MLDL

Machine Learning and Deep Learning Conference 2020

SAND2020-7790PE

Estimating Predictive Uncertainty in Scientific Machine Learning

A Library of Methods and Test Problems

8/5/2020

Ahmad A. Rushdi (1463), *Optimization and Uncertainty Quantification*

ASC-AML: Aubrey C. Eckert (1544), Gabriel Huerta (9136)

Dakota: Laura P. Swiler (1463), Brian M. Adams (1463)

Funding Sources: ASC AML, ASC NextGen, Dakota

Table of Contents

1. Problem Statement
2. Predictive Uncertainty
3. Approach
4. Numerical Experiments
5. Conclusions

Problem Statement

Abstract. Neural network models have attracted a lot of research attention in Scientific Machine Learning (SciML) problems. However, they tend to be overconfident when reporting typical point-estimate predictions in classification and regression problems. This could be very harmful when dealing with costly numerical simulations or high-stakes decisions in national security applications. In this work, we assess uncertainty quantification techniques for neural network models. To understand their variability, we rely on different sources of randomness associated with training samples, weight initialization, dropout methods, and ensemble formations. Motivated by typical SciML situations, we assume a limited sample budget, noisy training data, and suggest approaches for reporting and possibly reducing uncertainty.

Supervised Machine Learning: extract *models* from *data* and use them to make predictions.

For example, given data: $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\} \subset \mathcal{X} \times \mathcal{Y}$

minimize avg squared error of a linear model: $\frac{1}{N}\sum_i \|y_i - (\mathbf{x}_i\mathbf{W} + \mathbf{b})\|$

For costly modeling and simulation domains: Machine and deep learning predictions need to be accompanied by an accurate quantification of their uncertainty – in order to efficiently use the sample budget and build confidence in the model predictions.

Problem Statement

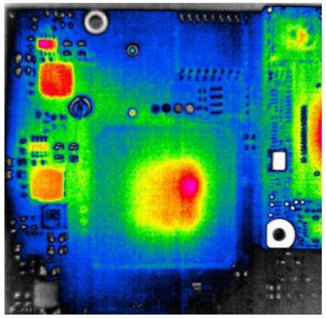
Deep learning “black box” models are popular, yet are difficult to interpret and understand.

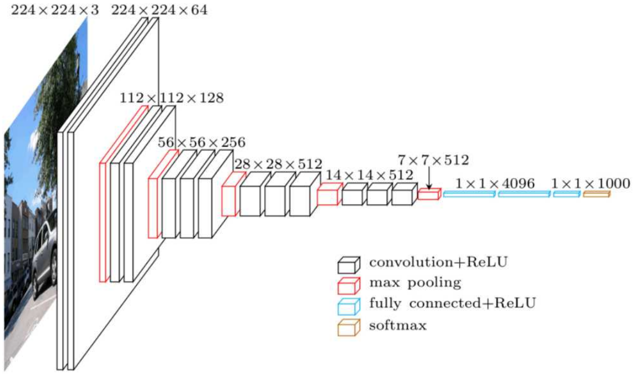
UQ in model-based critical decision making

UQ today underpins many decision processes in nuclear security, our risk management and associated investments, which can be at the scale of billions of dollars. Predictions without UQ are neither predictions nor actionable. The data-rich world of ML, especially the powerful deep learning (DL) models, poses parallel challenges.

To develop consequential decision support from ‘learned’ models built on complex datasets, there is an important need to co-develop UQ for this domain.

Feature engineering in deep learning: rethinking architectures

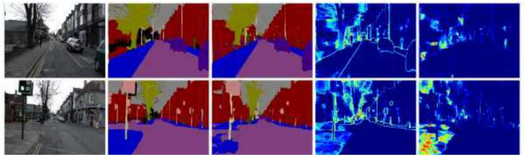




Variance as a measure of lack of confidence

The **variance can be seen as a measure of uncertainty** — but where is the neural network uncertain or what is it uncertain about?

