Bayesian Incremental Learning for Deep Neural Networks

Max Kochurov
maxim.v.kochurov@gmail.com
Dmitry Molchanov
dmolch111@gmail.com
Timur Garipov
timgaripov@gmail.com
Arsenii Ashukha
ars.ashuha@gmail.com
Dmitry Podoprikhin
timmyofmexico@gmail.com
Dmitry Vetrov
vetrodim@gmail.com

p(B|A)yesgroup.ru
SAMSUNG
Skoltech
Skolkovo Institute of Science and Technology

Key Results

Bayesian Incremental Learning allows to sequentially update model parameters without the use of old training data:

$$\text{Model}_{old},\ \text{Data}_{new} \ \rightarrow\ \text{Model}_{new}$$

Our contributions can be summarized as follows:

- We apply sequential Bayesian inference to the incremental learning setting
- We evaluate different posterior approximations
- We propose a way to use pretrained models

Bayesian Incremental Learning

- Dataset is divided into T parts $\mathcal{D}_1, \dots, \mathcal{D}_T$, which arrive sequentially during training
- The goal is to update model $p(w \mid \mathcal{D}_1, \dots, \mathcal{D}_{t-1})$ with $p(\mathcal{D}_t \mid w)$
- Bayesian approach can be applied

$$p(w \mid \mathcal{D}_1, \dots, \mathcal{D}_t) = \frac{p(\mathcal{D}_t \mid w) p(w \mid \mathcal{D}_1, \dots, \mathcal{D}_{t-1})}{\int p(\mathcal{D}_t \mid w) p(w \mid \mathcal{D}_1, \dots, \mathcal{D}_{t-1})\, dw}$$

In most cases the posterior distribution $p(w \mid \mathcal{D}_1, \dots, \mathcal{D}_t)$ is intractable

Scalable Bayesian Incremental Learning

- $p(w \mid \mathcal{D}_1, \dots, \mathcal{D}_t)$ – intractable, approximate it with $q(w \mid \phi_t)$
- Old approximation $q(w \mid \phi_{t-1})$ is reused as a prior
- $\mathcal{D}_t$ can be large, minibatch training can be applied

Using variational inference we get an optimization problem

$$\underbrace{\mathbb{E}_{q(w \mid \phi_t)} \log p(\mathcal{D}_t \mid w)}_{\text{Data term (likelihood)}} - \underbrace{D_{\mathrm{KL}}(q(w \mid \phi_t) \,||\, q(w \mid \phi_{t-1}))}_{\text{KL term (regularizer)}} \rightarrow \max_{\phi_t}$$

Pretraining

- First incremental step requires prior distribution to be specified
- Usually only pretrained weights are available (a point estimate)

One can use a Gaussian prior

$$p(w) = \mathcal{N}(w \mid w^\star, \sigma^2),$$

where $w^\star$ are pretrained weights and σ^2 is a hyper-parameter to be specified

How to set σ^2?

Grid search
\+ easy to implement
– low flexibility
– computationally expensive

Laplace approximation
\+ fits $\hat{\sigma}^2$ for every weight
– requires old data

Fully Factorized Gaussian Approximation (FFG)

Consider a dense layer with input and output dimensions I, O, respectively

$$q_\phi(w) = \prod_{i=1}^{I} \prod_{j=1}^{O} \mathcal{N}(w_{ij} \mid \mu_{ij}, \sigma_{ij}^2)$$

Fully Factorized Gaussian is a widely used approximation

Discussion

\+ fast, stable and easy to use approximation family
– low expressiveness

The approximate posterior for a convolutional layer factorizes similarly over all kernel parameters

Channel Factorized Gaussian Approximation (CFG)

Consider a convolutional layer with N filters and C channels with filter size $H \times W$. $L_{nc} \in \mathcal{R}^{HW \times HW}$ denotes a Cholesky factor for the covariance matrix

$$q_\phi(w) = \prod_{n=1}^{N} \prod_{c=1}^{C} \mathcal{N}(w_{nc} \mid \mu_{nc}, L_{nc} L_{nc}^\top)$$

Discussion

\+ preserves dependencies within kernel parameters channel-wise
\+ tractable reparametrization trick
– $\mathcal{O}(H^2 W^2)$ more parameters

Multiplicative Normalizing Flow Approximation (MNF)

Consider a convolutional layer, z_0 follows a simple fixed distribution $q(z_0)$ and NF is a normalizing flow

$$q_\phi(w \mid z) = \prod_{n=1}^{N} \prod_{c=1}^{C} \prod_{i=1}^{H} \prod_{j=1}^{W} \mathcal{N}(w_{ncij} \mid z_{nc} \mu_{ncij}, \sigma_{ncij}^2), \quad z = NF(z_0)$$

