Objective and Derivatives for MMMF using the Natural Parameter Multinomial

Jason D. M. Rennie
jrennie@gmail.com

February 22, 2006*

Abstract

We provide basic calculations for applying MMMF to text using a natural parameter Multinomial model.

1 Introduction

We provide the math that extends Maximum-Margin Matrix Factorization [3] to text. We replace the hinge (classification) loss function with the multinomial negative-log likelihood. We retain the trace norm regularization of MMMF, but the maximum-margin loss is gone, replaced by the multinomial negative log-likelihood.

2 Natural Parameter Multinomial

We use the natural parameter formulation of the multinomial, as discussed in [1]. We assume we are given a term frequency matrix, Y, for a set of documents. We use X to represent the matrix of parameters for the multinomial. The likelihood for document i is

$$P(Y_i|X_i) = \frac{\left(\sum_j Y_{ij}\right)!}{\prod_j Y_{ij}!} \prod_j \left(\frac{\exp(X_{ij})}{\sum_{j'} \exp(X_{ij})}\right)^{Y_{ij}}. \qquad (1)$$

The negative log-likelihood is

$$-\log P(Y_i|X_i) = \sum_j Y_{ij} \left[\log\left(\sum_{j'} \exp(X_{ij'})\right) - X_{ij}\right] + C, \qquad (2)$$

where $C = \sum_j \log Y_{ij}! - \log\left(\sum_j Y_{ij}\right)!$ is a function of Y_i only. We use the negative log-likelihood summed across documents as the loss for the data.

*Updated March 29, 2006.

3 Learning

For MMMF, we want to minimize the data loss subject to a constraint on the trace norm, or minimize the trace norm subject to a constraint on the data loss. Equivalently, we can minimize a combined objective,

$$J(X) = \lambda \|X\|_\Sigma - \sum_i \log P(Y_i|X_i). \tag{3}$$

$\|X\|_\Sigma$ is the trace norm (sum of singular values) of the matrix X. The coefficient $\lambda \in [0, \infty)$ provides a trade-off between minimization of the trace norm and minimization of the data loss. By controlling λ, we can achieve solutions to any of the posed problems.

The given objective, J, is not easy to optimize. However, we can pose a different, easier-to-optimize objective with the same global minimum. We make use of the fact that the trace norm of a matrix is equal to the minimum over factorizations, $\|X\|_\Sigma = \min_{U,V} \frac{1}{2}(\|U\|^2_{\text{Fro}} + \|V\|^2_{\text{Fro}})$. Let U_i be the i^{th} row of U. Let V_j be the j^{th} row of V. Our alternate objective simply substitutes this identity,

$$J'(U, V) = \frac{\lambda}{2}(\|U\|^2_{\text{Fro}} + \|V\|^2_{\text{Fro}}) - \sum_i \log P(Y_i|U_i V^T). \tag{4}$$

Since we are minimizing J' over U, V, it is immediately clear that the global minima of the two objectives are identical, $\min_{U,V} J'(U, V|Y) = \min_X J(X|Y)$. Unfortunately, this alternate objective is not convex. However, empirical tests indicate that local minima are, at worst, rare [2].

Appendix: Implementation Details

We optimize J' using gradient descent. To do this, we make use of the objective and gradient. We write out the math in detail. We assume functions and operations (e.g. log(), exp(), ^{2}, $*$, $/$) are applied element-wise). We use 1 to represent the ones column vector. First, we calculate the objective,

$$J(X) = \|X\|_\Sigma + \sum_{i,j} Y_{ij} \left[\log \left(\sum_{j'} \exp(X_{ij'}) \right) - X_{ij} \right] \tag{5}$$

$$\begin{aligned} J'(U, V) &= \frac{\lambda}{2}(\|U\|^2_{\text{Fro}} + \|V\|^2_{\text{Fro}}) + \sum_{i,j} Y_{ij} \left[\log \left(\sum_{j'} \exp(U_i V_{j'}^T) \right) - U_i V_j^T \right] \\ &= \frac{\lambda}{2}(1^T U^2 1 + 1^T V^2 1) + \log(\exp(UV^T)1)^T (Y1) - 1^T (Y * UV^T) 1. \end{aligned} \tag{6}$$

Next, we calculate the partial derivative with respect to U,

$$\frac{\partial J'}{\partial U_{ia}} = \lambda U_{ia} + \frac{\sum_j V_{ja} \exp(U_i V_j^T)}{\sum_j \exp(U_i V_j^T)} \sum_j Y_{ij} - \sum_j Y_{ij} V_{ja} \tag{7}$$

$$= \lambda U_{ia} + \frac{\exp(U_i V^T) V_{\cdot a}}{\exp(U_i V^T) 1} * (Y_i 1) - Y_i V_{\cdot a} \tag{8}$$

$$\frac{\partial J'}{\partial U_i} = \lambda U_i + \frac{\exp(U_i V^T) V}{(\exp(U_i V^T) 1) 1^T} * (Y_i 1) 1^T - Y_i V \tag{9}$$

$$\frac{\partial J'}{\partial U} = \lambda U + \frac{\exp(U V^T) V}{(\exp(U V^T) 1) 1^T} * (Y 1) 1^T - Y V. \tag{10}$$

Finally, we calculate the partial derivative with respect to V,

$$\frac{\partial J'}{\partial V_{ja}} = \lambda V_{ja} + \sum_i \frac{U_{ia} \exp(U_i V_j^T)}{\sum_k \exp(U_i V_k^T)} \sum_k Y_{ik} - \sum_i Y_{ij} U_{ia} \tag{11}$$

$$= \lambda V_{ja} + \left[\frac{\exp(U V_j^T) * Y 1}{\exp(U V^T) 1} \right]^T U_{\cdot a} - Y_{\cdot j}^T U_{\cdot a} \tag{12}$$

$$\frac{\partial J'}{\partial V_j} = \lambda V_j + \left[\frac{\exp(U V_j^T) * Y 1}{\exp(U V^T) 1} \right]^T U - Y_{\cdot j}^T U \tag{13}$$

$$\frac{\partial J'}{\partial V} = \lambda V + \left[\frac{\exp(U V^T) * (Y 1) 1^T}{(\exp(U V^T) 1) 1^T} \right]^T U - Y^T U. \tag{14}$$

References

[1] J. D. M. Rennie. Mixtures of multinomials. http://people.csail.mit.edu/jrennie/writing, September 2005.

[2] J. D. M. Rennie and N. Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd International Conference on Machine Learning*, 2005.

[3] N. Srebro, J. D. M. Rennie, and T. S. Jaakkola. Maximum-margin matrix factorization. In *Advances in Neural Information Processing Systems 17*, 2005.