Supplemental Material for

Multi-Task Learning using Generalized t Process

1 Detailed Proofs

In this supplemental material, we provide the proofs for Eqs. (3), (4) and (8).

Before we present our proofs, we first review some relevant properties of the matrix-variate normal distribution and the Wishart distribution as given in [1].

Lemma 1 ([1], Corollary 2.3.10.1) *If* $\mathbf{X} \sim \mathcal{MN}_{q\times s}(\mathbf{M}, \mathbf{\Sigma} \otimes \mathbf{\Psi})$*,* $\mathbf{d} \in \mathbb{R}^q$ *and* $\mathbf{c} \in \mathbb{R}^s$*, then*

$$\mathbf{d}^T\mathbf{X}\mathbf{c} \sim \mathcal{N}(\mathbf{d}^T\mathbf{M}\mathbf{c}, (\mathbf{d}^T\mathbf{\Sigma}\mathbf{d})(\mathbf{c}^T\mathbf{\Psi}\mathbf{c})).$$

Lemma 2 ([1], Theorem 2.3.5) *If* $\mathbf{X} \sim \mathcal{MN}_{q\times s}(\mathbf{M}, \mathbf{\Sigma} \otimes \mathbf{\Psi})$ *and* $\mathbf{A} \in \mathbb{R}^{s\times s}$*, then*

$$\mathbb{E}(\mathbf{X}\mathbf{A}\mathbf{X}^T) = tr(\mathbf{A}^T\mathbf{\Psi})\mathbf{\Sigma} + \mathbf{M}\mathbf{A}\mathbf{M}^T.$$

Lemma 3 ([1], Theorem 3.3.16) *If* $\mathbf{S} \sim \mathcal{MN}_q(a, \mathbf{\Sigma})$ *where* $a - q - 1 > 0$*, then*

$$\mathbb{E}(\mathbf{S}^{-1}) = \frac{\mathbf{\Sigma}^{-1}}{a-q-1},$$

where $\mathbf{S}^{-1}$ *denotes the inverse of* $\mathbf{S}$*.*

For Eq. (3), using Lemma 1 and the fact that $\mathbf{W} \sim \mathcal{MN}_{d'\times m}(\mathbf{0}_{d'\times m}, \mathbf{I}_{d'} \otimes \mathbf{\Sigma})$, we can get

$$f_j^i \stackrel{\text{def}}{=} \phi(\mathbf{x}_j^i)^T\mathbf{w}_i = \phi(\mathbf{x}_j^i)^T\mathbf{W}\mathbf{e}_{m,i} \sim \mathcal{N}(0, (\phi(\mathbf{x}_j^i)^T\mathbf{I}_{d'}\phi(\mathbf{x}_j^i))(\mathbf{e}_{m,i}^T\mathbf{\Sigma}\mathbf{e}_{m,i})).$$

Since $\phi(\mathbf{x}_j^i)^T\mathbf{I}_{d'}\phi(\mathbf{x}_j^i) = k(\mathbf{x}_j^i, \mathbf{x}_j^i)$ and $\mathbf{e}_{m,i}^T\mathbf{\Sigma}\mathbf{e}_{m,i} = \Sigma_{ii}$, we can get $f_j^i \sim \mathcal{N}(0, \Sigma_{ii}k(\mathbf{x}_j^i, \mathbf{x}_j^i))$.

For Eq. (4), we have

$$\begin{aligned}\langle f_j^i, f_s^r\rangle &= \int \phi(\mathbf{x}_j^i)^T\mathbf{W}\mathbf{e}_{m,i}\mathbf{e}_{m,r}^T\mathbf{W}^T\phi(\mathbf{x}_s^r)p(\mathbf{W})d\mathbf{W}\\ &= \phi(\mathbf{x}_j^i)^T\mathbb{E}(\mathbf{W}\mathbf{e}_{m,i}\mathbf{e}_{m,r}^T\mathbf{W}^T)\phi(\mathbf{x}_s^r),\end{aligned}$$