CNN illustration: *Make3D Depth regression* (input, truth, depth prediction, aleatoric, epistemic)



- Begoli E, Tanmoy B, and Dimitri K "The need for uncertainty quantification in machine-assisted medical decision making." Nature Machine Intelligence, 2019
- Mehta, Pankaj, et al. "A high-bias, low-variance introduction to machine learning for physicists." Physics reports 810 (2019): 1-124

Predictive Uncertainty

Why do we need a measure of uncertainty or low-confidence?

Very limited, typically expensive, training data

Attempting to extrapolate far away from observed data

Value on incorporating UQ in ML

- Support and augment decision making processes, e.g., Cyber, MRI analysis, etc.
- Reinforcement learning, involves time and resources we wish not to waste, e.g., elf-learning agents (autonomous robots/vehicles)
- Model explainability using sensitivity analysis measures – check out: *Assessing the Accuracy of ML Explanations for Model Credibility* (Mike R. Smith, day 2, session 1)

Human expert time is expensive. Many experiments are also "very" expensive!

Active Learning (human annotator)

- The model chooses which unlabeled data are most informative

Adaptive Sampling (design of experiment and sample design)

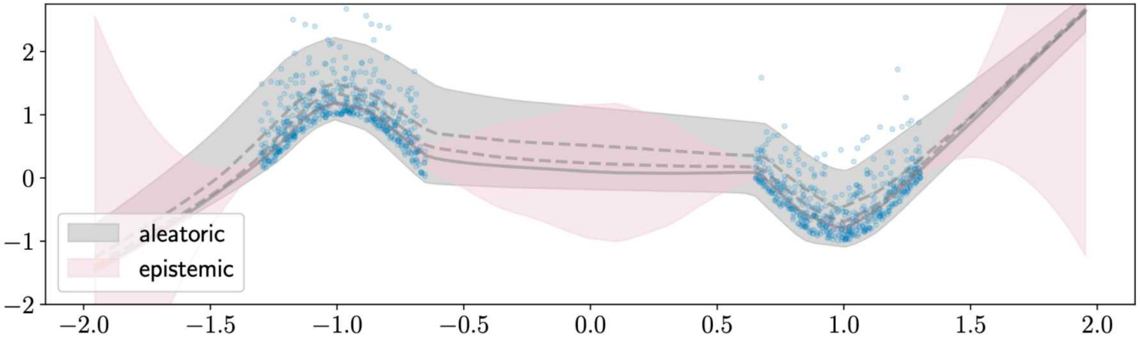
- An uncertainty estimator (e.g., variance) suggests points of highest value for model accuracy improvement

Predictive Uncertainty

1. **Aleatoric Uncertainty**
   - Input data corruption: noise levels, measurement errors, etc.
   - Can *not* be reduced by the model designer
   - Can be reduced by increasing measurement precision
2. **Epistemic Uncertainty**
   - Fidelity of the model (parameters and/or structure) when representing data
   - Decreases as the training data size increases

95% prediction interval,
Simultaneous Quantile Regression

Orthonormal Certificates



Quantifying predictive uncertainty, methods need to consider both *aleatoric* and *epistemic* uncertainties.

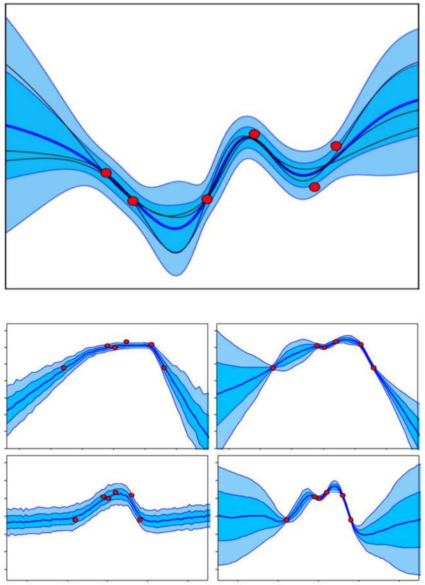
- Hüllermeier E, and Willem W, "Aleatoric and epistemic uncertainty in machine learning: A tutorial introduction." *arXiv preprint* (2019).
- Gal, Yarin. "Uncertainty in deep learning." University of Cambridge 1.3 (2016).

