Estimating Density Ratio: Learning Changes of Patterns

Change Detection, Graphical Models, and Transfer Learning

Song Liu(liu@ism.ac.jp)

The Institute of Statistical Mathematics, Japan

February 15, 2016

Overview

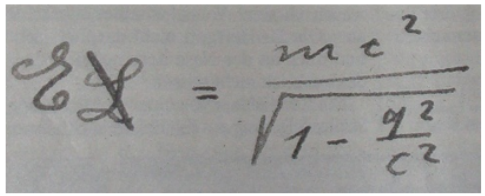
Song Liu

- BEng, Soochow University, China.
- MSc, University of Bristol, UK.
- DEng, Tokyo Institute of Technology, Japan
- Posdoc, Tokyo Institute of Technology and The Institute of Statistical Mathematics, Japan

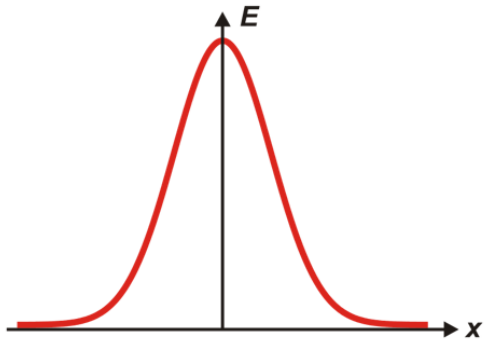
The Institute of Statistical Mathematics (ISM), Japan

- Japan's statistical research organization.
- Hirotugu Akaike was a former researcher and director at ISM.

Data is Big and Blurry



Data is so big, so we look for compressive patterns.



Data involving uncertainty, we prefer statistical patterns .

Data Changes

- Smart technologies provide us ways of **updating information**.
-  People use mobiles to send tweets, and trend topics.
- Challenging our traditional view of statistical learning.
- Would you learn a pattern **today** knowing it is going to change **tomorrow**?
  - Particularly, when learning a pattern is expensive (Deep Net?)!
- Dataset shift problem [Quionero-Candela et al., 2009].

Changes between Patterns

- Knowing the change itself can be helpful (Part I and II).
  - Change Detection, Outlier Detection, etc.
- The changes of patterns are also **relative patterns**.
- Use it to make adjustment on our old pattern (Part III).

Density Ratio, Measuring the Changes of Patterns

- Given a set of samples, $\mathcal{D}_p := \{\boldsymbol{x}_p^{(i)}\}_{i=1}^{n_p} \sim P$
- Density $p(\boldsymbol{x})$ describes the static pattern of $\mathcal{D}_p$
- Given another set of samples, $\mathcal{D}_q := \{\boldsymbol{x}_q^{(i)}\}_{i=1}^{n_q} \sim Q$
- Density ratio $\frac{p(\boldsymbol{x})}{q(\boldsymbol{x})}$ describes the **changes** between datasets.
  - Ratio is **directional**!

Models of Density Ratio

A density model

$$p(\boldsymbol{x};\boldsymbol{\theta}_p) = \frac{1}{Z(\boldsymbol{\theta}_p)} \exp(\boldsymbol{\theta}_p^\top \boldsymbol{f}(\boldsymbol{x}))$$

Taking the ratio:

$$\frac{p(\boldsymbol{x};\boldsymbol{\theta}_p)}{q(\boldsymbol{x};\boldsymbol{\theta}_q)} \propto \frac{\exp(\boldsymbol{\theta}_p^\top \boldsymbol{f}(\boldsymbol{x}))}{\exp(\boldsymbol{\theta}_q^\top \boldsymbol{f}(\boldsymbol{x}))} = \exp((\boldsymbol{\theta}_p - \boldsymbol{\theta}_q)^\top \boldsymbol{f}(\boldsymbol{x}))$$

Letting $\boldsymbol{\theta} = \boldsymbol{\theta}_p - \boldsymbol{\theta}_q$

$$g(\boldsymbol{x};\boldsymbol{\theta}) := \frac{p(\boldsymbol{x};\boldsymbol{\theta}_p)}{q(\boldsymbol{x};\boldsymbol{\theta}_q)} = \frac{1}{N(\boldsymbol{\theta})} \exp(\boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x}))$$

Models of Density Ratio

- Density ratio needs to be normalized.
- $\int q(\boldsymbol{x}) g(\boldsymbol{x}; \boldsymbol{\theta}) \mathrm{d}\boldsymbol{x} = 1$.
- Let $N(\boldsymbol{\theta}) = \int q(\boldsymbol{x}) \exp(\boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x})) \mathrm{d}\boldsymbol{x}$ suffices.
  - can be approximated using samples:

$$\hat{N}(\boldsymbol{\theta}) := \frac{1}{n_q} \sum_{j=1}^{n_q} \exp\left(\boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x}^{(j)})\right)$$

- We denote $\hat{g}(\boldsymbol{x}; \boldsymbol{\theta}) := \frac{\exp\left(\boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x})\right)}{\hat{N}(\boldsymbol{\theta})}$.

Models of Density Ratio

- A few variations are available:
- use linear model instead of log-linear model:

$$g(\boldsymbol{x};\boldsymbol{\theta}) := \frac{1}{N(\boldsymbol{\theta})}\boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x}), N(\boldsymbol{\theta}) := \int q(\boldsymbol{x})\boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.$$

- or drop out the normalization term completely

$$g(\boldsymbol{x};\boldsymbol{\theta}) := \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x}),$$

and use other methods to enforce the normalization.

Estimating the Density Ratio

- Need to **measure the difference** between the true quantity $\frac{p}{q}$ and the estimated model g_θ.
- No natural distances apply here.
- There are difference measures for distributions though, such as Kullback-Leibler (KL) divergence [Kullback and Leibler, 1951]:

$$\mathrm{KL}\left[p\|q\right] = \int p(x) \log \frac{p(x)}{q(x)} dx$$

Estimating the Density Ratio

- **Idea**: we can **reconstruct** a "density model" from density ratio model:

$$p_{\theta}(\boldsymbol{x}) = q(\boldsymbol{x})g(\boldsymbol{x};\boldsymbol{\theta})$$

  and minimize the difference between $p(\boldsymbol{x})$ and $p_{\theta}(\boldsymbol{x})$.
  - Won't be able to compute this "density" model for a specific $\boldsymbol{x}$.
  - Not interested in modelling the individual densities anyway.

Kullback-Leibler Importance Estimation Procedure (KLIEP)

Criterion [Sugiyama et al., 2008a]

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \mathrm{KL}\left[p \| p_{\boldsymbol{\theta}}\right],$$

where

$$\mathrm{KL}\left[p \| p_{\boldsymbol{\theta}}\right] = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{p_{\boldsymbol{\theta}}(\boldsymbol{x})} \mathrm{d}\boldsymbol{x} = -\int p(\boldsymbol{x}) \log g(\boldsymbol{x}; \boldsymbol{\theta}) + C,$$

where C is some constant.

We can approximate the above criterion using sample average:

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmax}_{\boldsymbol{\theta}} \frac{1}{n_p} \sum_{i=1}^{n_p} \log g(\boldsymbol{x}^{(i)}; \boldsymbol{\theta}).$$

Kullback-Leibler Importance Estimation Procedure (KLIEP)

Plug in $g(\boldsymbol{x}; \boldsymbol{\theta})$:

$$\begin{aligned}
\hat{\boldsymbol{\theta}} &= \operatorname*{argmax}_{\boldsymbol{\theta}} \frac{1}{n_p} \sum_{i=1}^{n_p} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x}^{(i)}) - \log N(\boldsymbol{\theta}) \\
&\approx \operatorname*{argmax}_{\boldsymbol{\theta}} \frac{1}{n_p} \sum_{i=1}^{n_p} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x}^{(i)}) - \log \hat{N}(\boldsymbol{\theta}) \\
&\approx \operatorname*{argmax}_{\boldsymbol{\theta}} \underbrace{\frac{1}{n_p} \sum_{i=1}^{n_p} \boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x}^{(i)}) - \log \frac{1}{n_q} \sum_{j=1}^{n_q} \exp(\boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x}^{(j)}))}_{\ell_{\mathrm{KLIEP}}(\boldsymbol{\theta})}
\end{aligned}$$

Concave, unconstrained, objective.

A Few Simplification...

Let's denote

$$\mathbb{E}_p\left[f(x)\right] := \int p(x) f(x) \mathrm{d}x$$

as the population expectation and

$$\hat{\mathbb{E}}_p\left[f(x)\right] := \frac{1}{n} \sum_{i=1}^{n} f(x^{(i)})$$

as empirical expectation of $f(x)$ given samples $\{\boldsymbol{x}^i\}_{i=1}^n \sim P$.

Kullback-Leibler Importance Estimation Procedure (KLIEP)

KLIEP (again)

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmax}_{\boldsymbol{\theta}} \ell_{\mathrm{KLIEP}}(\boldsymbol{\theta}) = \operatorname*{argmax}_{\boldsymbol{\theta}} \hat{\mathbb{E}}_p[\boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x})] - \log \hat{\mathbb{E}}_q \left[\exp\left(\boldsymbol{\theta}^\top \boldsymbol{f}(\boldsymbol{x})\right)\right],$$

and

$$\nabla_{\boldsymbol{\theta}} \ell_{\mathrm{KLIEP}}(\boldsymbol{\theta}) = \hat{\mathbb{E}}_p\left[\boldsymbol{f}(\boldsymbol{x})\right] - \hat{\mathbb{E}}_q\left[\hat{g}(\boldsymbol{x};\boldsymbol{\theta})\boldsymbol{f}(\boldsymbol{x})\right].$$

Variations of Density Ratio Estimators

- Can we measure the difference between true density ratio and model density?
- How about least square (ℓ^2) distance?

