On diagonal approximations to the extended Kalman filter for online training of Bayesian neural networks

Peter G. Chang PETERCHANG@CHICAGOBOOTH.EDU *U. Chicago*
Matt Jones MCJONES@GOOGLE.COM *U. Colorado, Google Research*
Kevin Murphy KPMURPHY@GOOGLE.COM *Google Research*

Abstract

We present two approaches to approximate online Bayesian inference for the parameters of DNNs. Both are based on diagonal Gaussian approximations and linearize the network at each step to ensure efficient computation. The first approach optimizes the exclusive KL, $D_{\mathbb{KL}}(q \parallel p)$; this amounts to matching the marginal mean and *precision* of p and q. The second approach optimizes the inclusive KL, $D_{\mathbb{KL}}(q \parallel p)$, which amounts to matching the marginal mean and *variance* of p and q. The latter approach turns out to be equivalent to the previously proposed "fully decoupled EKF" approach. We show experimentally that exclusive KL is more effective than both inclusive KL and one-pass SGD.

Keywords: Bayesian inference, variational inference, online learning, extended Kalman filter, deep neural networks, non-stationary distributions, continual learning.

1. Introduction

In this short paper, we consider the problem of online Bayesian inference for the parameters of a neural network. That is, we want to recursively compute $p(\boldsymbol{\theta}_t|\mathbf{y}_{1:t}, \mathbf{x}_{1:t})$ in an efficient manner, where $\mathbf{x}_t \in \mathbb{R}^{N_x}$ are the input features, $\mathbf{y}_t \in \mathbb{R}^{N_y}$ are the output labels, and $\boldsymbol{\theta}_t \in \mathbb{R}^{N_z}$ are the model parameters. We assume the likelihood has the form $p(\mathbf{y}_t|\boldsymbol{\theta}_t, \mathbf{x}_t) = \text{expfam}(\mathbf{y}_t|g^{-1}(h(\mathbf{x}_t, \boldsymbol{\theta}_t)))$, where h is a neural network that predicts the natural parameters $\boldsymbol{\eta}$ of the exponential family output, and g is some link function that maps from natural parameters $\boldsymbol{\eta}$ to moment parameters $\mathbf{m}$. To handle non-stationary distributions, we allow the parameters to drift according to a Gaussian random walk, so $p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}) = N(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}, \mathbf{Q}_t)$, where $\mathbf{Q}_t = \text{diag}(\mathbf{q}_t)$ is the process noise. For continual learning problems where a single task drifts over time, this method tracks the optimal parameters and their uncertainty. If instead we assume $\mathbf{q}_t = 0$, so the parameters are stationary, then the method can also be used to compute $p(\boldsymbol{\theta}|\mathbf{x}_{1:T}, \mathbf{y}_{1:T})$ with a single pass over the data.

Exact Bayesian inference in this model family is intractable, so we must resort to approximations. In the signal processing and engineering communities, a standard approach (e.g., Singhal and Wu, 1989; Puskorius and Feldkamp, 2003) is the extended Kalman filter (EKF), which amounts to linearizing the observation model h at each step, and then performing an exact Bayesian update. In the machine learning literature, it is more common (e.g., Challis and Barber, 2013; Broderick et al., 2013; Nguyen et al., 2018; Lambert et al., 2021b) to solve the following variational inference problem, using exclusive KL divergence:

$$q_t = \arg\min_q D_{\mathbb{KL}}(q(\boldsymbol{\theta}_t) \parallel p(\boldsymbol{\theta}_t|\mathbf{y}_{1:t})) \approx \arg\min_q D_{\mathbb{KL}}(q(\boldsymbol{\theta}_t) \parallel p(\mathbf{y}_t|\boldsymbol{\theta}_t) q_{t|t-1}(\boldsymbol{\theta}_t)) \tag{1}$$

where we omit the conditioning on $\mathbf{x}_t$ for brevity, and where the one-step-ahead predictive distribution is given by

$$q_{t|t-1}(\boldsymbol{\theta}_t) = \int p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1})q_{t-1}(\boldsymbol{\theta}_{t-1})d\boldsymbol{\theta}_t = \mathcal{N}(\boldsymbol{\theta}_t|\boldsymbol{\mu}_{t|t-1}, \boldsymbol{\Sigma}_{t|t-1}) \tag{2}$$

A third approach (e.g., Hernández-Lobato and Adams, 2015; Soudry et al., 2014; Ghosh et al., 2016) based on assumed density filtering (ADF), minimizes the inclusive KL divergence:

$$q_t = \arg\min_q D_{\mathbb{KL}}(p(\boldsymbol{\theta}_t|\mathbf{y}_{1:t}) \parallel q(\boldsymbol{\theta}_t)) \tag{3}$$

In this paper, we discuss efficient and deterministic methods to optimize both of these KL objectives in $O(N_z)$ time. To do this, we use two approximations: first, we use (block) diagonal Gaussian approximations of the form $q_t(\boldsymbol{\theta}_t) = \mathcal{N}(\boldsymbol{\theta}_t|\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$, where $\boldsymbol{\Sigma}_t = \text{diag}(\boldsymbol{\Sigma}_t^{ii})$; second, we use the EKF approximation to locally linearize the observation model at each step. When we apply this approach to the exclusive KL objective, the resulting algorithm is a novel diagonal variant of the mean field VI method (Blundell et al., 2015). When we apply this approach to the inclusive KL objective, we recover the “decoupled EKF” method (Puskorius and Feldkamp, 1991, 2003; Murtuza and Chorian, 1994).[1]

In Section 2, we describe our inference methods in more detail, and in Section 3, we perform an experimental comparison. We show that the exclusive KL objective is more statistically efficient than both inclusive KL and stochastic gradient descent (SGD), which requires multiple passes over the data, making it harder to apply to the streaming setting. We conclude in Section 4. Our code is available at `https://github.com/probml/dynamax/tree/main/dynamax/generalized_gaussian_ssm/dekf`.

2. Methods

In this section, we briefly describe our methods. For the derivation, please see the appendix. (We use the notation $\mathbf{A}_i$ and $\mathbf{A}^i$ to represent the i^{th} row and column of a matrix $\mathbf{A}$, and $\mathbf{A}^{ii}$ to denote the i^{th} (block) diagonal entry, if $\mathbf{A}$ is square.)

