Heterogenous Data Fusion via a Probabilistic Latent-Variable Model

Kai Yu, Volker Tresp

Information and Communications,
Siemens Corporate Technology,
Otto-Hahn-Ring, 6
81730 Munich, Germany

Abstract. In a pervasive computing environment, one is facing the problem of handling heterogeneous data from different sources, transmitted over heterogeneous channels and presented on heterogeneous user interfaces. This calls for adaptive data representations keeping as much relevant information as possible while keeping the representation as small as possible. Typically, the gathered data can be high-dimensional vectors with different types of attributes, e.g. continuous, binary and categorical data. In this paper we present - as a first step - a probabilistic latent-variable model, which is capable of fusing high-dimensional heterogenous data into a unified low-dimensional continuous space, and thus brings great benefits for multivariate data analysis, visualization and dimensionality reduction. We adopt a variational approximation to the likelihood of observed data and describe an EM algorithm to fit the model. The advantages of the proposed model are illustrated on toy data and used on real-world painting image data for both visualization and recommendation.

1 Introduction

Among others, pervasive computing will be characterized by the processing of heterogeneous and high-dimensional data. For example, results provided by Internet search engines may contain text, pictures, hyperlinks, categorial and binary data. The demand for clearly structured information presented to end users, but also the limitations of telecommunication networks as well as user interfaces calls for a lower-dimensional representation providing the most relevant information. Promising candidates for this task of dimensionality reduction are latent variable models.

Latent variable analysis is a family of data modelling approaches that factorizes high-dimensional observations with a reduced set of latent variables. The latent variables offer explanations of the dependencies between observed variables. An example is the probabilistic variant of the widely used principal component analysis, PPCA, where observations are explained by a linear projection of a set of Gaussian hidden variables, plus additive observation noise [10]. Standard PCA is widely used for data reduction, pattern recognition and exploratory

data analysis. Recent studies on PCA reveals its connections to statistical factor analysis (FA) [7].

While existing PCA or FA approaches rely on continuous-valued observations, data analysis on mixed types of data (discrete and continous observations) is often desirable:

- In solving typical data mining problems, one is always faced with mixed data. For example, a hospital patient's record typically includes fields like age (discrete real-valued), gender (binary), various examination results (real-valued or categorical), binary indicator variables for the presence of symptoms or even textual descriptions. A unified means to explore the dependencies of these data are needed.
- If applied to dimensionality reduction for pattern recognition, PCA is purely unsupervised. Thus, the resulting projection can be not indicative of the targeted pattern distribution. A generalized PCA which allows class membership as additional attributes (binary or categorical) may obviously provide a better solution.
- For heterogeneous data, it is often difficult to derive a small set of common features describing the total data. For example, in a web-based image retrieval system, each image can be characterized by its visual features, accompanying words, categories, and user visit records.

For these reasons, we will present a probabilistic latent variable model to fit observations with both continuous and binary attributes in this paper. Since categorical attributes can always be encoded by sets of binary attributes[1] (e.g. 1-of-c coding scheme), this model can be applied to a wide range of situations. We call this model generalized probabilistic PCA, GPPCA.

In the next section we describe the latent variable model and derive an efficient variational expectation-maximization (EM) formalism to learn the model from data. In Sec. 3 we discuss properties of the model and connections to previous work. In Sec. 5 we present empirical results based on toy data and image data, with focus on both data visualization and information filtering.

2 A Generalized Probabilistic PCA Model

The goal of a latent variable model is to find a representation for the distribution $p(\boldsymbol{t})$ of observed data in an M-dimensional space $\boldsymbol{t} = (t_1, \ldots, t_M)$ in terms of a number of L latent variables $\boldsymbol{x} = (x_1, \ldots, x_L)$. In our setting of interest, we consider a total of M continuous and binary attributes. We use $m \in \mathcal{R}$ to indicate that the variable t_m is continuous-valued, and $m \in \mathcal{B}$ for binary variables (i.e. $\{0,$

[1] To be precise, an additional constraint is required here, which we drop for simplicity.

