Large-Margin Metric Learning for Constrained Partitioning Problems

Rémi Lajugie REMI.LAJUGIE@ENS.FR
Sylvain Arlot SYLVAIN.ARLOT@ENS.FR
Francis Bach FRANCIS.BACH@INRIA.FR
Département d'Informatique de l'Ecole Normale Supérieure, (CNRS/INRIA/ENS), Paris, France

Abstract

We consider unsupervised partitioning problems based explicitly or implicitly on the minimization of Euclidean distortions, such as clustering, image or video segmentation, and other change-point detection problems. We emphasize on cases with specific structure, which include many practical situations ranging from mean-based change-point detection to image segmentation problems. We aim at learning a Mahalanobis metric for these unsupervised problems, leading to feature weighting and/or selection. This is done in a supervised way by assuming the availability of several (partially) labeled datasets that share the same metric. We cast the metric learning problem as a large-margin structured prediction problem, with proper definition of regularizers and losses, leading to a convex optimization problem which can be solved efficiently. Our experiments show how learning the metric can significantly improve performance on bioinformatics, video or image segmentation problems.

1. Introduction

Unsupervised partitioning problems are ubiquitous in machine learning and other data-oriented fields such as computer vision, bioinformatics or signal processing. They include (a) traditional *unsupervised clustering* problems, with the classical K-means algorithm, hierarchical linkage methods (Gower & Ross, 1969) and spectral clustering (Ng et al., 2002), (b) *unsupervised image segmentation* problems where two neighboring pixels are encouraged to be in the same cluster, with mean-shift techniques (Cheng, 1995) or normalized cuts (Shi & Malik, 1997), and (c) *change-point detection* problems adapted to multivariate sequences (such as video) where segments are composed of contiguous elements, with typical window-based algorithms (Desobry et al., 2005) and various methods looking for a change in the mean of some features (Chen & Gupta, 2011).

All the algorithms mentioned above rely on a specific distance (or more generally a similarity measure) on the space of configurations and a good metric is crucial to their performance, especially in high-dimensional settings where many dimensions may be irrelevant to the partitioning task. While the choice of such a metric has originally been tackled manually (often by trial and error), recent work has considered learning such metric directly from data. Without any supervision, the problem is ill-posed and methods based on generative models may learn a metric or reduce dimensionality (see, e.g., De la Torre & Kanade 2006), but typically with no guarantees that they lead to better partitions. In this paper, we consider the same goal as Bar-Hillel et al. (2006), Xing et al. (2002), Bach & Jordan (2003) and Finley & Joachims (2008), that is learning a metric for one or several partitioning problems sharing a common metric, assuming that one or several fully (or partially) labeled partitioned datasets are available during the learning phase. While such labeled datasets are typically expensive to produce, there are several scenarios where these datasets have already been built, often for evaluation purposes. These occur in video segmentation tasks (see Section 5.3), image segmentation tasks (Section 5.5) as well as change-point detection tasks in bioinformatics (see Hocking et al. 2013 and Section 4.2). This global framework is sometimes referred to as *supervised clustering* (Finley & Joachims, 2005; 2008).

Related work. The need for metric learning goes far beyond unsupervised partitioning problems. Weinberger et al. (2006) proposed a large-margin framework for learning a metric in nearest-neighbours algorithms based on sets of must-link/must-not-link constraints, while Goldberger et al. (2004) considered a probability-based non-convex formulation. For these frameworks, a single dataset is fully labeled and the goal is to learn a metric leading to good testing performance on unseen data. Metric learning has also been considered in semi-supervised clustering of a single dataset, where some partial constraints are given. This in-

Proceedings of the 31^{st} International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32.

cludes the works of Bar-Hillel et al. (2006) and Xing et al. (2002). As shown in Section 5, these can be used in our setting as well by stacking several datasets into a single one. However, our discriminative large-margin approach outperforms these, because we consider explicitly the clustering performance, for each dataset, through a structured large-margin approach. Moreover, these approaches cannot readily use additional prior knowledge on the partitions.

The task of learning how to partition has been tackled by Bach & Jordan (2003) for spectral clustering. The problem set-up is the same (availability of several fully partitioned datasets), however, their formulation is non-convex and relies on the unstable optimization of eigenvectors.

Finley & Joachims (2005) considered the same convex large-margin set-up as ours but for correlation clustering, a clustering method based on greedy algorithms or convex relaxations. Finley & Joachims (2008) instead considered distortion-based clustering methods that can be applied to large-scale problems, in particular at test time. The latter approach can be seen as special case of our work, with an a priori known number of clusters, approximate decoding and does no structured priors on partitions.

Other approaches do not require any supervision (De la Torre & Kanade, 2006), and perform simultaneous dimensionality reduction and clustering, by alternating between the computation of a low-rank matrix and clustering of the data using the corresponding metric. However, they cannot take advantage of the labeled information that we use.

Our approach can also be related to the work of Szummer et al. (2008): given a small set of labeled instances, they use a similar large-margin framework, inspired by Tsochantaridis et al. (2005) and Taskar et al. (2003), to learn parameters of Markov random fields, using graph cuts for solving the "loss-augmented inference problem" of structured prediction. However, their segmentation framework does not apply to unsupervised segmentation. In this paper, we present a supervised learning framework aiming at learning how to perform an unsupervised task.

Structured SVM have been used to solve other learning problems, for instance to learn weights for graph matching (Caetano et al., 2007) or a metric for ranking tasks (Mcfee & Lanckriet, 2010). In computer vision, it has also been used to build task-driven image representations (Kim et al., 2012).

Beyond existing approaches. The existing approaches for learning the metric in a supervised way have two main drawbacks: (1) they are unable to deal clearly with the common case in which the number of clusters in the data is unknown a priori except in the case of Finley & Joachims (2005) for which unknown number of cluster is indirectly partly taken into account, (2) they do not incorporate any prior knowledge on partititions, which is a significant limitation because in most applications, extra prior information—hard or soft—may be used to make the clustering problem less ill-defined.