then using Lemma 2 and the fact that $\mathbf{W} \sim \mathcal{MN}_{d'\times m}(\mathbf{0}_{d'\times m}, \mathbf{I}_{d'} \otimes \mathbf{\Sigma})$, we can get

$$\begin{aligned}\langle f_j^i, f_s^r\rangle &= \phi(\mathbf{x}_j^i)^T\mathrm{tr}(\mathbf{e}_{m,r}\mathbf{e}_{m,i}^T\mathbf{\Sigma})\mathbf{I}_{d'}\phi(\mathbf{x}_s^r)\\ &= \mathrm{tr}(\mathbf{e}_{m,r}\mathbf{e}_{m,i}^T\mathbf{\Sigma})k(\mathbf{x}_j^i, \mathbf{x}_s^r)\\ &= \mathbf{e}_{m,i}^T\mathbf{\Sigma}\mathbf{e}_{m,r}k(\mathbf{x}_j^i, \mathbf{x}_s^r)\\ &= \Sigma_{ir}k(\mathbf{x}_j^i, \mathbf{x}_s^r).\end{aligned}$$

The second last equation holds because $\mathbf{e}_{m,i}$ and $\mathbf{e}_{m,r}$ are two vectors.

For Eq. (8), recall that the two random variables $\mathbf{S} \sim \mathcal{W}_{d'}(\nu + d' - 1, \mathbf{I}_{d'})$ and $\mathbf{Z} \sim \mathcal{MN}_{d'\times m}(\mathbf{0}_{d'\times m}, \mathbf{I}_{d'} \otimes \mathbf{\Psi})$ are independent and $\mathbf{W} = \mathbf{S}^{-1/2}\mathbf{Z}$. Then we can get

$$\begin{aligned}\langle f_j^i, f_s^r\rangle &= \int\int \phi(\mathbf{x}_j^i)^T\mathbf{w}_i\mathbf{w}_r^T\phi(\mathbf{x}_s^r)p(\mathbf{w}_i)p(\mathbf{w}_r)d\mathbf{w}_id\mathbf{w}_r\\ &= \int \phi(\mathbf{x}_j^i)^T\mathbf{W}\mathbf{e}_{m,i}\mathbf{e}_{m,r}^T\mathbf{W}^T\phi(\mathbf{x}_s^r)p(\mathbf{W})d\mathbf{W}\\ &= \int\int \phi(\mathbf{x}_j^i)^T\mathbf{S}^{-1/2}\mathbf{Z}\mathbf{e}_{m,i}\mathbf{e}_{m,r}^T\mathbf{Z}^T\mathbf{S}^{-1/2}\phi(\mathbf{x}_s^r)p(\mathbf{Z})p(\mathbf{S})d\mathbf{Z}d\mathbf{S}\\ &= \int \phi(\mathbf{x}_j^i)^T\mathbf{S}^{-1/2}\mathbb{E}(\mathbf{Z}\mathbf{e}_{m,i}\mathbf{e}_{m,r}^T\mathbf{Z}^T)\mathbf{S}^{-1/2}\phi(\mathbf{x}_s^r)p(\mathbf{S})d\mathbf{S}\\ &= \Psi_{ir}\int \phi(\mathbf{x}_j^i)^T\mathbf{S}^{-1}\phi(\mathbf{x}_s^r)p(\mathbf{S})d\mathbf{S} \qquad \text{(Using Lemma 2)}\\ &= \Psi_{ir}\phi(\mathbf{x}_j^i)^T\mathbb{E}(\mathbf{S}^{-1})\phi(\mathbf{x}_s^r)\\ &= \frac{\Psi_{ir}k(\mathbf{x}_j^i, \mathbf{x}_s^r)}{\nu - 2}. \qquad \text{(Using Lemma 3)}\end{aligned}$$

Moreover, according to Lemma 3, ν is required to be larger than 2.

2 Some More Theoretical Results

Similar to [2], we give here an upper bound on the learning curve.

