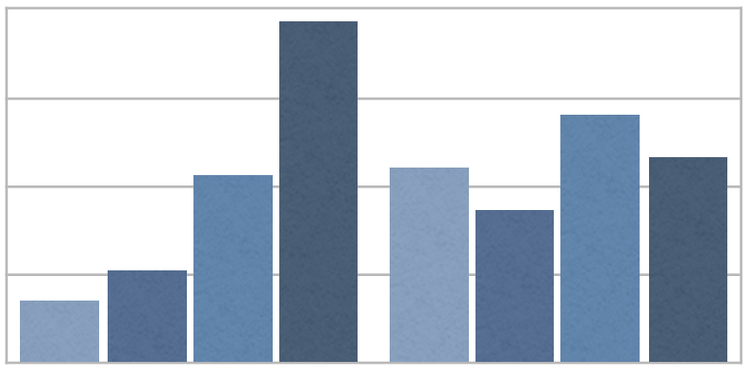
Donsker-Varadhan Dual Representation:

$$D_{KL}(P \parallel Q) = \sup_T \mathbf{E}_P[T] - \log(\mathbf{E}_Q[e^T])$$

# MINE: Neural Network Approximation



n samples $\sim$ P



n samples $\sim$ Q

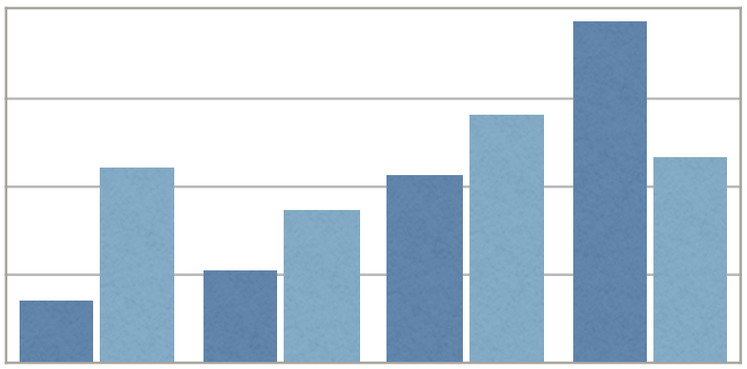
Donsker-Varadhan Dual Representation:

$$D_{KL}(P \parallel Q) = \sup_T \mathbf{E}_P[T] - \log(\mathbf{E}_Q[e^T])$$

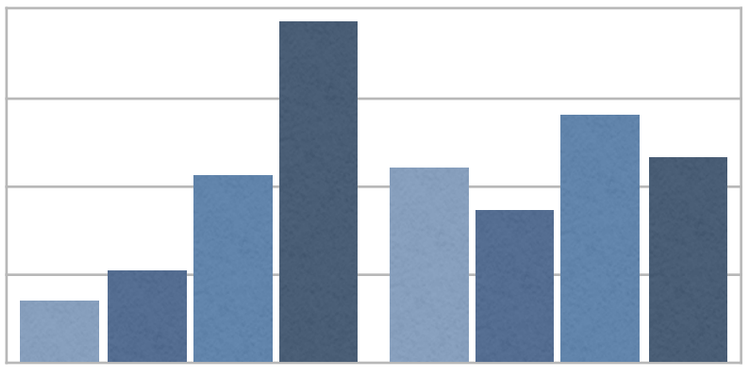
- $T \leftarrow$ Rich NN class
- $\mathbf{E} \leftarrow$ Sample Averages
- $\sup_T \leftarrow$ Obtained via Stochastic Gradient search