Dealing with unknown number of clusters. None of the aforementioned methods is suited for learning a penalty term for selecting the number of clusters, as they do not include any model selection term. However, the scenario where the number of clusters is unknown is in practice very common. For instance in bioinformatics for a-CGH segmentation (Hocking et al., 2013), it is unrealistic to assume to know a priori in how many segments a sequence should be split. The same remark holds for the segmentation of long videos: at test time, it is not realistic to assume the number of segments is known. We explore these applications in Sections 5.3 and 5.4.

Hard priors. A common prior is the sequential structure that can be found everywhere in signal processing, from audio (Gillet et al., 2007) to bioinformatics with a-CGH or EEG segmentation (Hocking et al., 2013; Brodsky & Darkhovsky, 1993). Our work focuses on hard-coding such a prior by restricting the set of authorized partitions. This leads to the well-known change-point detection problem. In particular, we show that in that case and using a similar structured SVM as in Finley & Joachims (2008), the loss-augmented inference problem can now be solved *exactly* in polynomial time using a dynamic programming method.

Soft priors. A popular application of clustering is image segmentation. Unfortunately, simple K-means-based algorithms do not take into account the two-dimensional structure and do not lead to meaningful partitions. In this paper, we consider adding a Laplacian-based penalty term that take into account the spatial structure, thus proving an alternative to normalized cuts, for which the metric can be learned efficiently.

Summary. We make the following contributions:

- We propose an efficient algorithm that learns the metric for some partitioning problem, given several labeled datasets sharing the same metric. Our algorithm chooses automatically the number of clusters at test time, and can deal with hard or soft priors about the partitions. Experiments in Section 5 show that our algorithm can significantly improve the performance compared to previous works that can be used in our setting, on synthetic examples as well as for video and image segmentation tasks.
- When imposing that segments should be contiguous (change-point detection) we propose a dynamic programming algorithm that can solve the loss-augmented inference problem in polynomial time (Algorithm 1).
- We propose an extension of our algorithm that can learn a metric from *partially labeled* datasets (Section 4.1).

– We propose in Section 4.2 a way to detect changes in the full distribution of univariate time series, rather than only in the mean, with application to bioinformatics.

2. Partitioning through matrix factorization

In this section, we consider T multi-dimensional observations $x_1, \dots, x_T \in \mathbb{R}^p$, which may be represented in a design matrix $X \in \mathbb{R}^{T \times p}$. Partitioning the T observations into K classes is equivalent to finding an *assignment matrix* $Y \in \{0,1\}^{T \times K}$, such that $Y_{ij} = 1$ if the i-th observation is assigned to cluster j and 0 otherwise. For general partitioning problems, no additional constraints are used, but for change-point detection problems, it is assumed that the segments are contiguous and with increasing labels. That is, the matrix Y is block-diagonal with each block equal to $\mathbf{1}_{T_j}$, where $\mathbf{1}_D \in \mathbb{R}^D$ is the D-dimensional vector with constant components equal to one, and T_j is the number of elements in cluster j. For any partition, we may re-order (non uniquely) the data points so that the assignment matrix has the same form; this is typically useful for the understanding of partitioning problems.

2.1. Distortion measure

In this paper, we consider partitioning models where each datapoint in cluster j is modeled by a vector (often called a centroid or a mean) $c_j \in \mathbb{R}^p$, the overall goal being to find a partition and a set of means so that the distortion measure $\sum_{i=1}^T \sum_{j=1}^K Y_{ij} \|x_i - c_j\|^2$ is as small as possible, where $\| \cdot \|$ is the Euclidean norm in $\mathbb{R}^p$. By considering the Frobenius norm defined through $\|A\|_F^2 = \mathrm{Tr}(AA^\top) = \sum_{i=1}^T \sum_{j=1}^P A_{ij}^2$, this is equivalent to minimizing $\|X - YC\|_F^2$ with respect to an assignment matrix Y and a centroid matrix $C \in \mathbb{R}^{K \times p}$.

2.2. Representing partitions

Following Bach & Jordan (2003) and De la Torre & Kanade (2006), the quadratic minimization problem in C can be solved in closed form, with solution $C = (Y^\top Y)^{-1} Y^\top X$ (found by computing the matrix gradient and setting it to zero). So, the partitioning problem (with known number of clusters K) of minimizing the distortion from Section 2.1, is equivalent to:

$$\min_{Y \in \{0,1\}^{T \times K},\ Y \mathbf{1}_K = \mathbf{1}_p} \|X - Y(Y^\top Y)^{-1} Y^\top X\|_F^2 \ . \quad (1)$$

Thus, the problem is naturally parametrized by the $T \times T$-matrix $M = Y(Y^\top Y)^{-1} Y^\top$. This matrix, which we refer to as a *rescaled equivalence matrix*, has a specific structure. Since the matrix $Y^\top Y$ is diagonal, with i-th diagonal element equal to the number of elements in the cluster containing the i-th data point, $M_{ij} = 0$ if i and j are in different clusters and otherwise equal to $1/D$ where D is the number of elements in the cluster containing the i-th data point. If the points are re-ordered so that the segments are composed of contiguous elements, then M is block-diagonal with blocks $\mathbf{1}\mathbf{1}^\top / T_j$, $j = 1, \dots, K$. In this paper, we use this representation of partitions. Note the difference with alternative representations $YY^\top$ which has values in $\{0,1\}$, used for instance by Joulin et al. (2010).

We denote by $\mathcal{M}_K$ the set of rescaled equivalence matrices with K clusters, i.e., matrices $M = Y(Y^\top Y)^{-1} Y^\top \in \mathbb{R}^{T \times T}$ for some assignment matrix $Y \in \mathbb{R}^{T \times K}$. For situations where the number of clusters is unspecified, we denote by $\mathcal{M}$ the union of all $\mathcal{M}_K$ for $K \in \{1, \dots, T\}$.

