A LOCALLY ADAPTIVE NORMAL DISTRIBUTION IN SEMI-SUPERVISED LEARNING

Ioannis Kavadakis, Ramiro Mata and Ola Rønning

ABSTRACT

Recently the locally adaptive normal distribution (LAND) has been put forward [1], which has the property of fitting nonlinear data structures. Arvanitidis et al. explore its usage in the unsupervised setting. Here we develop a LAND mixture model (LMM) in the semi-supervised setting where initialization of the LAND's parameters (μ, Σ, π) is based solely on labeled data. We explore its behavior under three different labeled data regimes: (1) one label placed at the end of each class, (2) 10% evenly distributed labeled data, and (3) uneven label percentage in each class. We find that the LMM's convergence in the semi-supervised setting is sensitive to initial starting position. Further, we compare the LMM to the Gaussian Mixture Model (GMM) by fitting both to nonlinear synthetic data.

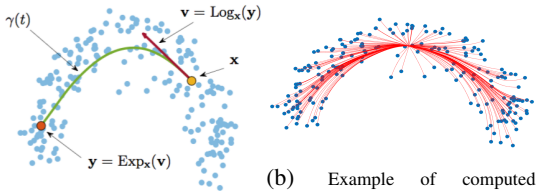
Index Terms— LAND, semi-supervised learning, gaussian mixture model, riemannian manifold

1. INTRODUCTION

Often data can lie near an underlying smooth, nonlinear structure (known as the Manifold Assumption). Nonlinear data can be difficult to represent with a normal distribution due to the linear distance measure used (i.e. Mahalanobis distance). LAND, which can be considered a generalization of the normal distribution, solves this problem by constructing a distribution that 'adjusts' to the nonlinearity of the data. Extending this method to the semi-supervised setting can be important in real world applications as labels can be expensive. Further, it can prove useful in certain machine learning applications by allowing to integrate unlabeled data in supervised learning tasks. In this paper we briefly introduce the LAND (for more extensive description of LAND see [1]), extend it to the semi-supervised setting, and present findings on its behavior under different labeling schemes.

2. NECESSARY NOTIONS OF RIEMANNIAN GEOMETRY

We begin with an exposition of Riemannian geometry to introduce necessary notions for understanding LAND. First, we denote the Riemannian manifold learned from the data as $\mathcal{M}$, its tangent space at a data point ($\mathbf{x} \in \mathcal{M}$) as $\mathcal{T}_{\mathbf{x}}\mathcal{M}$, and the Riemannian metric as $\mathbf{M}$. The metric is a smoothly changing inner product on the tangent space of each data point within the learned manifold.



(a) Exponential/Logarithm map

(b) Example of computed geodesics

Fig. 1

To measure the distance between two points on the manifold we use *geodesics*. A geodesic can be thought of as the curve or path with "least resistance" (as defined by the metric) that connects two points within the manifold, and which can be expressed as:

$$\gamma = \mathrm{argmin}_{\gamma} \int_0^1 \sqrt{\langle \gamma'(t), \mathbf{M}(\gamma(t)\gamma'(t)) \rangle} dt,$$

and can be found by solving the 2nd order ODE:

$$\gamma''(t) = -\frac{1}{2}\mathbf{M}^{-1}(\gamma(t)) \left[\frac{\partial vec[\mathbf{M}(\gamma(t))]}{\partial \gamma(t)}\right]^{\top} (\gamma'(t) \otimes \gamma'(t))\text{"[1]}$$

Finally to map between the tangent space and the manifold we use the *exponential map* and *logarithm map*. The exponential map is the mapping of a vector $\mathbf{v}$ on $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ at a point $\mathbf{x}$ to the uniquely defined point $\mathbf{y}$. The logarithm map is the inverse mapping. $\|\mathrm{Log}_{\mathbf{x}}(\mathbf{y})\|$ is equal to the geodesic distance between the two points, see Figure 1a.

