Model Merging by Gradient Matching

Anonymous Author(s)
Affiliation
Address
email

Appendix

1 Derivations

1.1 Derivation of Task Arithmetic using Gradient Mismatch

We proceed by first writing the respective stationarity conditions for the LLM $\boldsymbol{\theta}_{\text{LLM}}$, fine-tuned models $\boldsymbol{\theta}_t$, and target model $\boldsymbol{\theta}_{1:T}$,

$$\begin{aligned}\boldsymbol{\theta}_{\text{LLM}} &= -\nabla\bar{\ell}_{\text{LLM}}(\boldsymbol{\theta}_{\text{LLM}})\\ \mathbf{H}_0(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) &= -\nabla\bar{\ell}_t(\boldsymbol{\theta}_t),\ \text{for all } t = 1, 2, \dots, T\\ \mathbf{H}_0(\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_{\text{LLM}}) &= \sum_{t=1}^{T} -\alpha_t \nabla\bar{\ell}_t(\boldsymbol{\theta}_{1:T}).\end{aligned}$$

Next, we multiply the second equation with α_t for each t, then sum it over $t = 1, 2, \dots, T$, and finally subtract it from the third equation to get the following,

$$\mathbf{H}_0(\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_{\text{LLM}}) - \sum_{t=1}^{T} \alpha_t \mathbf{H}_0(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) = -\sum_{t=1}^{T} \alpha_t \Big[\nabla\bar{\ell}_t(\boldsymbol{\theta}_{1:T}) - \nabla\bar{\ell}_t(\boldsymbol{\theta}_t)\Big]. \tag{1}$$

Multiplying by $\mathbf{H}_0^{-1}$ and rearranging gives us

$$\boldsymbol{\theta}_{1:T} = \underbrace{\boldsymbol{\theta}_{\text{LLM}} + \sum_{t=1}^{T} \alpha_t(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}})}_{=\bar{\boldsymbol{\theta}}_{\text{TA}}} - \sum_{t=1}^{T} \alpha_t \mathbf{H}_0^{-1} \underbrace{\left[\nabla\bar{\ell}_t(\boldsymbol{\theta}_{1:T}) - \nabla\bar{\ell}_t(\boldsymbol{\theta}_t)\right]}_{\text{Gradient mismatch for } \boldsymbol{\theta}_t \text{ on } \bar{\ell}_t}. \tag{2}$$

.

1.2 Derivation of the New Method

By substituting Taylor's approximation, the equation reduces to the first expression below which is linear in $\boldsymbol{\theta}_{1:T}$,

$$\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_{\text{LLM}} \approx \sum_{t=1}^{T} \alpha_t(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) - \sum_{t=1}^{T} \alpha_t \mathbf{H}_0^{-1} \left[\mathbf{H}_t(\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_t)\right]. \tag{3}$$

We then add and subtract $\boldsymbol{\theta}_{\text{LLM}}$ in the last term above,

$$\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_{\text{LLM}} \approx \sum_{t=1}^{T} \alpha_t(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) - \sum_{t=1}^{T} \alpha_t \mathbf{H}_0^{-1} \left[\mathbf{H}_t(\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_{\text{LLM}}) - \mathbf{H}_t(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}})\right], \tag{4}$$

and multiply the whole expression by $\mathbf{H}_0$ and rearrange it to get the second expression in Eq. 3,

$$\begin{aligned}\Big(\mathbf{H}_0 + \sum_{t=1}^{T} \alpha_t \mathbf{H}_t\Big)(\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_{\text{LLM}}) \;&\approx\; \sum_{t=1}^{T} \alpha_t \mathbf{H}_0(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) + \sum_{t=1}^{T} \alpha_t \mathbf{H}_t(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) \\ &=\; \sum_{t=1}^{T} \alpha_t (\mathbf{H}_0 + \mathbf{H}_t)(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}).\end{aligned} \tag{5}$$

