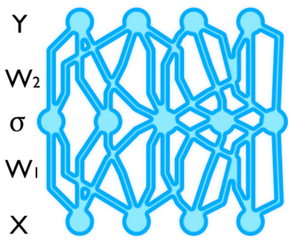
Modern Deep Learning through Bayesian Eyes

Casting modern deep learning as performing approximate inference in a Bayesian setting.

Yarin Gal (**yg279@cam.ac.uk**), Zoubin Ghahramani (**zg201@cam.ac.uk**), University of Cambridge

UNIVERSITY OF CAMBRIDGE

Modern deep learning

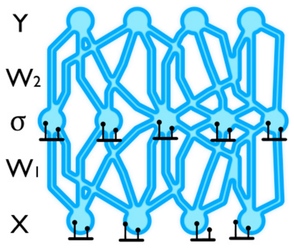


Conceptually simple models

- Tremendous attention from popular media,
- Fundamentally affected how ML is used in industry,
- Driven by pragmatic developments...
- of tractable models...
- that work well...
- and scale well.

Yet we don't understand many of these tools...

- E.g. stochastic regularisation techniques
- Used in most modern deep learning models
- Dropout randomly sets units to zero
- MGN multiplies units by $\mathcal{N}(1,1)$
- This somehow circumvents over-fitting
- And improves performance

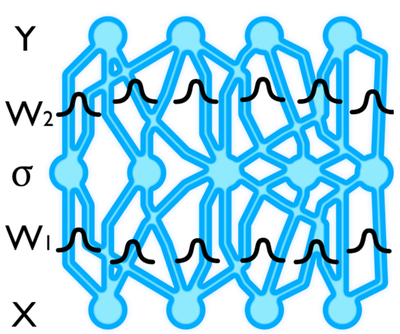


Bayesian Neural Networks

- Place prior $p(\mathbf{W}_i)$:

$$\mathbf{W}_i \sim \mathcal{N}(0, \mathbf{I})$$

for $i \leq L$ (and write $\boldsymbol{\omega} := \{\mathbf{W}_i\}_{i=1}^L$).



- Output is a r.v. $\mathbf{f}(\mathbf{x}, \boldsymbol{\omega}) = \mathbf{W}_L \sigma(...\mathbf{W}_2 \sigma(\mathbf{W}_1 \mathbf{x} + b_1)...)$.
- Softmax likelihood for classification: $p(y|\mathbf{x}, \boldsymbol{\omega}) = \text{softmax}(\mathbf{f}(\mathbf{x}, \boldsymbol{\omega}))$ or a Gaussian for regression: $p(\mathbf{y}|\mathbf{x}, \boldsymbol{\omega}) = \mathcal{N}(\mathbf{y}; \mathbf{f}(\mathbf{x}, \boldsymbol{\omega}), \tau^{-1}\mathbf{I})$.
- But difficult to evaluate posterior $p(\boldsymbol{\omega}|\mathbf{X}, \mathbf{Y})$.

Modern deep learning as Approximate inference

- Define $q_\theta(\boldsymbol{\omega})$ to approximate posterior $p(\boldsymbol{\omega}|\mathbf{X}, \mathbf{Y})$:

$$q_\theta(\boldsymbol{\omega}) = \prod q_{\mathbf{M}_i}(\mathbf{W}_i)$$
$$\mathbf{W}_i = \mathbf{M}_i \cdot \text{diag}([\mathbf{z}_{i,j}]_{j=1}^{K_i})$$
$$\mathbf{z}_{i,j} \sim \text{Bernoulli}(p_i) \text{ for } i = 1, ..., L,\ j = 1, ..., K_{i-1}$$

with $\mathbf{z}_{i,j}$ Bernoulli r.v. and variational params $\theta = \{\mathbf{M}_i\}_{i=1}^L$ (set of matrices).

- KL divergence to minimise:

$$\text{KL}(q_\theta(\boldsymbol{\omega}) \,||\, p(\boldsymbol{\omega}|\mathbf{X}, \mathbf{Y})) \propto -\int q_\theta(\boldsymbol{\omega}) \log p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\omega}) \text{d}\boldsymbol{\omega} + \text{KL}(q_\theta(\boldsymbol{\omega}) \,||\, p(\boldsymbol{\omega}))$$
$$=: \mathcal{L}(\theta)$$

- Approximate the integral with MC integration $\widehat{\boldsymbol{\omega}} \sim q_\theta(\boldsymbol{\omega})$:

$$\widehat{\mathcal{L}}(\theta) := -\log p(\mathbf{Y}|\mathbf{X}, \widehat{\boldsymbol{\omega}}) + \text{KL}(q_\theta(\boldsymbol{\omega}) \,||\, p(\boldsymbol{\omega}))$$