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \int \| \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} - g(\boldsymbol{x}; \boldsymbol{\theta}) \|^2 \mathrm{d}\boldsymbol{x}$$

Won't do, no way to compute that integral.

- However, with a little change (called uLSIF [Kanamori et al., 2009])...

$$\begin{aligned}\hat{\boldsymbol{\theta}} &= \operatorname*{argmin}_{\boldsymbol{\theta}} \int q(\boldsymbol{x}) \| \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} - g(\boldsymbol{x}; \boldsymbol{\theta}) \|^2 \mathrm{d}\boldsymbol{x} \\ &= \operatorname*{argmin}_{\boldsymbol{\theta}} \int q(\boldsymbol{x}) g(\boldsymbol{x}; \boldsymbol{\theta})^2 - 2p(\boldsymbol{x}) g(\boldsymbol{x}; \boldsymbol{\theta}) \mathrm{d}\boldsymbol{x} + C.\end{aligned}$$

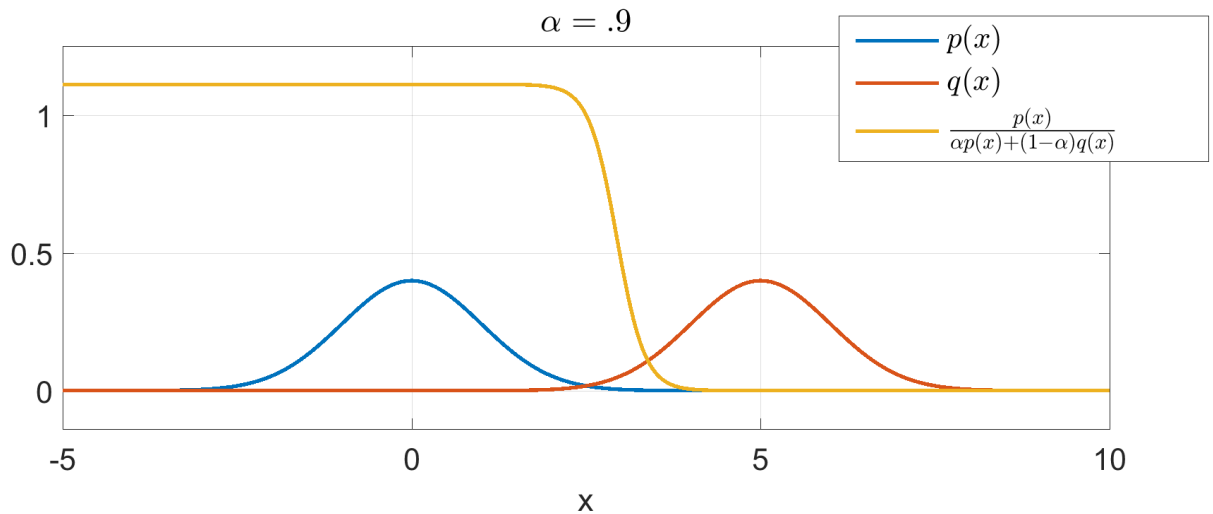
Then, sample average...

Variations of Density Ratio Estimators

- $\frac{p(x)}{q(x)}$ can go to infinity!
  - which is a bad news for the estimation.
  - a few outliers may trick the estimator to think two distributions are dramatically different.
- Bound the density ratio function! [Yamada et al., 2013]

$$\frac{p(x)}{\alpha p(x) + (1-\alpha)q(x)} < \frac{1}{\alpha}, \alpha \in (0,1).$$

Estimate the ratio between p and an α-mixture of p and q (called RuLSIF).

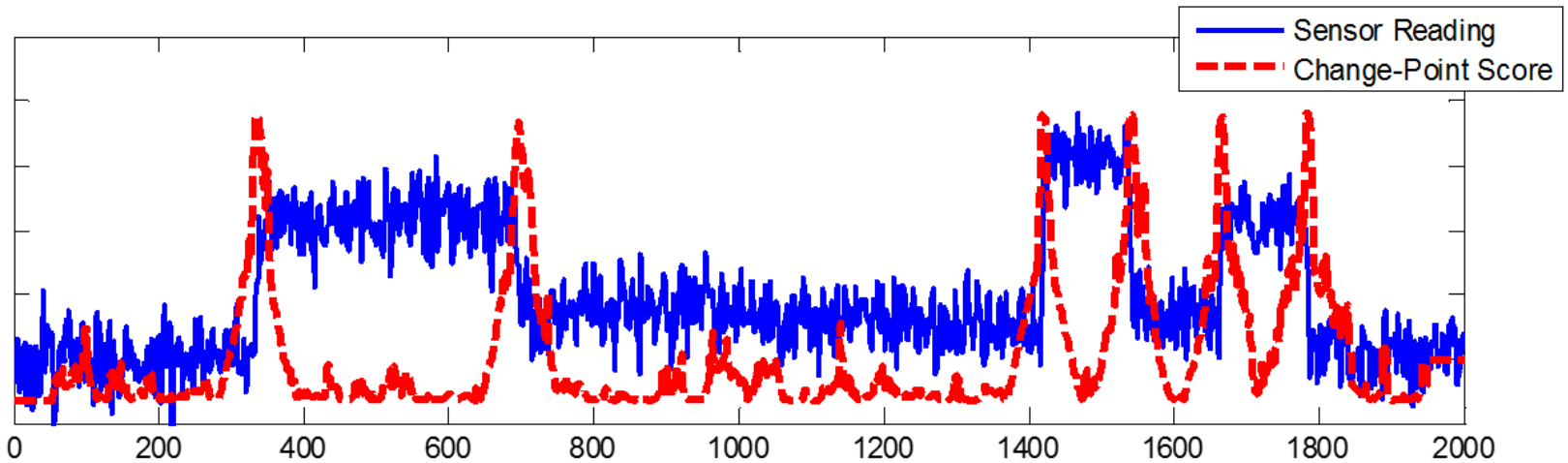


Which One to Use?

- There is no definitive answer, and it's all up to the application.
- Computational efficiency:
  - KLIEP solves a non-linear optimization, and uLSIF and RuLSIF has analytical solutions.
  - Computing KLIEP gradient requires $\mathcal{O}(max(n_p, n_q)m)$.
  - Computing uLSIF solution requires $\mathcal{O}(max(n_q, m)m^2)$.
  - m is the number of dimensions of the parameter vector.
- Outlier affect KLIEP more than least square based methods.
  - "log" term is a bit troublesome.
  - "$\log g \to -\infty$" if $g \to 0$.

Change-point Detection [Liu et al., 2013]

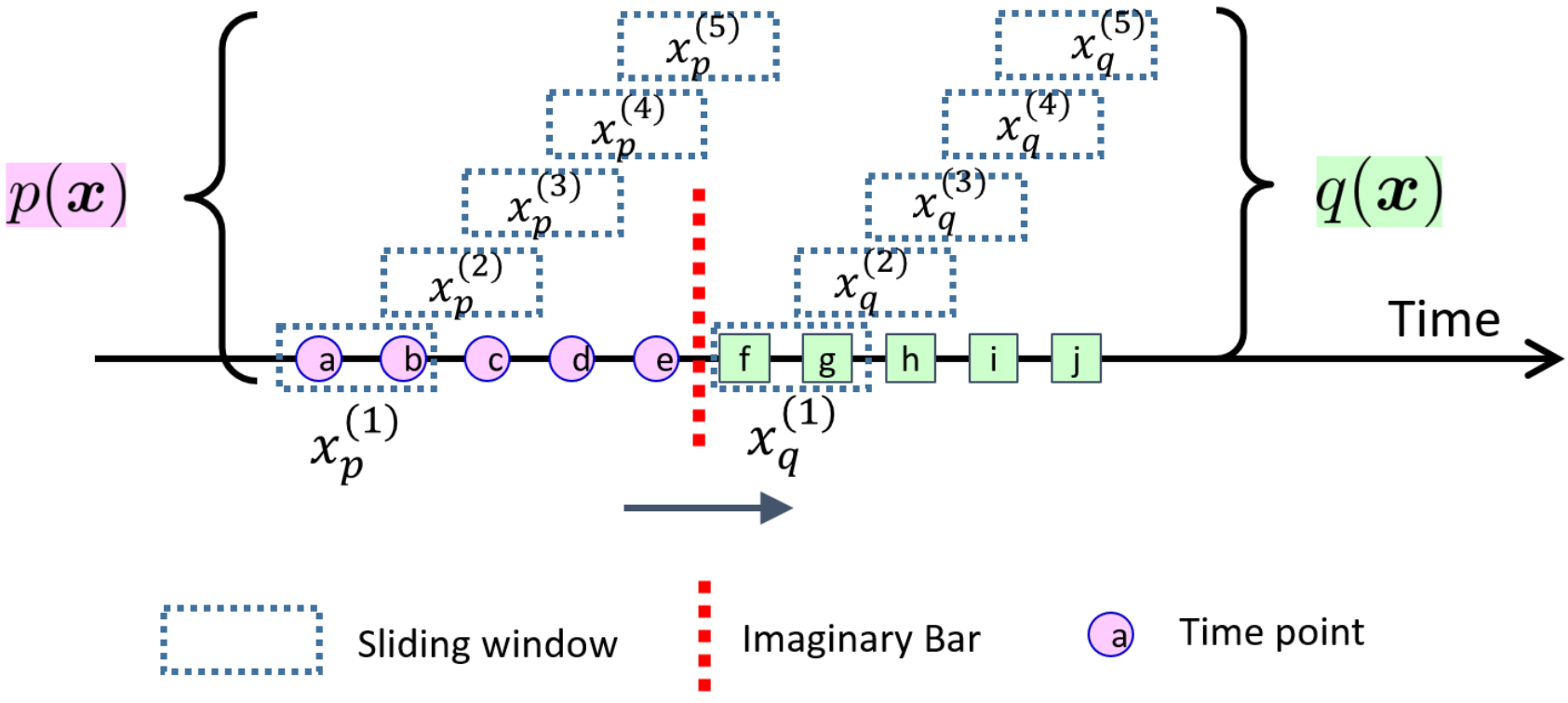
Well-log Data



- **Objective**: Detecting abrupt changes lying among time-series data
- **Change-point score**: Plausibility of changes that have happened

Problem Formulation
[Kawahara et al., 2007, Liu et al., 2013]

- Construct samples by using sliding window.
- Set an imaginary bar in the middle divides samples into two groups.
- Test divergence between two groups of samples.