Multiplying the equation by inverse of $\bar{\mathbf{H}} = \mathbf{H}_0 + \sum_{t=1}^{T} \alpha_t \mathbf{H}_t$ and taking $\boldsymbol{\theta}_{\text{LLM}}$ to the right hand side gives us

$$\hat{\boldsymbol{\theta}}_{1:T} = \boldsymbol{\theta}_{\text{LLM}} + \sum_{t=1}^{T} \alpha_t \,(\bar{\mathbf{H}}^{-1} \mathbf{H}_{0+t})\,(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}). \tag{6}$$

1.3 Derivation of Data Removal

Our target model is the following model trained using

$$\boldsymbol{\theta}_{\text{LLM}} = \arg\min_{\boldsymbol{\theta}} \;\bar{\ell}_{\text{LLM}}(\boldsymbol{\theta}) + \tfrac{1}{2}\delta\|\boldsymbol{\theta}\|^2, \text{ where } \bar{\ell}_{\text{LLM}}(\boldsymbol{\theta}) = \sum_{i \in \mathcal{D}_{\text{Large}}} \ell_i(\boldsymbol{\theta}). \tag{7}$$

but without using $\mathcal{D}_t$,

$$\boldsymbol{\theta}_{\backslash t} = \arg\min_{\boldsymbol{\theta}} \;\bar{\ell}_{\backslash t}(\boldsymbol{\theta}) + \frac{\delta}{2}\|\boldsymbol{\theta}\|^2, \quad \text{ where } \bar{\ell}_{\backslash t}(\boldsymbol{\theta}) = \sum_{i \in \{\mathcal{D}_{\text{Large}} \backslash \mathcal{D}_t\}} \ell_i(\boldsymbol{\theta}). \tag{8}$$

The LLM objective can then be written in terms of this objective:

$$\boldsymbol{\theta}_{\text{LLM}} = \arg\min_{\boldsymbol{\theta}} \;\bar{\ell}_{\backslash t}(\boldsymbol{\theta}) + \alpha_t \bar{\ell}_t(\boldsymbol{\theta}) + \frac{\delta}{2}\|\boldsymbol{\theta}\|^2, \tag{9}$$

where we assume that $\bar{\ell}_t$ is multiplied by a constant α_t in the original model.

As before, we can write the stationary conditions of $\boldsymbol{\theta}_{\text{LLM}}$, $\boldsymbol{\theta}_t$, and $\boldsymbol{\theta}_{\backslash t}$, respectively:

$$\begin{aligned} \delta\boldsymbol{\theta}_{\text{LLM}} &= -\nabla\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\text{LLM}}) - \alpha_t \nabla\bar{\ell}_t(\boldsymbol{\theta}_{\text{LLM}}), \\ \mathbf{H}_0(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) &= -\nabla\bar{\ell}_t(\boldsymbol{\theta}_t), \\ \delta\boldsymbol{\theta}_{\backslash t} &= -\nabla\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\backslash t}). \end{aligned} \tag{10}$$

Because our goal is to analyze $\boldsymbol{\theta}_{\backslash t} - \alpha_t(\boldsymbol{\theta}_{\text{LLM}} - \boldsymbol{\theta}_t)$, we multiply the second equation by α_t, subtract it from the first equation, and then subtract the resultant from the third equation to get, the following,

$$\delta(\boldsymbol{\theta}_{\backslash t} - \boldsymbol{\theta}_{\text{LLM}}) + \alpha_t \mathbf{H}_0(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) = -\left[\nabla\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\backslash t}) - \nabla\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\text{LLM}})\right] + \alpha_t \left[\nabla\bar{\ell}_t(\boldsymbol{\theta}_{\text{LLM}}) - \nabla\bar{\ell}_t(\boldsymbol{\theta}_t)\right]. \tag{11}$$

We can now use Taylor's approximation to reduce gradient matching,

$$\nabla\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\backslash t}) \approx \nabla\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\text{LLM}}) + \nabla^2\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\text{LLM}})(\boldsymbol{\theta}_{\backslash t} - \boldsymbol{\theta}_{\text{LLM}}).$$