1}). The generative model is:

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \tag{1}$$

$$\boldsymbol{y}|\boldsymbol{x} = \boldsymbol{W}^T\boldsymbol{x} + \boldsymbol{b} \tag{2}$$

$$t_m|y_m \sim \mathcal{N}(y_m, \sigma^2) \qquad m \in \mathcal{R} \tag{3}$$

$$t_m|y_m \sim \mathrm{Be}\big(g(y_m)\big) \qquad m \in \mathcal{B} \tag{4}$$

By $\mathrm{Be}(p)$ we denote a Bernoulli distribution with parameter p (the probability of giving a 1). $\boldsymbol{W}$ is an $L \times M$ matrix with column vectors $(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_M)$, $\boldsymbol{b}$ an M-dimensional column vector, and $g(a)$ the sigmoid function $g(a) = 1/(1 + \exp(-a))$. We assume that observed vectors $\boldsymbol{t}$ are generated from a prior Gaussian distribution with zero mean[2] unit covariance. Note that we assume a common noise variance σ^2 for all continuous variables. To match this assumption, we sometimes need to use scaling or whitening as a pre-processing step for the continuous data in our experiments.

The likelihood[3] of an observation vector $\boldsymbol{t}$ given the latent variables $\boldsymbol{x}$ and model parameters θ is

$$\begin{aligned} p(\boldsymbol{t}|\boldsymbol{x}, \theta) &= p(\boldsymbol{t}^{\mathcal{R}}|\boldsymbol{x}, \theta)p(\boldsymbol{t}^{\mathcal{B}}|\boldsymbol{x}, \theta) \\ &= \prod_{m\in\mathcal{R}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big\{ -\frac{1}{2}\frac{(y_m - t_m)^2}{\sigma^2} \Big\} \prod_{m\in\mathcal{B}} g\big((2t_m - 1)y_m\big) \end{aligned} \tag{5}$$

where $y_m = \boldsymbol{w}_m^T\boldsymbol{x} + b_m$. The distribution in $\boldsymbol{t}$-space, for a given value of θ is then obtained by integration over the latent variables $\boldsymbol{x}$

$$p(\boldsymbol{t}|\theta) = \int p(\boldsymbol{t}|\boldsymbol{x}, \theta)p(\boldsymbol{x})d\boldsymbol{x} \tag{6}$$

For a given set of N observation vectors, the log likelihood of data D is

$$\mathcal{L}(\theta) = \log p(D|\theta) = \sum_{n=1}^{N} \log p(\boldsymbol{t}_n|\theta) \tag{7}$$

We estimate the model parameters $\theta = \{\boldsymbol{W}, \boldsymbol{b}, \sigma^2\}$ using a maximum likelihood approach, which can be achieved by the expectation-maximization (EM) algorithm. However, given parameters θ estimated from the previous M-step, the integral Eq. (6) in the E-step can not be solved analytically. We thus have to resort to an approximated solution. Previous work on mixed latent variable models has concentrated, for example, on approximating the (equivalent of the) integral Eq. (6) by Monte Carlo sampling [6] or by Gauss-Hermite numerical integration [8]. These approaches demonstrate good performance in many cases, but introduce a rather high computational cost. In the next section, we will present a variational approximation to solve this problem.

[2] A non-zero mean and non-identity covariance matrix can be moved to parameters $\boldsymbol{W}$ and $\boldsymbol{b}$ without loss of generality.

[3] A full Bayesian treatment would require prior distributions for the parameters θ. We do not go for a full Bayesian solution here, thus implicitly assuming a non-informative prior.

2.1 A Variational EM Algorithm for Model Fitting

In order to select the parameters θ that maximize Eq. (7), we employ a variational EM algorithm. A variational EM algorithm constructs a lower bound (the variational approximation) for the likelihood of observations, Eq. (7), by first introducing additional variational parameters ψ. Then, it iteratively maximizes the lower bound with respective to the variational parameters (at the E-step) and the parameters θ of interest (at the M-step). This idea has been applied by Tipping [9] to a hidden-variable model for binary data only.

