On Masked Pre-training and the Marginal Likelihood

Supplementary Material

Pablo Moreno-Muñoz
Section for Cognitive Systems
Technical University of Denmark (DTU)
pabmo@dtu.dk

Pol G. Recasens
CROMAI, Barcelona Supercomputing Center
Universitat Politècnica de Catalunya (UPC)
pol.garcia@bsc.es

Søren Hauberg
Section for Cognitive Systems
Technical University of Denmark (DTU)
sohau@dtu.dk

In this appendix, we provide additional details about the theoretical and empirical results included in the manuscript. These indicate that masked pre-training optimizes according to a stochastic gradient of the model's log-marginal likelihood. We remark that our main proof relies on a previous observation from Fong and Holmes (2020), who shows that log-marginal likelihood (LML) is equivalent to an *exhaustive* cross-validation score over *all* training-test data partitions in the dataset. While Fong and Holmes' formal proof uses properties of probability to link cross-validation on observations with marginal likelihood, we use them to prove that self-conditional probabilities across masked features also lead to the model's LML. Additionally, we include the code for experiments and extra details on the tractable linear model as well as the initial setup of hyperparameters for the reproducibility of our results.

A Proof of Proposition

Proof. Consider an i.i.d. dataset $\boldsymbol{x}_{1:n}$, where each i^{th} object $\boldsymbol{x}_i \in \mathcal{X}^D$ for some continuous or discrete domain $\mathcal{X}$. We define a *latent variable model* where the likelihood function is defined as $p_\theta(\boldsymbol{x}|\boldsymbol{z})$, where $\boldsymbol{z} \in \mathcal{Z}^K$ are the latent objects for some domain $\mathcal{Z}$ and the prior distribution is $p(\boldsymbol{z})$. These assumptions lead us to a log-marginal likelihood (LML) of the model that factorises across observations, such that $\log p_\theta(\boldsymbol{x}_{1:n}) = \sum_{i=1}^n \log p_\theta(\boldsymbol{x}_i)$, where $p_\theta(\boldsymbol{x}_i) = \int p_\theta(\boldsymbol{x}_i|\boldsymbol{z}_i)p(\boldsymbol{z}_i)\mathrm{d}\boldsymbol{z}_i$.

Using the properties of probability, we can rewrite the LML of each i^{th} object as a sum of conditional distributions between dimensions or *features*. This sum is of the form

$$\log p_\theta(\boldsymbol{x}) = \sum_{t=1}^{D} \log p_\theta\left(x_t|\boldsymbol{x}_{t+1:D}\right), \tag{A.1}$$

where we omitted the i^{th} subscript to keep the notation uncluttered. Here, we see that the value of $\log p_\theta(\boldsymbol{x})$ is *invariant* to the choice of the conditional probabilities in (A.1) if these ones follow the *chain-rule* of probability according to the D dimensions of $\boldsymbol{x}$. Additionally, this indicates that we have $D!$ different choices for the sum of conditional probabilities in (A.1), which allows us to write

$$\log p_\theta(\boldsymbol{x}) = \frac{1}{D!}\sum_{\pi=1}^{D!}\sum_{t=1}^{D} \log p_\theta\left(x^{(\pi)}_{\mathcal{M}(t)}|\boldsymbol{x}^{(\pi)}_{\mathcal{M}(t+1:D)}\right). \tag{A.2}$$

Here, we defined $\mathcal{M}$ as the *indexing mask*, which consists of indices drawn from $\{1, 2, \ldots, D\}$, and we initially assume in (A.2) that $|\mathcal{M}| = D$. The π^{th} superscript indicates the *order* of indices used to produce the conditional chain-rule.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

If we then swap the order of sums in (A.2) and we fix the index (t) in (A.2), we can see that there are $(D-t+1)$ choices for the *tokens* under evaluation by the probability distribution and $\binom{D}{t-1}$ choices for the rest of conditional factors. We can then write

$$\sum_{\pi=1}^{D!} \log p_\theta \left(x^{(\pi)}_{\mathcal{M}(t)} | \boldsymbol{x}^{(\pi)}_{\mathcal{M}(t+1:D)} \right) = \sum_{\pi=1}^{\mathcal{C}_t} \sum_{j=1}^{D-t+1} \log p_\theta \left(x^{(\pi)}_{\mathcal{M}(j)} | \boldsymbol{x}^{(\pi)}_{\mathcal{M}(t+1:D)} \right).$$

To match notation with masked pre-training (MPT), we set $\mathcal{M}$ to be the *masked* subset of indices sampled from $\{1, 2, \ldots, D\}$, such that $M < D$ and the rest of *unmasked* indices shape the complementary subset $\mathcal{R} = \{1, 2, \ldots, D\} \setminus \mathcal{M}$. Using the previous sum and notation in (A.2), we can finally state that the LML is a *cumulative* sum of averages, such that

$$\log p_\theta(\boldsymbol{x}) = \sum_{t=1}^{D} \frac{1}{\mathcal{C}_t} \sum_{\pi=1}^{\mathcal{C}_t} \frac{1}{D-t+1} \sum_{j=1}^{D-t+1} \log p_\theta \left(x^{(\pi)}_{\mathcal{M}(j)} | \boldsymbol{x}^{(\pi)}_{\mathcal{R}(t+1:D)} \right). \tag{A.3}$$

Setting $M = D - t + 1$ and rearranging $\mathcal{C}_t$ as $\mathcal{C}_M = \binom{D}{D-M}$ gives us the formal result included in Proposition 1.