*Benghazi et al, **MINE : Mutual Information Neural Estimation**, *ICML 2018*

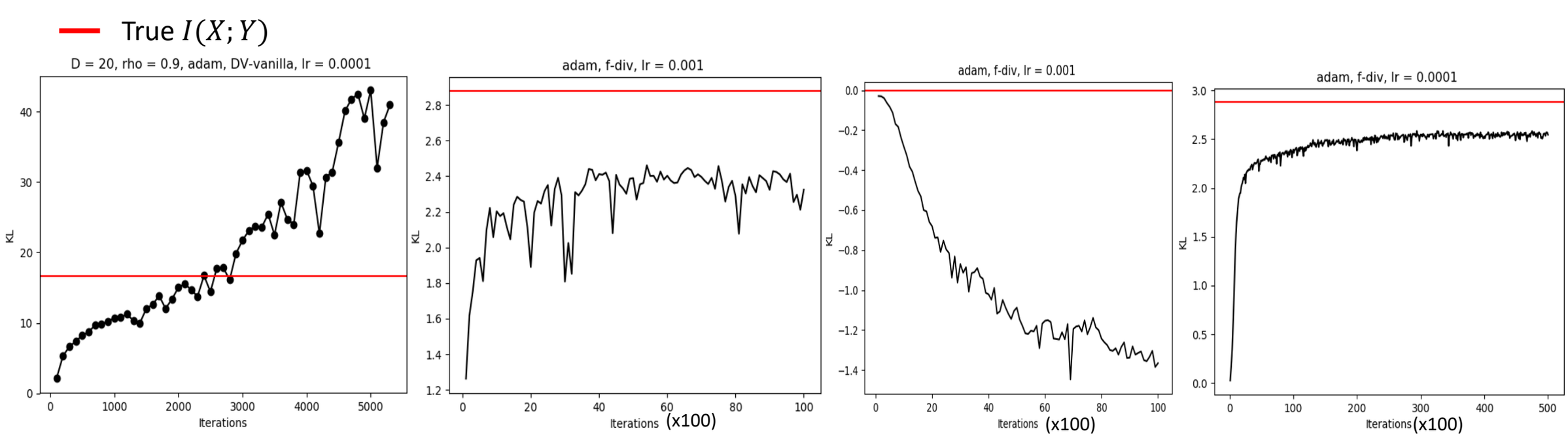
# MINE is Unstable to Train



n samples $\sim$ P



n samples $\sim$ Q



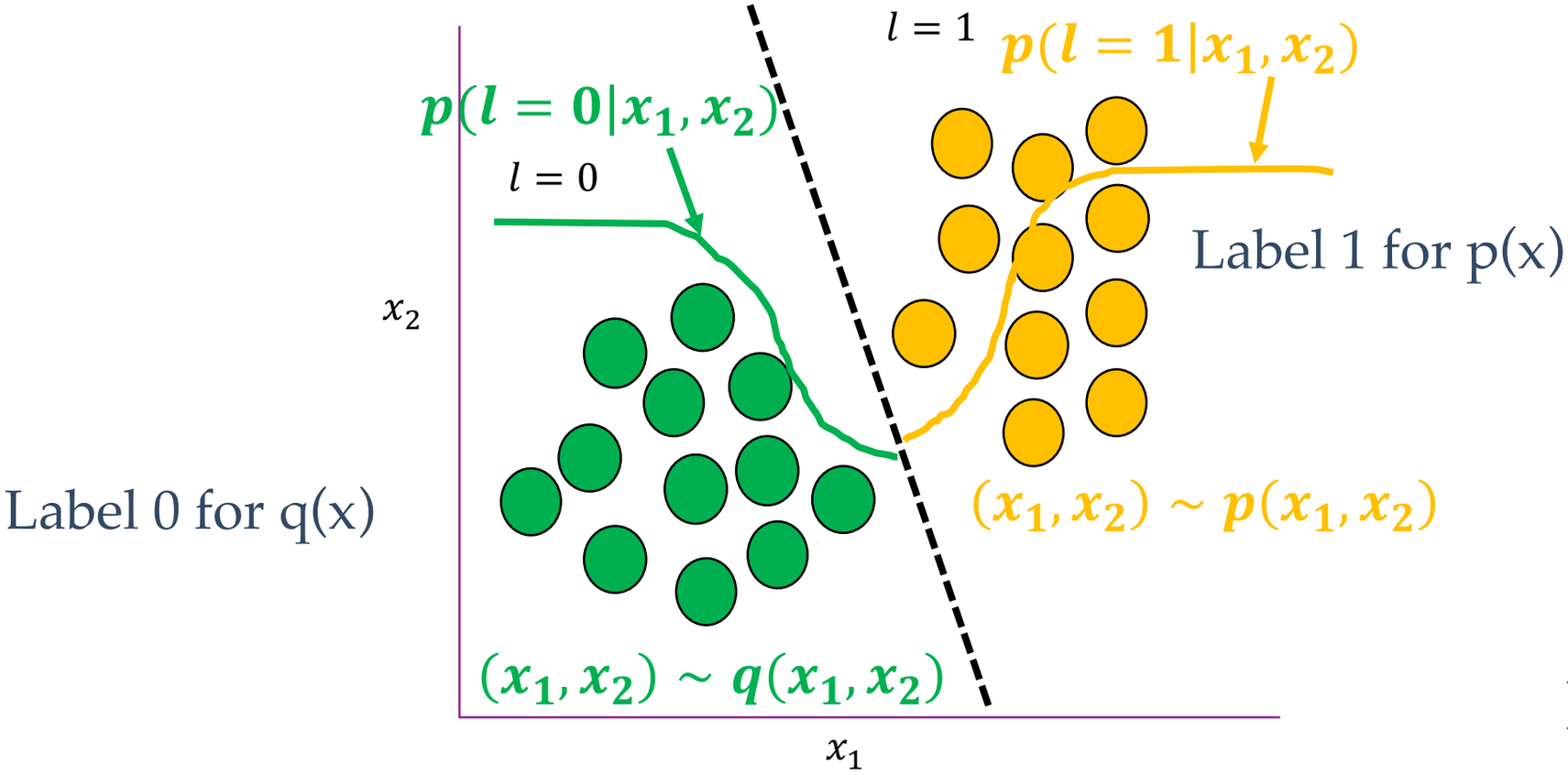

*Benghazi et al, **MINE : Mutual Information Neural Estimation**, *ICML 2018*

# Divergence estimation via Classification

- $D_{KL}(p\|q) = \sup_{f \in \mathcal{F}} \mathbb{E}_{x\sim p(x)}[f(x)] - \log(\mathbb{E}_{x\sim q(x)}[\, e^{f(x)}])$

