Non-Linear Matrix Factorization

Neil D. Lawrence[1], Raquel Urtasun[2]
School of Computer Science, University of Manchester, UK [1]
UC Berkeley EECS & ICSI, Berkeley, U.S.A.[2]
neill@cs.man.ac.uk, rurtasun@csail.mit.edu

Introduction

A standard approach to matrix factorization is a singular value decomposition. In this paper we show how a probabilistic matrix factorization is equivalent to probabilistic principal component analysis. We then extend this model in a non-linear way to give a probabilistic non-linear matrix factorization. A popular application of matrix factorization is collaborative filtering. We apply our approach to non-linear decomposition of several collaborative filtering benchmarks. Our non-linear approach consistently outperforms all other published approaches on these data sets.

The latent factor approach to collaborative filtering sees the data as a sparsely populated matrix of ratings. For a data set with N items and D users we take this matrix to be $\mathbf{Y} \in \Re^{N\times D}$. The objective is to factorize $\mathbf{Y}$ into a lower rank form, $\mathbf{Y} \approx \mathbf{U}^\top\mathbf{V}$ [see *e.g.* 1, 2], where $\mathbf{U} \in \Re^{q\times N}$ and $\mathbf{V} \in \Re^{q\times D}$. Prediction can then be done by estimating ($\mathbf{U}$, $\mathbf{V}$) from the training data and computing the resulting approximation to $\mathbf{Y}$. In this paper we consider a non-linear generalization of this approach.

Consider the very natural probabilistic interpretation of this factorization given by [3] and referred to as *probabilistic matrix factorization* (PMF) that takes the form

$$p\left(\mathbf{Y}|\mathbf{U},\mathbf{V},\sigma^2\right) = \prod_{i=1}^{N} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{V}^\top\mathbf{u}_{:,i},\sigma^2\mathbf{I}\right).$$

where $\mathbf{u}_{:,i}$ is the ith column of $\mathbf{U}$ and $\mathbf{y}_{i,:}$ is a column vector taken from the ith row of $\mathbf{Y}$ containing ratings of the ith item from the users. In practice $\mathbf{y}_{i,:}$ will have many missing values, but we will ignore this aspect for the moment.

It turns out that the unconstrained PMF is probabilistically equivalent to Bayesian PCA [4]. This is a little easier to see with a small change of notation. Consider a matrix of latent variables, $\mathbf{X} \equiv \mathbf{U}^\top \in \Re^{N\times q}$ and a mapping matrix which goes from the latent space to the observed data space, $\mathbf{W} \equiv \mathbf{V}^\top \in \Re^{D\times q}$. Using this new notation,

$$p\left(\mathbf{Y}|\mathbf{W},\mathbf{X},\sigma^2\right) = \prod_{i=1}^{N} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:},\sigma^2\mathbf{I}\right), \tag{1}$$

the likelihood has a familiar look. It is a multi-output linear regression from a q dimensional feature matrix $\mathbf{X}$ to matrix targets $\mathbf{Y}$. In this model we can either place a Gaussian prior over $\mathbf{X}$ or over $\mathbf{W}$ and in both cases optimizing with respect to the other variable recovers probabilistic PCA [5, 6]. If we were to marginalize both $\mathbf{X}$ and $\mathbf{W}$ we would recover Bayesian PCA, but this would require further approximations [4, 7, 8]. Marginalizing over $\mathbf{X}$ leads to

$$p\left(\mathbf{Y}|\mathbf{X},\sigma^2,\alpha_x\right) = \prod_{j=1}^{D} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\alpha_w^{-1}\mathbf{X}\mathbf{X}^\top + \sigma^2\mathbf{I}\right).$$

When D is large and there are many missing values, this marginal likelihood can be optimized by stochastic gradient descent (SGD) over the D users. Alternatively $\mathbf{X}$ can be marginalized and SGD over the N items may be carried out.

Non-linear Model

The marginal likelihood above can be seen as part of a larger class of models called Gaussian process latent variable models (GP-LVM) [6]. Briefly put, the likelihood can be "kernelized" by replacing $\mathbf{X}\mathbf{X}^\top$ with a Mercer kernel. The result is a product of Gaussian processes [9]. Importantly, if the data matrix $\mathbf{Y}$ is relatively sparse, the resulting log likelihood can be efficiently maximized using stochastic gradient descent, enabling application of the model to data sets with many thousands of users and items.

