A Permutation-Based Kernel Conditional Independence Test

Gary Doran
gary.doran@case.edu

Krikamol Muandet
krikamol@tuebingen.mpg.de

Kun Zhang
kzhang@tuebingen.mpg.de

Bernhard Schölkopf
bs@tuebingen.mpg.de

UAI 2014
Structure Learning

Learning a causal DAG requires determining conditional independence relationships.

Existing approaches like the PC algorithm (Spirtes, Glymour, & Scheines 1993) assume the existence of a conditional independence test.
Hypothesis Testing

\[ \mathcal{H}_0 : X \perp Y \mid Z \]

\[ \mathcal{H}_a : X \not\perp Y \mid Z \]

Given sample: \( \{ (x_i, y_i, z_i) \}_{i=1}^{n} \)

Desired Type I error rate: \( \alpha \)
## Desired Properties of Test

<table>
<thead>
<tr>
<th></th>
<th>Continuous Data</th>
<th>Weak Assumptions</th>
<th>High Dimensionality</th>
<th>Well-Calibrated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baba et al. 2004</td>
<td>✓</td>
<td>✗</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Margaritis 2005</td>
<td>✗</td>
<td>✓</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Fukumizu et al. 2008</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>?</td>
</tr>
<tr>
<td>Zhang et al. 2011</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>?</td>
</tr>
</tbody>
</table>
Kernel Conditional Independence Permutation Test (KCIPT)
Kernel Conditional Independence Permutation Test (KCIPT)

Permute

Induce

$X \perp Y \mid Z$
Kernel Conditional Independence Permutation Test (KCIPT)

Permute

Induce

\[ X \perp Y \mid Z \]

Two-Sample Test
How Permutations Induce Independence

\[ P(X, Y) \]
How Permutations Induce Independence

\[ P(X, Y) \]
How Permutations Induce Independence

$P(X, Y) = P(X)P(Y)$
Permuting for Conditional Independence

\[ P(X, Y, Z) \]
Permuting for Conditional Independence

\[
P(X, Y, Z) = P(X, Z)P(Y)
\]
Permuting for Conditional Independence

\begin{align*}
P(X, Y, Z) & = P(X, Z)P(Y) \\
\end{align*}
Permuting for Conditional Independence

\[ P(X, Y, Z) \]
\[ P(X, Z)P(Y) \]
\[ P(X | Z)P(Y | Z)P(Z) \]
Real-Valued $Z$

$$z_i \approx z_j \implies \Pr(Y \mid z_i) \approx \Pr(Y \mid z_j)$$

We want a permutation that, if applied to $Z$, leaves the values of $Z$ approximately invariant.

$$\delta(z, Pz) \approx 0$$

$$\delta : \mathbb{Z}^n \times \mathbb{Z}^n \to \mathbb{R} \quad \text{“Distortion Measure”}$$
Minimizing Distortion

When distortion measure is a pairwise distance:

\[ D_{ij} = d(z_i, z_j) \]

\[
\min \text{Tr}(PD) \\
P \in \text{Permutations}
\]
Minimizing Distortion

When distortion measure is a pairwise distance:

\[ D_{ij} = d(z_i, z_j) \]

\[
\begin{align*}
\min & \quad \text{Tr}(PD) \\
\text{P} & \in \text{Permutations}
\end{align*}
\]

This can be relaxed to an LP:

\[
\begin{align*}
\min & \quad \text{Tr}(PD) \\
\text{P} & \in \text{Doubly Stochastic}
\end{align*}
\]
KCIPT

Permute

\[
\min \text{Tr}(PD)
\]

P \in \text{Doubly Stochastic}

Induce

\[
X \perp Y \mid Z
\]

Two-Sample Test
Two-Sample Test
Two-Sample Test

A Kernel Two-Sample Test
(Gretton et al. 2012)
The Kernel Mean Map

Maps distributions into an RKHS:

\[
k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \quad \phi : \mathcal{X} \to \mathcal{H}
\]

\[
\mu(P) = \mathbb{E}_{x \sim P} [\phi(x)] \quad \hat{\mu}(X) = \frac{1}{|X|} \sum_{x \in X} \phi(x)
\]
A Kernel Two-Sample Test

When a kernel is characteristic (e.g. Gaussian kernel), the mean map is injective, so a two-sample test amounts to checking whether:

$$\| \hat{\mu} [(x_1, y_1, z_1)] - \hat{\mu} [(x_2, Py_2, z_2)] \|^2_{\mathcal{H}} \approx 0$$
Approach: Kernel Conditional Independence Permutation Test (KCIPT)

\[ \| \hat{\mu} \left[ (x_1, y_1, z_1) \right] - \hat{\mu} \left[ (x_2, P y_2, z_2) \right] \|_H^2 \approx 0 \]
Distortion Measure

$$\min \text{Tr}(PD)$$

$$P \in \text{Doubly Stochastic}$$

$$D_{ij} = \| \phi(z_i) - \phi(z_j) \|_\mathcal{H}$$

Produces a consistent test as long as good permutations can be found with respect to this distance metric.
Splitting the Sample

\[(x, y, z)\]

\[(x_1, y_1, z_1)\] → Split Sample

\[(x_2, y_2, z_2)\] → Learn Permutation

\[(x_2, Py_2, z_2)\] → Two-Sample Test

Test Statistic

Null Distribution
Bootstrapping the Test

\((x, y, z)\)