For the second gradient term, we do not need to use the Taylor's approximation because it does not depend on $\boldsymbol{\theta}_{\backslash t}$, but our goal is to improve over task arithmetic, so we do it to derive a preconditioner,

$$\nabla\bar{\ell}_t(\boldsymbol{\theta}_{\text{LLM}}) \approx \nabla\bar{\ell}_t(\boldsymbol{\theta}_t) + \mathbf{H}_t(\boldsymbol{\theta}_{\text{LLM}} - \boldsymbol{\theta}_t). \tag{12}$$

Note that it is also possible to do the Taylor's approximation not around $\boldsymbol{\theta}_t$ but $\boldsymbol{\theta}_{\text{LLM}}$. Plugging these in Eq. 11, we can write,

$$\begin{aligned} &\delta(\boldsymbol{\theta}_{\backslash t} - \boldsymbol{\theta}_{\text{LLM}}) + \alpha_t \mathbf{H}_0(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) = -\nabla^2\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\text{LLM}})(\boldsymbol{\theta}_{\backslash t} - \boldsymbol{\theta}_{\text{LLM}}) + \alpha_t \left[\mathbf{H}_t(\boldsymbol{\theta}_{\text{LLM}} - \boldsymbol{\theta}_t)\right] \\ \implies\; &\left[\delta\mathbf{I} + \nabla^2\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\text{LLM}})\right](\boldsymbol{\theta}_{\backslash t} - \boldsymbol{\theta}_{\text{LLM}}) = -\alpha_t\,(\mathbf{H}_0 + \mathbf{H}_t)\,(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) \\ \implies\; &\boldsymbol{\theta}_{\backslash t} = \boldsymbol{\theta}_{\text{LLM}} - \alpha_t \left[\delta\mathbf{I} + \nabla^2\bar{\ell}_{\backslash t}(\boldsymbol{\theta}_{\text{LLM}})\right]^{-1}(\mathbf{H}_0 + \mathbf{H}_t)\,(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) \end{aligned}$$

which gives us the desired update of

$$\hat{\boldsymbol{\theta}}_{\backslash t} = \boldsymbol{\theta}_{\text{LLM}} - \alpha_t \bar{\mathbf{H}}_{\backslash t}^{-1} \mathbf{H}_{0+t}(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}), \tag{13}$$

1.4 Proof that our update for data-removal is exact for linear regression

The task removal update derived above is closely related to previous works on data removal. For instance, for linear model, our update recovers the popular influence function. We will now show this. Consider a large linear model (coincidentally also abbreviated as LLM) with full data $\mathcal{D} = (\mathbf{X}, \mathbf{y})$ where $\mathbf{y}$ is a vector of outputs and $\mathbf{X}$ is a matrix containing each feature vector as a row. The loss is $\bar{\ell}_{\text{LLM}}(\boldsymbol{\theta}) = \frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|^2$. Now, suppose we want to remove $\mathcal{D}_t = (\mathbf{X}_t, \mathbf{y}_t)$ from it. Then, we have a closed form solution for the full model and the model with removed data,

$$\boldsymbol{\theta}_{\text{LLM}} = \bar{\mathbf{H}}^{-1}\mathbf{X}^\top\mathbf{y}, \qquad \boldsymbol{\theta}_{\setminus t} = \bar{\mathbf{H}}_{\setminus t}^{-1}\mathbf{X}^\top\mathbf{y},$$

where $\bar{\mathbf{H}} = \nabla^2\left[\frac{1}{2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|^2 + \frac{1}{2}\|\boldsymbol{\theta}\|^2\right] = \mathbf{X}^\top\mathbf{X} + \delta\mathbf{I}$, and similarly $\bar{\mathbf{H}}_{\setminus t} = \mathbf{X}_{\setminus t}^\top\mathbf{X}_{\setminus t} + \delta\mathbf{I}$. A well known result in the influence function literature Cook (1977) is that the two quantities are related as

$$\boldsymbol{\theta}_{\setminus t} - \boldsymbol{\theta}_{\text{LLM}} = \bar{\mathbf{H}}_{\setminus t}^{-1}\mathbf{X}_t^\top(\mathbf{X}_t\boldsymbol{\theta}_{\text{LLM}} - \mathbf{y}_t). \tag{14}$$

We will now show that our previously proposed update reduces to this for linear models.