A variational approximation for the likelihood contributions of binary variables, $t_m \in \mathcal{B}$ in Eq. (4) is given by

$$\begin{aligned} p(t_m|\boldsymbol{x},\theta) &\geq \widetilde{p}(t_m|\boldsymbol{x},\theta,\psi_m) \\ &= g(\psi_m)\exp\big\{(A_m-\psi_m)/2+\lambda(\psi_m)(A_m^2-\psi_m^2)\big\} \end{aligned} \tag{8}$$

where $A_m = (2t_m-1)(\boldsymbol{w}_m^T\boldsymbol{x}+b_m)$ and $\lambda(\psi_m) = [0.5-g(\psi_m)]/2\psi_m$. For a fixed value of $\boldsymbol{x}$, we get the perfect approximation where the lower bound is maximized to be $p(t_m|\boldsymbol{x},\theta)$ by setting $\psi_m = A_m$.[4] The variational approximation for the log likelihood Eq. (7) of data D becomes

$$\mathcal{L}(\theta) \geq \mathcal{F}(\theta,\Psi) = \log\prod_{n=1}^{N}\int \widetilde{p}(\boldsymbol{t}_n|\boldsymbol{x},\theta,\boldsymbol{\psi}_n)p(\boldsymbol{x})d\boldsymbol{x} \tag{9}$$

where

$$\widetilde{p}(\boldsymbol{t}_n|\boldsymbol{x},\theta,\boldsymbol{\psi}_n) = \prod_{m\in\mathcal{R}} p(t_{mn}|\boldsymbol{x},\boldsymbol{w}_m,\sigma)\prod_{m\in\mathcal{B}}\widetilde{p}(t_{mn}|\boldsymbol{x},\boldsymbol{w}_m,\psi_{mn}) \tag{10}$$

We denote the total set of $N\times|\mathcal{B}|$ variational parameters by Ψ. Since the variational approximation depends on $\boldsymbol{x}$ only quadratically in the exponent and the prior $p(\boldsymbol{x})$ is Gaussian, the integrals to obtain the approximation $\mathcal{F}(\theta,\Psi)$ can be solved in closed form.

The variational EM algorithm starts with an initial guess of θ and then iteratively maximizes $\mathcal{F}(\theta,\Psi)$ with respect to Ψ (E-step) and θ (M-step), respectively, holding the other fixed. Each iteration increases the lower bound, but will not necessary maximize the true log likelihood $\mathcal{L}(\theta)$. However, since the E-step results a very close approximation of $\mathcal{L}(\theta)$, we expect that, at M-step, the true log likelihood is increased. Details are given in the following:

(i) E-step: $\Psi^{k+1} \leftarrow \arg\max_\Psi \mathcal{F}(\theta^k,\Psi)$. The optimization can be achieved by a normal EM approach. Given $\boldsymbol{\psi}_n^{\text{old}}$ updated from the previous step, the algorithm iteratively estimates the sufficient statistics for the posterior approximation $\tilde{p}(\boldsymbol{x}_n|\boldsymbol{t}_n,\theta^k,\boldsymbol{\psi}_n^{\text{old}})$[5], which is again a Gaussian with covariance and mean

[4] However, in the case of $\boldsymbol{x}$ distributed over a Gaussian prior $\mathcal{N}(0,\boldsymbol{I})$, maximization of the corresponding lower bound with respect to ψ_m is not straightforward.

[5] Based on Bayes' rule, the posterior approximation is derived by normalizing $\widetilde{p}(\boldsymbol{t}_n|\boldsymbol{x}_n,\theta^k,\boldsymbol{\psi}_n^{\text{old}})p(\boldsymbol{x}_n)$ and thus is a proper density, no longer a lower bound.

given by

$$\boldsymbol{C}_n = \big[\frac{1}{\sigma^2}\sum_{m\in\mathcal{R}} \boldsymbol{w}_m\boldsymbol{w}_m^T + \boldsymbol{I} - 2\sum_{m\in\mathcal{B}} \lambda(\psi_{mn}^{\text{old}})\boldsymbol{w}_m\boldsymbol{w}_m^T\big]^{-1} \tag{11}$$

$$\boldsymbol{\mu}_n = C_n\Big\{\frac{1}{\sigma^2}\sum_{m\in\mathcal{R}}(t_{mn}-b_m)\boldsymbol{w}_m + \sum_{m\in\mathcal{B}}\big[\frac{2t_{mn}-1}{2} + 2b_m\lambda(\psi_{mn}^{\text{old}})\big]\boldsymbol{w}_m\Big\} \tag{12}$$