Approach

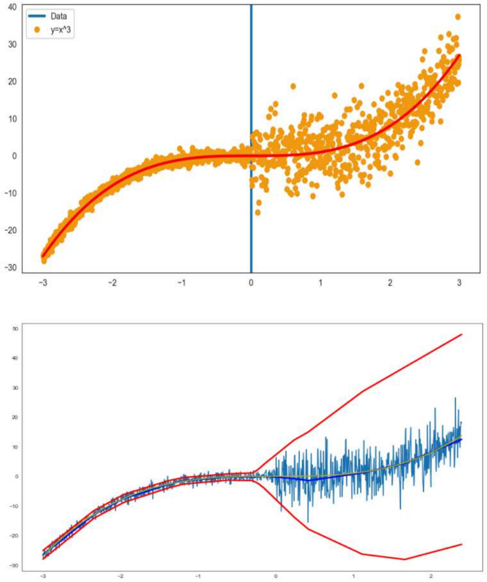
We represent DL predictions as **distributions** (μ, σ^2) rather than softmax-based point estimates $\hat{p}$.

Scientific Machine Learning

1. Training sample **budget is limited**, causing Out of Distribution (OoD) issues.
2. Different **types of noise**, imposed onto the input data and propagate that noise into an uncertainty metric in the resulting prediction.



Very few training samples

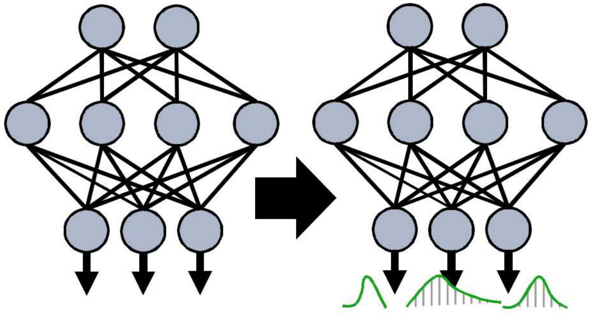


Noisy training data

Quality of prediction measures: Our approach is to assign a numerical score to a prediction $p_\theta(y|\mathbf{x})$, rewarding *better calibrated predictions* over worse. We leverage randomness in Neural Networks' initialization, weight learning, and backpropagation steps.

Approach – *Methods*

Bayesian NNs



Bayesian Neural Networks (BNNs): place a prior distribution over the network weights and use data to learn a posterior distribution.

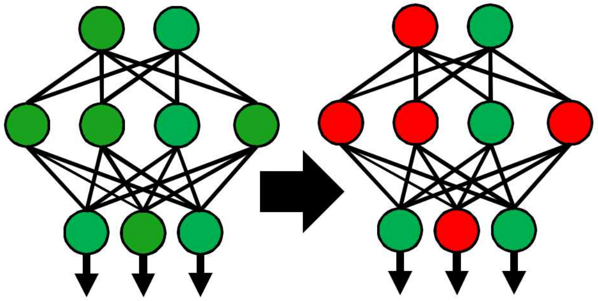
$$\begin{aligned}\mathbf{w}^{\mathrm{MLE}} &= \arg\max_{w} \log p(\mathcal{D}|\mathbf{w}) \\ &= \arg\max_{w} \textstyle\sum_i \log p(\boldsymbol{Y}_i|\boldsymbol{X}_i, \mathbf{w}) \\ \mathbf{w}^{\mathrm{MAP}} &= \arg\max_{w} \log p(\mathbf{w}|\mathcal{D}) \\ &= \arg\max_{w} \log p(\mathcal{D}|\mathbf{w}) + \log p(\mathbf{w})\end{aligned}$$

Variational Inference (VI) approximation:

$$\theta^* = \arg\min KL\big(q_\theta(\mathbf{w}|\boldsymbol{D})||p(\mathbf{w}|\boldsymbol{D})\big)$$

Graves, Alex. "Practical variational inference for neural networks." *Advances in neural information processing systems*. 2011.

