Learning with Whom to Share in Multi-task Feature Learning: Supplementary Material

Zhuoliang Kang ZKANG@USC.EDU
Department of Computer Science, U. of Southern California, Los Angeles, CA 90089

Kristen Grauman GRAUMAN@CS.UTEXAS.EDU
Department of Computer Science, U. of Texas, Austin, TX 78701

Fei Sha FEISHA@USC.EDU
Department of Computer Science, U. of Southern California, Los Angeles, CA 90089

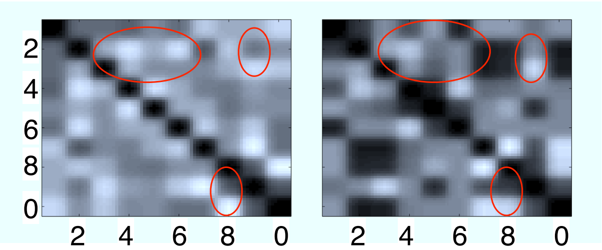

Figure 1. Likelihoods of pairwise tasks being assigned to the same cluster, for USPS (left) and MNIST (right). While the two are largely different, there are some regions (highlighted with red ellipses) for which the two are similar, suggesting that at least for some digits, the grouping structure our algorithm discovers is not highly sensitive to the exact instances or dataset used during learning.

1. Additional Experiments

1.1. Handwritten digit recogntiion

1.1.1. Visualizing discovered task groups

Given that both datasets consist of very related content—handwritten characters—we are interested in seeing whether the task group structure learned in either case has any similarities. To analyze this, we compute the likelihoods for pairwise tasks being assigned to the same cluster, as follows: we take the task-cluster membership assignment q_{gt} and form a matrix $\boldsymbol{Q} \in \mathbb{R}^{\mathsf{G}\times\mathsf{T}}$. We then compute $\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{Q}$, and average over the final results of all experiments. If the s-th task and the t-th task are often assigned to the same cluster, then the averaged $\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{Q}$ will have high values at (s, t)-th element.

Fig. 1 visualizes this matrix for the USPS (left) and MNIST (right) datasets. Despite their very different results on MTL classification accuracy, we see that the two matrices do share some common regional patterns. (Note, in the figure, the diagonal elements are set to zero to better highlight the other elements on paper.)

A. Proof of Theorem 1

Our proof starts by observing that $T(\boldsymbol{Q})$ can be expressed in an equivalent form

$$T(\boldsymbol{Q}) = \sum_g \min_{\boldsymbol{\Omega}_g} \mathsf{Trace}\left[\boldsymbol{\Omega}_g^{-1}\boldsymbol{W}\sqrt{\boldsymbol{Q}_g}\sqrt{\boldsymbol{Q}_g}^{\mathrm{T}}\boldsymbol{W}^{\mathrm{T}}\right] \quad (1)$$

where $\boldsymbol{\Omega}_g$ is constrained to be positive definitive. Furthermore, $\mathsf{Trace}[\boldsymbol{\Omega}_g] = 1$. Let $\boldsymbol{\Psi}_g = \boldsymbol{W}^{\mathrm{T}}\boldsymbol{\Omega}_g^{-1}\boldsymbol{W}$, we have

$$T(\boldsymbol{Q}) = \min \sum_g \mathsf{Trace}\left[\boldsymbol{\Psi}_g\boldsymbol{Q}_g\right] \quad (2)$$

Since $\boldsymbol{Q}_g$ is a diagonal matrix, we have immediately

$$T(\boldsymbol{Q}) = \min \sum_g \sum_t \psi_{tt}^g q_{gt} \quad (3)$$

where ψ_{tt}^g is the t-th diagonal element of $\boldsymbol{\Psi}_g$. Thus, in terms of q_{gt}, eq. (9) of Theorem 1 is just to minimize over a linear function of these variables over the polytope defined by $q_{gt} \geq 0$ and $\sum_g q_{gt} = 1$. Therefore, by appealing to the basic property of linear programming, the statement of the theorem is obviously true.

B. Calculating the gradient

For notation simplicity, we consider calculating the gradient of $\|\boldsymbol{W}\sqrt{\boldsymbol{Q}}\|_*$ with the t-th element q_t of the diagonal matrix $\boldsymbol{Q}$.

From the definition, we have

$$\|\boldsymbol{W}\sqrt{\boldsymbol{Q}}\|_* = \mathsf{Trace}\left[\boldsymbol{W}\boldsymbol{Q}\boldsymbol{W}^{\mathrm{T}}\right]^{1/2} \quad (4)$$

We decompose $\boldsymbol{W}\boldsymbol{Q}\boldsymbol{W}^{\mathrm{T}}$ in its eigenvalues and eigenvectors,

$$\boldsymbol{W}\boldsymbol{Q}\boldsymbol{W}^{\mathrm{T}} = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^{\mathrm{T}} \tag{5}$$

where $\boldsymbol{\Lambda}$ is the diagonal matrix composed of the eigenvalue λ_t. This leads to

$$\|\boldsymbol{W}\sqrt{\boldsymbol{Q}}\|_* = \sum_i \sqrt{\lambda_i} \tag{6}$$

To calculate the gradient of $\|\boldsymbol{W}\sqrt{\boldsymbol{Q}}\|_*$ with respect to q_t, we need to compute the gradient of λ_i with respect to q_t

$$\frac{\partial \lambda_i}{\partial q_t} = \boldsymbol{p}_i^{\mathrm{T}}\boldsymbol{w}_t\boldsymbol{w}_t^{\mathrm{T}}\boldsymbol{p}_i \tag{7}$$

where $\boldsymbol{p}_i$ is the eigenvector corresponding to λ_i and $\boldsymbol{w}_t$ is the t-th column in $\boldsymbol{W}$, ie, the parameter vector of the t-th task. Combining everything together, we have,

$$\frac{\partial \|\boldsymbol{W}\sqrt{\boldsymbol{Q}}\|_*}{\partial q_t} = \sum_i \frac{1}{2\sqrt{\lambda_i}}\boldsymbol{p}_i^{\mathrm{T}}\boldsymbol{w}_t\boldsymbol{w}_t^{\mathrm{T}}\boldsymbol{p}_i \tag{8}$$