- Unbiased estimator converges to the same optima as $\mathcal{L}(\theta)$

$$\mathbb{E}_{\widehat{\boldsymbol{\omega}} \sim q_\theta(\boldsymbol{\omega})}(\widehat{\mathcal{L}}(\theta)) = \mathcal{L}(\theta)$$

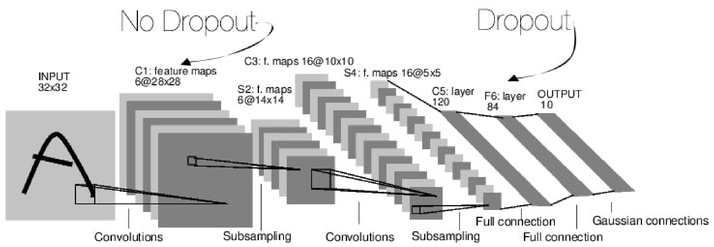
- For inference, repeat:
  - Sample $\widehat{\boldsymbol{\omega}} \sim q_\theta(\boldsymbol{\omega})$
  - And minimise w.r.t. θ (one step)

$$\widehat{\mathcal{L}}(\theta) = -\log p(\mathbf{Y}|\mathbf{X}, \widehat{\boldsymbol{\omega}}) + \text{KL}(q_\theta(\boldsymbol{\omega}) \,||\, p(\boldsymbol{\omega}))$$

= Dropout training.

Bayesian Convolutional Neural Networks

How do we use dropout et al. with convolutional neural networks (convnets)?



LeNet convnet structure (LeCun et al., 1998)

Why not use dropout et al. with convolutions?

- ~~It doesn't work~~, ~~Low co-adaptation in convolutions~~
- Because it's not used correctly
  - Standard dropout multiples weights by 0.5 with normal forwards pass
  - Instead, use predictive mean, approximated with MC integration:

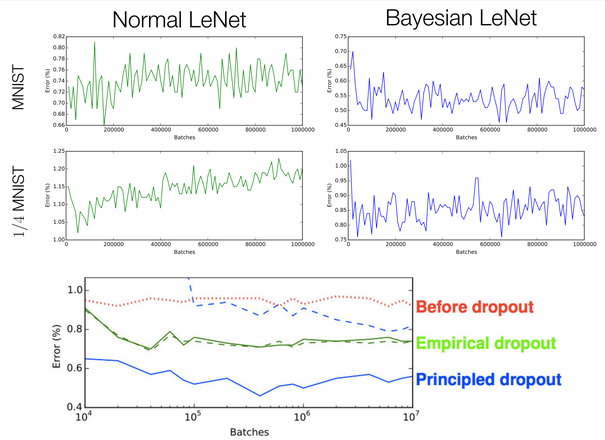
$$\mathbb{E}_{q_\theta(\mathbf{y}^*|\mathbf{x}^*)}(\mathbf{y}^*) = \frac{1}{T}\sum_{t=1}^{T} \widehat{\mathbf{y}}(\mathbf{x}^*, \widehat{\boldsymbol{\omega}}_t) \text{ with } \widehat{\boldsymbol{\omega}}_t \sim q_\theta(\boldsymbol{\omega}).$$

  - In practice, average stochastic forward passes through the network

Performing dropout after convolutions and averaging forward passes

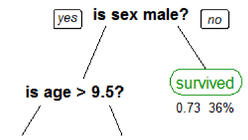
= approximate inference in Bayesian convnets.

MNIST results

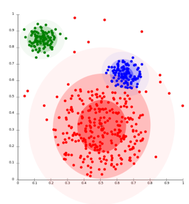


Many unanswered questions left...

- Interpretable models?
  - Rich literature in interpretable Bayesian models
  - Combine Bayesian and deep models in a principled way?



- Combine Bayesian techniques & deep models?



  - Unsupervised learning – Bayesian data analysis?
  - Bayesian models with complex data? (sequence data, image data)

- Practical deep learning uncertainty?
  - Capture language ambiguity?
  - Weight uncertainty for model debugging?

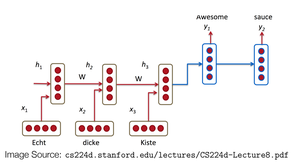
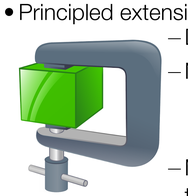


Image Source: cs224d.stanford.edu/lectures/CS224d-Lecture8.pdf

- Principled extensions of deep learning?



  - Dropout in recurrent networks?
  - New appr. distributions = new stochastic reg. techniques

$$q_\theta(\boldsymbol{\omega}) = ?$$

  - Model compression: $\mathbf{W}_i \sim$ discrete distribution with continuous base measure?

Work in progress!

Full paper: "On Modern Deep Learning and Variational Inference". Photos taken from Wikimedia or original work.