MC Dropout



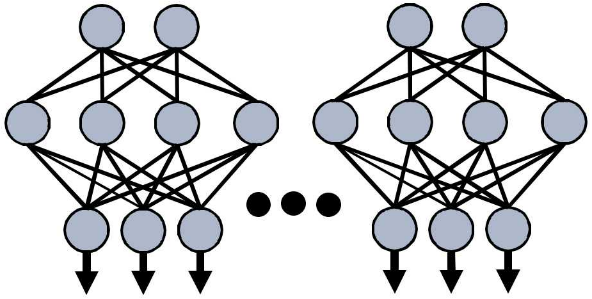
Neurons are randomly dropped in each iteration, with some probability (*p*), often fixed at empirical values (e.g., 0.3).

$$\mathbb{E}_{q(y^*|x^*)}(y^*) \approx \frac{1}{T}\sum_{t=1}^{T} \hat{y}^*(x^*, W_1^t, \cdots, W_L^t)$$

Average of *T* stochastic forward passes through the network.

Hinton, Geoffrey E., et al. "Improving neural networks by preventing co-adaptation of feature detectors." *arXiv preprint arXiv:1207.0580* (2012).

Deep Ensembles



Estimate 2 outputs: mean and variance

Modified loss function + adversarial training

$$-\log p_\theta(y_n|\mathbf{x}_n) = \frac{\log \sigma_\theta^2(\mathbf{x})}{2} + \frac{(y-\mu_\theta(\mathbf{x}))^2}{2\sigma_\theta^2(\mathbf{x})}$$