We start with an expression for $\boldsymbol{\theta}_t$ trained using

$$\boldsymbol{\theta}_t = \arg\min_{\boldsymbol{\theta}} \ \bar{\ell}_t(\boldsymbol{\theta}) + \tfrac{1}{2}\|\boldsymbol{\theta} - \boldsymbol{\theta}_{\text{LLM}}\|^2_{\mathbf{H}_0}, \tag{15}$$

but with the loss $\bar{\ell}_t(\boldsymbol{\theta}) = \frac{1}{2}\|\mathbf{y}_t - \mathbf{X}_t\boldsymbol{\theta}\|^2$. Using the second equation in the optimality condition of Eq. 10, we can write:

$$\mathbf{H}_0(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}) = \mathbf{X}_t^\top(\mathbf{y}_t - \mathbf{X}_t\boldsymbol{\theta}_t) \qquad \Longrightarrow \qquad (\mathbf{H}_0 + \mathbf{H}_t)\boldsymbol{\theta}_t = \mathbf{H}_0\boldsymbol{\theta}_{\text{LLM}} + \mathbf{X}_t^\top\mathbf{y}_t$$

where we use the fact that for linear models $\mathbf{H}_t = \mathbf{X}_t^\top\mathbf{X}_t$. We now simplify our update of Eq. 13 with $\alpha_t = 1$ where we use the above relationship in the third line below,

$$\begin{aligned}
\hat{\boldsymbol{\theta}}_{\setminus t} &= \boldsymbol{\theta}_{\text{LLM}} - \bar{\mathbf{H}}_{\setminus t}^{-1}\left(\mathbf{H}_0 + \mathbf{H}_t\right)\left(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}\right) \\
&= \boldsymbol{\theta}_{\text{LLM}} - \bar{\mathbf{H}}_{\setminus t}^{-1}\left[\left(\mathbf{H}_0 + \mathbf{H}_t\right)\boldsymbol{\theta}_t - \left(\mathbf{H}_0 + \mathbf{H}_t\right)\boldsymbol{\theta}_{\text{LLM}}\right] \\
&= \boldsymbol{\theta}_{\text{LLM}} - \bar{\mathbf{H}}_{\setminus t}^{-1}\Big(\mathbf{H}_0\boldsymbol{\theta}_{\text{LLM}} + \mathbf{X}_t^\top\mathbf{y}_t - \left(\mathbf{H}_0 + \mathbf{H}_t\right)\boldsymbol{\theta}_{\text{LLM}}\Big) \\
&= \boldsymbol{\theta}_{\text{LLM}} - \bar{\mathbf{H}}_{\setminus t}^{-1}\Big(\mathbf{X}_t^\top\mathbf{y}_t - \mathbf{H}_t\boldsymbol{\theta}_{\text{LLM}}\Big) \\
&= \boldsymbol{\theta}_{\text{LLM}} - \bar{\mathbf{H}}_{\setminus t}^{-1}\Big(\mathbf{X}_t^\top\mathbf{y}_t - \mathbf{X}_t^\top\mathbf{X}_t\boldsymbol{\theta}_{\text{LLM}}\Big) \\
&= \boldsymbol{\theta}_{\text{LLM}} + \bar{\mathbf{H}}_{\setminus t}^{-1}\mathbf{X}_t^\top\left(\mathbf{X}_t\boldsymbol{\theta}_{\text{LLM}} - \mathbf{y}_t\right).
\end{aligned} \tag{16}$$

Therefore, our update reduces to Eq. 14.

