Machine learning for brain imaging across distinct databases

Pablo M. Olmos, Jussi Tohka

olmos@tsc.uc3m.es, jtohka@ing.uc3m.es

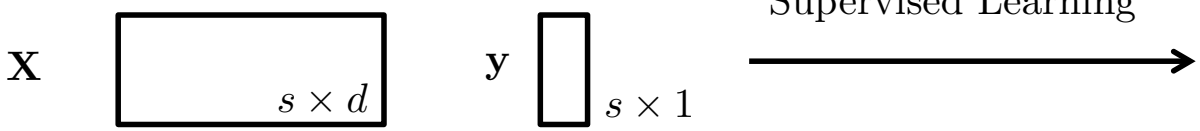
Universidad
Carlos III de Madrid

Today

- The **MULTIBRAIN** project: scalable domain adaptation and transfer learning for brain imaging.
- A (personal) literature review on domain adaptation techniques.
- Discussion, brainstorming, suggestions ...

Index

Data aggregation in supervised learning



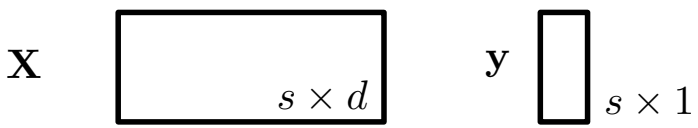
Discriminative approach

$h_\theta(\mathbf{x}) \approx p(y|\mathbf{x})$

Generative approach

$h_\theta(\mathbf{x}, y) \approx p(\mathbf{x}, y)$

Data aggregation in supervised learning



Supervised Learning

Discriminative approach

$$h_\theta(\mathbf{x}) \approx p(y|\mathbf{x})$$

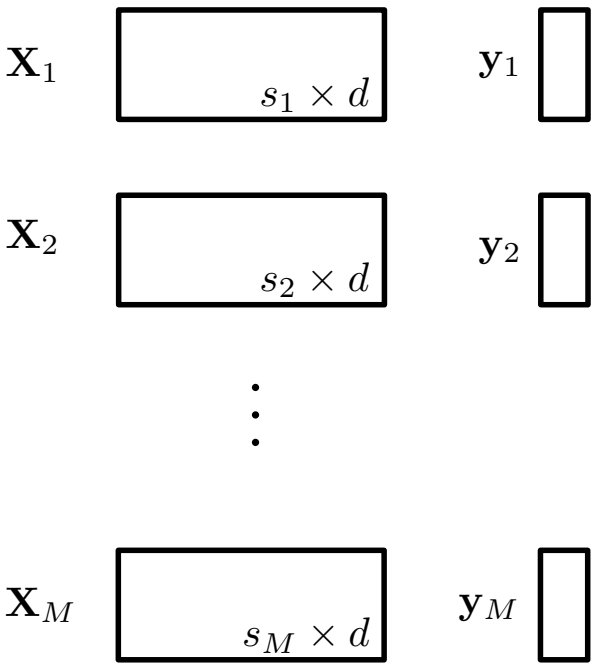
Generative approach

$$h_\theta(\mathbf{x}, y) \approx p(\mathbf{x}, y)$$

Alzheimers disease detection using brain MRIs

- Single database $\rightarrow$ all subjects under the same measurement conditions (hopefully)
- Number of subjects s is small (hundreds in the best case?)
- d is huge (E.g. $256 \times 256 \times 170$ voxels, $d \approx 10^7$).

Data aggregation in supervised learning



Data agregation

Supervised Learning

Discriminative approach

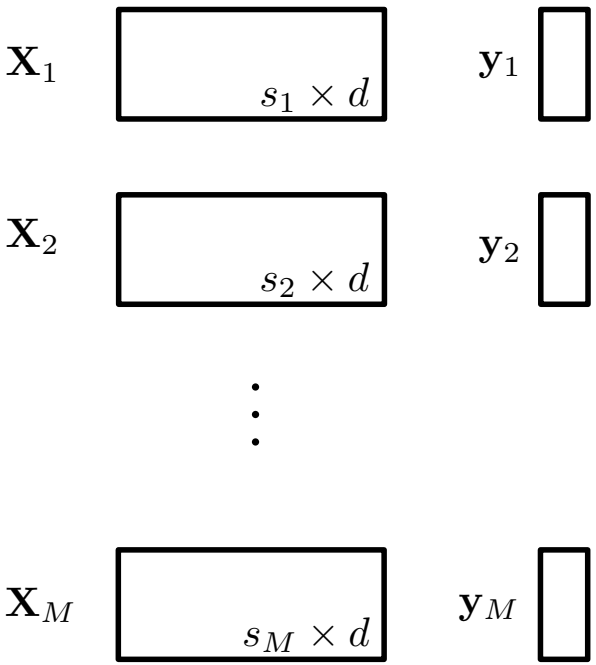
$$h_\theta(\mathbf{x}) \approx p(y|\mathbf{x})$$

Generative approach

$$h_\theta(\mathbf{x}, y) \approx p(\mathbf{x}, y)$$

When the multisite data are acquired in an **uncontrolled manner**, traditional supervised learning methods suffer from considerable problems. **Simple data aggregation does not work well**

Data aggregation in supervised learning



Data agregation

Supervised Learning

Discriminative approach

$h_\theta(\mathbf{x}) \approx p(y|\mathbf{x})$

Generative approach

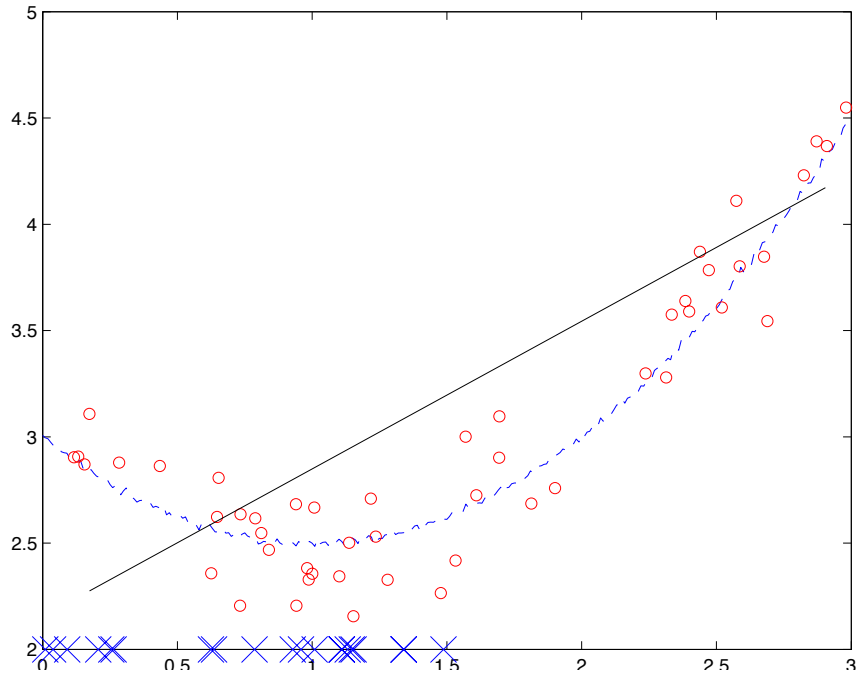
$h_\theta(\mathbf{x}, y) \approx p(\mathbf{x}, y)$