and then updates $\boldsymbol{\psi}_n$ by maximizing $E_n\big\{\log\widetilde{p}(\boldsymbol{t}_n,\boldsymbol{x}_n|\theta^k,\boldsymbol{\psi}_n)\big\}$ where the expectation is with respect to $\tilde{p}(\boldsymbol{x}_n|\boldsymbol{t}_n,\theta^k,\boldsymbol{\psi}_n^{\text{old}})$. Taking the derivative of $E_n\big\{\log\widetilde{p}(\boldsymbol{t}_n,\boldsymbol{x}_n|\theta^k,\boldsymbol{\psi}_n)\big\}$ with respect to $\boldsymbol{\psi}_n$ and setting it to zero leads to the updates

$$\psi_{mn}^2 = E_n\big\{(\boldsymbol{w}_m^T\boldsymbol{x}_n + b_m)^2\big\} = \boldsymbol{w}_m^T E_n(\boldsymbol{x}_n\boldsymbol{x}_n^T)\boldsymbol{w}_m + 2b_m\boldsymbol{w}_m^T E_n(\boldsymbol{x}_n) + b_m^2 \tag{13}$$

where $E_n(\boldsymbol{x}_n\boldsymbol{x}_n^T) = \boldsymbol{C}_n + \boldsymbol{\mu}_n\boldsymbol{\mu}_n^T$ and $E_n(\boldsymbol{x}_n) = \boldsymbol{\mu}_n$. The two-stage optimization updates ψ and monotonously increases $\mathcal{F}(\theta^k,\Psi)$. The experiments showed that this procedure converges rapidly, most often in only two steps.

(ii) M-step: $\theta^{k+1} \leftarrow \arg\max_\theta \mathcal{F}(\theta,\Psi^{k+1})$. Similar to the former E-step, this can also be achieved by iteratively first estimating the sufficient statistics of $\tilde{p}(\boldsymbol{x}_n|\boldsymbol{t}_n,\theta^{\text{old}},\boldsymbol{\psi}_n^{k+1})$ through Eq. (11) and Eq. (12), and then maximizing $\sum_{n=1}^N E_n\big\{\log\widetilde{p}(\boldsymbol{t}_n,\boldsymbol{x}_n|\theta,\boldsymbol{\psi}_n^{k+1})\big\}$ with respect to θ, where $E_n(\cdot)$ denotes the expectation over $\tilde{p}(\boldsymbol{x}_n|\boldsymbol{t}_n,\theta^{\text{old}},\boldsymbol{\psi}_n^{k+1})$. For $m\in\mathcal{R}$, we derive the following updates

$$\boldsymbol{w}_m^T = \Big[\sum_{n=1}^N (t_{mn}-b_m)E_n(\boldsymbol{x}_n)^T\Big]\Big[\sum_{n=1}^N E_n(\boldsymbol{x}_n\boldsymbol{x}_n^T)\Big]^{-1} \tag{14}$$

$$\sigma^2 = \frac{1}{N|\mathcal{R}|}\sum_{n=1}^N\Big\{\sum_{m\in\mathcal{R}}\big[\boldsymbol{w}_m^T E_n(\boldsymbol{x}_n\boldsymbol{x}_n^T)\boldsymbol{w}_m + 2(b_m - t_{mn})\boldsymbol{w}_m^T E_n(\boldsymbol{x}_n) + (b_m - t_{mn})^2\big]\Big\} \tag{15}$$

where b_m, $m\in\mathcal{R}$, is directly estimated by the mean of t_{mn}. For $m\in\mathcal{B}$, we have the following updates

$$(\boldsymbol{w}_m^T, b_m)^T = -\Big[\sum_{n=1}^N 2\lambda(\psi_{mn})E_n(\widehat{\boldsymbol{x}}_n\widehat{\boldsymbol{x}}_n^T)\Big]^{-1}\Big[\sum_{n=1}^N (t_{mn}-0.5)E_n(\widehat{\boldsymbol{x}}_n)\Big] \tag{16}$$

where $\widehat{\boldsymbol{x}}_n = (\boldsymbol{x}^T, 1)^T$.

2.2 Inference

Finally, given the trained generative model, we can infer the *a posteriori* distribution of hidden variables for a complete observation vector $\boldsymbol{t}$ by using Bayes' rule

$$p(\boldsymbol{x}|\boldsymbol{t},\theta) = \frac{p(\boldsymbol{t}|\boldsymbol{x},\theta)p(\boldsymbol{x})}{\int p(\boldsymbol{t}|\boldsymbol{x},\theta)p(\boldsymbol{x})d\boldsymbol{x}} \tag{17}$$

However, since the integral is again infeasible, we need to derive a variational approximation by normalizing $\widetilde{p}(\boldsymbol{t}|\boldsymbol{x}, \theta, \boldsymbol{\psi})p(\boldsymbol{x})$, where $\boldsymbol{\psi}$ is obtained by maximizing the lower bound $\tilde{p}(\boldsymbol{t}|\theta, \boldsymbol{\psi})$.