Refer to Appendix A.6 for the different ways we can compute the posterior predictive distribution.

2.1. Extended Kalman filter

For completeness, we summarize the usual EKF procedure, specialized to our setting. For notational simplicity, we drop the conditioning on inputs $\mathbf{x}_t$.

The predict step is given by Eq. (2), where $\boldsymbol{\mu}_{t|t-1} = \boldsymbol{\mu}_{t-1}$ and $\boldsymbol{\Sigma}_{t|t-1} = \boldsymbol{\Sigma}_{t-1} + \mathbf{Q}_t$.

[1] We can extend the diagonal EKF + inclusive KL approach to the offline setting by using repeated forward-backward passes; this gives a diagonal version of the method in Kamthe et al. (2022), which approximates the local expectation propagation updates by using linearization. However, in this paper, we focus on the online setting.

If we assume a Gaussian likelihood, $p(\mathbf{y}_t|\boldsymbol{\theta}_t) = \mathcal{N}(\mathbf{y}_t|h(\boldsymbol{\theta}_t), \mathbf{R}_t)$, then the update step becomes $p(\boldsymbol{\theta}_t|\mathbf{y}_{1:t}) = N(\boldsymbol{\theta}_t|\boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t})$ where

$$\begin{aligned}\boldsymbol{\mu}_{t|t} &= \boldsymbol{\mu}_{t|t-1} + \mathbf{K}_t(\mathbf{y}_t - h(\boldsymbol{\mu}_{t|t-1})) && (4)\\ \mathbf{S}_t &= \mathbf{R}_t + \mathbf{H}_t\boldsymbol{\Sigma}_{t|t-1}\mathbf{H}_t^{\mathsf{T}} && (5)\\ \mathbf{K}_t &= \boldsymbol{\Sigma}_{t|t-1}\mathbf{H}_t^{\mathsf{T}}\mathbf{S}_t^{-1} && (6)\\ \boldsymbol{\Sigma}_{t|t} &= \boldsymbol{\Sigma}_{t|t-1} - \mathbf{K}_t\mathbf{H}_t\boldsymbol{\Sigma}_{t|t-1} = \boldsymbol{\Sigma}_{t|t-1} - \boldsymbol{\Sigma}_{t|t-1}\mathbf{H}_t^{\mathsf{T}}\mathbf{S}_t^{-1}\mathbf{H}_t\boldsymbol{\Sigma}_{t|t-1} && (7)\end{aligned}$$

where $\mathbf{H}_t \equiv \text{Jac}(h)(\boldsymbol{\mu}_{t|t-1})$ is the $N_y \times N_z$ Jacobian matrix of the observation model and $\mathbf{K}_t$ is the $N_z \times N_y$ Kalman gain matrix.

We now derive an alternative update form that we will use below. The **Woodbury matrix inversion lemma** states the following:

$$(\mathbf{A} + \mathbf{UCV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{C}^{-1} + \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1} \tag{8}$$

Letting $\mathbf{A} = \boldsymbol{\Sigma}_{t|t-1}^{-1}, \mathbf{U} = \mathbf{H}_t^{\mathsf{T}}, \mathbf{C} = \mathbf{R}_t^{-1}, \mathbf{V} = \mathbf{H}_t$ and applying the inversion lemma gives the covariance update in information form, as follows:

$$\boldsymbol{\Sigma}_{t|t}^{-1} = \boldsymbol{\Sigma}_{t|t-1}^{-1} + \mathbf{H}_t^{\mathsf{T}}\mathbf{R}_t^{-1}\mathbf{H}_t \tag{9}$$

2.2. Proposed diagonal approximations

Let $p = p(\boldsymbol{\theta}_t|\mathbf{y}_{1:t}) = \mathcal{N}(\boldsymbol{\theta}_t|\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$ be the full covariance computed by EKF, where $\boldsymbol{\mu}_p = \boldsymbol{\mu}_{t|t}$ and $\boldsymbol{\Sigma}_p = \boldsymbol{\Sigma}_{t|t}$, as in Section 2.1. In general the EKF takes $O(N_y^3 + N_z^2)$ time to compute, due to the need to invert $\mathbf{S}_t \in \mathbb{R}^{N_y \times N_y}$ and then compute the $N_z \times N_z$ matrix $\mathbf{K}_t\mathbf{H}_t\boldsymbol{\Sigma}_{t|t-1}$. We seek to approximate this in $O(N_z)$ time by computing a diagonal approximation, $q = \mathcal{N}(\boldsymbol{\theta}_t|\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)$, where $\boldsymbol{\Sigma}_q = \text{diag}(\boldsymbol{\Sigma}^{ii})$.

To derive our method, first note that if the previous posterior $\boldsymbol{\Sigma}_{t-1}$ is (block) diagonal, then the prior predictive covariance for the next state is also diagonal, $\boldsymbol{\Sigma}_{t|t-1}^{ii} = \boldsymbol{\Sigma}_{t-1}^{ii} + \mathbf{Q}_t^{ii}$. Also, the posterior covariance after the next observation can be efficiently computed using

$$\mathbf{S}_t = \mathbf{R}_t + \sum_j \mathbf{H}_t^j\boldsymbol{\Sigma}_{t|t-1}^{jj}(\mathbf{H}_t^j)^{\mathsf{T}} \tag{10}$$

From Eq. (7), the blocks of the (exact) EKF posterior covariance matrix are as follows:

$$\boldsymbol{\Sigma}_{t|t}^{ij} = \boldsymbol{\Sigma}_{t|t-1}^{ij} - \boldsymbol{\Sigma}_{t|t-1}^{ii}(\mathbf{H}_t^i)^{\mathsf{T}}\mathbf{S}_t^{-1}\mathbf{H}_t^j\boldsymbol{\Sigma}_{t|t-1}^{jj} \tag{11}$$

The above equations tell us how to compute p. Now suppose we want to approximate this by a diagonal q so as to minimize $D_{\mathbb{KL}}(p \parallel q)$, which is mode covering. In the appendix we show that the result is given by setting $\boldsymbol{\mu}_q = \boldsymbol{\mu}_p$, and by zeroing out the off-diagonal blocks of the posterior covariance, $\boldsymbol{\Sigma}_{t|t}^{ij}$ for $i \neq j$. From Eq. (11) this gives

$$\boldsymbol{\Sigma}_{t|t}^{ii} = \boldsymbol{\Sigma}_{t|t-1}^{ii} - \boldsymbol{\Sigma}_{t|t-1}^{ii}(\mathbf{H}_t^i)^{\mathsf{T}}\left(\mathbf{R}_t + \sum_j \mathbf{H}_t^j\boldsymbol{\Sigma}_{t|t-1}^{jj}(\mathbf{H}_t^j)^{\mathsf{T}}\right)^{-1}\mathbf{H}_t^i\boldsymbol{\Sigma}_{t|t-1}^{ii} \tag{12}$$

This is equivalent to the fully decoupled EKF method of Puskorius and Feldkamp (2003); we therefore call this the "FD-EKF" method.