Experimental Evaluation

We consider three data sets to assess the quality of our probabilistic non-linear approach for collaborative filtering. The data sets are widely used benchmarks for collaborative filtering, each containing a set of item ratings from

	Weak NMAE	Strong NMAE
URP	0.4422 ± 0.0008	0.4557 ± 0.0008
Attitude	0.4520 ± 0.016	0.4550 ± 0.0023
MMMF	0.4397 ± 0.0006	0.4341 ± 0.0025
Item	0.4382 ± 0.0009	0.4365 ± 0.0024
E-MMMF	0.4287 ± 0.0023	0.4301 ± 0.0035
Ours Linear	**0.4209 ± 0.0017**	**0.4171 ± 0.0054**
Ours RBF	**0.4179 ± 0.0018**	**0.4134 ± 0.0049**

	Weak NMAE	Strong NMAE
URP	0.4341 ± 0.0023	0.4444 ± 0.0032
Attitude	0.4320 ± 0.0055	0.4375 ± 0.0028
MMMF	0.4156 ± 0.0037	0.4203 ± 0.0138
Item	0.4096 ± 0.0029	0.4113 ± 0.104
E-MMMF	0.4029 ± 0.0027	0.4071 ± 0.0093
Ours linear	0.4052 ± 0.0011	**0.4071 ± 0.0081**
Ours RBF	**0.4026 ± 0.0020**	**0.3994 ± 0.0145**

Left: EachMovie data set and **Right**: 1M MovieLens data. Our non-linear approach with RBF kernel is consistently the best performer across all these benchmarks.

different users. These data sets are the EachMovie, the 1M MovieLens, and the recently released 10M MovieLens data[1].

We followed the widely used experimental setup for these data suggested by Marlin for the 1M MovieLens and EachMovie data sets as this allows us to compare directly to the best published results which include: the user rating profile [URP, 10], Attitude [11], maximum margin matrix factorization [MMMF, 1], the approach of [12] that combines collaborative filtering with item proximities, and ensembles of MMMF [2]. To our knowledge the results of [2] were the best reported to date on the EachMovie and 1M MovieLens databases.

Marlin used the normalized mean absolute error (NMAE) as an error measure so that random guessing produces a score of 1. Marlin defines two types of generalization, "weak" and "strong". *Weak* generalization is a single step process which involves filling-in missing data in the rating matrix. *Strong* generalization is a two-stage process, where the models are trained on one set of users and the test predictions are on a disjoint set of users. The learner is given sample ratings of those users, but may not use those ratings until after the initial model is constructed.

Note that our approach, with either a linear or an RBF kernel, outperforms significantly all the baselines. The tables above show the NMAE for the baselines as well as for our approach. Using an RBF covariance function our approach results in the best performance for both weak and strong generalization.

Finally the 10M MovieLens data set consists of 10 million ratings for $71,567$ users and $10,681$ movies, with ratings ranging $\{1, 2, \cdots, 5\}$. There are currently no other published results to compare with, but our approach gave results in a NMAE of (**0.3968** ± 0.0165), and a RMSE of (**0.8740** ± 0.0278) using a 10 dimensional latent space.

Discussion

We have introduced an approach for probabilistic non-linear matrix factorization which outperforms all other published approaches on two important collaborative filtering benchmark data sets. The approach makes use of Gaussian processes to non-linearize the matrix factorization and predict missing ratings.

References

[1] Rennie, J. D. M. & Srebro, N. In de Raedt, L. & Wrobel, S. (eds.) *ICML*, vol. 22, 713–719 (2005).

[2] DeCoste, D. In Ghahramani, Z. (ed.) *ICML*, vol. 24 (Omnipress, 2007).

[3] Salakhutdinov, R. & Mnih, A. In Platt, J. C., Koller, D., Singer, Y. & Roweis, S. (eds.) *NIPS*, vol. 20, 1257–1264 (MIT, Cambridge, MA, 2008). In press.

[4] Bishop, C. M. In Kearns, M. J., Solla, S. A. & Cohn, D. A. (eds.) *NIPS*, vol. 11, 482–388 (MIT, Cambridge, MA, 1999).

[5] Tipping, M. E. & Bishop, C. M. *JRSSb* **6**, 611–622 (1999).

[6] Lawrence, N. D. *JMLR* **6**, 1783–1816 (2005).

[7] Minka, T. P. In Leen, T. K., Dietterich, T. G. & Tresp, V. (eds.) *NIPS*, vol. 13, 598–604 (MIT, Cambridge, MA, 2001).

[8] Salakhutdinov, R. & Mnih, A. In Roweis, S. & McCallum, A. (eds.) *ICML*, vol. 25 (Omnipress, 2008). In Press.

[9] Rasmussen, C. E. & Williams, C. K. I. *Gaussian Processes for Machine Learning* (MIT, Cambridge, MA, 2006).

[10] Marlin, B. In Thrun, S., Saul, L. & Schölkopf, B. (eds.) *NIPS*, vol. 16 (MIT, Cambridge, MA, 2004).

[11] Marlin, B. *Collaborative filtering: A machine learning perspective*. Master's thesis, University of Toronto (2004).

[12] Park, S.-T. & Pennock, D. M. In *13th ACM SIGKDD* (2007).

[1] See `http://www.grouplens.org/`.