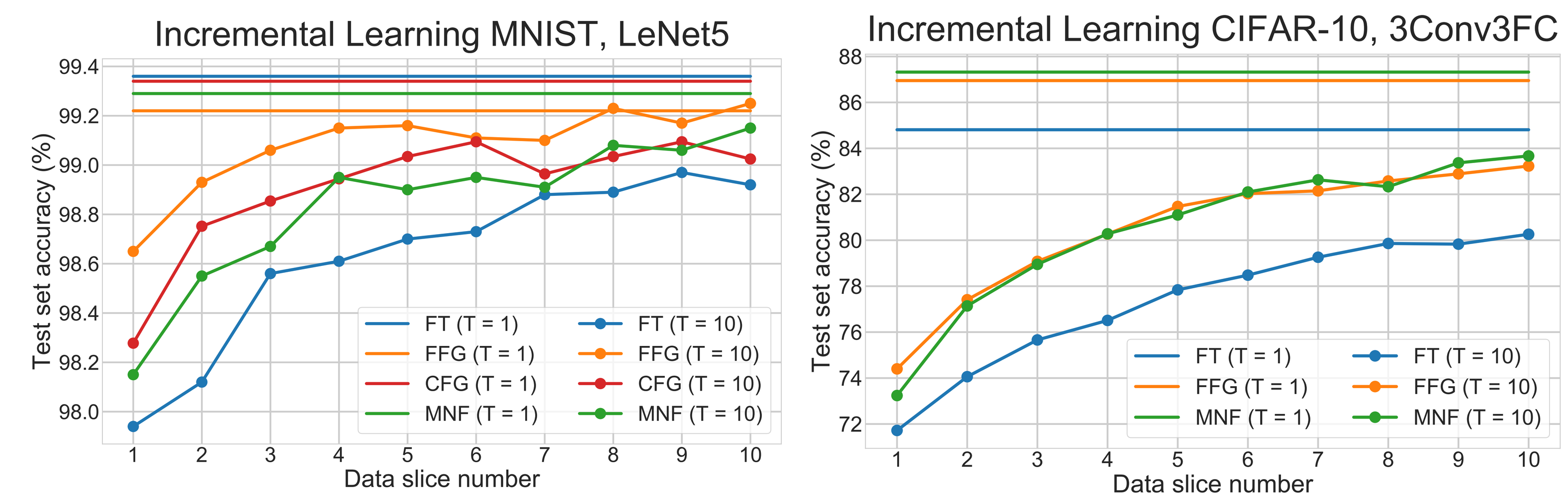
- Marginal approximate posterior that should be reused as a prior $q(w \mid \phi)$ is intractable [3]
- We derive a new variational lower bound and optimize the joint approximate posterior $q(w, z \mid \phi)$ instead

$$\mathcal{L} = \mathbb{E}_{q(w,z \mid \phi_t)} \log p(\mathcal{D}_t \mid w) - D_{\mathrm{KL}}(q(w, z \mid \phi_t) \,||\, q(w, z \mid \phi_{t-1}))$$

Discussion

\+ expressive family
\+ captures multi-modality
– many parameters
– slow training

Experiments: Incremental Learning on MNIST and CIFAR-10



(a) Results on MNIST (b) Results on CIFAR-10

Figure: Incremental learning experiments without pretraining

- Fine-tuning performs poorly on incremental learning task
- Fully Factorized approximation was sufficient on these datasets
- We had to downscale the KL Term for CIFAR-10 task to get good performance

Experiments: Incremental Learning with Pretraining

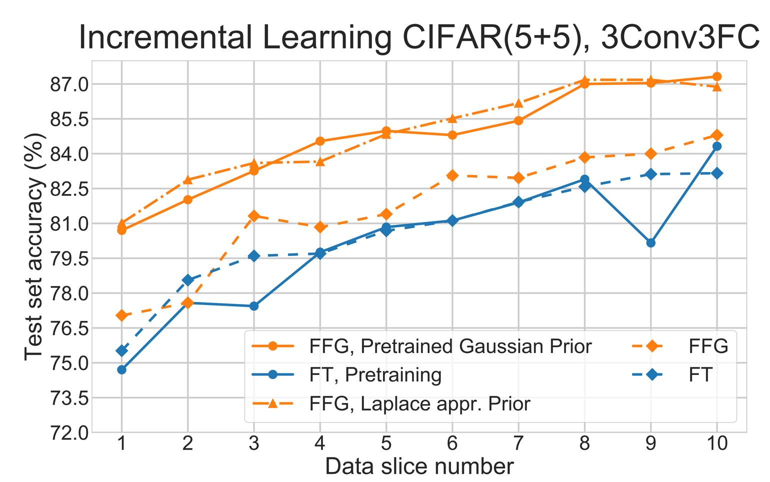


Figure: Incremental learning experiments with pretraining

Pretraining was done on randomly selected 5 classes. Incremental learning was performed on the rest ones using 3Conv3FC architecture

- Fine-tuning does not benefit from pretraining
- Pretraining helps Bayesian models
- Laplace approximation works well without grid search for σ^2

Discussion

- Bayesian framework provides intuitive tools to perform incremental learning procedure. Variational inference is required in most cases.
- It is possible to use pretrained models in Bayesian inference improving final quality. Laplace approximation is a reasonable way to choose a prior using old data and pretrained weights.
- Additional tricks (KL rescaling) are needed on larger problems.

Links and References

ArXiv: goo.gl/DpSpcq

[1] James Kirkpatrick et al., Overcoming catastrophic forgetting in neural networks, PNAS 2017

[2] Cuong V. Nguyen, Yingzhen Li, Thang D. Bui and Richard E. Turner, Variational Continual Learning, 2017

[3] Christos Louizos and Max Welling. Multiplicative normalizing flows for variational Bayesian neural networks, 2017