From Ratio to Divergence

- How do we convert ratio to divergence?

$$D(p\|q) = \int q(x) f\left(\frac{p(x)}{q(x)}\right) \mathrm{d}x$$

  where f is a convex function, and $f(1) = 0$.
  - $f(t) = t \log t$, we get KL divergence.
  - $f(t) = (t-1)^2$, we get Pearson divergence.
- Divergence is not symmetric, we symmetrize it by

$$D(p\|q) + D(q\|p)$$

.

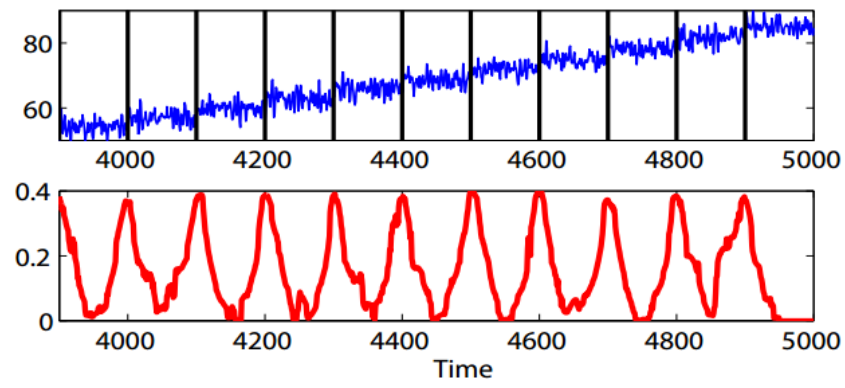
From Ratio to Divergence

- $\mathrm{KL}[p\|q] \approx \frac{1}{n_p}\sum_{i=1}^{n_p} \hat{g}(\boldsymbol{x}^{(i)}, \hat{\boldsymbol{\theta}})$
  - where $\hat{g}$ is estimated from KLIEP.
- $\mathrm{PE}[p\|q] \approx -\frac{1}{2n_q}\sum_{i=1}^{n_q} \hat{g}^2(\boldsymbol{x}^{(i)}, \hat{\boldsymbol{\theta}}) + \frac{1}{n_q}\sum_{j=1}^{n_p} \hat{g}(\boldsymbol{x}^{(j)}, \hat{\boldsymbol{\theta}}) - \frac{1}{2}$
  - where $\hat{g}$ is estimated from uLSIF.
- $$\begin{aligned}&\mathrm{rPE}[p\|q]\\ &\approx -\frac{\alpha}{2n_p}\sum_{i=1}^{n_p} \hat{g}^2(\boldsymbol{x}^{(i)}, \hat{\boldsymbol{\theta}}) - \frac{1-\alpha}{2n_q}\sum_{i=1}^{n_q} \hat{g}^2(\boldsymbol{x}^{(i)}, \hat{\boldsymbol{\theta}})\\ &\quad + \frac{1}{n_q}\sum_{j=1}^{n_p} \hat{g}(\boldsymbol{x}^{(j)}, \hat{\boldsymbol{\theta}}) - \frac{1}{2}\end{aligned}$$
  - where $\hat{g}$ is estimated from RuLSIF.
- You may mix them up, but they won't be optimal.

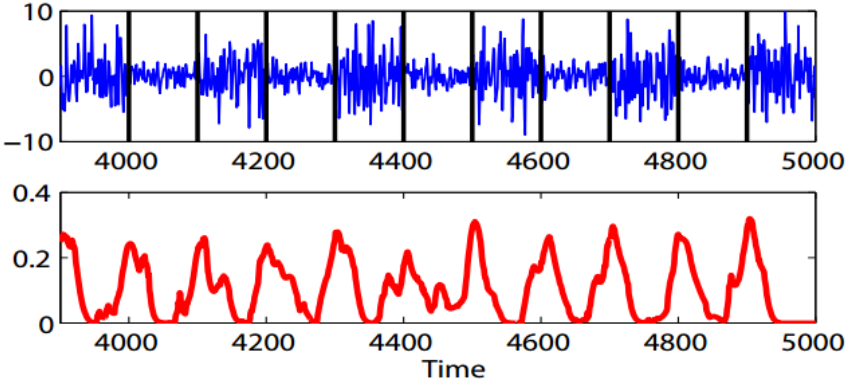
Fun facts [Nguyen et al., 2010]

- Estimating density ratio using KLIEP is actually maximizing the lower-bound of Kullback-leibler divergence.
- Estimating density ratio using uLSIF is actually maximizing the lower-bound of Pearson divergence.
- Estimating density ratio using RuLSIF is actually maximizing the lower-bound of *Relative* Pearson divergence.
- Fenchel Duality

