Dynamic metric learning from pairwise comparisons

Kristjan Greenewald, *Student Member, IEEE,* Stephen Kelley, and Alfred O. Hero III, *Fellow, IEEE*

***Abstract*—Recent work in distance metric learning has focused on learning transformations of data that best align with specified pairwise similarity and dissimilarity constraints, often supplied by a human observer. The learned transformations lead to improved retrieval, classification, and clustering algorithms due to the better adapted distance or similarity measures. Here, we address the problem of learning these transformations when the underlying constraint generation process is nonstationary. This nonstationarity can be due to changes in either the ground-truth clustering used to generate constraints or changes in the feature subspaces in which the class structure is apparent. We propose Online Convex Ensemble StrongLy Adaptive Dynamic Learning (OCELAD), a general adaptive, online approach for learning and tracking optimal metrics as they change over time that is highly robust to a variety of nonstationary behaviors in the changing metric. We apply the OCELAD framework to an ensemble of online learners. Specifically, we create a retro-initialized composite objective mirror descent (COMID) ensemble (RICE) consisting of a set of parallel COMID learners with different learning rates, demonstrate RICE-OCELAD on both real and synthetic data sets and show significant performance improvements relative to previously proposed batch and online distance metric learning algorithms.**

I. Introduction

The effectiveness of many machine learning and data mining algorithms depends on an appropriate measure of pairwise distance between data points that accurately reflects the learning task, e.g., prediction, clustering or classification. The kNN classifier, K-means clustering, and the Laplacian-SVM semi-supervised classifier are examples of such *distance-based* machine learning algorithms. In settings where there is clean, appropriately-scaled spherical Gaussian data, standard Euclidean distance can be utilized. However, when the data is heavy tailed, multimodal, or contaminated by outliers, observation noise, or irrelevant or replicated features, use of Euclidean inter-point distance can be problematic, leading to bias or loss of discriminative power.

To reduce bias and loss of discriminative power of distance-based machine learning algorithms, data-driven approaches for optimizing the distance metric have been proposed. These methodologies, generally taking the form of dimensionality reduction or data "whitening", aim to utilize the data itself to learn a transformation of the data that embeds it into a space where Euclidean distance is appropriate. Examples of such techniques include Principal Component Analysis [1], Multidimensional Scaling [2], covariance estimation [2], [1], and manifold learning [3]. Such unsupervised methods do not exploit human input on the distance metric, and they overly rely on prior assumptions, e.g., local linearity or smoothness.

K. Greenewald and A. Hero III are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, USA. This work was partially supported by US Army Research Office grant W911NF-15-1-0479.

In distance metric learning one seeks to learn transformations of the data associated with a distance metric that is well matched to a particular task specified by the user. Point labels or constraints indicating point similarity or dissimilarity are used to learn a transformation of the data such that similar points are "close" to one another and dissimilar points are distant in the transformed space. Learning distance metrics in this manner allows a more precise notion of distance or similarity to be defined that is better related to the task at hand.

Many supervised and semi-supervised distance metric learning approaches have been developed [4]. This includes online algorithms [5] with regret guarantees for situations where similarity constraints are received sequentially.

This paper proposes a new distance metric tracking method that is applicable to the non-stationary time varying case of distance metric drift and has provably *strongly adaptive* tracking performance.

Specifically, we suppose the underlying ground-truth (or optimal) distance metric from which constraints are generated is evolving over time, in an unknown and potentially nonstationary way. We propose a strongly adaptive, online approach to track the underlying metric as the constraints are received. We introduce a framework called Online Convex Ensemble StrongLy Adaptive Dynamic Learning (OCELAD), which at every time step evaluates the recent performance of and optimally combines the outputs of an ensemble of online learners, each operating under a different drift-rate assumption. We prove strong bounds on the dynamic regret of every subinterval, guaranteeing strong adaptivity and robustness to nonstationary metric drift such as discrete shifts, slow drift with a widely-varying drift rate, and all combinations thereof. Applying OCELAD to the problem of nonstationary metric learning, we find that it gives excellent robustness and low regret when subjected to all forms of nonstationarity.

A. Related Work

Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA) are classic examples of using linear transformations for projecting data into more interpretable low dimensional spaces. Unsupervised PCA seeks to identify a set of axes that best explain the variance contained in the data. LDA takes a supervised approach, minimizing the intra-class variance and maximizing the inter-class variance given class labeled data points.

Much of the recent work in Distance Metric Learning has focused on learning Mahalanobis distances on the basis of pairwise similarity/dissimilarity constraints. These methods have the same goals as LDA; pairs of points labeled "similar" should be close to one another while pairs labeled "dissimilar"

should be distant. MMC [6], a method for identifying a Mahalanobis metric for clustering with side information, uses semidefinite programming to identify a metric that maximizes the sum of distances between points labeled with different classes subject to the constraint that the sum of distances between all points with similar labels be less than some constant.

Large Margin Nearest Neighbor (LMNN) [7] similarly uses semidefinite programming to identify a Mahalanobis distance. In this setting, the algorithm minimizes the sum of distances between a given point and its similarly labeled neighbors while forcing differently labeled neighbors outside of its neighborhood. This method has been shown to be computationally efficient [8] and, in contrast to the similarly motivated Neighborhood Component Analysis [9], is guaranteed to converge to a globally optimal solution. Information Theoretic Metric Learning (ITML) [10] is another popular Distance Metric Learning technique. ITML minimizes the Kullback-Liebler divergence between an initial guess of the matrix that parameterizes the Mahalanobis distance and a solution that satisfies a set of constraints. For surveys of the vast metric learning literature, see [4], [11], [12].

In a dynamic environment, it is necessary to track the changing metric at different times, computing a sequence of estimates of the metric, and to be able to compute those estimates online. Online learning [13] meets these criteria by efficiently updating the estimate every time a new data point is obtained, instead of solving an objective function formed from the entire dataset. Many online learning methods have regret guarantees, that is, the loss in performance relative to a batch method is provably small [13], [14]. In practice, however, the performance of an online learning method is strongly influenced by the learning rate, which may need to vary over time in a dynamic environment [15], [16], [17], especially one with changing drift rates.

