

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.

- 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: $\mathbb{P}(\boldsymbol{\theta} | \mathbf{X}) = \frac{\mathbb{P}(\mathbf{X} | \boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta})}{\mathbb{P}(\mathbf{X})}$

Maximum A Posteriori (MAP) estimation

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

Monte Carlo approach:

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

Goal

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

$$\begin{array}{ccc} \ln p(\mathbf{X}) = \text{KL}(q||p) + \text{ELBO}(q) & & \\ \updownarrow & & \updownarrow \\ \int q(\theta) \ln \frac{q(\theta)}{p(\theta|\mathbf{X})} d\theta & & \int q(\theta) \ln \frac{p(\mathbf{X}, \theta)}{q(\theta)} d\theta \end{array}$$

Minimization of **KL(q||p)** \Leftrightarrow Maximization of **ELBO(q)**

Automatic Differentiation Variational Inference (ADVI)

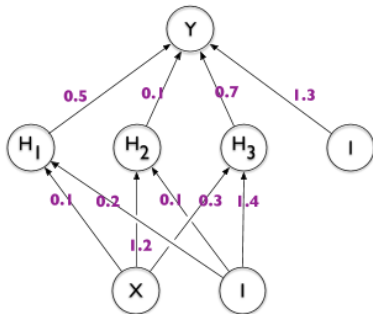
- Automatic transformation of constrained variables $\zeta = T(\theta)$;
Example: $\theta \in \mathbb{R}_+ \Rightarrow \zeta = T(\theta) = \log \theta$, then $\zeta \in \mathbb{R}$.
- $q(\zeta) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is diagonal;

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

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

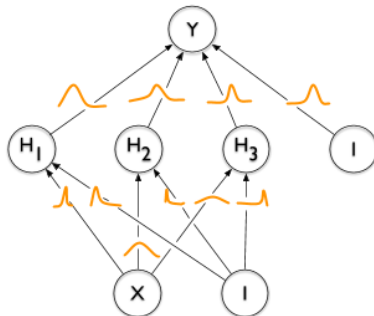
Neural Networks

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



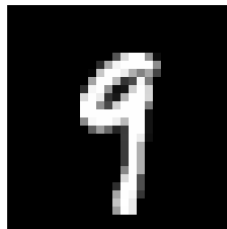
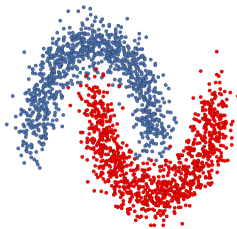
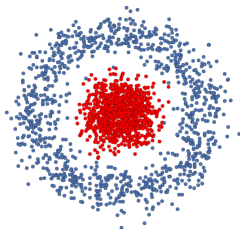
<http://bit.ly/2rMQuDq>

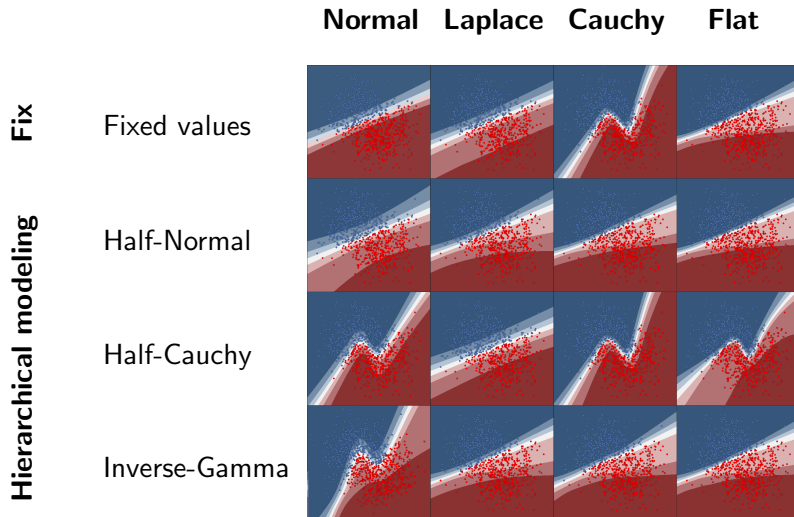
Experiments

Goals:

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

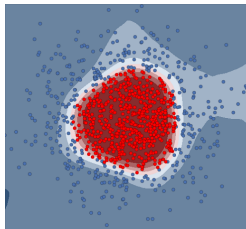
Datasets:



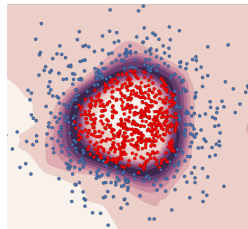


Synthetic data

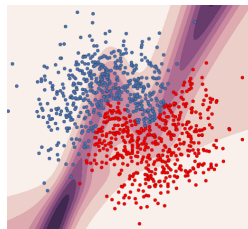
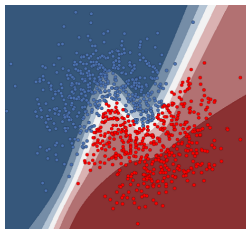
**Posterior
Probability**



