Bayesian Compression for Deep Learning

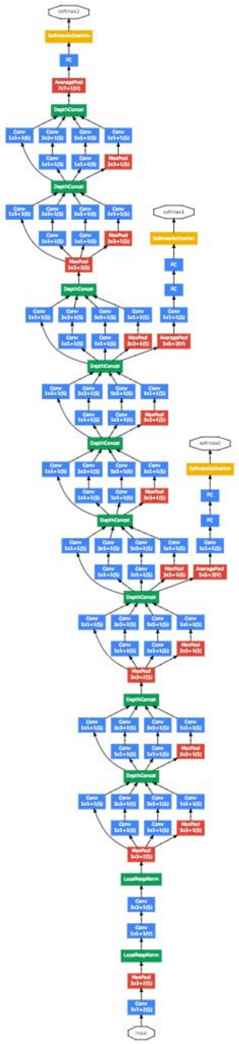
Alexander
Markovich

HIGHER SCHOOL OF ECONOMICS
NATIONAL RESEARCH
UNIVERSITY

Moscow 2018

Motivation

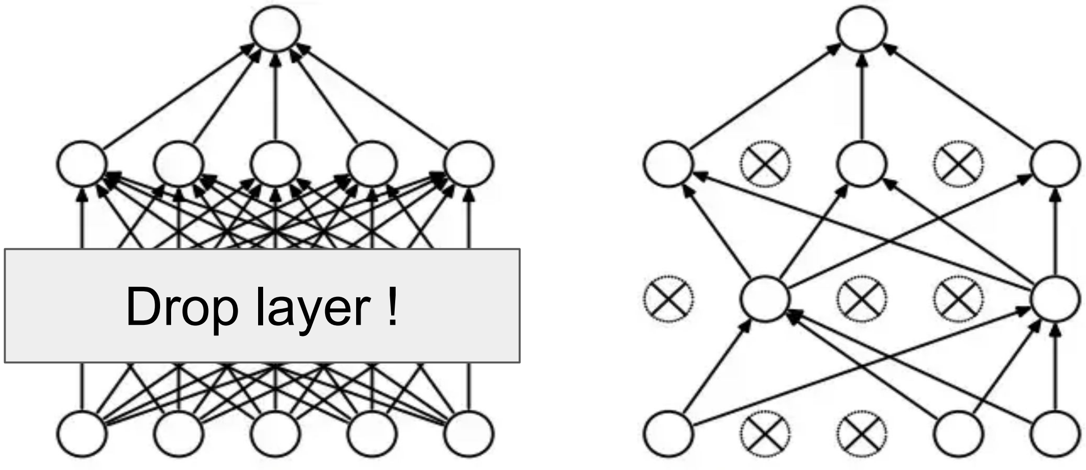
- Billions of parameters
- Slow inference
- Limited memory devices (smartphones, robots)
- Overfitting



Variety of methods

- Individual or structural sparsification / pruning
- Quantization
- Low rank approximation for weight matrices
- ...

Individual vs Structural



- Units
- Layers
- Filters
- ...


WE ARE THE BAYESIAN.
YOU WILL BE ASSIMILATED. YOUR TECHNOLOGICAL DISTINCTIVENESS WILL BE CONSIDERED A SPECIAL CASE OF OUR OWN. RESISTANCE IS FUTILE.
imgflip.com

Bayesian Inference <3

Given $$\mathcal{D} = \left\{(x_1, y_1), \ldots, (x_n, y_n)\right\}$$

Goal $$p(y \mid x, w)$$

Likelihood

Prior distribution

and then? $$p(w \mid \mathcal{D}) = \frac{p(\mathcal{D}|w)p(w)}{p(\mathcal{D})}$$

