

Label Shift Quantification

with Robustness Guarantees via Distribution Feature Matching

Bastien Dussap[†], Gilles Blanchard[†], Badr-Eddine Chérief-Abdellatif[§]

[†]Université Paris-Saclay, CNRS, Inria, Laboratoire de mathématiques d'Orsay

[§]CNRS, LPSM, Sorbonne Université, Université Paris Cité

September 19, 2023

université
PARIS-SACLAY

Inria



Introduction

Model

- \mathcal{X} : the data space.
- \mathcal{Y} : the label space, $\{1, \dots, c\}$.
- $\mathbb{P}_i = p(X|Y = i)$, conditional distribution.
- $\mathbb{P}_1, \dots, \mathbb{P}_c$: A list of c distributions, one for each class.

Label Shift

A "source" distribution.

$$\mathbb{P} = \sum_{i=1}^c \beta_i \mathbb{P}_i$$

A "target" distribution.

$$\mathbb{Q} = \sum_{i=1}^c \alpha_i^* \mathbb{P}_i$$

Training set.

$$\{(x_j, y_j)\}_{j \in [n]} \in (\mathcal{X} \times \mathcal{Y})^n$$

$$\hat{\mathbb{P}}_i := \frac{1}{n_i} \sum_{j \in [n]: y_j = i} \delta_{x_j}(\cdot)$$

Testing set.

$$\{x_{n+j}\}_{j \in [m]} \in \mathcal{X}^m$$

$$\hat{\mathbb{Q}} := \frac{1}{m} \sum_{j=1}^m \delta_{x_{n+j}}(\cdot)$$

→ The distributions differ only on the marginal Y .

Contaminated Label Shift

A "source" distribution.

$$\mathbb{P} = \sum_{i=1}^c \beta_i \mathbb{P}_i$$

A "target" distribution.

$$\mathbb{Q} = \sum_{i=1}^c \alpha_i^* \mathbb{P}_i + \alpha_0^* \mathbb{Q}_0$$

Training set.

$$\{(x_j, y_j)\}_{j \in [n]} \in (\mathcal{X} \times \mathcal{Y})^n$$

$$\hat{\mathbb{P}}_i := \frac{1}{n_i} \sum_{j \in [n]: y_j = i} \delta_{x_j}(\cdot)$$

Testing set.

$$\{x_{n+j}\}_{j \in [m]} \in \mathcal{X}^m$$

$$\hat{\mathbb{Q}} := \frac{1}{m} \sum_{j=1}^m \delta_{x_{n+j}}(\cdot)$$

→ \mathbb{Q}_0 is unknown.

Learning to Quantify

Goal: Quantification

Using $\{(x_j, y_j)\}_{j \in [n]} \in (\mathcal{X} \times \mathcal{Y})^n$ and $\{x_{n+j}\}_{j \in [m]} \in \mathcal{X}^m$, estimate α^* .



González, Castaño, Chawla, and Coz "A review on quantification learning". In *ACM Computing Surveys*, 2017.



Esuli, Fabris, Moreo and Sebastiani "Learning to Quantify". In *Springer Nature*, 2023

Learning to Quantify

Classify and Count (CC)

Use a classifier \hat{f} .

→ Count the number of times your classifier outputs each class.

$$\hat{\alpha}_{cc} = \left(\frac{1}{m} \sum_{j=1}^m 1_{\hat{f}(x_{n+j})=i} \right)_i = \hat{\mathbb{Q}}(\hat{f}(x) = i)$$

Problem

$$\alpha_{cc} = \mathbb{Q}(\hat{f}(x) = i) \neq \mathbb{Q}(y = i) = \alpha^*$$

Adjusted Classify and Count

Confusion matrix

$$\alpha_{cc} = M_{\hat{f}} \times \alpha^*$$

where M_f is the confusion matrix of \hat{f} .

Black-Box Shift Estimation (BBSE)

$$\hat{\alpha} = \hat{M}_{\hat{f}}^{-1} \hat{\alpha}_{cc}$$



Lipton, Wang, and Smola. "Detecting and correcting for label shift with black box predictor". In *ICML*, 2018.

Learning to Quantify

Label Shift ($\mathcal{L}\mathcal{S}$)

$$Q = \sum_{i=1}^c \alpha_i^* P_i$$

Distribution Matching

Using a Pseudo-Distance D :

$$\hat{\alpha} = \arg \min_{\alpha \in \Delta^c} D \left(\sum_{i=1}^c \alpha_i \hat{P}_i, \hat{Q} \right)$$

Where $\Delta^c := \{x \in \mathbb{R}_+^c : \sum x_i = 1\}$.

