PAKDD2015, Vietnam May 21, 2015

# Direct Change Detection without Identification

Masashi Sugiyama
University of Tokyo, Japan
sugi@k.u-tokyo.ac.jp
http://www.ms.k.u-tokyo.ac.jp/

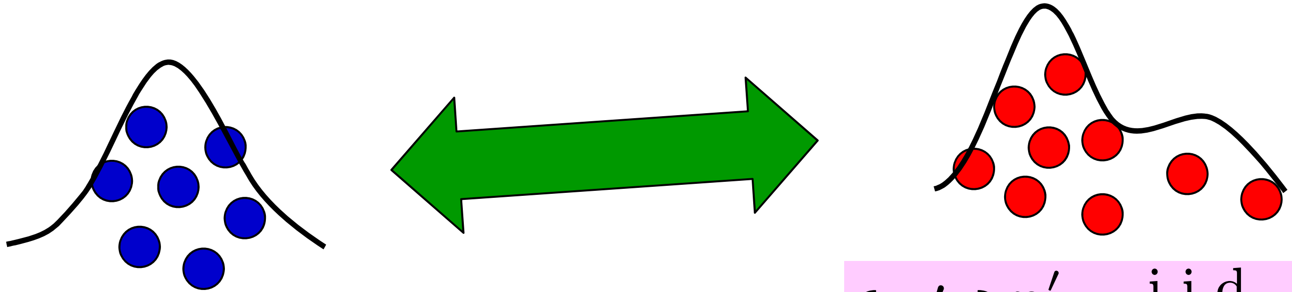
東京大学 THE UNIVERSITY OF TOKYO

大学共同利用機関法人 情報・システム研究機構
統計数理研究所
The Institute of Statistical Mathematics

Joint work with Song Liu
(my former student; now at Institute of Statistical Mathematics, Japan)

# Change Detection

- Goal: Given two sets of samples, we want to compare the probability distributions behind



$$\{\boldsymbol{x}_i\}_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}) \qquad \{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'} \overset{\text{i.i.d.}}{\sim} p'(\boldsymbol{x})$$

- Two approaches:
  - Distributional change detection: Flexible and robust
  - Structural change detection: Interpretable

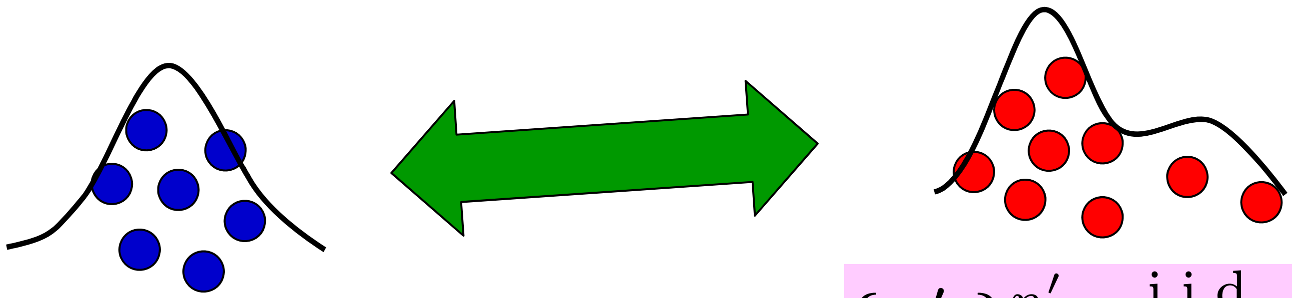
# Contents



1. Distributional change detection
   A) Problem setup and motivating examples
   B) Distances
   C) Distance Estimation
   D) Experiments
2. Structural change detection

# Distributional Change Detection

- Goal: Detect change in probability distributions behind two sets of samples through distance

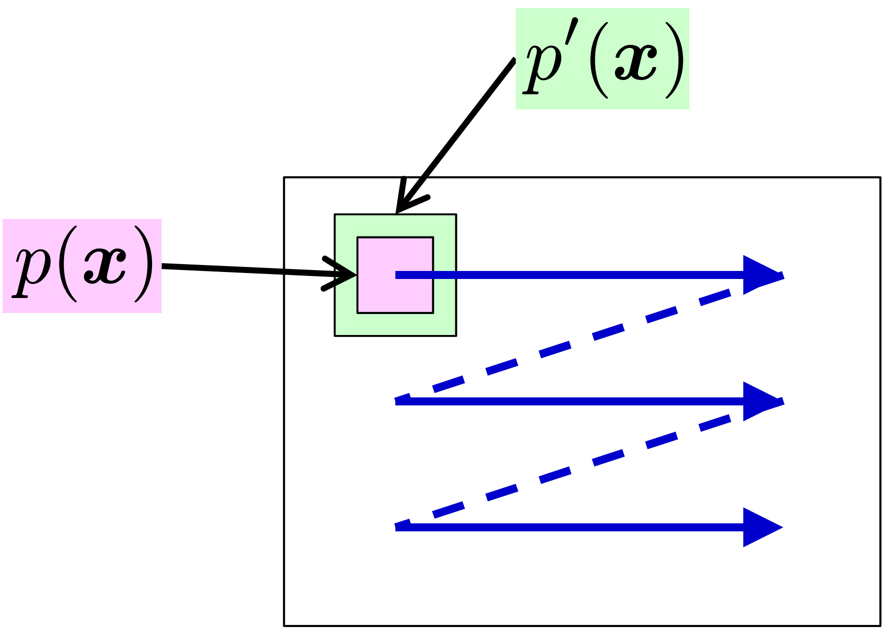


$$\{\boldsymbol{x}_i\}_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}) \qquad \{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'} \overset{\text{i.i.d.}}{\sim} p'(\boldsymbol{x})$$

$$\mathrm{Distance}(p, p') < \varepsilon \quad ?$$

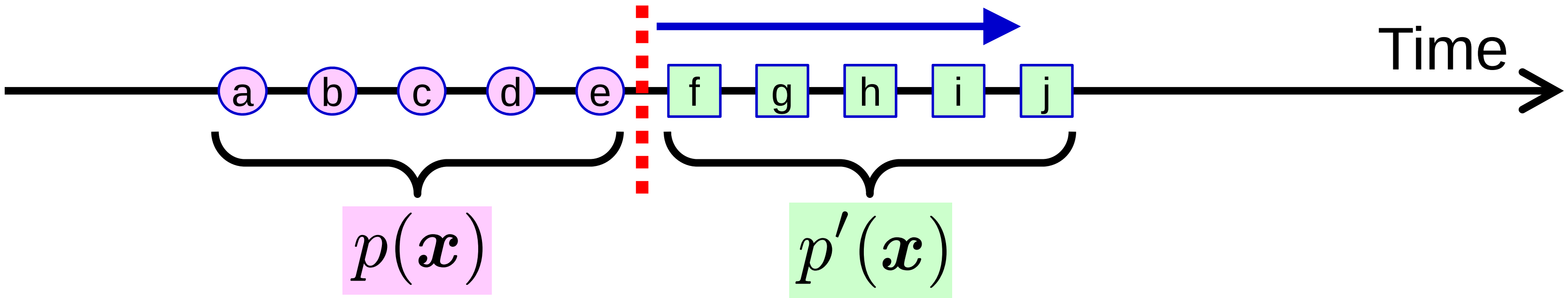
# Motivating Example 1

- Region-of-interest detection in images:
  - $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$ are significantly different when a visually salient object is included inside.



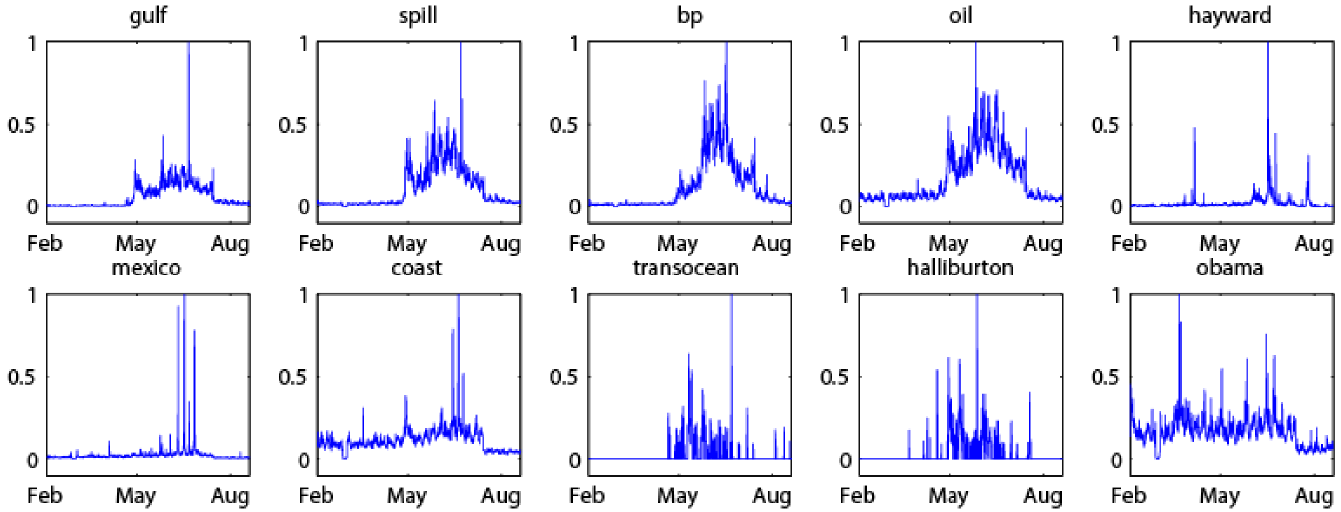
# Motivating Example 2

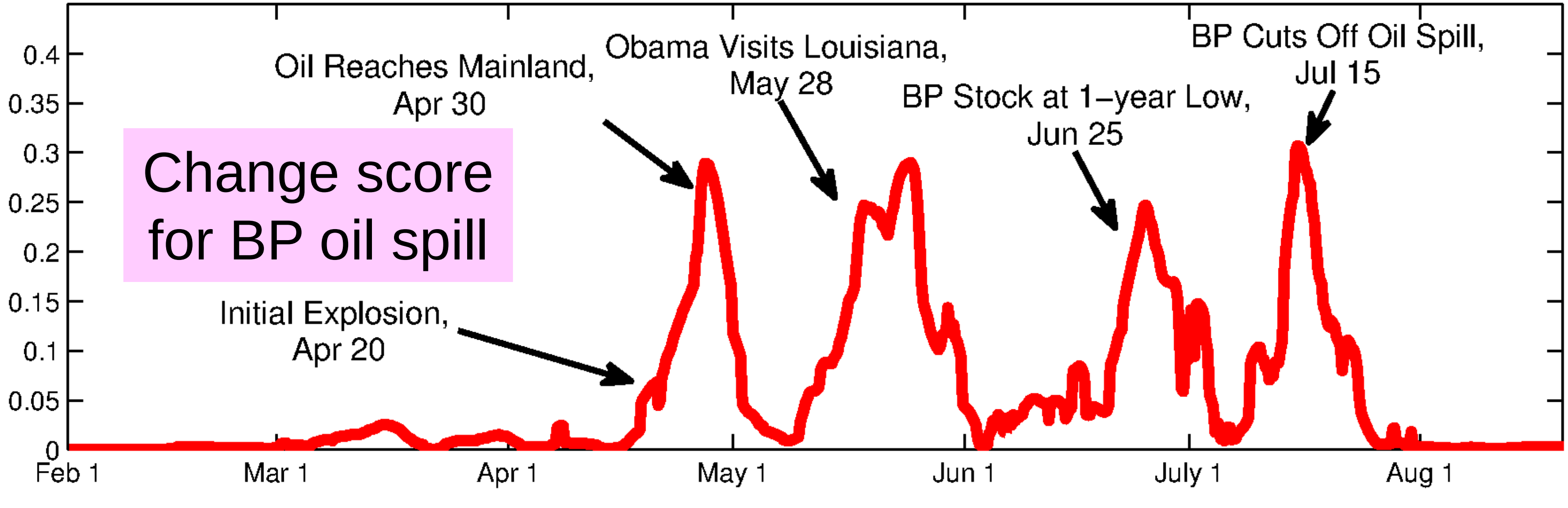
- Event detection in movies:
  - $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$ are significantly different when an irregular event occurs.



# Motivating Example 3

- Event detection from Twitter:



# Contents

1. Distributional change detection
   A) Problem setup and motivating examples
   B) Distances
   C) Distance Estimation
   D) Experiments
2. Structural change detection

$$\mathrm{Distance}(p, p') < \varepsilon \quad ?$$

# Kullback-Leibler Divergence

Kullback & Leibler (1951)

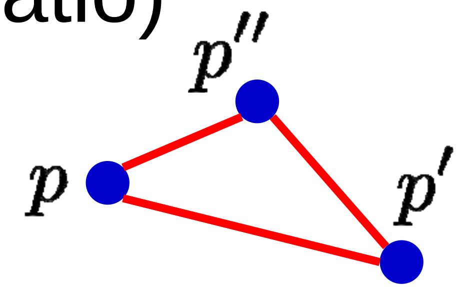
$$\mathrm{KL}(p\|p') = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})} \mathrm{d}\boldsymbol{x}$$

☺ Compatible with maximum likelihood.