Uncertainty



Prior: Cauchy
Hyperprior:
Inverse-Gamma
Accuracy: 0.735



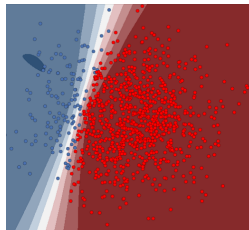
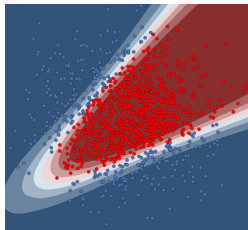
Prior: Normal
Hyperprior:
Inverse-Gamma
Accuracy: 0.851

Synthetic data

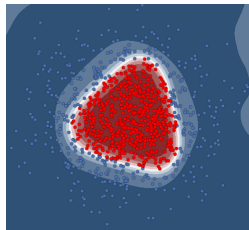
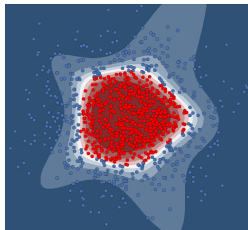
Prior: Normal

Prior: Laplace

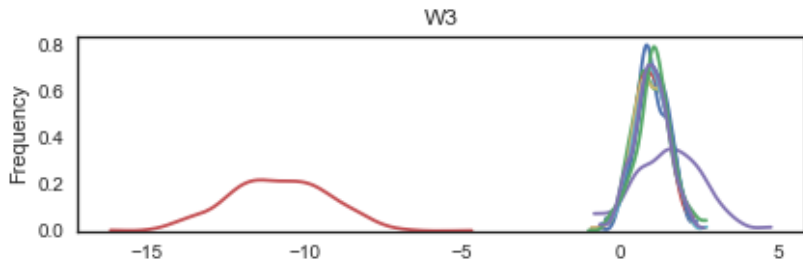
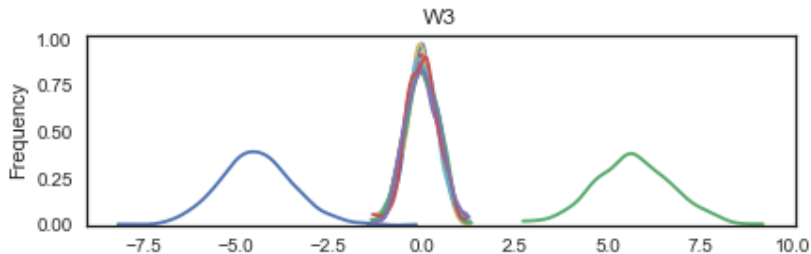
Hyperprior:
Fixed values

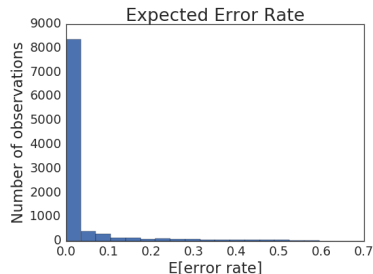
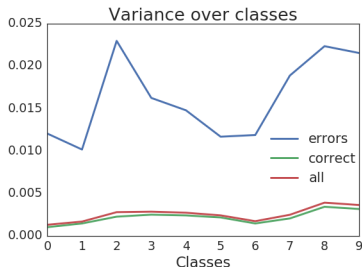


Hyperprior:
Inverse-Gamma



Synthetic data



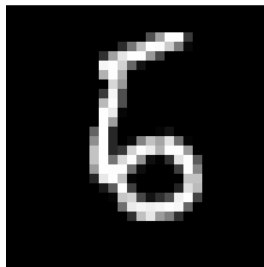
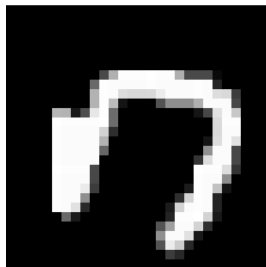


Conclusions:

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

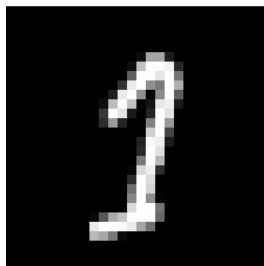
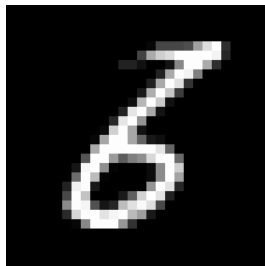
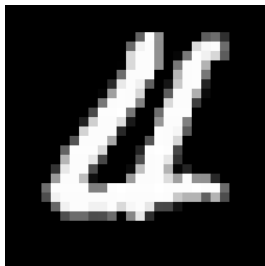
Misclassified pictures with **zero expected error rate**:

True	Prediction	True	Prediction	True	Prediction
7	0	9	7	5	6



Pictures with the **lowest confidence**:

True	Prediction	True	Prediction	True	Prediction
4	0	6	8	1	1



Conclusion

- Posterior distribution helps to make conclusions about uncertainties
- Variational inference allows to approximate posterior distribution for high-dimensional data
- Hierarchical models have more degrees of freedom