$$f^*(x) = log\frac{p(x)}{q(x)} \quad is\ optimal \quad \text{(RN derivative)}$$

Classifiers can estimate f*




$$f^* = \frac{p(l=1\,|\,x)}{p(l=0\,|\,x)}$$

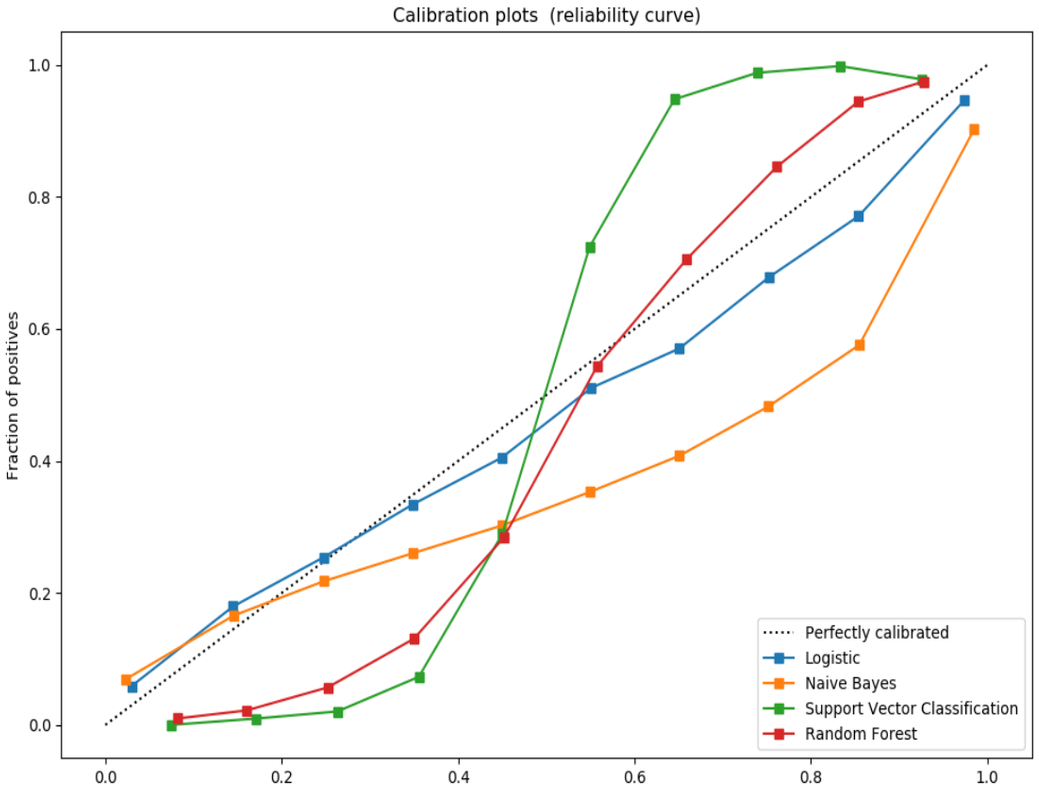
Plug in to DV-bound

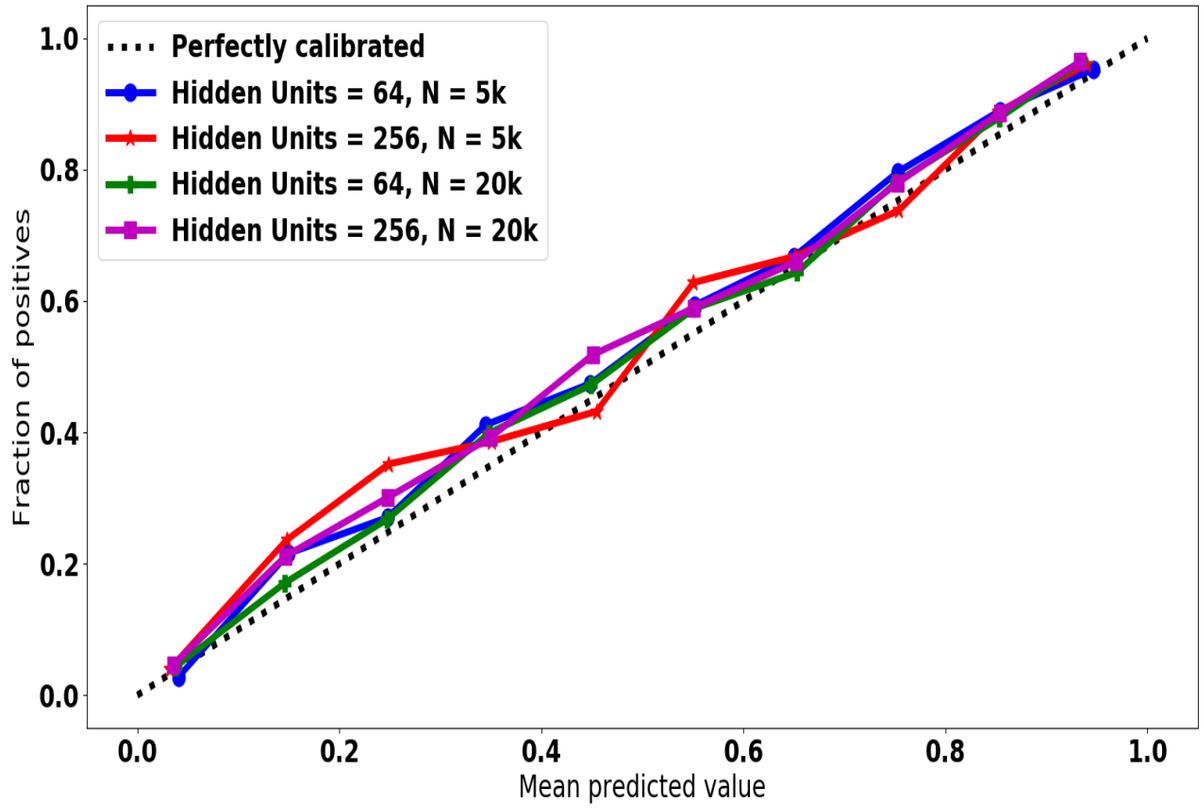
**Highly stable training!**

**True lower bound**

# Classifiers require calibration

Require classifiers that are calibrated => Require $p(l=1 \mid x)$




# Can get well-calibrated neural networks

Lakshminarayan et al, Simple and scalable predictive uncertainty estimation using deep ensembles, NeurIPS 17

# Mutual Information Estimation

$$I(X;Y) = D_{KL}(\mathbf{P}_{XY} \parallel \mathbf{P}_X \mathbf{P}_Y)$$

Samples

Permutated samples

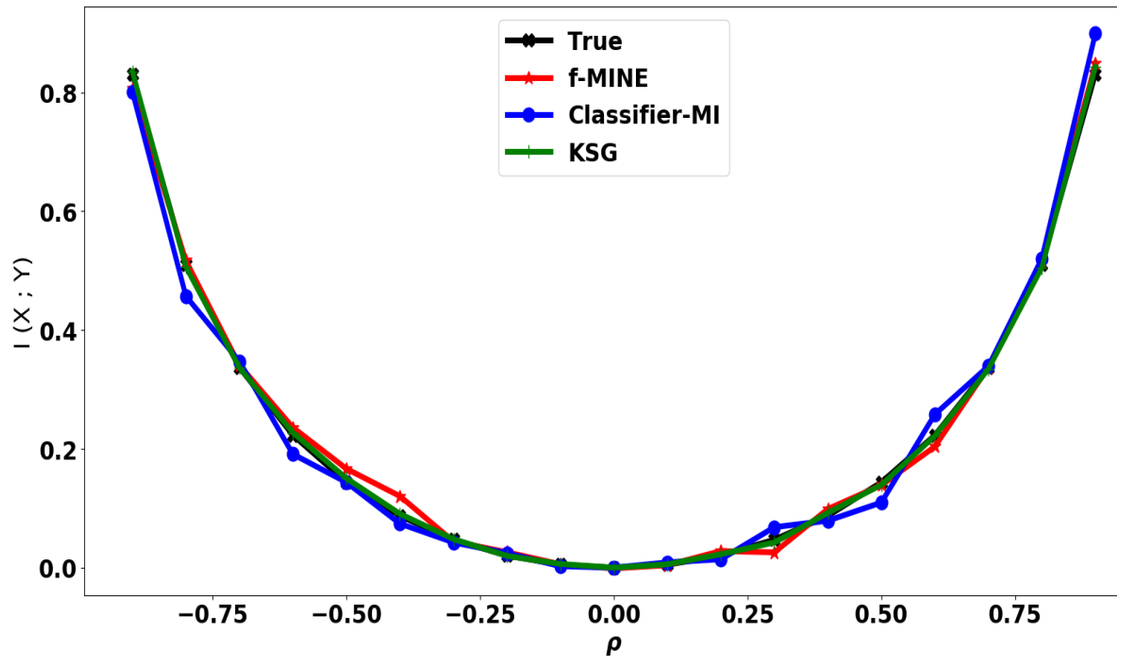
**Classifier-MI**: use classifier to estimate I(X;Y)