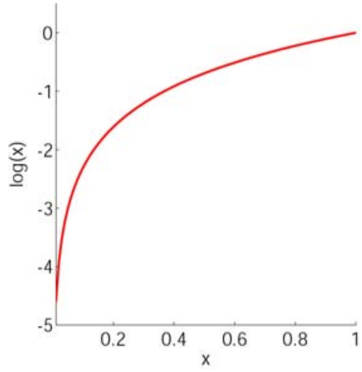
☺ Invariant under input transformation.
(Jacobians cancel in the density ratio)

$$\frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})}$$

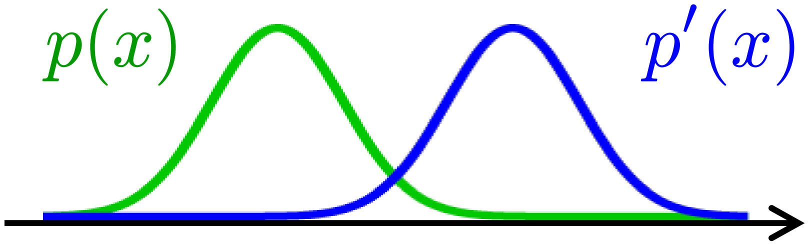
☹ Not a proper distance
(no symmetry and triangularity).



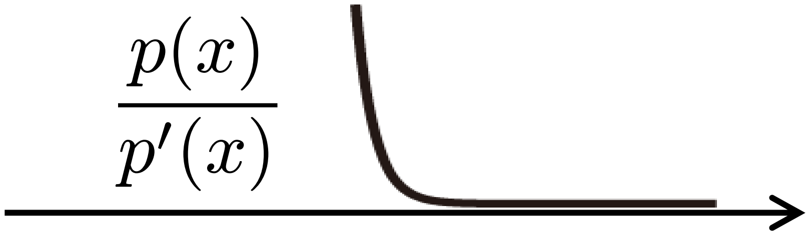

☹ Sensitive to outliers
(due to log and ratio).



# Density Ratio vs. Density Difference



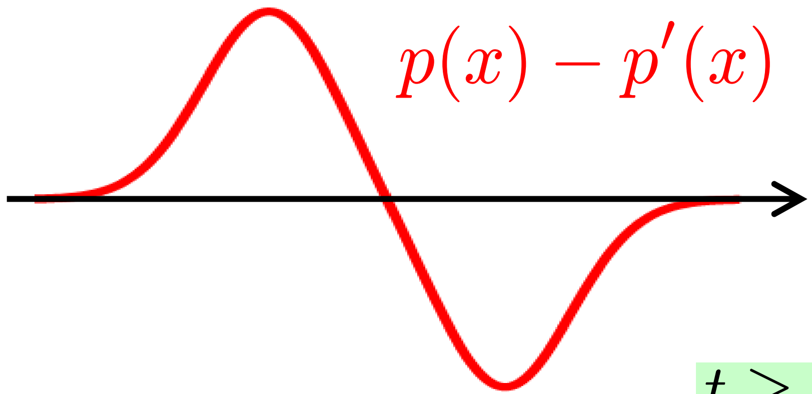

- **Density ratio based distance:**
  - Is the ratio 1?




$f$ : Convex function such that $f(1) = 0$

$$F(p\|p') = \int p'(\boldsymbol{x}) f\left(\frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})}\right) \mathrm{d}\boldsymbol{x}$$

- **Density difference based distance:**
  - Is the difference 0?




$t \geq 0$

$$L^t(p, p') = \int \left| p(\boldsymbol{x}) - p'(\boldsymbol{x}) \right|^t \mathrm{d}\boldsymbol{x}$$

# $L^2$-Distance



$$L^2(p, p') = \int \Big(p(\boldsymbol{x}) - p'(\boldsymbol{x})\Big)^2 \mathrm{d}\boldsymbol{x}$$

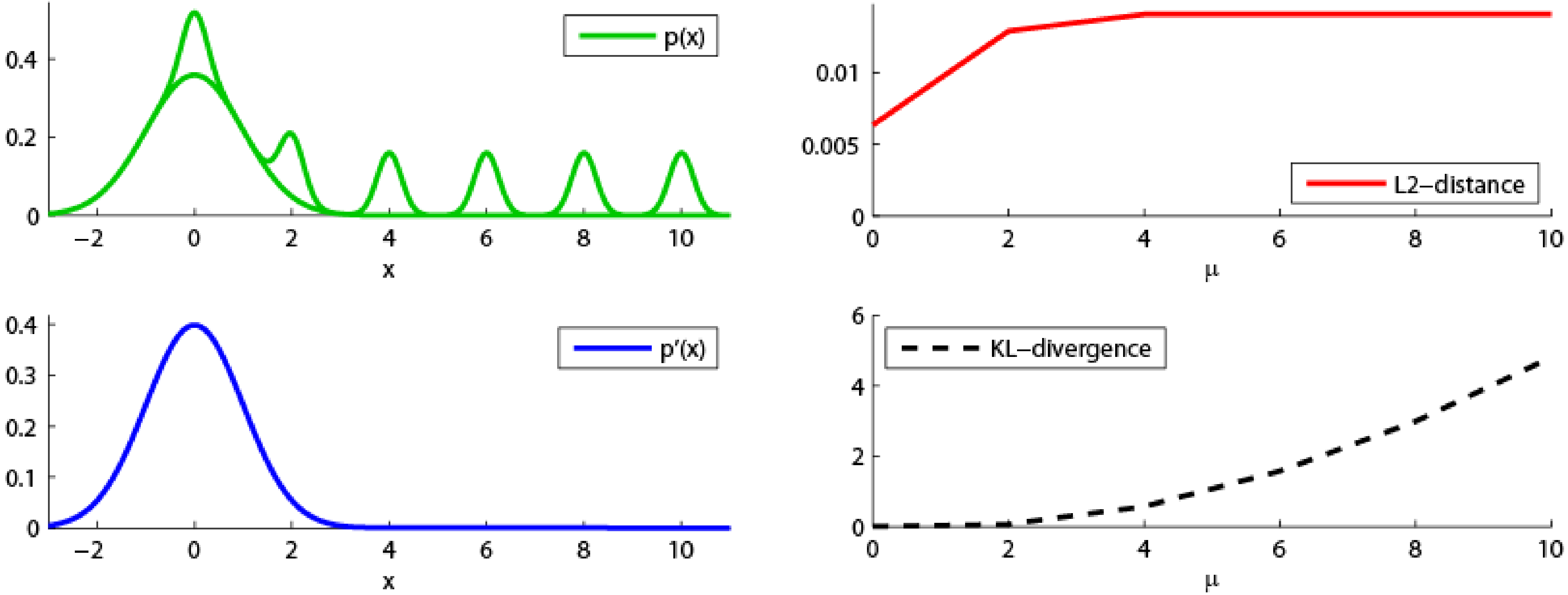
- ☺ Proper distance.
- ☺ Robust against outliers (no log, no ratio).
- ☺ Compatible with least squares.
- ☹ Not invariant under input transformation.

# KL vs. L$^2$ with Outliers

$$\mathrm{KL}(p\|p') = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})} \mathrm{d}\boldsymbol{x}$$

$$L^2(p, p') = \int \Big(p(\boldsymbol{x}) - p'(\boldsymbol{x})\Big)^2 \mathrm{d}\boldsymbol{x}$$

$$p(x) = 0.9p'(x) + 0.1q(x - \mu)$$




- L$^2$-distance is bounded.
- KL-divergence is unbounded.

# Contents

1. Distributional change detection
   - A) Problem setup and motivating examples
   - B) Distances
   - C) Distance Estimation
   - D) Experiments
2. Structural change detection

$$\{\boldsymbol{x}_i\}_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}) \qquad \{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'} \overset{\text{i.i.d.}}{\sim} p'(\boldsymbol{x})$$

$$L^2(p,p') = \int \Big(p(\boldsymbol{x}) - p'(\boldsymbol{x})\Big)^2 \mathrm{d}\boldsymbol{x}$$

# Distance Estimation via Density Estimation

1. Estimate densities $p(\boldsymbol{x}), p'(\boldsymbol{x})$ from samples:

$$\{\boldsymbol{x}_i\}_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}) \qquad \{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'} \overset{\text{i.i.d.}}{\sim} p'(\boldsymbol{x})$$

   - Maximum likelihood, Bayes, kernel smoother, nearest-neighbor, etc.

2. Plug-in the estimated densities $\widehat{p}(\boldsymbol{x}), \widehat{p}'(\boldsymbol{x})$:

$$\widehat{L}^2(p, p') = \int \left( \widehat{p}(\boldsymbol{x}) - \widehat{p}'(\boldsymbol{x}) \right)^2 \mathrm{d}\boldsymbol{x}$$

- However, this two-step method performs poorly:
  - Density estimation is performed without regards to the plug-in step performed later.

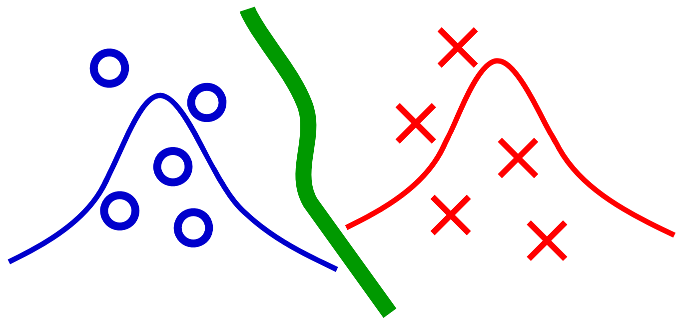
# Guiding Principle

- Vapnik's principle: Vapnik (Wiley 1998)

***When solving a problem of interest, one should not solve a more general problem as an intermediate step***

- Support vector machine avoids general density estimation and directly learns the boundary.

Cortes & Vapnik (MLJ1995)



- Let's avoid separately estimating $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$, and directly compare the densities!

# Vapnik's Principle in Distance Estimation

$$L^2(p,p') = \int \Big(p(\boldsymbol{x}) - p'(\boldsymbol{x})\Big)^2 \mathrm{d}\boldsymbol{x}$$

- Directly estimate the density difference

$$f(\boldsymbol{x}) = p(\boldsymbol{x}) - p'(\boldsymbol{x})$$

without estimating each density $p(\boldsymbol{x}), p'(\boldsymbol{x})$.

| Knowing densities $p(\boldsymbol{x}), p'(\boldsymbol{x})$ | $\rightarrow$<br>$\cancel{\leftarrow}$ | Knowing density difference $f(\boldsymbol{x}) = p(\boldsymbol{x}) - p'(\boldsymbol{x})$ |
|---|---|---|

# Least-Squares Density-Difference (LSDD) Estimation

Kim & Scott (IEEE-TPAMI2010)
Sugiyama *et al.* (NIPS2012, NeCo2013)

$$L^2(p,p') = \int f(\boldsymbol{x})^2 \mathrm{d}\boldsymbol{x}$$

$$f(\boldsymbol{x}) = p(\boldsymbol{x}) - p'(\boldsymbol{x})$$

- Directly approximate the density difference by LS:

$$\widehat{f} = \mathop{\mathrm{argmin}}_{\widetilde{f}} \int \left(\widetilde{f}(\boldsymbol{x}) - f(\boldsymbol{x})\right)^2 \mathrm{d}\boldsymbol{x}$$

$$= \mathop{\mathrm{argmin}}_{\widetilde{f}} \int \left(\widetilde{f}(\boldsymbol{x})\right)^2 \mathrm{d}\boldsymbol{x} - 2\int f(\boldsymbol{x})\widetilde{f}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$

- Expectation is approximated by empirical average.