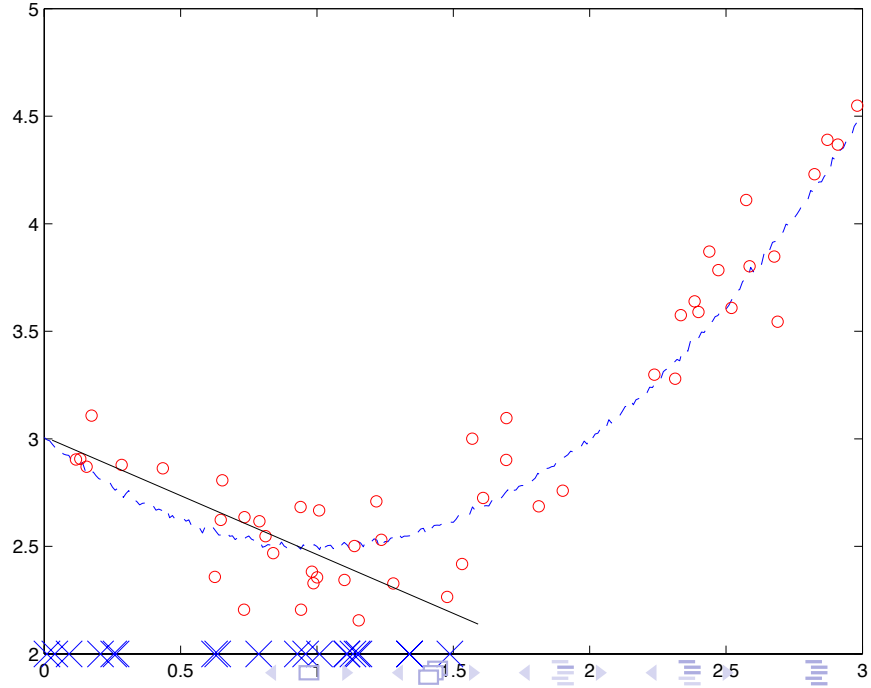
When the multisite data are acquired in an **uncontrolled manner**, traditional supervised learning methods suffer from considerable problems. **Simple data aggregation does not work well**

Underlying (typically wrong) assumption: all the data are drawn from the same feature space and the same distribution.

$\mathbf{X}_1$ $s \times d$ $\mathbf{y}$ $s \times 1$

$\mathbf{X}_2$



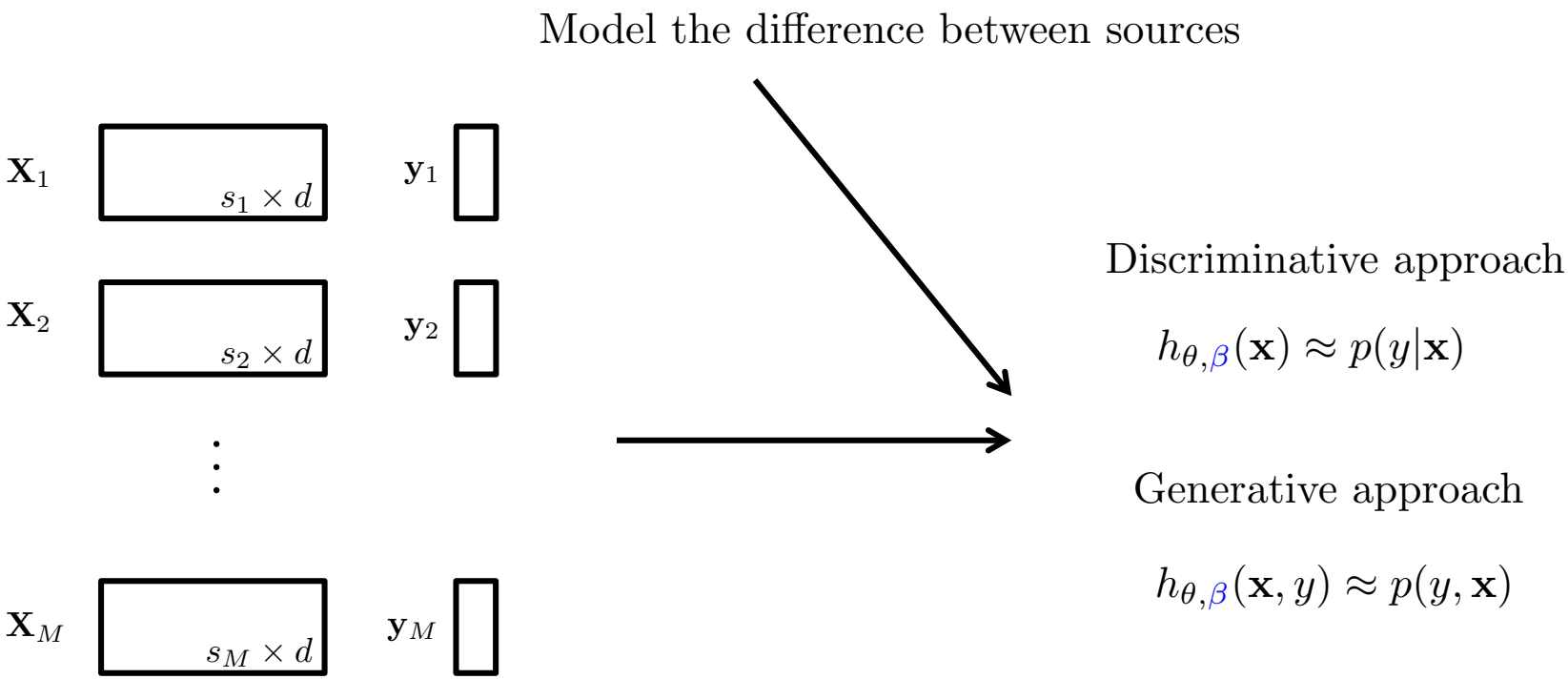


Domain Adaptation (DA)

- Discovering a good feature representation across domains.