B Full view of Probabilistic PCA

Probabilistic PCA (PPCA) (Tipping and Bishop, 1999) is a *latent variable model* in which the marginal likelihood distribution is tractable and the maximum likelihood solution for the parameters can be analytically found. The model also assumes that the data are D-dimensional observations $\boldsymbol{x}$. Additionally, we assume that there exists a low-dimensional, where each sample has a *latent* representation $\boldsymbol{z} \in \mathcal{Z}$ for each datapoint, where $\mathcal{Z} = \mathbb{R}^K$. The relationship between the latent variables and the observed data is linear and can be expressed as

$$\boldsymbol{x} = \boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu} + \epsilon,$$

where $\epsilon \sim \mathcal{N}(0, \sigma_0^2 \mathbb{I})$, $\boldsymbol{\mu} \in \mathbb{R}^D$ and $\boldsymbol{W} \in \mathbb{R}^{D \times K}$. The likelihood model for observations $\boldsymbol{x}$ can be then written as

$$p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{W}, \boldsymbol{\mu}, \sigma_0^2) = \mathcal{N}(\boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu}, \sigma_0^2 \mathbb{I}),$$

and more importantly, it allows the integration of the latent variables in closed-form. Thus, we can obtain the following marginal likelihood per datapoint in an easy manner

$$p_\theta(\boldsymbol{x}) = p(\boldsymbol{x}|\boldsymbol{W}, \boldsymbol{\mu}, \sigma_0^2) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{W}\boldsymbol{W}^\top + \sigma_0^2 \mathbb{I}),$$

where we used $\theta = \{\boldsymbol{W}, \boldsymbol{\mu}, \sigma_0^2\}$. Moreover, under the independence assumption taken in PPCA across n observations $\boldsymbol{x}_{1:n}$, the *global* log-marginal likelihood of the model can be expressed using the following sum

$$p_\theta(\boldsymbol{x}_{1:n}) = \prod_{i=1}^{n} p_\theta(\boldsymbol{x}_i).$$

Posterior predictive probabilities. The predictive distribution between the dimensions of $\boldsymbol{x}_i$ can be obtained from both latent variable integration or by properties of Gaussian conditionals. In our case, we use the latter example. Thus, having both *mask* $\mathcal{M}$ and *rest* $\mathcal{R}$ indices according to our previous notation, we can look to the multivariate normal distribution $p_\theta(\boldsymbol{x})$ using *block* submatrices, such that

$$p_\theta(\boldsymbol{x}) = \mathcal{N}\left(\begin{bmatrix} \boldsymbol{x}_\mathcal{M} \\ \boldsymbol{x}_\mathcal{R} \end{bmatrix} \middle| \begin{bmatrix} \boldsymbol{\mu}_\mathcal{M} \\ \boldsymbol{\mu}_\mathcal{R} \end{bmatrix}, \begin{bmatrix} \boldsymbol{S}_{\mathcal{M}\mathcal{M}} & \boldsymbol{S}_{\mathcal{M}\mathcal{R}} \\ \boldsymbol{S}^\top_{\mathcal{M}\mathcal{R}} & \boldsymbol{S}_{\mathcal{R}\mathcal{R}} \end{bmatrix} \right),$$

where we also defined $\boldsymbol{S} = \boldsymbol{W}\boldsymbol{W}^\top + \sigma_0^2 \mathbb{I}$. Using the properties of conditional probabilities on normal distributions, we can write the posterior predictive densities in closed-form, such that $p_\theta(\boldsymbol{x}_\mathcal{M}|\boldsymbol{x}_\mathcal{R}) = \mathcal{N}(\boldsymbol{m}_{\mathcal{M}|\mathcal{R}}, \boldsymbol{v}_{\mathcal{M}|\mathcal{R}})$, where parameters are obtained from

$$\boldsymbol{m}_{\mathcal{M}|\mathcal{R}} = \boldsymbol{\mu}_\mathcal{M} + \boldsymbol{S}^\top_{\mathcal{M}\mathcal{R}} \boldsymbol{S}^{-1}_{\mathcal{R}\mathcal{R}} (\boldsymbol{x}_\mathcal{R} - \boldsymbol{\mu}_\mathcal{R}), \quad \boldsymbol{v}_{\mathcal{M}|\mathcal{R}} = \boldsymbol{S}_{\mathcal{M}\mathcal{M}} + \boldsymbol{S}^\top_{\mathcal{M}\mathcal{R}} \boldsymbol{S}^{-1}_{\mathcal{R}\mathcal{R}} \boldsymbol{S}_{\mathcal{M}\mathcal{R}}.$$

C Experiments

The code for experiments is written in Python 3.9 and uses the Pytorch syntax for the automatic differentiation of the models. It can be found in the repository `https:github.com/pmorenoz/MPT-LML`, where we also included the *scripts* used to evaluate BERT (Devlin et al., 2018) and the area under the MPT curve for different masking rates and test subsets. All figures included in the manuscript are reproducible and we also provide *seeds*, the setup of learning hyperparameters as well as the initial values of parameters in the tractable model.

