andras **szep** & luke **pelegrin**

RELIABLE ML IN A WORLD OF UNCERTAINTY

created with the help of our mentors
lalitha **venkataramanan**
ravinath **viswanathan**
carine **simon**

Data

We learned on hard synthetic data with ↑ heteroskedasticity, ↓ signal to noise ratio, and ↑ non-linearity.

We applied our solution to 4 additional real-life data-sets:

- Mercury injection capillary pressure
- Concrete dataset
- Crimes dataset
- MPG dataset
- Synthetic data

Problem Statement

We aimed to answer the following questions:
- Can we make the ML algorithms **reliable** by also **predicting uncertainty** in predictions?
- Can we also **explain** the uncertainty predictions?

Machine learning techniques have been shown to provide uncertainties together with the point estimates for classification and regression problems. However, enabling these uncertainties to be correlated to prediction error is still considered challenging.

We introduce a new custom loss function, which explicitly provides the aleatoric and epistemic uncertainties and correlates them to the error.

Significance

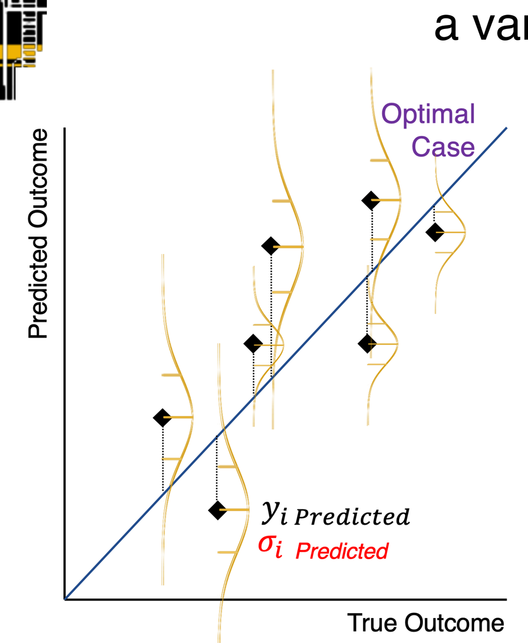
A lack of a good uncertainty estimate in the output of Deep Learning models has led to cases such as Google's classification of a black couple as gorillas or Uber's fatal self-driving accident.

For Schlumberger, calibrated uncertainty predictions would mean that it could provide more reliable information to its clients about the subsurface. But **better uncertainty estimation would impact industries** from finance to healthcare as **it would enable us to trust ML** predictions.

Thus, even though our Capstone project has ended, we are working **to publish our results** so that the world can benefit from our work.

Modeling Work

We derived a **custom loss function** that provides an accurate prediction of a variable of interest as well as an uncertainty in the prediction simultaneously.



Traditional Approach:

$$\min \sum_{i}^{N} (y_{i\,Predicted} - y_{i\,True})^2$$

Our Solution:

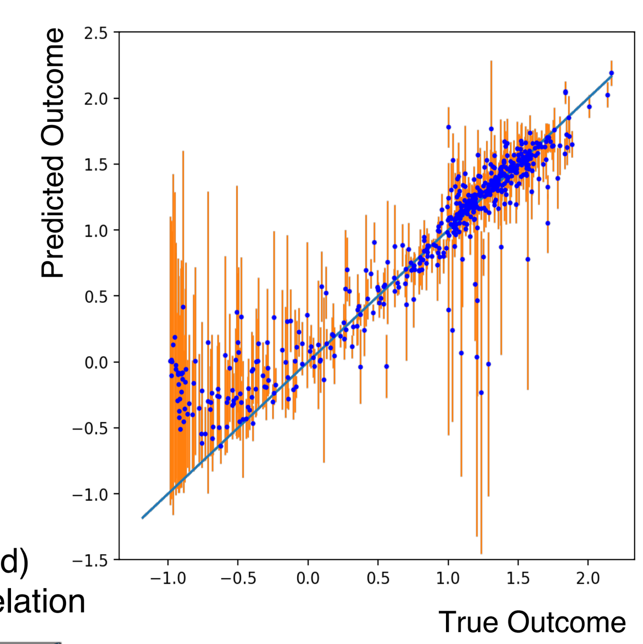
$$\min \sum_{i}^{N} \underbrace{\log(\sigma_i) + \left(\frac{y_{i\,Predicted} - y_{i\,True}}{\sigma_i}\right)^2}_{\text{Negative log-likelihood}} + \alpha * \underbrace{(\sigma_i - |y_{i\,Predicted} - y_{i\,True}|)^2}_{\text{Regularization term}}$$

