Projected Bayesian Neural Networks:

Avoiding weight-space pathologies by learning latent representations of neural network weights

ACML, Nov 17th, 2019

Melanie F. Pradier

CRCS Center for Research on Computation and Society
at Harvard John A. Paulson School of Engineering and Applied Sciences

HDSI | Harvard Data Science Initiative

Projected Bayesian Neural Networks:

Avoiding weight-space pathologies by learning latent representations of neural network weights

Joint work with my collaborators...


Weiwei Pan


Jiayu Yao


Soumya Ghosh


Finale Doshi-Velez

Projected Bayesian Neural Networks:

Avoiding weight-space pathologies by learning latent representations of neural network weights

Nov 17th, 2019

Melanie F. Pradier

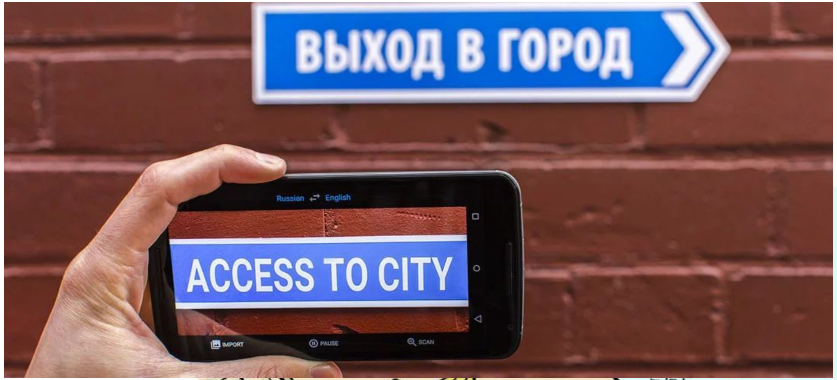
CRCS Center for Research on Computation and Society at Harvard John A. Paulson School of Engineering and Applied Sciences

HDSI | Harvard Data Science Initiative

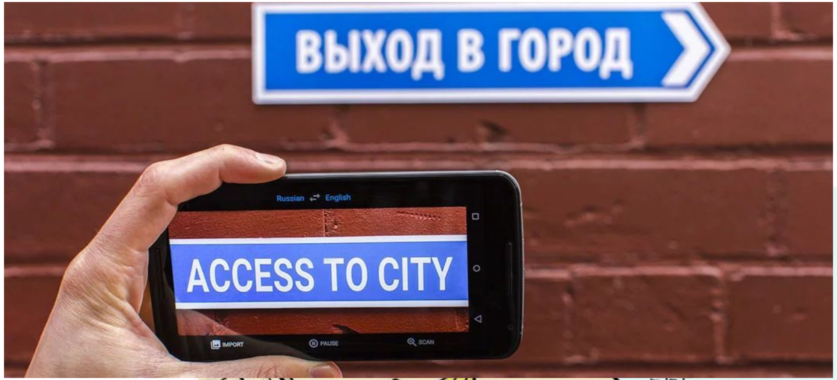
Success of Deep Learning

Success of Deep Learning

Success of Deep Learning

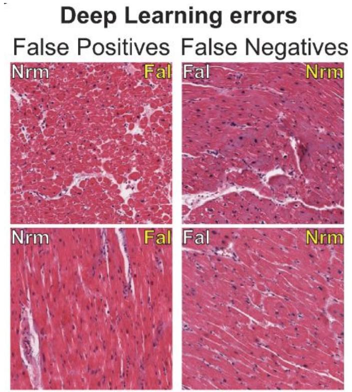


Success of Deep Learning



Huge amount of opportunities...

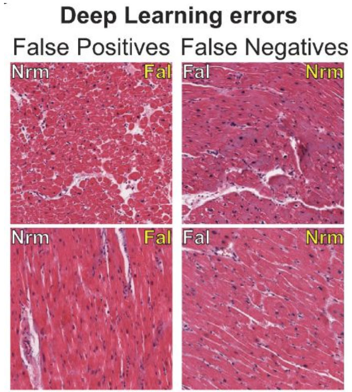
...but careful in high-stake decisions!



[Nirschi et.al, 2018]

[Eykholt et.al, 2018]

...but careful in high-stake decisions!



[Nirschi et.al, 2018]

[Eykholt et.al, 2018]

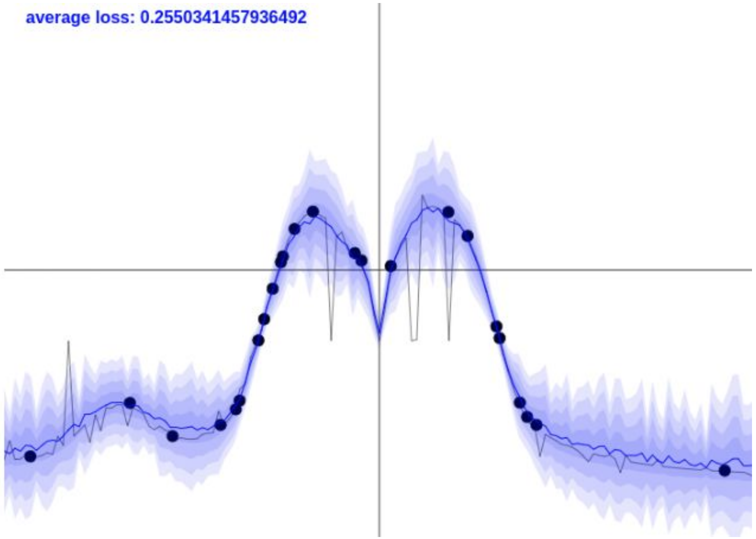
Our Goal: Quantify Uncertainty

With such uncertainty, we can:

- Alert humans in unclear situations
- Diagnose ML systems (when and how does it fail)
- Get better predictive accuracy

Focus: Bayesian Neural Networks

Bayesian Neural Network (BNN)



[What my deep model does not know, post of Yarin Gal, 2015]

$$\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{N}$$

$$\boldsymbol{y} = f_{\boldsymbol{w}}(\boldsymbol{x}) + \boldsymbol{\epsilon}$$

$$\boldsymbol{w} \sim \mathcal{N}(0, \sigma_w^2 \mathbf{I}), \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma_\epsilon^2 \mathbf{I})$$

Quantities of interest:

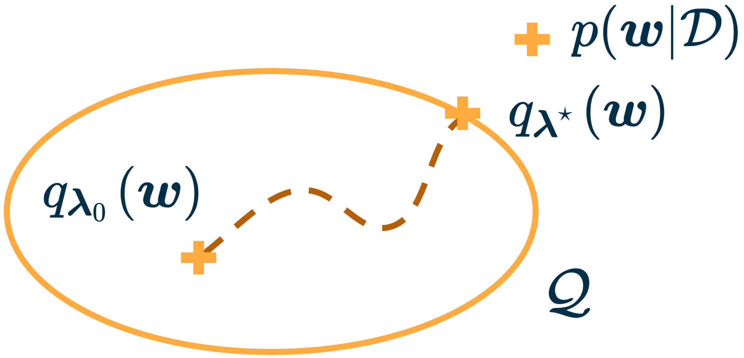
- Posterior of the weights $p(\boldsymbol{w}|\mathcal{D})$
- Predictive distribution

$$p(\mathbf{y}^\star|\mathbf{x}^\star, \mathcal{D}) = \int p(\mathbf{y}^\star|\mathbf{x}^\star, \boldsymbol{w}) p(\boldsymbol{w}|\mathcal{D}) d\boldsymbol{w}$$

Key challenge: $p(\boldsymbol{w}|\mathcal{D})$ intractable

Variational Inference: appealing for scalability

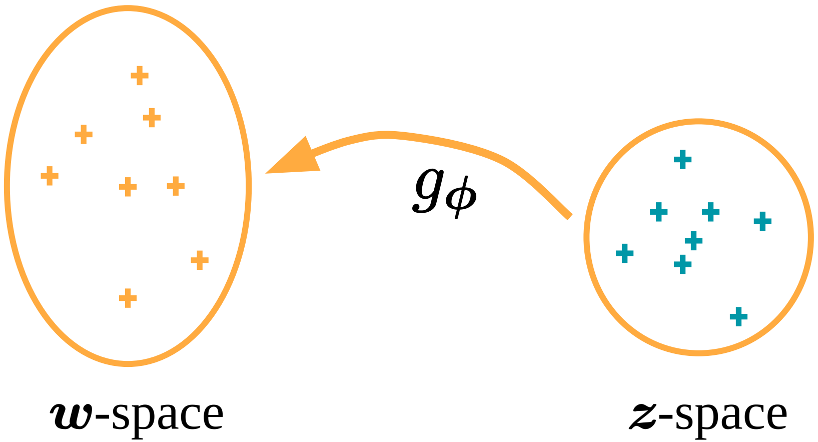
- Mean-field Approx. [Graves, 1993] [Blundell et.al, 2015]
- Structured Variational Approximations
  - Multivariate Gaussians [Louizos et.al, 2016; Sun et.al, 2017]
  - Hierarchical Variational Models [Ranganath et.al, 2016]
- Normalizing Flows and Transformations
  - Multiplicative Normalizing Flow [Louizos et. al, 2017]
  - Hypernetworks [Krueger et.al, 2017; Pawlowski et.al, 2017]



$q_{\boldsymbol{\lambda}}(\boldsymbol{w}) \in \mathcal{Q}$

Our Approach

Projected Bayesian NN (proj-BNN)



$$D_w \gg D_z$$

$$\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^N$$

$$\boldsymbol{y} = f_{\boldsymbol{w}}(\boldsymbol{x}) + \boldsymbol{\epsilon} \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma_\epsilon^2 \mathbf{I})$$

$$\boldsymbol{w} = g_{\boldsymbol{\phi}}(\boldsymbol{z}), \quad \boldsymbol{z} \sim p(\boldsymbol{z}), \quad \boldsymbol{\phi} \sim p(\boldsymbol{\phi}),$$

Quantities of interest:

- Posterior of the weights $p(z, \phi | \mathcal{D})$
- Predictive distribution

$$p(\mathbf{y}^\star | \mathbf{x}^\star, \mathcal{D}) = \int p(\mathbf{y}^\star | \mathbf{x}^\star, z, \phi) p(z, \phi | \mathcal{D}) d\boldsymbol{w}$$

How about inference?