Adaptive online learning methods attempt to address the learning rate problem by continuously updating the learning rate as new observations become available. For learning static parameters, AdaGrad-style methods [16], [17] perform gradient descent steps with the step size adapted based on the magnitude of recent gradients. Follow the regularized leader (FTRL) type algorithms adapt the regularization to the observations [18]. Recently, a method called Strongly Adaptive Online Learning (SAOL) has been proposed for learning parameters undergoing K discrete changes when the loss function is bounded between 0 and 1. SAOL maintains several learners with different learning rates and randomly selects the best one based on recent performance [15]. Several of these adaptive methods have provable regret bounds [18], [19], [20]. These typically guarantee low total regret (i.e. regret from time 0 to time T) at every time [18]. SAOL, on the other hand, attempts to have low *static* regret on every subinterval, as well as low regret overall [15]. This allows tracking of discrete changes, but not slow drift. Our work improves upon the capabilities of SAOL by allowing for unbounded loss functions, using a convex combination of the ensemble instead of simple random selection, and providing guaranteed low regret when all forms of nonstationarity occur, not just discrete shifts. All of these additional capabilities are shown in the results to be critical for good metric learning performance.

The remainder of this paper is structured as follows. In Section II we formalize the time varying distance metric tracking problem, and section III presents the basic COMID online learner and our Retro-Initialized COMID Ensemble (RICE) of learners with dyadically scaled learning rates. Section IV presents our OCELAD algorithm, a method of adaptively combining learners with different learning rates. Strongly adaptive bounds on the dynamic regret of OCELAD and RICE-OCELAD are presented in Section V, and results on both synthetic data and a text review dataset are presented in Section VI. Section VII concludes the paper.

II. Nonstationary Metric Learning

Metric learning seeks to learn a metric that encourages data points marked as similar to be close and data points marked as different to be far apart. The time-varying Mahalanobis distance at time t is parameterized by $\mathbf{M}_t$ as

$$d^2_{M_t}(\mathbf{x}, \mathbf{z}) = (\mathbf{x} - \mathbf{z})^T \mathbf{M}_t (\mathbf{x} - \mathbf{z}) \tag{1}$$

where $\mathbf{M}_t \in \mathbb{R}^{n \times n} \succeq 0$.

Suppose a temporal sequence of similarity constraints are given, where each constraint is the triplet $(\mathbf{x}_t, \mathbf{z}_t, y_t)$, $\mathbf{x}_t$ and $\mathbf{z}_t$ are data points in $\mathbb{R}^n$, and the label $y_t = +1$ if the points $\mathbf{x}_t, \mathbf{z}_t$ are similar at time t and $y_t = -1$ if they are dissimilar.

Following [5], we introduce the following margin based constraints:

$$\begin{aligned} t|y_t = 1 :\ & d^2_{M_t}(\mathbf{x}_t, \mathbf{z}_t) \leq \mu - 1; \\ t|y_t = -1 :\ & d^2_{M_t}(\mathbf{x}_t, \mathbf{z}_t) \geq \mu + 1, \end{aligned} \tag{2}$$

where μ is a threshold that controls the margin between similar and dissimilar points. A diagram illustrating these constraints and their effect is shown in Figure 1. These constraints are softened by penalizing violation of the constraints with a convex loss function ℓ. This gives a loss function

$$\begin{aligned} \mathcal{L}(\{\mathbf{M}_t, \mu\}) &= \frac{1}{T} \sum_{t=1}^{T} \ell(y_t(\mu - \mathbf{u}_t^T \mathbf{M}_t \mathbf{u}_t)) + \rho r(\mathbf{M}_t) \\ &= \frac{1}{T} \sum_{t=1}^{T} f_t(\mathbf{M}_t, \mu), \end{aligned} \tag{3}$$

where $\mathbf{u}_t = \mathbf{x}_t - \mathbf{z}_t$, r is the regularizer and ρ the regularization parameter. Kunapuli and Shavlik [5] propose using nuclear norm regularization ($r(\mathbf{M}) = \|\mathbf{M}\|_*$) to encourage projection of the data onto a low dimensional subspace (feature selection/dimensionality reduction), and we have also had success with the elementwise L1 norm ($r(\mathbf{M}) = \|\mathrm{vec}(\mathbf{M})\|_1$). In what follows, we develop an adaptive online method to minimize the loss subject to nonstationary smoothness constraints on the sequence of metric estimates $\mathbf{M}_t$.

III. Retro-initialized COMID ensemble (RICE)

Viewing the acquisition of new data points as stochastic realizations of the underlying distribution [5] suggests the use of composite objective stochastic mirror descent techniques

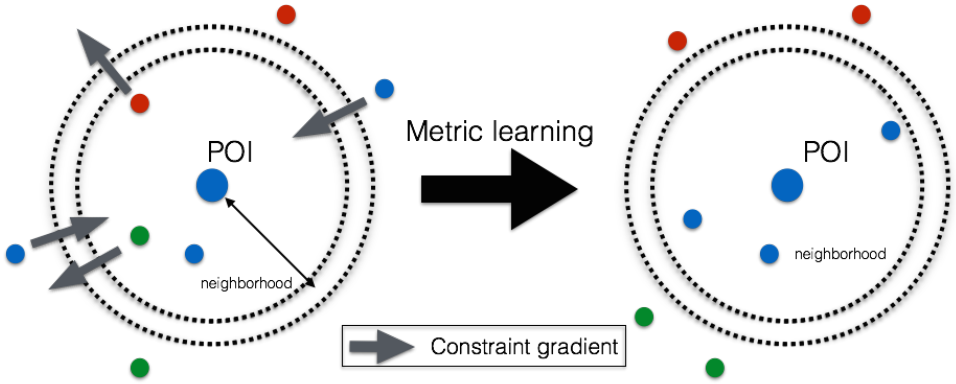


Fig. 1. Visualization of the margin based constraints (2), with colors indicating class. The goal of the metric learning constraints is to move target neighbors towards the point of interest (POI), while moving points from other classes away from the target neighborhood.