Note that the number of clusters may be obtained from M, since $\mathrm{Tr}\, M = \mathrm{Tr}\, Y(Y^\top Y)^{-1} Y^\top = \mathrm{Tr}(Y^\top Y)^{-1} Y^\top Y = K$. This can also be seen by noticing that $M^2 = Y(Y^\top Y)^{-1} Y^\top Y (Y^\top Y)^{-1} Y^\top = M$, i.e., M is a projection matrix, with eigenvalues in $\{0,1\}$, and the number of eigenvalues equal to one is exactly the number of clusters. Thus, $\mathcal{M}_K = \big\{ M \in \mathcal{M},\ \mathrm{Tr}\, M = K \big\}$.

Learning the number of clusters K. Given the number of clusters K, we have seen from Eq.(1) that the partitioning problem is equivalent to

$$\min_{M \in \mathcal{M}_K} \|X - MX\|_F^2 = \min_{M \in \mathcal{M}_K} \mathrm{Tr}\big[XX^\top (I - M)\big]. \quad (2)$$

Note that in change-point detection problems, an extra constraint of contiguity of segments is added (see Section 2.3).

In the common situation where the number of clusters K is unknown, it may be estimated directly from data by penalizing the distortion measure with a term proportional to the number of clusters, as usually done for instance in change-point detection (Lavielle, 2005). This is a classical idea that can be traced back to the AIC criterion (Akaike, 1974). Given that the number of clusters for a rescaled equivalence matrix M is $\mathrm{Tr}\, M$, this leads to the following formulation:

$$\min_{M \in \mathcal{M}} \mathrm{Tr}\big[XX^\top (I - M)\big] + \lambda\, \mathrm{Tr}\, M \ . \quad (3)$$

Note that our metric learning algorithm of Section 3 also learns this extra parameter λ.

Thus, the two types of partitioning problems (with fixed or unknown number of clusters) can be cast as the problem of maximizing a linear function of the form $\mathrm{Tr}(AM)$ with respect to $M \in \mathcal{M}$, with the potential constraint that $\mathrm{Tr}\, M = K$. In general, such optimization problems may not be solved in polynomial time. In Section 2.3, we present a polynomial-time dynamic programming approach that can solve the problem with additional hard contiguity constraint. For general situations, the K-means algorithm, although not exact, can be used to get a good partitioning in polynomial time. In Section 2.4, we provide a spectral relaxation, which can be used when adding a soft constraint.

Algorithm 1 Dynamic programming for maximizing $\mathrm{Tr}(AM)$ such that $M \in \mathcal{M}^{\mathrm{seq}}$

Input: Cost matrix $A \in \mathbb{R}^{T\times T}$ and its image integral I ($I_{k,j} = \sum_{p_1\leq k, p_2\leq j} A_{p_1,p_2}$)
$\forall i \in [[1 \dots T]]$, initialize $C(1,i) = I(i,i)/i$.
for $t = 1$ to $T-1$ **do**
  $M = \max_{i\leq t} C(i,t)$
  for $u = t+1$ **to** T **do**
    $C(t+1,u) = \frac{I(t,t)+I(u,u)-I(u,t)+I(t,u)}{u-t} + M$
  end for
end for
Backtracking step:
$T_c(1) = T$
repeat
  $T_c(end+1) = \mathrm{argmax}_{i<T_c(end)} C(i, T_c(end))$
until $T_c(end) > 1$
Output: Time of changes T_c

2.3. Hard prior : change-point detection by dynamic programming

The change-point detection problem is a restriction of the general partitioning problem where the segments are composed of contiguous elements. We denote by $\mathcal{M}^{\mathrm{seq}}$ the set of partition matrices for the change-point detection problem, and $\mathcal{M}_K^{\mathrm{seq}}$, its restriction to partitions with K segments.

The problem is thus of solving Eq.(2) (known number of clusters) or Eq.(3) (unknown number of clusters) with the extra constraint that $M \in \mathcal{M}^{\mathrm{seq}}$. This may be cast as maximizing $\mathrm{Tr}(AM)$ with respect to $\mathcal{M}_K^{\mathrm{seq}}$ or $\mathcal{M}_K$, for a certain matrix A. When A is positive-semidefinite and a square root is known, the contiguity constraint leads to *exact* polynomial-time algorithms based on dynamic programming (see, e.g., Rigaill 2010; Killick et al. 2012). However, both for Eq. (3) and more generally for all loss-augmented inference problems in Section 3, we need to maximize $\mathrm{Tr}(AM)$ where A is any symmetric matrix.

Algorithm 1 above solves $\max_{M\in\mathcal{M}^{\mathrm{seq}}} \mathrm{Tr}(AM)$ for any matrix A, potentially with negative eigenvalues. It has complexity $O(T^2)$.

It only requires some preprocessing of the input matrix A, namely computing its summed area table I (or image integral), defined to have the same size as A and with $I_{ij} = \sum_{i'\leq i,\ j'\leq j} A_{i'j'}$ (i.e., the sum of the elements of A which are above i and to the left of j). A similar algorithm can be derived in the case where $M \in \mathcal{M}_K^{\mathrm{seq}}$, with complexity $O(KT^2)$.

2.4. Spectral relaxation for soft priors

For soft priors, instead of considering a subset of $\mathcal{M}_K$, we consider all possible rescaled equivalence matrices and optimize the following penalized model, where L is the Laplacian matrix of a certain graph:

$$\min_{M\in\mathcal{M}_K} \|X - MX\|_F^2 + \mathrm{Tr}(LM) \Leftrightarrow \max_{M\in\mathcal{M}_K} \mathrm{Tr}((XX^T - L)M) \tag{4}$$

Since the problem of optimizing this distortion is NP-hard (Aloise et al., 2009), we need to approximately perform the decoding. Following Shi & Malik (1997) and Ng et al. (2002), we now present a spectral relaxation of this problem. This is done by relaxing the set $\mathcal{M}$ to the set of matrices that satisfy $M^2 = M$ (i.e., removing the constraint that M takes a finite number of distinct values). When the number of clusters is known, this leads to the classical spectral relaxation, i.e., $\max_{M\in\mathcal{M},\ \mathrm{Tr}\, M=K} \mathrm{Tr}(AM) \leqslant \max_{M^2=M,\ \mathrm{Tr}\, M=K} \mathrm{Tr}(AM)$, which is equal to the sum of the K largest eigenvalues of A; the optimal matrix M of the spectral relaxation is the orthogonal projector on the eigenvectors of A with K largest eigenvalues.