- Split Sample
- Learn Permutation
- Two-Sample Test

- Split Sample
- Learn Permutation
- Two-Sample Test

- Split Sample
- Learn Permutation
- Two-Sample Test

Test Statistic

Null Distribution
## Desired Properties of Test

<table>
<thead>
<tr>
<th></th>
<th>Continuous Data</th>
<th>Weak Assumptions</th>
<th>High Dimensionality</th>
<th>Well-Calibrated</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Baba et al. 2004</strong></td>
<td>✓</td>
<td>✗</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td><strong>Margaritis 2005</strong></td>
<td>✗</td>
<td>✓</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td><strong>Fukumizu et al. 2008</strong></td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>?</td>
</tr>
<tr>
<td><strong>Zhang et al. 2011</strong></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>?</td>
</tr>
<tr>
<td><strong>Our Approach</strong></td>
<td>✓</td>
<td>✓</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
**Fighting Dimensionality**

**Assumption:** $Y$ is a smooth, but nonlinear function of $Z$ plus some Gaussian noise.

$$Y = f(Z) + \mathcal{N}(0, \Sigma)$$

$$D_{ij} = \| f(z_i) - f(z_j) \|$$
## Desired Properties of Test

<table>
<thead>
<tr>
<th></th>
<th>Continuous Data</th>
<th>Weak Assumptions</th>
<th>High Dimensionality</th>
<th>Well-Calibrated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baba et al. 2004</td>
<td>✓</td>
<td>✗</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Margaritis 2005</td>
<td>✗</td>
<td>✓</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Fukumizu et al. 2008</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>?</td>
</tr>
<tr>
<td>Zhang et al. 2011</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>?</td>
</tr>
<tr>
<td>Our Approach</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>?</td>
</tr>
</tbody>
</table>
## Desired Properties of Test

<table>
<thead>
<tr>
<th></th>
<th>Continuous Data</th>
<th>Weak Assumptions</th>
<th>High Dimensionality</th>
<th>Well-Calibrated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baba et al. 2004</td>
<td>✓</td>
<td>✗</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Margaritis 2005</td>
<td>✗</td>
<td>✓</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Fukumizu et al. 2008</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>?</td>
</tr>
<tr>
<td>Zhang et al. 2011</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>?</td>
</tr>
<tr>
<td><strong>Our Approach</strong></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>?</td>
</tr>
</tbody>
</table>
Experiments: Baselines

Conditional Hilbert-Schmidt Independence Criterion (CHSIC) (Fukumizu et al. 2008)

Kernel Conditional Independence Test (KCIT) (Zhang et al. 2011)
Experiments: Metrics
Experiments: Metrics
Experiments: Metrics

(*Across 300 Trials)
Experiments: Metrics

Area Under Power Curve $\mathcal{H}_a$

PercentRejecting NullHypothesis

$\alpha$

(*Across 300 Trials)
Experiments: Metrics

Percent Rejecting Null Hypothesis

Area Under Power Curve

Calibratedness

(*Across 300 Trials)
Experiments: Datasets

Post-Nonlinear Noise:

\[ X, Y = G(F(Z_1) + \mathcal{N}(\mu, \sigma)) \]

Chaotic Hénon Map:

\[ X = (X_t^{(1)}, X_t^{(2)}), \quad Y = (Y_t^{(1)}, Y_t^{(2)}) \]

\[ X_t^{(1)} = 1.4 - X_{t-1}^{(1)} + 0.3X_{t-1}^{(2)} \]

\[ Y_t^{(1)} = 1.4 - \left[ \gamma X_{t-1}^{(1)} Y_{t-1}^{(1)} + (1 - \gamma)Y_{t-1}^{(1)} \right]^2 + 0.3Y_{t-1}^{(2)} \]

\[ X_t^{(2)} = X_{t-1}^{(1)}, \quad Y_t^{(2)} = Y_{t-1}^{(1)} \]
Results: Post-Nonlinear Noise

\[ X, Y = G(F(Z_1) + \mathcal{N}(\mu, \sigma)) \quad n = 400 \]
Results: Chaotic Time Series

\[ Y_t^{(1)} = 1.4 - \left[ \gamma X_{t-1}^{(1)} Y_{t-1}^{(1)} + (1 - \gamma) Y_{t-1}^{(1)} \right]^2 + 0.3 Y_{t-1}^{(2)} \quad n = 400 \]
## Desired Properties of Test

<table>
<thead>
<tr>
<th></th>
<th>Continuous Data</th>
<th>Weak Assumptions</th>
<th>High Dimensionality</th>
<th>Well-Calibrated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baba et al. 2004</td>
<td>✓</td>
<td>✗</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Margaritis 2005</td>
<td>✗</td>
<td>✓</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Fukumizu et al. 2008</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Zhang et al. 2011</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>Our Approach</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
Conclusions and Future Work

• We propose a new conditional independence test with several desirable properties absent in prior approaches.

• In future work, we plan to apply our approach to structure learning problems.

• Evaluate in domains with sparse data.

Code available online: http://engr.case.edu/doran_gary/code.html