# LSDD for Linear Model



- Linear density-difference model:

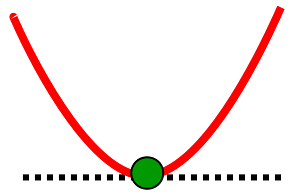
$$f_{\boldsymbol{\alpha}}(\boldsymbol{x}) = \sum_{j=1}^{b} \alpha_j \phi_j(\boldsymbol{x}) = \boldsymbol{\alpha}^\top \boldsymbol{\phi}(\boldsymbol{x})$$

$\boldsymbol{\phi}(\boldsymbol{x}) = (\phi_1(\boldsymbol{x}), \ldots, \phi_b(\boldsymbol{x}))^\top$ : Basis functions
$\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_b)^\top$: Parameters

- $\ell_2$-regularized solution is given analytically:

$$\widehat{\boldsymbol{\alpha}} = \mathop{\mathrm{argmin}}_{\boldsymbol{\alpha}} \left[ \boldsymbol{\alpha}^\top \boldsymbol{G} \boldsymbol{\alpha} - 2\widehat{\boldsymbol{h}}^\top \boldsymbol{\alpha} + \lambda \boldsymbol{\alpha}^\top \boldsymbol{\alpha} \right]$$

$$= (\boldsymbol{G} + \lambda \boldsymbol{I})^{-1} \widehat{\boldsymbol{h}}$$



$$\boldsymbol{G} = \int \boldsymbol{\phi}(\boldsymbol{x}) \boldsymbol{\phi}(\boldsymbol{x})^\top \mathrm{d}\boldsymbol{x}$$

$\lambda \geq 0$ : Regularization parameter
$\boldsymbol{I}$ : Identity matrix

$$\widehat{\boldsymbol{h}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\phi}(\boldsymbol{x}_i) - \frac{1}{n'} \sum_{i'=1}^{n'} \boldsymbol{\phi}(\boldsymbol{x}'_{i'})$$

- Scalable to big data, as long as $b$ is moderate.
- Cross-validation is possible for model selection.

# Theoretical Properties

$\{\boldsymbol{x}_i\}_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x})$

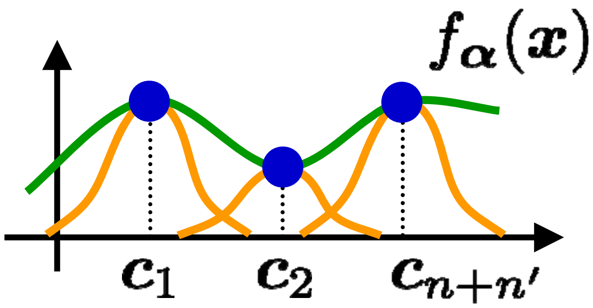
$\{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'} \overset{\text{i.i.d.}}{\sim} p'(\boldsymbol{x})$

## Parametric convergence:

- Learned parameter converges to the optimal value with rate $\sqrt{\frac{1}{n}+\frac{1}{n'}}$, which is optimal.

## Non-parametric convergence:

$$f_{\boldsymbol{\alpha}}(\boldsymbol{x}) = \sum_{j=1}^{n+n'} \alpha_j \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{c}_j\|^2}{2\sigma^2}\right)$$




$$(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_{n+n'}) = (\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n,\boldsymbol{x}'_1,\ldots,\boldsymbol{x}'_{n'})$$

- Learned function converges to the optimal function with rate $n^{-\frac{2\beta}{2\beta+\dim(\boldsymbol{x})}}$ ($\beta \geq 0$ represents a complexity of the true function), which is mini-max optimal.

$n = n'$

# $L^2$-Distance Estimation

$$f(\boldsymbol{x}) = p(\boldsymbol{x}) - p'(\boldsymbol{x}) \approx \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{\phi}(\boldsymbol{x}) \qquad \widehat{\boldsymbol{\alpha}} = (\boldsymbol{G} + \lambda \boldsymbol{I})^{-1} \widehat{\boldsymbol{h}}$$

- Two ways to approximate the $L^2$-distance based on LSDD:
  - $$L^2(p,p') = \int f(\boldsymbol{x})^2 \mathrm{d}\boldsymbol{x} \approx \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{G} \widehat{\boldsymbol{\alpha}}$$

    $$\boldsymbol{G} = \int \boldsymbol{\phi}(\boldsymbol{x}) \boldsymbol{\phi}(\boldsymbol{x})^\top \mathrm{d}\boldsymbol{x}$$
  - $$L^2(p,p') = \int \Big( p(\boldsymbol{x}) - p'(\boldsymbol{x}) \Big) f(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \approx \widehat{\boldsymbol{h}}^\top \boldsymbol{\alpha}$$

    $$\widehat{\boldsymbol{h}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\phi}(\boldsymbol{x}_i) - \frac{1}{n'} \sum_{i'=1}^{n'} \boldsymbol{\phi}(\boldsymbol{x}'_{i'})$$

# Bias Reduction



- Consider their linear combination:

$$\kappa \widehat{\boldsymbol{h}}^\top \widehat{\boldsymbol{\alpha}} + (1-\kappa)\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{G} \widehat{\boldsymbol{\alpha}} \qquad \kappa \in \mathbb{R}$$

  - For small regularization parameter $\lambda$,

$$\kappa \widehat{\boldsymbol{h}}^\top \widehat{\boldsymbol{\alpha}} + (1-\kappa)\widehat{\boldsymbol{\alpha}}^\top \boldsymbol{G} \widehat{\boldsymbol{\alpha}}$$

$$= \widehat{\boldsymbol{h}}^\top \boldsymbol{G}^{-1} \widehat{\boldsymbol{h}} - \lambda(2-\kappa)\widehat{\boldsymbol{h}}^\top \boldsymbol{G}^{-2} \widehat{\boldsymbol{h}} + o_p(\lambda)$$

  - $\kappa = 2$ removes the regularization-induced bias:

$$\widehat{L}^2(\mathcal{X}, \mathcal{X}') = 2\widehat{\boldsymbol{h}}^\top \widehat{\boldsymbol{\alpha}} - \widehat{\boldsymbol{\alpha}}^\top \boldsymbol{G} \widehat{\boldsymbol{\alpha}}$$

# A Few Lines in MATLAB!

$$\widehat{\boldsymbol{\alpha}} = (\boldsymbol{G} + \lambda \boldsymbol{I})^{-1} \widehat{\boldsymbol{h}}$$

$$f_{\boldsymbol{\alpha}}(\boldsymbol{x}) = \sum_{j=1}^{n+n'} \alpha_j \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{c}_j\|}{2\sigma^2}\right)$$

$$(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{n+n'}) = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_{n'})$$

$$G_{j,j'} = (\pi\sigma^2)^{\dim(\boldsymbol{x})/2} \exp\left(-\frac{\|\boldsymbol{c}_j - \boldsymbol{c}_{j'}\|^2}{4\sigma^2}\right)$$

$$\widehat{h}_j = \frac{1}{n} \sum_{i=1}^{n} \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{c}_j\|^2}{2\sigma^2}\right) - \frac{1}{n'} \sum_{i'=1}^{n'} \exp\left(-\frac{\|\boldsymbol{x}'_{i'} - \boldsymbol{c}_j\|^2}{2\sigma^2}\right)$$

```
% Data generation
n=100; x=randn(1,n/2); y=randn(1,n/2)+1; z=[x y];
% LSDD
a=repmat(z.^2,n,1); b=a+a'-2*z'*z; G=sqrt(pi)*exp(-b/4);
h=mean(exp(-b(:,1:n/2)/2),2)-mean(exp(-b(:,n/2+1:n)/2),2);
t=(G+0.1*eye(n))¥h; plot(z,G*t,'*'); L2=2*t'*h-t'*G*t
```

# Contents

1. Distributional change detection
   - A) Problem setup and motivating examples
   - B) Distances
   - C) Distance Estimation
   - D) Experiments
2. Structural change detection

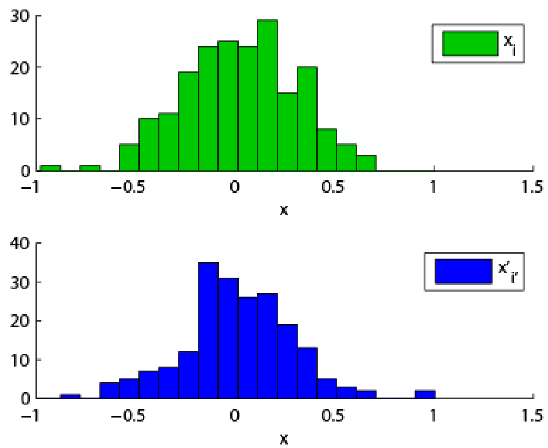
```
% Data generation
n=100; x=randn(1,n/2); y=randn(1,n/2)+1; z=[x y];
% LSDD
a=repmat(z.^2,n,1); b=a+a'-2*z'*z; G=sqrt(pi)*exp(-b/4);
h=mean(exp(-b(:,1:n/2)/2),2)-mean(exp(-b(:,n/2+1:n)/2),2);
t=(G+0.1*eye(n))¥h; plot(z,G*t,'*'); L2=2*t'*h-t'*G*t
```

# Density-Difference Estimation 1

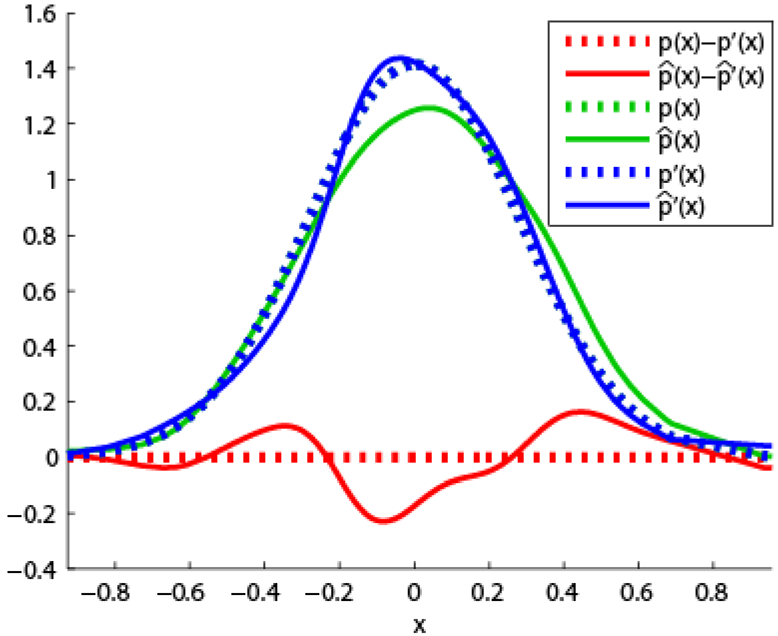


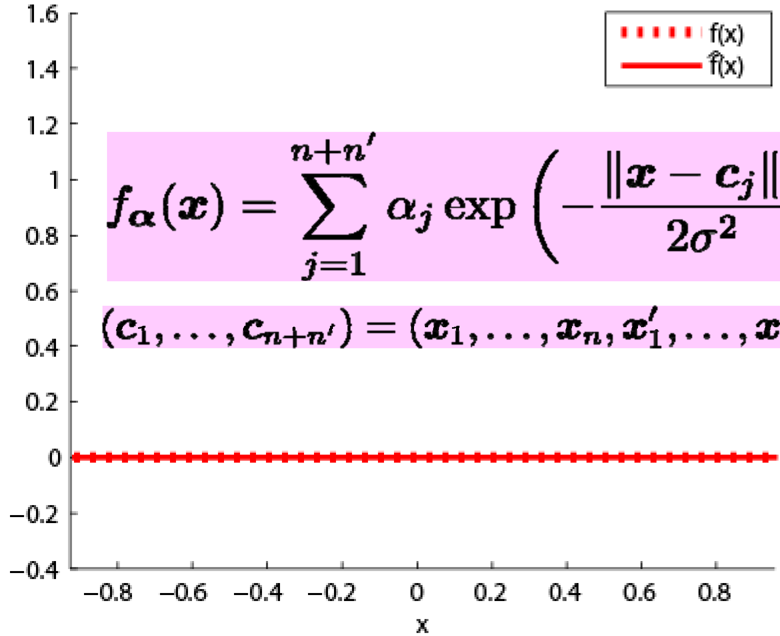
- $p(x) = p'(x) = N(x; 0, (4\pi)^{-1})$

$$n = n' = 200$$




| Difference of kernel density estimators (KDE) | Least-squares density-difference estimation (LSDD) |
| --- | --- |



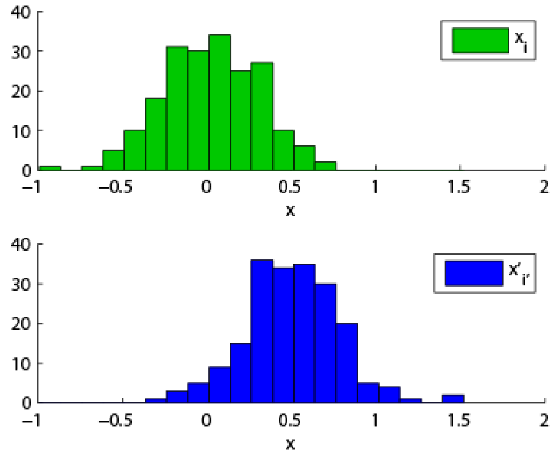




$$f_{\boldsymbol{\alpha}}(\boldsymbol{x}) = \sum_{j=1}^{n+n'} \alpha_j \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{c}_j\|^2}{2\sigma^2}\right)$$

$$(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{n+n'}) = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_{n'})$$