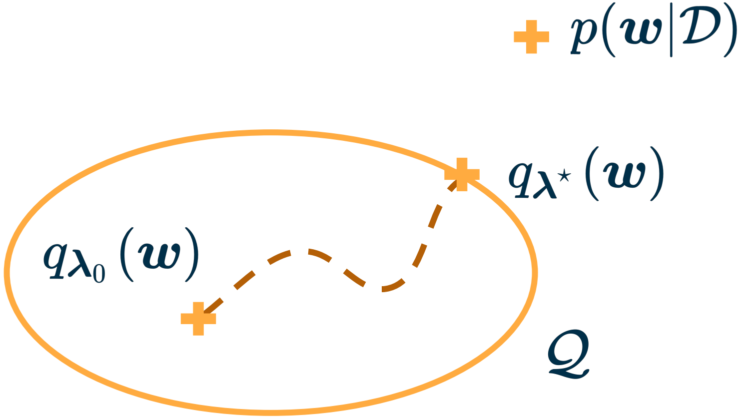
Objective: approximate $p(\boldsymbol{w}|\mathcal{D})$

$$q_{\boldsymbol{\lambda}}(\boldsymbol{w}) \in \mathcal{Q}$$

$$\underset{\boldsymbol{\lambda}^\star}{\operatorname{argmin}}\ D_{\mathrm{KL}}\Big(q_{\boldsymbol{\lambda}}(\boldsymbol{w})||p(\boldsymbol{w}|\mathcal{D})\Big)$$

$$\Updownarrow$$

$$\underset{\boldsymbol{\lambda}^\star}{\operatorname{argmax}}\ \mathcal{L}(\boldsymbol{\lambda}) = \mathbb{E}_q\Big[\log p\big(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}\big)\Big] - D_{\mathrm{KL}}\big(q_{\boldsymbol{\lambda}}(\boldsymbol{w})||p(\boldsymbol{w})\big)$$



Black-box VI [Ranganath et.al, 2013] + reparametrization trick [Kingma et.al, 2014; Rezende et.al, 2015]

How about inference?

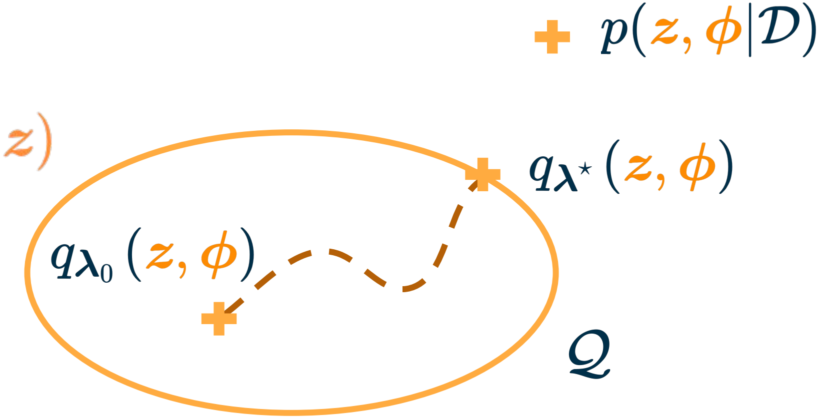
Objective: approximate $p(\boldsymbol{z}, \boldsymbol{\phi} | \mathcal{D})$

$$\boldsymbol{z} \sim q_{\boldsymbol{\lambda}_z}(\boldsymbol{z}), \quad \boldsymbol{\phi} \sim q_{\boldsymbol{\lambda}_\phi}(\boldsymbol{\phi}), \quad \boldsymbol{w} = g_{\boldsymbol{\phi}}(\boldsymbol{z})$$

$$\underset{\boldsymbol{\lambda}^\star}{\operatorname{argmin}}\ D_{\mathrm{KL}}\left(q_{\boldsymbol{\lambda}}(\boldsymbol{z}, \boldsymbol{\phi}) || p(\boldsymbol{z}, \boldsymbol{\phi} | \mathcal{D})\right)$$

$$\Updownarrow$$

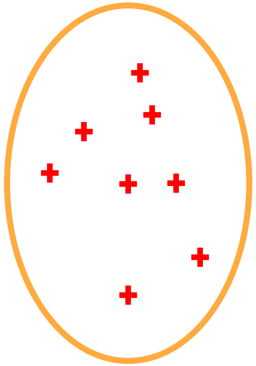
$$\underset{\boldsymbol{\lambda}^\star}{\operatorname{argmax}}\ \mathcal{L}(\boldsymbol{\lambda}) = \mathbb{E}_q\left[\log p(\boldsymbol{y} | \boldsymbol{x}, g_{\boldsymbol{\phi}}(\boldsymbol{z}))\right] - D_{\mathrm{KL}}\left(q_{\boldsymbol{\lambda}_z}(\boldsymbol{z}) || p(\boldsymbol{z})\right) - D_{\mathrm{KL}}\left(q_{\boldsymbol{\lambda}_\phi}(\boldsymbol{\phi}) || p(\boldsymbol{\phi})\right)$$



