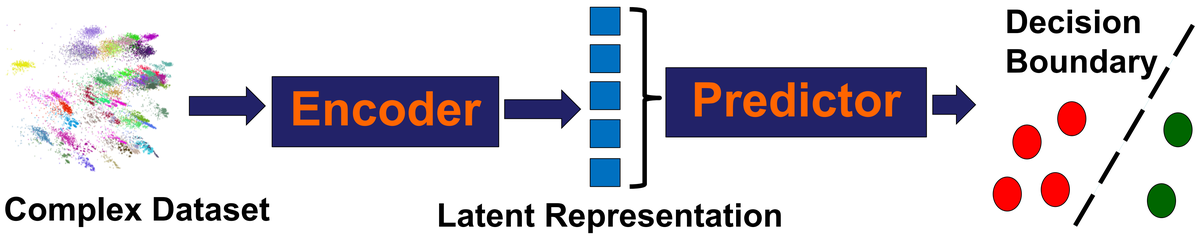
Deep Kernels with Probabilistic Embeddings for Small-Data Learning

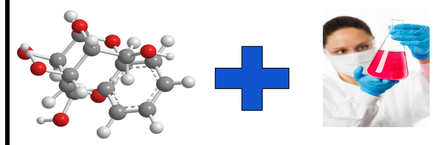
Ankur Mallick, Chaitanya Dwivedi, Bhavya Kailkhura, Gauri Joshi, T. Yong-Jin Han

Carnegie Mellon University

1. Small-Data Learning: Motivation & Requirements



Motivation:

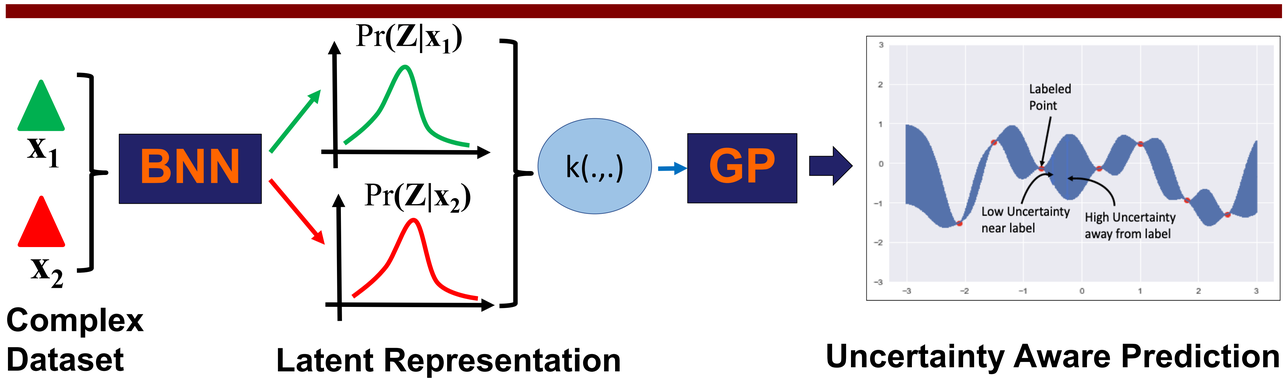


- **Scientific data is expensive to collect and label**
- **Small datasets typically lead to poor model performance (inaccurate/overconfident)**

Requirements:

- **Sample Efficiency:** Accurate predictions on test data even with small amounts of training data
- **Uncertainty Quantification:** Signal of low model confidence in underrepresented regions
- **Representation Learning:** Capturing patterns in complex/high-dimensional data

2. Our Solution : Deep Probabilistic Kernel Learning



- **Bayesian Neural Network (BNN)** maps data points to probability distributions
- **Gaussian Process (GP)** uses a kernel between probability distributions

$$K_{ij} = \mathbb{E}_{\mathbf{z}\sim Pr(\mathbf{Z}|\mathbf{x}_i),\ \mathbf{z}'\sim Pr(\mathbf{Z}|\mathbf{x}_i)}[\frac{1}{2}\exp(-||\mathbf{z}-\mathbf{z}'||^2)]$$

- BNN is trained by minimizing the GP negative log likelihood

$$L[p] = -\log Pr(\mathbf{y}|\mathbf{X}, p(\mathbf{W})) = \frac{1}{2}\mathbf{y}^T(\mathbf{K}[p]+\sigma^2\mathbf{I})^{-1}\mathbf{y} + \frac{1}{2}\log(\det(\mathbf{K}[p]+\sigma^2\mathbf{I}))$$

3. Training Algorithm: Functional GD

1.Initialization:

$$\boldsymbol{w}_1^{(0)}, \ldots, \boldsymbol{w}_m^{(0)} \sim N(\boldsymbol{0}, \boldsymbol{I}) \qquad [\boldsymbol{w}_1^{(0)}, \ldots, \boldsymbol{w}_m^{(0)} \sim p^{(0)}(\boldsymbol{w})]$$

2. Update (iterative):

Weighted average of gradients

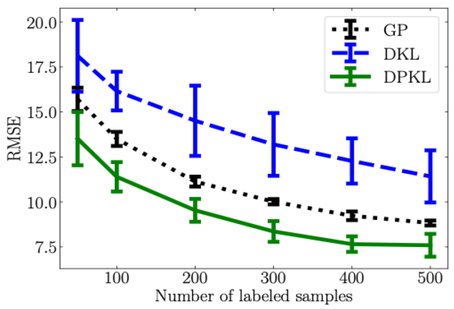
$$\boldsymbol{w}_j^{(t+1)} = \boldsymbol{w}_j^{(t)} - \varepsilon \sum_i \kappa(\boldsymbol{w}_j^{(t)}, \boldsymbol{w}_i^{(t)})\, \nabla L(\boldsymbol{w}_i^{(t)}) \quad [\boldsymbol{w}_1^{(t)}, \ldots, \boldsymbol{w}_m^{(t)} \sim p^{(t)}(\boldsymbol{w})]$$

Weights (ensure that the samples don't diverge)

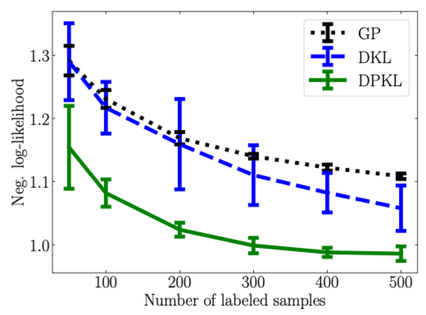
3.Convergence:

$$\boldsymbol{w}_1^*, \ldots, \boldsymbol{w}_m^* \sim p^*(\boldsymbol{w}) = \operatorname{argmin} L[p]$$

4. Results (UCI CTSlice Dataset)



Accuracy



Uncertainty Quantification

- Results for CTSlice (384 dimensional) Dataset
- Number of training samples : {50,100,200,300,400,500}
- Baselines : Vanilla GP, Deep Kernel Learning (DKL)

5. Take Home Message

- Small data learning requires models that can provide uncertainty aware prediction and representation learning using only a few samples
- DPKL achieves this goal by combining BNNs and GPs in a principled fashion and training the model end-to-end using Functional Gradient Descent
- A rigorous understanding of the sample complexity of BNNs is needed

DPKL (our method) gives lower RMSE (better accuracy) and lower negative log-likelihood (better Uncertainty Quantification) for all sample sizes

ICLR 2021 S2D-OLAD Workshop