When the number of clusters is unknown, we can penalize the model in Eq. (4) by the same term as in Eq. (3) and consider the whole set $\mathcal{M}$. Then we have $\max_{M\in\mathcal{M}} \mathrm{Tr}(AM) \leqslant \max_{M^2=M} \mathrm{Tr}(AM) = \mathrm{Tr}(A)_+$, where $\mathrm{Tr}(A)_+$ is the sum of positive eigenvalues of A. The optimal matrix M of the spectral relaxation is the orthogonal projector on the eigenvectors of A with positive eigenvalues. Note that in the formulation from Eq. (3), this corresponds to thresholding all eigenvalues of $XX^\top$ which are less than λ.

We denote by $\mathcal{M}^{\mathrm{spec}} = \{M \in \mathbb{R}^{T\times T},\ M^2 = M\}$ and $\mathcal{M}_K^{\mathrm{spec}} = \{M \in \mathbb{R}^{T\times T},\ M^2 = M,\ \mathrm{Tr}\, M = K\}$ the relaxed sets of rescaled equivalence matrices.

Spectral decoding. From the relaxed solution, it can sometimes be of interest to get a hard one. This problem is closely related to spectral clustering (Ng et al., 2002), and one way to obtain an hard assignment is to run K-means over the K leading eigenvectors of the spectral solution.

2.5. Metric learning

In this paper, we consider learning a *Mahalanobis metric*, which may be parametrized by a positive definite matrix $B \in \mathbb{R}^{p\times p}$. Equivalently, we replace the dot-products $x_i^\top x_j$ by $x_i^\top B x_j$, and $XX^\top$ by $XBX^\top$. Thus, in the case of the sequential hardcoded prior of Section 2.3, this corresponds to:

$$\min_{M\in\mathcal{M}_K^{\mathrm{seq}}} \mathrm{Tr}\left[XBX^\top(I-M)\right] . \tag{5}$$

When the number of segments is unknown, we penalize by adding $\lambda\,\mathrm{Tr}\, M$. In this case, note that by replacing B by $B\lambda$ and dividing the equation by λ, we can use an equivalent formulation with $\lambda = 1$, that is:

$$\min_{M\in\mathcal{M}^{\mathrm{spec}}} \mathrm{Tr}\left[XBX^\top(I-M)\right] + \mathrm{Tr}\, M . \tag{6}$$

For the soft prior of Section 2.4, the corresponding models become $\min_{M\in\mathcal{M}_K} \operatorname{Tr}[XBX^\top(I-M)+LM]$ and $\min_{M\in\mathcal{M}} \operatorname{Tr}[XBX^\top(I-M)+LM] + \operatorname{Tr} M$. The key aspect of the partitioning problem is that it is formulated as optimizing with respect to M a function *linearly* parametrized by B. The linear parametrization in M will be useful when defining proper losses and efficient loss-augmented inference in Section 3.

Note that we may allow B to be just positive semi-definite. In that case, the zero-eigenvalues of the pseudo-metric correspond to irrelevant directions. This means in particular we have performed dimensionality reduction on the input data. We propose a simple way to encourage this desirable property in Section 3.4.

3. Structured prediction for metric learning

Our goal is to learn a positive definite matrix B, in order to improve the performance of the structured output algorithm that solves the minimization problem of Eq. (5) or Eq. (6). The partitioning problem can be cast as $\max_{M\in\mathcal{M}_K} \langle w, \varphi(X,M)\rangle$ or $\max_{M\in\mathcal{M}} \langle w, \varphi(X,M)\rangle$, where $\langle A, B\rangle$ is the Frobenius dot product. When the number of clusters is known ($\operatorname{Tr} M = K$), then $\varphi(X,M) = X^\top M X$ and $w = B$; otherwise, $\varphi(X,M) = \frac{1}{T}\operatorname{Diag}(X^\top M X, M)$ and $w = \operatorname{Diag}(B, -I)$.

Let $\mathcal{F}$ denote the vector space where the above-defined w lies. Our goal is to estimate $w \in \mathcal{F}$ from N pairs of observations $(X_i, M_i) \in \mathcal{X}\times\mathcal{M}$. This is exactly the goal of large-margin structured prediction (Tsochantaridis et al., 2005), which we now present. We denote by $\mathcal{N}$ a generic set of matrices, which may either be $\mathcal{M}$, $\mathcal{M}^{\rm spec}$, $\mathcal{M}^{\rm seq}$, $\mathcal{M}_K$, $\mathcal{M}_K^{\rm spec}$, $\mathcal{M}_K^{\rm seq}$, depending on the situation (see also Section 3.3).