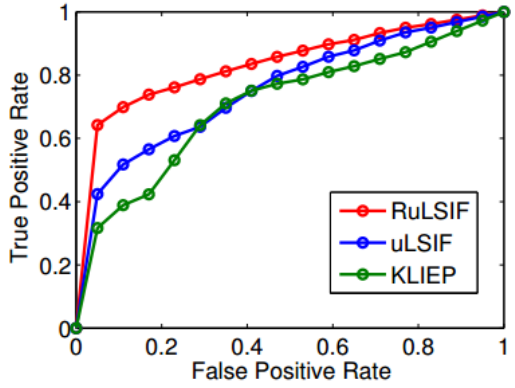
Toy Dataset



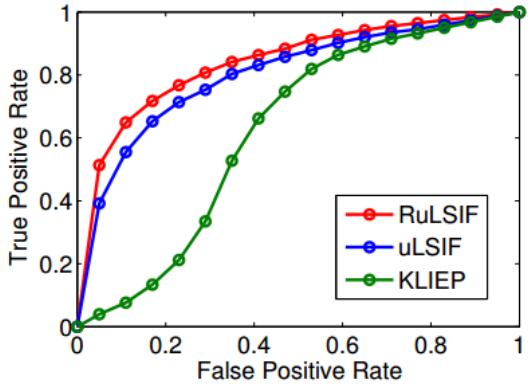
(a) Dataset1



(b) Dataset2

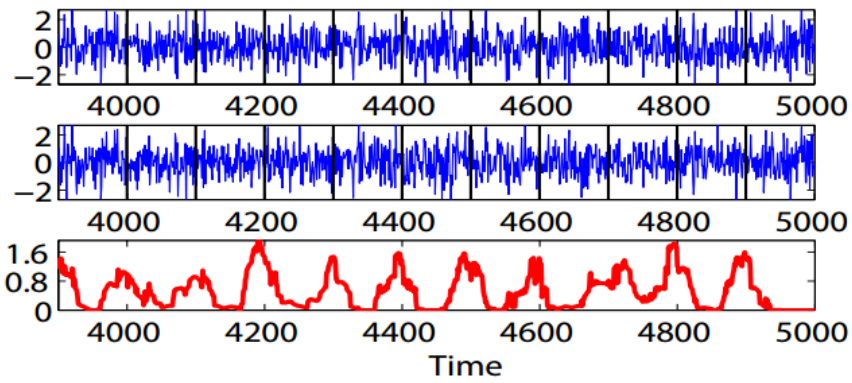


(a) Dataset1

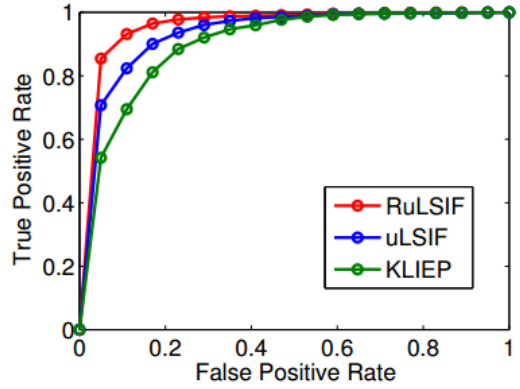


(b) Dataset2

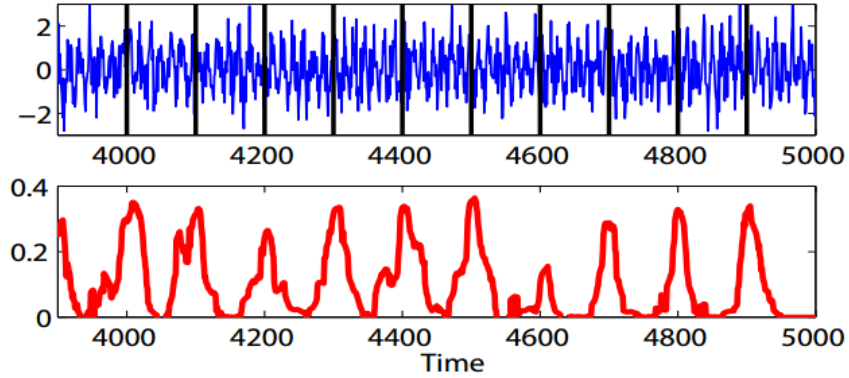
Toy Dataset



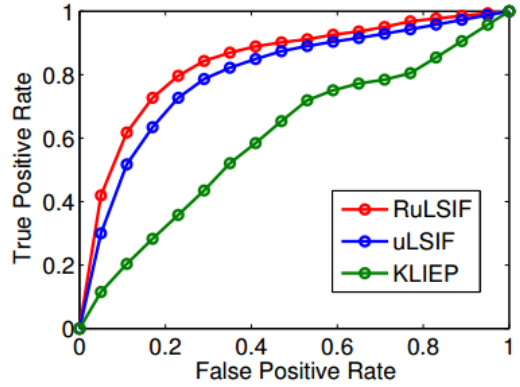
(c) Dataset3



(c) Dataset3

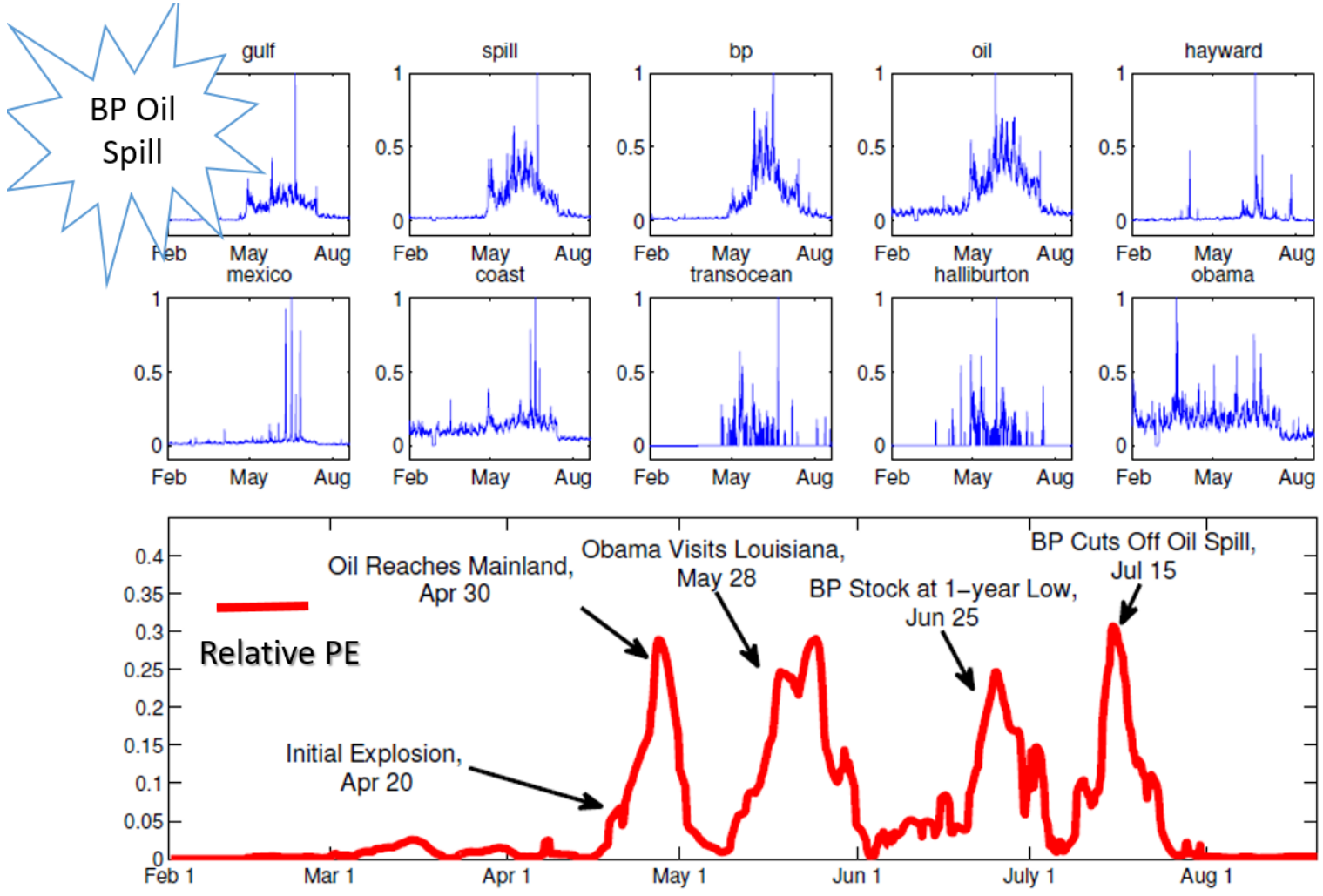


(d) Dataset4

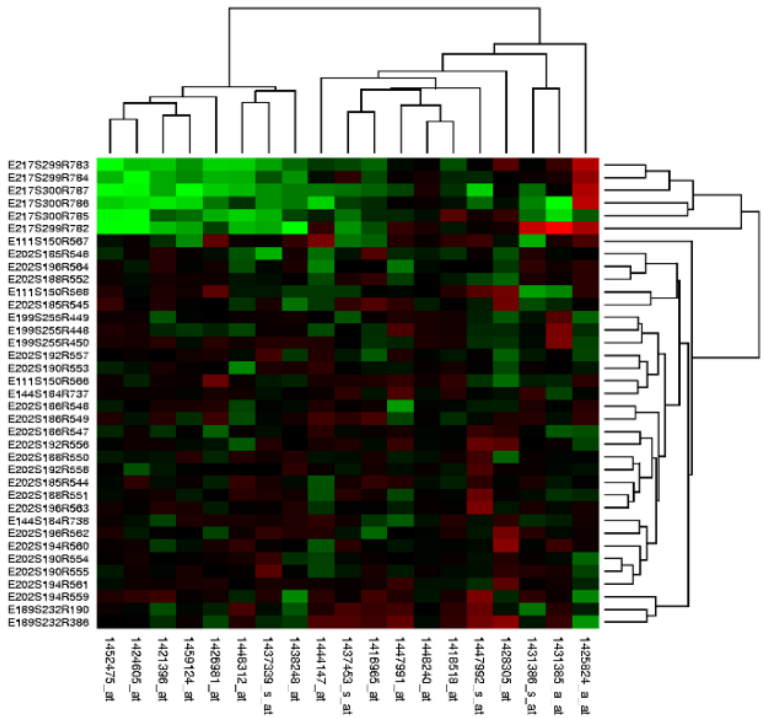


(d) Dataset4

Twitter Data Change Detection

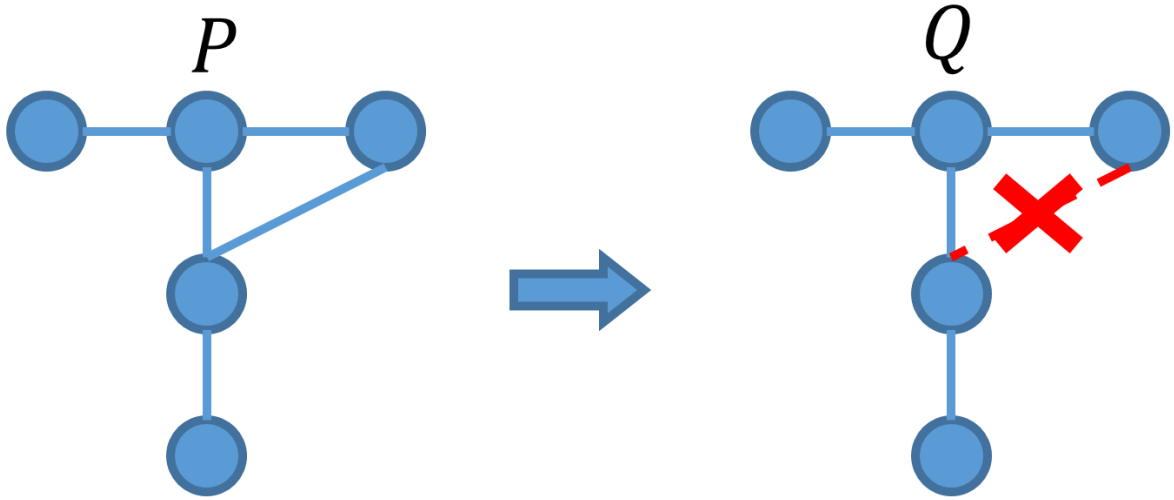


Changes in Interactions



- It is interesting to know interactions in many applications.
- However, the interactions change over time.

Changes in Graphical Models



- Given two sets of data

$$\{\boldsymbol{x}_p^{(i)}\}_{i=1}^{n_p} \sim P, \{\boldsymbol{x}_q^{(i)}\}_{i=1}^{n_q} \sim Q$$

  - where P and Q are Markov Networks (MNs) with respect to **undirected graphs** G_P and G_Q.
- We would like to know the changes from G_P to G_Q

Graphical Lasso [Friedman et al., 2008]

- One naive way is to learn two graphical models separately.
  - then take their differences.
- If you assume the density is Gaussian:

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

- We can learn a **sparse** Gaussian MN:

$$\hat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta}}{\operatorname{argmin}} - \sum_{i=1}^{n_p} \log p(\boldsymbol{x}^{(i)}; \boldsymbol{\Theta}) + \lambda \|\boldsymbol{\Theta}\|_1$$

- The sparsity of $\hat{\boldsymbol{\Theta}}$ indicates the conditional independence between random variables.

