
Multitask Learning via Mixture of Linear Subspaces

Piyush Rai

School of Computing
University of Utah
piyush@cs.utah.edu

Hal Daumé III

Dept. of Computer Science
University of Maryland
hal@umiac.umd.edu

Abstract

We propose a probabilistic generative model for multitask learning that exploits the cluster structure of the task parameters, and *additionally* imposes a low-rank constraint on the set of task parameters within each cluster. This leads to a sharing of statistical strengths of multiple tasks at two levels: (1) via cluster assumption, and (2) via a subspace assumption within each cluster. Our work brings in the benefits of both these aspects of task relationship, each of which has been addressed only individually in prior work. We assume a mixture of linear subspaces model on the latent task parameters that can capture both these aspects simultaneously. Furthermore, the mixture of subspaces assumption can model the fact that the task parameters could potentially live on a non-linear manifold instead of a linear subspace which is a restriction of earlier work on multitask learning based on the linear subspace assumption.

1 Introduction

Multitask learning is an attractive learning paradigm as it allows different models to share their statistical strengths and learn even if each individual task has access to a very small number of labeled examples. Task relatedness is captured usually by having some kind of assumption on their relatedness; for example, by assuming that all task parameters are drawn from a shared prior [12], have a cluster structure [15], live on a low-dimensional linear subspace [16, 13], have an explicit hierarchical structure among them [8], or by modeling task relationships in a task covariance matrix in a Gaussian Process framework [7].

A danger that many multitask learning algorithms often run into is the problem of *negative transfer*, i.e., some outlier tasks can adversely affect the learning of legitimate tasks. One way to address this issue is to assume that the tasks have a cluster structure [5, 15]: similar tasks have their task parameters belonging to the same cluster, and dissimilar tasks belong to different clusters. Although conceptually appealing, this assumption often can be restrictive and also prone to overfitting. The task clustering based multitask learning attempts to learn a mixture model over the *latent* task parameters. However, since the task parameter of each task is usually very high dimensional (equal to the number of features) and the number of tasks itself may be much smaller, the standard mixture model based parameter estimation may not be reliable, eventually leading to poor estimates of the task parameters we are trying to learn. This problem is basically akin to the one faced in standard mixture models of *data* (note the distinction however; in clustering assumption based multitask learning, we learn the mixture model over the *latent* task parameters, not the data).

In the context of learning mixture models for the clustering of high dimensional data with very small number of examples, the mixture of factor analysis [11] (MFA) model is an attractive choice. Instead of learning the usual mixture of Gaussians, the low-rank assumption of each factor analysis model in the mixture circumvents the problem of overfitting and poor parameter estimation in such cases. We propose a similar approach for the multitask learning problem where instead of modeling the task parameters by a Gaussian mixture model (or even an infinite mixture model [15]), we model them using a mixture of factor analyzers models, which essentially means that each task parameter is generated from a mixture of low-rank Gaussians. Our work is also in contrast with

prior work on multitask learning [16] that assumes that the task parameters live on a (single) low-dimensional linear subspace. Our mixture of subspaces assumption can capture the notion that the task parameters could potentially live on a *non-linear* subspace [9]. Therefore, our model could be considered as capturing two notions of task relatedness simultaneously: cluster assumption and the subspace assumption (in particular, a non-linear subspace). Furthermore, another benefit of using a mixture of subspaces instead of a single subspace is that now different groups of task parameters could live in different subspaces, which should prevent the problem of negative transfer which may arise if we let all task parameters share the same linear subspace.

2 Mixture of Factor Analyzers for Multitask Learning

We assume that we are given M tasks with task parameters $\Theta = \{\theta_1, \dots, \theta_M\}$, $\forall \theta_m \in \mathbb{R}^d$. In the mixture of factor analyzers setting, we assume that each of these task parameters are generated by a mixture of L low-rank Gaussians, and K is the subspace dimensionality within each mixture component. Note however that, in principle, the subspace dimensionality can be different for each mixture component but we assume it to be the same for simplicity. The generative model is as follows: $p(\theta_m) = \sum_{i=1}^L \pi_i p(\theta_m|i)$, where $p(\theta_m|i)$ denotes a single factor analysis model for the i -th mixture component and π_i denotes the corresponding mixing proportion. For the i -th component factor analyzer of θ_m , there is a K dimensional latent variable s_{mi} , and another latent variable z_{mi} which is 1 if θ_m is generated from s_{mi} (i.e., from the i -th component), and 0 otherwise. We write $p(\theta_m|i)$ as a latent factor model:

$$\theta_m = \mu_i + W_i s_{mi} + \epsilon_i$$

where $W_i \in \mathbb{R}^{d \times K}$, $s_{mi} \sim \text{Nor}(s_{mi}|0, I) \in \mathbb{R}^{K \times 1}$, and $\epsilon_i \sim \text{Nor}(\epsilon_i|0, \Psi)$.