3.1. Large-margin structured output learning

In the margin-rescaling framework of Tsochantaridis et al. (2005), using a loss ℓ: $\mathcal{N}\times\mathcal{N}\to\mathbb{R}_+$ between elements of $\mathcal{N}$ (here partitions), the goal is to minimize with respect to $w\in\mathcal{F}$, $\frac{1}{N}\sum_{i=1}^N \ell\big(\operatorname{argmax}_{M\in\mathcal{N}}\langle w, \varphi(X_i,M)\rangle, M_i\big) + \Omega(w)$, where Ω is any (typically convex) regularizer. This framework is standard in machine learning in general and metric learning in particular (see, e.g, Jain et al. 2012). The loss function $w\mapsto \ell(\operatorname{argmax}_{M\in\mathcal{N}}\langle w,\varphi(X_i,M)\rangle, M_i)$ is not convex in M, and can be replaced by the convex surrogate $L_i(w) = \max_{M\in\mathcal{N}}\big\{\ell(M,M_i) + \langle w, \varphi(X_i,M) - \varphi(X_i,M_i)\rangle\big\}$, leading to an estimator $\widehat{w}$ minimizing

$$\frac{1}{N}\sum_{i=1}^N L_i(w) + \Omega(w) \ . \tag{7}$$

In order to apply this framework, several elements are needed: (a) a regularizer Ω, (b) a loss function ℓ, and (c) the associated efficient algorithms for computing L_i, i.e., solving the *loss-augmented inference* problem $\max_{M\in\mathcal{N}}\big\{\ell(M,M_i) + \langle w, \varphi(X_i,M) - \varphi(X_i,M_i)\rangle\big\}$.

Optimization in w. Given that the objective function is not smooth, we have used projected subgradient descent (stochastic and non-stochastic), with convergence rates of $O(1/t)$ after t iterations (Shalev-Shwartz et al., 2007).

3.2. Loss between partitions

In this paper, we consider the following loss:

$$\ell(M,N) = \|M-N\|_F^2 = \operatorname{Tr}(M) + \operatorname{Tr}(N) - 2\operatorname{Tr}(MN), \tag{8}$$

which is practical (it is a bilinear function of M and N) and corresponds to a well-known loss between partitions (Hubert & Arabie., 1985; Bach & Jordan, 2003). If the partitions encoded by M and N have clusters $A_1,\dots,A_K$ and $B_1,\dots,B_L$, then $\ell(M,N) = K + L - 2\sum_{k,l}\frac{|A_k\cap B_l|^2}{|A_k|\cdot|B_l|}$. This loss is equal to zero if and only if the partitions are equal, always larger than $|K-L|$ and smaller than $K+L-2$. Other choices are classical in the literature of metric learning, in particular the loss associated to the Rand index (Hubert & Arabie., 1985; Finley & Joachims, 2005), that is, $d_{\rm Rand} = 1 - \operatorname{Rand}$ where $\operatorname{Rand}(P,Q) := 1 - \frac{1}{T(T-1)}\left\|Y_P Y_P^T - Y_Q Y_Q^T\right\|_F^2$ for two partitions P and Q with equivalence matrices $Y_P Y_P^T$ and $Y_Q Y_Q^T$ respectively. The Rand index/loss is not necessarily well suited to our problem, since intuitively it doesn't take into account the size of each cluster, whereas our concern is to optimize intra class variance which is a rescaled indicator.

3.3. Loss-augmented inference problem

Efficient minimization is key to the applicability of large-margin structured prediction and this problem is a classical computational bottleneck. In our situation the cardinality of $\mathcal{N}$ is exponential in T, but our choice for the loss ℓ leads to the problem $\max_{M\in\mathcal{N}}\operatorname{Tr}(A_i M)$ where $A_i = \frac{1}{T}(X_i B X_i^\top - 2M_i + \operatorname{Id})$ if the number of clusters is known, and $A_i = \frac{1}{T}(X_i B X_i^\top - 2M_i)$ otherwise. Thus, the loss-augmented problem can be performed exactly for the change-point problems (that is, $\mathcal{N}\in\{\mathcal{M}^{\rm seq}, \mathcal{M}_K^{\rm seq}\}$; see Section 2.3) or through a spectral relaxation otherwise (that is, $\mathcal{N}\in\{\mathcal{M}^{\rm spec}, \mathcal{M}_K^{\rm spec}\}$; see Section 2.4).

3.4. Regularizer

Several regularizers Ω and parametrizations for B can be chosen. The most popular choice for Ω is the Frobenius norm (see, e.g, Tsochantaridis et al. 2005; Jain et al. 2012). The following two variants are often needed depending on the application.

Low-rank metric. A desirable property for the learned metric is to be interpretable. Ideally, we would like to have a pseudo-metric with a small rank. The classical relaxation

of the rank is the sum of the singular values, that is, $\Omega(w) = \mathrm{Tr}(B)$ since B is symmetric positive semi-definite.

Diagonal metric. Considering only diagonal matrices $B = \mathrm{Diag}(b)$ for some $b \in \mathbb{R}^p$ with $b \geqslant 0$ limits the number of parameters to learn, and reduces the metric learning problem to reweighting the coordinates of the data. Then, the two proposed regularizers can be written $\|b\|_2^2$ and $\|b\|_1 = \mathbf{1}_p^\top b$, the latter leading to variable selection.

4. Extensions

4.1. Partial labellings

The large-margin convex optimization framework relies on fully labeled datasets $(X_i, M_i)_{i=1,\dots,N}$ where X_i is a time series and M_i the corresponding rescaled equivalence matrix. In many situations however, only partial information is available about the partition associated to each X_i. Then, starting from the PCA metric, we propose to iterate between (a) label all datasets using the current metric and respecting the constraints imposed by the partial labels and (b) learn the metric using Section 3 from the fully labeled datasets. See an application in Section 5.2.

4.2. Detecting changes in the distribution of temporal signals

A priori, the approach to change-point detection presented in Section 3.3 can only detect changes in the mean of the distribution of the x_j because it starts from the distortion measure of Section 2.1. Nevertheless, in the literature, change-point detection refers to the more general problem of finding changes in the whole distribution of the x_j (Basseville & Nikiforov, 1993), taking into account other features of the distribution like the variance or the kurtosis. In order to tackle this problem when $x_j \in \mathbb{R}$, we propose to apply our approach to the time series $(f_i(x_j))_{i=1\dots r} \in \mathbb{R}^r$, $j = 1 \dots T$, where the f_i are well-chosen functions so that changes in the distribution of x_j appear through changes in the mean of $f_i(x_j)$. For instance, in order to detect changes in the first moments of the distribution, a naive choice is $f_i(x) = x^i$, but the x_j^i explode when i grows. A way to prevent them from exploding is to use the robust Hermite moments (Welling, 2005), that is, to take $f_i(x) = H_i(x) = 2\sqrt{2^i \pi i!} e^{-\frac{x^2}{2\sigma^2}} (-1)^i 2^{i/2} e^{\frac{x^2}{2}} \frac{d^i}{dx^i}\left(e^{\frac{-x^2}{2}}\right)$ the i-th Hermite function. See an application in Section 5.4.