(COMID). For convenience, we set $\ell_t(\mathbf{M}_t, \mu_t) = \ell(y_t(\mu - \mathbf{u}_t^T \mathbf{M}_t \mathbf{u}_t))$.

For the loss (3) and learning rate η_t, COMID [14] gives

$$\begin{aligned} \hat{\mathbf{M}}_{t+1} = &\arg\min_{\mathbf{M}\succeq 0} B_\psi(\mathbf{M}, \hat{\mathbf{M}}_t) \\ &+ \eta_t \langle \nabla_M \ell_t(\hat{\mathbf{M}}_t, \hat{\mu}_t), \mathbf{M} - \hat{\mathbf{M}}_t \rangle + \eta_t \rho \|\mathbf{M}\|_* \\ \hat{\mu}_{t+1} = &\arg\min_{\mu \geq 1} B_\psi(\mu, \hat{\mu}_t) + \eta_t \nabla_\mu \ell_t(\hat{\mathbf{M}}_t, \hat{\mu}_t)'(\mu - \hat{\mu}_t), \end{aligned} \tag{4}$$

where B_ψ is any Bregman divergence. In [5] a closed-form algorithm for solving the minimization in (17) with $r(\mathbf{M}) = \|\mathbf{M}\|_*$ is developed for a variety of common losses and Bregman divergences, involving rank one updates and eigenvalue shrinkage.

The output of COMID depends strongly on the choice of η_t. Critically, the optimal learning rate η_t depends on the rate of change of $\mathbf{M}_t$ [21], and thus will need to change with time to adapt to nonstationary drift. Choosing an optimal sequence for η_t is clearly not practical in an online setting with nonstationary drift, since the drift rate is changing. We thus propose to maintain an ensemble of learners with a range of η_t values, whose output we will adaptively combine for optimal nonstationary performance. If the range of η_t is diverse enough, one of the learners in the ensemble should have good performance on every interval. Critically, the optimal learner in the ensemble may vary widely with time, since the drift rate and hence the optimal learning rate changes in time. For example, if a large discrete change occurs, the fast learners are optimal at first, followed by increasingly slow learners as the estimate of the new value improves. In other words, the optimal approach is fast reaction followed by increasing refinement, in a manner consistent with the attractive $O(1/\sqrt{t})$ decay of the learning rate of optimal nonadaptive algorithms.

Define a set $\mathcal{I}$ of intervals $I = [t_{I1}, t_{I2}]$ such that the lengths $|I|$ of the intervals are proportional to powers of two, i.e. $|I| = I_0 2^j$, $j = 0, \ldots,$ with an arrangement that is a dyadic partition of the temporal axis, as in [15]. The first interval of length $|I|$ starts at $t = |I|$ (see Figure 2), and additional intervals of length $|I|$ exist such that the rest of time is covered.

Every interval I is associated with a base COMID learner that operates on that interval. Each learner (17) has a constant learning rate proportional to the inverse square of the length of the interval, i.e. $\eta_t(I) = \eta_0/\sqrt{|I|}$. Each learner (besides the coarsest) at level j ($|I| = I_0 2^j$) is initialized to the last estimate of the next coarsest learner (level $j-1$) (see Figure 2). This strategy is equivalent to "backdating" the interval learners so as to ensure appropriate convergence has occurred before the interval of interest is reached, and is effectively a quantized square root decay of the learning rate. We call our method of forming an ensemble of COMID learners on dyadically nested intervals the Retro-Initialized COMID Ensemble, or RICE, and summarize it in Figure 2.

At a given time t, a set $\mathrm{ACT}(t) \subseteq \mathcal{I}$ of $\mathrm{floor}(\log_2 t)$ intervals/COMID learners are active, running in parallel. Because the metric being learned is changing with time, learners designed for low regret at different scales (drift rates) will have different performance (analogous to the classical bias/variance tradeoff). In other words, there is a scale I_{opt} optimal at a given time.

To adaptively select and fuse the outputs of the ensemble, we introduce Online Convex Ensemble StrongLy Adaptive Dynamic Learning (OCELAD), a method that accepts an ensemble of black-box learners and uses recent history to select the optimal one at each time.

IV. OCELAD

To maintain generality, in this section we assume the series of random loss functions of the form $\ell_t(\theta_t)$ where θ_t is the time-varying unknown parameters. We assume that an ensemble $\mathcal{B}$ of online learners is provided on the dyadic interval set $\mathcal{I}$, each optimized for the appropriate scale. To select the appropriate scale, we compute weights $w_t(I)$ that are updated based on the learner's recent estimated regret. The weight update we use is inspired by the multiplicative weight (MW) literature [22], modified to allow for unbounded loss functions. At each step, we rescale the observed losses so they lie between -1 and 1, allowing for maximal selection ability

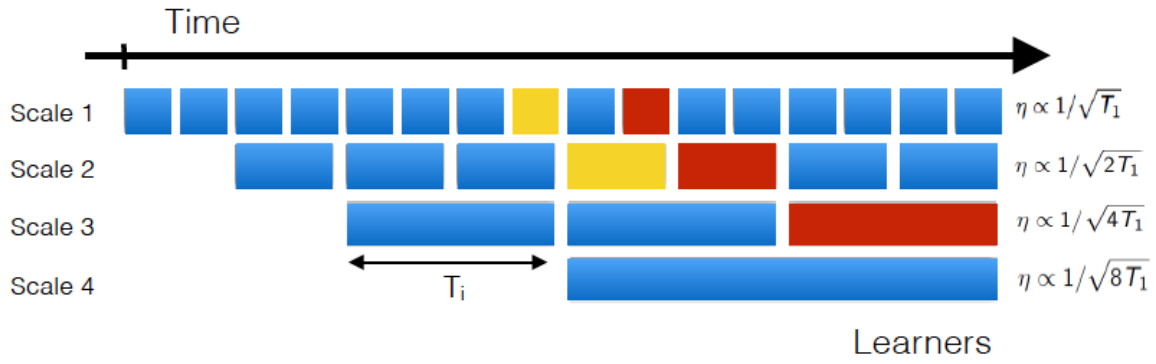


Fig. 2. Retro-initialized COMID ensemble (RICE). COMID learners at multiple scales run in parallel. Recent observed losses for each learner are used to create weights used to select the appropriate scale at each time. Each yellow and red learner is initialized by the output of the previous learner of the same color, that is, the learner of the next shorter scale.