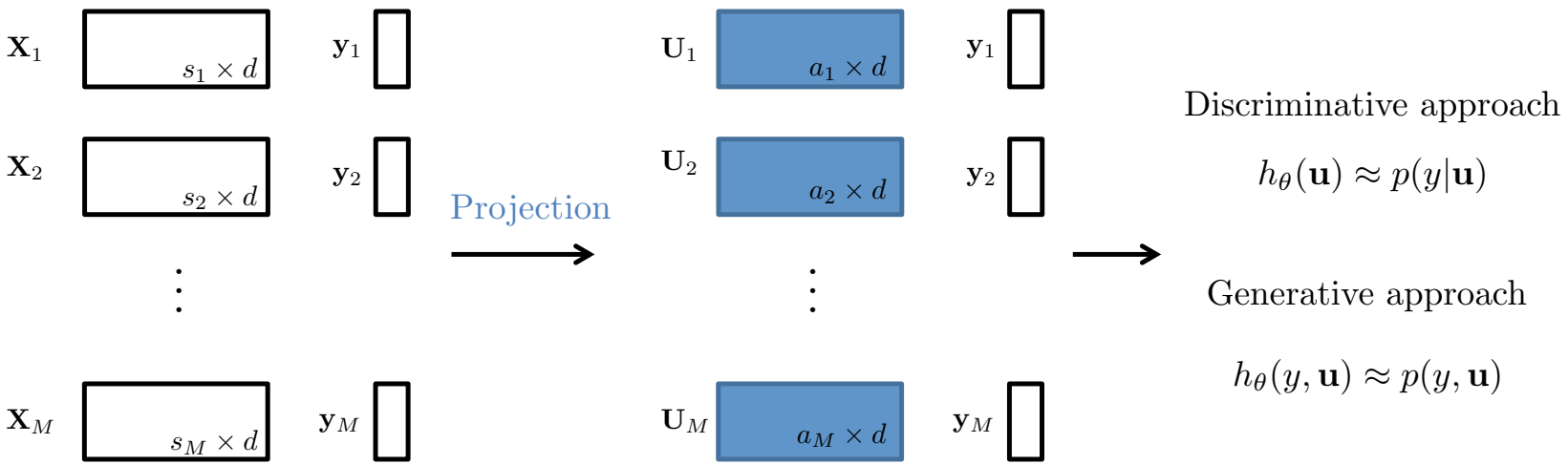
Domain Adaptation (DA)

- Discovering a good feature representation across domains.
- **Characterizing/modeling the divergence between the distributions**. Incorporating this model into the regression/classification task.



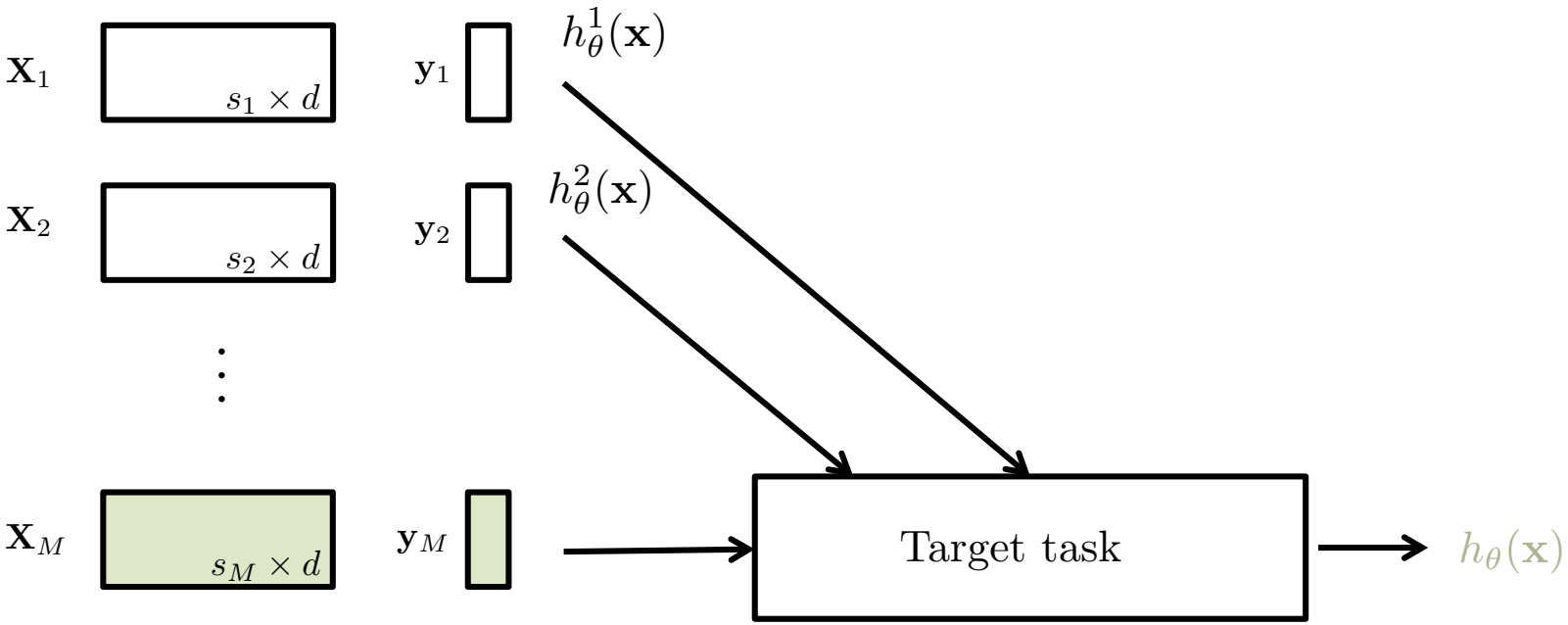
Domain Adaptation (DA)

- Discovering a good feature representation across domains.
- **Characterizing/modeling the divergence between the distributions**. Incorporating this model into the regression/classification task.
- **Projecting into a latent feature space where data from distinct databases are similarly distributed**. Perform the regression/classification task over this space.