Black-box VI [Ranganath et.al, 2013] + reparametrization trick [Kingma et.al, 2014; Rezende et.al, 2015]

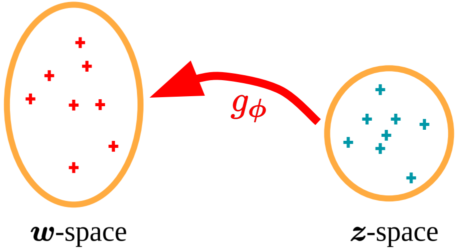
Smart-initialization Procedure

1. Characterize weight space



Sample multiple weight sets [Izmailov et.al, 2018]

2. Find point estimate g_ϕ



Train an autoencoder

3. Black-box VI (BBVI)

$$D_{\mathrm{KL}}\left(q_{\boldsymbol{\lambda}}(\boldsymbol{z},\boldsymbol{\phi})||p(\boldsymbol{z},\boldsymbol{\phi}|\mathcal{D})\right)$$

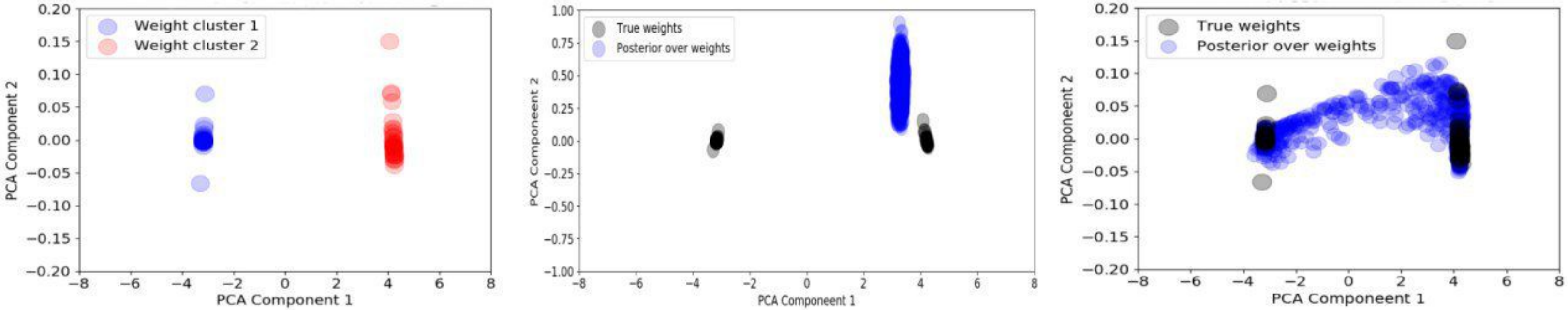
BBVI with smart initialization

Results

Results: Illustrative Toy Example

[Blundell et.al, 2015]

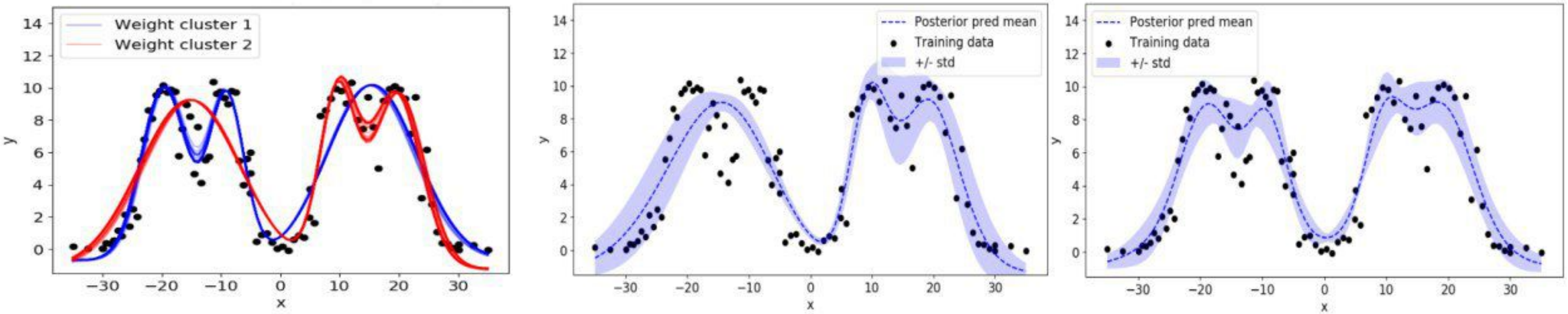
[Pradier et.al, 2018]