A generalization of Eq. 14 to neural network is considered in Koh & Liang (2017) for the case of one-example removal. Their approach when applied to remove multiple examples at once redues to

$$\hat{\boldsymbol{\theta}}_{\setminus t} = \boldsymbol{\theta}_{\text{LLM}} + \bar{\mathbf{H}}_{\setminus t}^{-1}\mathbf{g}_t,$$

where $\mathbf{g}_t = \nabla\bar{\ell}_t(\boldsymbol{\theta}_{\text{LLM}})$. Our approach also recovers this result if we do not use the second Taylor's approximation for the second gradient matching term in Eq. 11. Essentially, this removes the contribution of the fine-tuned model and the steps are completely based on $\boldsymbol{\theta}_{\text{LLM}}$. It is not clear which approach is better but in practice it may depend on the fine-tune process which by doing multiple gradient steps may be able to get more information than a single gradient step $\mathbf{g}_t$. We hope to explore this point in a future study.

1.5 Gradient Mismatch Reduction as Uncertainty Estimation

Both the gradient-mismatch connection and the new method are closely related to uncertainty estimation via approximate Bayesian methods. We show that

$$\boldsymbol{\theta}_{1:T} = \underbrace{\boldsymbol{\theta}_{\text{LLM}} + \sum_{t=1}^{T}\alpha_t(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}})}_{=\bar{\boldsymbol{\theta}}_{\text{TA}}} - \sum_{t=1}^{T}\alpha_t\mathbf{H}_0^{-1}\underbrace{\left[\nabla\bar{\ell}_t(\boldsymbol{\theta}_{1:T}) - \nabla\bar{\ell}_t(\boldsymbol{\theta}_t)\right]}_{\text{Gradient mismatch for } \boldsymbol{\theta}_t \text{ on } \bar{\ell}_t}. \tag{17}$$

is equivalent to a maximum-a-posteriori (MAP) estimate of the posterior over all data $\mathcal{D}_{1:T}$ while

$$\hat{\boldsymbol{\theta}}_{1:T} = \boldsymbol{\theta}_{\text{LLM}} + \sum_{t=1}^{T} \alpha_t \, (\bar{\mathbf{H}}^{-1}\mathbf{H}_{0+t}) \, (\boldsymbol{\theta}_t - \boldsymbol{\theta}_{\text{LLM}}), \tag{18}$$

is the same but for a posterior approximation obtained with Laplace's method (Laplace, 1774; Tierney & Kadane, 1986; MacKay, 1992). Based on these, we discuss some possible future directions for improvements.

We start by defining the posteriors. Assuming $p(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}_{\text{LLM}}, \mathbf{H}_0^{-1})$ to be the Gaussian prior and $p(\mathcal{D}_t|\boldsymbol{\theta}) \propto e^{-\bar{\ell}_t(\boldsymbol{\theta})}$ to be a valid likelihood function, we can define a weighted-posterior $p_\alpha(\boldsymbol{\theta}|\mathcal{D}_{1:T})$ over all datasets, shown below, along with an approximation obtained by Laplace's method,

$$p_\alpha(\boldsymbol{\theta}|\mathcal{D}_{1:T}) \propto \; p(\boldsymbol{\theta}) \prod_{t=1}^{T} e^{-\alpha_t \bar{\ell}_t(\boldsymbol{\theta})} \;\; \approx \;\; p(\boldsymbol{\theta}) \prod_{t=1}^{T} e^{-\frac{1}{2}\alpha_t \|\boldsymbol{\theta}-\boldsymbol{\theta}_t\|^2_{\mathbf{H}_t}} \propto q_\alpha(\boldsymbol{\theta}|\mathcal{D}_{1:T}). \tag{19}$$

Here, we use a second-order approximation at $\boldsymbol{\theta}_t$ to get $\bar{\ell}_t(\boldsymbol{\theta}) \approx \bar{\ell}_t(\boldsymbol{\theta}_t) + \frac{1}{2}\|\boldsymbol{\theta} - \boldsymbol{\theta}_t\|^2_{\mathbf{H}_t}$. The term $\bar{\ell}_t(\boldsymbol{\theta}_t)$ is an irrelevant constant and we get the approximation $q_\alpha(\boldsymbol{\theta}|\mathcal{D}_{1:T})$. The result below shows that the merged model is the MAP estimate corresponding to the approximate posterior.