Domain Adaptation (DA)

- Discovering a good feature representation across domains.
- **Characterizing/modeling the divergence between the distributions**. Incorporating this model into the regression/classification task.
- **Projecting into a latent feature space where data from distinct databases are similarly distributed**. Perform the regression/classification task over this space.
- **Combine models already trained in each database.**



Transfer Learning

- Closely related to DA.
- **Hal Daum III's Blog**: *"Roughly speaking, domain adaptation (DA) is the problem that occurs when $p(X)$ changes between training and test. Transfer learning (TL) is the problem that occurs when $p(Y|X)$ changes between training and test.*

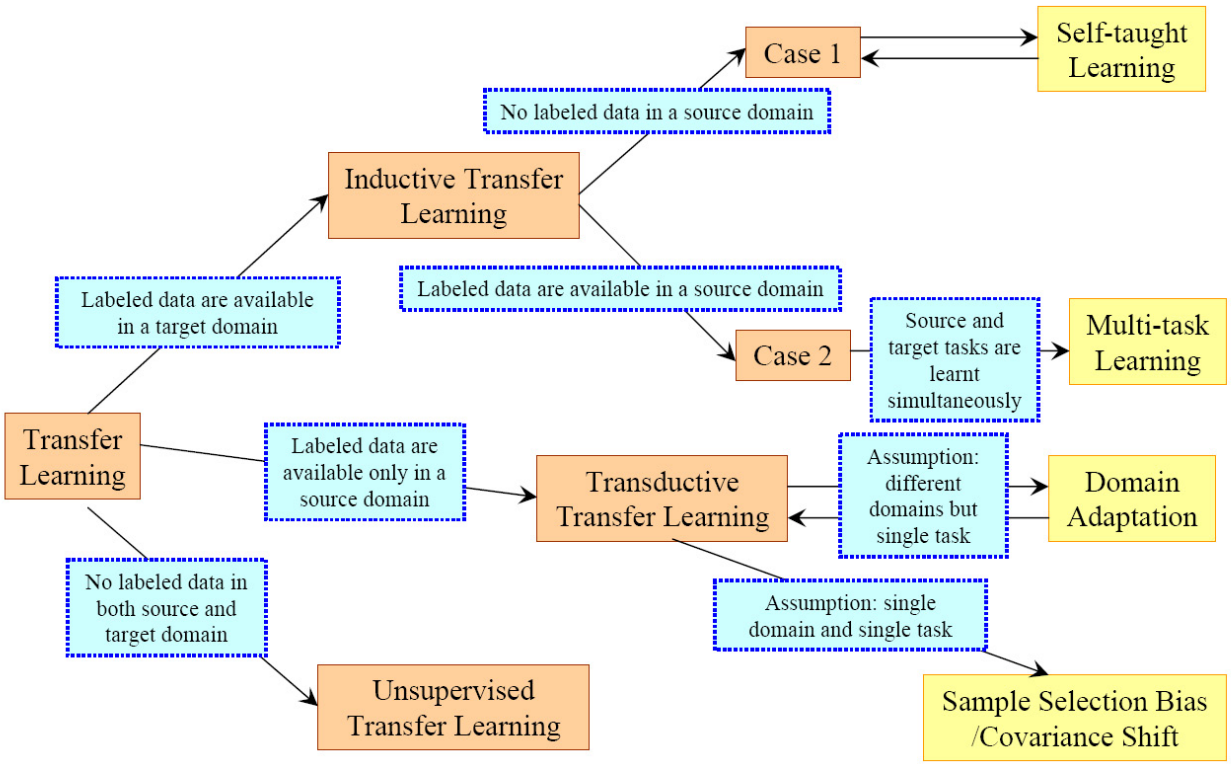


Figure: Source: A survey on Transfer Learning. Sinno Jialin Pan and Qiang Yang, IEEE Trans. Knowledge and Data Engineering

MULTIBRAIN project

- Develop **scalable domain adaptation techniques** to be able to combine data from **multiple brain imaging databases**, learning the relevant differences across databases and minimizing the across-site variability.
- This is radically novel in brain imaging, where the typical solution is try to standardize the data acquisition to minimize the across-site differences in the data.
- Important challenges: **Dimensionality of data (the number of voxels in the images of a single subject)**, **Small-sized databases**
- Simple models + Bayesian Learning (uncertainty estimation) + joint DA and regression/classification

Promising preliminary results, there's hope to have an impact in the field.

Index

Notation

- $\mathbf{x} \in \mathbb{R}^d$, $y \in \{-1, +1\}$ (classification), $y \in \mathbb{R}$ (regression)
- Source Domain:

$$p_s(\mathbf{x}, y) = p_s(y|\mathbf{x}_t)p_s(\mathbf{x}) \rightarrow \text{Samples: } \{\mathbf{x}_s^{(i)}, y_s^{(i)}\}_{i=1}^{n_s}$$

- Target Domain:

$$p_t(\mathbf{x}, y) = p_t(y|\mathbf{x}_t)p_t(\mathbf{x}) \rightarrow \text{Samples: } \{\mathbf{x}_t^{(i)}, y_t^{(i)}\}_{i=1}^{n_t}$$
$$\{\mathbf{x}_{t,u}^{(i)}\}_{i=1}^{n_u}$$

- **Unsupervised DA:** $n_t = 0$ (No target labels)
- **Semi-supervised DA:** $n_t << n_s$ (a few target labels)

Weighting the log-likelihood (Shimodaira'00)

- Unsupervised DA
- Assume $p_s(\mathbf{x})$ and $p_t(\mathbf{x})$ are defined over the same support
- **Covariance shift:** $p_s(y|\mathbf{x}_t) = p_t(y|\mathbf{x}_t) = p(y|\mathbf{x})$ but $p_s(\mathbf{x}) \neq p_t(\mathbf{x})$
- Let $h_\theta(\mathbf{x})$ be our model to $p(y|\mathbf{x})$

$$\theta^* = \max_\theta \sum_{i=1}^{n_s} \frac{p_t(\mathbf{x}_s^{(i)})}{p_s(\mathbf{x}_s^{(i)})} \log h_\theta(\mathbf{x}_s^{(i)})$$

In the limit $n_s \to \infty$,

$$n_s^{-1} \sum_{i=1}^{n_s} \frac{p_t(\mathbf{x}_s^{(i)})}{p_s(\mathbf{x}_s^{(i)})} \log h_\theta(\mathbf{x}_s^{(i)}) \approx \mathbb{E}_{p_s(\mathbf{x})p(y|\mathbf{x})} \left\{ \frac{p_t(\mathbf{x})}{p_s(\mathbf{x})} \log h_\theta(\mathbf{x}) \right\}$$

$$= \int p_s(\mathbf{x}) p(y|\mathbf{x}) \frac{p_t(\mathbf{x})}{p_s(\mathbf{x})} \log h_\theta(\mathbf{x}_s) \mathrm{d}\mathbf{x} \mathrm{d}y$$

$$= \mathbb{E}_{p_t(\mathbf{x})p(y|\mathbf{x})} \{ \log h_\theta(\mathbf{x}) \}$$

By maximizing the weighted log-likelihood function, we are maximizing the log-likelihood function in the target domain!