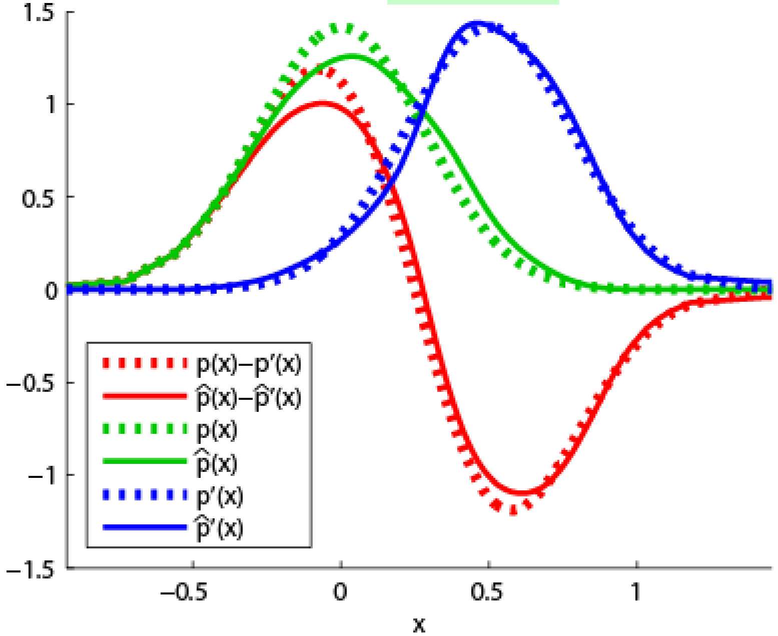
# Density-Difference Estimation 2

- $p(x) = N(x; 0, (4\pi)^{-1})$
- $p'(x) = N(x; 0.5, (4\pi)^{-1})$

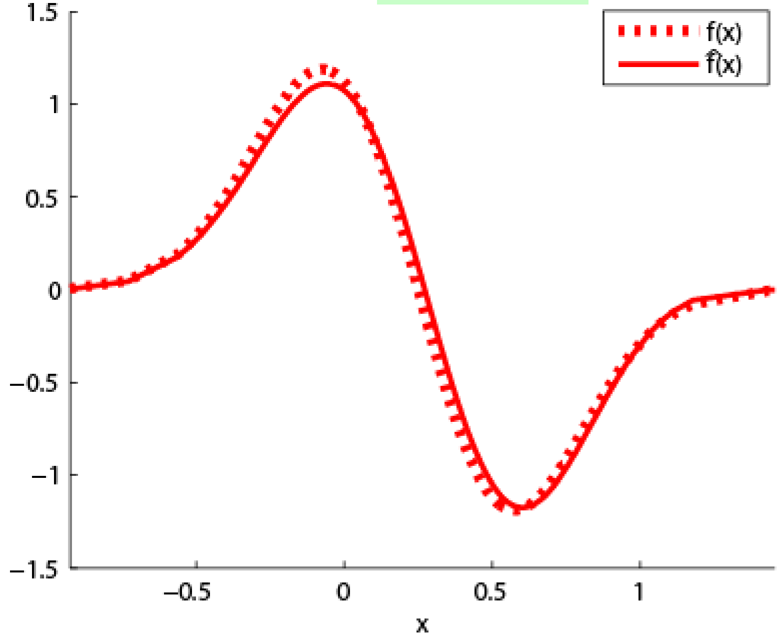



$$n = n' = 200$$

KDE



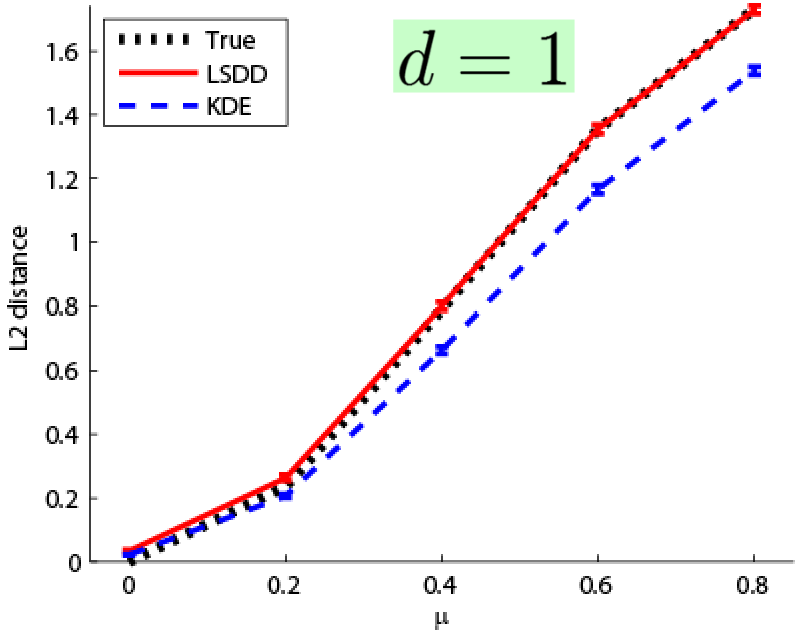

LSDD

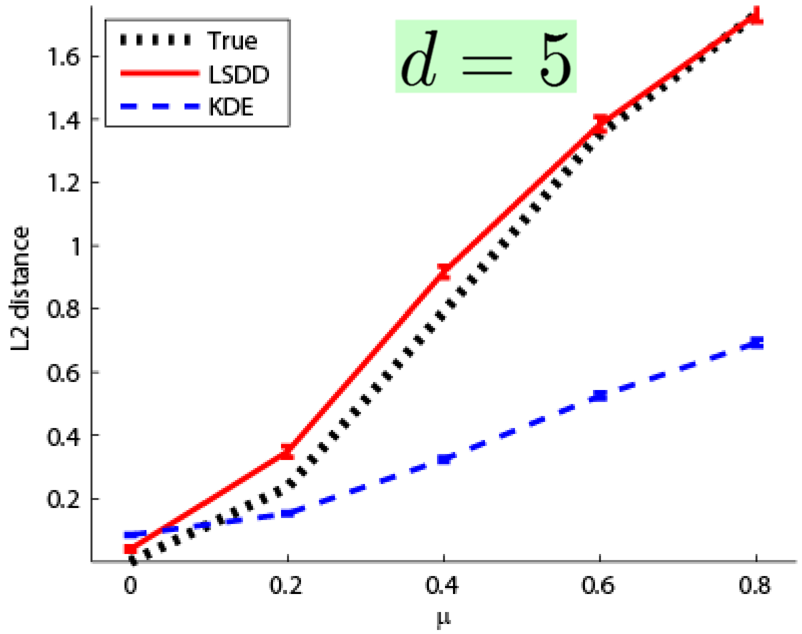


# $L^2$-Distance Estimation

- $p(\boldsymbol{x}) = N(\boldsymbol{x}; (\mu, 0, \ldots, 0)^\top, (4\pi)^{-1}\boldsymbol{I}_d)$
- $p'(\boldsymbol{x}) = N(\boldsymbol{x}; (0, 0, \ldots, 0)^\top, (4\pi)^{-1}\boldsymbol{I}_d)$

$n = n' = 100$



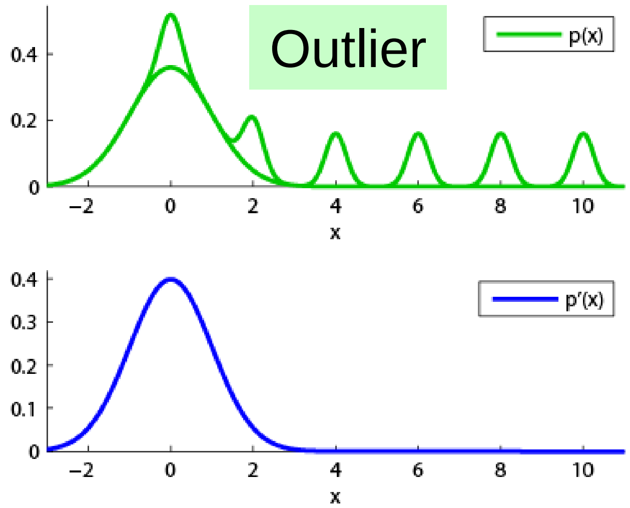



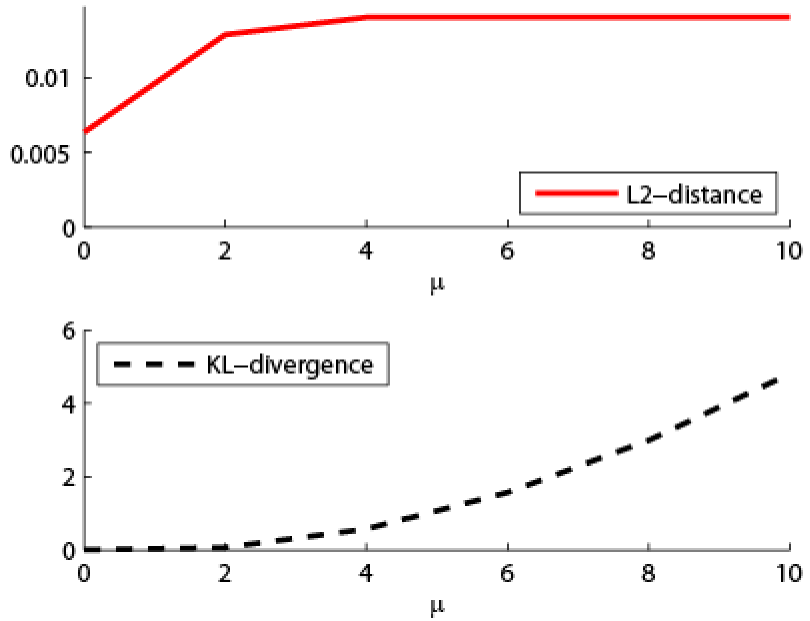

- KDE significantly under-estimates.
- LSDD slightly over-estimates.

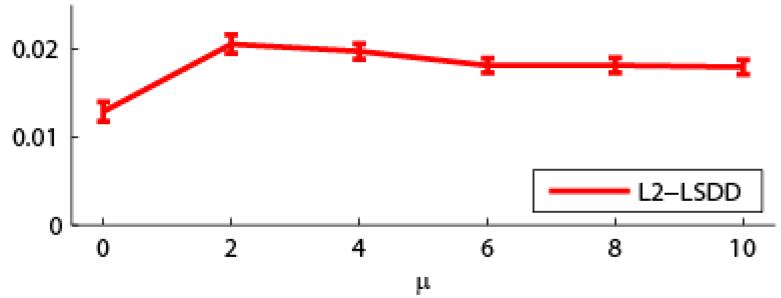
# $L^2$-Distance vs. KL-Divergence 

$$L^2(p, p') = \int \Big( p(\boldsymbol{x}) - p'(\boldsymbol{x}) \Big)^2 \mathrm{d}\boldsymbol{x}$$

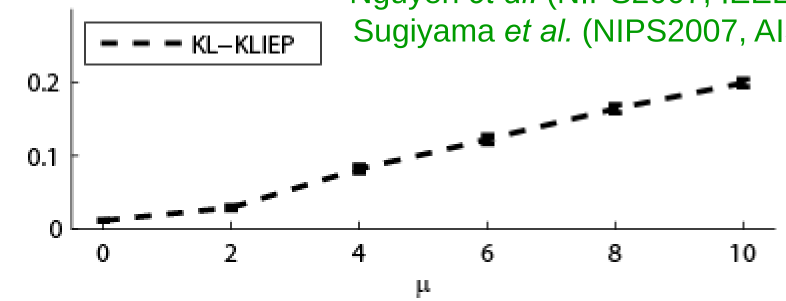
$$\mathrm{KL}(p \| p') = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})} \mathrm{d}\boldsymbol{x}$$




Nguyen *et al.* (NIPS2007, IEEE-IT2010)
Sugiyama *et al.* (NIPS2007, AISM2008)

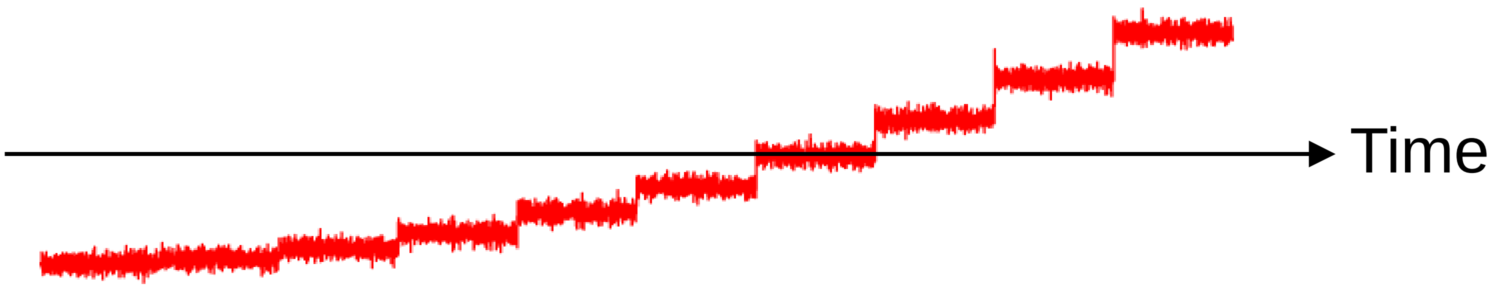



$L^2$-distance is less sensitive to outliers.