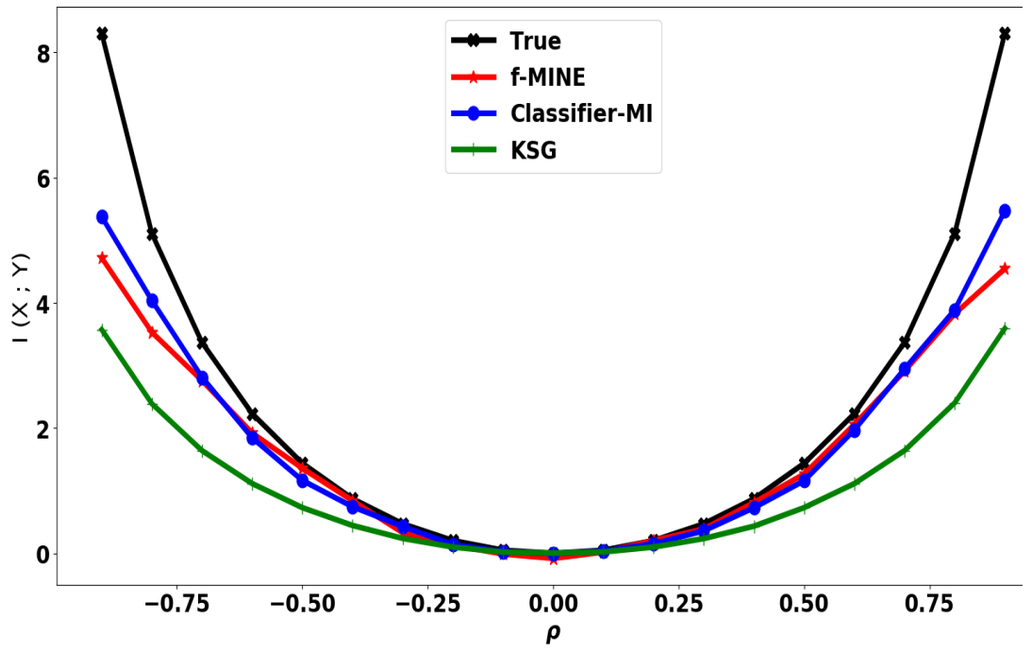
Theorem-1: NeuralNet-Classifier-MI is consistent

Theorem-2: Classifier-MI is a “true” lower-bound on MI

# Performance: MI Estimation

$$x_i, y_i \sim \mathcal{N}\left(\vec{0}, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}\right), \qquad X = (x_1, x_2, \dots\ x_d), \qquad Y = (y_1, y_2, \dots\ y_d), \qquad x_i \perp y_j\ (j \neq i)$$

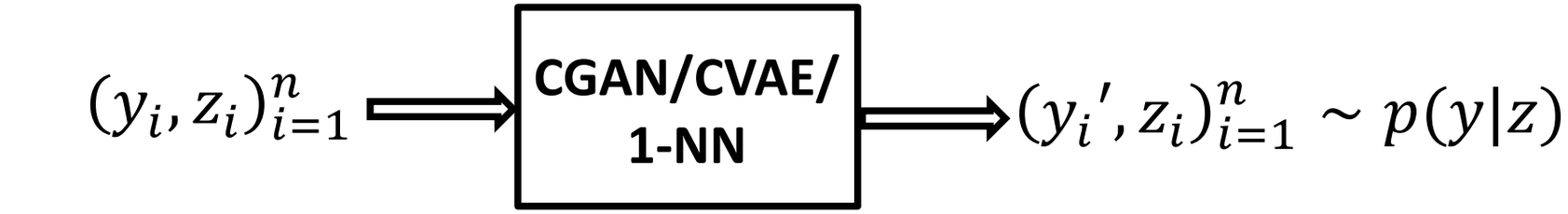


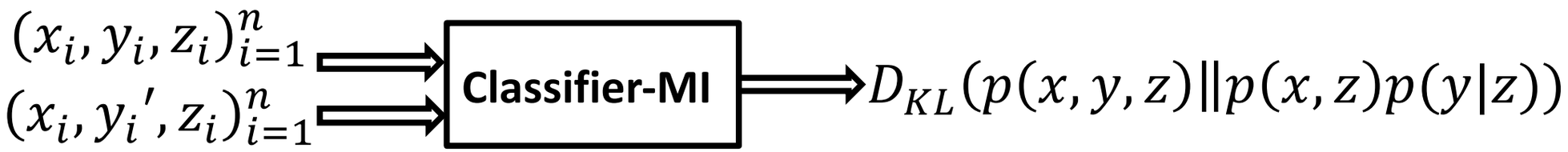



# Performance: Conditional MI Estimation

Estimate $I(X;Y \mid Z) = D_{KL}\left(p_{XYZ} \mid p_{XZ}p_{Y \mid Z}\right)$

- Modular Estimation




- $I(X;Y|Z) = I(X;Y,Z) - I(X;Z)$

# Performance: CMI

Model – I (Linear)

$X \sim \mathcal{N}(0,1)\,, Z \sim \mathcal{U}(-0.5, 0.5)^{d_z}\,, \epsilon \sim \mathcal{N}(Z_1, 0.01)$

$Y = X + \epsilon$

Model – II (Linear)

$X \sim \mathcal{N}(0,1)\,, Z \sim \mathcal{N}(0,1)^{d_z}\,, \epsilon \sim \mathcal{N}(w^T Z, 0.01)$

$Y = X + \epsilon$

Non-Linear models :