We have to estimate the weights ...

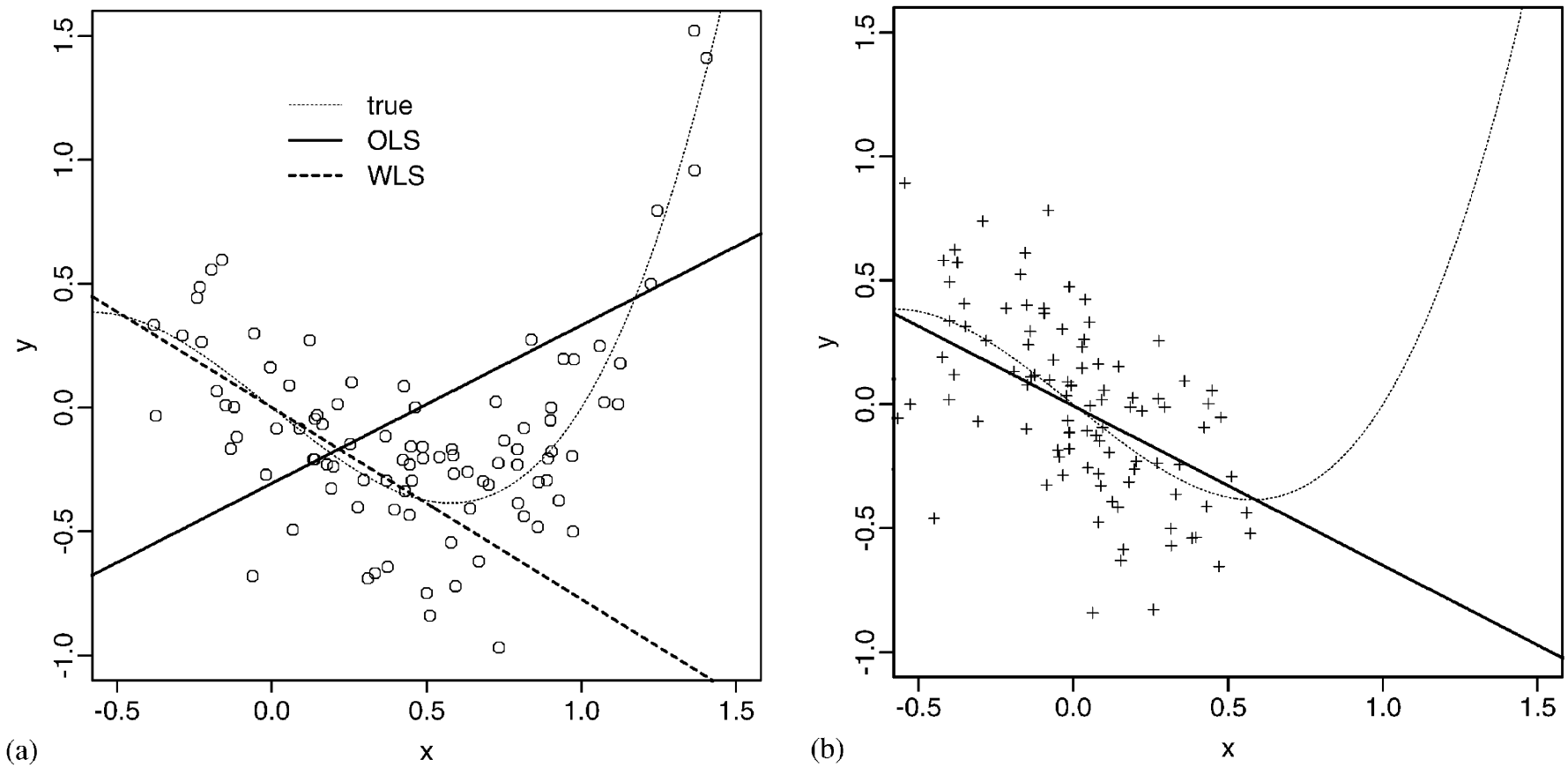


Figure: Source: Shimodaira'00

ELSEVIER

Journal of Statistical Planning and Inference 90 (2000) 227–244

journal of statistical planning and inference

www.elsevier.com/locate/jspi

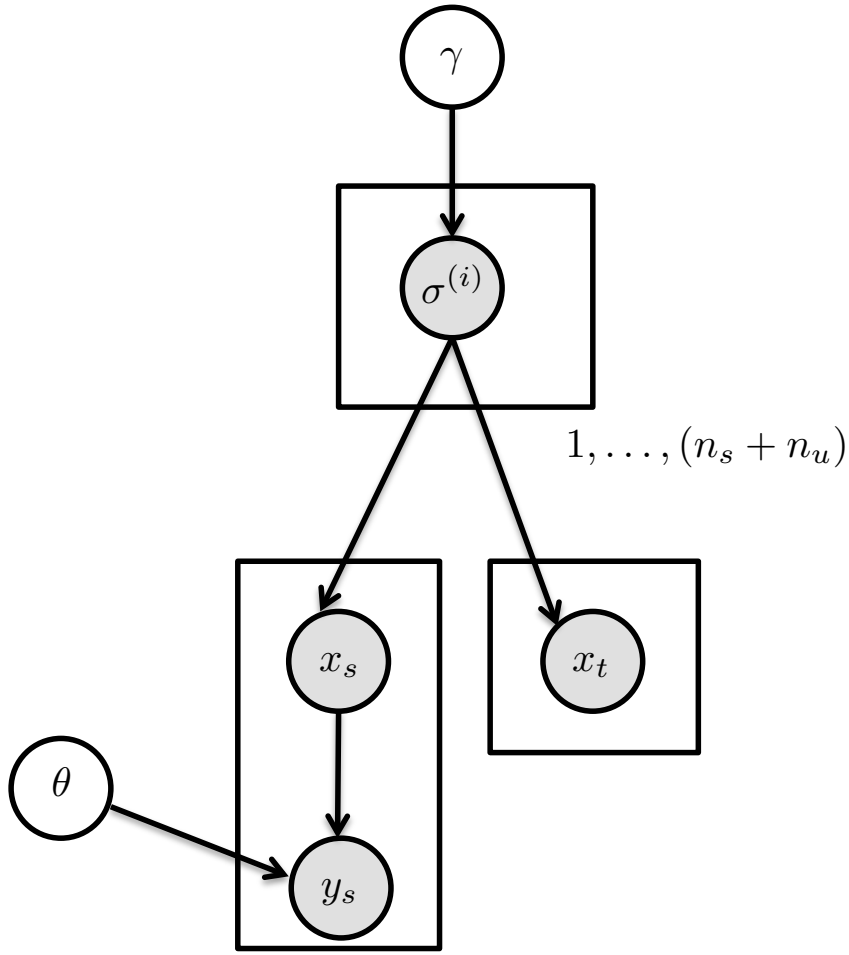
Improving predictive inference under covariate shift by weighting the log-likelihood function