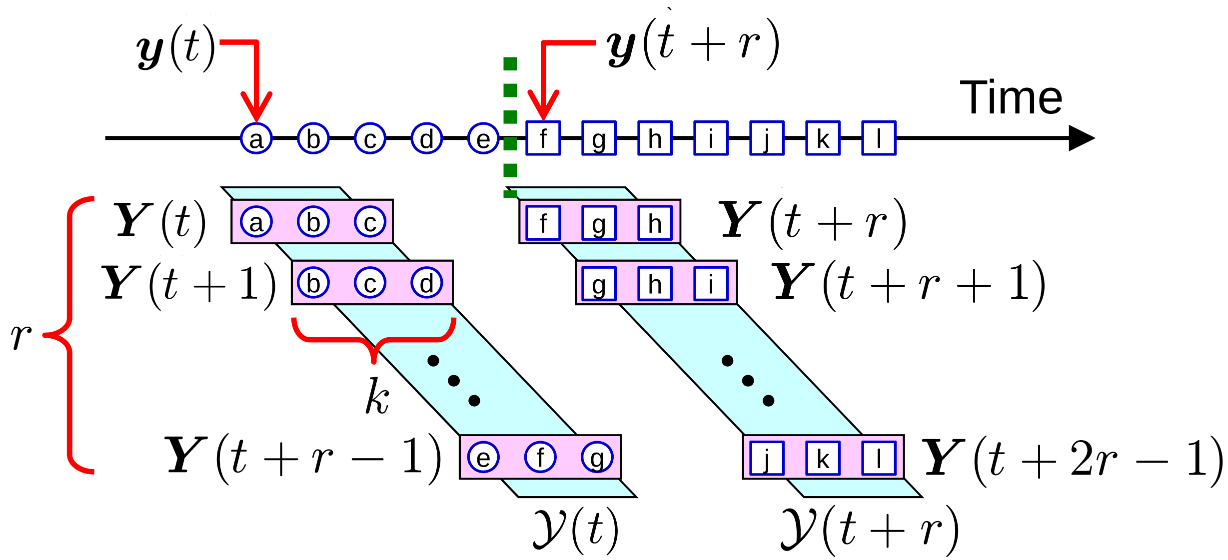
# Unsupervised Change Detection



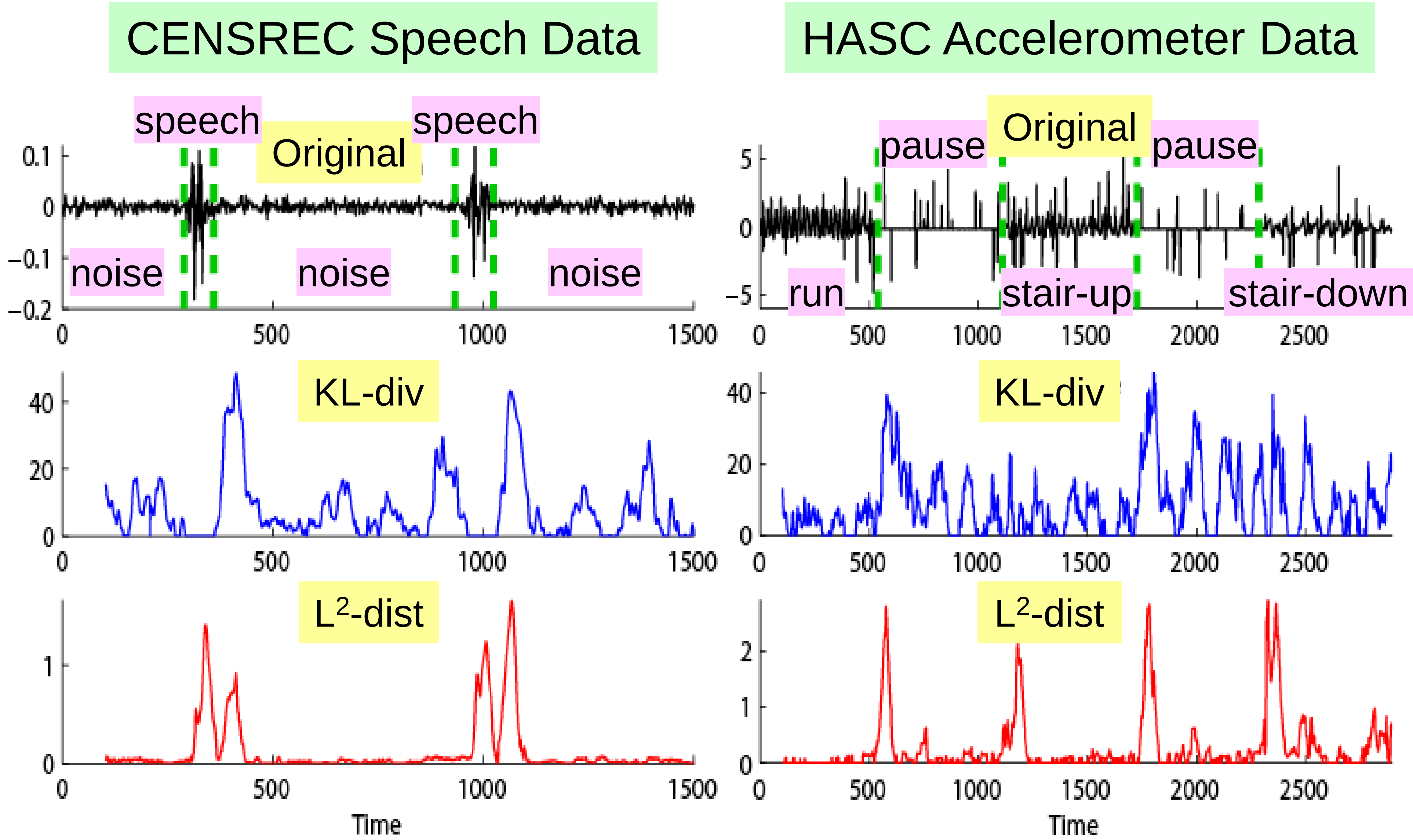
- Identify change points in time-series:




- Use the distance between the distributions of sliding-windowed past and current data.



# Results




L$^2$-distance is more robust!

# Summary of Distributional Change Detection

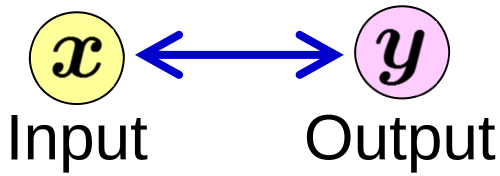
- Distance estimation between distributions:
  - Separate density estimation works poorly.
  - Direct density-difference estimation seems sensible.
- Don't simply use KL just because it is popular.
  - $L^2$-distance could be more robust against outliers and computationally more efficient.
- Quadratic mutual information (QMI) can be approximated by LSDD similarly:

$$\mathrm{QMI} = \iint \Big( p(\boldsymbol{x}, \boldsymbol{y}) - p(\boldsymbol{x})p(\boldsymbol{y}) \Big)^2 \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}$$

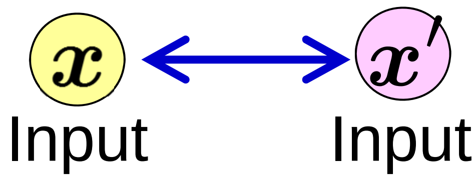
# Usages of QMI

$$\mathrm{QMI} = \iint \left(p(\boldsymbol{x},\boldsymbol{y}) - p(\boldsymbol{x})p(\boldsymbol{y})\right)^2 \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}$$

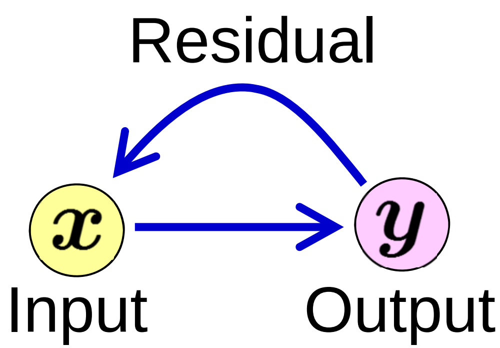
- QMI between input and output:
  - Feature selection/extraction
  - Clustering




- QMI between inputs:
  - Independent component analysis
  - Higher-order canonical correlation analysis
  - Unsupervised object matching




- QMI between input and residual:
  - Causal direction inference

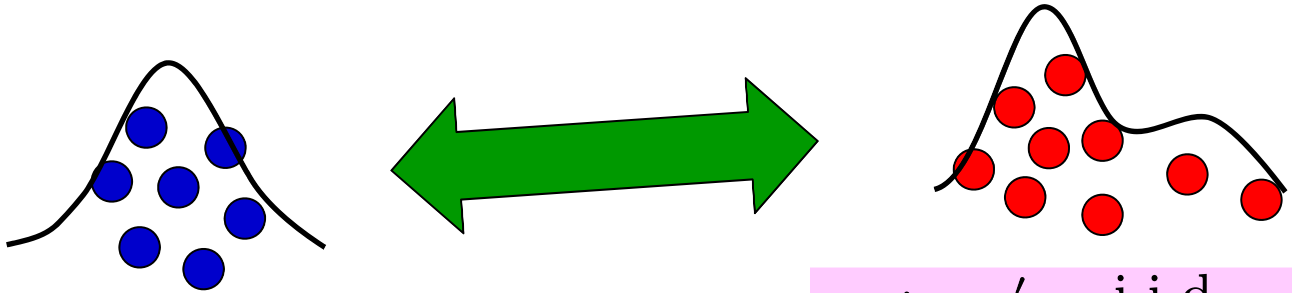


# Contents



1. Distributional change detection
2. Structural change detection
   A) Density estimation approach
   B) Density-ratio estimation approach

# From Distributional Change to Structural Change



$$\{\boldsymbol{x}_i\}_{i=1}^{n} \stackrel{\text{i.i.d.}}{\sim} p(\boldsymbol{x})$$

$$\{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'} \stackrel{\text{i.i.d.}}{\sim} p'(\boldsymbol{x})$$

- Through distance estimation, distributional change can be detected.
- Let's investigate how distributions are changed through interaction between variables.

$$\boldsymbol{x} = (x^{(1)}, \ldots, x^{(d)})^\top$$

# Motivating Examples

- Word co-occurrence in Twitter
- Gene regulatory networks
- Fraud detection in smart grid

# Contents

# Gaussian Model

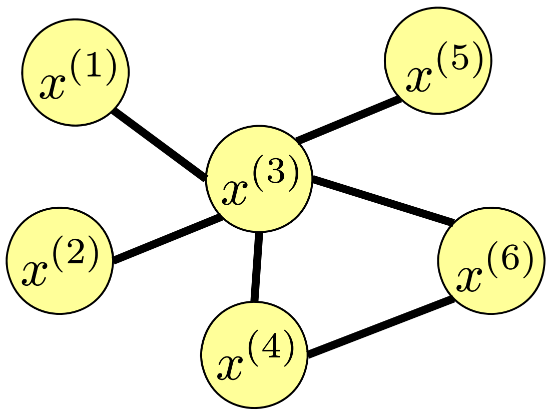
$$q(\boldsymbol{x};\boldsymbol{\Theta}) = \frac{\det(\boldsymbol{\Theta})^{1/2}}{(2\pi)^{d/2}} \exp\left(-\frac{1}{2}\boldsymbol{x}^\top \boldsymbol{\Theta}\boldsymbol{x}\right)$$

$\boldsymbol{\Theta}$ : (sparse) inverse covariance matrix

- Conditional independence: $\boldsymbol{x} = (x^{(1)}, \ldots, x^{(d)})^\top$

$$\Theta_{k,k'} = 0 \iff x^{(k)} \perp\!\!\!\perp x^{(k')} \mid \{x^{(\ell)}\}_{\ell \neq k,k'}$$

- Graphical representation:
  - Node: Each variable
  - Edge: Exists if $\Theta_{i,j} \neq 0$
  - Only connected variables affect!