(a) Projection of true weights

(b) BbB posterior over weights

(c) Proj-BNN posterior over weights

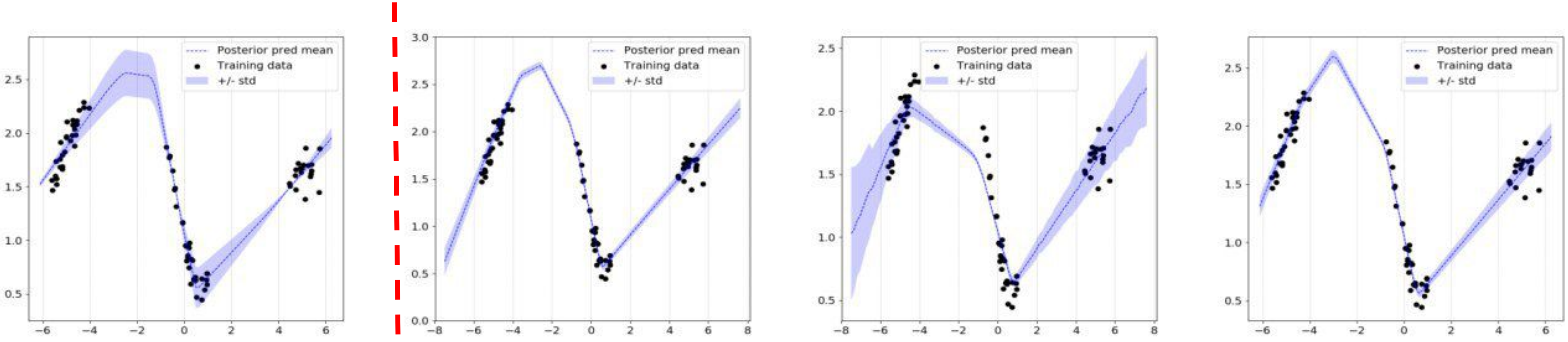


(d) Functions from true weights

(e) BbB posterior predictive

(f) Proj-BNN posterior predictive

Results: Uncertainty estimation

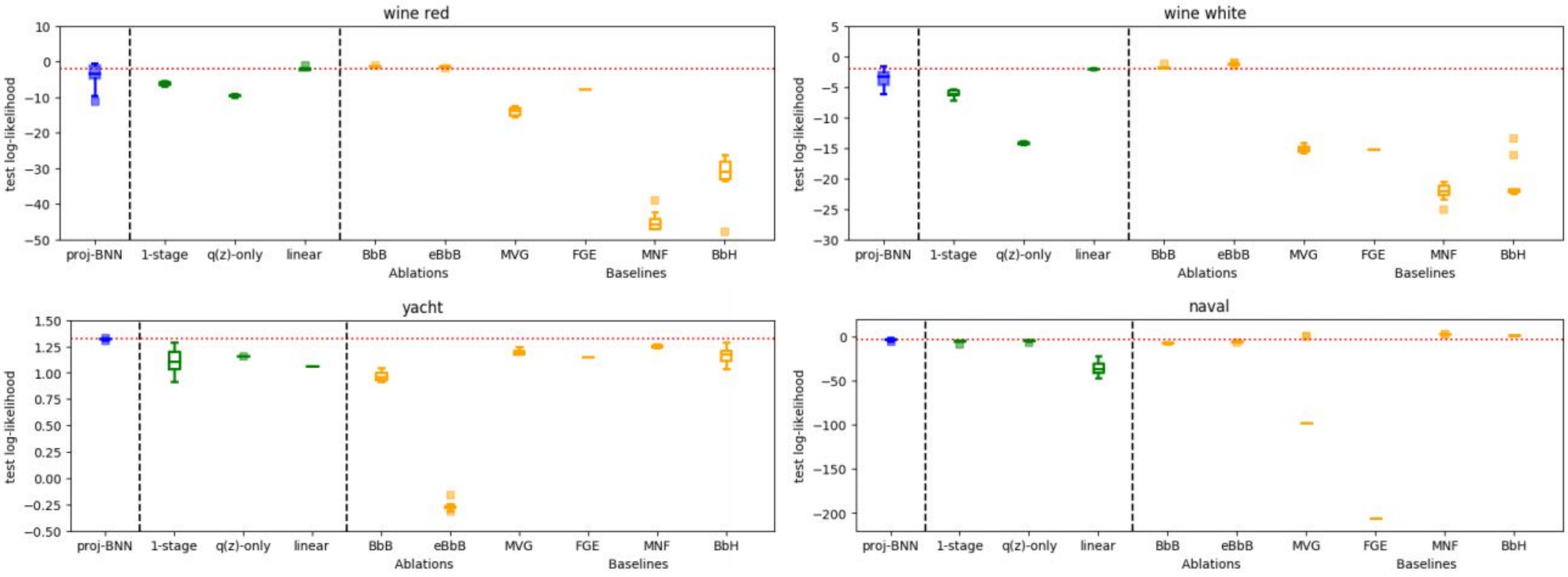


(a) Proj-BNN ($D_z = 2$) (b) BbB (c) MNF (d) MVG

- BbB: Bayes by Back Prop [Blundell et.al, 2015]
- MVG: Multivariate Gaussians [Louizos et.al, 2016]
- MNF: Multiplicative Normalizing Flow [Louizos et. al, 2017]