Fused Graphical Lasso [Zhang and Wang, 2010]

- Sparse changes does not necessarily come from sparse MNs
- A fancier way of learning changes is using the fused-lasso:

$$\{\hat{\boldsymbol{\Theta}}_p, \hat{\boldsymbol{\Theta}}_q\} = \underset{\boldsymbol{\Theta}_p, \boldsymbol{\Theta}_q}{\operatorname{argmin}} - \sum_{i=1}^{n_p} \log p(\boldsymbol{x}; \boldsymbol{\Theta}_p) - \sum_{i=1}^{n_q} \log q(\boldsymbol{x}; \boldsymbol{\Theta}_q) + \lambda \|\boldsymbol{\Theta}_p - \boldsymbol{\Theta}_q\|$$

- However, (Fused-) Graphical Lasso cannot handle non-Gaussian graphical model well due to the intractable normalization term.
  - e.g., in brain EEG analysis, the correlation is usually non-linear.

A pairwise MN parametrization

- Pairwise MN:

$$p(\boldsymbol{x};\boldsymbol{\theta}_p) = \frac{1}{Z(\boldsymbol{\theta}_p)} \exp\left(\sum_{u\leq v} \boldsymbol{\theta}_{p_{u,v}}^{\top} \boldsymbol{f}(x_u, x_v)\right)$$

$$Z(\boldsymbol{\theta}_p) = \int \exp\left(\sum_{u\leq v} \boldsymbol{\theta}_{p_{u,v}}^{\top} \boldsymbol{f}(x_u, x_v)\right) \mathrm{d}\boldsymbol{x}$$

- Computing $Z(\boldsymbol{\theta}_p)$ is hard!

Ratio Comes to Rescue

Taking the ratio:

$$\frac{p(\boldsymbol{x};\boldsymbol{\theta}_p)}{q(\boldsymbol{x};\boldsymbol{\theta}_q)} \propto \frac{\exp(\sum_{u\leq v} \boldsymbol{\theta}_{p_{u,v}}^\top \boldsymbol{f}(x_u, x_v))}{\exp(\sum_{u\leq v} \boldsymbol{\theta}_{q_{u,v}}^\top \boldsymbol{f}(x_u, x_v))}$$
$$= \exp(\sum_{u\leq v} (\boldsymbol{\theta}_{p_{u,v}} - \boldsymbol{\theta}_{q_{u,v}})^\top \boldsymbol{f}(x_u, x_v))$$

Letting $\boldsymbol{\theta} = \boldsymbol{\theta}_p - \boldsymbol{\theta}_q$

$$g(\boldsymbol{x};\boldsymbol{\theta}) := \frac{p(\boldsymbol{x};\boldsymbol{\theta}_p)}{q(\boldsymbol{x};\boldsymbol{\theta}_q)} = \frac{1}{N(\boldsymbol{\theta})} \exp(\sum_{u\leq v} \boldsymbol{\theta}_{u,v}^\top \boldsymbol{f}(x_u, x_v)).$$

where

$$N(\boldsymbol{\theta}) = \int q(\boldsymbol{x}) \exp(\sum_{u\leq v} \boldsymbol{\theta}_{u,v}^\top \boldsymbol{f}(x_u, x_v)) \mathrm{d}\boldsymbol{x}$$

$$\hat{N}(\boldsymbol{\theta}) = \frac{1}{n_q} \sum_{j=1}^{n_q} \exp(\sum_{u\leq v} \boldsymbol{\theta}_{u,v}^\top \boldsymbol{f}(x_u^{(j)}, x_v^{(j)})).$$

Learning Sparse Change Directly

- Since the parameter of density ratio reprents the difference between $\boldsymbol{\theta}_p$ and $\boldsymbol{\theta}_q$, we may apply sparsity inducing penalty on $\boldsymbol{\theta}$.

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \underbrace{-\sum_{i=1}^{n_p} \log \hat{g}(\boldsymbol{x}; \boldsymbol{\theta})}_{\ell(\boldsymbol{\theta})} + \lambda_{n_p} \sum_{u \leq v} \|\boldsymbol{\theta}_{u,v}\|_2$$

- By checking the sparsity pattern of subvector $\boldsymbol{\theta}_{u,v}$ we know whether the interactions between random variable X_u and X_v has changed or not.
- No problem if graphical model is not Gaussian.

Successful Change Detection Theorem [Liu et al., 2015]

Exists $\boldsymbol{\theta}^*$ such that $p(\boldsymbol{x}) = g(\boldsymbol{x}; \boldsymbol{\theta}^*)q(\boldsymbol{x})$.

Notations

- $H = \{(u, v) | u \leq v\}$,
- $S \in \{\boldsymbol{\theta}^*_{u,v} \neq \mathbf{0} | (u, v) \in H\}$, $S^c \in \{\boldsymbol{\theta}^*_{u,v} = \mathbf{0} | (u, v) \in H\}$
- Similarly, $\hat{S}$ and $\hat{S}^c$.

Assumptions

- $\Lambda_{\min}(\nabla_{\boldsymbol{\theta}_S} \nabla_{\boldsymbol{\theta}_S} \ell(\boldsymbol{\theta})) \geq \lambda_{\min} > 0$
- $\max_{t \in S^c} \left| \nabla_{\boldsymbol{\theta}_t} \nabla_{\boldsymbol{\theta}_S} \ell(\boldsymbol{\theta}) \left(\nabla_{\boldsymbol{\theta}_S} \nabla_{\boldsymbol{\theta}_S} \ell(\boldsymbol{\theta}) \right)^{-1} \right| \leq 1 - \alpha$
- $\nabla^2_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta} + \boldsymbol{\delta})$, $\max_{t \in S \cup S^c} \nabla_{\boldsymbol{\theta}_t} \nabla^2_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta} + \boldsymbol{\delta})$ bounded in spectral norm.
- $g(\boldsymbol{x}; \boldsymbol{\theta}) - 1$ is sub-Gaussian.

Successful Change Detection Theorem

Theorem

Suppose that Assumptions hold, as well as $\min_{t \in S} \|\boldsymbol{\theta}_t^*\| \geq \frac{10}{\lambda_{\min}} \sqrt{d} \lambda_{n_p}$ *are satisfied, where* d *is the number of changed edges defined as* $d = |S|$*, Suppose also*

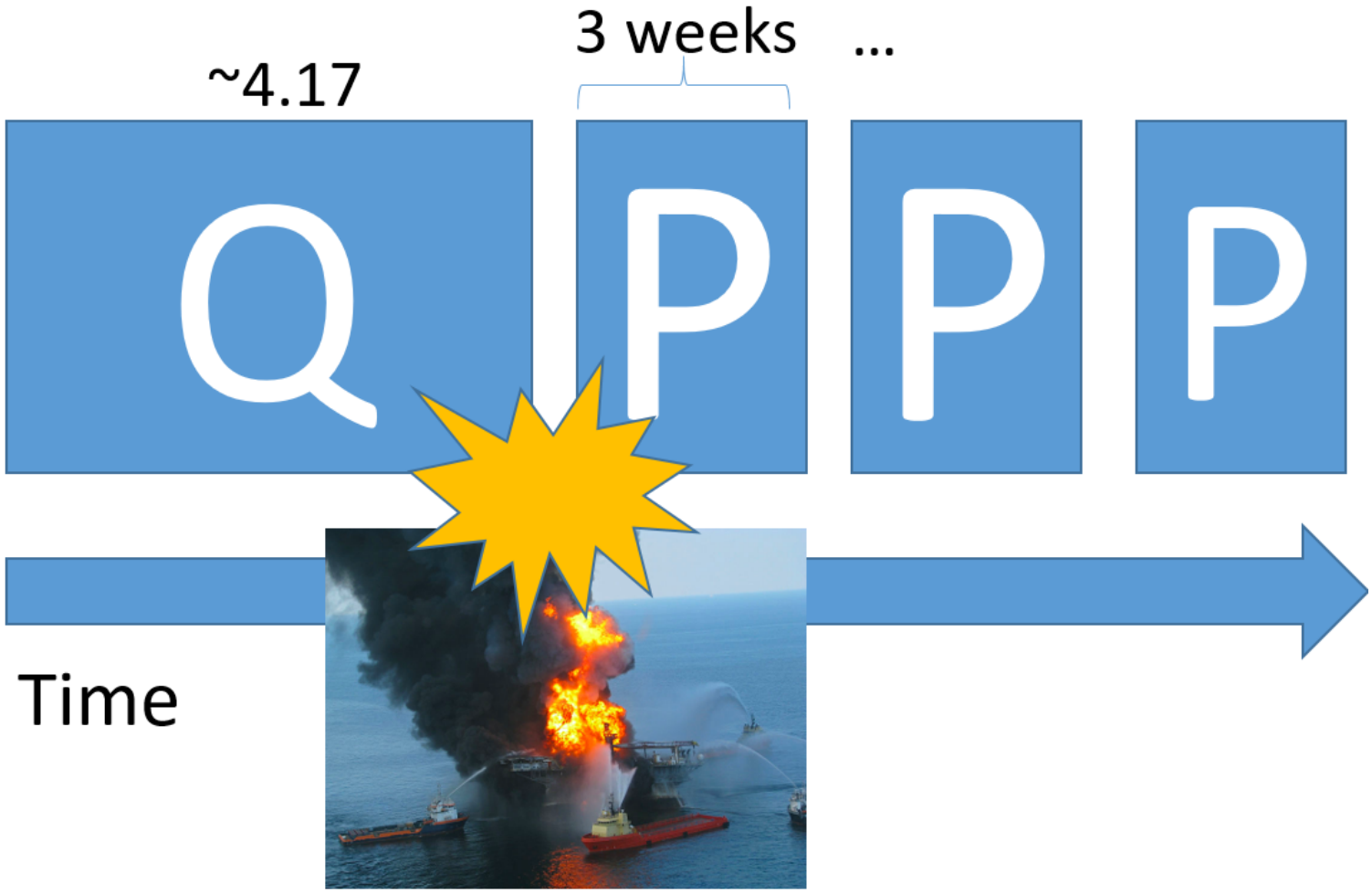
$$\frac{8(2-\alpha)}{\alpha} \sqrt{\frac{M_1 \log \frac{m^2+m}{2}}{n_p}} \leq \lambda_{n_p} \leq \frac{4(2-\alpha)M_1}{\alpha} \min\left(\frac{\|\boldsymbol{\theta}^*\|}{\sqrt{b}}, 1\right),$$

