Variance Networks

When Expectation Does Not Meet Your Expectations

Dmitry Molchanov

May 11, 2018

https://arxiv.org/abs/1803.03764

p(B|A)yesgroup.ru

Stochastic Neural Networks

- Dropout
- Batch Normalization
- Bayesian Neural Networks

- Train time:
  - Inject noise
- Test time:
  - Mean propagation
  - Ensembling
  - Distillation / fast dropout / …

Is noise informative?

- Dropout: 1 dropout rate per layer, no information in the noise
- FFG posterior
  - Small variance
  - Learns weight “uncertainty”; we can permute variances with almost no accuracy drop
- More complex posteriors?
  - Multiplicative normalizing flows? Maybe…

- Can we explicitly learn informative noise for better ensembling?

Outline

- Variance networks
- Variational dropout ⇔ variance network
- Mean propagation
- Open questions

Variance networks

- Consider a FFG distribution over the weights:

$$w_{ij} \sim N\left(\mu_{ij}, \sigma_{ij}^2\right)$$

$$w_{ij} = \mu_{ij} + \epsilon_{ij}\sigma_{ij}$$

- How to learn informative σ? Eliminate μ completely!

$$w_{ij} \sim N\left(0, \sigma_{ij}^2\right)$$

$$w_{ij} = \epsilon_{ij}\sigma_{ij}$$

- Works almost the same as usual models!*

* Conditions apply

- Okay, this is strange:
  - Weights have random signs
  - Mean activation is 0 for every object
  - Why would this even work?!

Variance networks demystified

- Let's look at the distribution of the activations:

$$y = Wx$$

$$w_{ij} \sim N\left(0, \sigma_{ij}^2\right)$$

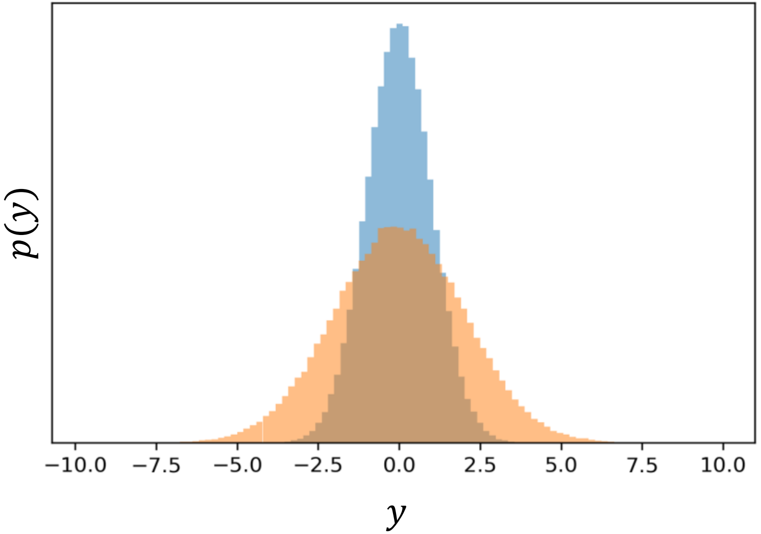
$$y_i \sim N\left(0, \left(\sigma_i^2\right)^\top \left(x^2\right)\right)$$

$$y_i = \epsilon_i \sqrt{\left(\sigma_i^2\right)^\top \left(x^2\right)}$$

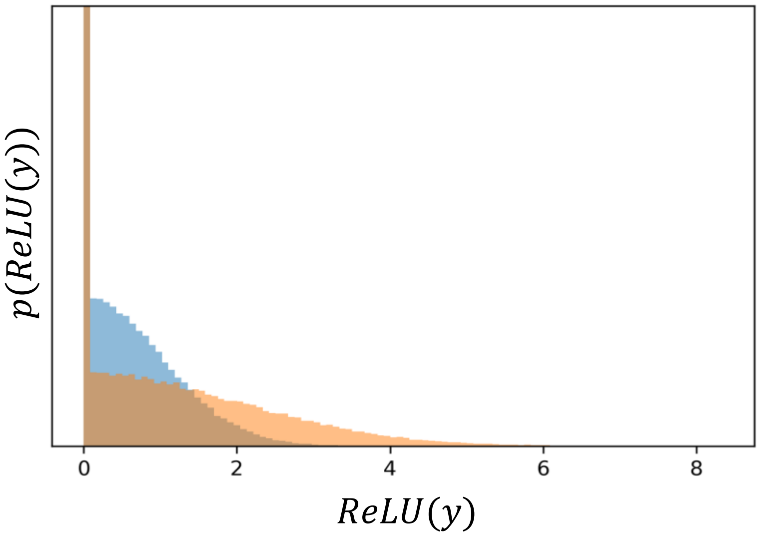
(a.k.a. the local reparameterization trick)

- Nonlinearities break this symmetry!

Variance networks demystified



ReLU

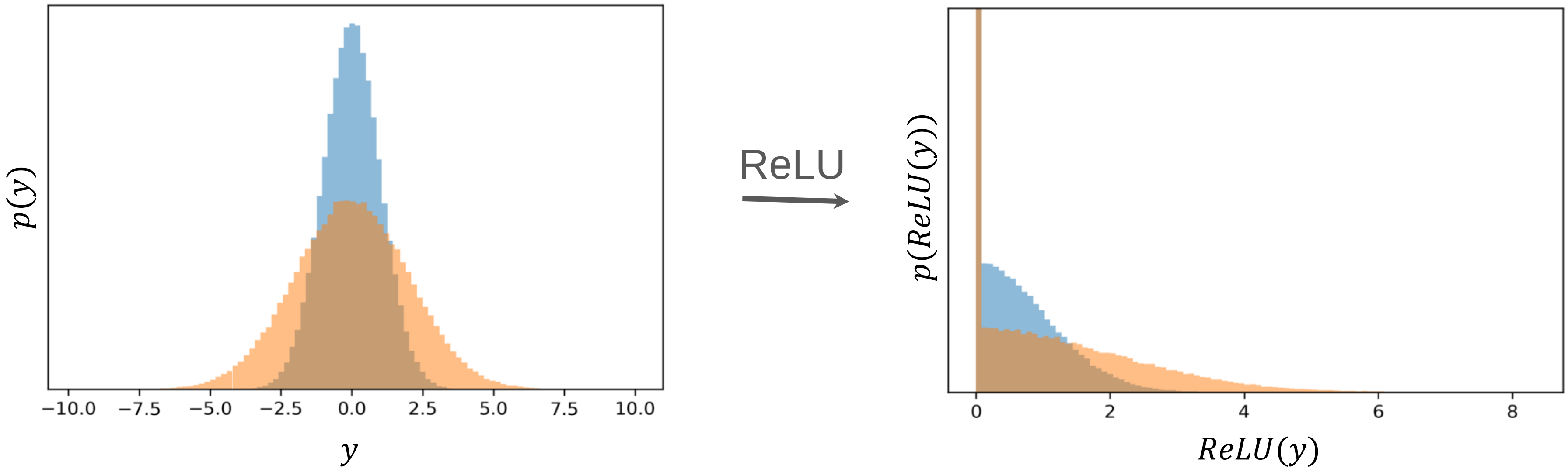


$$y_i = \epsilon_i \sqrt{\left(\sigma_i^2\right)^{\top} (x^2)}$$

$$ReLU(y_i) \sim \frac{1}{2} HalfNormal\left(y_i \mid 0, \left(\sigma_i^2\right)^{\top} \left(x^2\right)\right) + \frac{1}{2}\delta_0(y_i)$$

- ReLU = Abs + Binary Dropout

Variance networks demystified



- Nonlinearity breaks the symmetry
- The information is stored in the **magnitude** of the activations
- Biases add some more expressivity

Would other symmetric distributions work?