Now suppose we want to minimize $D_{\mathbb{KL}}(q \parallel p)$, which is mode seeking. We call this the "variational diagonal EKF" or "VD-EKF" method. In the appendix we show that the result is given by setting $\boldsymbol{\mu}_q = \boldsymbol{\mu}_p$ and by zeroing out the off-diagonal elements of the posterior *precision*, $\boldsymbol{\Sigma}_{t|t}^{-1}$. From Eq. (9) this gives

$$(\boldsymbol{\Sigma}_{t|t}^{ii})^{-1} = \left(\boldsymbol{\Sigma}_{t|t-1}^{ii}\right)^{-1} + (\mathbf{H}_t^i)^\mathsf{T} \mathbf{R}_t^{-1} \mathbf{H}_t^i \tag{13}$$

By using the matrix inversion lemma, we can also write the result as follows:

$$\boldsymbol{\Sigma}_{t|t}^{ii} = \boldsymbol{\Sigma}_{t|t-1}^{ii} - \boldsymbol{\Sigma}_{t|t-1}^{ii}(\mathbf{H}_t^i)^\mathsf{T} \left(\mathbf{R}_t + \mathbf{H}_t^i \boldsymbol{\Sigma}_{t|t-1}^{ii}(\mathbf{H}_t^i)^\mathsf{T}\right)^{-1} \mathbf{H}_t^i \boldsymbol{\Sigma}_{t|t-1}^{ii} \tag{14}$$

We see that the main difference between the inclusive KL in Eq. (12) and the exclusive KL in Eq. (14) is that the former uses the full $\mathbf{S}_t$ matrix, whereas the latter uses $\mathbf{S}_t^i := \mathbf{R}_t + \mathbf{H}_t^i \boldsymbol{\Sigma}_{t|t-1}^{ii}(\mathbf{H}_t^i)^\mathsf{T}$. Intuitively, using $\mathbf{S}_t$ instead of $\mathbf{S}_t^i$ captures more uncertainty. In particular, from the perspective of estimating $\boldsymbol{\theta}^i$, the missing terms in $\mathbf{S}_t^i$, namely $\sum_{j \neq i} \mathbf{H}_t^j \boldsymbol{\Sigma}_{t|t-1}^{jj}(\mathbf{H}_t^j)^\mathsf{T}$, play the same role as the observation covariance $\mathbf{R}_t$, but reflect uncertainty in the other parameters.

2.3. Handling non-Gaussian observations

To handle non-Gaussian observations, such as discrete labels, we follow the technique of Ollivier (2018) and Tronarp et al. (2018). Specifically, we assume $p(\mathbf{y}_t|\boldsymbol{\theta}_t) = \text{expfam}(\mathbf{y}_t|\mathbf{m}_t)$ is an exponential family distribution with mean parameter $\mathbf{m}_t = g^{-1}(\boldsymbol{\eta}_t)$, where $\boldsymbol{\eta}_t = h(\mathbf{x}_t, \boldsymbol{\theta}_t)$ are the natural parameters (e.g., for classification, $g^{-1} = \mathcal{S}$ is the softmax transform and $\boldsymbol{\eta}_t$ are the logits). We define the predicted observation to be $\hat{\mathbf{y}}_t = E[\mathbf{m}_t|\mathbf{y}_{1:t-1}] = h(\boldsymbol{\mu}_{t|t-1})$, and the error (innovation) term to be $\mathbf{e}_t = T(\mathbf{y}_t) - \hat{\mathbf{y}}_t$ where $T(\mathbf{y}_t)$ is the sufficient statistics vector. We define the observation covariance to be $\mathbf{R}_t = \text{Cov}\left[T(\mathbf{Y})|\hat{\mathbf{y}}_t\right]$.

In the case of the categorical distribution with K labels, we have $T(y_t) = [\mathbb{I}(y_t = 1), \ldots, \mathbb{I}(y_t = K)]$, which is the one-hot encoding. Similarly, we have $\hat{\mathbf{y}}_t = [p_t^1, \ldots, p_t^K]$, where $p_t^k = e^{\eta_t^k}/(\sum_j e^{\eta_t^j})$ are the softmax probabilities, and $\boldsymbol{\eta}_t$ are the logits. Finally, we have $\mathbf{R}_t = \text{diag}(\mathbf{p}_t) - \mathbf{p}_t\mathbf{p}_t^\mathsf{T}$. To avoid numerical problems with the sum-to-one constraint on $T(y_t)$, $\mathbf{p}_t$ and $\mathbf{R}_t$, we can drop the last row/column. Alternatively, we can replace matrix inverses with least squares solvers, which work even if their argument is singular. For example, we can compute $\mathbf{R}^{-1}$ using `jnp.linalg.lstsq(R, jnp.eye(D))[0]`.