3. SUMMARY OF CONSTRUCTING LAND

The central idea of the LAND is to substitute the mahalanobis distance found in the exponent of the normal distribution with its nonlinear extension in the Riemannian manifold, such that

$$p_{\mathcal{M}}(\mathbf{x}|\mu, \Sigma) = \frac{1}{\mathcal{C}} \exp\left(-\frac{1}{2}\langle \mathrm{Log}_{\mu}(\mathbf{x}), \mathbf{\Sigma}^{-1}\mathrm{Log}_{\mu}(\mathbf{x}) \rangle\right), \quad \mathbf{x} \in \mathcal{M}.$$

The procedure to compute the parameters of the distribution above is the following[1]:

1. Initialize μ, by finding the *karcher mean*[2] of the data points, due to its property (unlike the intrinsic least squares mean) that it favors high density regions.

2. Compute geodesics numerically as shown above.

3. Estimate the normalization constant $\mathcal{C}$, which by definition is

$$\mathcal{C}(\mu, \mathbf{\Sigma}) = \int_{\mathcal{M}} \exp\left(-\frac{1}{2}\langle \mathrm{Log}_{\mu}(\mathbf{x}), \mathbf{\Sigma}^{-1}\mathrm{Log}_{\mu}(\mathbf{x})\rangle\right) \mathrm{d}\mathcal{M}(\mathbf{x})$$

We evaluate this integral by first doing integration substitution to the tangent space, and then estimating it numerically using *Monte Carlo integration*.

4. Estimate distribution parameters $\{\hat{\mu}, \hat{\mathbf{\Sigma}}\}$ by minimizing the negative log-likelihood of the following objective function:

$$\phi(\mu, \mathbf{\Sigma}) = \frac{1}{2N}\sum_{n}^{N}\langle \mathrm{Log}_{\mu}(x_n), \Sigma^{-1}\mathrm{Log}_{\mu}(x_n)\rangle + \log(\mathcal{C}(\mu, \Sigma)).$$

4. EXTENDING THE GAUSSIAN MIXTURE MODEL TO THE SEMI-SUPERVISED SETTING

For clarity, we first explain the extension to the semi-supervised setting with the more common and familiar gaussian distribution in the context of mixture models. Then, we use this exposition to transition to the less familiar LAND in the LMM semi-supervised case.

We can compute a gaussian mixture model in the unsupervised setting by maximizing the gaussian objective function (GOF), shown below, with the EM-algorithm [3].

$$\sum_{k}^{K}\sum_{n}^{N}\gamma_{kn}\log\left[\pi_k|\Sigma_k|^{-1/2}e^{-1/2(x_n-\mu_k)^T\Sigma_k^{-1}(x_n-\mu_k)}\right]$$

where γ_{kn} gives the responsibility of the n'th datum to the k'th component of the mixture.

$$\gamma_{kn} = \begin{cases} I_k(t_n) & \text{if } t_n \neq 0 \\ \frac{\pi_k p(x_n|\mu_k,\Sigma_k)}{\sum_{kk}\pi_{kk}p(x_n|\mu_{kk},\Sigma_{kk})} & \text{otherwise} \end{cases}$$

We can extend GOF to the semi-supervised setting by assigning labels t_n for all data points, such that t_n denotes the component the n'th datum belongs to. For unlabeled data t_n is zero.

Using the normal distribution allows us to derive closed form solutions when updating θ in the maximization step. This is computationally attractive in that updates are linear with the data size, and in that we obtain linear convergence [4] without tuning hyper-parameters.

5. EXTENDING THE LAND MIXTURE MODEL TO THE SEMI-SUPERVISED SETTING

Similar to the gaussian case, LAND can be extended to a mixture model in the unsupervised setting by maximizing the LAND objective function (LOF), shown below, using the EM-algorithm[1].

$$\sum_{k}^{K}\sum_{n}^{N}\gamma_{kn}\left[\log(\mathcal{C}(\mu_k,\Sigma_k)) + \frac{1}{2}\langle \mathrm{Log}_{\mu k}(x_n), \Sigma_k^{-1}\mathrm{Log}_{\mu k}(x_n)\rangle - \log(\pi_k)\right]$$

with γ_{kn} again denoting the responsibility of the n'th datum to the k'th component.

$$\gamma_{kn} = \begin{cases} I_k(t_n) & \text{if } t_n \neq 0 \\ \frac{\pi_k p_{\mathcal{M}}(x_n|\mu_k,\Sigma_k)}{\sum_{kk}\pi_{kk}p_{\mathcal{M}}(x_n|\mu_{kk},\Sigma_{kk})} & \text{otherwise} \end{cases}$$

We can extend the mixture model to the semi-supervised setting by introducing labels t_n for each data point, as discussed in Section 4.

