Dropout-based Automatic Relevance Determination

Dmitry Molchanov
dmitry.molchanov@skolkovotech.ru

Arseniy Ashuha
ars.ashuha@phystech.edu

Dmitry Vetrov
vetrovd@yandex.ru

p(B|A)yesgroup.ru

Main result

We propose a new way to train adaptive individual dropout rates for each model weight. Our approach provides Automatic Relevance Determination and enforces model sparsity.

Motivation

- With Bayesian Machine Learning techniques we can construct sparse models like the Relevance Vector Machine
- Traditional Automatic Relevance Determination approach is hard to apply to deep neural networks
- Variational Dropout is an intriguing technique that allows to train individual dropout rates for each model weight, so we build upon this approach

Variational Dropout model definition and training

- Approximate posterior distribution and log-uniform prior

$$q(w_i \,|\, \theta_i, \alpha_i) = \mathcal{N}(\theta_i, \alpha_i \theta_i^2) \qquad p(\log|w_i|) \propto c$$

- Training: optimize the Variational Lower Bound w.r.t. θ, α

$$\mathcal{L}(q) = \mathbb{E}_{q(w)} \log p(T \,|\, X, w) - D_{KL}(q(w) \,\|\, p_{prior}(w)) \to \max_{\theta,\alpha}$$

- When α is fixed, VDO is equivalent to Gaussian Dropout:

$$w_i \sim q(w_i) \Leftrightarrow w_i = \theta_i \varepsilon_i, \; \varepsilon_i \sim \mathcal{N}(1, \alpha_i)$$

- Training procedure (Doubly Stochastic Variational Inference)

Require: X: train objects; T: train labels; M: mini-batch size
Ensure: optimal θ, α

1: $\theta, \alpha \leftarrow$: initial approximation;
2: **repeat**
3: $X, T^M \leftarrow$ Random mini-batch of M datapoints
4: ▷ Sample gradient estimate using the Local Reparametrization Trick
5: $g \leftarrow \nabla_{\theta,\alpha} \hat{\mathcal{L}}^M(\theta, \alpha; (X, T)^M)$
6: $(\theta, \alpha) \leftarrow$ Update parameters using gradient-based method like Adam
7: **until** convergence of parameters (θ, α)

- Only $\alpha \leq 1$ case is considered (authors report problems with large-variance gradients in case of large values of α)

Variational Dropout Automatic Relevance Determination

Main idea: driving dropout rates α_j to infinity pushes θ_j to 0 and enables Automatic Relevance Determination:

$$q(w_i \,|\, \theta_i, \alpha) = \mathcal{N}(w_i \,|\, \theta_i, \alpha_i \theta_i^2) = \delta(0)$$

To achieve this goal we offer

- Analytical analysis in case of linear regression
- A way to reduce gradient variance
- New approximation of KL divergence which is valid for all values of α
- Experiments that show that ARD effect persists in doubly stochastic setting

Linear Regression

Likelihood:

$$p(t \,|\, \mathrm{X}, w) = \mathcal{N}(t \,|\, \mathrm{X}w, \beta^{-1}\mathrm{I})$$

Objective:

$$\mathcal{L}(q) = \frac{N}{2} \log \beta - \frac{\beta}{2} \left[\|\mathrm{X}\theta - t\|^2 + \sum\nolimits_d \varkappa_d \theta_d^2 \right] - D_{KL}(\alpha)$$

Exact update for θ given fixed α:

$$\theta^* = (\mathrm{X}^\top \mathrm{X} - \mathrm{diag}(\varkappa))^{-1} \mathrm{X}^\top t, \qquad \varkappa_d = \alpha_d \sum_{n=1}^{N} x_{nd}^2$$

- Looks much like corresponding expression from the RVM regression:

$$w^* = (\mathrm{X}^\top \mathrm{X} - \mathrm{diag}(\alpha))^{-1} \mathrm{X}^\top t$$

- $\alpha_d \to +\infty \Rightarrow \mathcal{N}(\theta_d, \alpha_d \theta_d^2) \to \mathcal{N}(0, 0) = \delta(0)$

New parametrization

- Original multiplicative noise yields noisy gradients:

$$w_i = \theta_i (1 + \sqrt{\alpha_i} \cdot \varepsilon_i), \qquad \frac{\partial w_i}{\partial \theta_i} = 1 + \sqrt{\alpha_i} \cdot \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, 1)$$

- Proposed equivalent additive noise reduces variance of gradients:

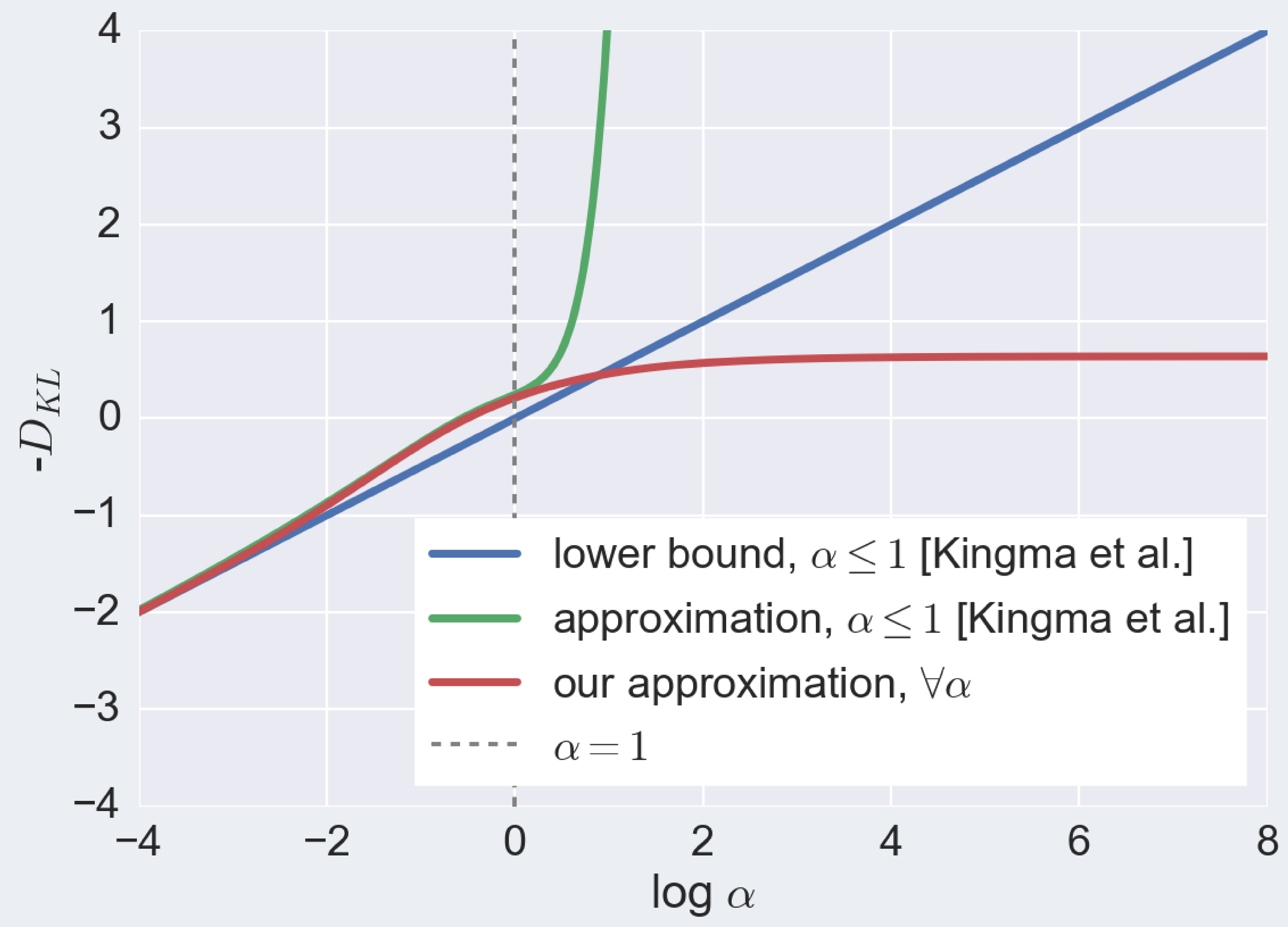
$$w_i = \theta_i + \sigma_i \cdot \varepsilon_i, \qquad \frac{\partial w_i}{\partial \theta_i} = 1, \qquad \varepsilon_i \sim \mathcal{N}(0, 1)$$

- We now use $q(w_i) = \mathcal{N}(w_i \,|\, \theta_i, \sigma_i^2)$ and can obtain α_i from $\alpha_i \theta_i^2 = \sigma_i^2$

New Approximation for KL Divergence

- Original VDO paper offered several KL divergence approximation for $\alpha \leq 1$
- To drive α's to infinity and achieve ARD effect we generalize approximation for all values of α by sampling the divergence and solving regression task

$$-D_{KL}(q\|p) \approx 0.64\,\sigma(1.5\,(1.3 + \log \alpha_i)) - 0.5\,\log(1 + \alpha_i^{-1}) + c$$



- We also showed that $\alpha \to +\infty \Rightarrow D_{KL}(\alpha) \to \mathrm{const}$

Automatic Relevance Determination in RVM vs ARD in VDO

The cause and nature of ARD in VDO is completely different from ARD in RVM

ARD in RVM

1. $\alpha_i \to +\infty$
2. $p(w_i \,|\, \alpha_i, Data) = \delta(0)$
3. α_i are prior parameters
4. Fit prior distribution parameters to data
5. ARD by shrinking the prior to a delta funciton

ARD in VDO

1. $\alpha_i \to +\infty$
2. $q(w_i \,|\, \theta_i, \alpha_i) = \delta(0)$
3. α_i are variational parameters
4. Prior distribution is fixed
5. ARD by introducing infinitely strong multiplicative noise

Our approach comes more naturally from the Bayesian framework, as the prior distribution should remain independent from the training data.