Evidence

(Stochastic) Variance Inference

$$p(w \mid \mathcal{D}) \approx q_\phi(w)$$ ← Approximation

$$\log p(\mathcal{D}) = \mathrm{ELBO} + D_{KL}(q_\phi(w) \| p(w \mid \mathcal{D}))$$

$$\mathrm{ELBO} = \underbrace{\textstyle\sum_{i=1}^{N} \mathbb{E}_{q_\phi(w)} \log p(y_i | x_i, w)}_{\text{Data term}} - \underbrace{D_{KL}(q_\phi(w) \| p(w))}_{\text{Regularization}}$$

$$\min_\phi D_{KL}(q_\phi(w) \| p(w \mid \mathcal{D})) \Leftrightarrow \max_\phi \mathrm{ELBO}$$

Relation to MDL

$$L(D) = \min_{H \in \mathcal{H}} \left\{ L(H) + L(D \mid H) \right\}$$

Hypothese Data

$$\mathrm{ELBO} = \underbrace{\mathbb{E}_{q_\phi(w)} \log p(\mathcal{D}|w)}_{\text{Error cost}} + \underbrace{\mathbb{E}_{q_\phi(w)} \log p(w) + \mathcal{H}(q_\phi(w))}_{\text{Complexity cost}}$$

(Sparse) Variational Dropout

$$B = (A \odot \Xi)W, \;\; \xi_{mi} \sim p(\xi)$$

$$A \in \mathbb{R}^{M \times I}$$
$$W \in \mathbb{R}^{I \times O}$$
$$B \in \mathbb{R}^{M \times O}$$

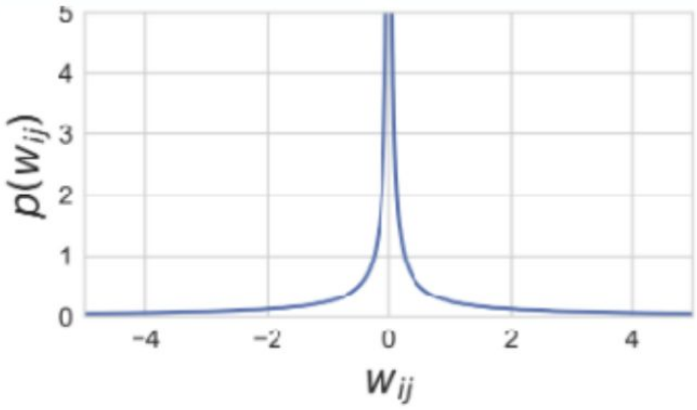
$$\xi_{mi} \sim \text{Bernoulli}(1 - p)$$ ← Binary Dropout

$$\xi_{mi} \sim \mathcal{N}(1, \alpha = \tfrac{p}{1-p})$$ ← Gaussian Dropout

$$w_{ij} = \theta_{ij}\xi_{ij} = \theta_{ij}(1 + \sqrt{\alpha}\epsilon_{ij}) \sim \mathcal{N}(w_{ij} \,|\, \theta_{ij}, \alpha\theta_{ij}^2)$$
$$\epsilon_{ij} \sim \mathcal{N}(0, 1)$$

What about prior?