Introducing labeled data in the expectation step, allows us to use the same methods of updating θ parameters in the maximization step, as used in the unsupervised setting. The numerical optimization is performed using block-coordinated descent, as closed form solutions for maximizing LOF with respect to θ parameters do not exist [1]. In block-coordinated descent we do the following:

1. maximize the LOF by updating the mean of each component, holding their corresponding co-variance fixed.

2. Then, maximize the LOF by updating the co-variances using fixed means computed in previous step.

We updated the means by walking in the direction (of unit length on the manifold) with the largest projection onto the steepest descent direction. Whilst the co-variance is updated by walking in the direction of steepest descent. In both cases we scale down the step length by a learning rate. The mean updates are a constraint to avoid "unstable gradients" [1].

6. DATA AND EXPERIMENTS

We use the same synthetic data sets as in Arvanitidis et al. [1]. The first data set, which we refer to as two moons, consists of 600 bivariate data points (300 for each class), see Figure 2. The second dataset, which we refer to as half moon, consists of 300 bivariate data points, see Figure 3.

The aim of the experiments was to understand the behavior of LMM in the semi-supervised learning setting. In all experiments of this study we initialize the θ parameters (in both LMM and GMM) based solely on the labeled data. In the first experiment our goal was to see how LMM behaves under a difficult scenario relative to a more general case. From the two moons data set we introduced only one label at the end (i.e. at the tail) of the data cloud for each class as shown in the **left** plot of Figure 2. We compare this "difficult scenario" labeling scheme with one where we introduce 10% evenly distributed labels in the same data set, see **right** plot in Figure 2. In the second experiment, our goal was to see whether we would get similar behavior on a data set with a different structure. Therefore, we replicated experiment 1 but on the half-moon data set. In the third experiment we wanted to compare the LMM with the GMM in a scenario where the LAND's nonlinear properties would presumably better fit the non-linearity in the data, and under a labeling scheme that

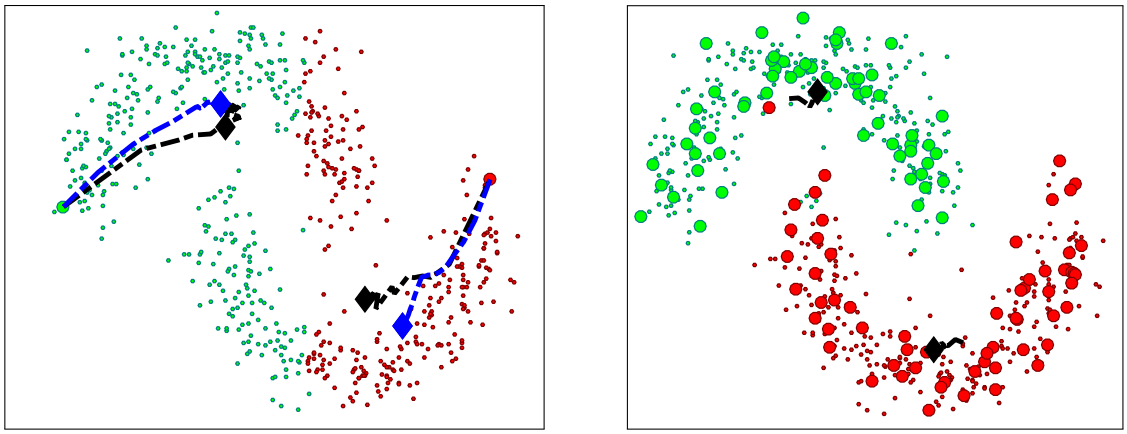


Fig. 2: The results of experiment 1 are shown above. **Left:** The case where only one label is provided on each half moon. Note that the blue dashed line and the blue diamond represent the convergence path when the step size is halved. **Right:** The case with 10% labels. The two different colors (green and red) of the data points represent the class they were assigned to by the LMM. The two larger circles represent the labeled data while the smaller circles represent the unlabeled data. The black dashed lines denotes the convergence path of the μ's and the black diamond represents the final estimated μ.

could present difficulties when using a linear metric, such as used in the GMM.