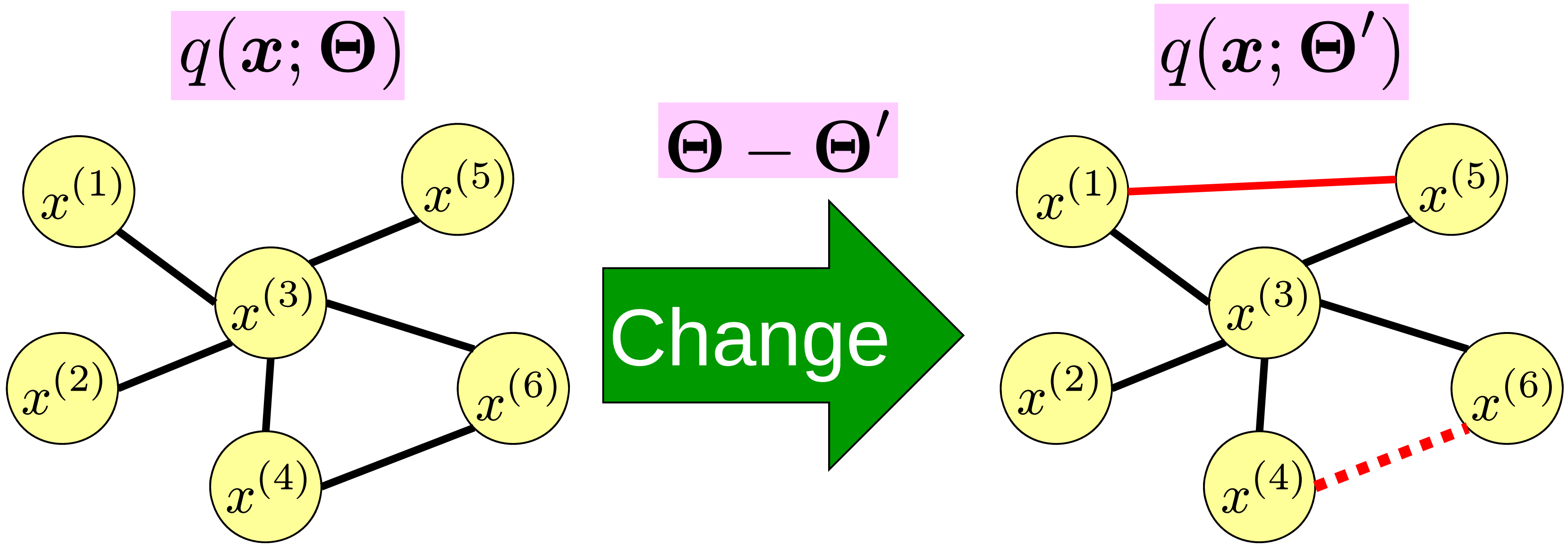

$$x^{(1)} \perp\!\!\!\perp x^{(2)} \mid x^{(3)}$$

# Structural Change Detection with Gaussian Models

- Use Gaussian models for $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$ :

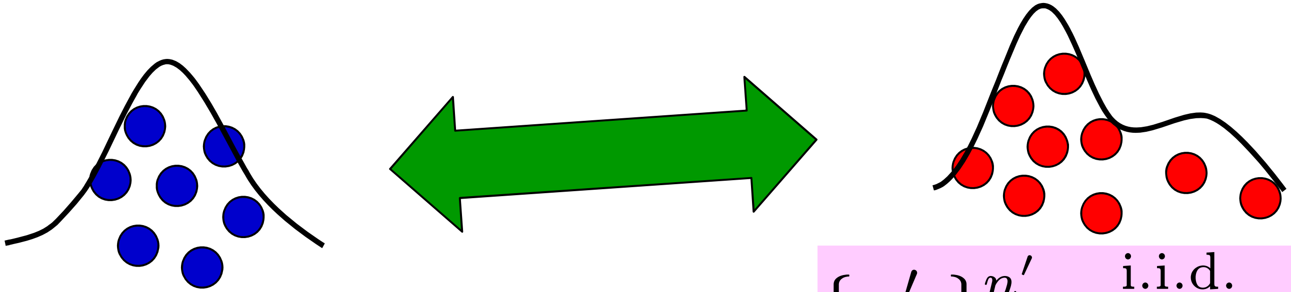
$$q(\boldsymbol{x};\boldsymbol{\Theta}) = \frac{\det(\boldsymbol{\Theta})^{1/2}}{(2\pi)^{d/2}} \exp\left(-\frac{1}{2}\boldsymbol{x}^\top \boldsymbol{\Theta}\boldsymbol{x}\right) \qquad q(\boldsymbol{x};\boldsymbol{\Theta}')$$

- Detect sparse change in covariance structure:



# Structural Change Detection by Graphical Lasso (Glasso)

Tibshirani (JRSS1996), Friedman *et al.* (Biostat2008)



$$\{\boldsymbol{x}_i\}_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}) \qquad \{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'} \overset{\text{i.i.d.}}{\sim} p'(\boldsymbol{x})$$

- Sparse maximum likelihood estimation:

$$\max_{\boldsymbol{\Theta}} \sum_{i=1}^{n} \log q(\boldsymbol{x}_i; \boldsymbol{\Theta}) - \lambda \|\boldsymbol{\Theta}\|_1 \qquad \max_{\boldsymbol{\Theta}'} \sum_{i'=1}^{n'} \log q(\boldsymbol{x}'_{i'}; \boldsymbol{\Theta}') - \lambda' \|\boldsymbol{\Theta}'\|_1$$

$$q(\boldsymbol{x}; \boldsymbol{\Theta}) = \frac{\det(\boldsymbol{\Theta})^{1/2}}{(2\pi)^{d/2}} \exp\left(-\frac{1}{2}\boldsymbol{x}^\top \boldsymbol{\Theta} \boldsymbol{x}\right) \qquad \lambda, \lambda' \geq 0$$

# Structural Change Detection by Glasso

$$\max_{\boldsymbol{\Theta}} \sum_{i=1}^{n} \log q(\boldsymbol{x}_i; \boldsymbol{\Theta}) - \lambda \|\boldsymbol{\Theta}\|_1 \qquad \max_{\boldsymbol{\Theta}'} \sum_{i'=1}^{n'} \log q(\boldsymbol{x}'_{i'}; \boldsymbol{\Theta}') - \lambda' \|\boldsymbol{\Theta}'\|_1$$

- ☺ Scalable to high-dimensional datasets.
- ☺ Statistical properties have been well studied.
  (sparse graphs can be easily recovered)

Ravikumar *et al.* (AS2010)

- ☹ Does not work if true $\boldsymbol{\Theta}$ and $\boldsymbol{\Theta}'$ are dense.

Both $\boldsymbol{\Theta}$ and $\boldsymbol{\Theta}'$ are sparse

Change $\boldsymbol{\Theta} - \boldsymbol{\Theta}'$ is sparse

- ☹ Choice of $\lambda$ and $\lambda'$ is not straightforward.

# Structural Change Detection by Fused Lasso (Flasso)

Tibshirani *et al.* (JRSS2005)
Zhang & Wang (UAI2010)

- Directly penalize the difference of parameters to be sparse:

$$\max_{\boldsymbol{\Theta},\boldsymbol{\Theta}'} \sum_{i=1}^{n} \log q(\boldsymbol{x}_i; \boldsymbol{\Theta}) + \sum_{i'=1}^{n'} \log q(\boldsymbol{x}'_{i'}; \boldsymbol{\Theta}') - \gamma \|\boldsymbol{\Theta} - \boldsymbol{\Theta}'\|_1$$

$$\gamma \geq 0$$

☺ Scalable to high-dimensional datasets.

☺ Work well even if true $\boldsymbol{\Theta}$ and $\boldsymbol{\Theta}'$ are dense.

# Contents

# Correlation and Dependence

$$q(\boldsymbol{x};\boldsymbol{\Theta}) = \frac{\det(\boldsymbol{\Theta})^{1/2}}{(2\pi)^{d/2}} \exp\left(-\frac{1}{2}\boldsymbol{x}^\top \boldsymbol{\Theta}\boldsymbol{x}\right)$$

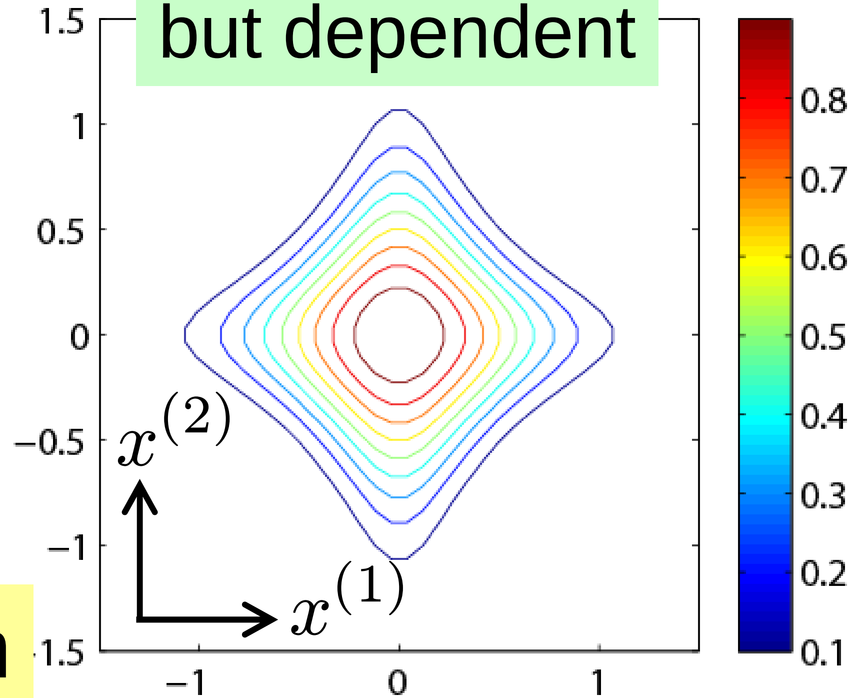
$\boldsymbol{\Theta}$: (sparse) inverse covariance matrix

- Gaussian models cannot capture higher-order correlations.
- No correlation does not imply independence.

Independence

No correlation

No correlation but dependent



# Nonparanormal Models



Han Liu *et al.* (JMLR2009)

- Gaussian after element-wise transformation:

$$q(\boldsymbol{x};\boldsymbol{\Theta}) = \frac{\det(\boldsymbol{\Theta})^{1/2}}{(2\pi)^{d/2}} \exp\left(-\frac{1}{2}\boldsymbol{f}(\boldsymbol{x})^\top \boldsymbol{\Theta} \boldsymbol{f}(\boldsymbol{x})\right) \prod_{k=1}^{d} |f_k'(x^{(k)})|$$

$$\boldsymbol{f}(\boldsymbol{x}) = (f_1(x^{(1)}), \ldots, f_d(x^{(d)}))^\top$$

$$\boldsymbol{x} = (x^{(1)}, \ldots, x^{(d)})^\top$$

$f_k$ : Monotone and differentiable function

☺ More flexible than ordinary Gaussian models.

☹ Still not flexible enough.

# Pairwise Markov Networks



$$q(\boldsymbol{x};\boldsymbol{\theta}) = \frac{\overline{q}(\boldsymbol{x};\boldsymbol{\theta})}{Z(\boldsymbol{\theta})} \qquad \overline{q}(\boldsymbol{x};\boldsymbol{\theta}) = \exp\left(\sum_{k\geq k'} \boldsymbol{\theta}_{k,k'}^\top \boldsymbol{f}(x^{(k)}, x^{(k')})\right)$$

$\boldsymbol{f}(x,x')$: feature vector

$$\boldsymbol{x} = (x^{(1)}, \ldots, x^{(d)})^\top$$

$$\boldsymbol{\theta} = (\boldsymbol{\theta}_{1,1}^\top, \ldots, \boldsymbol{\theta}_{d,d}^\top)^\top$$

- Gaussian: $\boldsymbol{f}(x,x') = xx'$
- Nonparanormal: $\boldsymbol{f}(x,x') = f(x)f(x')$
- Polynomial: $\boldsymbol{f}(x,x') = [x^t, x^{t-1}x', \ldots, x, x', 1]^\top$

☺ Highly flexible.

☹ Normalization $Z(\boldsymbol{\theta}) = \int \overline{q}(\boldsymbol{x};\boldsymbol{\theta})\mathrm{d}\boldsymbol{x}$ is intractable.

# Contents

# Avoiding Density Estimation

- Fused lasso for non-paranormal models: $\gamma \geq 0$

$$\max_{\boldsymbol{\Theta},\boldsymbol{\Theta}'} \sum_{i=1}^{n} \log q(\boldsymbol{x}_i; \boldsymbol{\Theta}) + \sum_{i'=1}^{n'} \log q(\boldsymbol{x}'_{i'}; \boldsymbol{\Theta}') - \gamma \|\boldsymbol{\Theta} - \boldsymbol{\Theta}'\|_1$$

☺ Work well even if true $\boldsymbol{\Theta}$ and $\boldsymbol{\Theta}'$ are dense.

☺ Higher correlations can be partially captured.

☹ Handling non-Gaussian model is not easy.

☹ Still need explicit modeling of $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$.