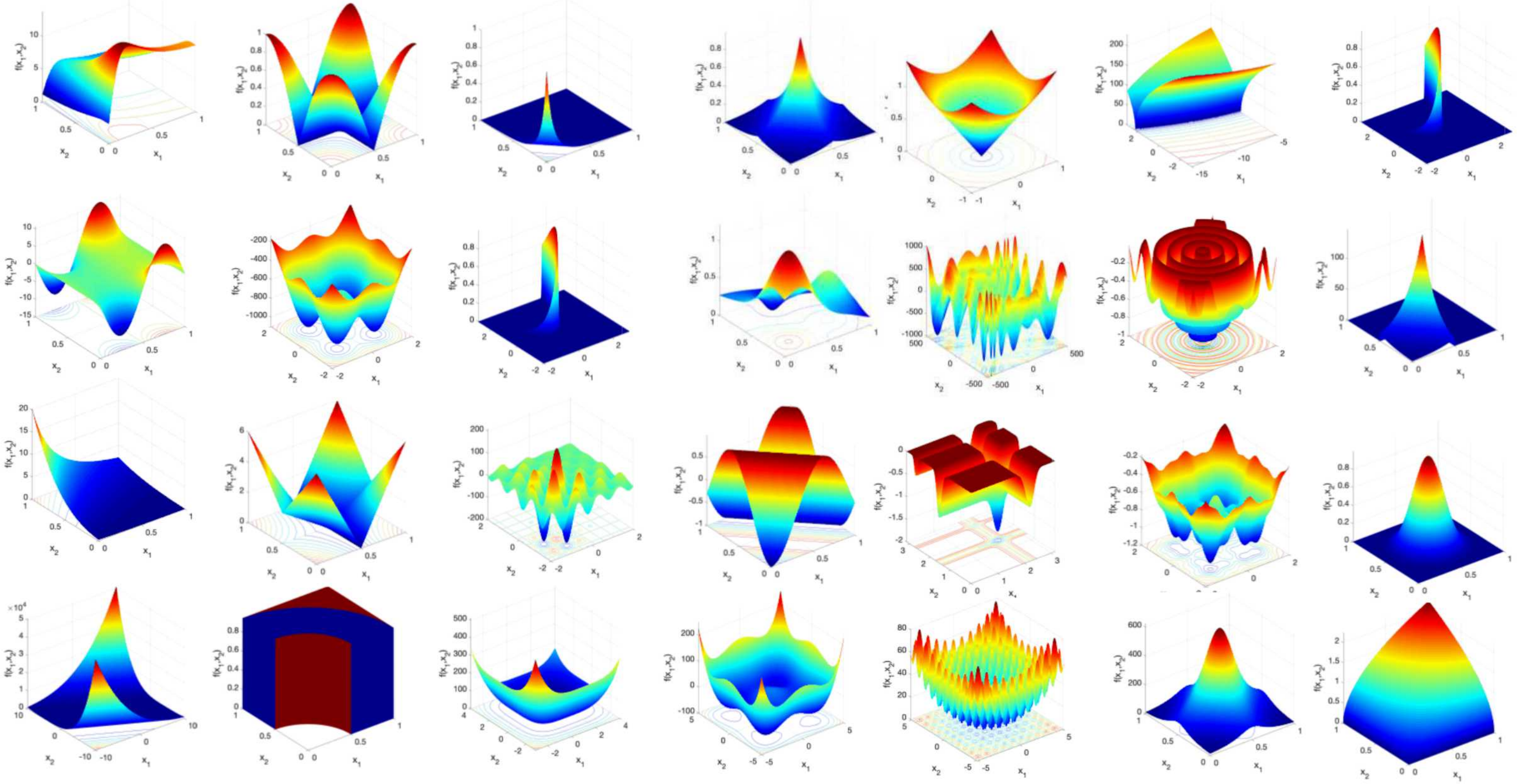
Decomposed uncertainty

$$\mu_e = \frac{1}{M}\sum_{i=1}^{M}\mu_i \text{ and } \sigma_e^2 = \underbrace{\frac{1}{M}\sum_{i=1}^{M}\sigma_i^2}_{\textit{Aleatoric}} + \underbrace{\left[\frac{1}{M}\sum_{i=1}^{M}\mu_i^2 - \mu_e^2\right]}_{\textit{Epistemic}}$$

Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell. "Simple and scalable predictive uncertainty estimation using deep ensembles." Advances in neural information processing systems. 2017.

Approach – *Data*

A library of **high-dimensional analytical test functions**, frequently used for QoI experimental studies (optimization, numerical integration, uncertainty quantification, and multi-fidelity analysis).



- Dakota www.Dakota.sandia.gov. Adams, B.M., Bohnhoff, W.J., Dalbey, K.R., Ebeida, M.S., Eddy, J.P., Eldred, M.S., Geraci, G., Hooper, R.W., Hough, P.D., Hu, K.T., Jakeman, J.D., Khalil, M., Maupin, K.A., Monschke, J.A., Ridgway, E.M., Rushdi, A.A., Stephens, J.A., Swiler, L.P., Vigil, D.M., Wildey, T.M., and Winokur, J.G., "Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.11 User's Manual," Sandia Technical Report SAND2014-4633, July 2014; updated November 2019.
- Virtual Library of Simulation Experiments https://www.sfu.ca/~ssurjano/index.html

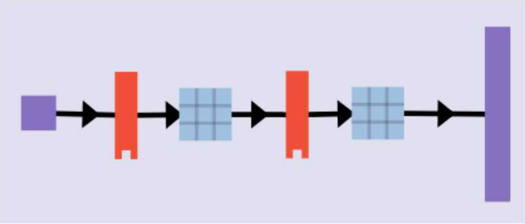
Numerical Experiments

Data:

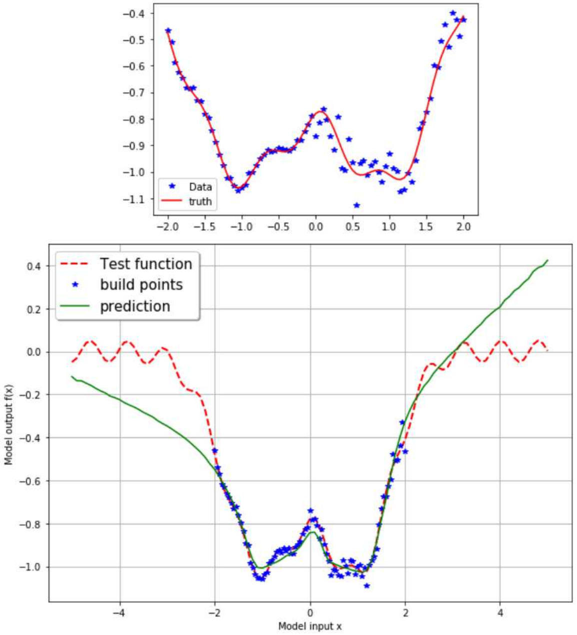
function samples + $\begin{cases} AWGN\,(0,\sigma_1) \\ AWGN\,(0,\sigma_2) \end{cases}$