3. Experimental results

In this section, we provide an experimental comparison of VD-EKF, FD-EKF and SGD for fitting a small LeNet CNN model to MNIST. We approximate the posterior predictive distribution by using a Monte Carlo approximation (see Appendix A.6 for details). The results are shown in Fig. 1. We see that FD-EKF is similar to SGD, but that VD-EKF is consistently better than both. This is perhaps surprising given that VD-EKF is arguably a "less accurate" posterior approximation, since it ignores more outcome variance when

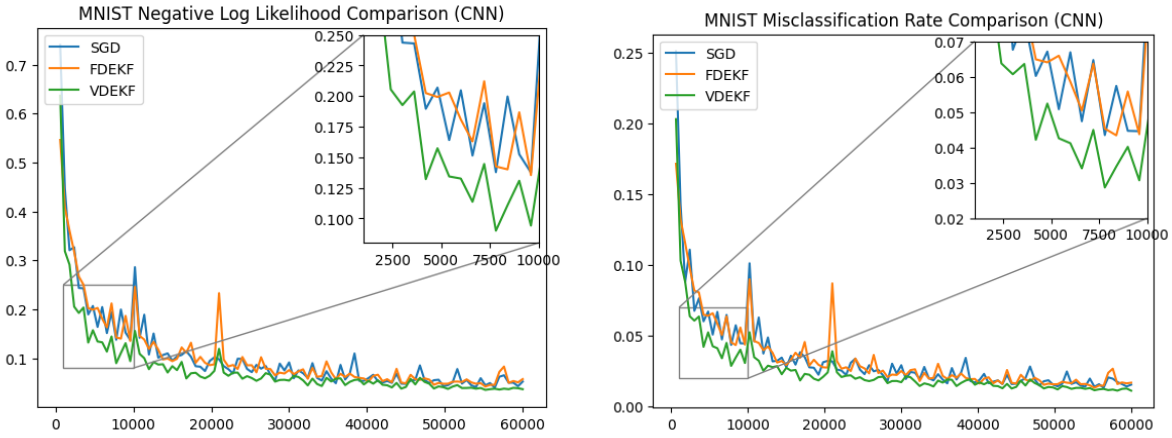


Figure 1: Comparison of negative log-likelihood values (left) and misclassification rates (right), using a CNN trained on MNIST dataset. FD-EKF and VD-EKF values are computed using Monte Carlo predictive distribution with sample size 100.

computing $\mathbf{S}_t$. However, we conjecture that the mode seeking behavior of VD-EKF results in more robust estimates than the mode-covering behavior of FD-EKF. See A.7 for more thorough experimental results.

4. Conclusion and future work

We have shown how we can efficiently compute a diagonal approximation to the posterior by linearizing the likelihood model, as in the EKF, and then optimizing $D_{\mathbb{KL}}(q\|p)$ or $D_{\mathbb{KL}}(p\|q)$ deterministically in closed form. Perhaps surprisingly, we find that the former objective does better.

In the future, we would like to extend this approach to include more expressive approximations, such as diagonal plus low rank (e.g., see Mishkin et al., 2018; Lambert et al., 2021a). We also want to apply it to more complex non-stationary and continual learning problems, which may be achievable with more sophisticated priors for how the weights evolve over time. Finally we want to compare to online SGD variants that use replay buffers, such as Hu et al. (2021).

References

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. In *ICML*, 2015. URL http://arxiv.org/abs/1505.05424.

Tamara Broderick, Nicholas Boyd, Andre Wibisono, Ashia C Wilson, and Michael I Jordan. Streaming variational bayes. In *NIPS*, 2013. URL http://arxiv.org/abs/1307.6769.

E Challis and D Barber. Gaussian Kullback-Leibler approximate inference. *JMLR*, 14:2239–2286, 2013. URL https://www.jmlr.org/papers/volume14/challis13a/challis13a.pdf.

Soumya Ghosh, Francesco Maria Delle Fave, and Jonathan Yedidia. Assumed density filtering methods for learning bayesian neural networks. In *AAAI*, 2016. URL https://jonathanyedidia.files.wordpress.com/2012/01/assumeddensityfilteringaaai2016final.pdf.

José Miguel Hernández-Lobato and Ryan P Adams. Probabilistic backpropagation for scalable learning of bayesian neural networks. In *ICML*, 2015. URL http://arxiv.org/abs/1502.05336.

Huiyi Hu, Ang Li, Daniele Calandriello, and Dilan Gorur. One pass ImageNet. In *NeurIPS 2021 Workshop on Imagenet: past, present and future*, November 2021. URL http://arxiv.org/abs/2111.01956.

Alexander Immer, Maciej Korzepa, and Matthias Bauer. Improving predictions of bayesian neural nets via local linearization. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, pages 703–711. PMLR, 2021. URL https://proceedings.mlr.press/v130/immer21a.html.

Sanket Kamthe, So Takao, Shakir Mohamed, and Marc Peter Deisenroth. Iterative state estimation in non-linear dynamical systems using approximate expectation propagation. *Trans. on Machine Learning Research*, 2022. URL https://openreview.net/pdf?id=xyt4wfdo4J.

Marc Lambert, Silvère Bonnabel, and Francis Bach. The limited-memory recursive variational gaussian approximation (L-RVGA). December 2021a. URL https://hal.inria.fr/hal-03501920.

Marc Lambert, Silvère Bonnabel, and Francis Bach. The recursive variational gaussian approximation (R-VGA). *Stat. Comput.*, 32(1):10, December 2021b. URL https://hal.inria.fr/hal-03086627/document.

Aaron Mishkin, Frederik Kunstner, Didrik Nielsen, Mark Schmidt, and Mohammad Emtiyaz Khan. SLANG: Fast structured covariance approximations for bayesian deep learning with natural gradient. In *NIPS*, pages 6245–6255. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/7862-slang-fast-structured-covariance-approximations-for-bayesian-deep-learning-with-natu
pdf.

S Murtuza and S F Chorian. Node decoupled extended kalman filter based learning algorithm for neural networks. In *Proceedings of 1994 9th IEEE International Symposium on Intelligent Control*, pages 364–369, August 1994. URL http://dx.doi.org/10.1109/ISIC.1994.367790.

Cuong V Nguyen, Yingzhen Li, Thang D Bui, and Richard E Turner. Variational continual learning. In *ICLR*, 2018. URL https://openreview.net/forum?id=BkQqq0gRb.

Yann Ollivier. Online natural gradient as a kalman filter. *Electron. J. Stat.*, 12(2):2930–2961, 2018. URL https://projecteuclid.org/euclid.ejs/1537257630.

G V Puskorius and L A Feldkamp. Decoupled extended kalman filter training of feedforward layered networks. In *International Joint Conference on Neural Networks*, volume i, pages 771–777 vol.1, 1991. URL http://dx.doi.org/10.1109/IJCNN.1991.155276.

Gintaras V Puskorius and Lee A Feldkamp. Parameter-based kalman filter training: Theory and implementation. In Simon Haykin, editor, *Kalman Filtering and Neural Networks*, pages 23–67. John Wiley & Sons, Inc., 2003. URL https://onlinelibrary.wiley.com/doi/10.1002/0471221546.ch2.