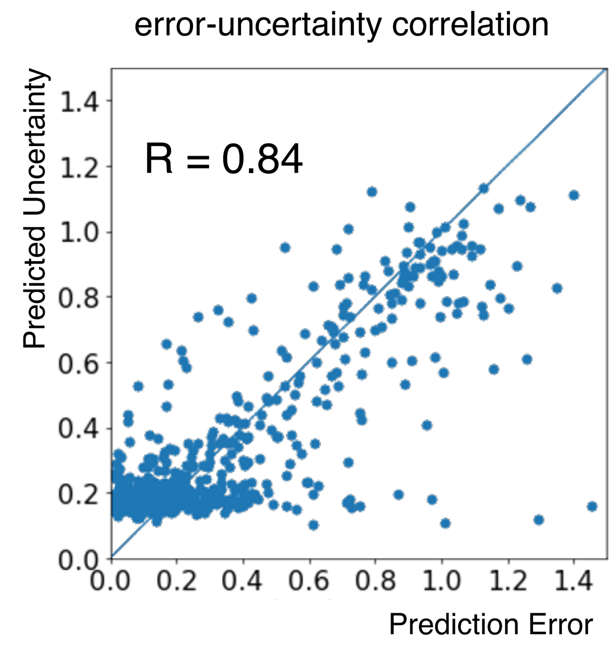
We incorporated our custom loss function by constructing **neural networks** with two output neurons for the predicted target variable y_i and its uncertainty σ_i.

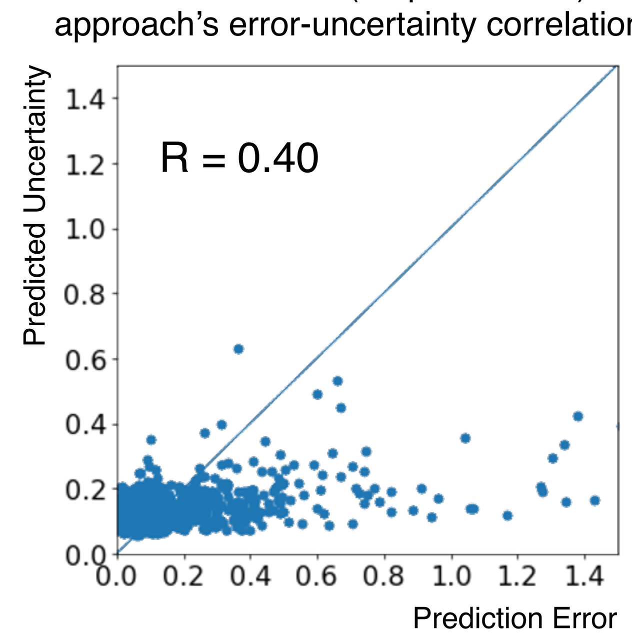
To obtain the value of information for each feature, we implemented **LIME** and **SHAP** to explain our neural networks' y_i and σ_i predictions.

Results

Test set results on our synthetic data show that our custom loss function produces **accurate predictions** and gives **uncertainty estimates that are correlated with the true prediction error.**