- Vapnik's principle:

***Don't solve a more general problem!***

# Contents



1. Distributional change detection
2. Structural change detection
   - A) Density estimation approach
   - B) Density-ratio estimation approach
     - I. Algorithm and properties
     - II. Experiments

# Direct Change Modeling in Markov Networks



Liu *et al.* (ECML2013, NeCo2014)

- Without separately modeling $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$, let's directly model the density ratio $p(\boldsymbol{x})/p'(\boldsymbol{x})$ :

$$r(\boldsymbol{x}) = \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})} \approx \frac{q(\boldsymbol{x};\boldsymbol{\theta})}{q(\boldsymbol{x};\boldsymbol{\theta}')} \propto \exp\left(\sum_{k \geq k'} (\boldsymbol{\theta}_{k,k'} - \boldsymbol{\theta}'_{k,k'})^\top \boldsymbol{f}(x^{(k)}, x^{(k')})\right)$$

$$q(\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left(\sum_{k \geq k'} \boldsymbol{\theta}_{k,k'}^\top \boldsymbol{f}(x^{(k)}, x^{(k')})\right)$$

- Individual parameters $\boldsymbol{\theta}, \boldsymbol{\theta}'$ are not necessary, but their difference $\boldsymbol{\alpha} = \boldsymbol{\theta} - \boldsymbol{\theta}'$ is sufficient.

# Ratio of Markov Network Models



$$r_{\boldsymbol{\alpha}}(\boldsymbol{x}) = \frac{1}{N(\boldsymbol{\alpha})} \exp\left( \sum_{k \geq k'} \boldsymbol{\alpha}_{k,k'}^\top \boldsymbol{f}(x^{(k)}, x^{(k')}) \right)$$

$$\boldsymbol{\alpha} = (\boldsymbol{\alpha}_{1,1}^\top, \ldots, \boldsymbol{\alpha}_{d,d}^\top)^\top$$

- Normalization:

$$r(\boldsymbol{x}) = \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})} \Longrightarrow \int p'(\boldsymbol{x}) r(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \int p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = 1$$

☺ Naïve sample averaging is consistent:

$$N(\boldsymbol{\alpha}) = \int \underline{p'(\boldsymbol{x})} \exp\left( \sum_{k \geq k'} \boldsymbol{\alpha}_{k,k'}^\top \boldsymbol{f}(x^{(k)}, x^{(k')}) \right) \mathrm{d}\boldsymbol{x}$$

$$\approx \frac{1}{n'} \sum_{i'=1}^{n'} \exp\left( \sum_{k \geq k'} \boldsymbol{\alpha}_{k,k'}^\top \boldsymbol{f}({x'}_{i'}^{(k)}, {x'}_{i'}^{(k')}) \right)$$

# KL Density-Ratio Estimation



Nguyen *et al.* (NIPS2007, IEEE-IT2010)
Sugiyama *et al.* (NIPS2007, AISM2008)

- Density-ratio matching under KL-divergence:

$$\min_{\boldsymbol{\alpha}} \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x}) r_{\boldsymbol{\alpha}}(\boldsymbol{x})} \mathrm{d}\boldsymbol{x}$$

$$r_{\boldsymbol{\alpha}}(\boldsymbol{x}) \approx \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})}$$

- Naïve sample approximation gives

$$\min_{\boldsymbol{\alpha}} \log \frac{1}{n'} \sum_{i'=1}^{n'} \exp\left( \sum_{k \geq k'} \boldsymbol{\alpha}_{k,k'}^{\top} \boldsymbol{f}(x_{i'}^{\prime(k)}, x_{i'}^{\prime(k')}) \right) - \frac{1}{n} \sum_{i=1}^{n} \sum_{k \geq k'} \boldsymbol{\alpha}_{k,k'}^{\top} \boldsymbol{f}(x_i^{(k)}, x_i^{(k')})$$

  - Tractable for any feature $\boldsymbol{f}(x^{(k)}, x^{(k')})$.

- Add a smoothing regularizer: $+\eta \|\boldsymbol{\alpha}\|^2$
- Add a group-sparsity regularizer: $+\gamma \sum_{k \geq k'} \|\boldsymbol{\alpha}_{k,k'}\|$

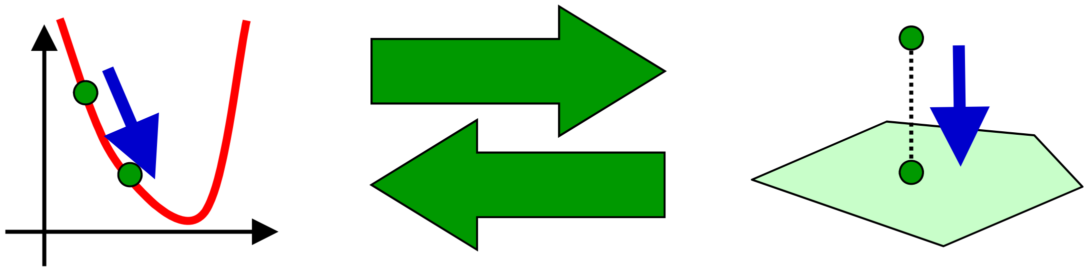
# Primal Optimization

$$\min_{\boldsymbol{\alpha}} \log \frac{1}{n'} \sum_{i'=1}^{n'} \exp\left( \sum_{k \geq k'} \boldsymbol{\alpha}_{k,k'}^{\top} \boldsymbol{f}(x_{i'}'^{(k)}, x_{i'}'^{(k')}) \right)$$

$$-\frac{1}{n} \sum_{i=1}^{n} \sum_{k \geq k'} \boldsymbol{\alpha}_{k,k'}^{\top} \boldsymbol{f}(x_{i}^{(k)}, x_{i}^{(k')}) + \eta \|\boldsymbol{\alpha}\|^2$$

$$\text{subject to} \quad \sum_{k \geq k'} \|\boldsymbol{\alpha}_{k,k'}\| \leq C_{\gamma}$$

- Simple gradient-projection gives the global solution.
- Efficient when more samples than parameters.



# Dual Optimization

$$\min_{\boldsymbol{\beta}} \sum_{i'=1}^{n'} \beta_{i'}^{\top} \log \beta_{i'} + \frac{1}{2\eta} \sum_{k \geq k'} \max(0, \|\boldsymbol{m}_{k,k'}\| - \gamma)^2$$

$$\text{subject to } \beta_1, \ldots, \beta_{n'} \geq 0, \sum_{i'=1}^{n'} \beta_{i'} = 1$$

$$\boldsymbol{m}_{k,k'} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{f}(\boldsymbol{x}_i^{(k)}, \boldsymbol{x}_i^{(k')}) - \frac{1}{n'} \sum_{i'=1}^{n'} \beta_{i'} \boldsymbol{f}(\boldsymbol{x}'^{(k)}_{i'}, \boldsymbol{x}'^{(k')}_{i})$$

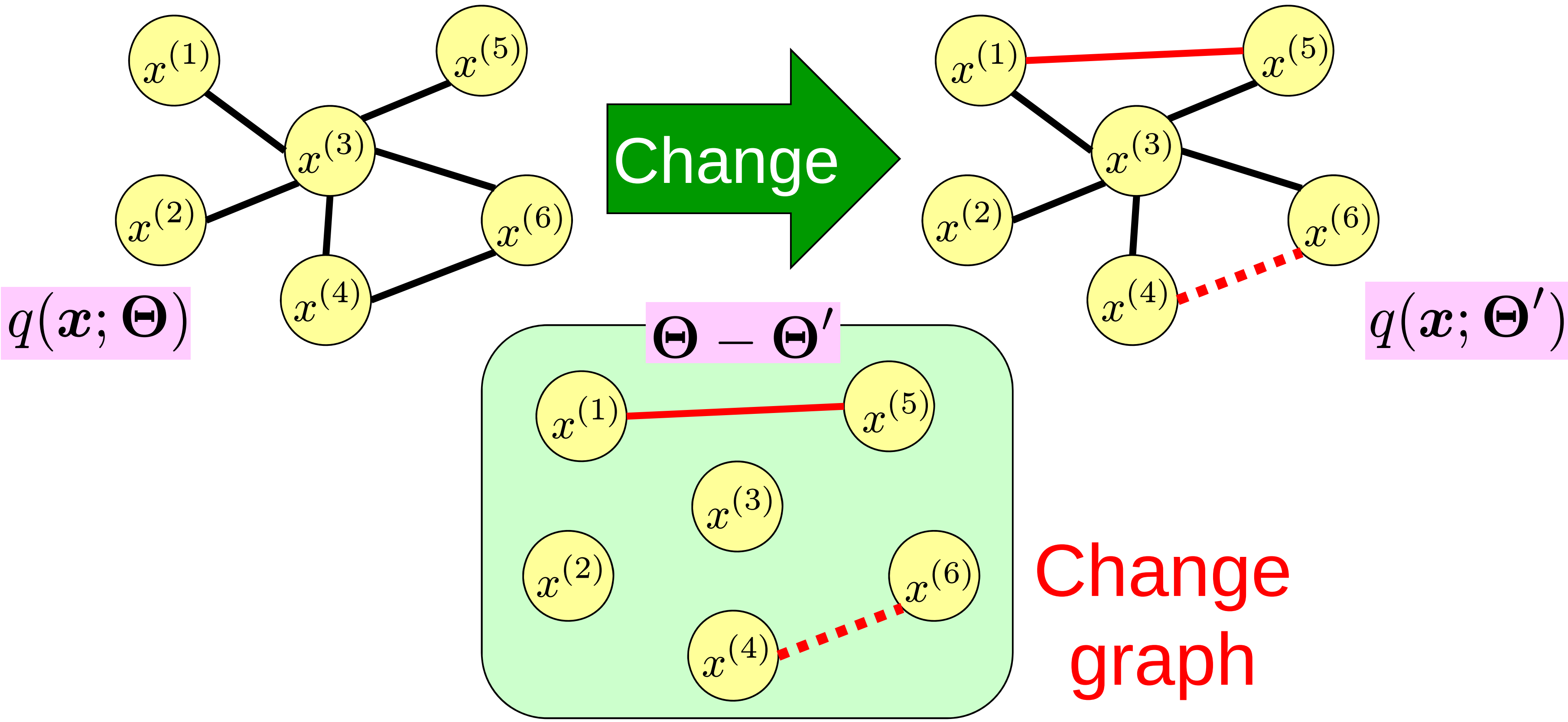
$$\Rightarrow \quad \boldsymbol{\alpha}_{k,k'} = \max\left(0, \|\boldsymbol{m}_{k,k'}\| - \gamma\right) \frac{\boldsymbol{m}_{k,k'}}{\eta \|\boldsymbol{m}_{k,k'}\|}$$

- Simple gradient-projection gives the global solution.
- Efficient when more parameters than samples.

# Theoretical Properties

Liu *et al.* (AAAI2015)

- Change detection is easy as long as the change graph is sparse.
  - Each graph does not have to be sparse.

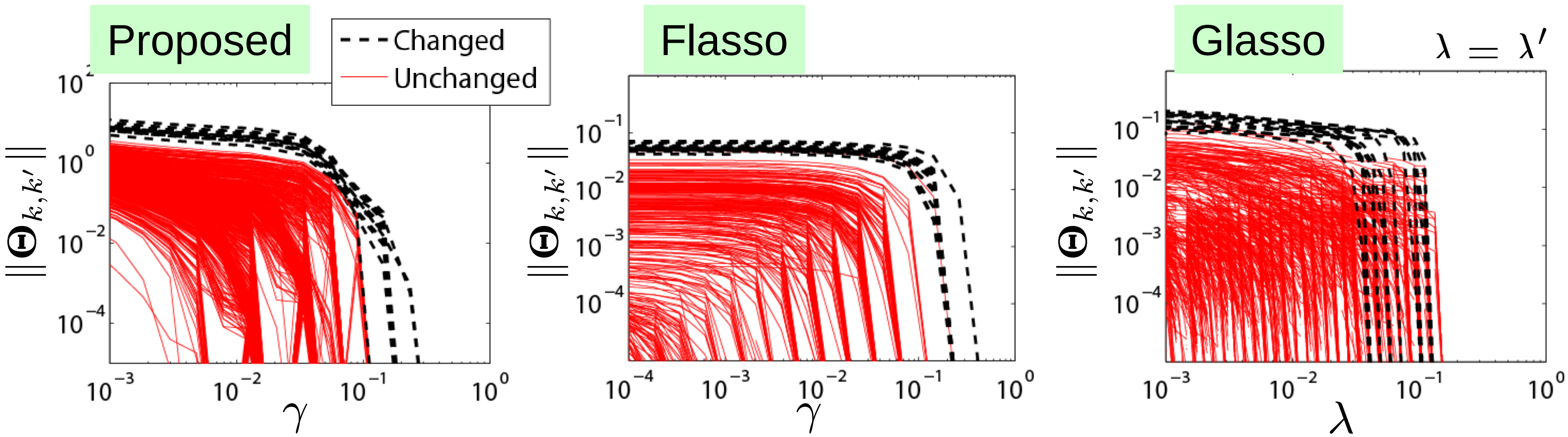


# Contents

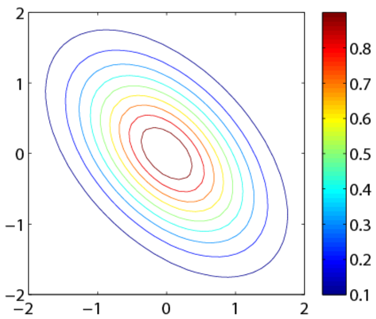


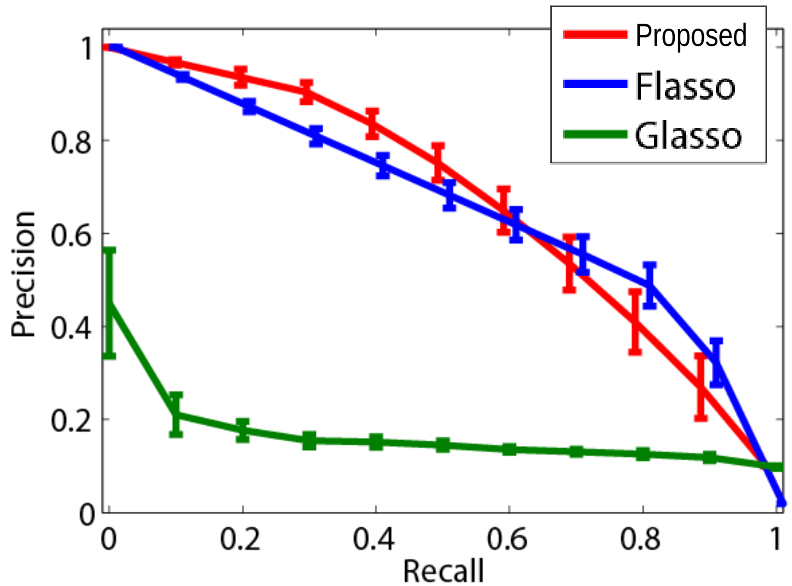
1. Distributional change detection
2. Structural change detection
   A) Density estimation approach
   B) Density-ratio estimation approach
      I. Algorithm and properties
      II. Experiments