and preventing negative weights.

$$r_t(I) = \left(\sum_I \frac{w_t(I)}{\sum_I w_t(I)} \ell_t(\theta_t(I))\right) - \ell_t(\theta_t(I)) \tag{5}$$
$$w_{t+1}(I) = w_t(I)\left(1 + \eta_I \frac{r_t(I)}{\max_{I\in \mathrm{ACT}(t)} |r_t(I)|}\right), \quad \forall t \in I.$$

These hold for all $I \in \mathcal{I}$, where $\eta_I = \min\{1/2, 1/\sqrt{|I|}\}$, $\mathbf{M}_t(I), \mu_t(I)$ are the outputs at time t of the learner on interval I, and $r_t(I)$ is called the estimated regret of the learner on interval I at time t. The initial value of $w(I)$ is η_I. Essentially, this is highly weighting low loss learners and lowly weighting high loss learners.

For any given time t, the outputs of the learners of interval $I \in \mathrm{ACT}(t)$ are combined to form the weighted ensemble estimate

$$\hat{\theta}_t = \frac{\sum_{I\in\mathrm{ACT}(t)} w_t(I)\theta_t(I)}{\sum_{I\in\mathrm{ACT}(t)} w_t(I)} \tag{6}$$

The weighted average of the ensemble is reasonable here due to our use of a convex loss function (proven in the next section), as opposed to the possibly non-convex losses of [22], necessitating a randomized selection approach. OCELAD is summarized in Algorithm 1, and the joint RICE-OCELAD approach as applied to metric learning of $\{\mathbf{M}_t, \mu_t\}$ is shown in Algorithm 2.

Algorithm 1 Online Convex Ensemble Strongly Adaptive Dynamic Learning (OCELAD)

Provide dyadic ensemble of online learners $\mathcal{B}$.
Initialize weight: $w_1(I)$.
for $t = 1$ to T **do**
  Observe loss function $\ell_t(\cdot)$ and update $\mathcal{B}$ ensemble.
  Obtain $|\mathrm{ACT}(t)|$ estimates $\theta_t(I)$ from the $\mathcal{B}$ ensemble.
  Compute weighted ensemble average $\hat{\theta}_t$ via (6) and set as estimate.
  Update weights $w_{t+1}(I)$ via (5).
end for
Return $\{\hat{\theta}_t\}$.

Algorithm 2 RICE-OCELAD for Nonstationary Metric Learning

Initialize weight: $w_1(I)$
for $t = 1$ to T **do**
  Obtain constraint $(\mathbf{x}_t, \mathbf{z}_t, y_t)$, compute loss function $\ell_{t,c}(\mathbf{M}_t, \mu_t)$.
  Initialize new learner in RICE if needed. New learner at scale $j > 0$: initialize to the last estimate of learner at scale $j - 1$.
  COMID update $\mathbf{M}_t(I), \mu_t(I)$ using (17) for all active learners in RICE ensemble.
  Compute
$$\hat{\mathbf{M}}_t \leftarrow \frac{\sum_{I\in\mathrm{ACT}(t)} w_t(I)\mathbf{M}_t(I)}{\sum_{I\in\mathrm{ACT}(t)} w_t(I)}$$
$$\hat{\mu}_t \leftarrow \frac{\sum_{I\in\mathrm{ACT}(t)} w_t(I)\mu_t(I)}{\sum_{I\in\mathrm{ACT}(t)} w_t(I)}.$$
  for $I \in \mathrm{ACT}(t)$ **do**
    Compute estimated regret $r_t(I)$ and update weights according to (5) with $\theta_t(I) = \{\mathbf{M}_t(I), \mu_t(I)\}$.
  end for
end for
Return $\{\hat{\mathbf{M}}_t, \hat{\mu}_t\}$.

V. Strongly Adaptive Dynamic Regret

The standard static regret is defined as

$$R_{\mathcal{B},static}(I) = \sum_{t\in I} f_t(\hat{\theta}_t) - \min_{\theta\in\Theta} \sum_{t\in I} f_t(\theta). \tag{7}$$

where $f_t(\theta_t)$ is a loss with parameter θ_t. Since in our case the optimal parameter value θ_t is changing, the static regret of an algorithm $\mathcal{B}$ on an interval I is not useful. Instead, let $\mathbf{w} = \{\theta_t\}_{t\in[0,T]}$ be an arbitrary sequence of parameters. Then, the *dynamic regret* of an algorithm $\mathcal{B}$ relative to a comparator sequence $\mathbf{w}$ on the interval I is defined as

$$R_{\mathcal{B},\mathbf{w}}(I) = \sum_{t\in I} f_t(\hat{\theta}_t) - \sum_{t\in I} f_t(\theta_t), \tag{8}$$

where $\hat{\theta}_t$ are generated by $\mathcal{B}$. This allows for a dynamically changing estimate.

In [21] the authors derive dynamic regret bounds that hold over all possible sequences $\mathbf{w}$ such that $\sum_{t\in I}\|\theta_{t+1}-\theta_t\|\leq\gamma$, i.e. bounding the total amount of variation in the estimated parameter. Without this temporal regularization, minimizing the loss would cause θ_t to grossly overfit. In this sense, setting the comparator sequence $\mathbf{w}$ to the "ground truth sequence" or "batch optimal sequence" both provide meaningful intuitive bounds.

Strongly adaptive regret bounds [15] have claimed that static regret is low on every subinterval, instead of only low in the aggregate. We use the notion of dynamic regret to introduce strongly adaptive dynamic regret bounds, proving that *dynamic regret is low on every subinterval $I\subseteq[0,T]$ simultaneously*. In a later work, we prove the following. Suppose there are a sequence of random loss functions $\ell_t(\theta_t)$. The goal is to estimate a sequence $\hat{\theta}_t$ that minimizes the dynamic regret.