(Sparse) Variational Dropout

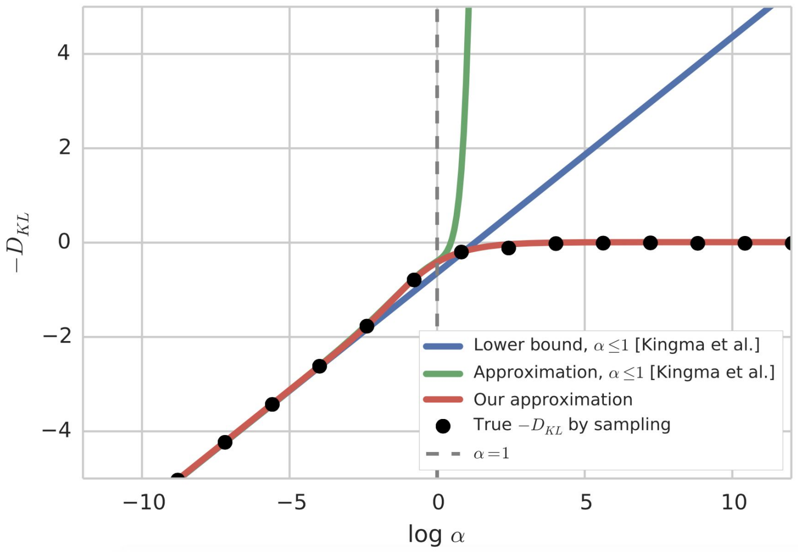


$$p(|w|) \propto \frac{1}{|w|}$$

$$q(w \mid \theta, \alpha) = \mathcal{N}(w \mid \theta, \alpha\theta^2)$$

(Sparse) Variational Dropout

$$\mathrm{ELBO} = \sum_{i=1}^{N} \mathbb{E}_{q(w|\theta,\alpha)} \log p(y_i \mid x_i, w) \boxed{- D_{KL}(q(w \mid \theta, \alpha) \| p(w))}$$



$$-D_{KL}(q(w_{ij} \mid \theta_{ij}, \alpha_{ij}) \| p(w_{ij})) = \boxed{\frac{1}{2} \log \alpha_{ij}} + \mathbb{E}_{\epsilon \sim \mathcal{N}(1, \alpha_{ij})} \log |\epsilon| + C$$

(Sparse) Variational Dropout

$$\alpha_{ij} \to \infty$$

$$\theta_{ij} \to 0, \quad \alpha_{ij}\theta_{ij}^2 \to 0$$

$$\Downarrow$$

$$q(w_{ij} \,|\, \theta_{ij}, \alpha_{ij}) \to \mathcal{N}(w_{ij} \,|\, 0, 0) = \delta(w_{ij})$$

Hierarchical prior

Scale mixture of normals

$$w \sim \mathcal{N}(w \mid 0, z^2), \quad z \sim p(z)$$

Prior

$$p(w) = \int p(z)\mathcal{N}(w \mid 0, z^2)dz$$

$$p(z) \propto \frac{1}{|z|} \qquad\qquad p(w) \propto \int \frac{1}{|z|}\mathcal{N}(w \mid 0, z^2) = \frac{1}{|w|}$$

$$p(z) \propto \sqrt{\det \mathcal{I}(z)}$$

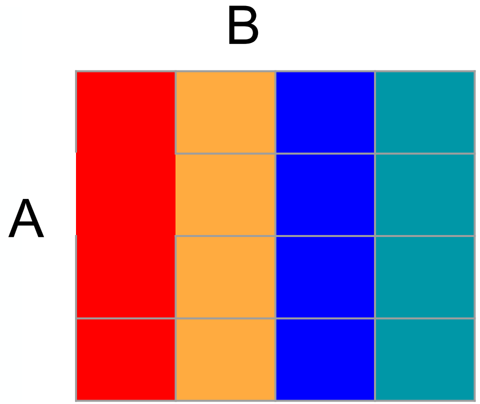
Group sparsity

$$p(W, z) \propto \prod_{i=1}^{A} \left[\frac{1}{|z_i|} \prod_{j}^{B} \mathcal{N}(w_{ij} \,|\, 0, z_i^2) \right]$$

$$q_\phi(W, z) = \prod_{i=1}^{A} \left[\mathcal{N}(z_i \,|\, \mu_{z_i}, \mu_{z_i}^2 \alpha_i) \prod_{j}^{B} \mathcal{N}(w_{ij} \,|\, z_i \mu_{ij}, z_i^2 \sigma_{ij}^2) \right]$$

Dropout rate per group

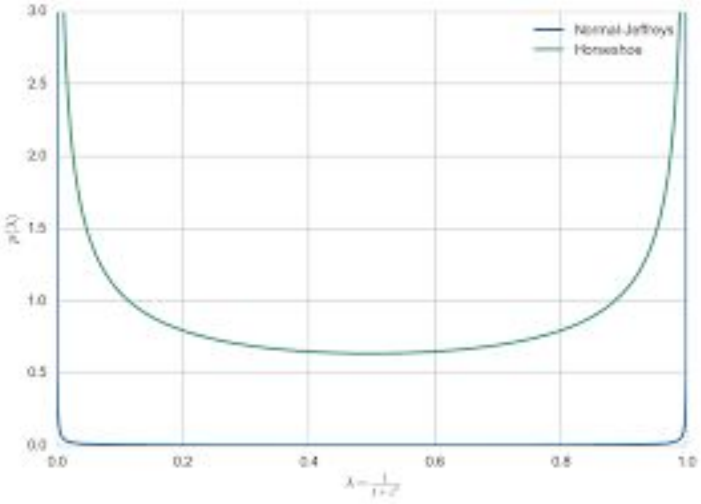
$$\log \alpha_i > \text{threshold}$$



Horseshoe prior

$$p(z) = \mathcal{C}^{+}(0, s) = 2\left(s\pi(1 + (z/s)^2)\right)^{-1}$$ half-Caushy distribution