Embedding

→ Let $\phi : \mathcal{X} \rightarrow \mathcal{F}$ be a fixed feature mapping from \mathcal{X} into a Hilbert space \mathcal{F} (possibly $\mathcal{F} = \mathbb{R}^D$).

Embedding

$$\phi(\mathbb{P}) := \mathbb{E}_{X \sim \mathbb{P}}[\phi(X)] \in \mathcal{F}$$

→ ϕ is chosen so that $\phi(\mathbb{P})$ **characterizes** the distribution \mathbb{P} .

Pseudo-Distance

$$D_\phi(\mathbb{P}, \mathbb{Q}) = \|\phi(\mathbb{P}) - \phi(\mathbb{Q})\|_{\mathcal{F}}$$

Distribution Feature Matching

Label Shift (\mathcal{LS})

$$Q = \sum_{i=1}^c \alpha_i^* P_i$$

Distribution Feature Matching (DFM)

$$\hat{\alpha} = \arg \min_{\alpha \in \Delta^c} \left\| \sum_{i=1}^c \alpha_i \phi(\hat{P}_i) - \phi(\hat{Q}) \right\|_{\mathcal{F}} \quad (\mathcal{P})$$

where $\Delta^c := \{x \in \mathbb{R}_+^c : \sum x_i = 1\}$

Examples

Examples

- $\phi(x) = (1\{\hat{f}(x) = i\})_{i=1,\dots,c} \in \mathbb{R}^c$. For \hat{f} a hard-classifier.
- $\phi(x) = \hat{f}(x) \in \mathbb{R}^c$. For \hat{f} a soft-classifier.
- $\phi(x)$ of a neural network : $\hat{f}(x) = w^T \phi(x) + b \in \mathbb{R}^c$
- $\phi(x) = (y \mapsto k(x, y)) \in \mathcal{H}_k$



Lipton, Wang, and Smola. "Detecting and correcting for label shift with black box predictor". In *ICML*, 2018.



Iyer, Nath, and Sarawagi. "Maximum mean discrepancy for class ratio estimation: Convergence bounds and kernel selection". In *ICML*, 2014.



Kawakubo, Christoffel du Plessis and Sugiyama. "Computationally efficient class-prior estimation under class balance change using energy distance". In *IEICE Transactions on Information and Systems*, 2016.

Theoretical Analysis

Assumptions

$$\sum_{i=1}^c \beta_i \phi(\mathbb{P}_i) = 0 \iff \beta = 0 \quad (\mathcal{A}_1)$$

and

$$\exists C > 0 : \|\phi(x)\|_{\mathcal{F}} \leq C \text{ for all } x. \quad (\mathcal{A}_2)$$

Theoretical Analysis

Definition

$$\hat{\mathbf{G}}_{ij} = \langle \phi(\hat{\mathbb{P}}_i), \phi(\hat{\mathbb{P}}_j) \rangle$$

$$\hat{\mathbf{M}}_{ij} = \langle \phi(\hat{\mathbb{P}}_i) - \bar{\phi}, \phi(\hat{\mathbb{P}}_j) - \bar{\phi} \rangle$$

with $\bar{\phi} = c^{-1} \sum_{k=1}^c \phi(\hat{\mathbb{P}}_k)$.

λ_{\min} is the **smallest** eigenvalue of $\hat{\mathbf{G}}$.

Δ_{\min} is the **second smallest** eigenvalue of $\hat{\mathbf{M}}$.

In particular, it holds:

$$\Delta_{\min} \geq \lambda_{\min}.$$

Main Theorem under \mathcal{LS}

Theorem

For any $\delta \in (0, 1)$, with probability greater than $1 - \delta$:

$$\|\hat{\alpha} - \alpha^*\|_2 \lesssim \frac{C\sqrt{\log(c/\delta)}}{\sqrt{\Delta_{\min}}} \left(\frac{\|w\|_2}{\sqrt{n}} + \frac{1}{\sqrt{m}} \right) \quad (1)$$

$$\lesssim \frac{C\sqrt{\log(c/\delta)}}{\sqrt{\Delta_{\min}}} \left(\frac{1}{\sqrt{\min_i n_i}} + \frac{1}{\sqrt{m}} \right), \quad (2)$$

where $w_i = \frac{\alpha_i^*}{\tilde{\beta}_i}$, $\tilde{\beta}_i$ the empirical proportion of class i in the Source and n_i is the number of points of class i in the Source.