Theorem 1. *Let $\mathbf{w}=\{\theta_1,\ldots,\theta_T\}$ be an arbitrary sequence of parameters and define $\gamma_{\mathbf{w}}(I)=\sum_{q\leq t<s}\|\theta_{t+1}-\theta_t\|$ as a function of $\mathbf{w}$ and an interval $I=[q,s]$. Choose an ensemble of learners $\mathcal{B}$ such that given an interval I the learner $\mathcal{B}_I$ creates an output sequence $\theta_t(I)$ satisfying the dynamic regret bound*

$$R_{\mathcal{B}_I,\mathbf{w}}(I)\leq C(1+\gamma_{\mathbf{w}}(I))\sqrt{|I|} \tag{9}$$

for some constant $C>0$. Then the strongly adaptive dynamic learner $OCELAD^{\mathcal{B}}$ using $\mathcal{B}$ as the ensemble creates an estimation sequence $\hat{\theta}_t$ satisfying

$$R_{OCELAD^{\mathcal{B}},\mathbf{w}}(I)\leq 8C(1+\gamma_{\mathbf{w}}(I))\sqrt{|I|}+40\log(s+1)\sqrt{|I|}$$

on every interval $I=[q,s]\subseteq[0,T]$.

In a dynamic setting, bounds of this type are particularly desirable because they allow for changing *drift rate* and guarantee quick recovery from *discrete changes*. For instance, suppose K discrete switches (large parameter changes or changes in drift rate) occur at times t_i satisfying $0=t_0<t_1<\cdots<t_K=T$. Then since $\sum_{i=1}^{K}\sqrt{|t_{i-1}-t_i|}\leq\sqrt{KT}$, this implies that the total expected dynamic regret on $[0,T]$ remains low ($O(\sqrt{KT})$), while simultaneously guaranteeing that an appropriate learning rate is achieved on each subinterval $[t_i,t_{i+1}]$.

Now, reconsider the dynamic metric learning problem of Section II. It is reasonable to assume that the transformed distance between any two points is bounded, implying $\|\mathbf{M}\|\leq c'$ and that $\ell_t(\mathbf{M}_t,\mu_t)\leq k=\ell(c'\max_t\|\mathbf{x}_t-\mathbf{z}_t\|_2^2)$. Thus the loss (and the gradient) are bounded. We can then show the COMID learners in the RICE ensemble have low dynamic regret. The proof of the following result is omitted for lack of space, and derives from a result in [21].

Corollary 1 (Dynamic Regret: Metric Learning COMID)**.** *Let the sequence $\hat{\mathbf{M}}_t,\hat{\mu}_t$ be generated by* (17)*, and let $\mathbf{w}=\{\mathbf{M}_t\}_{t=1}^{T}$ be an arbitrary sequence with $\|\mathbf{M}_t\|\leq c$. Then using $\eta_{t+1}\leq\eta_t$ gives*

$$R_{\mathbf{w}}([0,T])\leq\frac{D_{max}}{\eta_{T+1}}+\frac{4\phi_{max}}{\eta_T}\gamma+\frac{G_\ell^2}{2\sigma}\sum_{t=1}^{T}\eta_t \tag{10}$$

and setting $\eta_t=\eta_0/\sqrt{T}$,

$$\begin{aligned}&R_{\mathbf{w}}([0,T]) &(11)\\ &\leq\sqrt{T}\left(\frac{D_{max}+4\phi_{max}(\sum_t\|\mathbf{M}_{t+1}-\mathbf{M}_t\|_F)}{\eta_0}+\frac{\eta_0G_\ell^2}{2\sigma}\right)\\ &=O\left(\sqrt{T}\left[1+\sum_{t=1}^{T}\|\mathbf{M}_{t+1}-\mathbf{M}_t\|_F\right]\right). &(12)\end{aligned}$$

Since the COMID learners have low dynamic regret, we can use OCELAD on the RICE ensemble.

Theorem 2 (RICE-OCELAD Strongly Adaptive Dynamic Regret)**.** *Let $\mathbf{w}=\{\mathbf{M}_t\}_{t\in[0,T]}$ be any sequence of metrics with $\|\mathbf{M}_t\|\leq c$ on the interval $[0,T]$, and define $\gamma_{\mathbf{w}}(I)=\sum_{t\in I}\|\mathbf{M}_{t+1}-\mathbf{M}_t\|$. Let $\mathcal{B}$ be the RICE ensemble with $\eta_t(I)=\eta_0/\sqrt{|I|}$. Then the RICE-OCELAD metric learning algorithm (Algorithm 2) satisfies*

$$\begin{aligned}&R_{OCELAD,\mathbf{w}}(I)\leq &(13)\\ &\frac{4}{2^{1/2}-1}C(1+\gamma_{\mathbf{w}}(I))\sqrt{|I|}+40\log(s+1)\sqrt{|I|},\end{aligned}$$

for every subinterval $I=[q,s]\subseteq[0,T]$ simultaneously. C is a constant, and the expectation is with respect to the random output of the algorithm.

VI. Results

A. Synthetic Data

We run our metric learning algorithms on a synthetic dataset undergoing different types of simulated metric drift. We create a synthetic 2000 point dataset with 2 independent 50-20-30% clusterings (A and B) in disjoint 3-dimensional subspaces of $\mathbb{R}^{25}$. The clusterings are formed as 3-D Gaussian blobs, and the remaining 19-dimensional subspace is filled with iid Gaussian noise.

We create a scenario exhibiting nonstationary drift, combining continuous drifts and shifts between the two clusterings (A and B). To simulate continuous drift, at each time step we perform a small random rotation of the dataset. The drift profile is shown in 3. For the first interval, partition A is used and the dataset is static, no drift occurs. Then, the partition is changed to B, followed by an interval of first moderate, then fast, and then moderate drift. Finally, the partition reverts back to A, followed by slow drift.

We generate a series of T constraints from random pairs of points in the dataset, incorporating the simulated drift, running each experiment with 3000 random trials. For each experiment conducted in this section, we evaluate performance using two metrics. We plot the K-nearest neighbor error rate, using the learned embedding at each time point, averaging over all trials. We quantify the clustering performance by plotting the empirical probability that the normalized mutual information (NMI) of the K-means clustering of the unlabeled data points in the learned embedding at each time point exceeds 0.8 (out of a possible 1). We believe clustering NMI, rather than k-NN performance, is a more realistic indicator of metric learning performance, at least in the case where finding a relevant embedding is the primary goal.