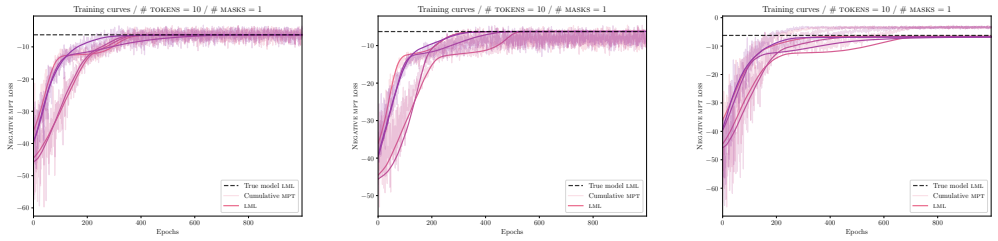
C.1 Longer discussion on the role of masking rates.

The fact that fixed size *held-out* sets induce a *biased* estimation of the marginal likelihood in cross-validation was previously observed in Moreno-Muñoz et al. (2022) and Fong and Holmes (2020). The former used this property to characterize stochastic approximations in Gaussian process models. On the other hand, the latter identified this effect as a result of the non-uniform sampling of the size of the held-out sets, e.g. setting it fixed, which in their formal results led to the biased estimate of the *cumulative cross-validation* term included in the Appendix.

In this paper, we make a similar observation on the *biased* estimation of the marginal likelihood in masked pre-training, where we are fixing the masking rate instead. Our empirical results with tractable models allow us to accurately identify this bias and prove that it does not affect the maximisation of the LML. Mainly, due to the bias is fixed during optimization (see Fig. 2). One important detail to consider is that this bias can be computed for tractable models, as it is the *expected* log-marginal likelihood on the *unmasked* tokens of the data.

C.2 Datasets

Our experiments make use of three well-known datasets: MNIST (LeCun et al., 1998), FMNIST (Xiao et al., 2017) and GLUE (Wang et al., 2019). The datasets MNIST and FMNIST were downloaded from the `torchvision` repository included in the Pytorch library. GLUE can be accessed via the public repository at `https://github.com/nyu-mll/GLUE-baselines` or `https://gluebenchmark.com/`. These particular datasets are not subject to use constraints related to our experiments or they include licenses which allow their use for research purposes.



(a) UNIFORM SAMPLING ($0\% - 100\%$) (b) UNIFORM SAMPLING ($50\% - 100\%$) (c) UNIFORM SAMPLING ($0\% - 50\%$)

Figure 1: Training curves of the negative cumulative MPT loss in PPCA vs. the ground truth (GT) LML. The number of samples is $N = 2000$ and the number of tokens is $D = 10$. All plots used $P = 1$ per epoch and five different initializations. **(Left).** The rate of masking is *unfixed* and it varies according to the range between 1% and 100%. It is obtained at each epoch via uniform sampling. **(Center)**. The masking rate is uniformly samples in the range between 50% and 100%. **(Right)**. The masking rate is uniformly samples in the range between 0% and 50%.

C.3 Additional results on uniform masking rate sampling

The aim of the experiments is to answer the question around the effect of *uniformly* sampling with MPT losses. So far, we have observed that the cumulative MPT is equivalent to an *unbiased* estimate of the log-marginal likelihood when we consider all possible numbers for the amount of masked tokens. To avoid having a *biased* estimate when fixing the masking rate (e.g., to 20%), one option

is to use an uniform distribution. In this way, we sampled the rate of *masking* at each epoch in the range $(0\% - 100\%)$ as it's shown in Fig. 1a. These results indicate that we are able to obtain such *unbiased* target losses. Importantly, we should also notice that the cumulative losses oscillate around the true value of the LML, that is being also maximised. For completeness of the experiments, we also included the empirical results when the masking rate is sampled in different ranges (e.g., $0\% - 50\%$). In this case, we have two different *biases* in the losses shows in Fig. 1b and Fig. 1c. These biases are related to the areas under the curves described in Sec. 3.2 and Fig. 4 of the main paper.

References

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

E. Fong and C. C. Holmes. On the marginal likelihood and cross-validation. *Biometrika*, 107(2):489–496, 2020.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

P. Moreno-Muñoz, C. W. Feldager, and S. Hauberg. Revisiting active sets for gaussian process decoders. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.

A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *International Conference on Learning Representations (ICLR)*, 2019.

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.