A. Network Architectures

We learn separate amortized inference networks to predict the mean $\mu$ and log-variance $\ln \sigma^2$ of the latent classification weight vectors $w_t$. Both networks have the same architecture, which depends on the feature extractor that is used. The inference networks are shared between the prior and approximate posterior distribution.

A.1. CONV-5 feature extractor

The embedding of the image returned by the CONV-5 feature extractor is a 256-dimensional vector. Each of the inference networks for mean and log variance of the classifier weights $w_t$ consists of three fully connected layers with 256 input and output features, and ELU non-linearity (Clevert et al., 2016) between the layers. There are two additional inference networks that predict mean and log variance of the classifier biases $b_t$. Both of them consist of two fully connected layers with 256 input and output features followed by ELU non-linearity, and a fully connected layer with 256 input and a single output feature. The design is the same as used by Gordon et al. (2019) to ensure comparability.

A.2. ResNet-12 feature extractor

With the ResNet-12 feature extractor, every image is embedded into a 512-dimensional feature vector. Each of the two inference networks consists of three fully connected layers with 512 input and output features, with skip connections and swish-1 non-linearity (Ramachandran et al., 2017) applied before addition.

B. Training details

We use 40k SGD updates with momentum 0.9, and early stopping based on meta-validation performance. We set the initial learning rate to 0.1, and decrease it by a factor ten after 20k, 25k and 30k updates. We clip gradients at 0.1, and set separate weight decay rates for the feature extractor, TEN, and inference networks: 0.0005, 0.0005 and 0.00001, respectively. The cosine classifier is scaled by $\alpha = 25$. We empirically find that regularization coefficient $\beta = \frac{K}{Nd}$ produces good results for 5-shot setup, and use a value of $\beta$ twice as large for the 1-shot setup. Here $d$ is the dimension of the feature vector $f_\theta(\cdot)$, $N$ is the number of classes in the task, and $K$ is the total number of query samples in the task.

Figure 1. Mean accuracy of the SAMOVAR-base classifiers sampled from prior and posterior as a function of $\beta$. While training, we fix the random seed of the data to generate the same series of miniImageNet tasks. The evaluation is performed over 5000 random tasks.

For the 5-shot setup, mini-batches consist of two episodes, each with 32 query images. For the 1-shot setup, we sample 5 episodes per mini-batch, and 12 query images per episode. In both cases query images are sampled uniformly across classes, without any restriction on the number per class. The auxiliary 64-way classification task is trained with batch size 64.

For ResNet-12, we report classification accuracy on miniImageNet with and without data augmentation. Data augmentation is performed with random horizontal flip, random crop, color jitter (brightness, contrast and saturation).

C. Impact of $\beta$-scaling

Typically, in autoencoders the dimensionality of the latent space is smaller than of the observed. This is not the case in the meta learning classification task where the output is merely a one-hot-encoded label of the class, while the latent space is of the same size as the output of the feature extractor. In our experiments we observe that a large KL term suppresses the reconstruction term resulting in a weaker performance. In particular, there is a trade off between these parts of the objective function $\hat{L}(\Theta)$ which can be regulated by $\beta$-scaling of the KL term. Figure 1 shows the dependence of the accuracy of SAMOVAR-base with CONV-5 feature extractor as a function of $\beta$. Even though in both setups there is a clear maximum, overall, the model is relatively robust to the setting of $\beta$. Let’s denote the optimum $\beta$ as
$\beta_{\text{opt}}$. Then for 5-shot setup the range at least from $0.83\beta_{\text{opt}}$ to $2\beta_{\text{opt}}$ produces results that are within 1% interval from the maximum accuracy at $\beta_{\text{opt}}$. For 1-shot setup, the same holds true for the range at least from $0.66\beta_{\text{opt}}$ to $2\beta_{\text{opt}}$.

D. Posterior collapse in VERSA

While training VERSA, every 250 optimization steps we keep track of the largest variance in weights and biases of the predicted classifier. Figure 2 shows how this variance decreases with time. The largest variance first falls below 0.001 at the step 4000 in the 5-shot setup, and at the step 3000 in the 1-shot setup.

References