- Symmetric binary dropout:

$$w_{ij} = \sigma_{ij}\epsilon_{ij}$$

$$P(\epsilon_{ij} = -1) = P(\epsilon_{ij} = 1) = \frac{1}{2}$$

- Symmetric uniform distribution:

$$w_{ij} = \sigma_{ij}\epsilon_{ij}$$

$$\epsilon_{ij} \sim U(-1, 1)$$

- All work approximately the same (but the LRT would be tricky…)

Outline

- Variance networks
- Variational dropout ⇔ variance network
- Mean propagation
- Open questions

Variational dropout is a variance network

Variational Dropout:

- FFG posterior

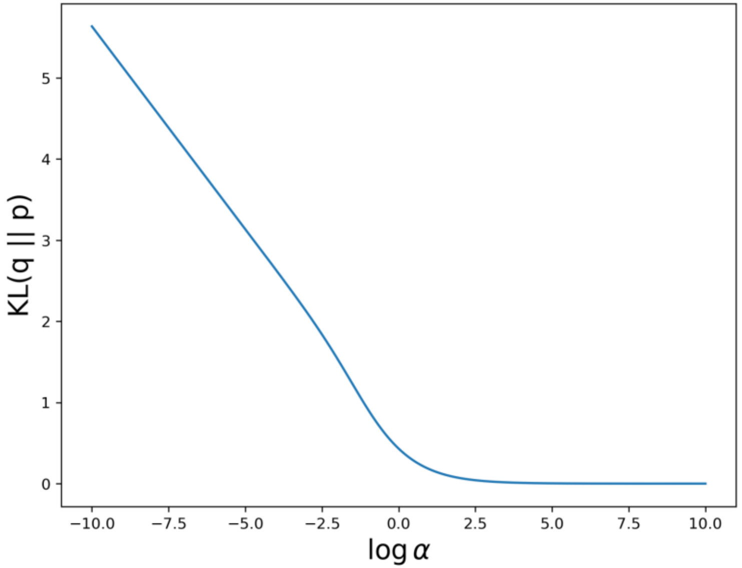
$$w_{ij} \sim q(w_{ij}) = N(\mu_{ij}, \alpha\mu_{ij}^2)$$

- Log-uniform prior distribution

$$p(w_{ij}) \propto \frac{1}{|w_{ij}|}$$

- ELBO favors large dropout rates α

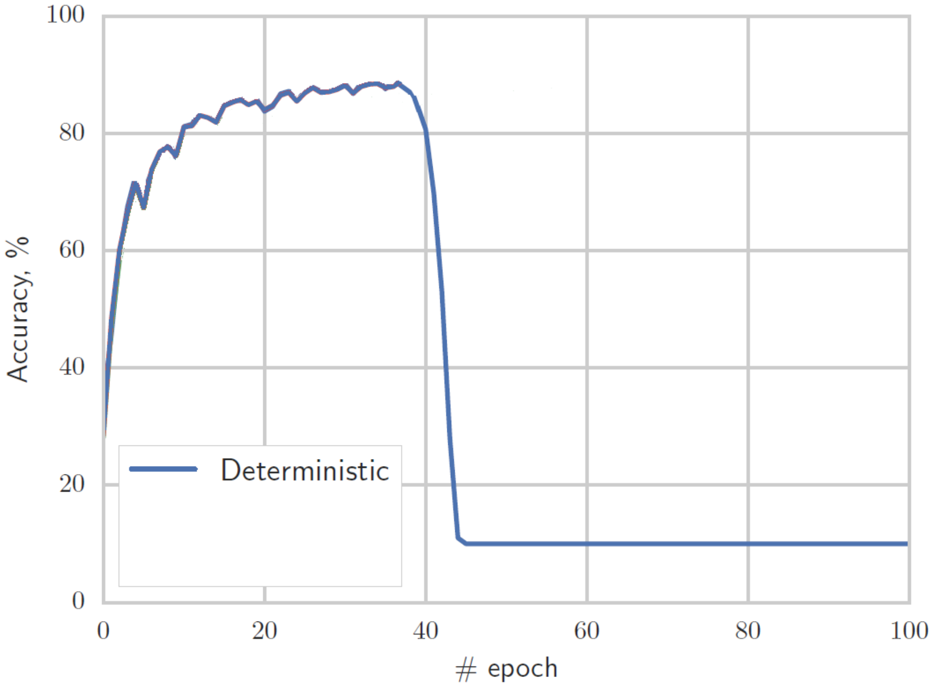
$$L = E_{q(w)} p(y \mid x, w) - KL(q(w) \,||\, p(w)) \to \max_{\mu, \alpha}$$