The same result holds when replacing α^* by the (unobserved) vector of empirical proportions $\tilde{\alpha}$ in the target sample, both on the left-hand side and in the definition of w .

soft-DFM for \mathcal{CLS}

A "target" distribution.

$$Q = \sum_{i=1}^c \alpha_i^* P_i + \alpha_0^* Q_0 \quad (\mathcal{CLS})$$

Soft-DFM

$$\hat{\alpha}_{\text{soft}} = \arg \min_{\alpha \in \text{int}(\Delta^c)} \left\| \sum_{i=1}^c \alpha_i \phi(\hat{P}_i) - \phi(\hat{Q}) \right\|_{\mathcal{F}}^2, \quad (\mathcal{P}_2)$$

where $\text{int}(\Delta^c) := \{x \in \mathbb{R}_+^c : \sum x_i \leq 1\}$

Main Theorems under \mathcal{CLS}

Theorem

If $\alpha_0^* = 0$, for any $\delta \in (0, 1)$, with probability greater than $1 - \delta$:

$$\begin{aligned}\|\hat{\alpha}_{\text{soft}} - \alpha^*\|_2 &\leq \frac{2CR_{c/\delta}}{\sqrt{\lambda_{\min}}} \left(\frac{\|w\|_2}{\sqrt{n}} + \frac{1}{\sqrt{m}} \right) \\ &\leq \frac{2CR_{c/\delta}}{\sqrt{\lambda_{\min}}} \left(\frac{1}{\sqrt{\min_j n_j}} + \frac{1}{\sqrt{m}} \right),\end{aligned}$$

Gram Matrix

λ_{\min} is the **smallest** eigenvalue of $\hat{\mathbf{G}}$ and Δ_{\min} the **second smallest** eigenvalue of $\hat{\mathbf{M}}$. In particular, it holds:

$$\Delta_{\min} \geq \lambda_{\min}.$$

Main Theorems under \mathcal{CLS}

Definition

Introduce $\bar{V} := \text{Span}\{\phi(\mathbb{P}_i), i \in [c]\}$ and let $\Pi_{\bar{V}}$ be the orthogonal projection on \bar{V} .

Theorem

With probability greater than $1 - \delta$:

$$\|\hat{\alpha}_{\text{soft}} - \alpha^*\|_2 \lesssim \frac{1}{\sqrt{\lambda_{\min}}} \left(3\epsilon_n + \epsilon_m + \sqrt{2\alpha_0^* \epsilon_n \|\phi(\mathbb{Q}_0)\|} + \|\Pi_{\bar{V}}(\phi(\mathbb{Q}_0))\|_{\mathcal{F}} \right),$$

with:

$$\epsilon_n = \sqrt{\frac{\log(c/\delta)}{\min_i n_i}}; \quad \epsilon_m = \sqrt{\frac{\log(c/\delta)}{m}};$$

Robustness to contamination

$$\|\hat{\alpha}_{\text{soft}} - \alpha^*\|_2 \lesssim \frac{1}{\sqrt{\lambda_{\min}}} \left(3\epsilon_n + \epsilon_m + \sqrt{2\alpha_0^* \epsilon_n \|\phi(\mathbb{Q}_0)\|} + \|\Pi_{\bar{V}}(\phi(\mathbb{Q}_0))\|_{\mathcal{F}} \right)$$

Observe that the bound shows the **robustness** of **soft**-DFM procedure against contaminations \mathbb{Q}_0 that are **orthogonal** to $\bar{V} := \text{Span}\{\phi(\mathbb{P}_i), i \in [c]\}$.

Gaussian kernel (KMM)

Two embeddings $\phi(\mathbb{P})$ and $\phi(\mathbb{P}')$ will be close to **orthogonal** if \mathbb{P} and \mathbb{P}' are well-**separated**.

We anticipate that **KMM** will be **robust** against contamination distributions \mathbb{Q}_0 whose main mass is **far away** from the source distributions.