Hidetoshi Shimodaira *

Sample rejection process to model the difference (Bickel et al, JMLR 07)

- **Covariance shift:** $p_s(y|\mathbf{x}_t) = p_t(y|\mathbf{x}_t) = p(y|\mathbf{x})$ but $p_s(\mathbf{x}) \neq p_t(\mathbf{x})$
- Generative model $\rightarrow$ explain how data was generated $\rightarrow$ estimate the weights $\frac{p_t(\mathbf{x}_s^{(i)})}{p_s(\mathbf{x}_s^{(i)})}$
- Let $h_\theta(\mathbf{x})$ be our model to $p(y|\mathbf{x})$
- We model the way the (observed) data has been generated using a latent random binary variable σ, the *selector variable*:
  - $\sigma = 1$ indicates that a sample $\mathbf{x}$ is drawn according to $p_t(\mathbf{x})$
  - $\sigma = 0$ indicates that a sample $\mathbf{x}$ is drawn according to $p_s(\mathbf{x})$
- Let $f_\gamma(\mathbf{x})$ be our model to $p(\sigma = 1|\mathbf{x})$ (E.g. logistic regression)

The generative process of the data



Likelihood of the model

$$\log p(\mathbf{X}_s, \mathbf{X}_t, \mathbf{y}_s|\theta, \gamma) = \log p(\mathbf{y}_s|\mathbf{X}_s, \theta, \gamma) + \log p(\mathbf{X}_s, \mathbf{X}_t|\gamma)$$

Likelihood of the model

$$\log p(\mathbf{X}_s, \mathbf{X}_t, \mathbf{y}_s|\theta, \gamma) = \log p(\mathbf{y}_s|\mathbf{X}_s, \theta, \gamma) + \log p(\mathbf{X}_s, \mathbf{X}_t|\gamma)$$

$$\log p(\mathbf{y}_s|\mathbf{X}_s, \theta, \gamma) = \sum_{i=1}^{n_s} \frac{p_t(\mathbf{x}_s^{(i)})}{p_s(\mathbf{x}_s^{(i)})} \log h_\theta(\mathbf{x}_s^{(i)})$$

Likelihood of the model

$$\log p(\mathbf{X}_s, \mathbf{X}_t, \mathbf{y}_s|\theta, \gamma) = \log p(\mathbf{y}_s|\mathbf{X}_s, \theta, \gamma) + \log p(\mathbf{X}_s, \mathbf{X}_t|\gamma)$$

$$\log p(\mathbf{y}_s|\mathbf{X}_s, \theta, \gamma) = \sum_{i=1}^{n_s} \frac{p_t(\mathbf{x}_s^{(i)})}{p_s(\mathbf{x}_s^{(i)})} \log h_\theta(\mathbf{x}_s^{(i)})$$

$$\frac{p_t(\mathbf{x})}{p_s(\mathbf{x})} = \frac{p(\sigma = 1)}{p(\sigma = 0)} \left(\frac{1}{p(\sigma = 1|\mathbf{x})} - 1 \right) \approx \frac{\pi}{1 - \pi} \left(\frac{1}{f_\gamma(\mathbf{x})} - 1 \right)$$

Likelihood of the model

$$\log p(\mathbf{X}_s, \mathbf{X}_t, \mathbf{y}_s|\theta, \gamma) = \log p(\mathbf{y}_s|\mathbf{X}_s, \theta, \gamma) + \log p(\mathbf{X}_s, \mathbf{X}_t|\gamma)$$

$$\max_{\theta,\gamma,\pi^*} \sum_{i=1}^{n_s} \frac{\pi}{1-\pi}\left(\frac{1}{f_\gamma(\mathbf{x}_s^i)} - 1\right) \log h_\theta(\mathbf{x}_s^{(i)}) + \sum_{i=1}^{n_s} \log f_\gamma(\mathbf{x}_s^i) + \sum_{i=1}^{n_u} \log(1 - f_\gamma(\mathbf{x}_u^i))$$

Journal of Machine Learning Research 10 (2009) 2137-2155 Submitted 4/08; Revised 7/09; Published 9/09

Discriminative Learning Under Covariate Shift

Steffen Bickel BICKEL@CS.UNI-POTSDAM.DE
Michael Brückner MIBRUECK@CS.UNI-POTSDAM.DE
Tobias Scheffer SCHEFFER@CS.UNI-POTSDAM.DE
University of Potsdam, Department of Computer Science
August-Bebel-Str. 89
14482 Potsdam, Germany

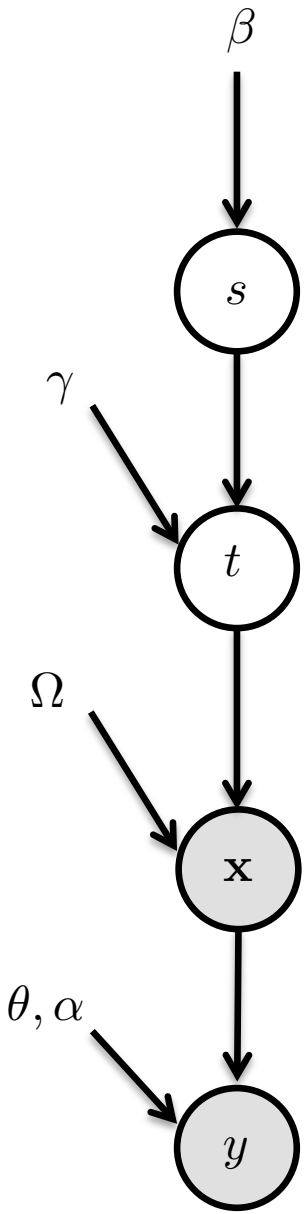
Mixture regression for Covariance Shift (Storkey, Sugiyama, NIPS 07)