5. Experiments

5.1. Synthetic example.

We consider synthetic time series of dimension $p = 300$ and length $T = 600$ with $K = 3$ relevant changes in the mean of a few dimensions. Among these dimensions, 10 are relevant time series with aligned ruptures in their mean that we want to detect. The others are 290 noisy random series, either with no changes in their global mean or changes which are not aligned with the ones we aim at detecting. By learning a metric, we hope to obtain high weights on the relevant coordinates and small weights on the others.

Given $N = 100$ instances of such time series with the same 10 relevant coordinates, we compare the performance of our algorithm to the Euclidean metric (that is, Eq. (6) with $B = \alpha I$ and $\alpha > 0$ learned on a validation set), the PCA metric (obtained simply by performing a PCA, keeping the three leading eigenvectors), and two state-of-the-art semi-supervised metric learning algorithms for clustering (Xing et al. 2002, and the RCA approach of Bar-Hillel et al. 2006), for which we stacked all datasets into a single one with the corresponding supervision. Note that all algorithms except ours are given the exact true number of change-points $K^\star$. Results are shown on Figure 1 (points at the extreme right of the graph), illustrating the interest of learning a metric (Euclidean is bad and PCA only slightly better), and showing our approach does significantly better than RCA. Note that RCA is not directly adapted to change-point detection, it requires moreover dimensionality reduction to work and the performance is not robust to the choice of the number of coordinates.

5.2. Robustness to partial labelling

We extended the above experiment to the case of partial labellings, using the approach of Section 4.1 for our algorithm. Results are presented on Figure 1, where the x-axis represents the fraction of the labels $M_1, \dots, M_N$ available to the two existing semi-supervised clustering methods and our algorithm, showing the same ordering between the algorithms, except when the fraction of available labels is very small.

5.3. Video segmentation

We applied our method to data coming from old TV shows (the length of the time series in that case is about 5400, with 60 to 120 change-points) where some speaking passages alternate with singing ones. The videos are from 60 up to 90 minutes long. We aim at recovering the segmentation induced by the speaking parts and the musical ones. As in Arlot et al. (2012), we use GIST features for the video part and MFCC features for the audio (12 first standard coefficients). We rescaled the videos frames to small size (64 by 64) before computing these GIST with 4 prefilters, 4 different scales (with 8 orientations at each scale and thus 32 filters in total). In the end each image was represented by a vector of length 512. The features were aggregated every second so that we consider time series of length a few thousands, which is still computationally tractable using Algorithm 1. Using 4 shows for train, 3 for validation,

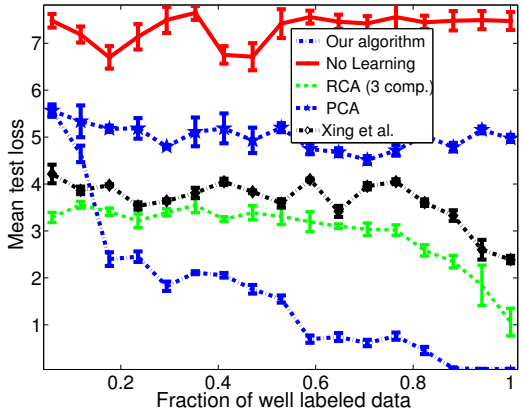

Figure 1. Performances on synthetic data vs. the quantity of information available in the time series, measured in terms of the loss ℓ defined by Eq. (8). Note the small error bars (90% quantiles). We compare ourselves against the Euclidean metric ('No learning'), a metric learned by RCA (with 3 components), PCA and Xing et al. (2002).

3 for test, we report below the test errors for each test show with the loss ℓ (smaller is better).

Method	Audio			Video			Both		
PCA	23	41	34	40	55	25	29	53	37
No metric learning	29	48	33	59	55	47	40	48	36
Our algorithm	**6.1**	**9.3**	**7**	**10**	**14**	**11**	**8.7**	**9.6**	**7.8**

We consider three different settings: using only the image stream, only the audio stream or both. In these three cases, we consider using the existing metric (no learning), PCA, or our approach. In all settings, metric learning improves performance. Note that the performance is best with only the audio stream; our metric learning, given both streams, manages to do almost as well as with only the audio stream, thus illustrating the robustness of using metric learning in this context where the video stream is not useful.

5.4. Bioinformatics application

Detection of change-points in DNA sequences for cancer prognosis provides a natural testbed for the approach of Section 4.2. Indeed, researchers from this field face data which are linked to the number of copies of each gene along the DNA (a-CGH data as used by Hocking et al. 2013). The presence of such changes is generally related to the development of certain types of cancers. On the data from the Neuroblastoma dataset (Hocking et al., 2013), some caryotypes with changes of distribution were manually annotated. We consider the approach of Section 4.2 and compare the Euclidean and a learned metric on the five first Hermite moments of the data. Without any metric learning, just by adjusting the AIC criterion by trial and error, we reach a global error rate in change-point identification of 12%. By learning a diagonal metric over Hermite features, we reach a rate of 6.9%, thus improving significantly the performance. Note that, in this application it is *irrelevant* to run standard metric learning like RCA, since they are unable to do model selection and learn the parameter λ of penalization of Eq. (3). Note that λ can be seen as the expected level of noise in the series.

Table 1. Test performance on the Horses dataset according to our loss and the standard Rand loss (1-Rand Index) Lower is better.