Robustness to contamination

$$\|\hat{\alpha}_{\text{soft}} - \alpha^*\|_2 \lesssim \frac{1}{\sqrt{\lambda_{\min}}} \left(3\epsilon_n + \epsilon_m + \sqrt{2\alpha_0^* \epsilon_n \|\phi(\mathbb{Q}_0)\|} + \|\Pi_{\bar{V}}(\phi(\mathbb{Q}_0))\|_{\mathcal{F}} \right)$$

Observe that the bound shows the **robustness** of a **soft**-DFM procedure against contaminations \mathbb{Q}_0 that are **orthogonal** to $\bar{V} := \text{Span}\{\phi(\mathbb{P}_i), i \in [c]\}$.

Classifier (BBSE)

The feature space is of the same dimension as the number of sources hence the **orthogonal component** will always be 0 and we expect **no robustness** property for BBSE.

Experiments

The source is a list of 5 Gaussian distributions. α_0^* ranges from from 0 to 0.3. We repeated the experiments with different dimensions.

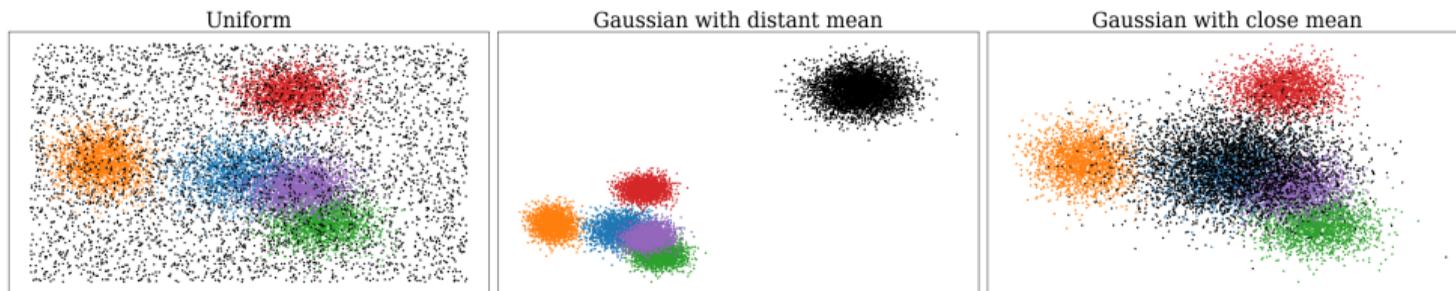


Figure: Three kinds of noise : On the left Q_0 is uniformly distributed over the data range, in the middle Q_0 is Gaussian with a mean distant from the other means and on the right Q_0 is Gaussian with a similar mean to the others.

Experiments

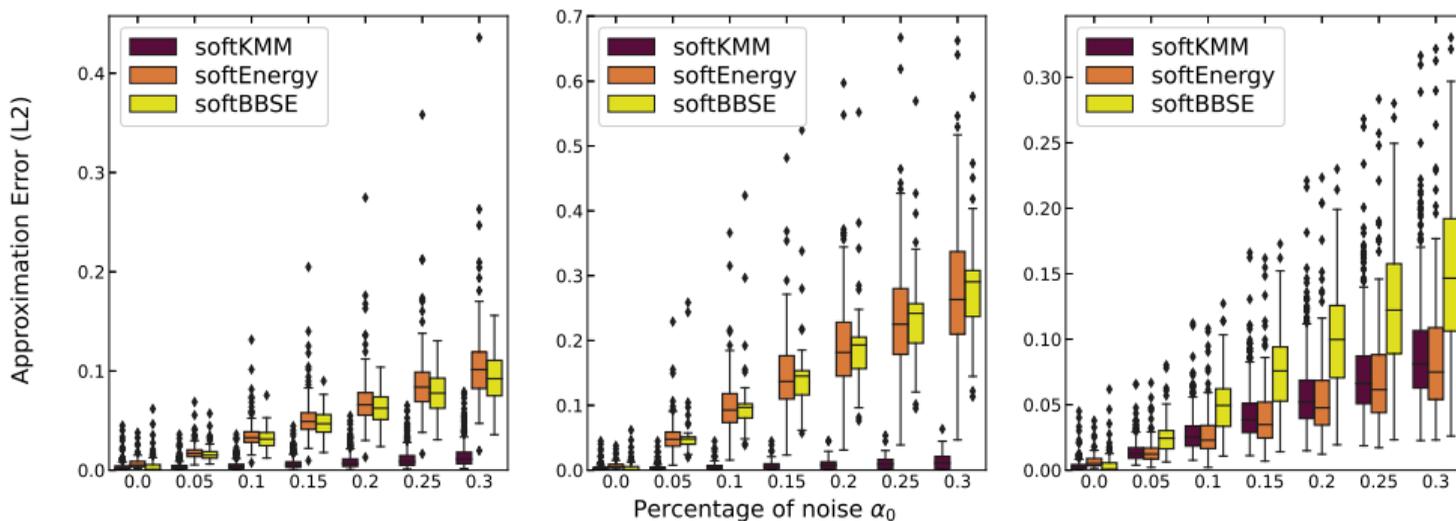


Figure: Robustness of the algorithms to three types of noise. Left: uniform noise; middle: noise is a new class far from the others; right: noise is a new class in the middle of the others.