$$s \sim \mathcal{C}^{+}(0, \tau_0); \qquad \tilde{z}_i \sim \mathcal{C}^{+}(0, 1); \qquad \tilde{w}_{ij} \sim \mathcal{N}(0, 1); \qquad w_{ij} = \tilde{w}_{ij}\tilde{z}_i s,$$



Horseshoe prior

$$s \sim \mathcal{C}^+(0, \tau_0); \qquad \tilde{z}_i \sim \mathcal{C}^+(0, 1); \qquad \tilde{w}_{ij} \sim \mathcal{N}(0, 1); \qquad w_{ij} = \tilde{w}_{ij}\tilde{z}_i s,$$

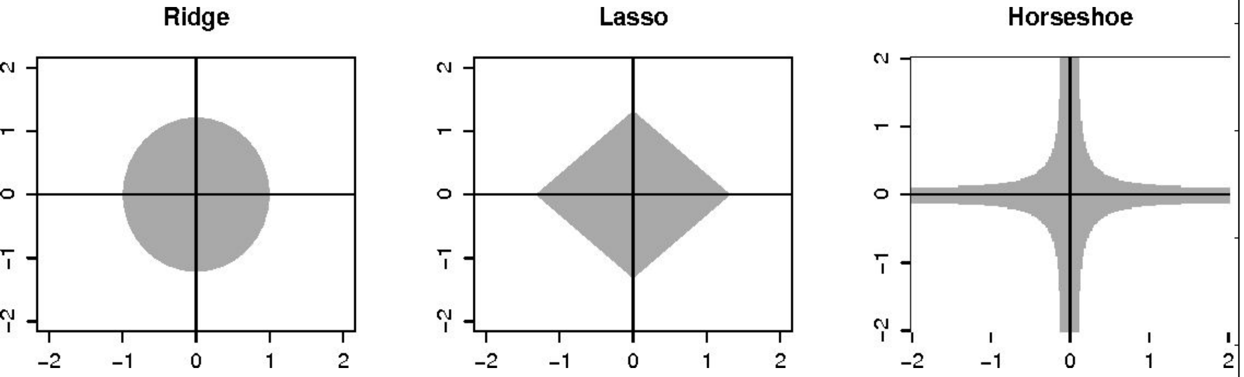
$$\downarrow$$

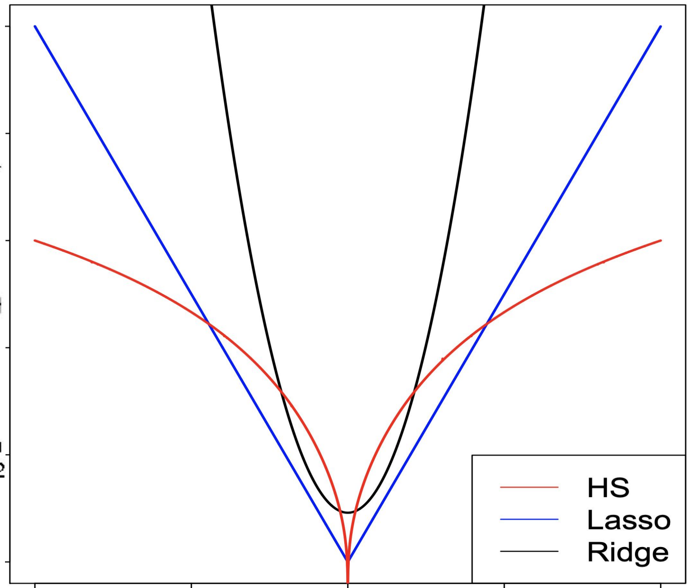
$$s_b \sim \mathcal{IG}(0.5, 1); \quad s_a \sim \mathcal{G}(0.5, \tau_0^2); \quad \tilde{\beta}_i \sim \mathcal{IG}(0.5, 1); \quad \tilde{\alpha}_i \sim \mathcal{G}(0.5, 1); \quad \tilde{w}_{ij} \sim \mathcal{N}(0, 1)$$