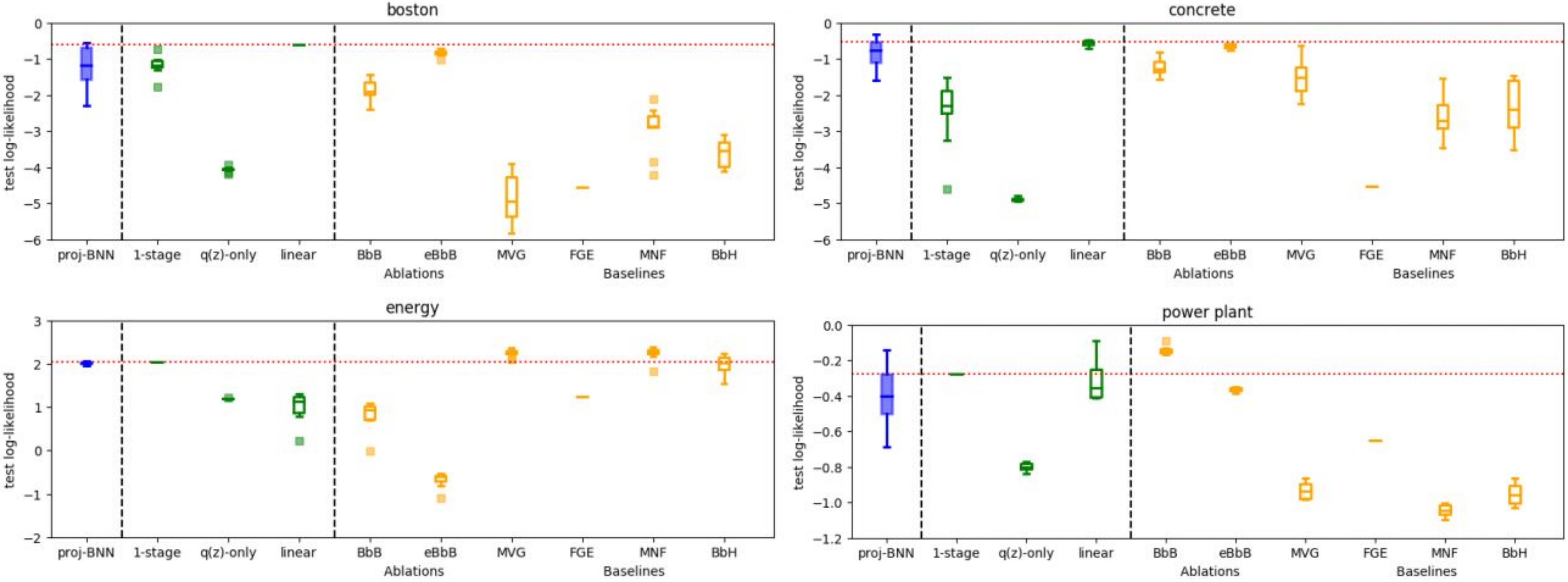
Results: Generalization

https://arxiv.org/abs/1811.07006



Results: Generalization

https://arxiv.org/abs/1811.07006



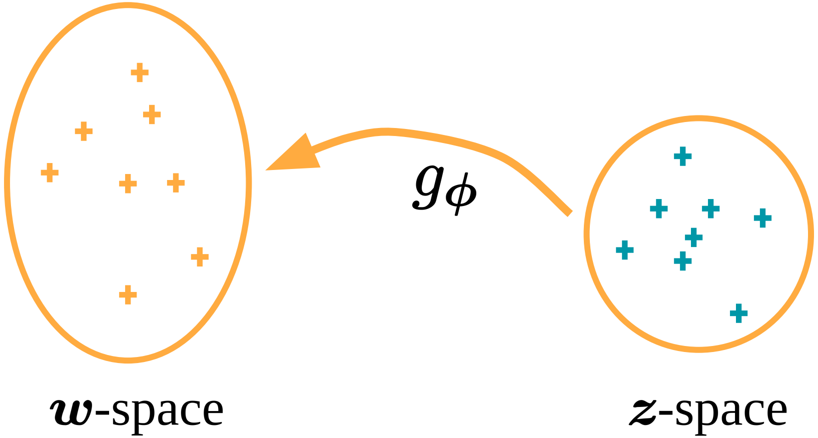
Conclusions

Contact: *melanie@seas.harvard.edu*

https://melaniefp.github.io/

In this talk...

- Alternative modeling for BNNs
- Better approximate inference



https://arxiv.org/abs/1811.07006

Thank you for listening!

Open questions

- Better evaluation of uncertainty?