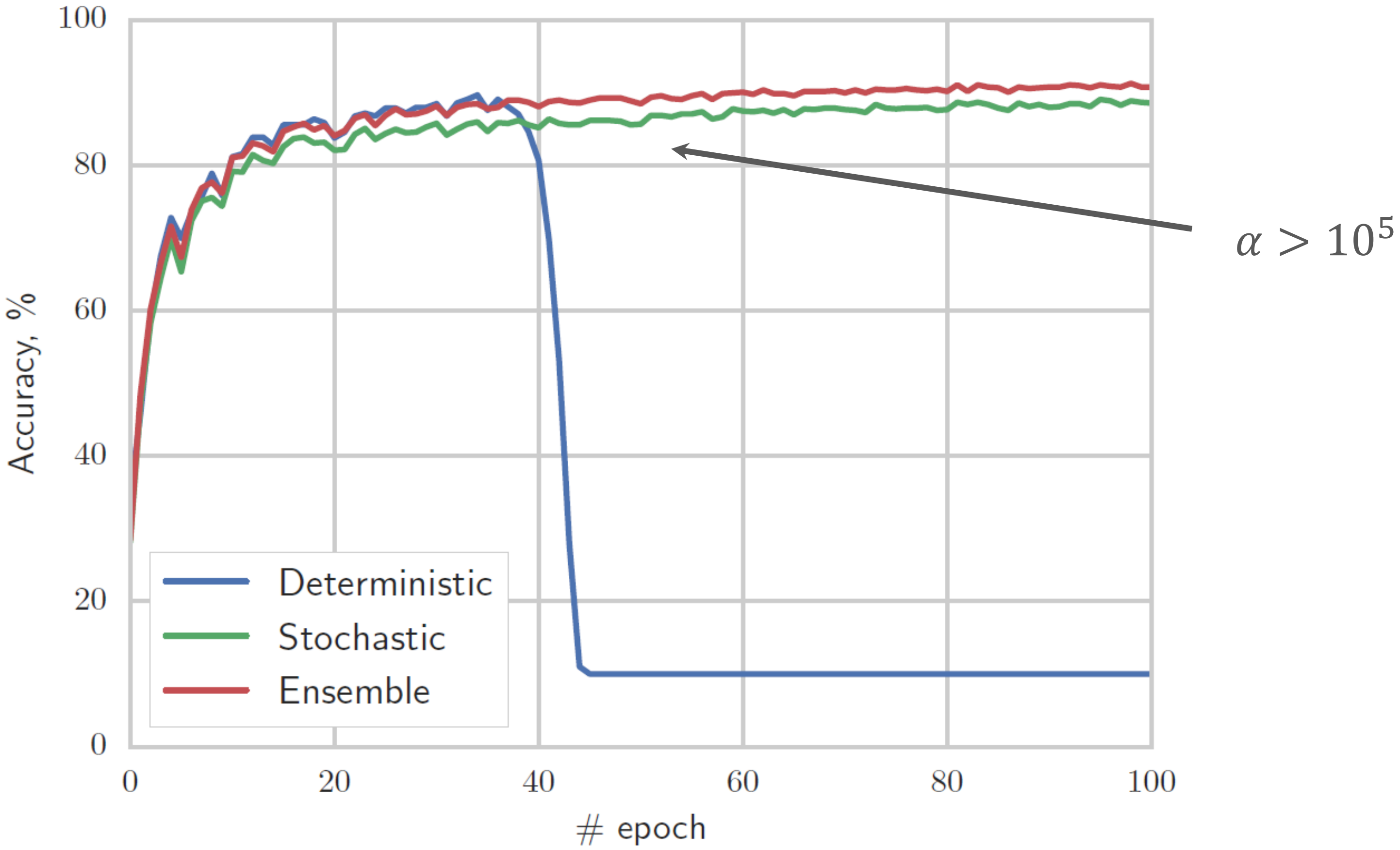
Variational dropout is a variance network

$$w_{ij} \sim q(w_{ij}) = N(\mu_{ij}, \alpha\mu_{ij}^2)$$

- Kingma et. al. 2015 clipped alpha. Why?



Variational dropout is a variance network



Variational dropout is a variance network

- Moreover, we can substitute

$$q(w_{ij}) = N(\mu_{ij}, \alpha\mu_{ij}^2) \approx N(0, \alpha\mu_{ij}^2)$$

$$KL\left(N(\mu_{ij}, \alpha\mu_{ij}^2)\ ||\ N(0, \alpha\mu_{ij}^2)\right) = \frac{\alpha\mu_{ij}^2 + \mu_{ij}^2}{2\alpha\mu_{ij}^2} - \frac{1}{2} = \frac{1}{2\alpha} \to 0$$

- The predictions remain the same
- It is a pure variance network now!
- Only works with layer-wise and neuron-wise parameterization

- Start training as a usual low-variance network
- Smoothly transition into a variance-only network
- Faster, more stable training than pure variance-only

Variance network is variational dropout

$$q\left(w_{ij}\right) = N\left(0, \sigma_{ij}^2\right)$$

$$KL\left(N\left(0, \sigma_{ij}^2\right) \,||\, LogU\right) = const$$

- ELBO is now fairly simple:

$$L = E_{q(w)} p(y \mid x, w) + const$$

- Ironic, one of the strongest priors results in no regularization…

Variational dropout is a varaince network

- Variance network is actually the best possible variational dropout network!
- Sparse Variational Dropout is just a poor local optimum ☹

	Layer	Neuron	Weight	Additive
ELBO	$\mathbf{-5.9 \cdot 10^2}$	$-7.7 \cdot 10^2$	$-6.4 \cdot 10^4$	$-2.3 \cdot 10^4$
Det. accuracy	11.3	11.3	81.3	96.3
Ens. accuracy	99.2	99.2	99.2	99.2

$$q(w_{ij}) = \mathcal{N}(w_{ij} \,|\, \mu_{ij}, \alpha\mu_{ij}^2) \qquad \text{layer-wise}$$

$$q(w_{ij}) = \mathcal{N}(w_{ij} \,|\, \mu_{ij}, \alpha_j\mu_{ij}^2) \qquad \text{neuron-wise}$$

$$q(w_{ij}) = \mathcal{N}(w_{ij} \,|\, \mu_{ij}, \alpha_{ij}\mu_{ij}^2) \qquad \text{weight-wise}$$

$$q(w_{ij}) = \mathcal{N}(w_{ij} \,|\, \mu_{ij}, \sigma_{ij}^2) \qquad \text{additive}$$

Variational dropout is a varaince network

- Variance network is actually the best possible variational dropout network!
- Sparse Variational Dropout is just a poor local optimum ☹

- Why it happens?

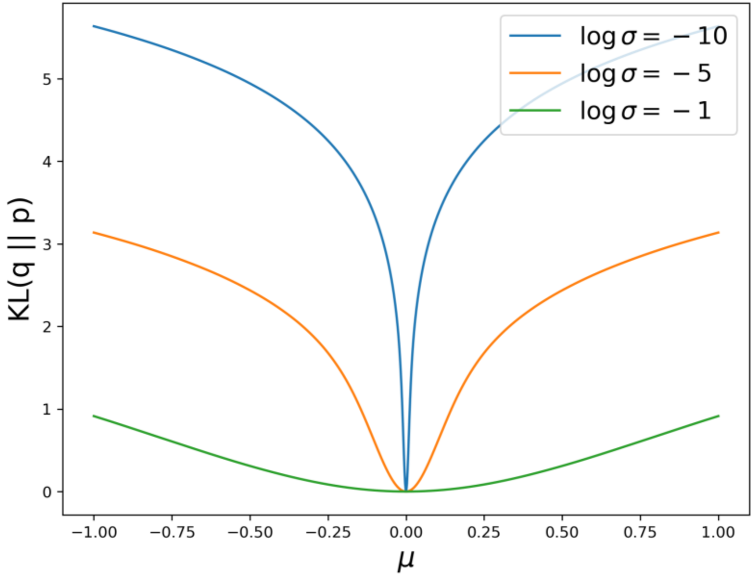
$$L = E_{q(w)} p(y \mid x, w) - KL\big(q(w) \,||\, p(w)\big)$$