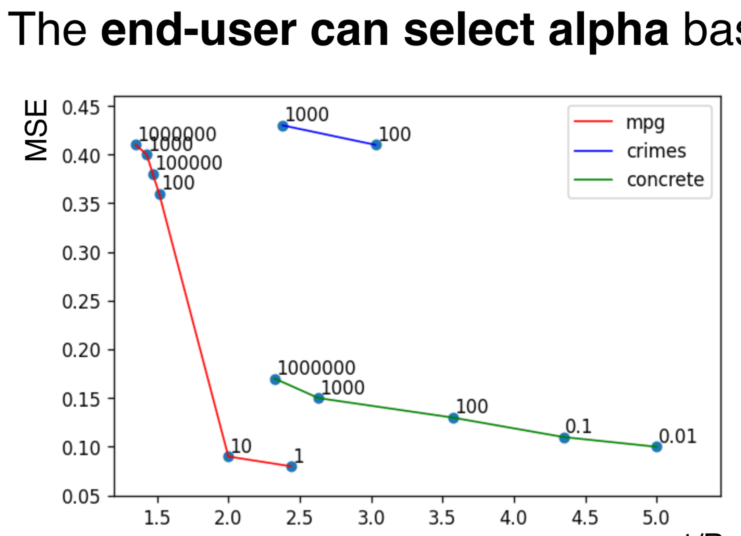
In fact, we significantly **outperform the state-of-the-art approach of uncertainty estimation**.

◀ test-set results

On all five tested datasets, our domain-independent solution produces a **better-calibrated uncertainty estimate** while achieving a similar prediction accuracy:

		Synthetic	MPG	Concrete	Crimes	SLB (MICP)
MSE	Our model	0.16	0.15	**0.14**	0.42	**0.57**
	Dropout	**0.15**	**0.09**	0.17	**0.31**	0.63
R	Our model	**0.84**	**0.50**	**0.50**	**0.51**	**0.55**
	Dropout	0.40	0.37	0.34	0.43	0.39

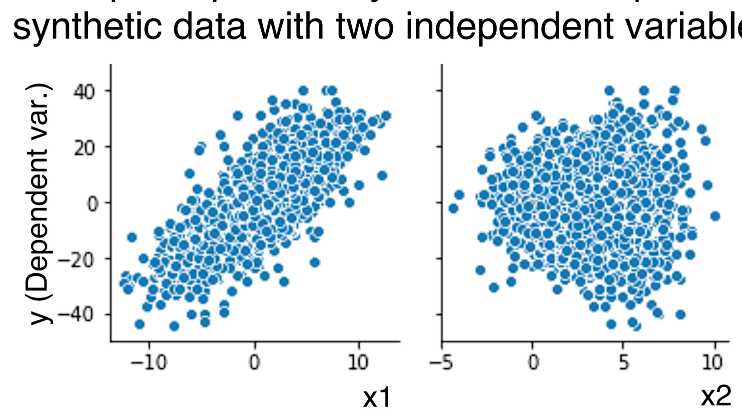
The **end-user can select alpha** based on MSE and uncertainty-quantification needs:



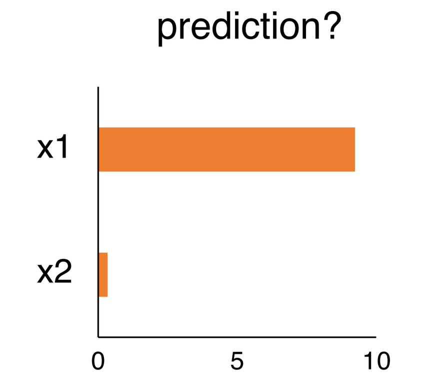
◀ The trade-off between accuracy (i.e., MSE) and error-uncertainty correlation (i.e., R) as a function different alpha values on the UCI datasets.

To **explain the uncertainty predictions** locally and globally, **we used LIME & SHAP:**

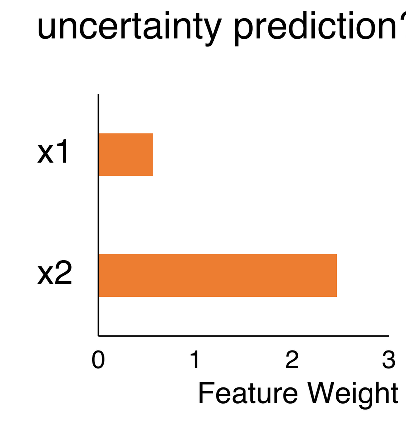
Example explainability results on simplified synthetic data with two independent variables:



Which feature drives the y prediction?



Which feature drives the uncertainty prediction?



15.089 CAPSTONE PROJECT

Schlumberger MIT OPERATIONS RESEARCH CENTER