``Test log likelihood can be misleading”

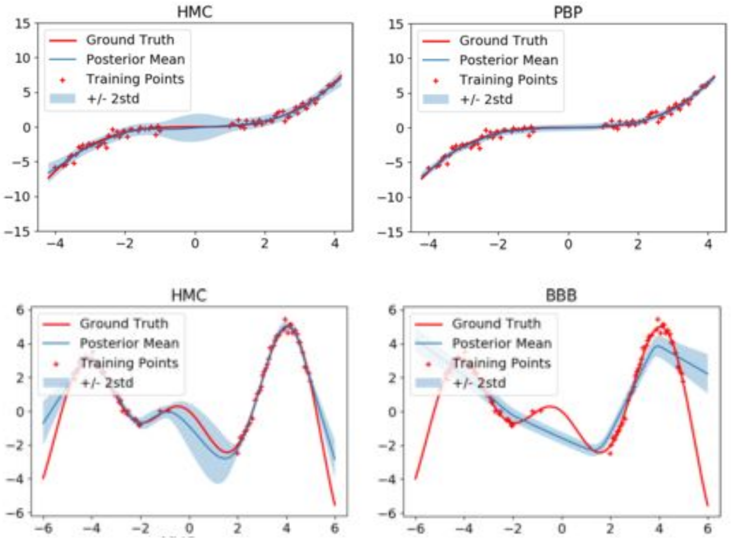
> Entangled sources of error: model, variational approx, optimization

- How does the topology in weight space looks like?

> Intuition misleading in high dimensions!

- How to exploit latent structure for interpretability?

[Yao et. al, ICML Workshop, 2019]



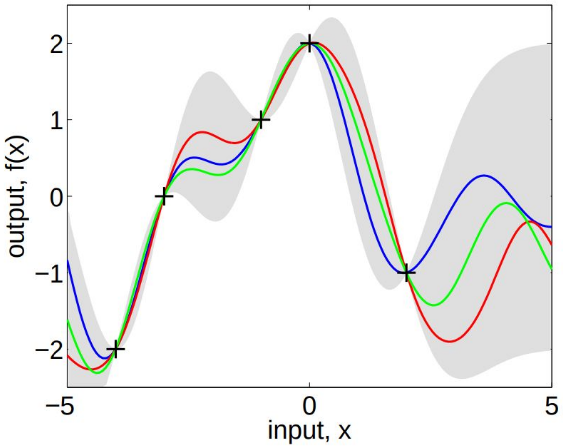
Related works on weight embeddings [Karaletsos et.al, 2018; Izmailov et.al, 2019]

Appendix

Uncertainty Estimation via GPs?

Gaussian Process (GP)

[Rasmussen et.al, 2016]



$$f(x) \sim \mathrm{GP}\left(m(x), k(x, x')\right)$$

Drawbacks of GPs

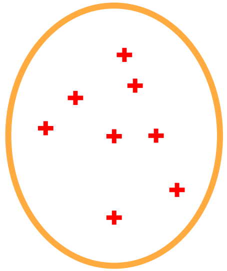
- Scalability
- Kernel learning is not trivial

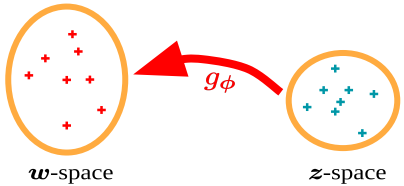
Alternative: Neural Networks with uncertainty

- Ensemble of Neural Networks
  [Lakshminarayanan et al., 2017; Pearce et.al, 2018]

- Bayesian Neural Networks
  [Buntine et al., 1991; MacKay, 1992; Neal, 1993]

Results: Generalization (Ablations)