- Each daum $\mathbf{x}$ is assumed to have been generated from one of a number of data sources using a mixture distribution.
- The proportion of each of the sources varies across source domain and target domain

$$p_s(\mathbf{x}) \rightarrow \sum_t \beta_1 \gamma_{1t} p_{1t}(\mathbf{x}) + \beta_2 \gamma_{2t} p_{2t}(\mathbf{x})$$

$$p_t(\mathbf{x}) \rightarrow \sum_t \alpha_{1t} p_{1t}(\mathbf{x})$$

- Regression model for $p_{1t}(\mathbf{x})$ sources, $p_1(y|\mathbf{x}, \theta)$
- Regression model for $p_{2t}(\mathbf{x})$ sources, $p_2(y|\mathbf{x}, \alpha)$



Parameters of the model are adjusted using EM

Mixture Regression for Covariate Shift

Amos J Storkey
Institute of Adaptive and Neural Computation
School of Informatics, University of Edinburgh
a.storkey@ed.ac.uk

Masashi Sugiyama
Department of Computer Science
Tokyo Institute of Technology
sugi@cs.titech.ac.jp

Learning of Discriminative Clusters for Unsupervised DA (Yuan Shi, Fei Sha, ICML 2012)

- It is assumed that there is a latent feature space (defined by a linear transformation $\mathbf{z} = \mathbf{L}\mathbf{x}$) where data in the source and target domains form well-separated clusters.
- The clusters from the source domain are geometrically close to those from the target domain if they are from the same labels.

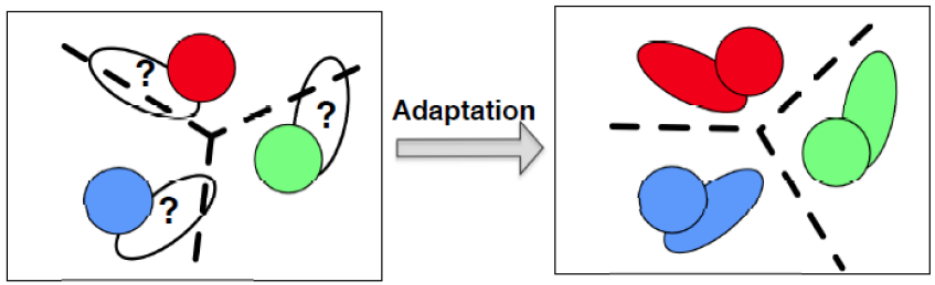


Figure: Source: (Yuan Shi, Fei Sha, ICML 2012)

- 1-NN classification in the feature space
- $\mathbf{L}$ is found my minimizing the misclassification error using both target and source labels, with certain regularity conditions.

Information-Theoretical Learning of Discriminative Clusters for Unsupervised Domain Adaptation

Yuan Shi YUANSHI@USC.EDU
U. of Southern California, Los Angeles, CA 90089 USA

Fei Sha FEISHA@USC.EDU
U. of Southern California, Los Angeles, CA 90089 USA

- A lot of interesting papers using a similar idea, but projecting in a Kernel Space.

A Kernel Method for the Two-Sample-Problem

Arthur Gretton
MPI for Biological Cybernetics
Tübingen, Germany
arthur@tuebingen.mpg.de

Karsten M. Borgwardt
Ludwig-Maximilians-Univ.
Munich, Germany
kb@dbs.ifi.lmu.de

Malte Rasch
Graz Univ. of Technology,
Graz, Austria
malte.rasch@igi.tu-graz.ac.at

Bernhard Schölkopf
MPI for Biological Cybernetics
Tübingen, Germany
bs@tuebingen.mpg.de

Alexander J. Smola
NICTA, ANU
Canberra, Australia
Alex.Smola@anu.edu.au

Kernel Manifold Alignment

Devis Tuia
Department of Geography, University of Zurich, Switzerland
devis.tuia@geo.uzh.ch
Gustau Camps-Valls
Image Processing Laboratory, Universitat de València, Spain
gustau.camps@uv.es

June 9, 2015

Semi-Supervised Kernel Matching for Domain Adaptation

Min Xiao and **Yuhong Guo**
Department of Computer and Information Sciences
Temple University
Philadelphia, PA 19122, USA
{minxiao, yuhong}@temple.edu

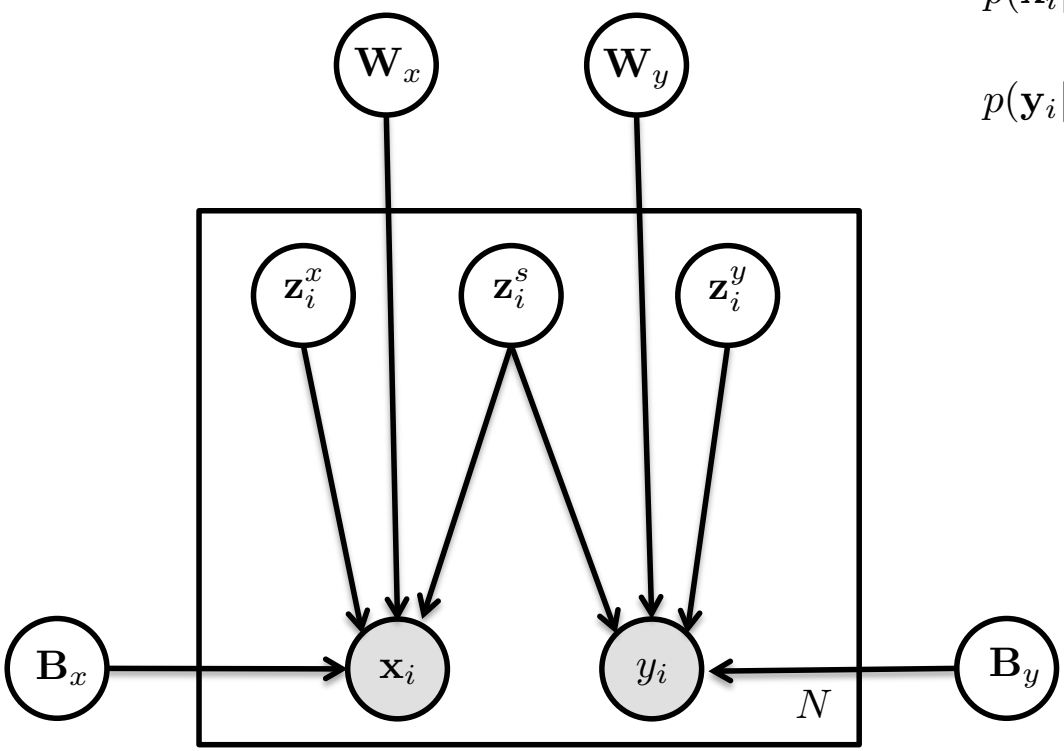
Projecting into a latent space via Canonical Correlation Analysis