Conclusion

- We introduced a **general approach** for Label Shift Quantification.
- We provided a **general theoretical analysis** of DFM, **improving** over previously known bounds derived for specific instantiations only.
- We analysed theoretically the behavior of DFM under departures from the label shift hypothesis, a situation not studied in earlier works, and put into light a **robustness** against certain types of perturbations, depending on the feature mapping ϕ used.

Thank you for your attention.



Dussap, Bastien and Blanchard, Gilles and Chérief-Abdellatif, Badr-Eddine.
*"Label Shift Quantification with Robustness Guarantees via Distribution
Feature Matching"*.

Kernel Mean Matching

Kernel Mean Embedding

$$\phi(\mathbb{P}) = \mathbb{E}_{\mathbb{P}}[k(x, \cdot)]$$

Kernel Trick

$$\langle \phi(\mathbb{P}), \phi(\mathbb{Q}) \rangle = \mathbb{E}_{\mathbb{P}, \mathbb{P}}[k(x, x')] + \mathbb{E}_{\mathbb{Q}, \mathbb{Q}}[k(y, y')] - 2\mathbb{E}_{\mathbb{P}, \mathbb{Q}}[k(x, y)]$$

→ Methods using Kernels are **quadratic**.

Random Fourier Features

Random Fourier Features are based on Bochner's Theorem:

Theorem

A continuous function φ on \mathbb{R}^D is positive definite if and only if φ is the Fourier transform of a non-negative measure.

Random Fourier Features

$$\begin{aligned}k(x, y) &= \mathbb{E}_{\omega \sim \Lambda_k} [e^{i\omega^T(x-y)}] \\ &= \mathbb{E}_{\omega \sim \Lambda_k} [\cos(\omega^T(x-y))]\end{aligned}$$

Random Fourier Features

Using a sample $(\omega_i)_{i=1}^{D/2}$ i.i.d. from Λ_k :

$$\phi(x) = \sqrt{\frac{2}{D}} \left[\cos(\omega_i^T x), \sin(\omega_i^T x) \right]_{i=1}^{D/2}$$

is such that

$$k(x, y) = \mathbb{E}[\phi(x)^T \phi(y)],$$

Random Fourier Feature Matching

Random Fourier Feature Matching (RFFM)

DFM method using:

$$\phi(x) = \sqrt{\frac{2}{D}} \left[\cos(\omega_i^T x), \sin(\omega_i^T x) \right]_{i=1}^{D/2}$$

Relying on RFF with D Fourier features induces a complexity of $O(D(n+m))$ since we only have to compute $\phi(\hat{P}_i)$ and $\phi(\hat{Q})$.

Computing $\phi(\hat{P})$ reduces to a **matrix multiplication**, for which **GPU** are well suited.



Rahimi and Recht. "Random features for large-scale kernel machines". In *Advances in neural information processing systems*, 2007.

Optimisation Problem

Optimisation Problem (DFM)

$$\hat{\alpha} = \arg \min_{\alpha \in \Delta^c} \left\| \sum_{i=1}^c \alpha_i \phi(\hat{\mathbb{P}}_i) - \phi(\hat{\mathbb{Q}}) \right\|_{\mathcal{F}}^2 \quad (\mathcal{P})$$

where $\Delta^c := \{x \in \mathbb{R}_+^c : \sum x_i = 1\}$

Solving (\mathcal{P}) amounts to solving a **Quadratic Programming** (QP) in dimension c . Indeed, we can rewrite the problem as:

$$\begin{aligned} & \text{minimise } \frac{1}{2} \alpha^T \hat{\mathbf{G}} \alpha + q^T \alpha && (\text{QP}) \\ & \text{subject to } \alpha \succeq 0_c \text{ and } \mathbf{1}_c^T \alpha = 1, \end{aligned}$$

with $q = \left(\langle \phi(\hat{\mathbb{P}}_i), \phi(\hat{\mathbb{Q}}) \rangle \right)_{i=1}^c$. This is a c -dimensional QP problem, which can be solved efficiently.