Theorem 1 *The gradient mismatch equation in Eq.* 2 *is the stationarity condition of a MAP problem written in terms of posterior* $p(\mathcal{D}_t|\boldsymbol{\theta})$ *(the equation on the left), while the merged model* $\hat{\boldsymbol{\theta}}_{1:T}$ *in Eq.* 18 *is the MAP estimate of the Laplace approximation (equation on the right).*

$$\boldsymbol{\theta}_{1:T} = \arg\max_{\boldsymbol{\theta}} \; p(\boldsymbol{\theta}) \prod_{t=1}^{D} \left[\frac{p(\boldsymbol{\theta}|\mathcal{D}_t)}{p(\boldsymbol{\theta})}\right]^{\alpha_t}, \qquad\qquad \hat{\boldsymbol{\theta}}_{1:T} = \arg\max_{\boldsymbol{\theta}} \; q_\alpha(\boldsymbol{\theta}|\mathcal{D}_{1:T}). \tag{20}$$

A detailed proof is given in Sec. 1.6. The result relates the gradient-mismatch approach to the posterior distribution and its approximation. The first equation expresses model merging as merging of posteriors $p(\boldsymbol{\theta}|\mathcal{D}_t)$ that are computed on different datasets. With a Bayesian approach, an exact solution can be recovered even when training on separate datasets. This is an instance of the Bayesian committee machine (Tresp, 2000) or Bayesian data fusion (Mutambara, 1998; Durrant-Whyte, 2001; Wu et al., 2022) which are extensively used for Gaussian processes (Deisenroth & Ng, 2015) and which should also be useful when using Neural Tangent Kernel for model merging (Ortiz-Jimenez et al., 2023). The second equation connects existing methods to a Gaussian approximation obtained using Laplace's method.

The gradient mismatch term in Eq. 2 arises due to the ratio $p(\boldsymbol{\theta}|\mathcal{D}_t)/p(\boldsymbol{\theta})$. To understand this, consider the simple case of linear regression. Suppose we learn two separate linear models with loss function $\bar{\ell}_t(\boldsymbol{\theta}) = \frac{1}{2}\|\mathbf{y}_t - \mathbf{X}_t\boldsymbol{\theta}\|^2$. The gradient and Hessian are $\nabla\bar{\ell}_t(\boldsymbol{\theta}) = \mathbf{X}_t^\top(\mathbf{X}_t\boldsymbol{\theta} - \mathbf{y}_t)$ and $\mathbf{H}_t = \mathbf{X}_t\mathbf{X}_t^\top$ respectively. Therefore, the gradient mismatch term can be written as,

$$\nabla\bar{\ell}_t(\boldsymbol{\theta}_{1:T}) - \nabla\bar{\ell}_t(\boldsymbol{\theta}_t) = \mathbf{X}_t^\top(\mathbf{X}_t\boldsymbol{\theta}_{1:T} - \mathbf{X}_t\boldsymbol{\theta}_t) = \mathbf{H}_t(\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_t) = \nabla \log \frac{p(\boldsymbol{\theta}|\mathcal{D}_t)}{p(\boldsymbol{\theta})}\bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{1:T}}.$$

For linear models, $p_\alpha(\boldsymbol{\theta}|\mathcal{D}_t) = q_\alpha(\boldsymbol{\theta}|\mathcal{D}_t)$ and therefore Taylor's approximation

$$\nabla\bar{\ell}_t(\boldsymbol{\theta}) \approx \nabla\bar{\ell}_t(\boldsymbol{\theta}_t) + \mathbf{H}_t(\boldsymbol{\theta} - \boldsymbol{\theta}_t) \tag{21}$$

is exact. The equation matches Jin et al. (2023, Eq. 1) who use this objective to merge linear parts of a transformer. Our approach extends such efforts to nonlinear problems.