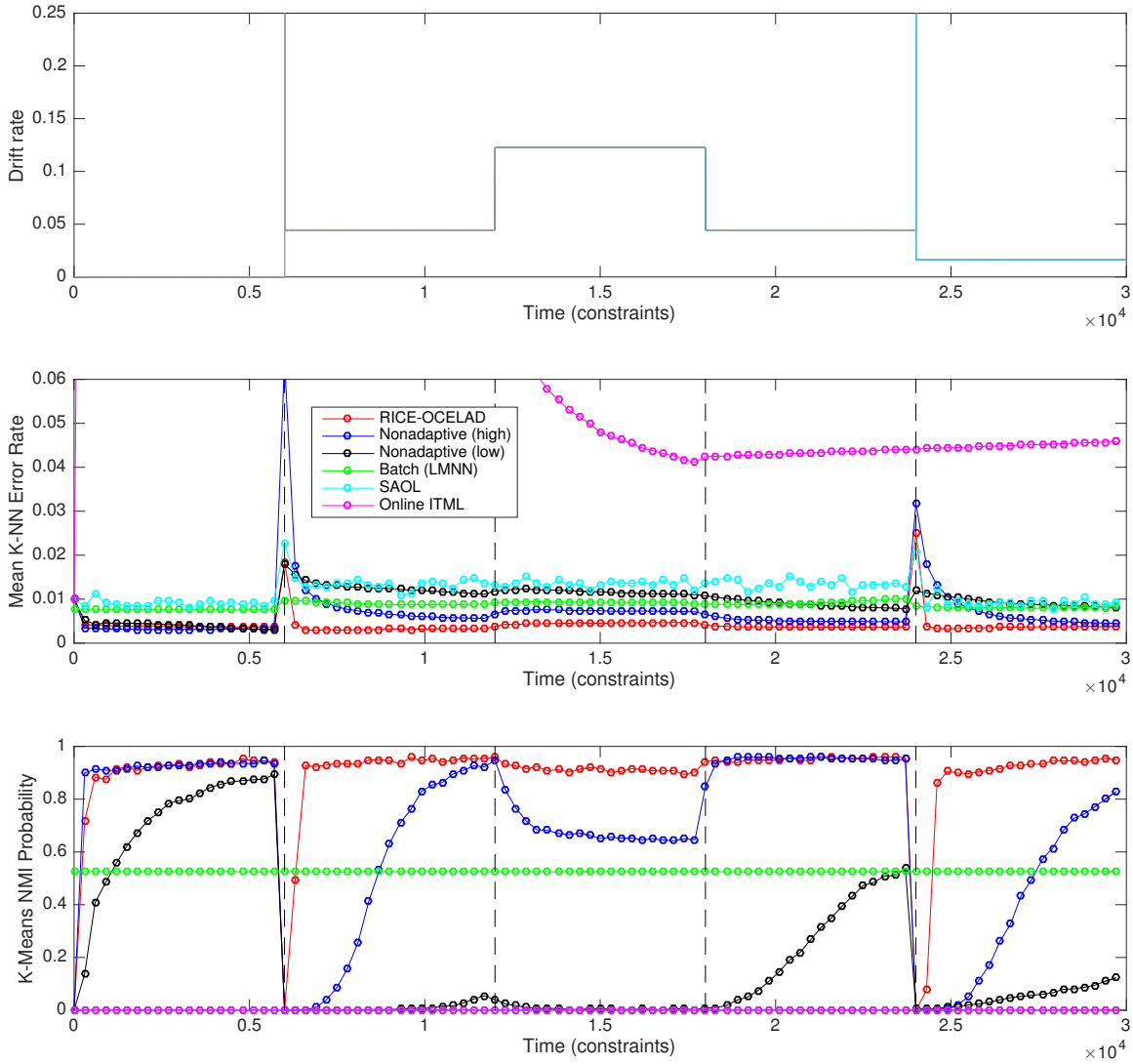
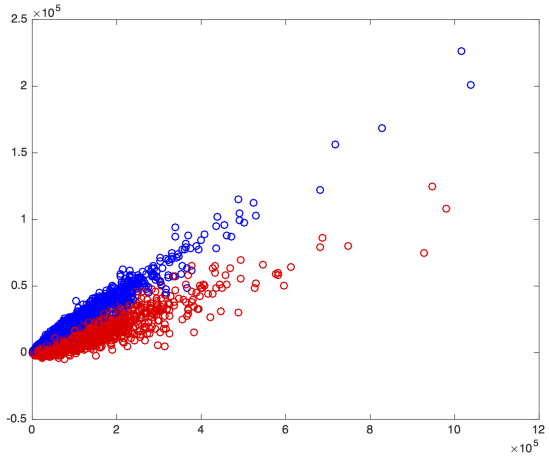


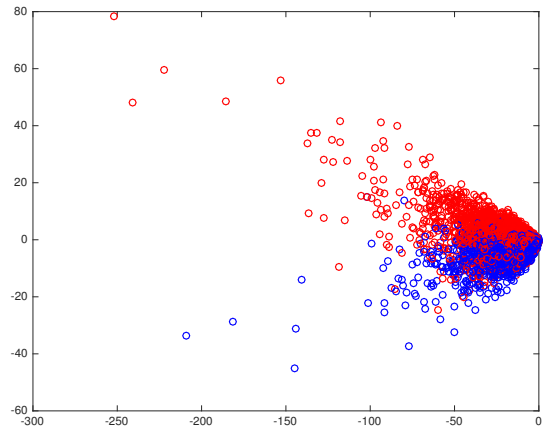
Fig. 3. Tracking of a changing metric. Top: Rate of change (scaled Frobenius norm per tick) of the generating metric as a function of time. The large changes result from a change in clustering labels. Metric tracking performance is computed for RICE-OCELAD (adaptive), nonadaptive COMID [5] (high learning rate), nonadaptive COMID (low learning rate), the batch solution (LMNN) [7], SAOL [15] and online ITML [10], averaged over 3000 random trials. Shown as a function of time is the mean k-NN error rate (middle) and the probability that the k-means NMI exceeds 0.8 (bottom). Note that RICE-OCELAD alone is able to effectively adapt to the variety of discrete changes and changes in drift rate, and that for NMI ITML and SAOL fail completely.