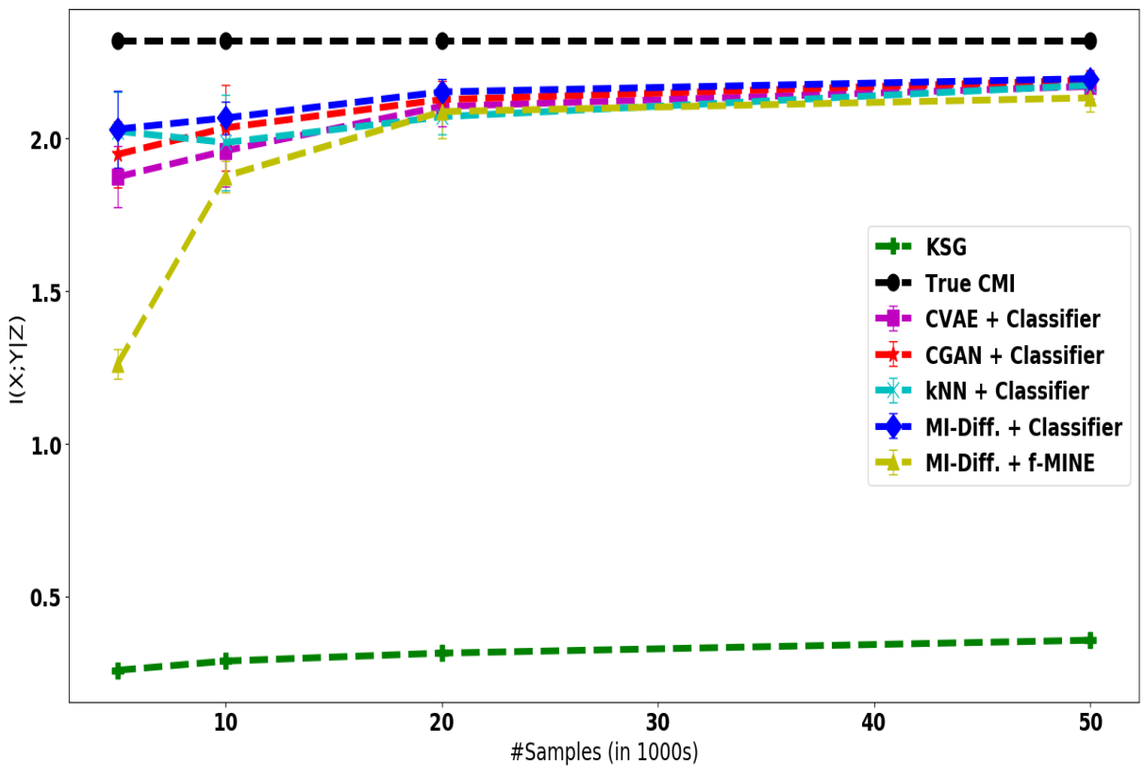
$$Z \sim \mathcal{N}\left(\mathbb{1}, I_{d_z}\right), \qquad X = f_1(\eta_1) \qquad Y = f_2\left(A_{xy}X + A_{zy}Z + \eta_2\right)$$

$$f_1, f_2 \sim \{\, tanh, cos, \exp(-|\cdot|)\}, \qquad \eta_1, \eta_2 \sim \mathcal{N}(0, 0.1)$$

