Semiparametric Latent Variable Models for Guided Representation

Jasper Snoek, Ryan Prescott Adams and Hugo Larochelle
Department of Computer Science,
University of Toronto,
6 King's College Rd., Toronto, ON, M5S 3G4
{*jasper,rpa,larocheh*}*@cs.toronto.edu*

Unsupervised discovery of latent representations, in addition to being useful for density modeling, visualisation and exploratory data analysis, is also increasingly important for learning features relevant to discriminative tasks. Autoencoders, in particular, have proven to be an effective way to learn latent codes that reflect meaningful variations in data.

A continuing challenge, however, is *guiding* an autoencoder toward representations that are useful for a particular discriminative task. While autoencoders are effective at capturing the statistics of data, if the salient variations in the data distribution are not relevant to the desired discriminative task, then the features learned by the autoencoder will not improve performance. A complementary challenge is to find codes that are explicitly *invariant* to *irrelevant* transformations of the data.

To address these difficulties, we introduce the *semiparametric latent variable model* (SPLVM), which combines an autoencoder with a Gaussian process latent variable model. The SPLVM enables an autoencoder's unsupervised representation to both incorporate relevant label information and ignore irrelevant variations.

Autoencoder Neural Networks

Our starting point for the SPLVM is the autoencoder, a special type of artificial neural network that is trained to reproduce the input at its output. Denoting the latent space by $\mathcal{X}=\mathbb{R}^J$, the visible (input) space by $\mathcal{Y}=\mathbb{R}^K$, the encoder as a function $g(\boldsymbol{y}\,;\,\phi):\mathcal{Y}\rightarrow\mathcal{X}$ and the decoder as a function $f(\boldsymbol{x}\,;\,\psi):\mathcal{X}\rightarrow\mathcal{Y}$, training an autoencoder under least-squares reconstruction corresponds to the optimization:

$$\phi^\star,\psi^\star=\arg\min_{\phi,\psi}L_{\text{auto}}(\{\boldsymbol{y}^{(n)}\}_{n=1}^N,\phi,\psi),\qquad L_{\text{auto}}(\{\boldsymbol{y}^{(n)}\}_{n=1}^N,\phi,\psi)=\sum_{n=1}^N\sum_{k=1}^K(y_k^{(n)}-f_k(g(\boldsymbol{y}^{(n)};\phi);\psi))^2,\tag{1}$$

where $f_k(\cdot)$ refers to the kth output dimension of $f(\cdot)$ and $\{\boldsymbol{y}^{(n)}\}_{n=1}^N$, $\boldsymbol{y}^{(n)}\in\mathcal{Y}$ are the training examples.

Gaussian Process Latent Variable Models

As in the autoencoder, the GPLVM assumes that the N observed input data $\{\boldsymbol{y}^{(n)}\}_{n=1}^N$ are the image of a homologous set $\{\boldsymbol{x}^{(n)}\}_{n=1}^N$, arising from a vector-valued "decoder" function $f(\boldsymbol{x})$. Analogously to the squared-loss, the GPLVM assumes observed data corrupted by zero mean Gaussian noise: $\boldsymbol{y}^{(n)}=f(\boldsymbol{x}^{(n)})+\varepsilon$ with $\varepsilon\sim\mathcal{N}(0,\sigma^2\mathbb{I}_K)$. The innovation of the GPLVM is to place a Gaussian process prior on the function $f(\boldsymbol{x})$ and then optimize the latent representation $\{\boldsymbol{x}^{(n)}\}_{n=1}^N$, while marginalizing out the unknown $f(\boldsymbol{x})$. To preserve local distances from the observed space within the latent embedding, [2] subsequently reformulated the GPLVM with the constraint that the hidden representation be the result of a smooth mapping $g(\boldsymbol{y}\,;\,\phi)$ from the observed space, acting much like an encoder. The marginal likelihood objective of this *back-constrained* GPLVM is:

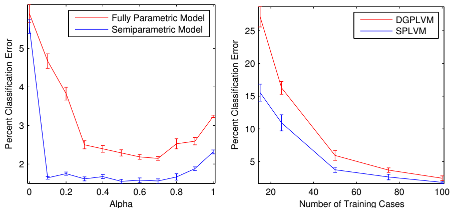
$$\phi^\star=\arg\min_{\phi}L_{\text{GP}}(\{\boldsymbol{y}^{(n)}\}_{n=1}^N,\phi),\text{ where }\;L_{\text{GP}}(\{\boldsymbol{y}^{(n)}\}_{n=1}^N,\phi)=\sum_{k=1}^K\ln|\boldsymbol{\Sigma}_{\theta_k,\phi}|+{\boldsymbol{y}_k^{(\cdot)}}^{\mathsf{T}}(\boldsymbol{\Sigma}_{\theta_k,\phi}+\sigma^2\mathbb{I}_N)^{-1}\boldsymbol{y}_k^{(\cdot)},\tag{2}$$
$$\text{and }[\boldsymbol{\Sigma}_{\theta_k,\phi}]_{n,n'}=C(g(\boldsymbol{y}^{(n)};\phi),g(\boldsymbol{y}^{(n')};\phi)\,;\,\theta_k).$$

The kth covariance matrix $\boldsymbol{\Sigma}_{\theta_k,\phi}$ depends on hyperparameters θ_k of kernel $C(\cdot,\cdot)$ and parameters ϕ of $g(\boldsymbol{y}\,;\,\phi)$.