In our results, we consider RICE-OCELAD, SAOL with COMID [15], nonadaptive COMID [5], LMNN (batch) [7], and online ITML [10].

For RICE-OCELAD, we set the base interval length $I_0 = 1$ throughout, and set η_0 via cross-validation in a scenario with no drift. All parameters for the other algorithms were set via cross validation, so as to err on the side of optimism in a truly online scenario. For nonadaptive COMID, we set the high learning rate using cross validation for moderate drift, and we set the low learning rate via cross validation in the case of no drift. The results are shown in Figure 3. Online ITML fails due to its bias agains low-rank solutions [10], and the batch method and low learning rate COMID fail due to an inability to adapt. The high learning rate COMID does well at first, but as it is optimized for slow drift it cannot adapt to the changes in drift rate as well or recover quickly from the two partition changes. SAOL, as it is designed for mildly-varying bounded loss functions without slow drift and does not use retro-initialized learners, completely fails in this setting (zero probability of NMI ¿ .8 throughout). RICE-OCELAD, on the other hand, adapts well throughout the entire interval, as predicted by the theory.

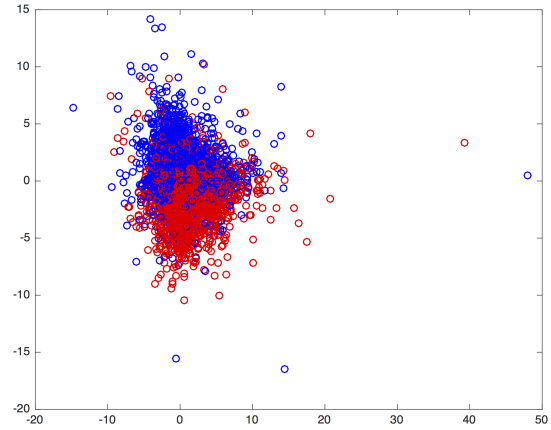


(a) OCELAD

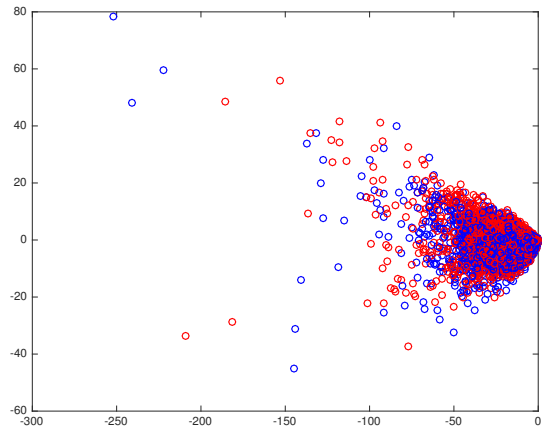


(b) PCA

Fig. 4. Metric learning for product type clustering. Book reviews blue, electronics reviews red. Original LOO k-NN error rate 15.3%. Top: First two dimensions of learned RICE-OCELAD embedding (LOO k-NN error rate 11.3%). Bottom: embedding from PCA (k-NN error 20.4%). Note improved separation of the clusters using RICE-OCELAD (cleaner border).



(a) OCELAD



(b) PCA

Fig. 5. Metric learning for sentiment clustering. Positive reviews blue, negative red. Original LOO k-NN error rate 35.7%. Top: First two dimensions of learned RICE-OCELAD embedding (LOO k-NN error rate 23.5%). Bottom: embedding from PCA (k-NN error 41.9%). Note improved separation of the clusters using RICE-OCELAD.

B. Clustering Product Reviews

As an example real data task, we consider clustering Amazon text reviews, using the Multi-Domain Sentiment Dataset [23]. We use the 11402 reviews from the Electronics and Books categories, and preprocess the data by computing word counts for each review and 2369 commonly occurring words, thus creating 11402 data points in $\mathbb{R}^{2369}$. Two possible clusterings of the reviews are considered: product category (books or electronics) and sentiment (positive: star rating 4/5 or greater, or negative: 2/5 or less).

Figures 4 and 5 show the first two dimensions of the embeddings learned by static COMID for the category and sentiment clusterings respectively. Also shown are the 2-dimensional standard PCA embeddings, and the k-NN classification performance both before embedding and in each embeddings. As expected, metric learning is able to find embeddings with improved class separability. We emphasize that while improvements in k-NN classification are observed, we use k-NN merely as a way to quantify the separability of the classes in the learned embeddings. In these experiments, we set the regularizer $r(\cdot)$ to the elementwise L1 norm to encourage sparse features.

We then conducted drift experiments where the clustering changes. The change happens after the metric learner for the original clustering has converged, hence the nonadaptive learning rate is effectively zero. For each change, we show the k-NN error rate in the learned RICE-OCELAD embedding as it adapts to the new clustering. Emphasizing the visualization and computational advantages of a low-dimensional embedding, we computed the k-NN error after projecting the data into the first 5 dimensions of the embedding. Also shown are the results for a learner where an oracle allows reinitialization of the metric to the identity at time zero, and the nonadaptive learner for which the learning rate is not increased. Figure 6