- CCA $\rightarrow$ modeling dependencies between two or more sets of variables $\rightarrow$ low-dimensional space defined by directions of maximal correlation.
- Multi-view learning, multi-label prediction, domain adaptation ...

$$p(\mathbf{z}_i) = \mathcal{N}(\mathbf{z}_i^s|\mathbf{0}, \mathbf{I}_{Ls})\mathcal{N}(\mathbf{z}_i^x|\mathbf{0}, \mathbf{I}_{Lx})\mathcal{N}(\mathbf{z}_i^y|\mathbf{0}, \mathbf{I}_{Ly})$$

$$p(\mathbf{x}_i|\mathbf{z}_i) = \mathcal{N}(\mathbf{x}_i|\mathbf{B}_x\mathbf{z}_i^x + \mathbf{W}_x\mathbf{z}_i^s + \mu_x, \sigma^2\mathbf{I}_{Dx})$$

$$p(\mathbf{y}_i|\mathbf{z}_i) = \mathcal{N}(\mathbf{y}_i|\mathbf{B}_y\mathbf{z}_i^y + \mathbf{W}_y\mathbf{z}_i^s + \mu_y, \sigma^2\mathbf{I}_{Dy})$$



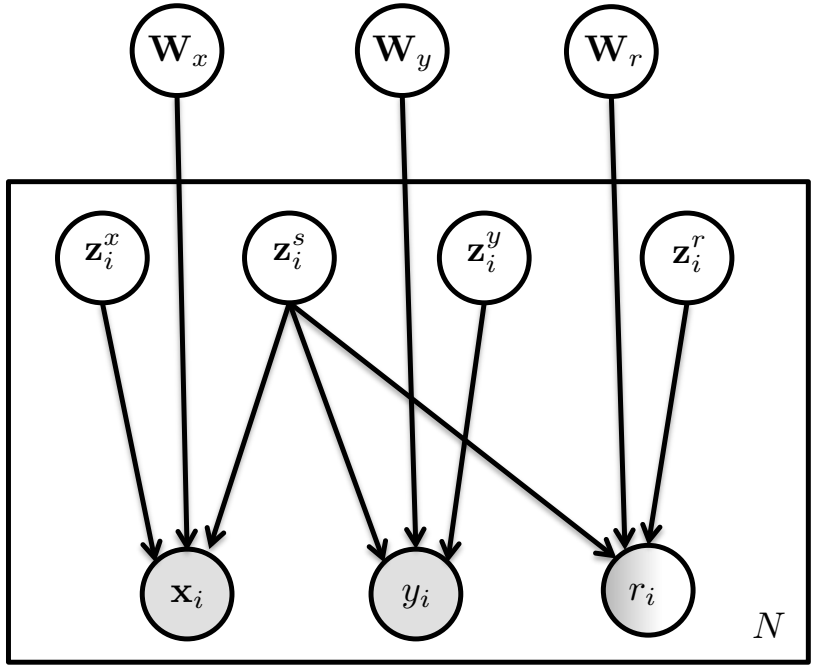
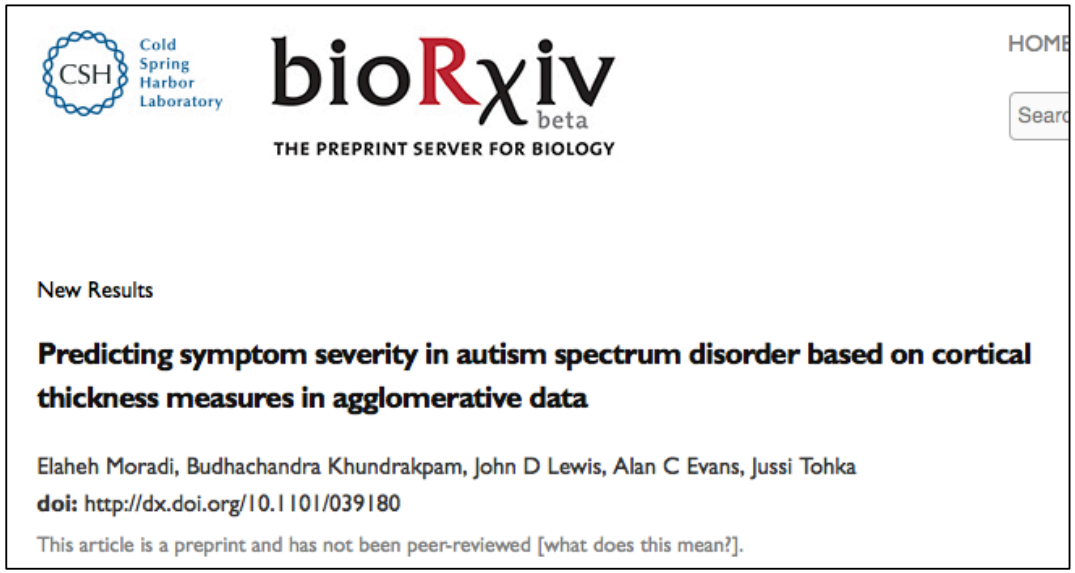


Image Site Indicator Target variable



Cold Spring Harbor Laboratory

bioRχiv beta

THE PREPRINT SERVER FOR BIOLOGY

HOME

Searc

New Results

Predicting symptom severity in autism spectrum disorder based on cortical thickness measures in agglomerative data

Elaheh Moradi, Budhachandra Khundrakpam, John D Lewis, Alan C Evans, Jussi Tohka

doi: http://dx.doi.org/10.1101/039180

This article is a preprint and has not been peer-reviewed [what does this mean?].

Index

- DA has been a hot topic in the past years.
- Hundreds of papers! (Much less for multi-source DA)
- Promising results for brain imaging with simple schemes.
- My first idea: formulate CCA model with Bayesian learning of the parameters.
- I already have a small data base, anyone wants to try something else?