where $M_1 = \lambda_{max} b + 2$*,* $n_q \geq M_2 n_p^2 g(m)$ *and* M_2 *is a positive constant. Then there exist some constants* L_1*,* K_1*, and* K_2 *such that if* $n_p \geq L_1 d^2 \log \frac{m^2+m}{2}$*, with the probability at least*

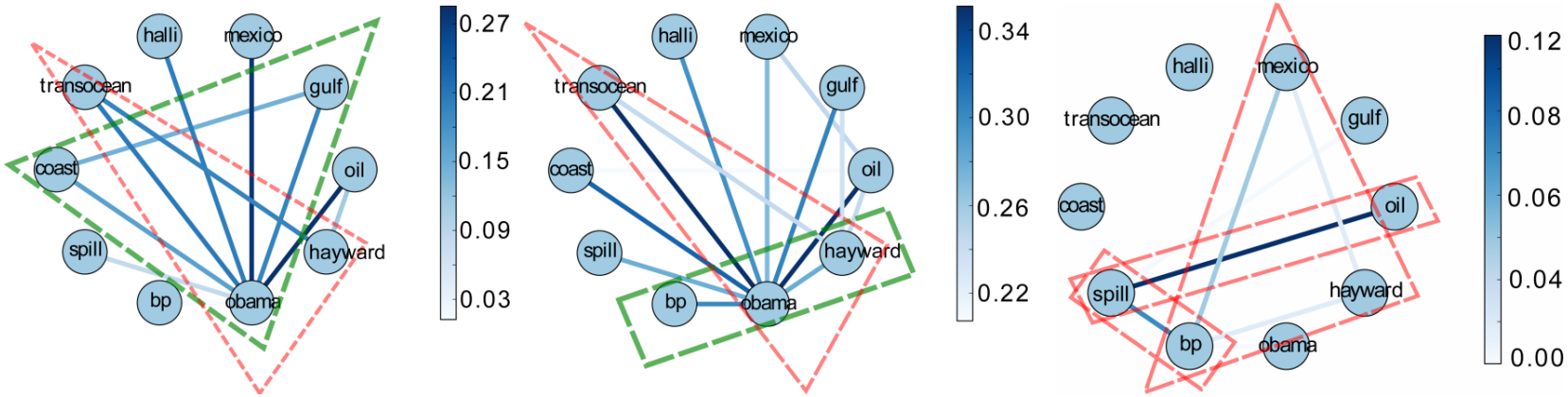
$$1 - \exp\left(-K_1 \lambda_{n_p}^2 n_p\right) - 4 \exp\left(-K_2 d n_q \lambda_{n_p}^4\right),$$

then the proposed method is consistent on learning changes between MNs.

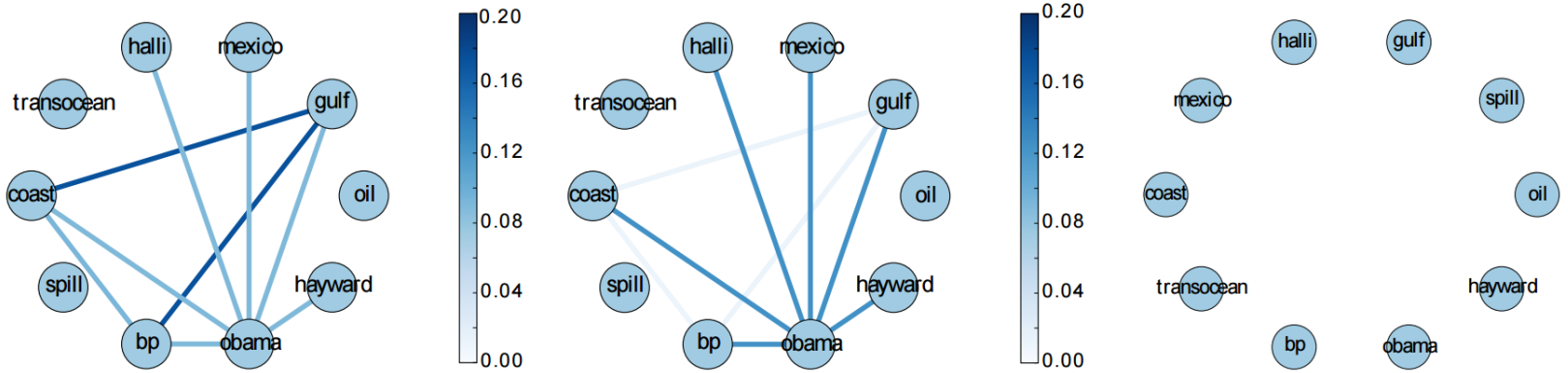
Twitter Dataset (BP Oil Spill)



Twitter Dataset ($m = 10, n = 84$)

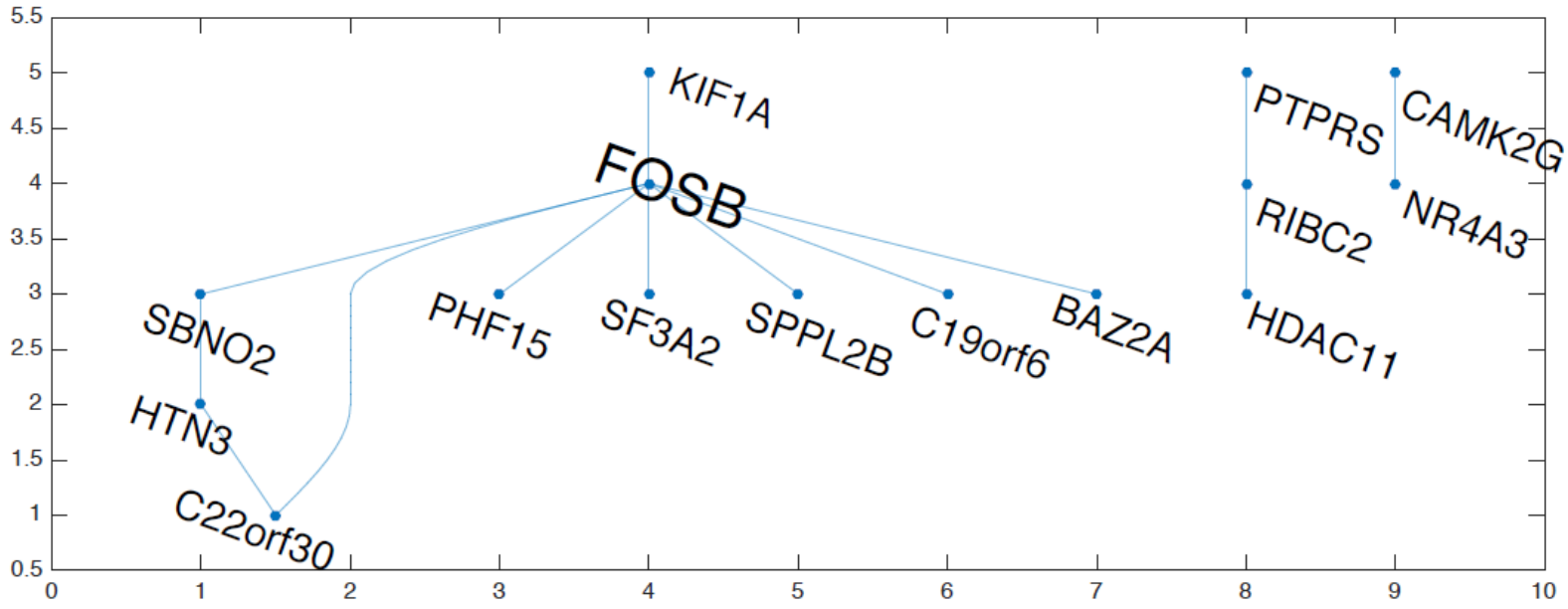


(a) April 17th–June 5th, KLIEP (b) June 6th–July 25th, KLIEP (c) July 26th–Sept. 14th, KLIEP



(d) April 17th–June 5th, Flasso (e) June 6th–July 25th, Flasso (f) July 26th–Sept. 14th, Flasso

Gene Dataset ($m = 1835, n_p = n_q = 28$)



FOSB gene is a member of the Fos family of transcription factors, regulating expressions of other genes.