# Gaussian Data
# (d=40, n=n'=100, Change in 15 Edges)

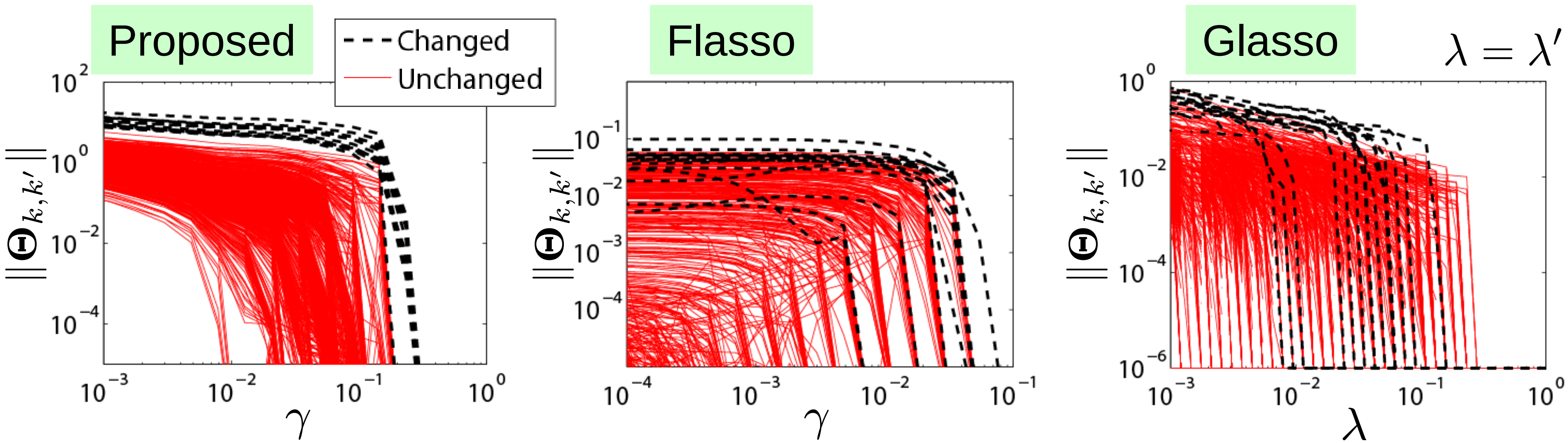



- All use the Gaussian model
- Proposed method and Flasso work well.



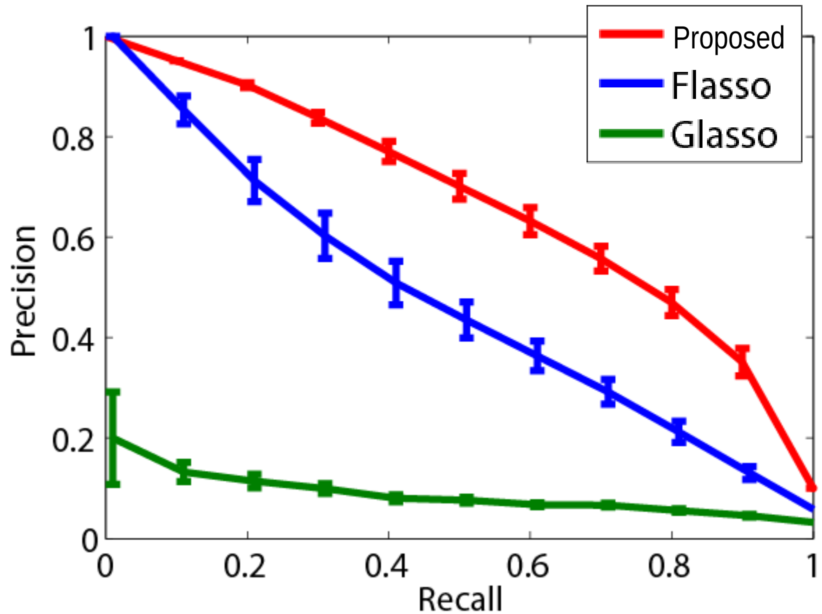


# Gaussian Data (d=40, n=n'=50, Change in 15 Edges)

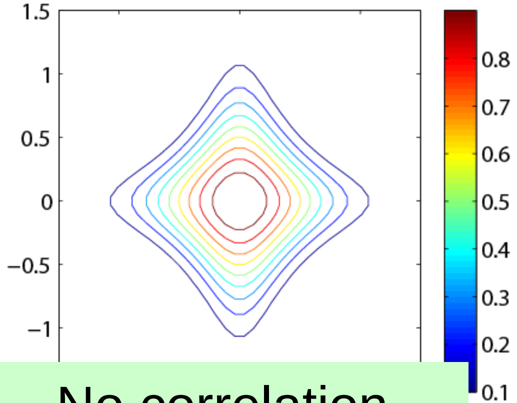



- Proposed method works well with small samples.

$$\alpha = \theta - \theta'$$



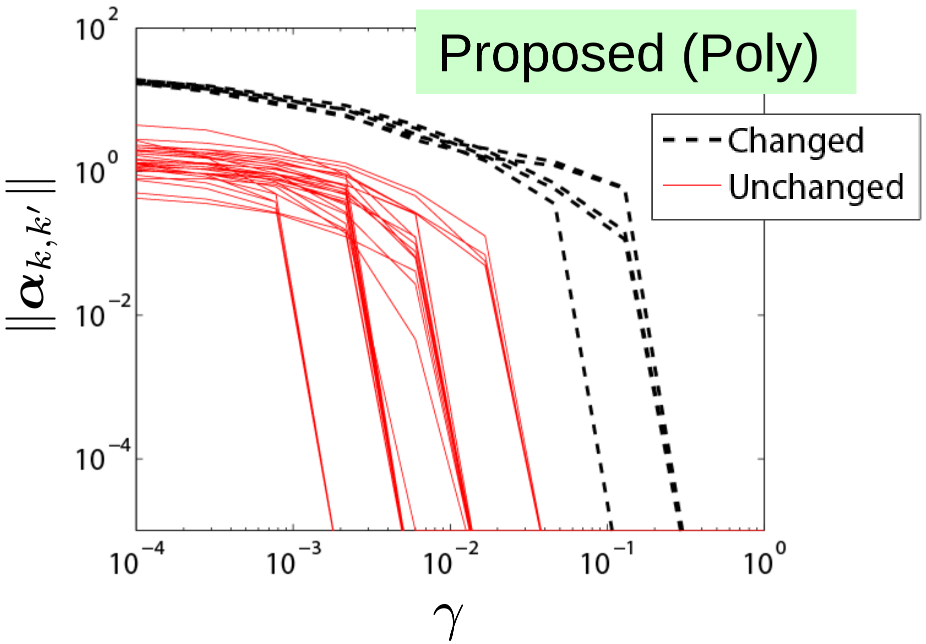
# Non-Gaussian Data (d=9, n=n'=5000, Change in 7 Edges)

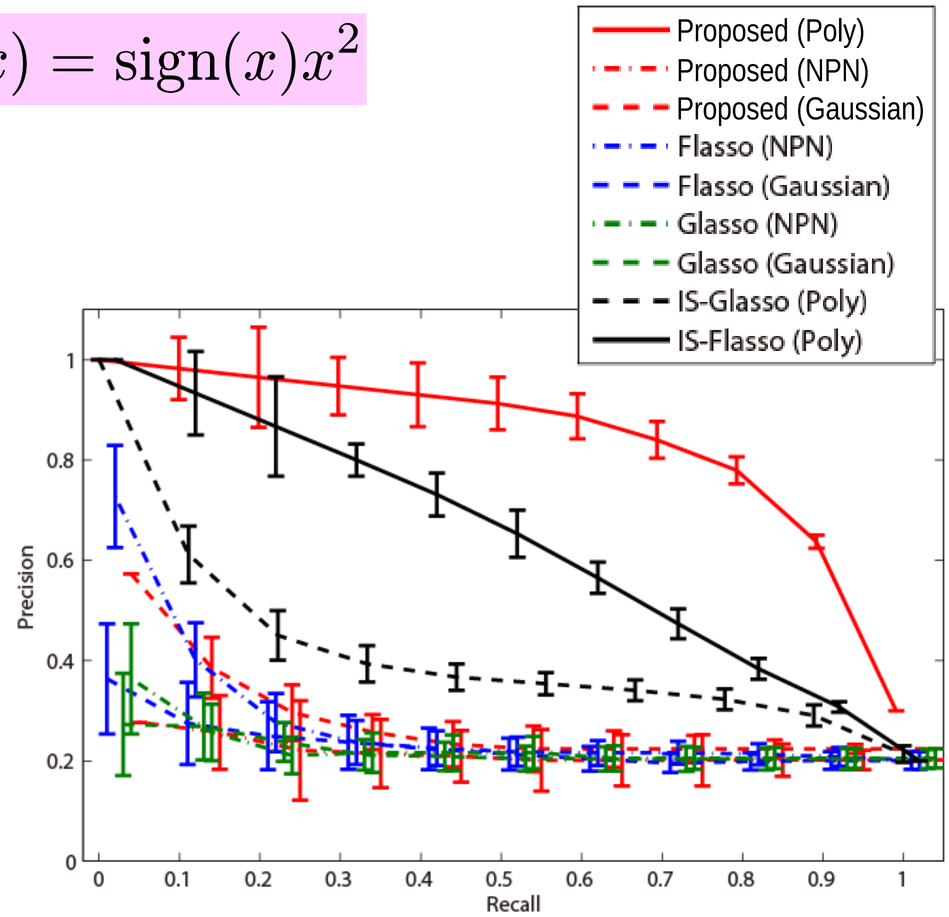



No correlation, no nonparanormal

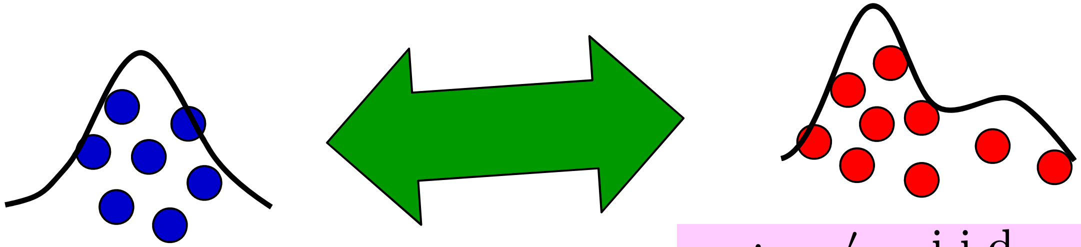
- Proposed method (Poly) works well.
  - Poly: $\boldsymbol{f}(x, x') = [x^t, x^{t-1}x', \dots, x, x', 1]^\top$
  - NPN: $f(x) = \mathrm{sign}(x)x^2$

Proposed (Poly)



# Take-Home Messages



$\{\boldsymbol{x}_i\}_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x})$ $\{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'} \overset{\text{i.i.d.}}{\sim} p'(\boldsymbol{x})$

***Don't solve a more general problem!***

- Learn the change directly:
  - Robust distributional change detection by direct density-difference estimation
  - Interpretable structural change detection by group-sparse direct density-ratio estimation
- Software: [http://www.ms.k.u-tokyo.ac.jp/](http://www.ms.k.u-tokyo.ac.jp/)