For a vector $\hat{\boldsymbol{t}}$ of partial observations, we can still infer the posterior distribution in a similar way. If only continuous variables are observed, a normal posterior calculation can be employed, without the need for a variational approximation. This solution is the same as calculating the posterior based on the standard probabilistic PCA model [10].

3 Properties of Generalized Probabilistic PCA

The rows of the ML estimator $\boldsymbol{W}$ that relates latent to observed variables span the principal subspace of the data. The GPPCA model allows a unified probabilistic modelling of continuous, binary and categorical observations, which can bring great benefits in real-world data analysis. Also, it can serve as a visualization tool for high-dimensional mixed data in a two-dimensional latent variable space. Existing models currently only visualize either continuous [1] or binary data [9]. Also, like PPCA [10], GPPCA specifies a full generative model, it can also handle missing observations in a principled way.

For pattern recognition tasks, GPPCA can provide a principled data transformation for general learning algorithms (which most often rely on continuous inputs) to handle data with mixed types of attributes. One such example, in the context of painting image recommender system incorporating visual features, artists, user ratings, will be shown in Sec. 5. Also, GPPCA can provide a principled approach to *supervised* dimensionality reduction, by allowing the target values as additional observation variables. GPPCA explores the dependence between inputs and targets via the hidden variables and maximizes the joint likelihood of both. It actually discovers a subspace of the joint space in which the projections of inputs have small projection loss and also have clear class distributions. A large number of methods have been developed to handle issue of supervised data reduction (see [4]), like partial least squares, discriminant analysis. However most of them, in general, can not handle missing data.

4 Relation to Previous Work

Jaakkola & Jordan [5] proposed a variational likelihood approximation for Bayesian logistic regression, and briefly pointed out that the same approximation can be applied to learn the "dual problem", i.e. a hidden-variable model for binary observations. Tipping [9] derived the detailed variational EM formalism to learn the model and used it to visualize high-dimensional binary data. Collins *et al.* [3] generalized PCA to various loss functions from the exponential family, in which the case of Bernoulli variables is similar to Tipping's model. Latent variable models for mixed observation variables were also studied by [6] and [8]. In contrast to our variational approach, [6] and [8] used numerical integration methods to handle the otherwise intractable integral in the EM algorithm. Latent variable

models for mixed data were already mentioned by Bishop [1] and Tipping [9], yet never explicitly implemented. Recently, Cohn [2] proposed *informed projections*, a version of supervised PCA, that minimizes both projection loss and inner-class dissimilarities. However, this requires tuning a parameter β to weight the two parts of the loss function,

5 Empirical Study

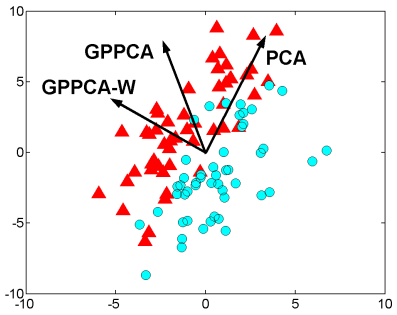
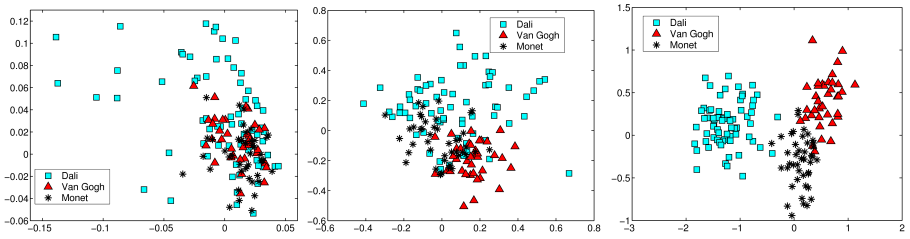