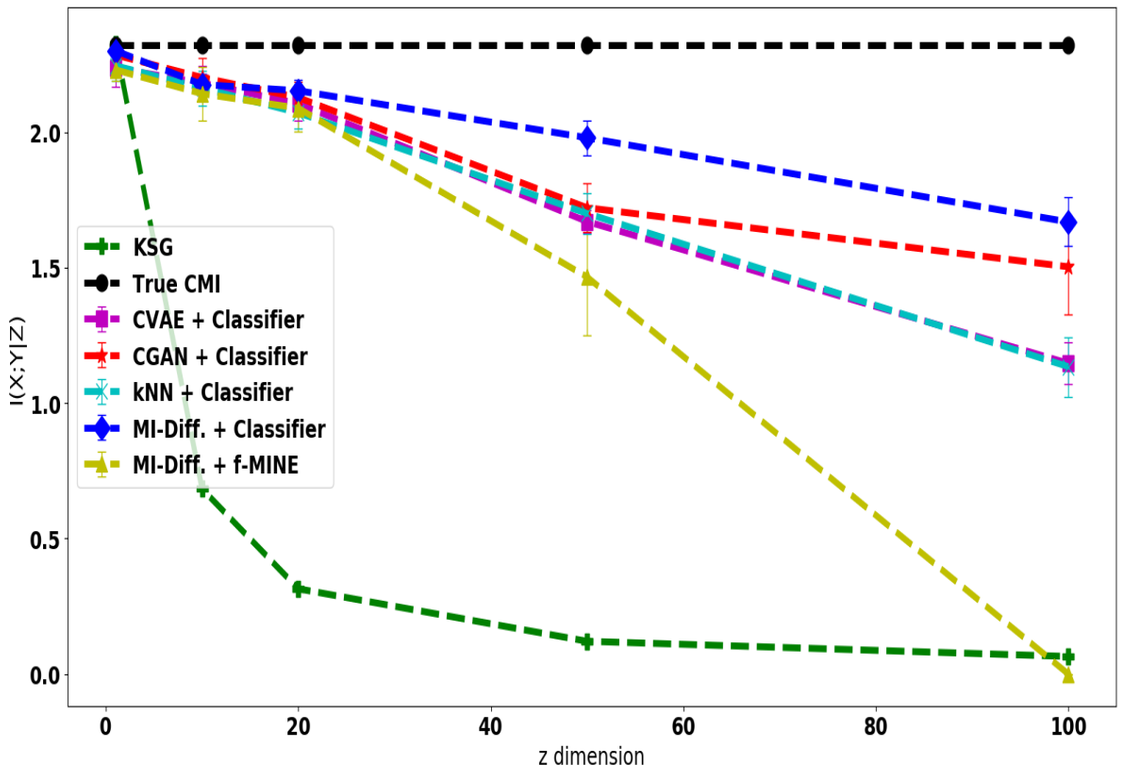
# Performance: CMI

## Model-1 (linear)

$d_z = 20$



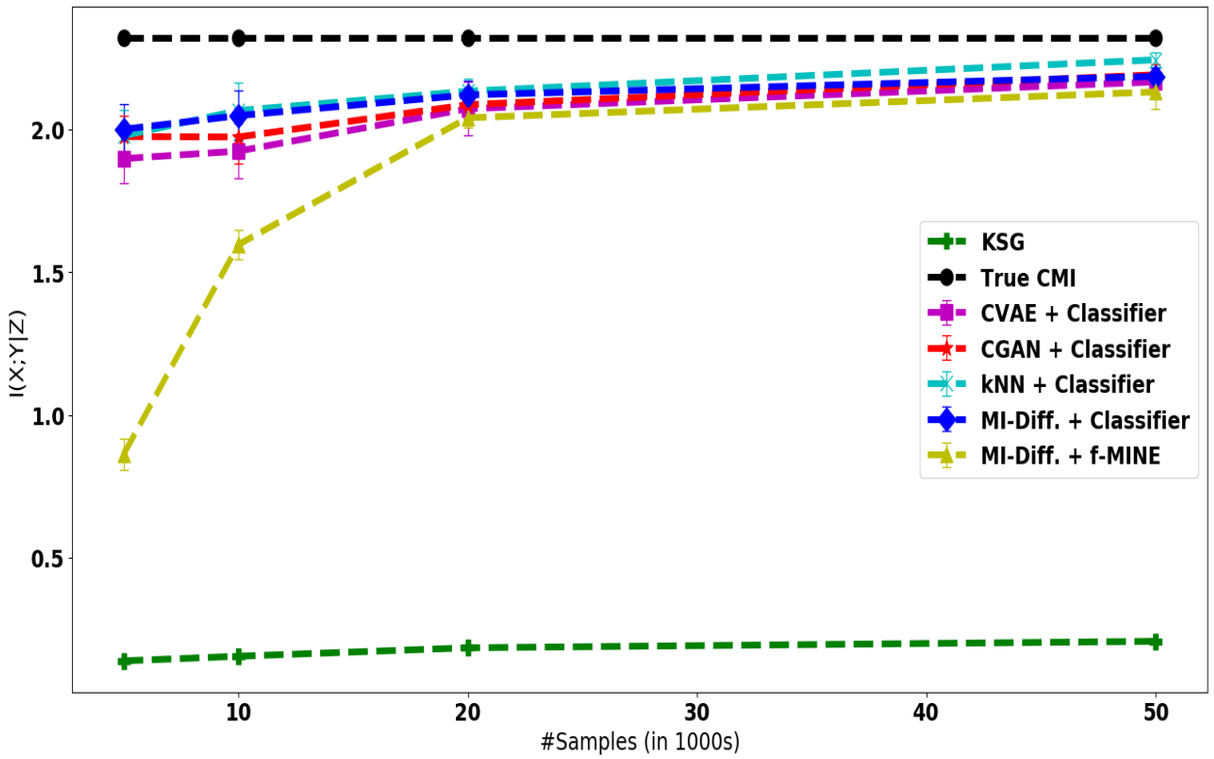

$n = 20,000$



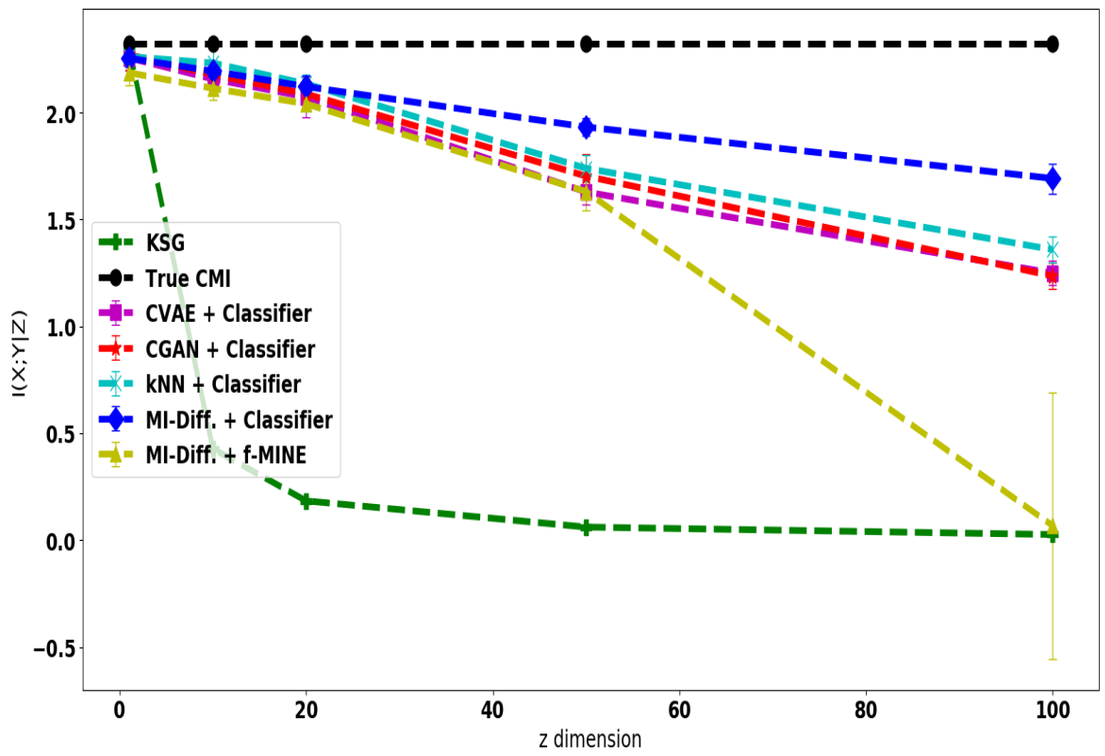
# Performance: CMI

## Model-2 (linear)

$d_z = 20$

