Bayesian Neural Networks

ROY team: Ilya Zharikov,
Roman Isachenko,
Artem Bochkarev

Skolkovo Institute of Science and Technology
Bayesian Methods course

May 25, 2017

Project goal

Goal

Estimate posterior distributions of the model parameters from data

Problem

Monte Carlo sampling is very slow for high-dimensional data

Probabilistic Programming:

- Uncertainty in predictions;
- Uncertainty in representations;
- Regularizations with priors;
- Transfer learning;
- Hierarchical Neural Networks.

Related work

1. Salvatier J, Wiecki T. V., Fonnesbeck C. Probabilistic programming in Python using PyMC3. // *PeerJ Computer Science*. 2016.
2. Blundell C. et al. Weight Uncertainty in Neural Network // *Proceedings of The 32nd International Conference on Machine Learning*. 2015.
3. Kucukelbir A. et al. Automatic Differentiation Variational Inference // *arXiv preprint arXiv:1603.00788*. – 2017.

Problem Statement

Inference problem

Bayes' theorem states: $\mathbb{P}(\boldsymbol{\theta} \mid \mathbf{X}) = \dfrac{\mathbb{P}(\mathbf{X} \mid \boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta})}{\mathbb{P}(\mathbf{X})}$

Maximum A Posteriori (MAP) estimation

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \left[\ln \mathbb{P}(\boldsymbol{\theta} \mid \mathbf{X})\right] = \arg\max_{\boldsymbol{\theta}} \left[\ln \mathbb{P}(\mathbf{X} \mid \boldsymbol{\theta}) + \ln \mathbb{P}(\boldsymbol{\theta})\right]$$

Monte Carlo approach:

- Metropolis-Hastings sampling;
- Gibbs sampling;
- No-U-Turn Sampling (NUTS).

Variational Inference

Goal

Approximate posterior distribution $p(X, \theta)$ by function $q(\theta)$ from parametric family.

$$
\begin{array}{ccc}
\ln p(\mathbf{X}) = & \mathrm{KL}(q||p) & + \; \mathrm{ELBO}(q) \\
& \Updownarrow & \Updownarrow \\
& \displaystyle\int q(\boldsymbol{\theta}) \ln \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\mathbf{X})} \mathrm{d}\boldsymbol{\theta} & \displaystyle\int q(\boldsymbol{\theta}) \ln \frac{p(\mathbf{X}, \boldsymbol{\theta})}{q(\boldsymbol{\theta})} \mathrm{d}\boldsymbol{\theta}
\end{array}
$$

Minimization of $\mathbf{KL(q||p)}$ $\Leftrightarrow$ Maximization of $\mathbf{ELBO(q)}$

Automatic Differentiation Variational Inference (ADVI)

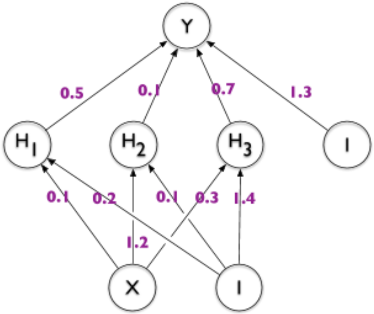
- Automatic transformation of constrained variables $\boldsymbol{\zeta} = T(\boldsymbol{\theta})$;
  Example: $\theta \in \mathbb{R}_+ \Rightarrow \zeta = T(\theta) = \log \theta$, then $\zeta \in \mathbb{R}$.
- $q(\boldsymbol{\zeta}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is diagonal;

$$\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^* = \arg\max_{\boldsymbol{\mu}, \boldsymbol{\Sigma}} \mathrm{ELBO}(q)$$

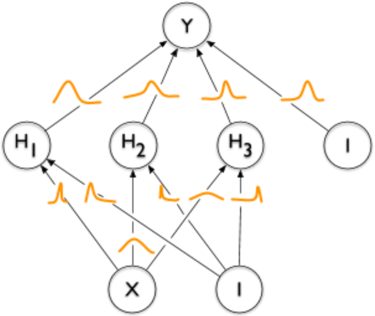
- Stochastic optimization;
- Reparametrization trick to apply automatic differentiation;
- Adaptive step-size.

Deep Learning

Neural Networks
Predict values of parameters by fitting complex model on the huge dataset



Bayesian Neural Networks
Predict the parameters of the weights distributions from the dataset



---

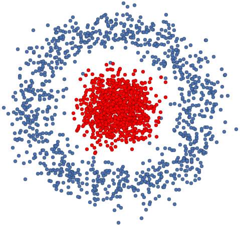
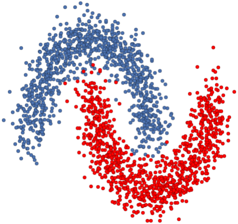
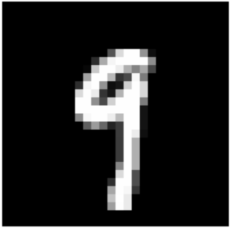
http://bit.ly/2rMQuDq

Experiments

Goals:

- investigate influence of different priors on the predictions
- visualize uncertainties in predictions
- analyze the model behaviour

Datasets:

Course of work

Priors: Normal, Laplace, Flat, Cauchy
Hyperpriors:

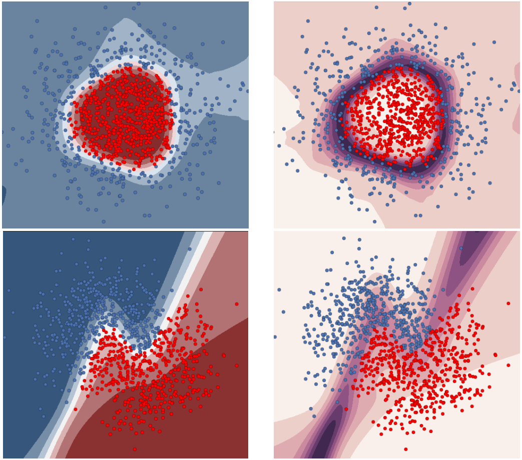
- fixed values of hyperparameters
- Hierarchical modelling: Half-Normal, Half-Cauchy, Inverse-Gamma

Synthetic data

Posterior Probability | Uncertainty

Prior: Cauchy
Hyperprior:
Inverse-Gamma
Accuracy: 0.735

Prior: Normal
Hyperprior:
Inverse-Gamma
Accuracy: 0.851

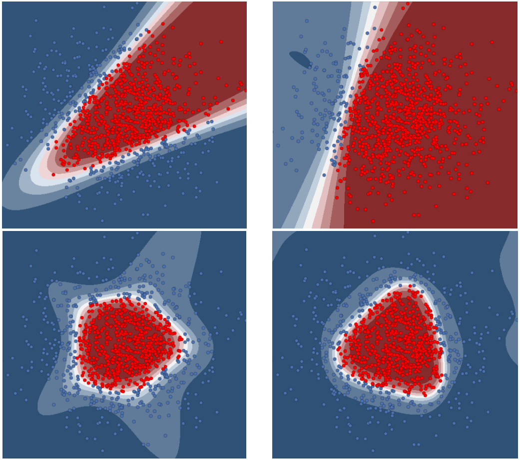


Synthetic data

Normal prior | Laplace prior

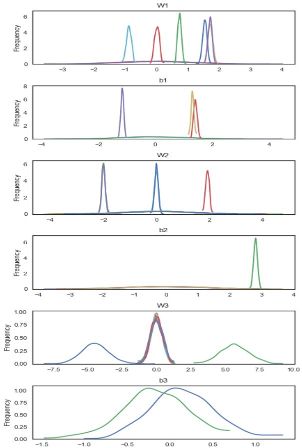
Fixed hyperparameters

Inverse-Gamma hyperprior on scaling parameter

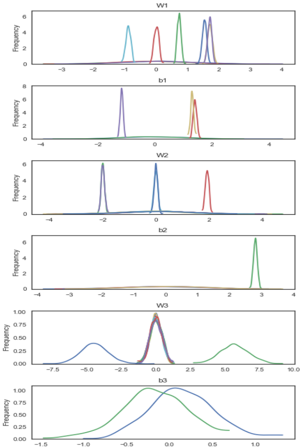


Synthetic data

Laplace prior does indeed provide sparser solutions compared to cauchy or gauss.

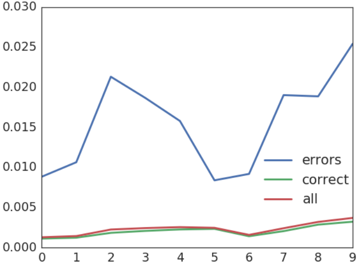


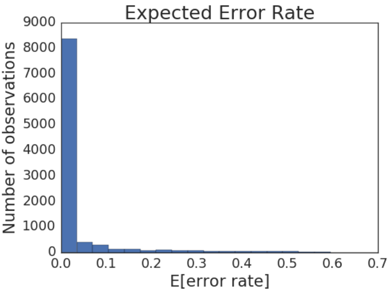
Laplace prior



Cauchy prior

MNIST



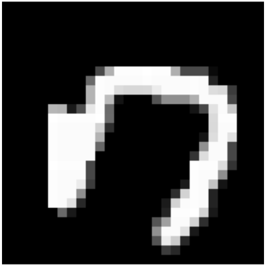


- Accuracy score: 97.7%
- Variance is much higher for misclassified pictures
- Model is not always confident

MNIST

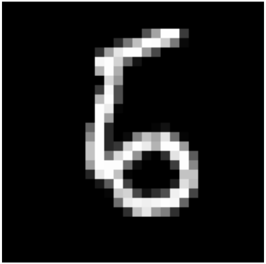
Misclassified pictures with zero expected error rate:

True: 7
Prediction: 0



True: 9
Prediction: 7

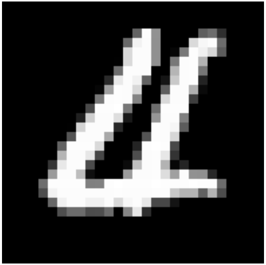
True: 5
Prediction: 6



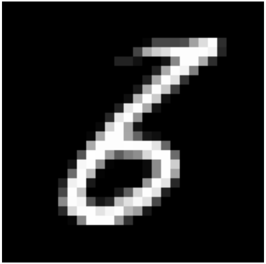
MNIST

Pictures with the lowest confidence

True: 4
Prediction: 0



True: 6
Prediction: 8



True: 1
Prediction: 1