Fig. 1. A toy problem: PCA, GPPCA and GPPCA-W solutions

5.1 A Toy Problem

We first illustrate GPPCA on a simple problem, where 100 two-dimensional samples are generated from two Gaussian distributions with mean $[-1, 1]$ and $[1, -1]$ respectively and equal covariance matrices. A third binary variable was added that indicates which Gaussian the sample belongs to. We perform GPPCA, as described in Sec. 2, and standard PCA on the data to identify the principal subspace. The results are illustrated in Fig. 1. As expected, the PCA solution is along the direction of largest variance. The GPPCA solution, on the other hand, also takes the class labels into account, and finds a solution that conveys more information about the observations. In an additional experiment, we pre-process the continuous variables with whitening and then perform GPPCA. We will refer to this as GPPCA-W in the following. With GPPCA-W, the solution even more clearly indicates the class distribution. Clearly, a change of the subspace in $\boldsymbol{W}$ corresponding to the whitened continuous variables will no longer change the likelihood contribution. Thus, the GPPCA EM algorithm will focus on the likelihood of binary observations only and thus lead to a result with clear class distribution.



(a) PCA solution (b) GPPCA solution (c) GPPCA-W solution

Fig. 2. Visualization of painting images

5.2 Visualization of Painting Images

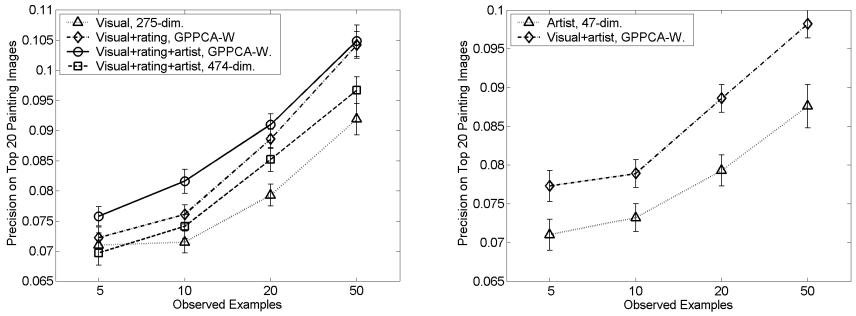
Next, we show an application of GPPCA to visualizing image data. We consider a data set of 642 painting images from 47 artists. An often encountered problem in the research on image retrieval is that low-level visual features (like color, texture, and edges) can hardly capture high-level information of images, like concept, style, etc. GPPCA allows to characterize images by more information than just those low-level features. In the experiment, we examine if it is possible to visualize different styles of painting images in a 2-dimensional space by incorporating the information about artists.

As the continuous data describing the images, we extract 275 low-level features (correlagram, wavelet texture, and color moment) for each image. We encode the artists in 47 binary attributes via a 1-of-c scheme, and obtain a 322-dimensional vector with mixed data for each image. The result of projecting this data to a 2-dimensional latent space is shown in Fig. 2, where we limit the shown data to the images of 3 particular artists.

The solution given by normal PCA does not allow a clear separation of artists. In contrast, the GPPCA solution, in particular when performing an additional whitening pre-processing for the continuous features, shows a very clear separation of artists. Note furthermore, that the distinction between Van Gogh and Monet is a bit fuzzy here—these artists do indeed share similarities in their style, in particular brush stroke, which is reflected by texture features.

5.3 Recommendation of Painting Images

Due to the deficiency of low-level visual features, building recommender systems for painting image is a challenging task. Here we will demonstrate that GPPCA allows a principled way of deriving compact and highly informative features. Thus the accuracy of recommender systems based on the new image features can be significantly improved.



(a) (b)

Fig. 3. Precision on painting image recommendation, based on different features

We use the same set of 642 painting images as in the previous section. 190 users' ratings (like, dislike, or no rated) were collected through an online survey [6]. For each image, we combine visual features (275-dim.), artist (47-dim.), and a set of M *advisory* users' ratings on it (M-dim.) to form an $(322+M)$-dimensional feature vector. This feature vector contains continuous, binary and missing data (because on average each user only rated 89 images). We apply GPPCA to map the features to a reduced 50-dimensional feature space. The rest of $190-M$ users are then treated as test users. For each test user, we hide some of his/her ratings and assume that only 5, 10, 20, or 50 ratings are observed. We skip one particular case if a user has not given that many ratings. Then we use the rated examples, in form of input (image features) – output(ratings) pairs, to train an RBF-SVM model to predict the user's ratings on unseen images and make a ranking. The performance of recommendation is evaluated by the top-20 precision, which is the fraction of actually liked images among the top-20 recommendations. We equally divide the 190 users into 5 groups, pick one group as the group of test users and treat the other 152 users as advisory users. For each tested case, we randomize 10 times and calculate the mean and error bars. The results are shown in Fig. 3.