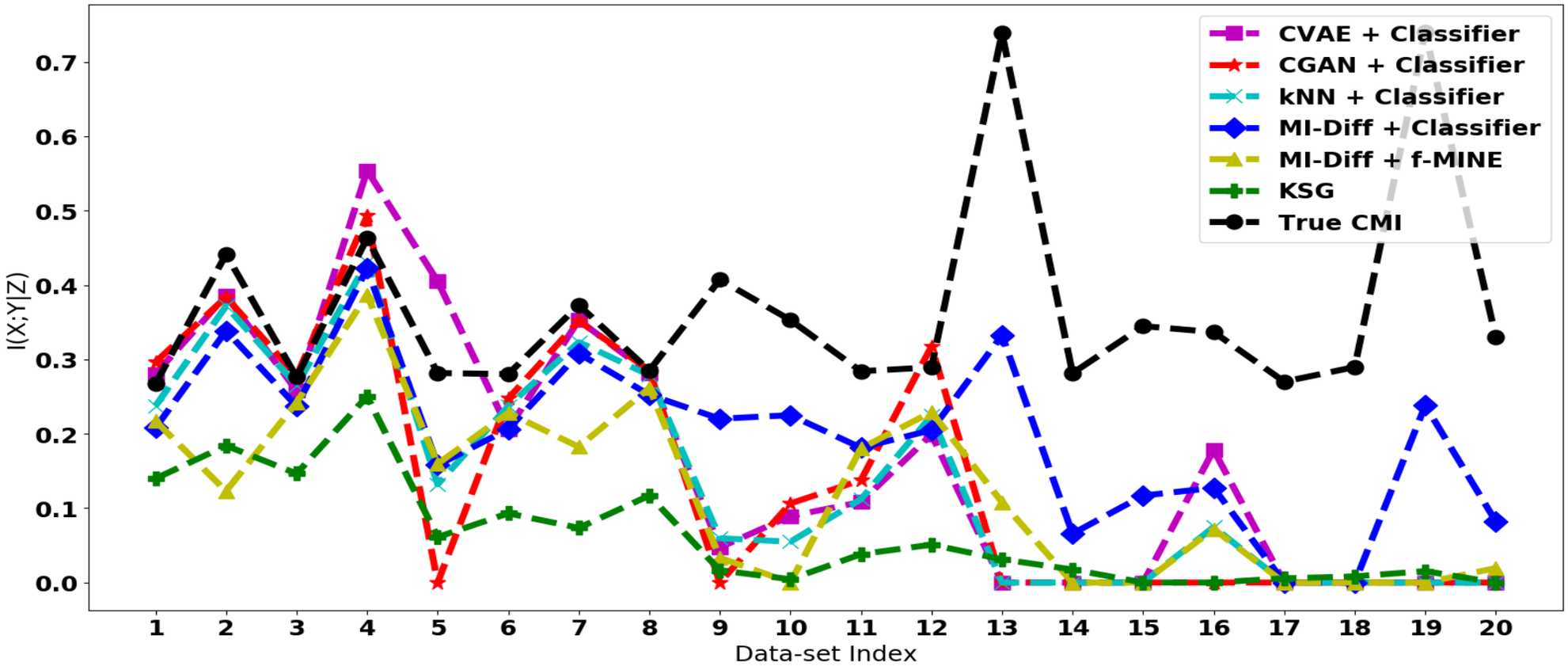



$n = 20,000$



# Performance: CMI

## Model-3 (Non-linear)

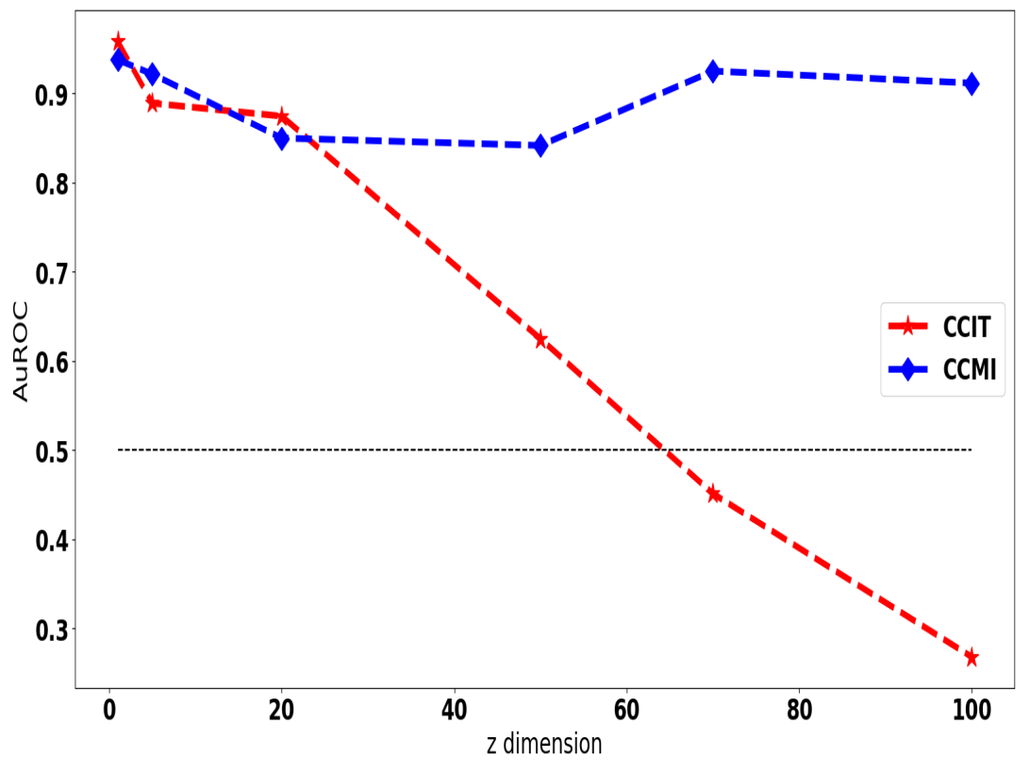


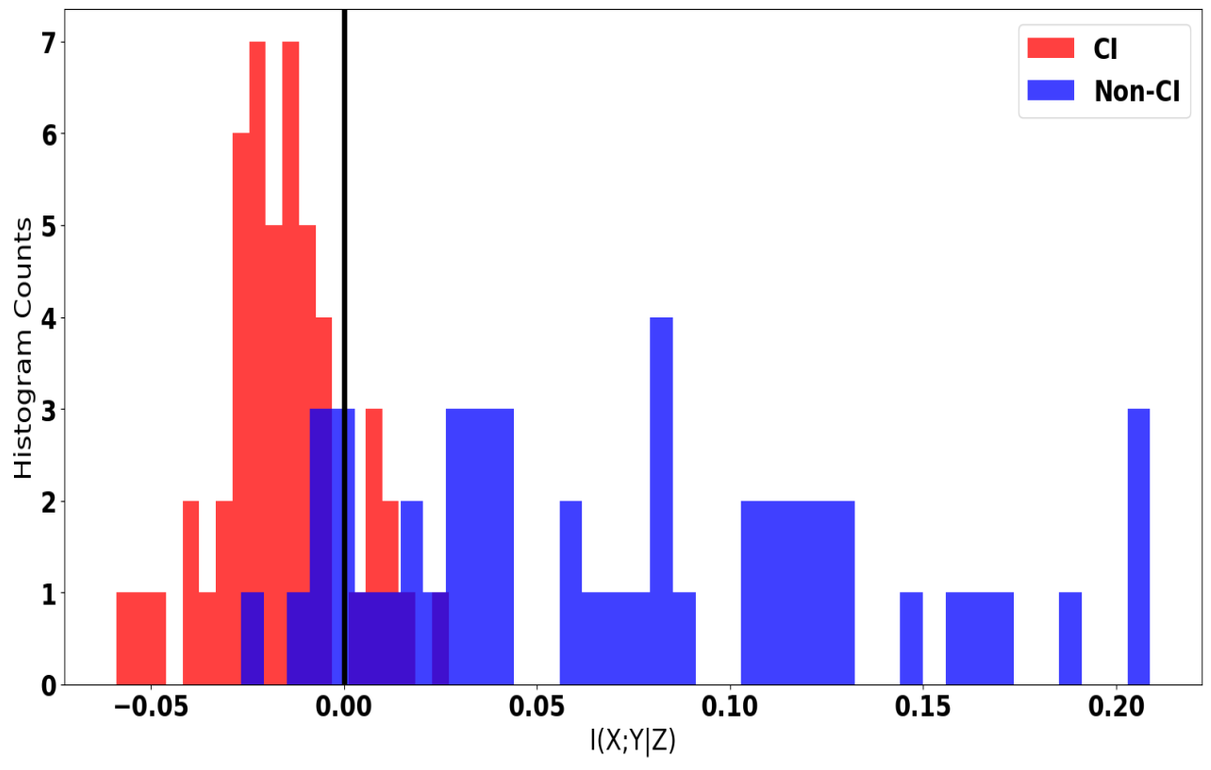
# Performance: Conditional Indep Testing