Model:

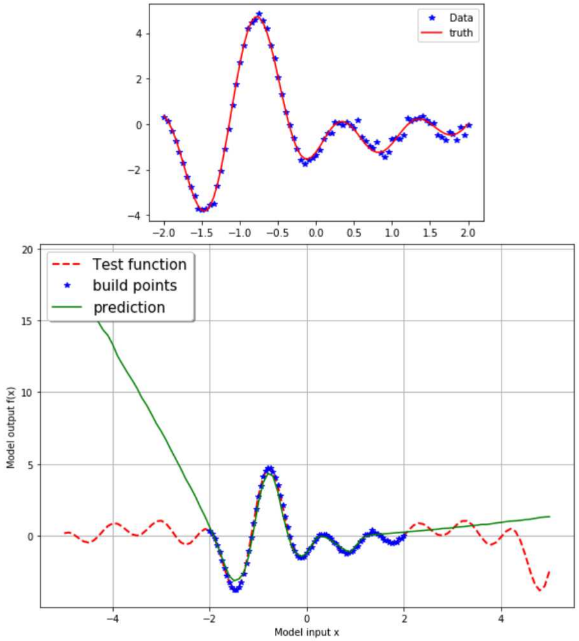
```
input = Input(shape=(1,))
x = Dense(512, activation="relu")(input)
x = Dropout(0.5)(x, training=True)
x = Dense(512, activation="relu")(x)
x = Dropout(0.5)(x, training=True)
output = Dense(1)(x)
                    Total params: 264,193
```