It is useful to see how the matrix $\mathbf{G} = (\mathbf{\Lambda}^{-1} + \mathbf{\Omega}\mathbf{D}^{-1}\mathbf{\Omega}^T)^{-1}$ changes when a new data point from the ith task is added to the training set. The change is

$$\mathbf{G}(n+1) - \mathbf{G}(n) = \left[\mathbf{G}^{-1}(n) + \sigma_i^{-2}\boldsymbol{\varphi}\boldsymbol{\varphi}^T\right]^{-1} - \mathbf{G}(n) = -\frac{\mathbf{G}(n)\boldsymbol{\varphi}\boldsymbol{\varphi}^T\mathbf{G}(n)}{\sigma_i^2 + \boldsymbol{\varphi}^T\mathbf{G}(n)\boldsymbol{\varphi}},$$

where $\boldsymbol{\varphi}$ is a column vector with the ith element $\psi_i(\mathbf{x}_\star^i)$ and $\mathbf{x}_\star^i$ is the newly added data point from the ith task. To get the exact learning curve, we have to average this change with respect to all training sets that include $\mathbf{x}_\star^i$. This is difficult to achieve though. Here we ignore the correlation between the numerator and denominator and average them separately. Moreover, we treat n as a continuous variable and get

$$\frac{\partial \mathbf{H}(n)}{\partial n} = -\frac{\mathbb{E}[\mathbf{G}^2(n)]}{\sigma_i^2 + \text{tr}(\mathbf{H}(n))},$$

where $\mathbf{H}(n) = \mathbb{E}[\mathbf{G}(n)]$. We also neglect the fluctuations in $\mathbf{G}(n)$ and then get $\mathbb{E}[\mathbf{G}^2(n)] = \mathbf{H}^2(n)$. So we can get

$$\frac{\partial \mathbf{H}(n)}{\partial n} = -\frac{\mathbf{H}^2(n)}{\sigma_i^2 + \text{tr}(\mathbf{H}(n))}$$

$$\frac{\partial \mathbf{H}^{-1}(n)}{\partial n} = -\mathbf{H}^{-1}(n)\frac{\partial \mathbf{H}(n)}{\partial n}\mathbf{H}^{-1}(n) = (\sigma_i^2 + \text{tr}(\mathbf{H}(n)))^{-1}\mathbf{I}.$$

Since $\mathbf{H}^{-1}(0) = \mathbf{\Lambda}^{-1}$, $\mathbf{H}^{-1}(n) = \mathbf{\Lambda}^{-1} + \sigma_i^{-2}n'\mathbf{I}$ where n' needs to obey the following

$$\frac{\partial \sigma_i^{-2}n'}{\partial n} = \frac{1}{\sigma_i^2 + \text{tr}(\mathbf{H}(n))} = \frac{1}{\sigma_i^2 + \text{tr}\left((\mathbf{\Lambda}^{-1} + \sigma_i^{-2}n'\mathbf{I})^{-1}\right)},$$

which is equivalent to

$$\frac{\partial n'}{\partial n} + \text{tr}\left((\mathbf{\Lambda}^{-1} + \sigma_i^{-2}n'\mathbf{I})^{-1}\right)\sigma_i^{-2}\frac{\partial n'}{\partial n} = 1.$$

Integrating both sides, we can see that n' satisfies the following equation

$$n' + \sum_j \ln(n' + \sigma_i^2\lambda_j^{-1}) = n.$$

Then we can get the upper bound as

$$\varepsilon_{UB}^i = \frac{\omega}{\nu - 2}\left[\sigma_i^2 + \text{tr}\left((\mathbf{\Lambda}^{-1} + \sigma_i^{-2}n'\mathbf{I})^{-1}\right)\right].$$

References

[1] A. K. Gupta and D. K. Nagar. *Matrix Variate Distributions*. Chapman & Hall, 2000.

[2] P. Sollich and A. Halees. Learning curves for Gaussian process regression: Approximations and bounds. *Neural Computation*, 14(6):1393–1428, 2002.