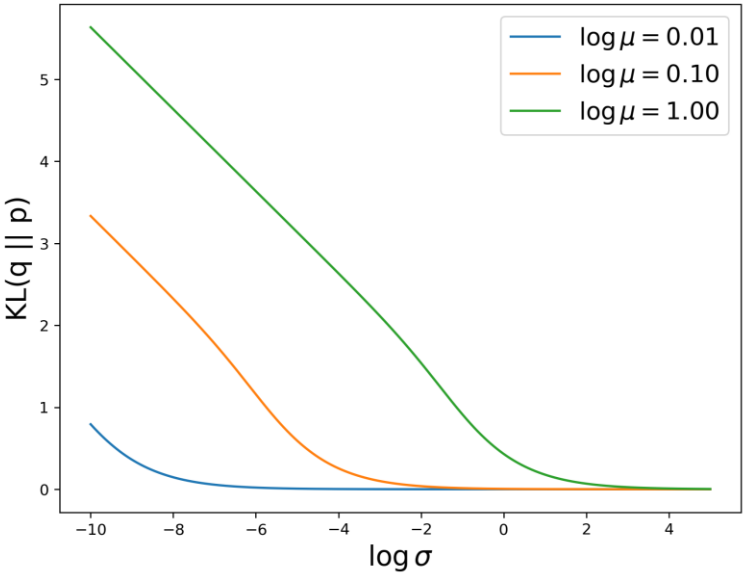
- Variance network “overfits”: 1.0 training accuracy, low cross entropy
- KL divergence is exactly zero!

- Sparse Variational Dropout also has 1.0 training accuracy and low loss…
- … but KL-term is huge

Sparsity in sparse variational dropout

- Then why is sparse variational dropout sparse?
- We aided the optimization process to get stuck in a sparse solution
- Variances are initialized with very small values
- In order to increase α it is easer to push μ to 0 than to increase $\log \sigma$





Do other models lead to variance networks?

Maybe the improper prior of Variational Dropout is at fault?

- Student's t-distribution:

$$KL\left(N(\mu,\alpha\mu^2)\,||\,Students(\nu)\right) \simeq$$

$$\simeq const - \frac{1}{2}\log\alpha - \frac{1}{2}\log\mu^2 + \frac{\nu+1}{2}E_\epsilon \log\left(\nu + \mu^2\left(1+\sqrt{\alpha}\epsilon\right)^2\right) \to$$

$$\to const - \frac{1}{2}\log\alpha + E_\epsilon \log\left(1+\sqrt{\alpha}\epsilon\right) \simeq KL\left(N(\mu,\alpha\mu^2)\,||\,LogU\right)$$

- LogU is a limit case of Student's t for $\nu = 0$
- Student's t with $\nu \approx 0$ behaves exactly like Variational Dropout

Do other models lead to variance networks?

Maybe the improper prior of Variational Dropout is at fault?

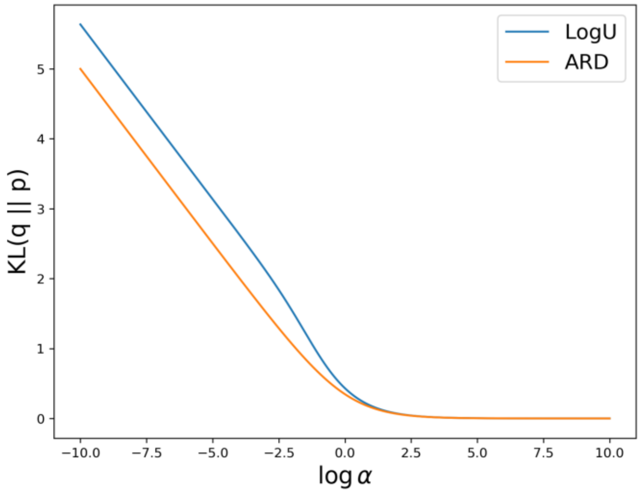
- ARD prior: $p(w_{ij}) = N(0, \sigma_{ij}^2), q(w_{ij}) = N(\mu_{ij}, \alpha\mu_{ij}^2)$

$KL\left(N(\mu, \alpha\mu^2) \mid\mid LogU\right) \simeq$

$\simeq const - \frac{1}{2}\log\alpha + E_\epsilon \log\left(1 + \sqrt{\alpha}\epsilon\right)$

$KL\left(N(\mu, \alpha\mu^2) \mid\mid N(0, \sigma_*^2)\right) = \frac{1}{2}\log(1 + \alpha^{-1})$

Almost the same KL divergence!



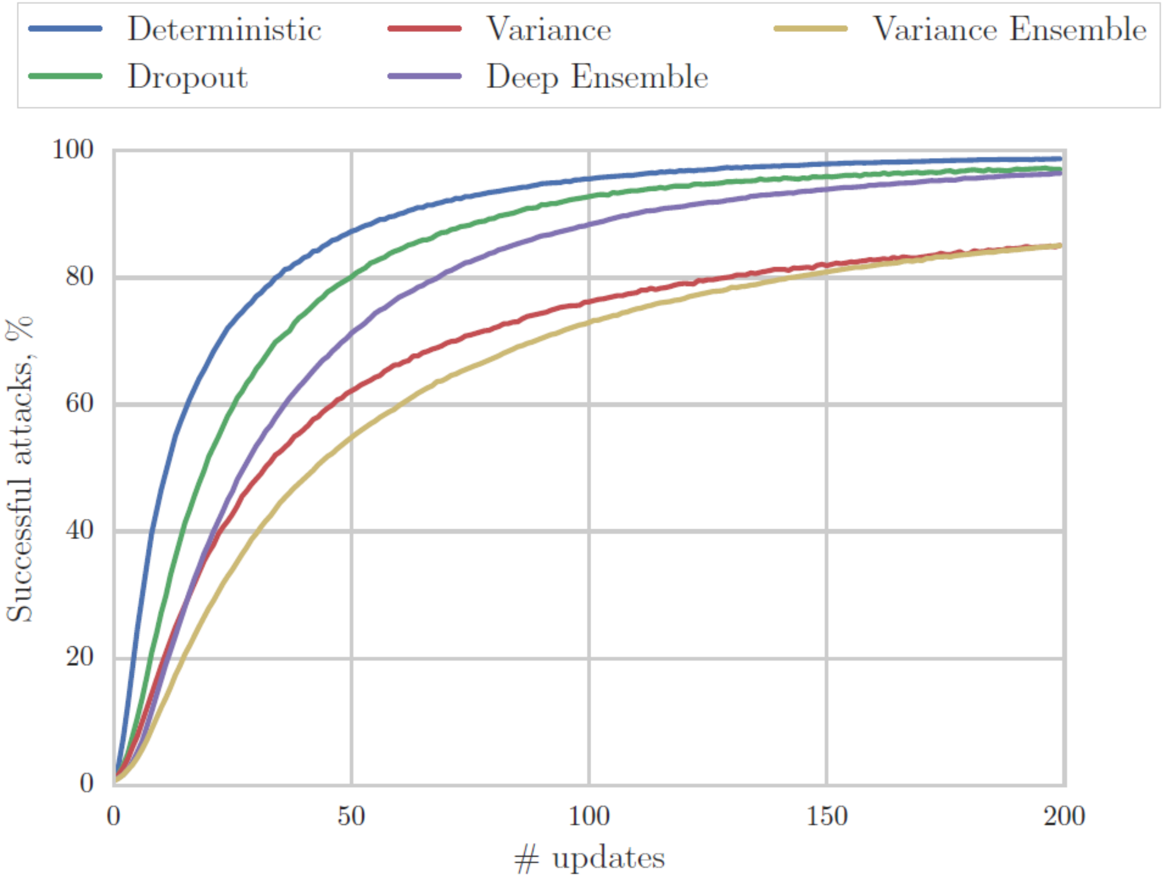
- ARD prior behaves exactly like Variational Dropout!