Sharad Singhal and Lance Wu. Training multilayer perceptrons with the extended kalman algorithm. In *NIPS*, volume 1, 1989. URL https://proceedings.neurips.cc/paper/1988/file/38b3eff8baf56627478ec76a704e9b52-Paper.pdf.

Daniel Soudry, Itay Hubara, and Ron Meir. Expectation backpropagation: Parameter-free training of multilayer neural networks with continuous or discrete weights. In *NIPS*, 2014. URL https://papers.nips.cc/paper/2014/file/076a0c97d09cf1a0ec3e19c7f2529f2b-Paper.pdf.

Filip Tronarp, Ángel F García-Fernández, and Simo Särkkä. Iterative filtering and smoothing in nonlinear and Non-Gaussian systems using conditional moments. *IEEE Signal Process. Lett.*, 25(3):408–412, 2018. URL https://acris.aalto.fi/ws/portalfiles/portal/17669270/cm_parapub.pdf.

Appendix A. Appendix

In this section, we give the derivation of the algorithms and more experimental results.

A.1. Objectives

Let $\boldsymbol{\mu}_p = \boldsymbol{\mu}_{t|t}$ and $\boldsymbol{\Sigma}_p = \boldsymbol{\Sigma}_{t|t}$ be the exact Gaussian posterior computed by EKF. In the VD-EKF, we want to find the approximation $q \sim \mathcal{N}(\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)$, where $\boldsymbol{\Sigma}_q$ is diagonal, which minimizes $D_{\mathbb{KL}}(q \parallel p)$, given by

$$D_{\mathbb{KL}}(q \parallel p) = \frac{1}{2}\left(\log\frac{|\boldsymbol{\Sigma}_p|}{|\boldsymbol{\Sigma}_q|} - D + \left(\boldsymbol{\mu}_q - \boldsymbol{\mu}_p\right)^{\mathsf{T}} \boldsymbol{\Sigma}_p \left(\boldsymbol{\mu}_q - \boldsymbol{\mu}_p\right) + \operatorname{tr}\left\{\boldsymbol{\Sigma}_p^{-1}\boldsymbol{\Sigma}_q\right\}\right) \tag{15}$$

In FD-EKF, we want to find the approximation that minimizes $D_{\mathbb{KL}}(p \parallel q)$, given by

$$D_{\mathbb{KL}}(p \parallel q) = \frac{1}{2}\left(\log\frac{|\boldsymbol{\Sigma}_q|}{|\boldsymbol{\Sigma}_p|} - D + \left(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q\right)^{\mathsf{T}} \boldsymbol{\Sigma}_q \left(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q\right) + \operatorname{tr}\left\{\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_p\right\}\right) \tag{16}$$

A.2. Solving for the mean

In both cases, the optimal solution for the mean is to set $\boldsymbol{\mu}_q = \boldsymbol{\mu}_p$. We can verify this by setting the partial derivative with respect to the mean elements $\mu_{q,i}$ to zero. The optimal $\boldsymbol{\mu}_p$ is given by

$$\boldsymbol{\mu}_{t|t} = \boldsymbol{\mu}_{t|t-1} + \mathbf{K}_t \left(\mathbf{y}_t - h(\boldsymbol{\mu}_{t|t-1}) \right) \tag{17}$$

where the Kalman gain from Eq. (6) is given by

$$\begin{aligned} \mathbf{K}_t &= \boldsymbol{\Sigma}_{t|t-1} \mathbf{H}_t^\intercal \left(\mathbf{R}_t + \mathbf{H}_t \boldsymbol{\Sigma}_{t|t-1} \mathbf{H}_t^\intercal \right)^{-1} & (18) \\ &= \boldsymbol{\Sigma}_{t|t-1} \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} \left(\mathbf{I} + \mathbf{H}_t \boldsymbol{\Sigma}_{t|t-1} \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} \right)^{-1} & (19) \end{aligned}$$

We now use the **push-through identity**, which states the following:

$$\mathbf{U}(\mathbf{I} + \mathbf{V}\mathbf{U})^{-1} = (\mathbf{I} + \mathbf{U}\mathbf{V})^{-1}\mathbf{U} \tag{20}$$

Letting $\mathbf{U} = \boldsymbol{\Sigma}_{t|t-1} \mathbf{H}_t^\intercal \mathbf{R}_t^{-1}$ and $\mathbf{V} = \mathbf{H}_t$ and applying the identity to Eq. (19):

$$\begin{aligned} \mathbf{K}_t &= \left(\mathbf{I} + \boldsymbol{\Sigma}_{t|t-1} \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} \mathbf{H}_t \right)^{-1} \boldsymbol{\Sigma}_{t|t-1} \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} & (21) \\ &= \left(\boldsymbol{\Sigma}_{t|t-1}^{-1} + \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} \mathbf{H}_t \right)^{-1} \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} & (22) \\ &= \boldsymbol{\Sigma}_{t|t} \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} & (23) \\ &\approx \operatorname{diag}\left(\left(\sigma_{t|t}^i \right)^2 \right) \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} & (24) \end{aligned}$$

In summary, the mean update is given by

$$\boldsymbol{\mu}_{t|t} = \boldsymbol{\mu}_{t|t-1} + \operatorname{diag}\left(\left(\sigma_{t|t}^i \right)^2 \right) \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} \left(\mathbf{y}_t - h(\boldsymbol{\mu}_{t|t-1}) \right) \tag{25}$$

A.3. Solving for the covariance

We next plug the optimal mean $\boldsymbol{\mu}_q = \boldsymbol{\mu}_p$ into the KL objectives and solve the resulting simplified objectives for $\boldsymbol{\Sigma}_q$. The result will be denoted by

$$\boldsymbol{\Sigma}_{t|t} = \operatorname{diag}\left(\left(\sigma_{t|t}^i \right)^2 \right) = \operatorname{diag}(\Sigma_{t|t}^{ii}) \tag{26}$$

where the form of $\sigma_{t|t}^i$ will be given below.