The Bayesian connection also gives direct ways to improve model merging and also reduce the computational burden. For example, one way to improve would be to take a few optimization steps aiming for the MAP estimate of the exact posterior, and then use the current iterate for the Taylor's approximation in Eq. 2. Solutions obtained this way will provably get better as the number of steps are increased. This is in contrast with other approaches, for example, by Ortiz-Jimenez et al. (2023) who propose to train in the linearized tangent space which may not always converge to the right solution. Another way to improve is to use better posterior approximation, for example, using variational inference (Graves, 2011; Blundell et al., 2015; Osawa et al., 2019) which is known to yield a more global approximation (Opper & Archambeau, 2009). Nevertheless, in this work we focus on improving merging without retraining and with computationally cheap estimates and leave the iterative optimization as future work.

The Bayesian view also connects to similar efforts in continual learning to avoid catastrophic forgetting (Kirkpatrick et al., 2017) where a Bayesian motivation is used to justify the choice of Fisher-based regularizer (Huszár, 2018). Our contribution essentially gives an extension of such ideas to model merging. Our approach is also connected to Knowledge-Adaptation priors (Khan & Swaroop, 2021) where a variety of adaptation tasks are solved by gradient reconstruction. The connection also justifies the choice of diagonal Fisher in place of the Hessian, which essentially forms a Generalized Gauss-Newton approximation (Schraudolph, 2002; Pascanu & Bengio, 2013; Martens, 2020) of it. In our experiments, we use a Monte-Carlo estimator $\sum_i [\nabla_{\boldsymbol{\theta}} \ell_i(\boldsymbol{\theta})]^2$ of the diagonal Fisher where i is summed over all examples in the data. Such estimates can also be obtained during training with Adam (Kingma & Ba, 2015) and provide a good estimate of the Hessian for small minibatch sizes (Khan et al., 2018, Thm. 1). The estimate can be normalized or unnormalized, and it is also possible to use another Fisher estimate. However, our derivation suggests to estimate it on the training data and not a held-out set as mentioned in Yadav et al. (2023).

1.6 Derivation of Model Merging as MAP of Bayes' Posterior

We will now connect our approach to Bayes' rule for linear regression. In this case, Eq. 2 can be rearranged to write as follows, where in the second term we have added and subtracted $\boldsymbol{\theta}_{1:T}$,

$$0 = -\mathbf{H}_0(\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_{\text{LLM}}) + \sum_{t=1}^{T} \alpha_t \mathbf{H}_0(\boldsymbol{\theta}_t - \boldsymbol{\theta}_{1:T} + \boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_{\text{LLM}}) - \sum_{t=1}^{T} \alpha_t \mathbf{H}_t(\boldsymbol{\theta}_{1:T} - \boldsymbol{\theta}_t).$$

This equation is a stationarity condition of the following optimization problem,

$$\boldsymbol{\theta}_{1:T} = \arg\min_{\boldsymbol{\theta}} \left(1 - \sum_{t=1}^{T} \alpha_t\right) \underbrace{\left[-\tfrac{1}{2}\|\boldsymbol{\theta} - \boldsymbol{\theta}_{\text{LLM}}\|^2_{\mathbf{H}_0}\right]}_{=\log p(\boldsymbol{\theta})} + \sum_{t=1}^{T} \alpha_t \underbrace{\left(-\tfrac{1}{2}\|\boldsymbol{\theta} - \boldsymbol{\theta}_t\|^2_{\mathbf{H}_0 + \mathbf{H}_t}\right)}_{=\log p(\boldsymbol{\theta}|\mathcal{D}_t)}.$$

where we identify the prior to be $p(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}_{\text{LLM}}, \mathbf{H}_0^{-1})$, and the posterior on $\mathcal{D}_t$ to be $p(\boldsymbol{\theta}|\mathcal{D}_t) = \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}_t, (\mathbf{H}_0 + \mathbf{H}_t)^{-1})$. We can therefore show that the stationarity condition corresponds to a maximum-a-posterior estimate of $p(\boldsymbol{\theta}|\mathcal{D}_{1:T})$ as follows,