BBSE as DFM

Examples

- $\phi(\mathbf{x}) = (1\{\hat{f}(x) = i\})_{i=1,\dots,c} \in \mathbb{R}^c$. For \hat{f} a hard-classifier.
- $\phi(\hat{\mathbb{Q}}) = \hat{\alpha}_{cc}$
- $\phi(\hat{\mathbb{P}}_i) = \hat{\mathbb{P}}(\hat{f}(x) = j \mid y = i)_j = (\hat{M}_{\hat{f}})_i$

BBSE

$$\hat{\alpha}_- = \hat{M}_{\hat{f}}^{-1} \hat{\alpha}_{cc}$$

DFM associated to ϕ

$$\hat{\alpha}_+ = \arg \min_{\alpha \in \Delta^c} \|\hat{\alpha}_{cc} - \hat{M}_{\hat{f}} \alpha\|_2$$

Proposition

If $\hat{\alpha}_- \geq 0$ then:

$$\hat{\alpha}_- = \hat{\alpha}_+$$

Properties of Δ_{\min}

Let (b_1, \dots, b_c) be a c -uple of vectors of \mathbb{R}^D assumed to be linearly independent. We denote M the **Gram matrix** of those vectors, i.e.

$$M_{ij} = \langle b_i, b_j \rangle.$$

Theorem

For any number of classes c , Δ_{\min} is equal to $\min_{\substack{\|u\|=1 \\ \mathbf{1}^T u=0}} u^T M u$.

→ $\Delta_{\min}(b_1, \dots, b_c)$ is **always greater** than the smallest eigenvalue of the Gram matrix.

Theorem

In particular for two classes, $\Delta_{\min}(b_1, b_2) = \frac{1}{2} \|b_1 - b_2\|^2$.

Experiments on Cytometry dataset

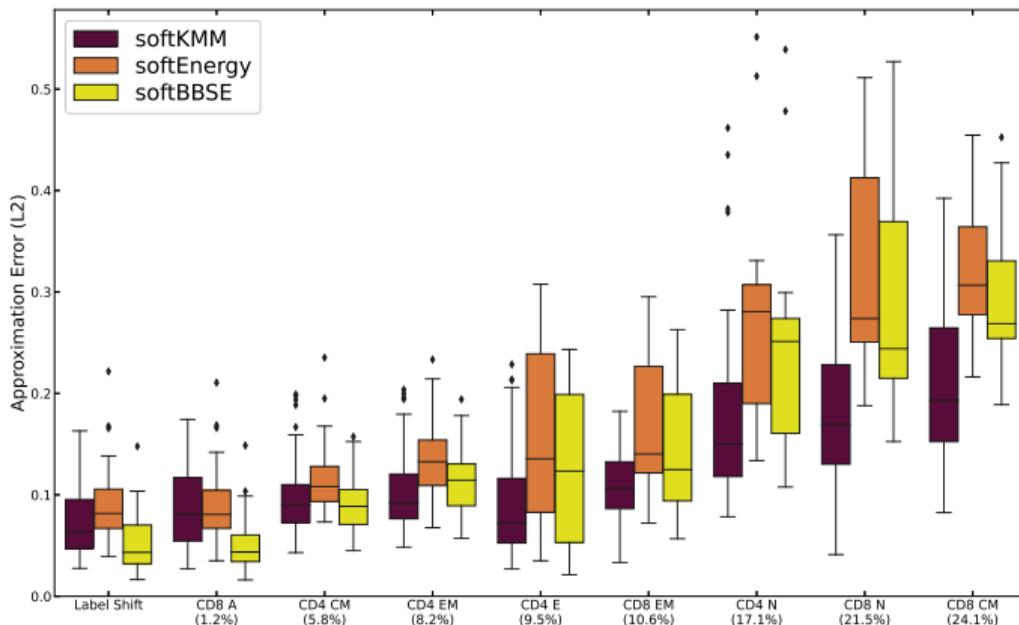


Figure: Each column represents the error — computed using the ℓ_2 norm between the true proportions and the estimated proportions — obtained when some class is absent from the source but present in the target distribution. The first column gives the results when no class is discarded. The class are sorted according to the average proportions they represent in the samples (x labels mention class held out from the source and its proportion)