7. RESULTS

7.1. Experiment 1

We find that convergence of the LMM in the semi-supervised setting is sensitive to starting position. In the case where our starting position is based solely on the one label placed at the tail of each class, we observe that it converges closer to the "intrinsic" mean. By intrinsic mean, we refer to the mean that minimizes the variance of a distribution under Euclidean distance. For both the half-moon and two moons data sets considered here, the intrinsic means lie outside the data cloud. In contrast, when 10% evenly distributed labels are provided, the initial μ centroids for both classes start closer to an optimal μ center. In such cases, we observe that the LMM converges inside the data cloud to a centroid that is more representative of the underlying data structure, see **right** plot in Figure 2).

7.2. Experiment 2

In experiment 2 we repeat experiment 1 but with a different data set; everything else (e.g. labeling scheme, tuning parameters) remains the same. We acknowledge that there is no inherent or obvious classes in this data set. For the purposes of evaluating a mixture model, however, we regard the two classes to lie in the left and right side of the data, respectively, as denoted by the color scheme in Figure 3. We obtain different results when repeating experiment 1 on the half-moon data set. We observe that in this case we are able to converge close to an optimal mean (one which is more representative of the class distribution) despite starting far from an optimal μ center as shown in **top** plot in Figure 3.

7.3. Experiment 3

In experiment 3 we compare the LMM to the GMM on the two moons data set. We provide only one label in the upper moon (class 1: green points), and place it near the optimal μ. We do the same for the lower moon (class 2: red points), but we additionally provide two more labels on each tail. We observe that given the labeling scheme in the lower moon, the GMM overestimates the co-variance in the estimated distribution for class 2. This results in misclassifying a substantial amount of data points that in fact pertain to class 1, see **bottom** plot of Figure4). Further, we observe that the mean of class 2 converges outside the data cloud. In contrast, both means of the LMM converge on the edge of the data cloud closer to an optimal μ. Further, the converged LAND distributions are able to cope with the nonlinear data structure (despite the uneven labeling scheme) and thus provide a better representation of the data as shown in the **bottom** plot of Figure4.

8. DISCUSSION

In this paper we have extended the LMM to the semi-supervised setting. In real world semi-supervised applications it might be tempting to initialize parameters (θ in this case) solely on known labels. However, as experiments 1 and 2 show, respectively, LMM is sensitive to the initial starting estimate of θ and the underlying data structure. Thus, our main finding is that label distribution can influence the quality of the estimated mixture of LANDs. Secondly, that

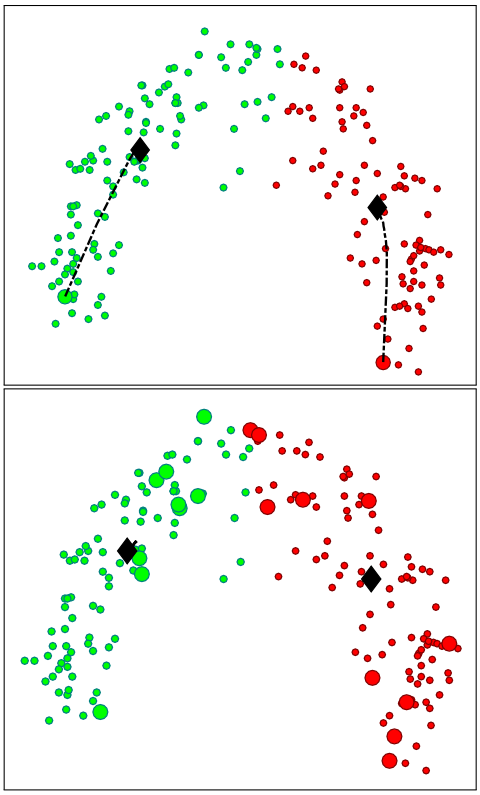


Fig. 3: The results of experiment 2 are shown above. **Top:** One label is provided per class. **Bottom:** 10% labels provided per class. The colors (red and green) represent the assigned class of data points by the LMM. Note how under a simpler data structure the LMM is still able to converge to a point close to an optimal μ despite starting far away from the solution.

this influence is not invariant to data structure; that is, the influence might be attenuated under a simpler data structure (as shown in experiment 2) or amplified under more complex data structures. These findings have important implications in regards to optimization heuristics in the semi-supervised LMM model.

