# Estimation of proportions

## under Open set Label Shift using Mahalanobis Projection

Dussap Bastien

Laboratoire de mathématiques d'Orsay
Université Paris-Saclay, Inria

Tuesday 28th May, 2024

# Introduction

## Model

- $\mathcal{X}$ : the data space, in our case $\mathbb{R}^d$.
- $\mathcal{Y}$ : the label space, $\{1, \cdots, c\}$.
- $c$ : the number of classes.
- $\mathbb{P}_1, \cdots, \mathbb{P}_c$ : A list of $c$ distributions, one for each class (sources).
- $\mathcal{N}$ : the *noise*.

# Open Set Label Shift

**Training sets.**

$$\left(x_1^i, \cdots x_{n_i}^i\right) \sim \mathbb{P}_i$$

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

$$n = \sum n_i.$$

**A "target" distribution.**

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

**Testing set.**

$$\left(x_{n+1}, \cdots, x_{n+m}\right) \sim \mathbb{Q}$$

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

# Estimation of proportions

**Goal: Quantification**

Using the training sets : $\hat{\mathbb{P}}_1 \cdots, \hat{\mathbb{P}}_c$ estimate the proportions $\alpha^*$ in the testing set.


- 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
- Dussap, Bastien and Blanchard, Gilles and Chérief-Abdellatif, Badr-Eddine *"Label Shift Quantification with Robustness Guarantees via Distribution Feature Matching"*. In *ECML/PKDD*, 2023

# Vectorisation

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

## Embedding

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

# Kernel

## Classical kernel

- $k(x,y) = x^T y$, linear.
- $k(x,y) = (\gamma x^T y + c_0)^d$, polynomial.
- $k(x,y) = \tanh(\gamma x^T y + c_0)$, sigmoid.
- $k(x,y) = \exp(-\gamma \|x-y\|_2^2)$, gaussian.
- $k(x,y) = \exp(-\gamma \|x-y\|_1)$, laplacian.
- $k(x,y) = \|x\| + \|y\| - \|x-y\|$, energy.
- $k(x,y) = \left(1 + \frac{\|x-y\|^2}{\sigma^2}\right)^{-1}$, cauchy.

# Kernel Mean Embedding

## Kernel Methods

For a positive-definite kernel $k$ there exists a functional Hilbert space $\mathcal{H}_k$ and an embedding $\phi_k\colon \mathcal{X} \mapsto \mathcal{H}_k$ such that:

$$k(x, y) = \langle \phi_k(x), \phi_k(y) \rangle_{\mathcal{H}_k}$$

## Embedding

$$\phi_k(x) := k(x, \cdot) = (y \mapsto k(x, y)) \in \mathcal{H}_k$$

# Kernel Mean Embedding

**Kernel Mean Embedding**

$$\begin{aligned}\Phi_k : \mathcal{M}_1^+(\mathcal{X}) &\to \mathcal{H}_k \\ \mathbb{P} &\mapsto \mathbb{E}_{X\sim\mathbb{P}}[\phi_k(X)] = \Phi_k(\mathbb{P})\end{aligned}$$

$\longrightarrow$ If $\Phi_k$ is injective we say that the kernel $k$ is characteristic.

# Random Fourier Features

## Random Fourier Features

$$
\begin{aligned}
z \colon \mathcal{X} &\to \mathbb{R}^D \\
x &\mapsto z(x),
\end{aligned}
$$

such that :

$$k(x, y) \approx z(x)^T z(y)$$

## Random Fourier Features

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

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

# Random Fourier Feature Matching

**Complexity**

Relying on RFF with $D$ Fourier features induces a complexity of $O(D(n+m))$ instead of $O((n+m)^2)$.
Computing $z_\omega(\hat{\mathbb{P}})$ reduces to a matrix multiplication, for which GPU are well suited.