Loss used	Constrained metric learning	Unconstrained metric learning	Ncuts
ℓ	1.51	1.73	1.81
Rand loss	0.37	0.43	0.48

5.5. Image segmentation

Most of popular image segmentation algorithms are not based on minimizing Euclidean distortions (even if sometimes the use of simple K-means, for instance over colors features, can lead to very good segmentation, see, e.g, Forsyth & Ponce 2002). The main drawback of such methods is that they do not push pixels which are close to belong to the same cluster.

The normalized cuts framework (Shi & Malik, 1997), which is popular for segmenting images, has some relation with our framework, since it can be cast as $\max(\mathrm{Tr}(WM))$ where W is some variant of the normalized Laplacian (Bach & Jordan, 2003).

Inspired by this connection, we propose a simple foreground/background segmentation model which consists in adding a prior term to K-means, as proposed in Eq. (4), namely considering a decoding of the type $\max_{M \in \mathcal{M}_{K=2}} \mathrm{Tr}(XBX^T M) - \mathrm{Tr}(LM)$ where L is the normalized Laplacian (in our experiments the use of the different versions of the Laplacian did not lead to significantly different performances) of the graph underlying the image. This second term permits to give to spatially contiguous clusters the preference over non contiguous ones. Note that in our experiments, we simply consider the graph associated to the 4-connected grid.

We consider the task of segmenting images of the Weizmann horses dataset (Borenstein & Ullman, 2004), using $N = 20$ training images with colour and dense SIFT features. Results are presented in Table 1, where we used both normalized cuts and an unconstrained metric learning technique (Finley & Joachims, 2008) as baselines. In Table 2, we present analogous results for the Oxford flowers (Nilsback & Zisserman, 2006) dataset, for which the training set size is bigger: 150 images. Note that the parameter of the structured SVM is simply adjusted using a validation set. To assess significancy of the difference between the mean of the loss obtained with our algorithm and baselines we

Table 2. Test performance on the Flowers dataset according to our loss and the standard (see Unnikrishnan et al. 2007) Rand loss.

Loss used	Constrained metric learning	Unconstrained metric learning	Ncuts
ℓ	0.94	1.31	1.59
Rand loss	0.25	0.38	0.46

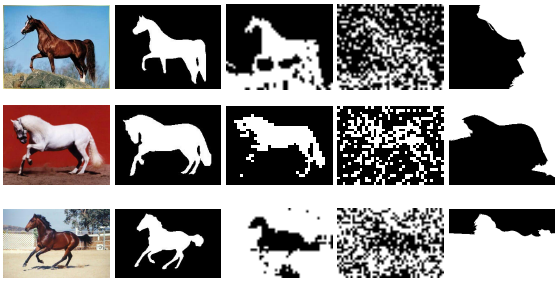

Figure 2. From left to right: original image, groundtruth segmentation, image segmented with our learned metric, image segmented learning just a metric with no prior, Ncuts with tuned parameters for colors and position features.

propose the boxplots of Fig. 4 in the case of the flowers dataset (similar results hold for the Horses dataset). Finally, in Fig. 2 and 3, we present some cases where the learning of the metric has lead to significant improvements over the baselines.

In these image segmentation experiments, the approach based on unconstrained metric learning leads to inferior performance while our approach based on appropriately constrained metric learning leads to improvements over normalized cuts.

6. Conclusion

In this paper we have addressed the problem of learning a metric in a supervised way for improving the performance of unsupervised partitioning algorithms. We have focused on the practically important case in which a prior over the resulting partition is available. More precisely we have demonstrated that, for the temporal sequential prior, such a metric can be learned in an efficient way using a structured SVM. We explored several applications, in

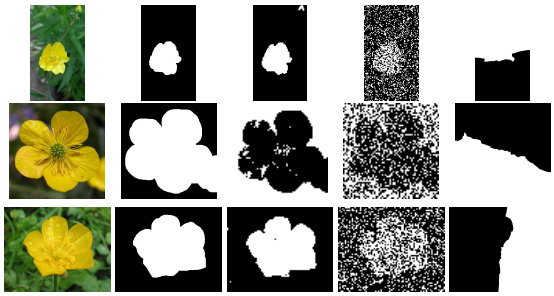

Figure 3. The images are in the same order as in Fig.2

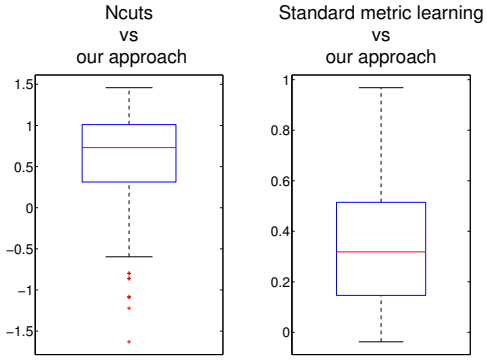

Figure 4. Boxplots for the differences of test error, with the loss ℓ, between baselines and our approach. Positive values mean our algorithm does better. Note that sometimes Ncuts outperforms our metric learning but the standard version never beats the constrained one. Evaluation was performed over a test set of 653 labeled instances.

particular the detection of change-points in video streams or DNA sequences and the problem of image segmentation, with a significant improvement in partitioning performance. Then, driven by the case in which the prior is given through a graph Laplacian, we have proposed a soft model based on Euclidean distortion that we plugged into a structured SVM. We demonstrated that approach is well founded with experiments on image segmentation datasets.

For future works, following recent trends in image segmentation (see, e.g., Joulin et al. 2010), it would be interesting to extend our change-point framework so that it allows unsupervised co-segmentation of several videos: each segment could then be automatically labeled so that segments from different videos but with the same label correspond to the same action. Another extension would be to generalize our algorithm to kernel learning. Indeed, some recent work (Jain et al., 2012) proved links between metric learning and kernel learning, allowing to kernelize any Mahalanobis distance learning problem.