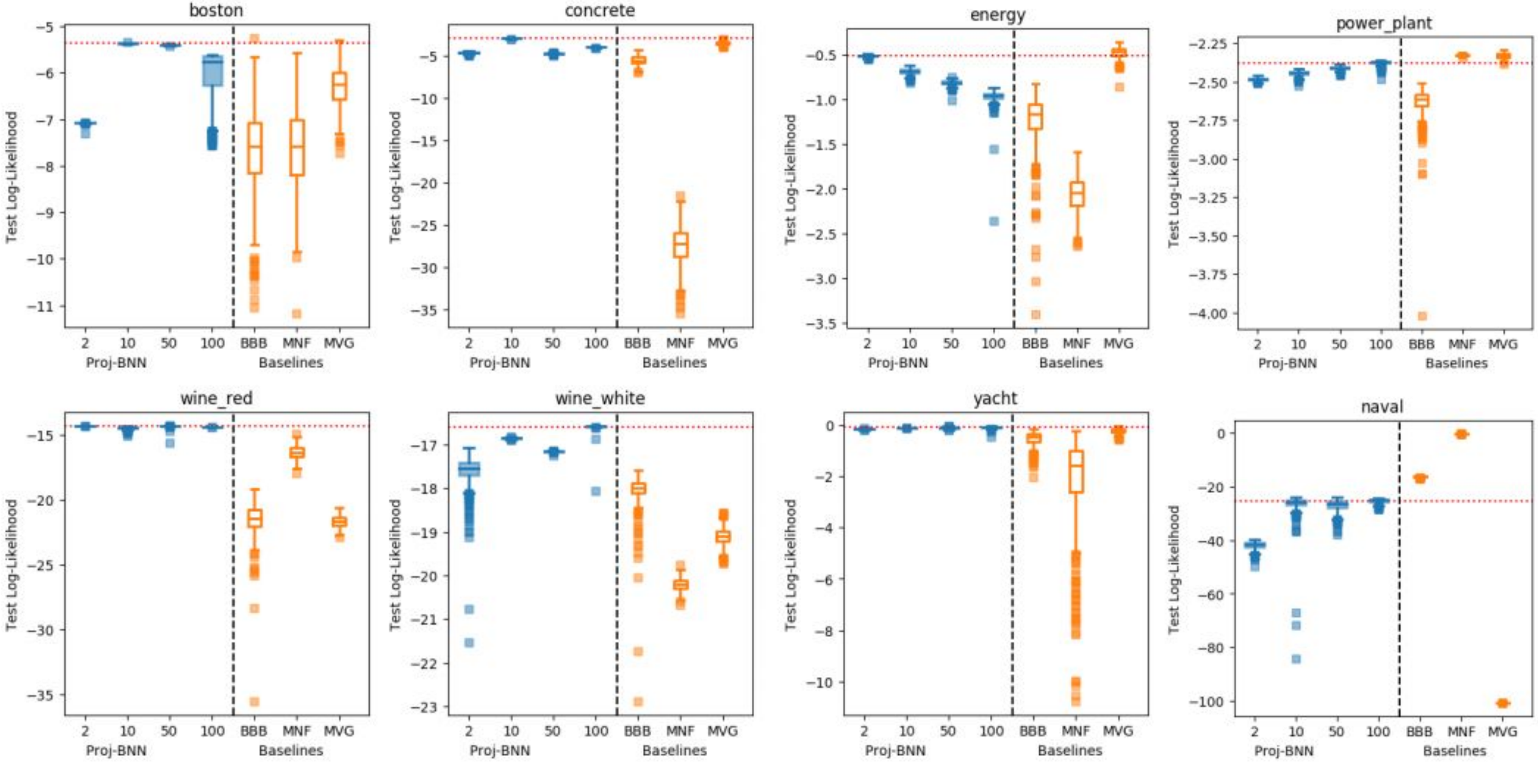


$$D_{\mathrm{KL}}\Big(q_{\boldsymbol{\lambda}}(\boldsymbol{z},\boldsymbol{\phi})\,||\,p(\boldsymbol{z},\boldsymbol{\phi}|\mathcal{D})\Big)$$

	1. Characterize w-space	2. Find point estimate g_ϕ	3. Black-box VI (BBVI)
1-stage	✗	✗	✓
linear	✓	linear	✓
q(z) only	✓	✓	$q_{\boldsymbol{\lambda}_z}(\boldsymbol{z})$

Cross-validation of latent dimension

https://arxiv.org/abs/1811.07006



Prediction-constrained Autoencoder

$$\{\boldsymbol{\theta}^\star, \boldsymbol{\phi}^\star\} = \operatorname*{argmin}_{\boldsymbol{\theta},\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \min_{\boldsymbol{\theta},\boldsymbol{\phi}} \left\{ \frac{1}{R} \sum_{r=1}^{R} \left(\mathbf{w_c}^{(r)} - g_{\boldsymbol{\phi}} \left(f_{\boldsymbol{\theta}} \left(\mathbf{w_c}^{(r)} \right) \right) + \gamma^{(r)} \right)^2 \right.$$
$$\left. + \, \beta \, \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\frac{1}{R} \sum_{r=1}^{R} \log p \big(y | x, g_{\boldsymbol{\phi}} \left(f_{\boldsymbol{\theta}} \left(\mathbf{w_c}^{(r)} \right) \right) \big) \right] \right\},$$

My research: probabilistic models for societal needs

Highly driven by real-world application, with special emphasis on…

A) Latent Representation Learning

M. F. Pradier, B. Reis, L. Jukofsky, F. Milletti, T. Ohtomo, F. Perez-Cruz, and O. Puig. **Case-control Indian Buffet Process identifies biomarkers of response to Codrituzumab**. *BMC Cancer*. 2019.

I. Valera, M. F. Pradier, M. Lomeli, and Z. Ghahramani. **General Latent Feature Models for Heterogeneous Datasets**. *In submission to Journal of Machine Learning Research*. 2018.

M. F. Pradier, W. Pan, M. Yau, R. Singh, and F. Doshi-Velez. **Hierarchical Stick-breaking Paintbox**. *BNP@NeurIPS Workshop*. Montreal (Canada), December 2018.

B) Uncertainty Quantification

M. F. Pradier, W. Pan, J. Yao, S. Ghosh, and F. Doshi-Velez. **Projected BNNs: Avoiding Pathologies in Weight Space by projecting Neural Network Weights**. Arxiv. 2019.

B. Coker, M. F. Pradier, and F. Doshi-Velez. **Poisson Process Radial Basis Function Networks.** (Arxiv coming soon)

W. Yang, L. Lorch, M. A. Graule, S. Srinivasan, A. Suresh, J. Yao, M. F. Pradier, and F. Doshi-Velez. **Output-Constrained Bayesian Neural Networks**. *ICML Workshop on Generalization*. 2019.