A.4. Variational diagonal EKF (reverse KL)

In this section, we use the reverse (exclusive) KL. We plug in the optimal $\boldsymbol{\mu}_q$ to get the following simplified objective for $\boldsymbol{\Sigma}_q$:

$$J(q) = \frac{1}{2} \left(-\log |\boldsymbol{\Sigma}_q| - \text{const} + \sum_j \left[\boldsymbol{\Sigma}_p^{-1} \boldsymbol{\Sigma}_q \right]_{jj} \right) \tag{27}$$

Suppose we chose $\mathbf{\Sigma}_q = \mathrm{diag}(\sigma_i^2)$. Then we get

$$J(q) = \frac{1}{2}\left(-\sum_j \log \sigma_j^2 - \mathrm{const} + \sum_j \left[\sigma_j^2 \left(\mathbf{\Sigma}_p^{-1}\right)_{jj}\right]\right) \tag{28}$$

Setting $\frac{\partial J(q)}{\partial \sigma_i} = 0$, we get

$$\frac{\partial J(q)}{\partial \sigma_i} = -\frac{1}{\sigma_i} + \sigma_i \left(\mathbf{\Sigma}_p^{-1}\right)_{ii} = 0 \tag{29}$$

which gives

$$\sigma_i^{-2} = \left(\mathbf{\Sigma}_p^{-1}\right)_{ii} \tag{30}$$

which says that we should match the marginal precisions of the two distributions.

Applying Eq. (30) to Eq. (9) we get

$$\begin{aligned}\left(\sigma_{t|t}^i\right)^{-2} &= \left(\mathbf{\Sigma}_{t|t-1}^{-1} + \mathbf{H}_t^\intercal \mathbf{R}_t^{-1} \mathbf{H}_t\right)_{ii} &(31)\\ &= \left(\sigma_{t|t-1}^i\right)^{-2} + \left(\mathbf{H}_t^\intercal \mathbf{R}_t^{-1} \mathbf{H}_t\right)_{ii} &(32)\end{aligned}$$

and therefore the posterior covariance has the following form, matching Eq. (13):

$$\mathbf{\Sigma}_{t|t} = \mathrm{diag}\left(\left[\left(\Sigma_{t|t}^{ii}\right)^{-1} + \left(\mathbf{H}_t^\intercal \mathbf{R}_t^{-1} \mathbf{H}_t\right)_{ii}\right]^{-1}\right) \tag{33}$$

We now discuss the time complexity of this update. Naively computing the diagonal matrix $\left(\mathbf{H}_t^\intercal \mathbf{R}_t^{-1} \mathbf{H}_t\right)_{ii}$ takes $O(N_z^2 N_y + N_z N_y^2 + N_y^3)$ time. However, we can leverage the following result: if $\mathbf{A} \in \mathbb{R}^{n \times m}$ and $\mathbf{B} \in \mathbb{R}^{m \times n}$, then the diagonal elements can be computed in $O(nm)$ time using

$$\left(\mathbf{AB}\right)_{ii} = \sum_{j=1}^m A_{ij} B_{ji} = (\mathbf{A}^i)^\intercal \mathbf{B}_i \tag{34}$$

Hence we can first compute $\mathbf{A} = \mathbf{H}_t^\intercal \mathbf{R}_t^{-1}$ in $O(N_z N_y^2 + N_y^3)$ time, and then compute $(\mathbf{AB})_{ii}$ in an additional $O(N_z N_y)$ time, where $\mathbf{B} = \mathbf{H}_t$. The total time is therefore linear in N_z.

A.5. Fully decoupled EKF (forwards KL)

We now consider the forwards (inclusive) KL objective. Plugging in $\boldsymbol{\mu}_q = \boldsymbol{\mu}_p$ we get the simplified objective

$$J(q) = \frac{1}{2}\left(\log |\mathbf{\Sigma}_q| + \sum_j \left[\mathbf{\Sigma}_q^{-1} \mathbf{\Sigma}_p\right]_{jj}\right) + \mathrm{const} \tag{35}$$

Choosing $\mathbf{\Sigma}_q = \mathrm{diag}(\sigma_i^2)$:

$$J(q) = \frac{1}{2}\left(\sum_j \log \sigma_j^2 + \sum_j \sigma_j^{-2} \left(\mathbf{\Sigma}_p\right)_{jj}\right) + \mathrm{const} \tag{36}$$

Setting $\frac{\partial J(q)}{\partial \sigma_i} = 0$:

$$\frac{\partial J(q)}{\partial \sigma_i} = \frac{1}{\sigma_i} - \sigma_i^{-3} \left(\mathbf{\Sigma}_p\right)_{ii} = 0 \tag{37}$$

And therefore the solution is simply taking the diagonal elements of $\mathbf{\Sigma}_p$:

$$\sigma_i^2 = \left(\mathbf{\Sigma}_p\right)_{ii} \tag{38}$$

Plugging in the full-covariance EKF posterior covariance $\mathbf{\Sigma}_p = \mathbf{\Sigma}_{t|t}$:

$$\begin{aligned} \sigma_i^2 &= \left(\mathbf{\Sigma}_{t|t-1} - \mathbf{K}_t \mathbf{H}_t \mathbf{\Sigma}_{t|t-1}\right)_{ii} && (39) \\ &= \left(\mathbf{\Sigma}_{t|t-1}\right)_{ii} - \left(\mathbf{K}_t \mathbf{H}_t \mathbf{\Sigma}_{t|t-1}\right)_{ii} && (40) \end{aligned}$$

Let $\mathbf{A}_i$ and $\mathbf{A}^i$ represent the i^{th} row and column of matrix $\mathbf{A}$, respectively. Then, constraining the prior to be diagonal $\mathbf{\Sigma}_{t|t-1} = \mathrm{diag}(\sigma_i'^2)$:

$$\begin{aligned} \sigma_i^2 &= \sigma_i'^2 - \left(\mathbf{K}_t \mathbf{H}_t \mathrm{diag}(\sigma_j'^2)\right)_{ii} && (41) \\ &= \sigma_i'^2 - \left(\mathbf{K}_t \mathbf{H}_t\right)_{ii} \sigma_i'^2 && (42) \\ &= \sigma_i'^2 - \left(\mathbf{K}_t\right)_i \mathbf{H}_t^i \sigma_i'^2 && (43) \end{aligned}$$

Plugging in the full expression for $\mathbf{K}_t$:

$$\begin{aligned} \left(\mathbf{K}_t\right)_i &= \left(\mathbf{\Sigma}_{t|t-1} \mathbf{H}_t^\intercal \mathbf{S}^{-1}\right)_i = \left(\mathrm{diag}(\sigma_j'^2) \mathbf{H}_t^\intercal \mathbf{S}^{-1}\right)_i && (44) \\ &= \sigma_i'^2 \left(\mathbf{H}_t^\intercal \mathbf{S}^{-1}\right)_i && (45) \\ &= \sigma_i'^2 \left(\mathbf{H}_t^\intercal\right)_i \mathbf{S}^{-1} && (46) \\ &= \sigma_i'^2 \left(\mathbf{H}_t^i\right)^\intercal \mathbf{S}^{-1} && (47) \end{aligned}$$

