Learning Weight Uncertainty with Stochastic Gradient MCMC for Shape Classification

Presenter: Chunyuan Li

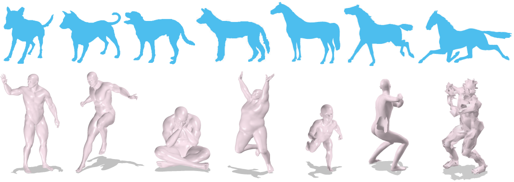
Chunyuan Li, Andrew Stevens, Changyou Chen,
Yunchen Pu, Zhe Gan, Lawrence Carin

Duke University

June 30, 2016

Deep Neural Nets for Shape Representations

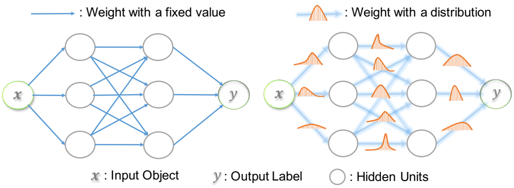
- Shapes in the real-world manifest rich variability.



- Learning deep representations of shapes with DNNs.
- While SGD with Backpropagation is popular, issues exist:
  - Overfitting: Make overly confident decisions on prediction

Stochastic Gradient MCMC: Motivation

- Weight Uncertainty of DNNs



Posterior inference of weight distributions

- Bring MCMC back to CV community to tackle “big data"
  - Traditional MCMC: was popular in CV a decade ago
    - inclduing Gibbs sampling, HMC, MH, etc; NOT scalable
  - Propose to use scalable MCMC to fill the gap

Stochastic Gradient MCMC: Algorithm

- Implementation
  1. Training: Adding noise to parameter update
  2. Testing: Model averaging
- SG-MCMC algorithms and their optimization counterparts

Algorithms	**SG-MCMC**	**Optimization**
Basic	SGLD	SGD
Preconditioning	pSGLD[⋆]	RMSprop/Adagrad
Momentum	SGHMC	momentum SGD
Thermostat	SGNHT	Santa[◇]

[⋆] *Preconditioned Stochastic Gradient Langevin Dynamics for Deep Neural Networks* Li et al, AAAI 2016

[◇] *Bridging the Gap between Stochastic Gradient MCMC and Stochastic Optimization* Chen et al, AISTATS 2016

Interpretation of Dropout and Batch Normaliation

- Dropout/DropConnect and SGLD share the same form of update rule, with the only difference being that the level of injected noise is different.
- The integration of binary Dropout with SG-MCMC can be viewed as learning weight uncertainty of mixtures of neural networks.
- Batch-Normalization can accelerate SG-MCMC training. It helps prevent the sampler from getting stuck in the saturated regimes of nonlinearities.

Results: Applications to Shape Classification

- A variety of 2D and 3D datasets
  – inclduing SHREC and ShapeNet etc

- Empirical observations
  - The use of Bayesian learning (SG-MCMC or Dropout) slows down training initially. This is likely due to the higher uncertainty imposed during learning, resulting in more exploration of parameter space.
  - Increased uncertainty, however, prevents overfitting and eventually results in improved performance.