Acknowledgments

We acknowledge the support of the GARGANTUA project (Mastodons program of CNRS), the grant SIERRA-23999 from the European Research Council and a PhD fellowship from the EADS Foundation. We also thank Damien Garreau, Zaid Harchaoui, Toby Hocking, Ivan Laptev, as well as the anonymous reviewers for their constructive remarks and discussions.

References

Akaike, H. A new look at the statistical model identification. *IEEE Trans. on Aut. Cont.*, 19(6):716 – 723, 1974.

Aloise, D., Deshpande, A., Hansen, P., and Popat, P. NP-hardness of euclidean sum-of-squares clustering. *Machine Learning*, 75 (2):245–248, 2009.

Arlot, S., Celisse, A., and Harchaoui, Z. Kernel change-point detection, 2012. arXiv:1202.3878.

Bach, F. and Jordan, M. Learning spectral clustering. In *Adv. NIPS*, 2003.

Bar-Hillel, A., Hertz, T., Shental, N., and Weinshall, D. Learning a mahalanobis metric from equivalence constraints. *Journal of Machine Learning Research*, 6:937, 2006.

Basseville, M. and Nikiforov, I. *Detection of abrupt changes: theory and application*. Prentice Hall Information and System Sciences Series. Prentice Hall, 1993.

Borenstein, E. and Ullman, S. Learning to segment. In *Proc. ECCV*, 2004.

Brodsky, B. and Darkhovsky, B. *Non-Parametric Statistical Diagnosis: Problems and Methods,*. Springer, 1 edition, April 1993.

Caetano, T.., Cheng, L., Le, Q., and Smola, A. Learning Graph Matching. 2007.

Chen, J. and Gupta, A. K. *Parametric Statistical Change Point Analysis*. Birkhäuser, 2011.

Cheng, Y. Mean shift, mode seeking, and clustering. *IEEE Trans. PAMI*, 17(8):790–799, 1995.

De la Torre, F. and Kanade, T. Discriminative cluster analysis. In *Proc. ICML*, 2006.

Desobry, F., Davy, M., and Doncarli, C. An online kernel change detection algorithm. *IEEE Trans. Sig. Proc.*, 53(8):2961–2974, 2005.

Finley, T. and Joachims, T. Supervised clustering with support vector machines. In *Proc. ICML*, 2005.

Finley, T. and Joachims, T. Supervised k-means clustering, 2008.

Forsyth, D. and Ponce, J. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference, 2002.

Gillet, O., Essid, S., and Richard, G. On the correlation of automatic audio and visual segmentations of music videos. *IEEE Trans. Circ. Syst. Vid. Tech.*, 17(3):347–355, 2007.

Goldberger, J., Roweis, S., Hinton, G., and Salakhutdinov, R. Neighbourhood component analysis. In *Adv. NIPS*, 2004.

Gower, J. C. and Ross, G. J. S. Minimum spanning trees and single linkage cluster analysis. *Applied statistics*, pp. 54–64, 1969.

Hocking, T., Schleiermacher, G., Janoueix-Lerosey, I., Boeva, V., Cappo, J., Delattre, O., Bach, F., and Vert, J. Learning smoothing models of copy number profiles using breakpoint annotations. *BMC Bioinformatics*, 14(164), 2013.

Hubert, L. J. and Arabie., P. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

Jain, P., Kulis, B., Davis, J., and Dhillon, I. Metric and kernel learning using a linear transformation. *Journal of Machine Learning Research*, 13:519–547, 2012.

Joulin, A., Bach, F., and Ponce, J. Discriminative clustering for image co-segmentation. In *Proc. CVPR*, 2010.

Killick, R., Fearnhead, P., and Eckley, I. A. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107(500):1590–1598, 2012.

Kim, Sungwoong, Nowozin, Sebastian, Kohli, Pushmeet, and Yoo, C. Task-specific image partitioning. *IEEE Trans. Image Proc.*, 2012.

Kulis, B. and Jordan, M. I. Revisiting K-means: New algorithms via Bayesian nonparametrics. In *Proc. ICML*, 2012.

Lavielle, Marc. Using penalized contrasts for the change-point problem. *Signal Proces.*, 85(8):1501–1510, 2005.

Mcfee, B. and Lanckriet, G. Metric learning to rank. In *Proc. ICML*, 2010.

Ng, A. Y., Jordan, M. I., and Weiss, Y. On spectral clustering: Analysis and an algorithm. *Adv. NIPS*, 2002.

Nilsback, M. and Zisserman, A. A visual vocabulary for flower classification. In *Proc. CVPR*, volume 2, pp. 1447–1454, 2006.

Rigaill, G. Pruned dynamic programming for optimal multiple change-point detection. Technical Report 1004.0887, arXiv, 2010.

Shalev-Shwartz, S., Singer, Y., and Srebro, N. Pegasos: Primal estimated sub-gradient solver for SVM. In *Proc. ICML*, 2007.

Shi, J. and Malik, J. Normalized cuts and image segmentation. *IEEE Trans. PAMI*, 22:888–905, 1997.

Szummer, M., Kohli, P., and Hoiem, D. Learning CRFs using graph cuts. *Proc. ECCV*, 2008.

Taskar, B., Guestrin, C., and Koller, D. Max-margin Markov networks. *Adv. NIPS*, 2003.

Tsochantaridis, I., Hoffman, T., Joachims, T., and Altun, Y. Support vector machine learning for interdependent and structured output spaces. *Journal of Machine Learning Research*, pp. 1453–1484, 2005.

Unnikrishnan, R., Pantofaru, C., and Hebert, M. Toward objective evaluation of image segmentation algorithms. *Trans. PAMI*, 29 (6):929–944, June 2007.

Weinberger, Kilian Q., Blitzer, John, and Saul, Lawrence K. Distance metric learning for large margin nearest neighbor classification. In *Adv. NIPS*, 2006.

Welling, M. Robust higher order statistics. *Proc. AISTATS*, 2005.

Xing, E. P., Ng, A. Y., Jordan, M. I., and Russell, S. Distance metric learning with applications to clustering with side-information. *Adv. NIPS*, 2002.