The Transfer Learning

- Build a target classifier from limited samples of the **target task**

$$\mathcal{D}_p := \{(y, \boldsymbol{x}_p^{(i)})\}_{i=1}^{n} \sim P$$

- By making use of another set of samples from a similar **source task**

$$\mathcal{D}_q := \{(y, \boldsymbol{x}_q^{(i)})\}_{i=1}^{n'} \sim Q$$

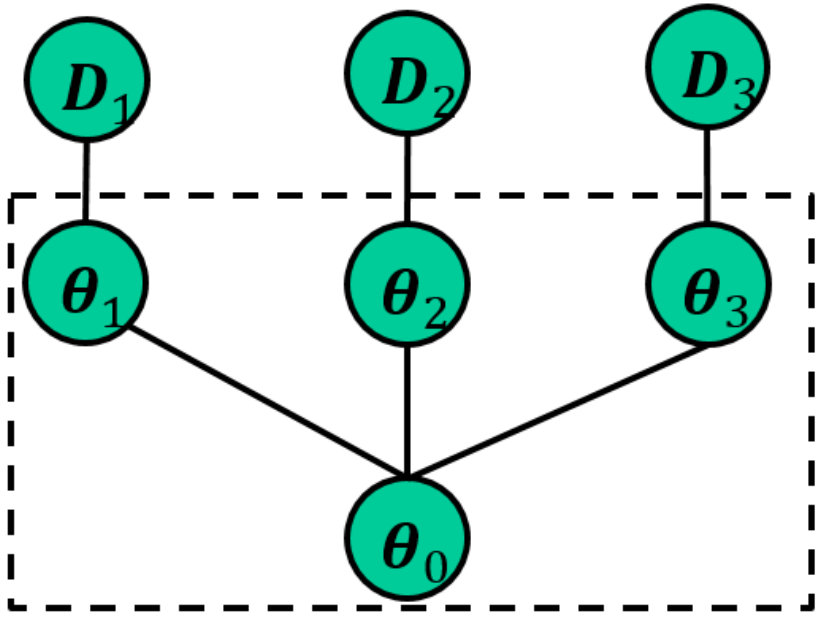
- $n \ll n'$
- We only consider learning a conditional probability $p(y|\boldsymbol{x})$ in this work.

The Transfer Learning

- In recent years, people seem to prefer complicated features (e.g., DNNs) that are computationally expensive.
- The transfer may get **unnecessarily** complicated!

Existing works

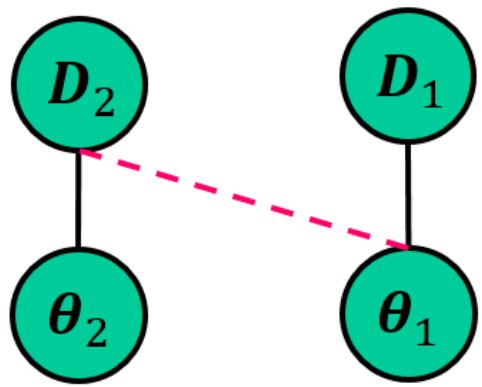
- Model Reuse: Parameters of predicting functions in similar tasks are **close** to each other.
  [Evgeniou and Pontil, 2004, Raina et al., 2006]
  - Solution: train two tasks simultaneously, and penalize the differences between parameters so they are not "too far away".
  - Problems: How close is close? Close in what metric?



A hierarchical model assumes parameters of similar tasks are generated from the same latent parameter.

Existing works

- Sample Reuse: Part of the source task samples can contribute to the target tasks. [Dai et al., 2007, Sugiyama et al., 2008b]
  - Solution: Weight samples!
  - Problems: Does not make use of the model similarity.



Both approaches have a common issue: during the **transferring stage**, the predicting function of the target task must be trained using the all features, however complicated they are.

A Composite Approac [Liu and Fukumizu,]

$$p(y|\boldsymbol{x}) = q(y|\boldsymbol{x})\frac{p(y|\boldsymbol{x})}{q(y|\boldsymbol{x})},$$

where $\frac{p(y|\boldsymbol{x})}{q(y|\boldsymbol{x})}$ is called **posterior ratio** and $q(y|\boldsymbol{x})$ is the source classifier.

- Idea: we can model and learn posterior ratio and source classifer separately!
- Hopefully, learning $\frac{p(y|\boldsymbol{x})}{q(y|\boldsymbol{x})}$ is computationally cheap!
  - Intuitively, the posterior ratio is an incremental pattern, that "patches" the source task predictor.
- Denote $g(y, \boldsymbol{x}; \boldsymbol{\theta})$ and $q(y, \boldsymbol{x}; \boldsymbol{\theta}_q)$ as the model of the ratio and the source classifier respectively.

A Composite Approach

- Naturally, we would like to minimize the KL-divergence between $p(y|\boldsymbol{x})$ and $g(y, \boldsymbol{x}; \boldsymbol{\theta})q(y, x; \boldsymbol{\theta}_q)$.
- However, directly minimizing such divergence still leads to a joint optimization.
  - $\sum_{y \in \{-1,1\}} g(y, \boldsymbol{x}; \boldsymbol{\theta})q(y, x; \boldsymbol{\theta}_q) = 1$.

Transfer Learning Upper-bound

if $\frac{p(y,\boldsymbol{x})}{q(y,\boldsymbol{x})} \leq C_{\max} < \infty$ and $0 < q_{\boldsymbol{\theta}} < 1$, then the following inequality holds

$$\mathrm{KL}\left[p \| g_{\boldsymbol{\theta}} \cdot q_{\boldsymbol{\theta}_q}\right] \leq \mathrm{KL}\left[p \| g_{\boldsymbol{\theta}} q\right] + C_{\max}\mathrm{KL}\left[q \| q_{\boldsymbol{\theta}_q}\right] + C',$$

where C' is a constant that is irrelevant to $\boldsymbol{\theta}$ or $\boldsymbol{\theta}$.

Separately learning two models become possible!

Is Learning the Posterior Ratio Easier?

Suppose

$$p(y|\boldsymbol{x};\boldsymbol{\theta}_q) \propto \exp\left(y \cdot \sum_{i=1}^{m} \theta_{q_i} h_i(\boldsymbol{x}))\right),$$

The ratio becomes

$$\frac{p(y|\boldsymbol{x};\theta_p)}{q(y|\boldsymbol{x};\theta_q)} \propto \exp\left(y \sum_{i=1}^{m} (\theta_{p,i} - \theta_{q,i}) h_i(\boldsymbol{x})\right),$$

and $(\theta_{p,i} - \theta_{q,i}) = 0$ if feature h_i does not contribute to the transfer!

Modelling Posterior Ratio

Thus, we write our posterior ratio model as

$$g(y, \boldsymbol{x}; \boldsymbol{\theta}) = \frac{1}{N(\boldsymbol{x}; \boldsymbol{\theta})} \exp\left(y \sum_{i \in S} \theta_i h_i(\boldsymbol{x}) \right),$$

where $S = \{i | \theta_{p,i} - \theta_{q,i} \neq 0\}$ and $N(\boldsymbol{x}; \boldsymbol{\theta})$ is the normalization term defined as

$$N(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{y \in \{-1,1\}} q(y|\boldsymbol{x}) \exp\left(y \sum_{i \in S} \theta_i h_i(\boldsymbol{x}) \right).$$

We assume $|S| \ll m$, so we have lightened the burden of transferring by not considering the full feature set.

Modelling Posterior Ratio

Define

$$\boldsymbol{f}(y, \boldsymbol{x}) := [yh_{a_1}(\boldsymbol{x}), yh_{a_2}(\boldsymbol{x}), \ldots, yh_{a_{m'}}(\boldsymbol{x})],$$

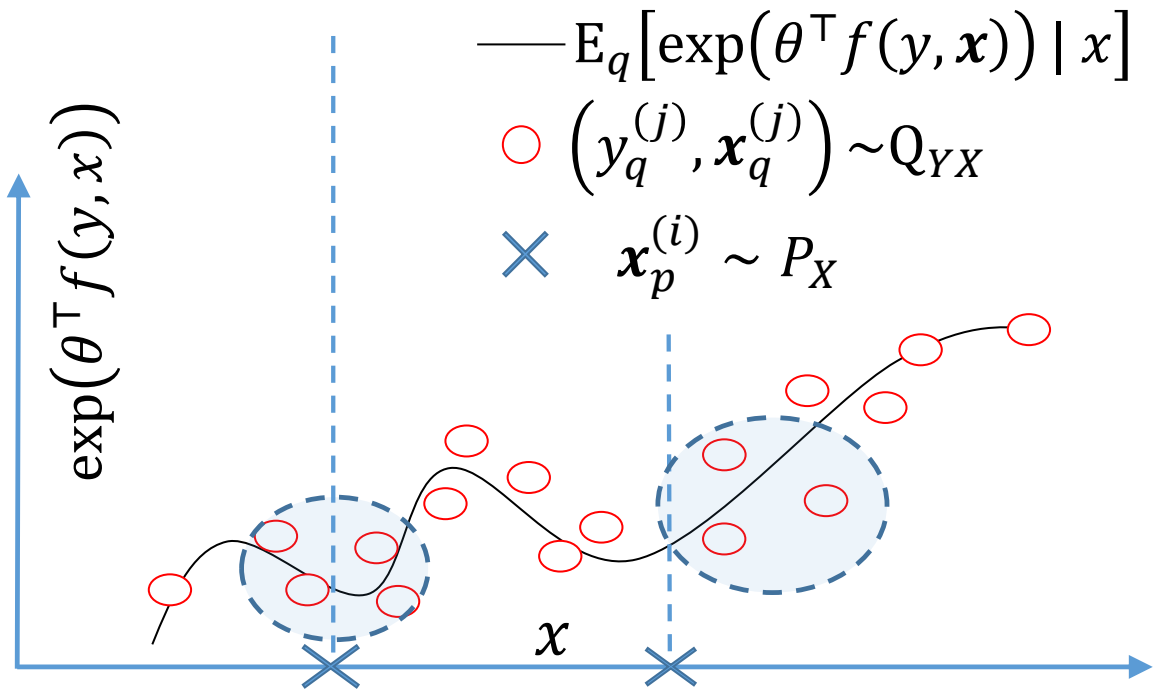
where $a_1, a_2, \ldots, a_{m'} \in S$.