# Methods

**Method if $\alpha_0^* = 0$**

$$\hat{\alpha} = \underset{\alpha \in \Delta^c}{\arg\min} \left\| \sum_{i=1}^{c} \alpha_i \Phi(\hat{\mathbb{P}}_i) - \Phi(\hat{\mathbb{Q}}) \right\|_{\mathcal{H}},$$

where $\Delta^c = \left\{ x \in \mathbb{R}_+^c : \sum_{i=1}^c x_i = 1 \right\}$.

**Method if $\alpha_0^* > 0$**

$$\hat{\alpha} = \underset{\alpha \in \mathrm{int}(\Delta^c)}{\arg\min} \left\| \sum_{i=1}^{c} \alpha_i \Phi(\hat{\mathbb{P}}_i) - \Phi(\hat{\mathbb{Q}}) \right\|_{\mathcal{H}},$$

where $\mathrm{int}(\Delta^c) = \left\{ x \in \mathbb{R}_+^c : \sum_{i=1}^c x_i \leq 1 \right\}$.

# Maximum Mean Discrepancy

**Maximum Mean Discrepancy**

$$\begin{aligned}\mathrm{MMD}^2(\mathbb{P},\mathbb{Q}) &= \|\Phi_k(\mathbb{P}) - \Phi_k(\mathbb{Q})\|^2_{\mathcal{H}_k} \\ &= \mathbb{E}_{\mathbb{P},\mathbb{P}}[k(X,X)] + \mathbb{E}_{\mathbb{Q},\mathbb{Q}}[k(Y,Y)] - 2\mathbb{E}_{\mathbb{P},\mathbb{Q}}[k(X,Y)]\end{aligned}$$

Gretton, Arthur and Borgwardt, Karsten and Rasch, Malte and Schölkopf, Bernhard and Smola, Alex *"A kernel method for the two-sample problem"*. In *Advances in neural information processing systems*, 2006.

# Goal

**Theorem**

*For any $\delta$, with probability greater than $1 - \delta$:*

$$\|\hat{\alpha} - \alpha^*\| \leq B_\delta(n, m) \to 0,$$

*where $\alpha^*$ are the proportions in the target.*

# Theoretical guarantees

**Theorem**

Under mild condition, if $\alpha_0^* = 0$, then with high probability:

$$\|\hat{\alpha} - \alpha^*\|_2 \lesssim \Delta_{\min}{}^{-1/2}\left(\frac{1}{\sqrt{\min_i n_i}} + \frac{1}{\sqrt{m}}\right),$$

where $n_i$ is the number of points in the $i$-th training set, $m$ is the number of points in the testing set, and $\Delta_{\min}$ is the second smallest eigenvalue of the centred gram matrix of the training sets embedding $(\Phi_k(\hat{\mathbb{P}}_i))$.

# Theoretical guarantees

**Theorem**

Under mild condition, if $\alpha_0^* \geq 0$, then with high probability:

$$\|\hat{\alpha} - \alpha^*\|_2 \lesssim \Delta_{\min}{}^{-1/2} \left( \left\|\Phi(\mathcal{N})\right\| \min_i n_i{}^{-1/4} + \left\|\Pi_V(\Phi(\mathcal{N}))\right\| \right)$$

where $n_i$ is the number of points in the $i$-th training set and $\Pi$ is the orthogonal projector on $V = \mathrm{Span}\left\{\Phi(\hat{\mathbb{P}}_i)\right\}$.

# Variance-aware methods

**Variance-aware methods**

$$\hat{\alpha} = \operatorname*{arg\,min}_{\alpha \in \mathrm{int}(\Delta^c)} \left\| M \left( \sum_{i=1}^{c} \alpha_i \Phi(\hat{\mathbb{P}}_i) - \Phi(\hat{\mathbb{Q}}) \right) \right\|_{\mathcal{H}}$$

Where $M$ is a linear operator, or matrix, on $\mathcal{H} \mapsto \mathcal{H}$ :

$$M := M\Big(\Sigma_1, \cdots, \Sigma_c\Big),$$

with $\Sigma_i := \Sigma_{\Phi(\mathbb{P}_i)}$.