In addition, we see that:

$$\mathbf{S} = \mathbf{R}_t + \mathbf{H}_t \mathrm{diag}(\sigma_j'^2) \mathbf{H}_t^\intercal = \mathbf{R}_t + \sum_j \mathbf{H}_t^j \sigma_j'^2 \left(\mathbf{H}_t^j\right)^\intercal \tag{48}$$

Plugging these in, therefore, we recover the covariance update equation for the fully-decoupled EKF, matching Eq. (12):

$$\sigma_i^2 = \sigma_i'^2 - \sigma_i'^2 \left(\mathbf{H}_t^i\right)^\intercal \left(\mathbf{R}_t + \sum_j \mathbf{H}_t^j \sigma_j'^2 \left(\mathbf{H}_t^j\right)^\intercal\right)^{-1} \mathbf{H}_t^i \sigma_i'^2 \tag{49}$$

A.6. Computing the posterior predictive distribution

After performing posterior inference for the parameters, we next compute the posterior predictive distribution. In the classification case, this becomes

$$p(y_t|\mathbf{y}_{1:t-1}, \mathbf{x}_{1:t}) = \int \mathrm{Cat}(y_t|\mathcal{S}(h(\mathbf{x}_t, \boldsymbol{\theta}_t)))\mathcal{N}(\boldsymbol{\theta}_t|\boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t})d\boldsymbol{\theta}_t \tag{50}$$

The simplest approach to this integral is to replace the Gaussian posterior with a delta function centered at the posterior mean/mode $\hat{\boldsymbol{\mu}}_t = \boldsymbol{\mu}_{t|t}$, which gives the plugin approximation

$$p(y_t|\mathbf{y}_{1:t-1}, \mathbf{x}_{1:t}) = \mathrm{Cat}(y_t|\mathcal{S}(h(\mathbf{x}_t, \hat{\boldsymbol{\mu}}_t))) \tag{51}$$

We call this the “plugin predictive”. Alternatively we can use a Monte Carlo approximation, in which we sample $\boldsymbol{\theta}_t$ from the Gaussian posterior and plug into h; we call this the “MC predictive”. However, Immer et al. (2021) argues that it is better to plug the samples into the linearized model, $\hat{h}(\mathbf{x}_t, \boldsymbol{\theta}_t) = \mathbf{H}_t\mathbf{x}_t$; we call this the “linearized predictive”.

A.7. More experimental results

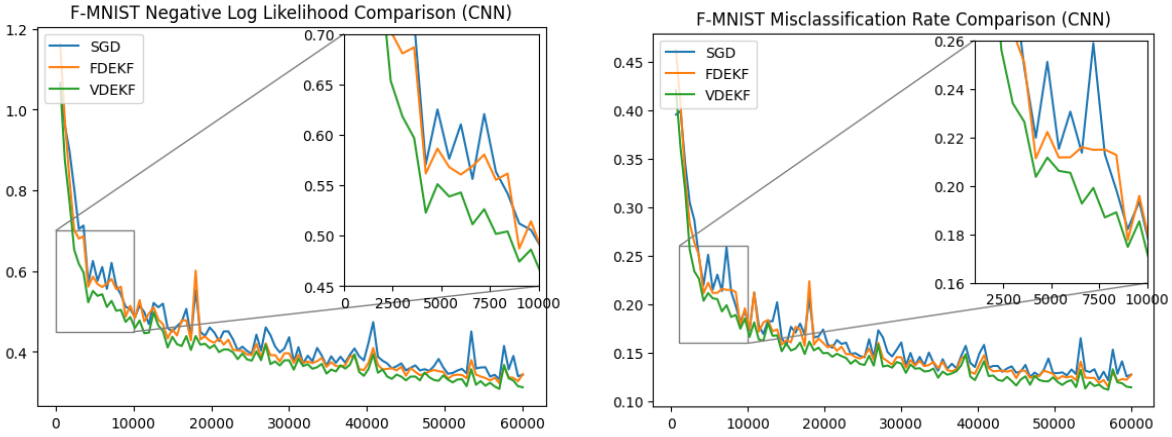


Figure 2: Comparison of negative log-likelihood values (left) and misclassification rates (right), using a CNN on Fashion-MNIST dataset. FD-EKF and VD-EKF values are computed using Monte Carlo predictive distribution with sample size 100.