$$w_{ij} = \tilde{w}_{ij}\sqrt{s_a s_b \tilde{\alpha}_i \tilde{\beta}_i}.$$

Interesting fact

$\tilde{\alpha}_i, \tilde{\beta}_i \to 0 \Leftrightarrow$ Normal-Jeffreys prior = Horseshoe prior





Compression Rate

Model Original Error %	Method	$\frac{\lvert\mathbf{w}\neq 0\rvert}{\lvert\mathbf{w}\rvert}\%$	Compression Rates (Error %) Pruning	 Fast Prediction	 Maximum Compression
LeNet-300-100	DC	8.0	6 (1.6)	-	40 (1.6)
	DNS	1.8	28* (2.0)	-	-
1.6	SWS	4.3	12* (1.9)	-	64(1.9)
	Sparse VD	2.2	21(1.8)	84(1.8)	113 (1.8)
	BC-GNJ	10.8	9(1.8)	36(1.8)	58(1.8)
	BC-GHS	10.6	9(1.8)	23(1.9)	59(2.0)
LeNet-5-Caffe	DC	8.0	6*(0.7)	-	39(0.7)
	DNS	0.9	55*(0.9)	-	108(0.9)
0.9	SWS	0.5	100*(1.0)	-	162(1.0)
	Sparse VD	0.7	63(1.0)	228(1.0)	365(1.0)
	BC-GNJ	0.9	108(1.0)	361(1.0)	573(1.0)
	BC-GHS	0.6	156(1.0)	419(1.0)	771(1.0)
VGG	BC-GNJ	6.7	14(8.6)	56(8.8)	95(8.6)
8.4	BC-GHS	5.5	18(9.0)	59(9.0)	116(9.2)

Acceleration

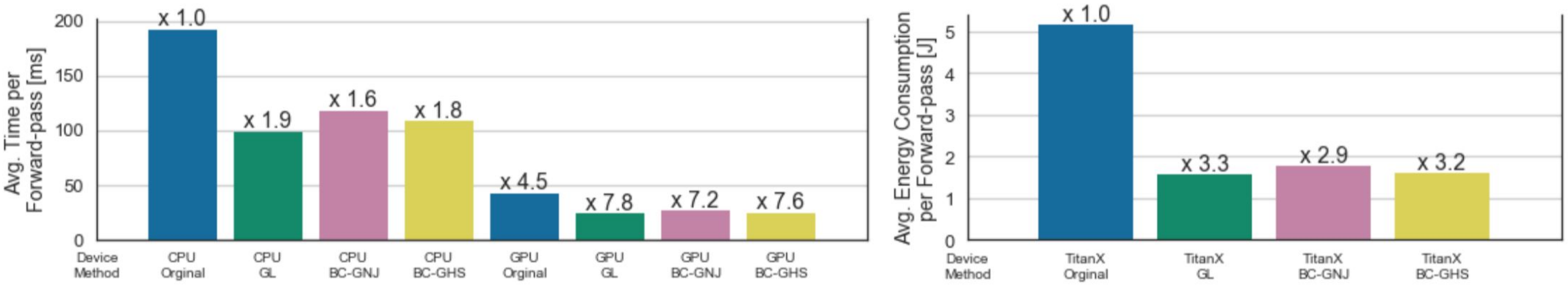


Figure 1: **Left:** Avg. Time a batch of 8192 samples takes to pass through LeNet-5-Caffe. Numbers on top of the bars represent speed-up factor relative to the CPU implementation of the original network. **Right:** Energy consumption of the GPU of the same process (when run on GPU).