In this model, $\mathcal{Z} = \{\Theta, S, Z\}$ are the latent variables where $S = \{s_{mi}\}$, $Z = \{z_{mi}\}$, and $\Omega = \{\pi_i, \mu_i, W_i, \Psi\}$ are the model parameters. Both the latent variables and the parameters can be estimated in an alternating fashion using the EM algorithm. Note that our primary goal is to estimate the weight vectors $\Theta = \{\theta_1, \dots, \theta_M\}$ of all the M tasks.

We denote the data for all the M tasks as $\mathcal{D} = \{(X^{(m)}, Y^{(m)})\}_{m=1}^M$ where number of examples for task m as N^m . The complete data log-likelihood under this model is given by:

$$\log p(\mathcal{D}, \mathcal{Z}|\Omega) = \sum_{m=1}^M \{p(\mathcal{D}^{(m)}|\theta_m) + p(\mathcal{Z}^{(m)}|\Omega)\} \quad (1)$$

which can be written as:

$$\log p(\mathcal{D}, \mathcal{Z}|\Omega) = \sum_{m=1}^M \left\{ \sum_{i=1}^{N^m} \log p(Y_i^{(m)}|X_i^{(m)}, \theta_m) + \log p(\mathcal{Z}^{(m)}|\Omega) \right\} \quad (2)$$

where $p(Y_i^{(m)}|X_i^{(m)}, \theta_m)$ can be any probabilistic discriminative model for classification or regression, and

$$\log p(\mathcal{Z}^{(m)}|\Omega) = \sum_{i=1}^L z_{mi} \{ \log \pi_i p(\theta_m|s_{mi}, \mu_i, W_i, \Psi) + \log p(s_{mi}|0, I) \}.$$

The EM algorithm for estimating the latent variables \mathcal{Z} and the parameters is as follows:

- **E-step:** Given the parameters $\Omega^{(t-1)}$ from the $(t-1)^{th}$ step, compute the distribution over the latent variables given the old parameters $\Omega^{(t-1)}$ and data \mathcal{D} : $p(\mathcal{Z}^{(m)}|\Omega^{(t-1)}, \mathcal{D})$.
- **M-step:** Maximize the expected log likelihood of the complete data (Equation (2), where the expectation is taken over the distribution over the latent variables from the E-step: $\Omega^t = \arg \max_{\Omega} \mathbb{E}_{p(\mathcal{Z}^{(m)}|\Omega^{(t-1)}, \mathcal{D})} \log p(\mathcal{D}, \mathcal{Z}|\Omega)$

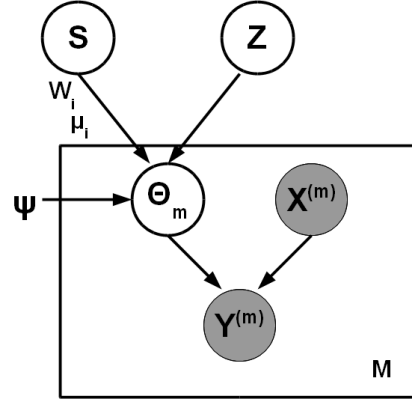


Figure 1: The generative model: A mixture component with mixing proportion π_i , mean μ_i , and loading matrix W_i is chosen for a task parameter θ_m . The task parameter θ_m is then generated from its low dimensional representation s_{mi} in the i^{th} mixture component.

Proceeding in a manner similar to [16], one can see that the M-step reduces to $\Omega^t = \arg \max_{\Omega} \sum_{m=1}^M \sum_{i=1}^L z_{mi} \{\log \pi_i p(\theta_m | s_{mi}, \mu_i, W_i, \Psi)\}$, since the rest of the terms in the complete log-likelihood expression of Equation (2) are independent of Ω . The E-step requires estimating $z_{mi} \propto \pi_i \mathcal{N}(\theta_m | \theta_m - \mu_i, W_i W_i^T + \Psi)$, and the first and second moments of θ_m and s_{mi} [11]. Since exact computations are intractable for the case of logistic classification (since the likelihood is not conjugate to the prior), the expectations in such cases can be computed using a variational Bayes approach as is done in [16]. We reserve the full details for a longer version.

3 Related Work

Our work can be considered as a probabilistic analogue to the model proposed in [4]. The model assumes that tasks can be partitioned into groups and tasks within each group share a low-dimensional representation. In particular, the tasks within each group share a kernel D_k such that the similarity between two examples of a task in some group k is given by the kernel $k(X_i, X_j) = \langle X_i, D_k^{-1} X_j \rangle$. This assumption is basically an extension of the earlier work [3] that assumes all tasks share the common kernel D (i.e., $K = 1$) such that the similarity between any two examples of each task is given by the kernel $k(X_i, X_j) = \langle X_i, D^{-1} X_j \rangle$. As shown in [3], this assumption is equivalent to the $d \times M$ matrix $\Theta = \{\theta_1, \dots, \theta_M\}$ of task parameters having a low rank (more precisely, a convex relaxation of the low rank constraint which amounts to minimizing the *trace norm* of Θ).