- Maybe Variational Dropout wasn't the right way to extend Gaussian Dropout?

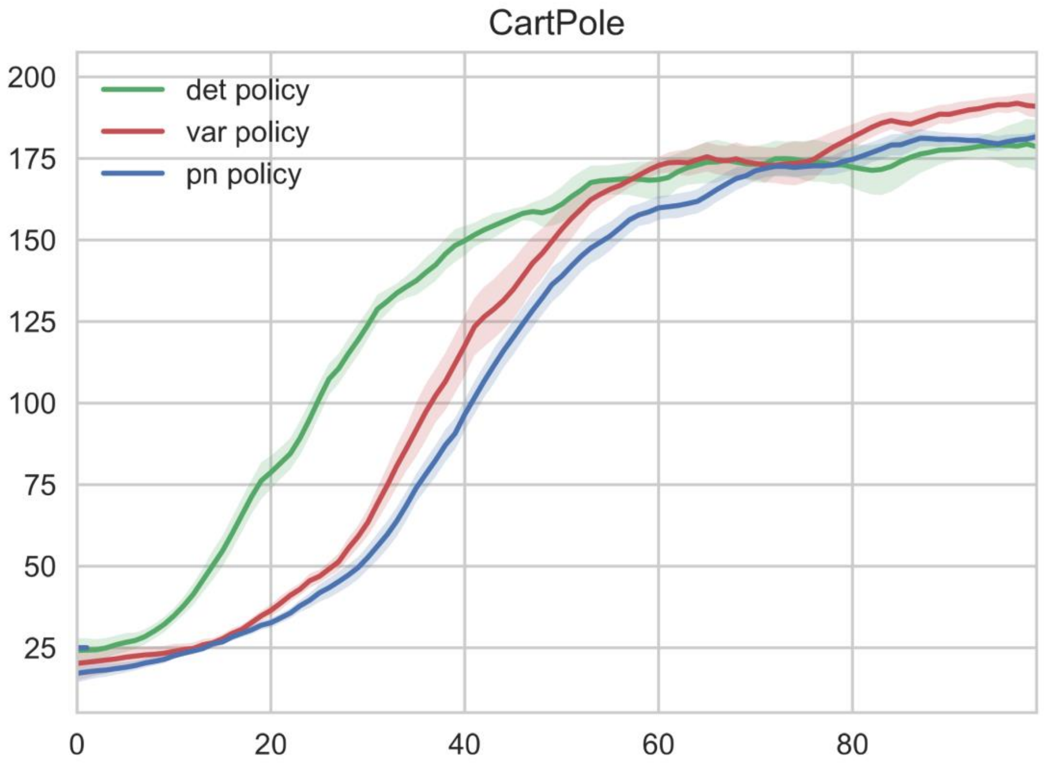
Experiments: classification

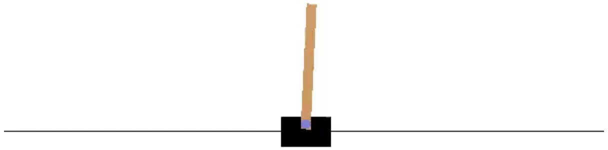
Architecture	Dataset	Network	Accuracy (%)		
			Stoch.	Det.	Ens.
LeNet5	MNIST	Dropout	99.1	99.4	99.4
		Variance	95.9	10.1	99.3
VGG-like	CIFAR10	Dropout	91.0	93.1	93.4
		Variance	91.3	10.0	93.4
VGG-like	CIFAR100	Dropout	77.5	79.8	81.7
		Variance	76.9	5.0	82.2

Experiments: adversarial attacks

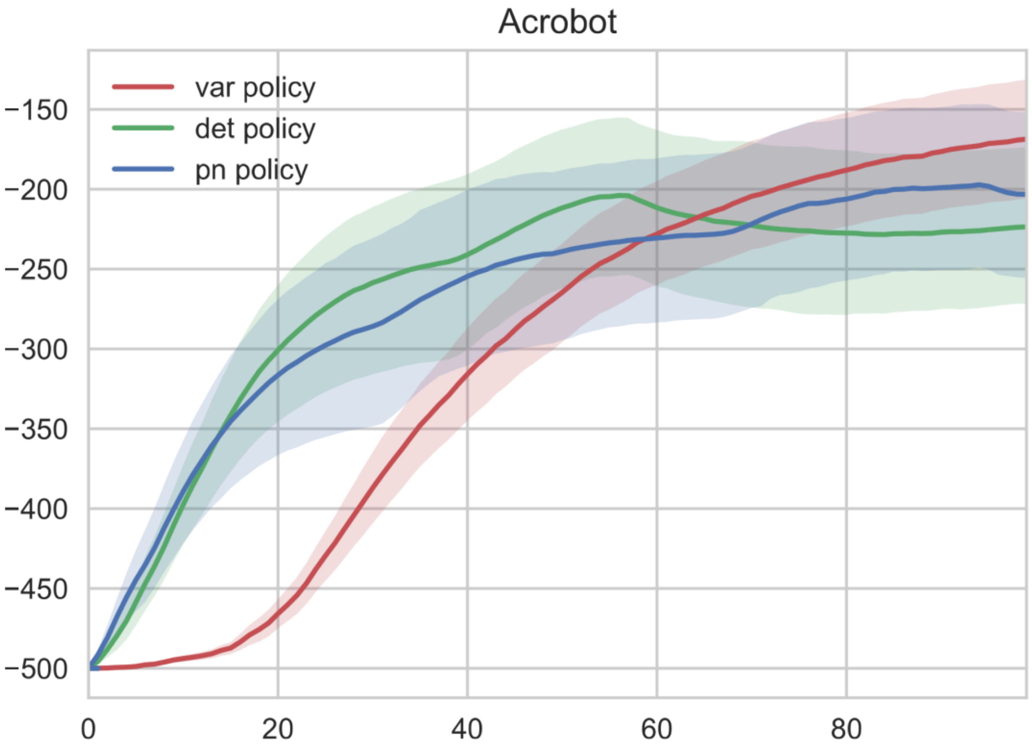


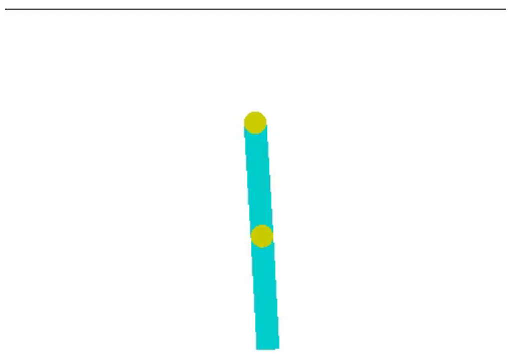
Experiments: reinforcement learning





Experiments: reinforcement learning





Outline

- Variance networks
- Variational dropout ⇔ variance network
- Mean propagation
- Open questions

Mean propagation

- In all conventional stochastic networks we could perform **mean propagation**
  - a.k.a. weight scaling rule
  - a.k.a. deterministic procedure
- It fails miserably on variance networks
- Why it fails and how to test whether it fails?