# Main Theorem

**Theorem**

Under mild condition, if $\alpha_0^* = 0$, then with high probability:

$$\|\hat{\alpha} - \alpha^*\| \lesssim \mathcal{O}\left(\min_i \frac{1}{n_i} + \frac{1}{m}\right) \quad (1)$$

$$+ \sqrt{\frac{Tr(M\Sigma_{\alpha^*}M^\top)}{\lambda_{\min}(\hat{\boldsymbol{G}}^M)}}\left(\min_i \frac{1}{\sqrt{n_i}} + \frac{1}{\sqrt{m}}\right), \quad (2)$$

with $\Sigma_{\alpha^*} = \sum_{i=1}^{c} \alpha^* \Sigma_i$ and $\Sigma_i$ is the covariance matrix of $\Phi(\mathbb{P}_i)$.

# Theoretical guarantees

**Theorem**

*For any given feature map $\Phi$ that verify mild conditions, the matrix that minimise the criterion*

$$\frac{\mathrm{Tr}(M\Sigma_{\alpha^*}M^\top)}{\lambda_{min}(\hat{\boldsymbol{G}}^M)}, \tag{3}$$

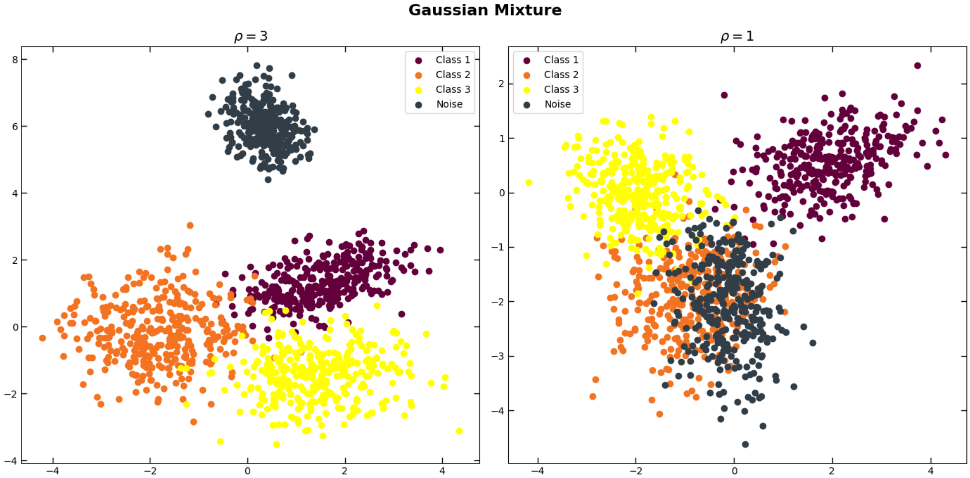
*is:*

$$M^\top M = \Sigma_{\alpha^*}^{-1/2}\left(\Sigma_{\alpha^*}^{-1/2}\hat{V}\hat{V}^\top\Sigma_{\alpha^*}^{-1/2}\right)^+\Sigma_{\alpha^*}^{-1/2}, \tag{$\mathcal{M}$}$$

*and the value of the criterion is then equals to*

$$\mathrm{Tr}\left((\Sigma_{\alpha^*}^{-1/2}\hat{V}\hat{V}^\top\Sigma_{\alpha^*}^{-1/2})^+\right). \tag{4}$$

# Data


Gaussian Mixture
$\rho = 3$
$\rho = 1$
Class 1
Class 2
Class 3
Noise
Class 1
Class 2
Class 3
Noise