Experiments

Model: VD-ARD multiclass logistic regression, trained with DSVI

	Accuracy			Sparsity		
Dataset	VD-ARD	L1-LR	RVM	VD-ARD	L1-LR	RVM
MNIST	**0.926**	0.919	N/A	**69.8%**	57.8%	N/A
DIGITS	0.948	**0.955**	0.945	**75.4%**	38.0%	74.6%
DIGITS + noise	0.930	**0.937**	0.846	**87.6%**	55.9%	85.7%

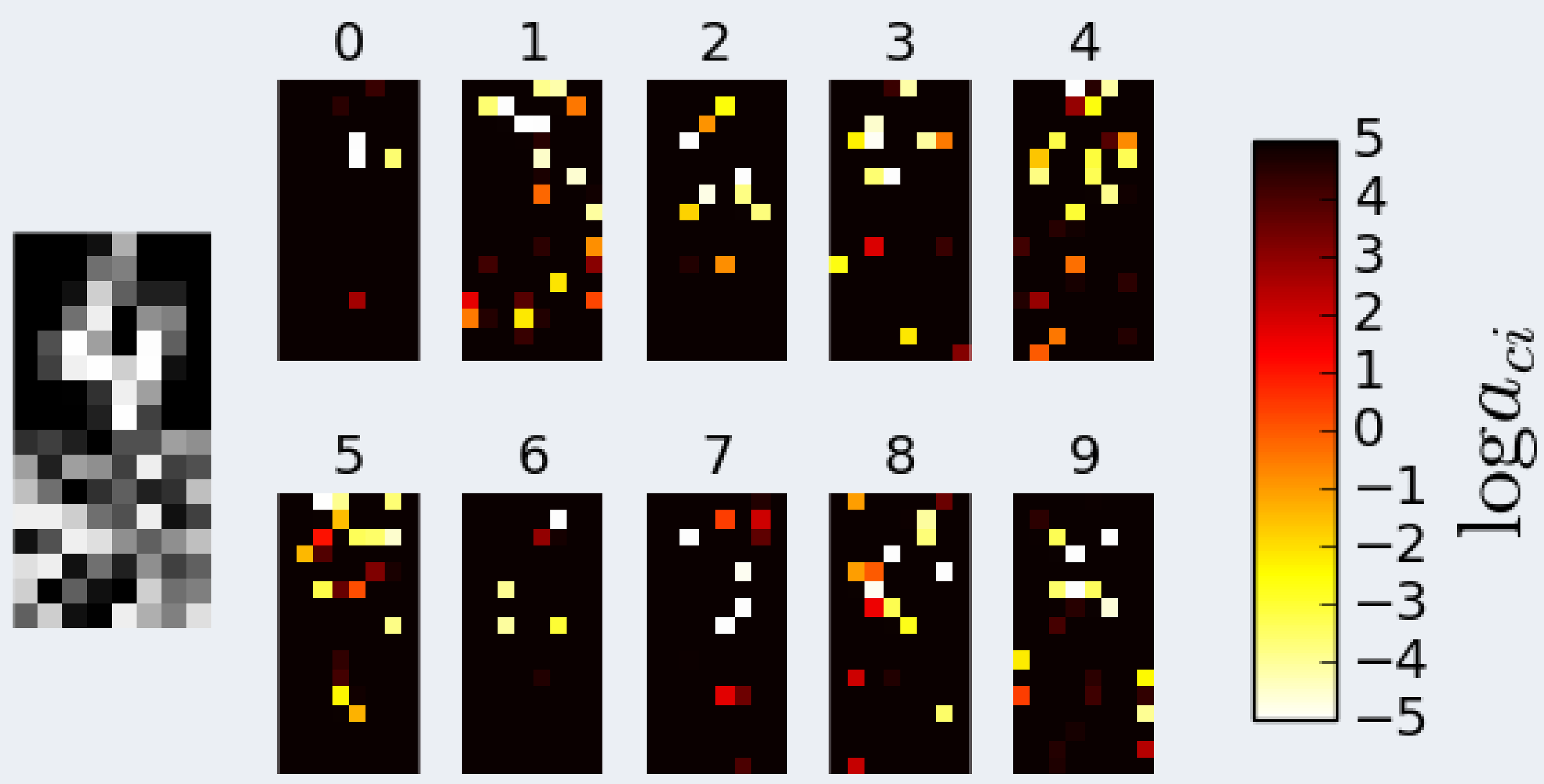


Figure: An example of an object with concatenated noise. Trained dropout rates.

Discussion

- Variance of the gradients in VDO can be decreased by replacing multiplicative noise with equivalent additive noise
- Variational Dropout can be used to obtain ARD effect
- This effect still holds in doubly stochastic scenario (can be applied to DNNs!)
- Can't share α's and therefore can't enforce group sparsity