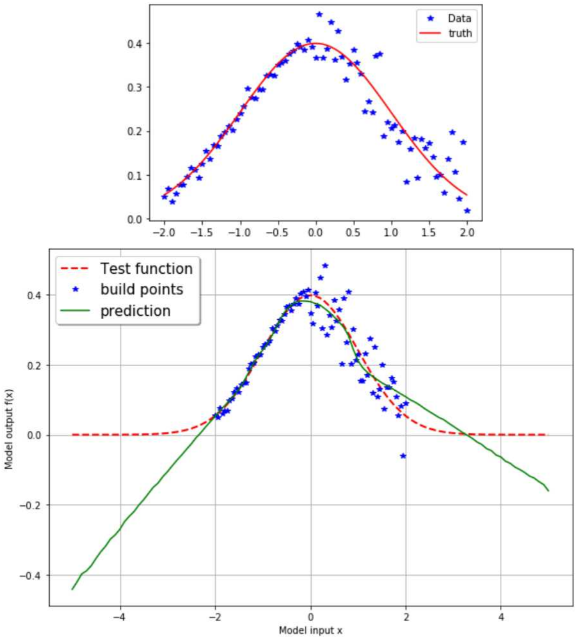
Herbie



Shubert



Gaussian



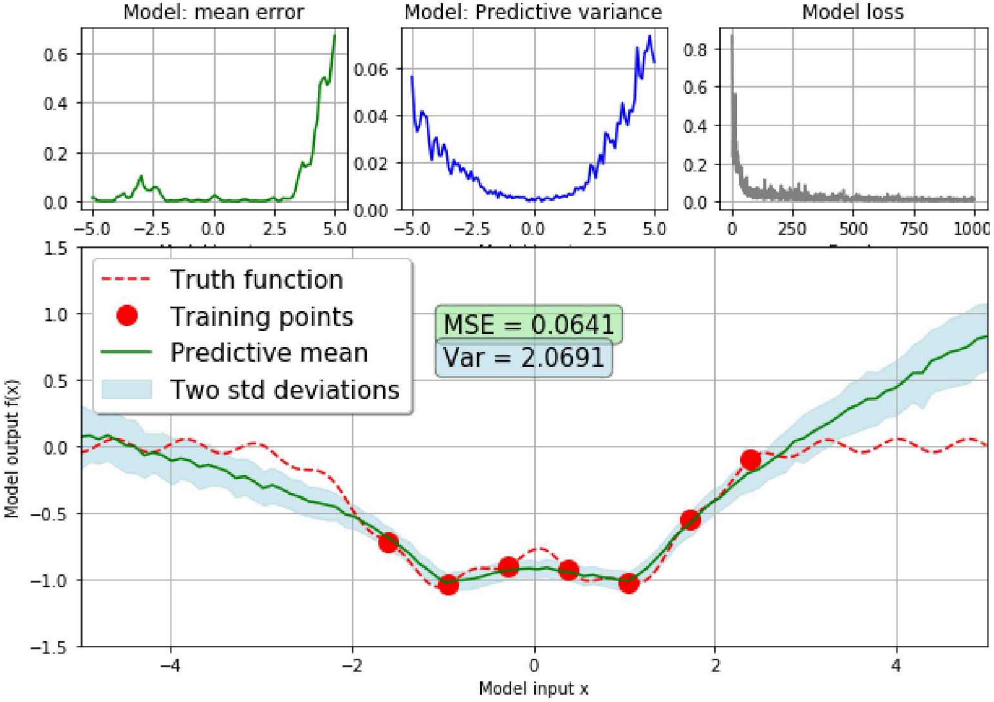
A model with 200K trainable parameters still needs an assessment of confidence/uncertainty.

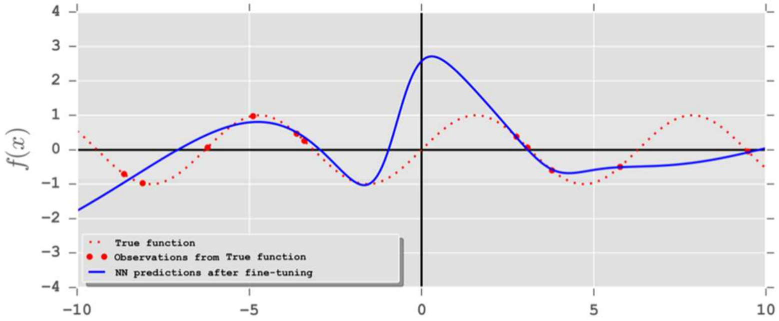
Numerical Experiments

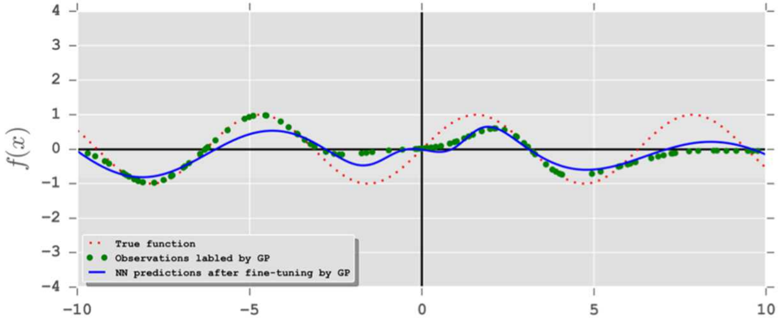
Uncertainty quantification at a new **test point**.

Overall model uncertainty/confidence within a **test domain**.