# Results

| **Percentage of noise $\epsilon$** | **Embedding** | Number of classes = 5 | | |
|---|---|---|---|---|
| | | dim = 2 | dim = 5 | dim = 10 |
| 0.0 | Classifier | 4.11 ; 3.0 | 1.23 ; 3.0 | 0.97 ; 3.0 |
| 0.0 | KME | 1.75 ; 2.0 | 0.82 ; 2.0 | 0.84 ; 2.0 |
| 0.0 | VA-KME | **1.55 ; 1.0** | **0.71 ; 1.0** | **0.71 ; 1.0** |
| 0.2 | Classifier | 32.88 ; 3.0 | 23.96 ; 3.0 | 21.67 ; 3.0 |
| 0.2 | KME | **3.17 ; 1.5** | **1.85 ; 1.0** | **1.87 ; 1.0** |
| 0.2 | VA-KME | **3.27 ; 1.5** | **1.92 ; 2.0** | **2.03 ; 2.0** |
| 0.5 | Classifier | 66.21 ; 3.0 | 60.31 ; 3.0 | 55.56 ; 3.0 |
| 0.5 | KME | **6.00 ; 1.0** | **4.42 ; 1.0** | **4.78 ; 1.0** |
| 0.5 | VA-KME | **6.56 ; 2.0** | **4.50 ; 2.0** | **4.89 ; 2.0** |
| 0.7 | Classifier | 82.88 ; 3.0 | 77.70 ; 3.0 | 75.12 ; 3.0 |
| 0.7 | KME | **6.74 ; 1.0** | **5.64 ; 1.0** | **6.88 ; 1.0** |
| 0.7 | VA-KME | **7.03 ; 2.0** | 5.94 ; 2.0 | **6.94 ; 2.0** |

# Results

| **Percentage of noise $\epsilon$** | **Quantifier** | Number of classes = 5 | | |
|---|---|---|---|---|
| | | dim = 2 | dim = 5 | dim = 10 |
| 0.0 | Classifier | 4.11 ; 3.0 | 1.23 ; 3.0 | 0.97 ; 3.0 |
| 0.0 | KME | 1.75 ; 2.0 | 0.82 ; 2.0 | 0.84 ; 2.0 |
| 0.0 | VA-KME | **1.55 ; 1.0** | **0.71 ; 1.0** | **0.71 ; 1.0** |
| 0.2 | Classifier | 26.90 ; 3.0 | 15.43 ; 3.0 | 12.42 ; 2.5 |
| 0.2 | KME | **16.20 ; 2.0** | 12.76 ; 2.0 | 12.22 ; 2.5 |
| 0.2 | VA-KME | **17.38 ; 2.0** | **11.69 ; 1.0** | **10.94 ; 1.0** |
| 0.5 | Classifier | 52.43 ; 3.0 | 39.42 ; 3.0 | 31.95 ; 3.0 |
| 0.5 | KME | **30.70 ; 1.0** | 31.65 ; 2.0 | 30.51 ; 2.0 |
| 0.5 | VA-KME | 33.98 ; 2.0 | **29.88 ; 1.0** | **27.88 ; 1.0** |
| 0.7 | Classifier | 67.25 ; 3.0 | 52.79 ; 3.0 | 44.55 ; 3.0 |
| 0.7 | KME | **45.76 ; 1.0** | 44.09 ; 2.0 | 41.21 ; 2.0 |
| 0.7 | VA-KME | 52.71 ; 2.0 | **42.76 ; 1.0** | **39.50 ; 1.0** |

# Experiments

| **Embedding** | Optimal | $\Sigma_{\alpha^*}^{-1/2}$ | Identity |
|---|---|---|---|
| Classifier | **0.17 ; 2.0** | **0.17 ; 2.5** | **0.17 ; 1.5** |
| KME | **0.15 ; 1.5** | **0.15 ; 1.5** | 0.20 ; 3.0 |
| Classifier + KME | **0.14 ; 1.5** | **0.14 ; 1.5** | 0.16 ; 3.0 |
| Mean | **0.33 ; 1.5** | **0.33 ; 1.75** | 0.36 ; 3.0 |

**Thank you for your attention.**