- DNN is a highly non-linear function of its weights
- Which weights can be substituted with their expectations?

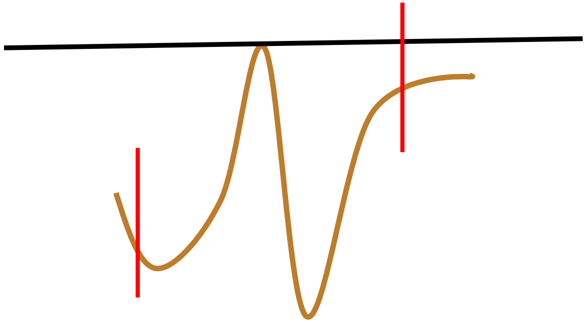
Mean propagation

- DNN is a highly non-linear function of its weights
- Which weights can be substituted with their expectations?

$$q(w_{ij}) = N(\mu_{ij}, \sigma_{ij}^2)$$

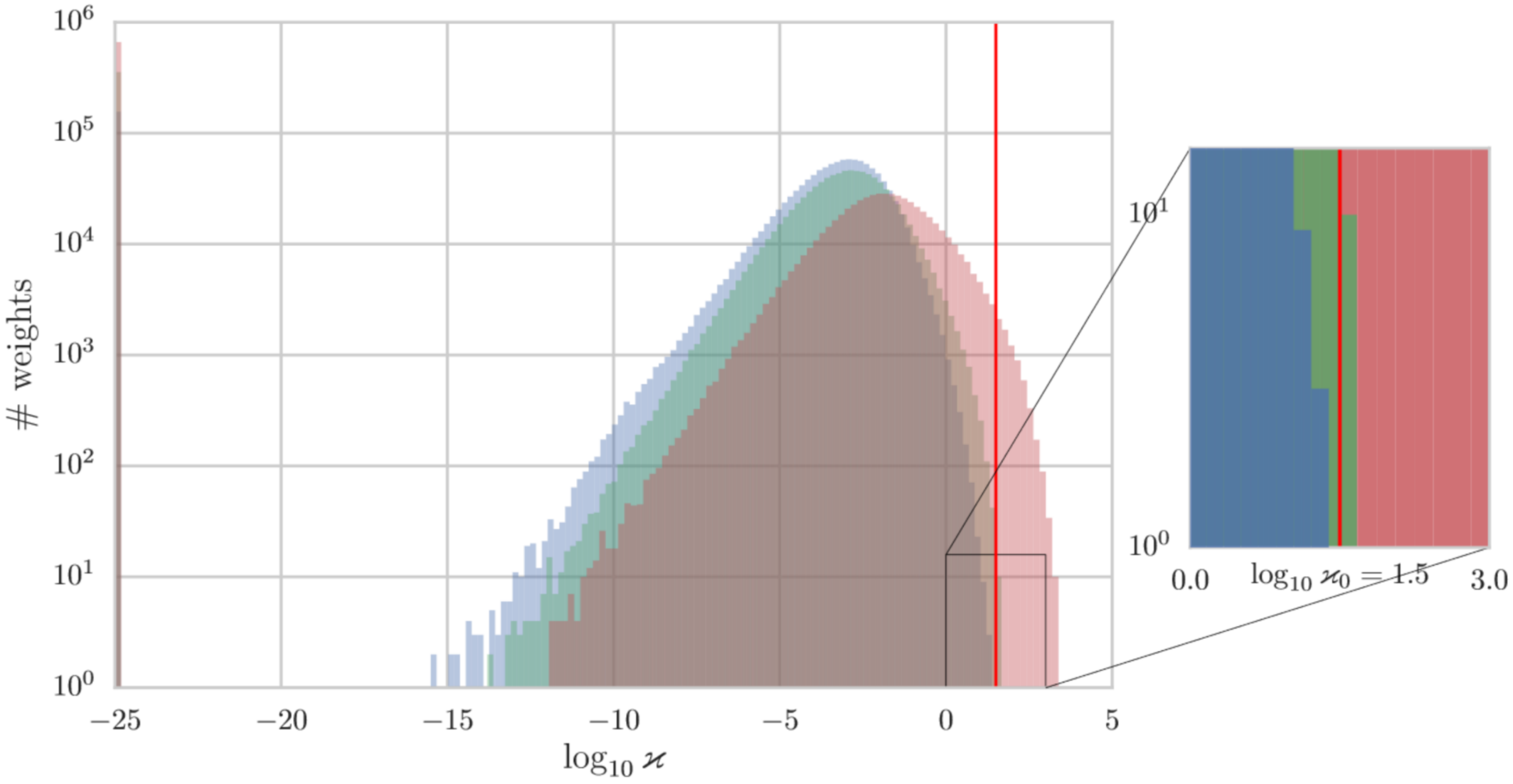
- Can propagate $\mathrm{w_{ij}} \to \mu_{ij}$ ⇔ DNN is almost linear in $w_{ij} \in (\mu_{ij} - \sigma_{ij}, \mu_{ij} + \sigma_{ij})$
- Compare the curvature with the posterior variance:

$$\varkappa_{ij} = \sigma_{ij}^2 \frac{\partial^2}{\partial W_{ij}^2} [\log p(y \,|\, x, W, W_{\text{net}})] \Big|_{W=M}$$