The multitask learning model with K groups of tasks [4] can be seen as minimizing the following regularized loss function:

$$\inf_{D_1, \dots, D_K > 0} \sum_{m=1}^M \min_{k \in \{1, \dots, K\}} \min_{\theta_m \in \mathbb{R}^d} \sum_{n=1}^{N_m} \ell(Y_i^{(m)}, \theta_m^T X_i^{(m)}) + \lambda \langle \theta_m, D_k^{-1} \theta_m \rangle$$

s.t. $\text{trace}(D_k) \leq 1$. The outer minimization in the above objective function can be seen as first finding the optimal group (kernel) for each task, and the inner minimization can be seen as learning the task parameter given that group/kernel. It is a non-convex problem and is solved using stochastic gradient descent, and has been shown to converge to a local optimum [4].

The generative model we proposed in this paper offers a number of advantages over the above model such as the ability to deal with missing data in a principled manner [14], and extensions such as doing automatic model complexity control in a fully Bayesian nonparametric setting.

The assumption of task parameters living in a nonlinear subspace has been used in the manifold based multitask learning model of [1]. However, the model assumes a *single* manifold shared by all task parameters and it does not have a build-in mechanism to deal with outlier (or negatively related) tasks. Therefore it seems likely that a few outlier or negatively related tasks could adversely affect the performance of this model.

4 Future Work and Discussion

We proposed a generative model for multitask learning based on the assumption that the task parameters are generated by a mixture of linear subspaces. The mixture assumption allows effective sharing of information among related tasks (via the cluster and the subspace assumptions of our model), and at the same time leads to better robustness against noise by allowing segregation of outlier tasks. The proposed model requires specifying the number of mixture components and the subspace dimensionality within each mixture component. However, taking a fully Bayesian approach, the model can be made fully nonparametric by using a Dirichlet Process mixture model [2] on the mixture components, and automatic relevance determination [6] (ARD) prior or the Indian Buffet Process [10] to choose the subspace dimensionality within each mixture component. By unleashing the power of such flexible models, only minimal assumptions are required to express the task structures shared by multiple tasks, and we expect that it would lead to more expressive and more robust multitask and transfer learning models.

References

- [1] A. Agarwal, S. Gerber, and H. Daumé III. Learning multiple tasks using manifold regularization. In *NIPS*, 2010.
- [2] C. E. Antoniak. Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The Annals of Statistics*, 2(6), 1974.
- [3] A. Argyriou, T. Evgeniou, and M. Pontil. Multi-task feature learning. In *NIPS*, 2007.
- [4] A. Argyriou, A. Maurer, and M. Pontil. An algorithm for transfer learning in a heterogeneous environment. In *ECML*, 2008.
- [5] B. Bakker and T. Heskes. Task clustering and gating for bayesian multitask learning. *JMLR*, 4, 2003.
- [6] C. M. Bishop. Bayesian PCA. In *NIPS*, 1999.
- [7] E. V. Bonilla, K. M. A. Chai, and C. K. I. Williams. Multi-task gaussian process prediction. In *NIPS*, 2007.
- [8] H. Daumé III. Bayesian Multitask Learning with Latent Hierarchies. In *UAI*, 2009.
- [9] Z. Ghahramani and M. J. Beal. Variational inference for bayesian mixtures of factor analysers. In *NIPS*, 2000.
- [10] Z. Ghahramani, T. Griffiths, and P. Sollich. Bayesian Nonparametric Latent Feature Models. In *Bayesian Statistics 8. Oxford University Press*, 2007.
- [11] Z. Ghahramani and G. E. Hinton. The em algorithm for mixtures of factor analyzers. Technical report, 1997.
- [12] T. Heskes. Empirical Bayes for learning to learn. *ICML*, 2000.
- [13] P. Rai and H. Daumé III. Infinite predictor subspace models for multitask learning. In *AISTATS*, Sardinia, Italy, 2010.
- [14] C. Wang, X. Liao, L. Carin, and D. B. Dunson. Classification with incomplete data using dirichlet process priors. In *JMLR*, 2010.
- [15] Y. Xue, X. Liao, L. Carin, and B. Krishnapuram. Multi-task Learning for Classification with Dirichlet Process Priors. *JMLR*, 2007.
- [16] J. Zhang, Z. Ghahramani, and Y. Yang. Learning multiple related tasks using latent independent component analysis. In *NIPS*, 2006.