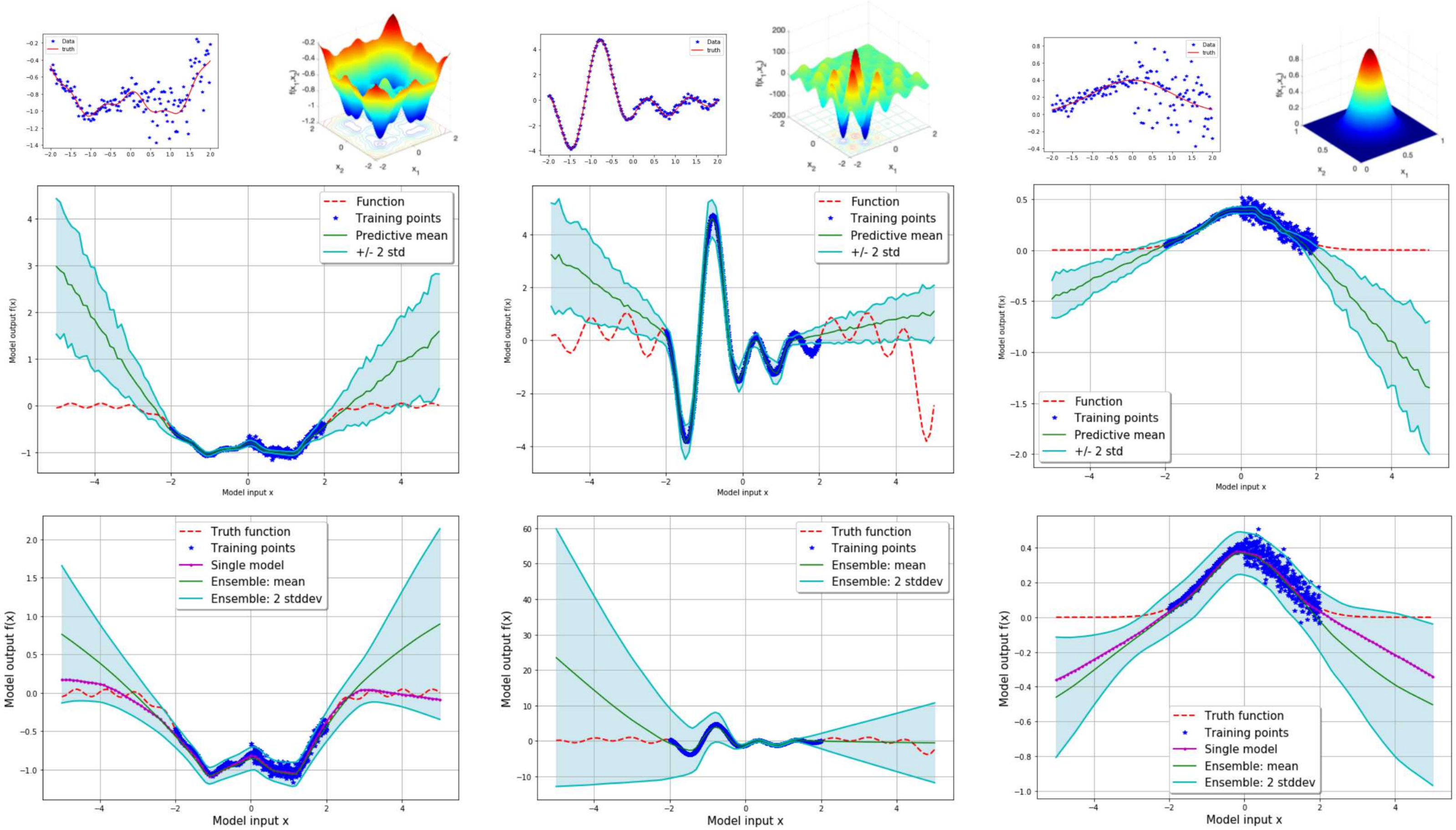
Guidance of **adaptive sampling** towards points/regions of high estimated variance.







Numerical Experiments



Analytical Functions

Dropout

Deep Ensembles

Methods vary in terms of complexity, accuracy, scalability, and computational cost.

Observations & Conclusions

- **Ongoing work.** No one solution fits all. Classes of functions vary in terms of smoothness, oscillation, discontinuities, etc.
- Methods vary in terms of complexity, accuracy, scalability, and computational cost.
- We use analytical test functions for comparisons, but in real-world problems, no "ground truth" uncertainty estimates are available.

Preliminary Results:

- Deep ensembles
  - Smoothed out by adversarial training
  - Most conservative
  - Low training cost (e.g., with M=5 models)
- Ongoing comparisons to:
  - [Madras 2020] Detecting Extrapolation with Local Ensembles
  - [Ashukha 2020] Pitfalls of In-Domain Uncertainty Estimation and Ensembling in Deep Learning

Thank you!