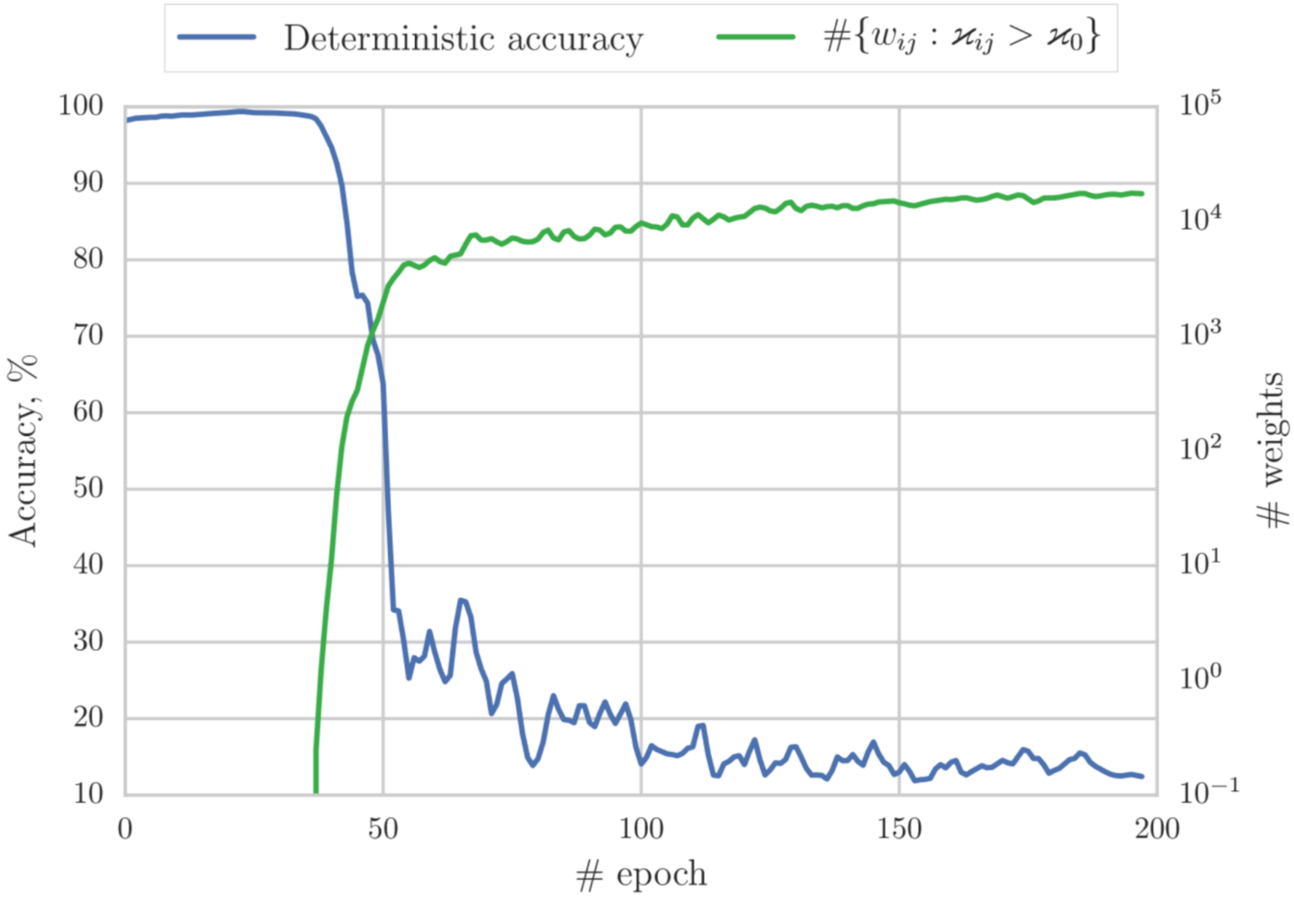


- Replace $\mathrm{w_{ij}} \to \mu_{ij}$ if $\kappa_{ij} < threshold$, else sample $\mathrm{w_{ij}} \sim q(w_{ij})$
  - Replace if σ_{ij} is small enough or if function is essentially linear

Mean propagation:

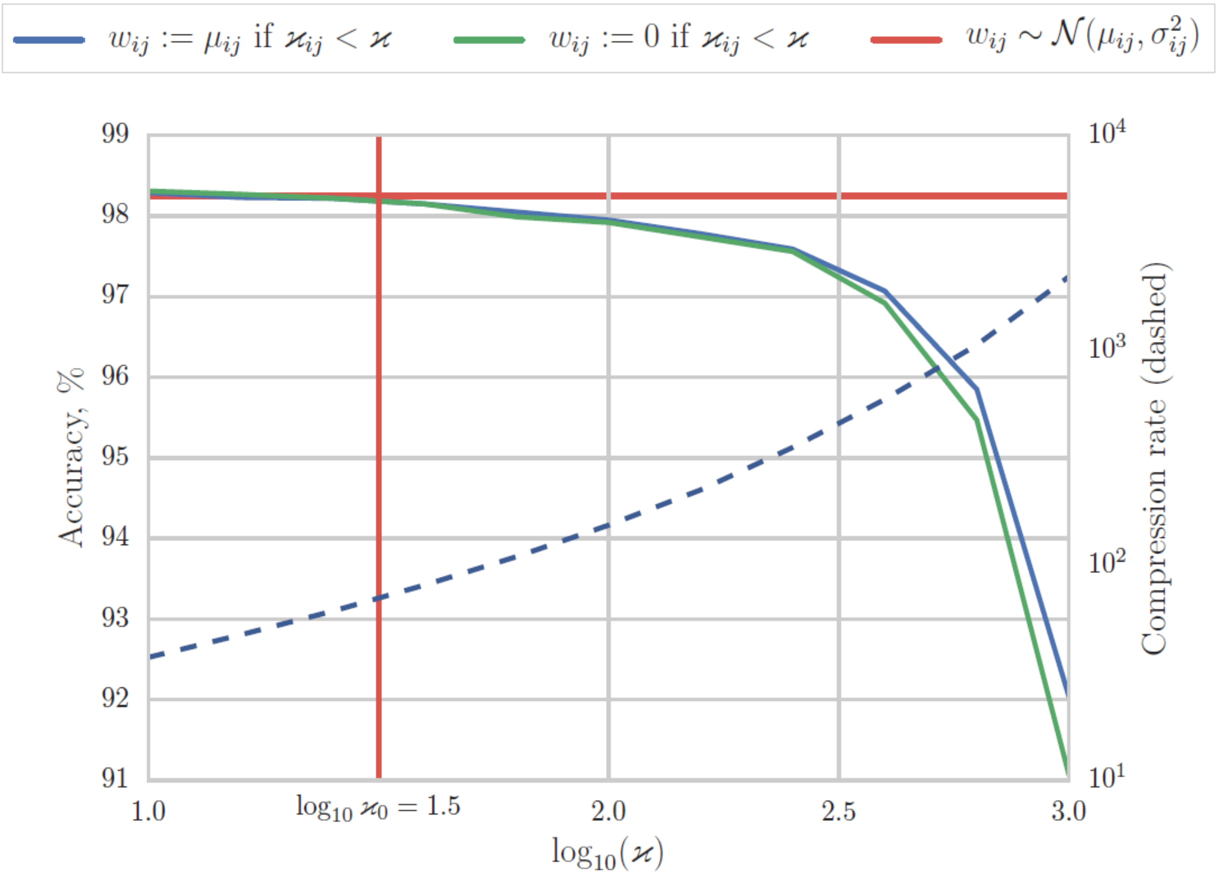


Mean propagation



Mean propagation:

- We can potentially make sigma very sparse!
- Probably sparsity follows from an optimization issue ☹



Outline

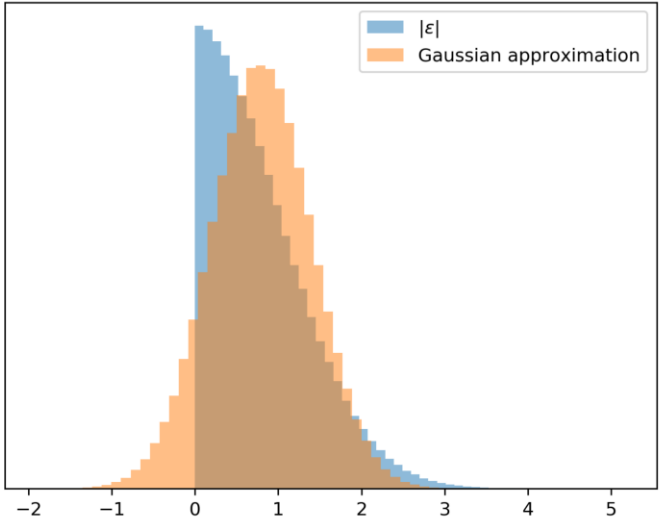
- Variance networks
- Variational dropout ⇔ variance network
- Mean propagation
- Open questions

Is it really more diverse?

- Consider a variance layer, followed by “abs” non-linearity:

$$|y_i| = \left|\epsilon_i \sqrt{(\sigma_i^2)^\top (x^2)}\right| = |\epsilon_i| \sqrt{(\sigma_i^2)^\top (x^2)}$$

- $\mathrm{E}|\epsilon_i| \approx 0.8, \sqrt{D|\epsilon_i|} \approx 0.6$
- Approximation $|\epsilon_i| \approx N(\mathrm{E}|\epsilon_i|, D|\epsilon_i|)$ has the same performance!
- Now the level of noise is similar to Gaussian dropout $N(1, 0.5)$
- Mean propagation?

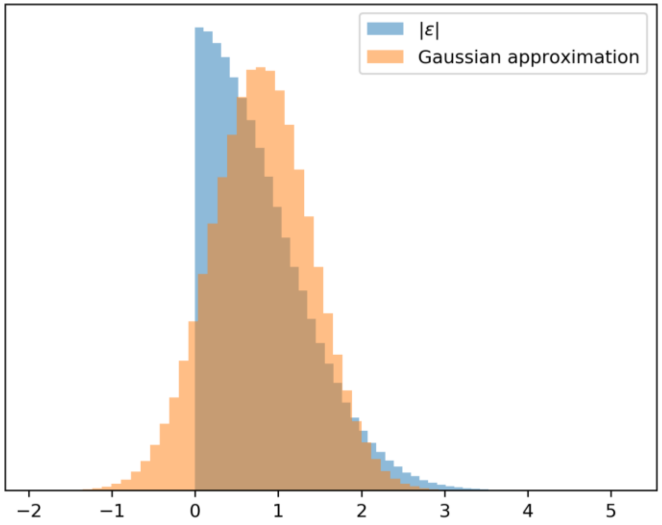


Is it really more diverse?

- So Gaussian dropout $N(1, +\infty)$ is equivalent to $N(1, 0.5)$?!
- The non-linearity is at fault
  - ReLU is better (adds binary dropout on top)

- Empirically the uncertainty of variance networks is similar to dropout
  - Out-of-domain uncertainty
  - Toy regression

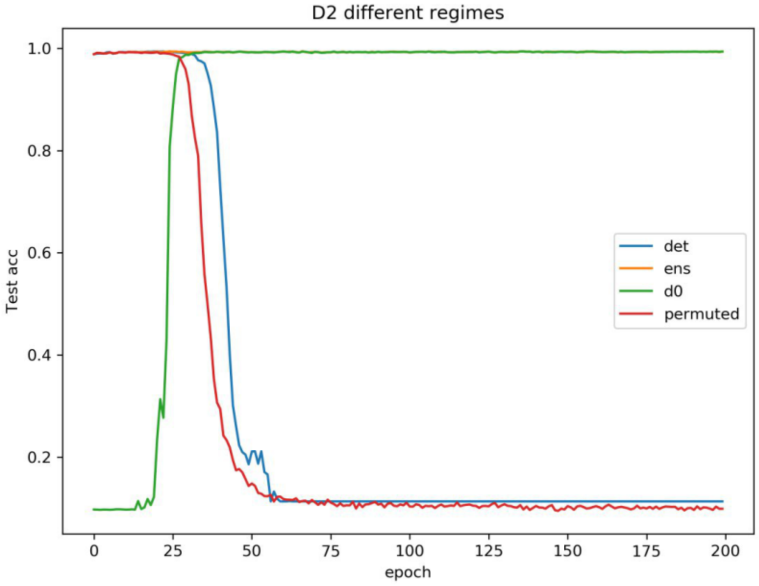
- What is the maximum effective amount of noise we can inject?

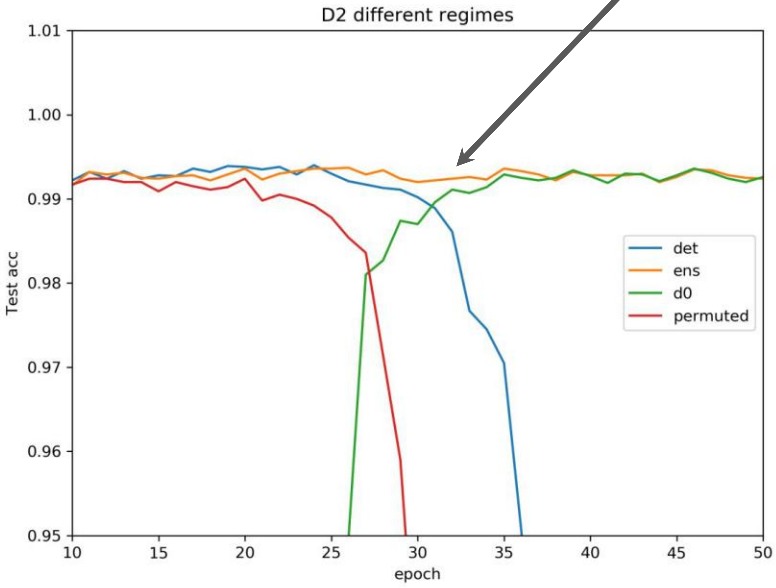


What happens during the phase transition?

- Before phase transition: information in the weights
- After phase transition: information in the variances
- During phase transition - ???

Both mean propagation and mean zeroing work well!





Variance networks

- A fun counter-intuitive model
- First practical example where mean propagation fails that hard
- Variational dropout leads to unexpected results
- We probably need better ways to approximate the posterior to obtain better ensembles…

- Why do we need variance networks? Are they any good? What are the implications of the DNN loss structure, robustness to noise, etc.?