$$p(\boldsymbol{\theta}|\mathcal{D}_{1:T}) \propto p(\boldsymbol{\theta}) \prod_{t=1}^{D} p(\mathcal{D}_t|\boldsymbol{\theta})^{\alpha_t} = p(\boldsymbol{\theta}) \prod_{t=1}^{D} \left[\frac{p(\boldsymbol{\theta}|\mathcal{D}_t)}{p(\boldsymbol{\theta})}\right]^{\alpha_t} = p(\boldsymbol{\theta})^{1-\sum_{t=1}^{T}\alpha_t} \prod_{t=1}^{T} p(\boldsymbol{\theta}|\mathcal{D}_t)^{\alpha_t},$$

where log of the last term is equivalent to the objective function.

References

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International Conference on Machine Learning (ICML)*, 2015. pages 4

R Dennis Cook. Detection of influential observation in linear regression. *Technometrics*, 19(1):15–18, 1977. pages 3

Marc Deisenroth and Jun Wei Ng. Distributed gaussian processes. In *International Conference on Machine Learning*, pp. 1481–1490. PMLR, 2015. pages 4

Hugh Durrant-Whyte. Data fusion in decentralised sensing networks. In *Fourth International Conference on Information Fusion, 2001*, 2001. pages 4

Alex Graves. Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2011. pages 4

Ferenc Huszár. Note on the quadratic penalties in elastic weight consolidation. *Proceedings of the National Academy of Sciences*, 115(11):E2496–E2497, 2018. pages 5

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models. In *International Conference on Learning Representations (ICLR)*, 2023. URL `https://openreview.net/forum?id=FCnohuR6AnM`. pages 4

Mohammad Emtiyaz Khan and Siddharth Swaroop. Knowledge-adaptation priors. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. pages 5

Mohammad Emtiyaz Khan, Didrik Nielsen, Voot Tangkaratt, Wu Lin, Yarin Gal, and Akash Srivastava. Fast and scalable bayesian deep learning by weight-perturbation in adam. In *International Conference on Machine Learning (ICML)*, 2018. pages 5

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. pages 5

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences (PNAS)*, 114(13):3521–3526, 2017. pages 5

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning (ICML)*, 2017. pages 3

Pierre-Simon Laplace. Mémoires de mathématique et de physique. *Tome Sixieme*, 1774. pages 4

David JC MacKay. A practical Bayesian framework for backpropagation networks. *Neural Computation*, 4(3):448–472, 1992. pages 4

James Martens. New insights and perspectives on the natural gradient method. *J. Mach. Learn. Res. (JMLR)*, 21(1):5776–5851, 2020. pages 5

Arthur G. O. Mutambara. *Decentralized estimation and control for multisensor systems*. Routledge, 1998. pages 4

Manfred Opper and Cédric Archambeau. The variational Gaussian approximation revisited. *Neural computation*, 21(3):786–792, 2009. pages 4

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models, 2023. URL `http://arxiv.org/abs/2305.12827`. arXiv:2305.12827. pages 4

Kazuki Osawa, Siddharth Swaroop, Mohammad Emtiyaz Khan, Anirudh Jain, Runa Eschenhagen, Richard E Turner, and Rio Yokota. Practical deep learning with Bayesian principles. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. pages 4

Razvan Pascanu and Yoshua Bengio. Revisiting natural gradient for deep networks, 2013. URL `http://arxiv.org/abs/1301.3584`. arXiv:1301.3584. pages 5

Nicol N Schraudolph. Fast curvature matrix-vector products for second-order gradient descent. *Neural Computation*, 14(7):1723–1738, 2002. pages 5

Luke Tierney and Joseph B Kadane. Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86, 1986. pages 4

Volker Tresp. A Bayesian committee machine. *Neural Computation*, 12(11):2719–2741, 2000. pages 4

Peng Wu, Tales Imbiriba, Victor Elvira, and Pau Closas. Bayesian data fusion with shared priors, 2022. URL `http://arxiv.org/abs/2212.07311`. arXiv:2212.07311. pages 4

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Resolving interference when merging models, 2023. URL `http://arxiv.org/abs/2306.01708`. arXiv:2306.01708. pages 5