GPLVM as an Infinite Autoencoder

Previous work [3] has established the relationship between Gaussian processes and artificial neural networks. One overlooked consequence of this relationship is that it also connects autoencoders and the back-constrained GPLVM. One can start from the autoencoder and notice that, for a linear decoder with squared-loss and zero-mean Gaussian prior over its weights, the decoder can be integrated out. Learning then corresponds to the minimization of Eqn. (2) with a linear kernel. Any non-degenerate positive definite kernel corresponds to a decoder of infinite size, and also recovers the general back-constrained GPLVM algorithm.

Oil Flow Dataset



(A) **(B)**

CIFAR-10 Dataset

Experiment	α	Accu.
1. Full Images	0.0	46.91
	0.1	56.75
2. 28x28 Windows	0.0	63.20
	0.8	65.71
3. Convolutional	0.0	71.48
	0.01	72.28

Small NORB Dataset

Model	Error
Autoencoder	10.97
Autoencoder + Logreg	10.44
SPLVM (class labels)	8.60
SPLVM (all labels)	7.89

Figure 1: **Left:** experiments on Oil Flow dataset. **(A)** compares the supervised guidance of SPLVM and the parametric guidance of [1] when varying the amount of supervised guidance α and **(B)** compares SPLVM with the discriminative GPLVM of [4] (state of the art on this dataset) when varying the training set size. **Middle:** CIFAR-10 accuracies of SPLVM, for different experimental setups. **Right:** Comparisons on Small NORB dataset, for which irrelevant labels (lighting,elevation,azimuth) are available.

Supervised Guiding of Latent Representations

When the salient variations in the input are only weakly informative about a particular discriminative task, it can be useful to incorporate label information into unsupervised learning. To this end, [1] proposed to add a parametric mapping $c(\boldsymbol{x}\,;\,\Lambda) : \mathcal{X} \rightarrow \mathcal{Z}$ (e.g. a logistic regressor) from the latent representation's space $\mathcal{X}$ to the label space $\mathcal{Z}$ and backpropagate error gradients from the output to the representation.

There are two disadvantages to this strategy. First, the assumption of a specific parametric form for the mapping $c(\boldsymbol{x}\,;\,\Lambda)$ restricts the guidance to classifiers within that family of mappings. The second and more fundamental problem is that the learned representation is further committed to one particular setting of the parameters Λ. Consider the learning dynamics of gradient descent optimization for this strategy. At every iteration t of descent (with current state $\phi_t, \psi_t, \Lambda_t$), the gradient from supervised guidance encourages the latent representation (currently parametrized by ϕ_t, ψ_t) to become more predictive of the labels under the current label map $c(\boldsymbol{x}\,;\,\Lambda_t)$. Such behavior discourages moves in ϕ, ψ space that make the latent representation more predictive under some other label map $c(\boldsymbol{x}\,;\,\Lambda^\star)$ where $\Lambda^\star$ is distant from Λ_t. Hence, while the problem would seem to be alleviated by the fact that Λ is learned jointly, this constant pressure towards representations that are immediately useful seems likely to increase the difficulty of representation learning.

Semiparametric Latent Variable Model

Rather than directly specifying a particular label mapping, we would prefer to find latent representations that are consistent with many such maps. One way to arrive at such a guidance mechanism is to marginalize out the parameters Λ of a label map $c(\boldsymbol{x}\,;\,\Lambda)$ under a distribution that permits a wide family of such functions. We have seen previously that this is specifically what GPLVM does for the decoder $f(\boldsymbol{x}\,;\,\psi)$. We follow the same reasoning and do this instead for $c(\boldsymbol{x}\,;\,\Lambda)$. The result is a hybrid of the autoencoder and a back-constrained GPLVM acting in $\mathcal{Z}$ space, where the encoder network $g(\boldsymbol{y}\,;\,\phi)$ is shared across models:

$$\phi^\star, \psi^\star = \arg\min_{\phi,\psi} (1-\alpha) L_{\text{auto}}(\{\boldsymbol{y}^{(n)}\}_{n=1}^N, \phi, \psi) + \alpha L_{\text{GP}}(\{\boldsymbol{z}^{(n)}\}_{n=1}^N, \phi) \tag{3}$$

where $\{\boldsymbol{z}^{(n)}\}_{n=1}^N$ are the available labels (for discrete labels, we use a "one-hot" encoding.). We call this approach to guided latent representation the *semiparametric latent variable model*, or SPLVM.

In experiments on the Oil, CIFAR and NORB datasets (see Figure 1), we evaluated how useful representations learned by SPLVM are under a logistic regressor. We observe that SPLVM can indeed improve the discriminative performance of autoencoder latent representations. SPLVM also improves over the guidance mechanism proposed by [1], confirming the usefulness of semiparametric guidance. Finally, on NORB, SPLVM is shown to even take avantage of irrelevant labels, i.e. labels informative of variations under which the class label should be invariant.

References

[1] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. In *NIPS*, pages 153–160, 2007.

[2] Neil D. Lawrence and Joaquin Quiñonero-Candela. Local distance preservation in the GP-LVM through back constraints. In *ICML*, 2006.

[3] Radford Neal. Bayesian learning for neural networks. *Lecture Notes in Statistics*, 118, 1996.

[4] Raquel Urtasun and Trevor Darrell. Discriminative Gaussian process latent variable model for classification. In *ICML*, 2007.