First, initialization far from the true mean can lead to getting stuck in local minima or saddle points (as shown in experiment 1). We must remark that when we halved the step-length in the one-label scenario in experiment 1, the LMM was able to converge closer to an optimal μ solution as shown in the blue dashed line in the **right** plot of Figure 2. This suggests that adjusting step-length according to label distribution and data structure complexity can be beneficial; however, further studies are required to reliably determine appropriate step size in the context of this remark. Second, starting far from the true mean will likely result in more iterations under the current LAND implementation. Computing the geodesics is the most computationally expensive part of the LAND, and this has to be done in every iteration to update the means.

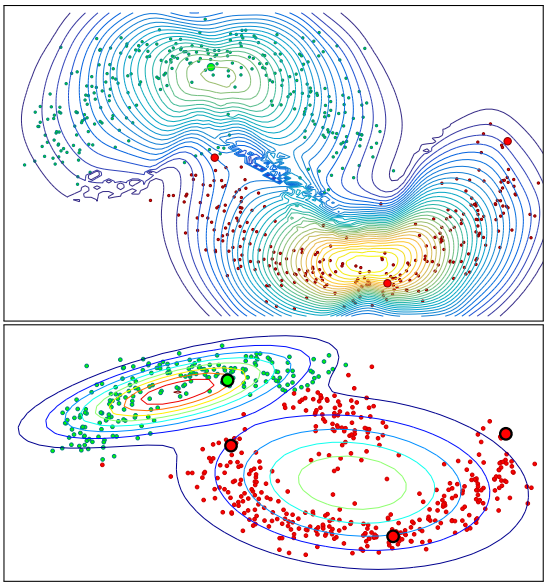


Fig. 4: The results of experiment 3 is shown above. **Top:** Contours represents probability mass of learned LMM. **Bottom:** Contours represent probability mass of learned GMM. The colors (red and green) represent the class assigned by the respective mixture model. Note that the GMM misclassifies a substantial fraction of class 1 (upper moon).

Therefore, minimizing the number of iterations is imperative to obtaining reasonable computational times in practice.

To this end, we explored possibilities of improving the optimization time of LMM in the semi-supervised setting. While in practice line search methods can be suitable and provide faster convergence times, under the current implementation of the LAND, these methods are not practical. The main reason is the expensive computation of the geodesics, which are necessary to evaluate the objective function. Line search methods compute the objective function (potentially) many times to find a suitable step length. This can be very beneficial in models where the computational burden of evaluating the objective function is relatively small. However, in the case of the LAND model, evaluating the objective function is computationally expensive, and thus line search methods become impractical.

To overcome these difficulties, a better heuristic method for LMM in the semi-supervised setting is to initialize θ based on all available data - whereby initialization is based on information stemming from the labeled and from the unlabeled data. This has to potential to attenuate the pitfalls of initializing based on labeled data only, and prevent, for instance, starting with an initial θ estimate that is far from the solution as can occur when the labeling distribution is skewed.

In conclusion, we presented experiments that shed light on the behavior of the LMM when extended to the semi-supervised setting. We found that optimization heuristics can

be critical in the quality of the estimated LANDs; and that under the θ initialization method considered in this study, the LMM is sensitive to the initial θ estimate (as determined by the labeled data). Finally, this sensitivity can potentially vary according to the complexity of the data structure.

9. REFERENCES

[1] Georgios Arvanitidis, Lars K Hansen, and Sø ren Hauberg, "A locally adaptive normal distribution," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds., pp. 4251–4259. Curran Associates, Inc., 2016.

[2] Hermann Karcher, "Riemannian center of mass and mollifier smoothing," *Communications on pure and applied mathematics*, vol. 30, no. 5, pp. 509–541, 1977.

[3] Christopher M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[4] Richard A. Redner and Homer F. Walker, "Mixture densities, maximum likelihood and the em algorithm," *SIAM Review*, vol. 26, no. 2, pp. 195–239, 1984.