$$N(\boldsymbol{\theta}, \boldsymbol{x}_p^{(i)}) \approx \widehat{N}\left(\boldsymbol{\theta}; \boldsymbol{x}_p^{(i)}\right) = \frac{1}{k} \sum_{j \in \mathcal{N}_{n'}\left(\boldsymbol{x}_p^{(i)}, k\right)} \exp\left(\boldsymbol{\theta}^\top \boldsymbol{f}\left(y_q^{(j)}, \boldsymbol{x}_q^{(j)}\right)\right),$$

where $\mathcal{N}_{n'}(\boldsymbol{x}_p^{(i)}, k) = \left\{ j \middle| \boldsymbol{x}_q^{(j)} \text{is one of the } k\text{-NNs of } \boldsymbol{x}_p^{(i)} \right\}$.

Finally, plug $\hat{g}(\boldsymbol{x}; \boldsymbol{\theta}) := \exp(\boldsymbol{\theta}^\top \boldsymbol{f}(y, \boldsymbol{x}))/\hat{N}(\boldsymbol{\theta})$ into KLIEP procedure, and we are done!

Modelling Posterior Ratio



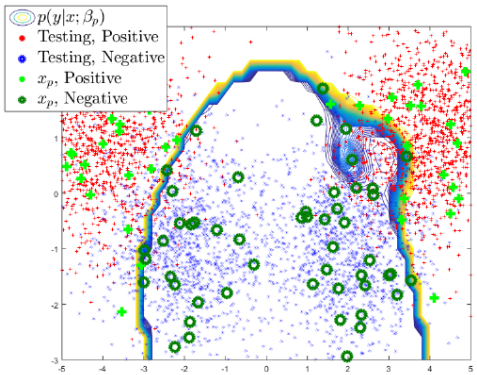
Consistency

Assumptions

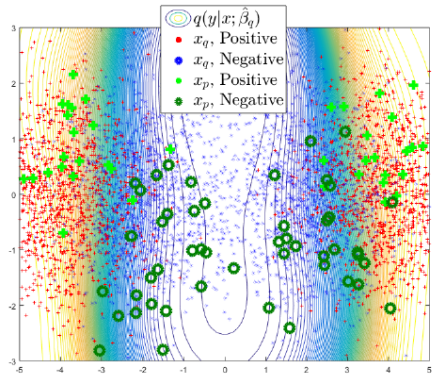
- The support of $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$ overlaps.
- The posterior model is bounded and is identifiable.

The estimated parameter $\hat{\boldsymbol{\theta}}$ converges to the true parameter $\boldsymbol{\theta}^*$ if $n \to \infty$, $n' \to \infty$, $k_{n'} / \log n' \to \infty$ and $k_{n'} / n' \to 0$.

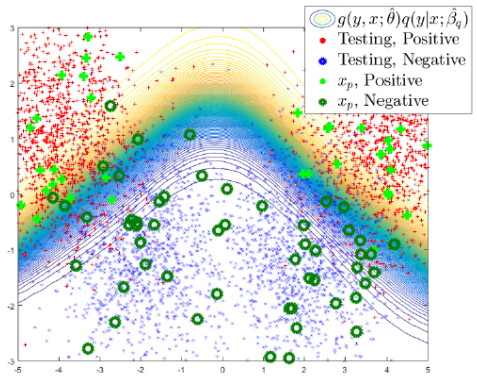
Experiments



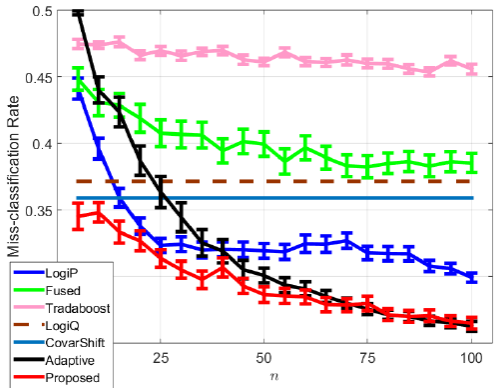
(d) $p(y|\boldsymbol{x};\hat{\boldsymbol{\beta}}_p)$, miss-rate: 13.8%.



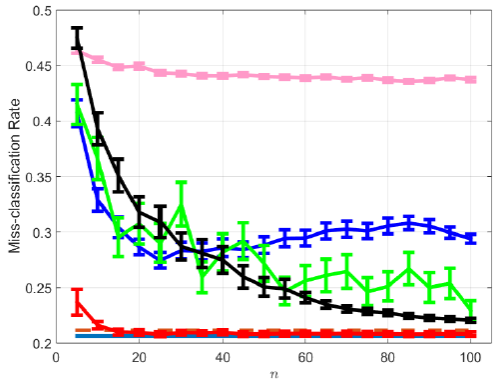
(e) $q(y|\boldsymbol{x};\hat{\boldsymbol{\beta}}_q)$, miss-rate: 15.2%



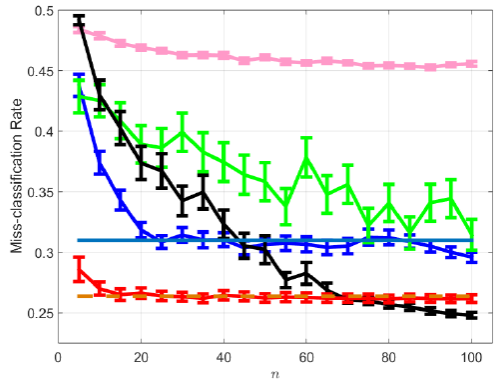
(f) $g(y,\boldsymbol{x};\hat{\boldsymbol{\theta}})q(y|\boldsymbol{x};\hat{\boldsymbol{\beta}}_q)$, miss-rate: 8.0%



(a) kitchen



(b) dvd



(c) books


Dai, W., Yang, Q., Xue, G. R., and Yu, Y. (2007).
Boosting for transfer learning.
In *Proceedings of the 24th International Conference on Machine Learning*, pages 193–200. ACM.


Evgeniou, T. and Pontil, M. (2004).
Regularized multi–task learning.
In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 109–117. ACM.


Friedman, J., Hastie, T., and Tibshirani, R. (2008).
Sparse inverse covariance estimation with the graphical lasso.
Biostatistics, 9(3):432–441.


Kanamori, T., Hido, S., and Sugiyama, M. (2009).
A least-squares approach to direct importance estimation.
Journal of Machine Learning Research, 10:1391–1445.


Kawahara, Y., Yairi, T., and Machida, K. (2007).

Change-point detection in time-series data based on subspace identification.
In *Proceedings of the 7th IEEE International Conference on Data Mining*, pages 559–564.


Kullback, S. and Leibler, R. A. (1951).
On information and sufficiency.
The Annals of Mathematical Statistics, 22:79–86.


Liu, S. and Fukumizu, K.
Estimating posterior ratio for classi
cation: Transfer learning from probabilistic perspective.
In *SIAM International Conference of Data Mining*.
To appear.


Liu, S., Suzuki, T., and Sugiyama, M. (2015).
Support consistency of direct sparse-change learning in markov networks.
In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI2015)*.


Liu, S., Yamada, M., Collier, N., and Sugiyama, M. (2013).

Change-point detection in time-series data by relative density-ratio estimation.
Neural Networks, 43:72–83.


Nguyen, X., Wainwright, M. J., and Jordan, M. I. (2010).
Estimating divergence functionals and the likelihood ratio by convex risk minimization.
IEEE Transactions on Information Theory, 56(11):5847–5861.


Quionero-Candela, J., Sugiyama, M., Schwaighofer, A., and Lawrence, N. D. (2009).
Dataset shift in machine learning.
The MIT Press.


Raina, R., Ng, A. Y., and Koller, D. (2006).
Constructing informative priors using transfer learning.
In *Proceedings of the 23rd International Conference on Machine Learning*, pages 713–720. ACM.


Sugiyama, M., Nakajima, S., Kashima, H., von Bünau, P., and Kawanabe, M. (2008a).

Direct importance estimation with model selection and its application to covariate shift adaptation.
In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T., editors, *Advances in Neural Information Processing Systems 20*. Curran Associates, Inc.


Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., and Kawanabe, M. (2008b).
Direct importance estimation for covariate shift adaptation.
Annals of the Institute of Statistical Mathematics, 60(4):699–746.


Yamada, M., Suzuki, T., Kanamori, T., Hachiya, H., and Sugiyama, M. (2013).
Relative density-ratio estimation for robust distribution comparison.
Neural Computation, 25(5):1324–1370.


Zhang, B. and Wang, Y. (2010).
Learning structural changes of Gaussian graphical models in controlled experiments.

In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence (UAI2010)*, pages 701–708.