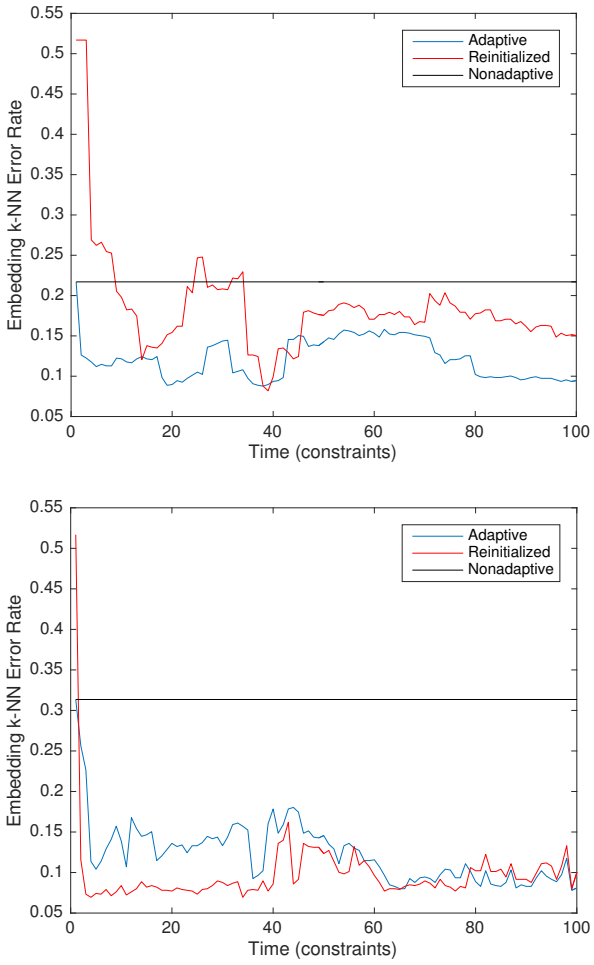


Fig. 6. Metric drift in Amazon review data. Left: Change from product type + sentiment clustering to simply product type; Right: Change from sentiment to product type clustering. The proposed OCELAD adapts to changes, tracking the clusters as they evolve. The oracle reinitialized mirror descent method (COMID) learner has higher tracking error and the nonadaptive learner (straight line) does not track the changes at all.

(left) shows the results when the clustering changes from the four class sentiment + type partition to the two class product type only partition, and Figure 6 (right) shows the results when the partition changes from sentiment to product type. In the first case, the similar clustering allows RICE-OCELAD to significantly outperform even the reinitialized method, and in the second remain competitive where the clusterings are unrelated.

VII. Conclusion and Future Work

Learning a metric on a complex dataset enables both unsupervised methods and/or a user to home in on the problem of interest while de-emphasizing extraneous information. When the problem of interest or the data distribution is nonstationary, however, the optimal metric can be time-varying. We considered the problem of tracking a nonstationary metric and presented an efficient, strongly adaptive online algorithm (OCELAD), that combines the outputs of any black box learning ensemble (such as RICE), and has strong theoretical regret guarantees. Performance of our algorithm was evaluated both on synthetic and real datasets, demonstrating its ability to learn and adapt quickly in the presence of changes both in the clustering of interest and in the underlying data distribution.

Potential directions for future work include the learning of more expressive metrics beyond the Mahalanobis metric, the incorporation of unlabeled data points in a semi-supervised learning framework [24], and the incorporation of an active learning framework to select which pairs of data points to obtain labels for at any given time [25].

References

[1] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[2] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin, "The elements of statistical learning: data mining, inference and prediction," *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2005.

[3] J. A. Lee and M. Verleysen, *Nonlinear dimensionality reduction*. Springer Science & Business Media, 2007.

[4] B. Kulis, "Metric learning: A survey." *Foundations and Trends in Machine Learning*, vol. 5, no. 4, pp. 287–364, 2012.

[5] G. Kunapuli and J. Shavlik, "Mirror descent for metric learning: a unified approach," in *Machine Learning and Knowledge Discovery in Databases*. Springer, 2012, pp. 859–874.

[6] E. P. Xing, M. I. Jordan, S. Russell, and A. Y. Ng, "Distance metric learning with application to clustering with side-information," in *Advances in Neural Information Processing Systems*, 2002, pp. 505–512.

[7] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Advances in Neural Information Processing System*, 2005, pp. 1473–1480.

[8] K. Q. Weinberger and L. K. Saul, "Fast solvers and efficient implementations for distance metric learning," in *ICML*, 2008, pp. 1160–1167.

[9] J. Goldberger, G. E. Hinton, S. T. Roweis, and R. Salakhutdinov, "Neighbourhood components analysis," in *Advances in neural information processing systems*, 2004, pp. 513–520.

[10] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, "Information-theoretic metric learning," in *ICML*, 2007, pp. 209–216.

[11] A. Bellet, A. Habrard, and M. Sebban, "A survey on metric learning for feature vectors and structured data," *arXiv preprint arXiv:1306.6709*, 2013.

[12] L. Yang and R. Jin, "Distance metric learning: A comprehensive survey," *Michigan State Universiy*, vol. 2, 2006.

[13] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge University Press, 2006.

[14] J. C. Duchi, S. Shalev-Shwartz, Y. Singer, and A. Tewari, "Composite objective mirror descent," in *COLT*. Citeseer, 2010, pp. 14–26.

[15] A. Daniely, A. Gonen, and S. Shalev-Shwartz, "Strongly adaptive online learning," *ICML*, 2015.

[16] H. B. McMahan and M. Streeter, "Adaptive bound optimization for online convex optimization," in *COLT*, 2010.

[17] J. C. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," in *COLT*, 2010.

[18] H. B. McMahan, "Analysis techniques for adaptive online learning," *arXiv preprint arXiv:1403.3465*, 2014.

[19] M. Herbster and M. K. Warmuth, "Tracking the best expert," *Machine Learning*, vol. 32, no. 2, pp. 151–178, 1998.

[20] E. Hazan and C. Seshadhri, "Adaptive algorithms for online decision problems," in *Electronic Colloquium on Computational Complexity (ECCC)*, vol. 14, no. 088, 2007.

[21] E. Hall and R. Willett, "Online convex optimization in dynamic environments," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 9, no. 4, pp. 647–662, June 2015.

[22] A. Blum and Y. Mansour, "From external to internal regret," in *Learning theory*. Springer, 2005, pp. 621–636.

[23] J. Blitzer, M. Dredze, F. Pereira *et al.*, "Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification," in *ACL*, vol. 7, 2007, pp. 440–447.

[24] M. Bilenko, S. Basu, and R. J. Mooney, "Integrating constraints and metric learning in semi-supervised clustering," in *ICML*, 2004, p. 11.

[25] B. Settles, "Active learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 6, no. 1, pp. 1–114, 2012.