- $Z \sim \mathcal{N}(\mathbb{1}, I_{d_z}), X = \cos(a_x Z + \eta_1)$

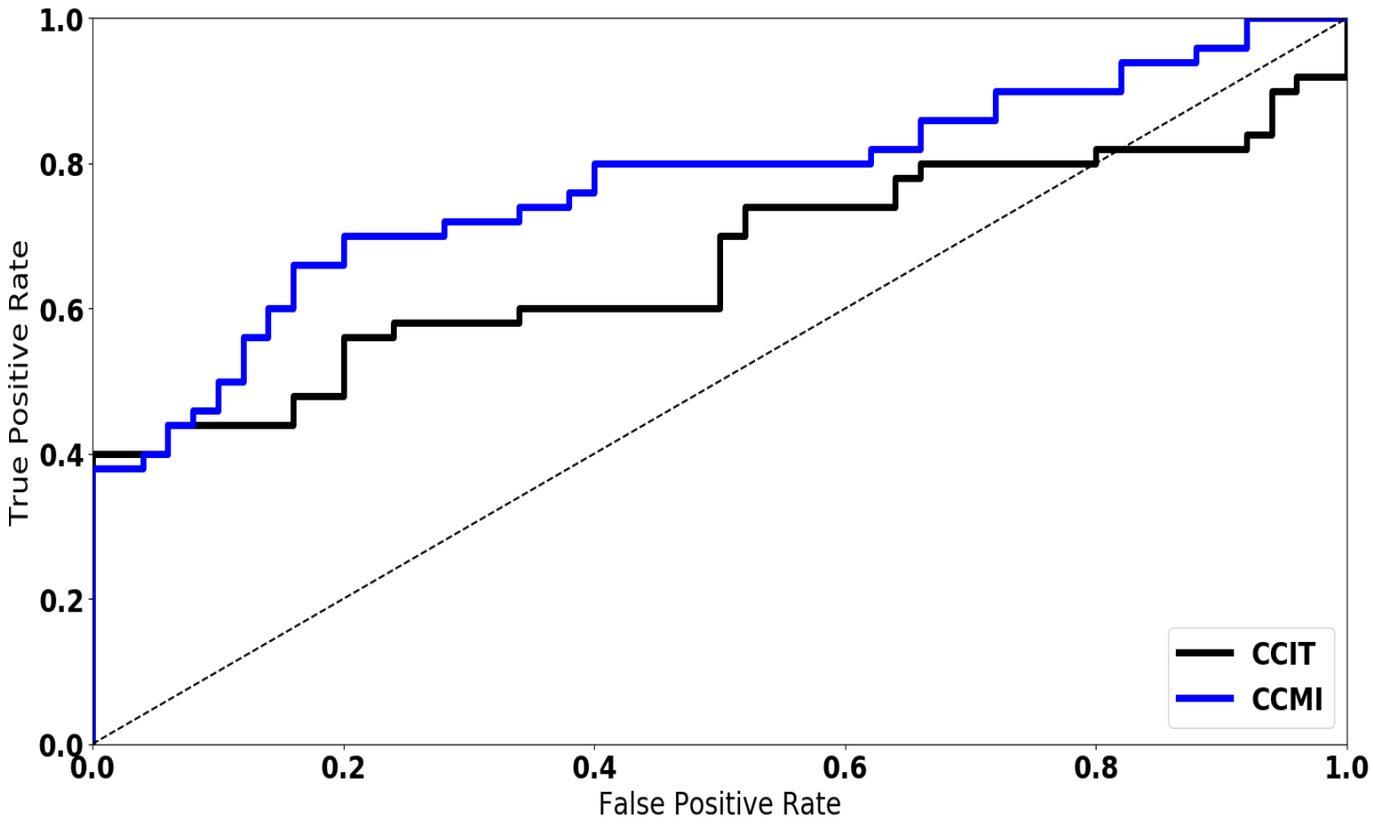
$$Y = \begin{cases} \cos(b_y Z + \eta_2) & if \quad X \perp Y \mid Z \\ \cos(cX + b_y Z + \eta_2) & if \quad X \perp Y \mid Z \end{cases}$$

$a_x, b_y \sim \mathcal{U}(0,1)^{d_z}, ||a|| = 1, \big|\big|b\big|\big| = 1,$
$c \sim \mathcal{U}(0,2), \eta_i \sim \mathcal{N}(0, 0.25).$



# Performance: Real Data (Flow Cytometry)




- 50 CI and 50 NI relations.
- Examples –
  pkc ⊥ akt | raf, mek, p38, pka, jnk
  pip3 ⊥ pip2 | p38, raf, pkc, jnk, erk, akt

- CCMI (Mean AuROC) = 0.75
  CCIT (Mean AuROC) = 0.66

# Open Problems

- Closeness testing problems
  - Beyond DTV: Distance measure estimation using classifiers?
  - Stable trainability
  - Time-series data (Directed information estimation and testing)

- Statistical property testing
  - Uniformity testing