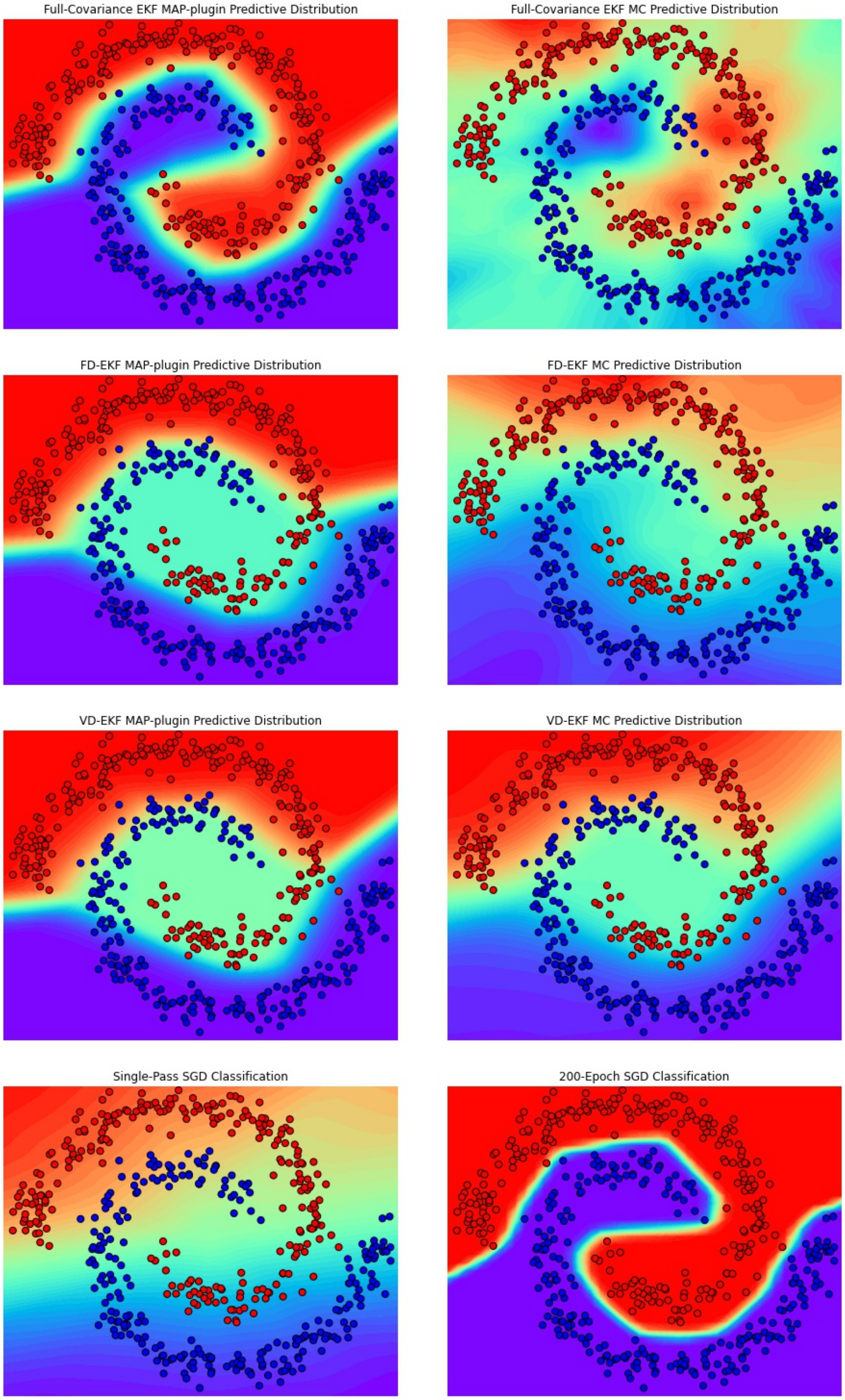
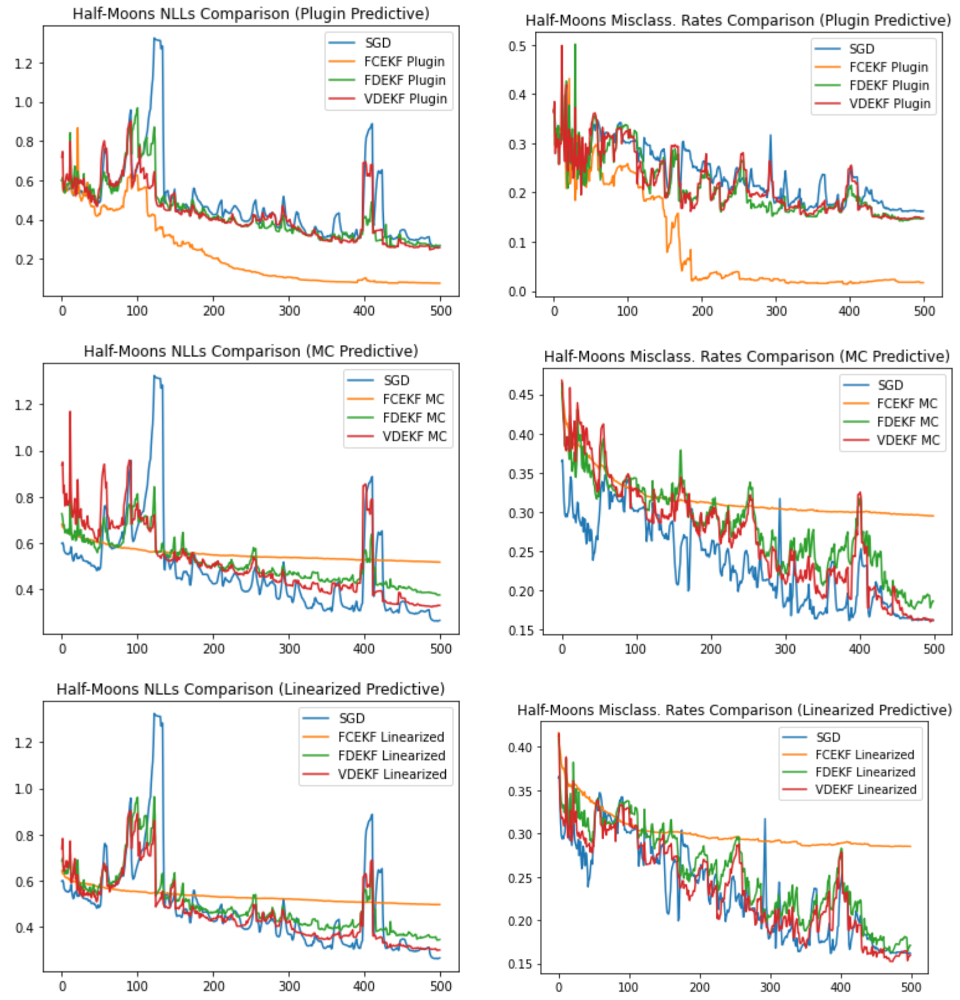


Figure 3: MAP-plugin predictive distribution (left) and Monte Carlo posterior predictive distribution (right) for FCEKF vs FDEKF vs VDEKF. Last row is single-pass and 200-epoch SGD classifications for reference.



Half-Moons NLLs Comparison (Plugin Predictive)

Half-Moons Misclass. Rates Comparison (Plugin Predictive)

Half-Moons NLLs Comparison (MC Predictive)

Half-Moons Misclass. Rates Comparison (MC Predictive)

Half-Moons NLLs Comparison (Linearized Predictive)

Half-Moons Misclass. Rates Comparison (Linearized Predictive)

Figure 4: Test-set negative log likelihoods (left) and misclassification rates (right) vs training step comparison for SGD vs FCEKF vs FDEKF vs VDEKF, where for the Bayesian methods, using plugin (top), Monte Carlo (middle), and linearized posterior predictive distributions, with sample size 100 for the Monte Carlo and linearized methods.