Fig. 3(a) shows that GPPCA improves the precision in all the cases by effectively incorporating richer information. This is not surprising since the information about artists is a good indicator of painting styles. Advisory users' opinions on a painting actually reflect some high-level properties of the painting from a different individual's perspective. GPPCA here provides a princpled way to represent different information sources into a unified form of continuous data, and allows accurate recommendations based on the reduced data. Interestingly, as shown in Fig. 3(a), a recommender system working with direct combination of

[6] http://honolulu.dbs.informatik.uni-muenchen.de:8080/paintings/index.jsp

the three aspects of information shows a much lower precision than the compact form of features. This indicates that GPPCA effectively detects the 'signal subspace' of high dimensional mixed data, while eliminating irrelevant information. Note that there are over 80 percent of missing data in the user ratings. GPPCA also provides an effective means to handle this problem. Fig. 3(b) shows that GPPCA incorporating visual features and artist information significantly outperforms a recommender sytem that only works on artist information. This indicates that GPPCA working on the pre-whitened continuous data does not remove the influence of visual features.

6 Conclusion

This paper describes generalized probabilistic PCA (GPPCA), a latent-variable model for mixed types of data, with continuous and binary observations. By adopting a variational approximation, an EM algorithm can be formulated that allows an efficient learning of the model parameters from data. The model generalizes probabilistic PCA and opens new perspectives for multivariate data analysis and machine learning tasks. We demonstrated the advantages of the proposed GPPCA model on toy data and data from painting images. GPPCA allows an effective visualization of data in two-dimensional hidden space that takes into account both information from low-level image features and artist information. Our experiments on an image retrieval task show that the model provides a principled solution to incorporating different information sources, thus significantly improving the achievable precision. Currently the described model reveals the linear principal subspace for mixed high dimensional data. It might be interesting to pursue non-linear hidden variable model to handle mixed types of data.

This approach and its possibly extensions may provide the basis for - even adaptively - compactifying data representations in future pervasive computing environments thus increasing their performance and acceptance.

7 Acknowledgments

We would thank Dr. Rudolf Sollacher and Anton Schwaighofer for their very constructive comments to this work.

References

[1] Bishop, C. M., Svensen, M., and Williams, C. K. GTM: The generative topographic mapping. *Neural Compuation*, 10(1):215–234, 1998.

[2] Cohn, D. Informed projections. In S. Becker, S. Thrun, and K. Obermayer, eds., *Advances in Neural Information Processing Systems*, 15. MIT Press, 2003.

[3] Collins, M., Dasgupta, S., and Schapire, R. A generalization of principal component analysis to the exponential family. In T. K. Leen, T. G. Dietterich, and V. Tresp, eds., *Advances in Neural Information Processing Systems*, 13. MIT Press, 2001.

[4] Hastie, T., Tibshirani, R., and Friedman, J. *The Elements of Satatistical Learning*. Springer Verlag, 2001.

[5] Jaakkola, T. and Jordan, M. Bayesian parameter estimation via variational methods. *Statistics and Computing*, pp. 25–37, 2000.

[6] Moustaki, I. A latent trait and a latent class model for mixed observed variables. *British Journal of Mathematical and Statistical Psychology*, 49:313–334, 1996.

[7] Roweis, S. and Ghahramani, Z. A unifying review of linear gaussian models. *Neural Computaion*, 11:305–345, 1999.

[8] Sammel, M. D., Ryan, L. M., and Legler, J. M. Latent variable models for mixed discrete and continuous outcomes. *Journal of the Royal Statistical Society*, Series B 59:667–678, 1997.

[9] Tipping, M. E. Probabilistic visualization of high-dimensional binary data. In M. S. Kearns, S. A. Solla, and D. A. Cohn, eds., *Advances in Neural Information Processing Systems*, 11, pp. 592–598. MIT Press, 1999.

[10] Tipping, M. E. and Bishop, C. M. Probabilistic principal component analysis. *Journal of the Royal Statisitical Scoiety*, B(61):611–622, 1999.