

Surrogate uncertainty estimation for your time series forecasting black-box: learn when to trust

Problem they are trying to solve / Purpose of method

- The purpose of the model is to equip an already trained forecasting model with an uncertainty estimate
- Be able to equip uncertainty estimation on any forecasting model as long as you have a trained model and the data trained on the model

How does it differ from other methods?

- Instead of a built in method in the forecasting model, the method is an extension.
- The surrogate model is trained on same training data as the base model (black box model), but also enhanced by the base model

How the method works

Overview

BAMOES (Base-model-enhanced surrogate model)

- The surrogate model is a realization of gaussian process regression.
- The surrogate model trained with a combined loss function.
- They train a surrogate model to mimic the base model (already trained model).
- The uncertainty prediction will be a normal distribution (gaussian) where the mean is the prediction from the base model and the standard deviation from the surrogate model

New dataset

The method samples L amount of random datapoints from the domain the training points are in. This is used to query the base model to make new predictions the surrogate model will train to mimic the base model

Loss function

- The loss function is two parts
- Loglikelihood between the training data and the surrogate model
- Mse between surrogate model predictions and the base model predictions. This is only on the new sampled data (from new dataset) and not the original training data

$$L(s, \hat{f}, \mathcal{D}, \mathcal{D}^{\hat{f}}) = - (1 - C) \log p(\mathbf{y}|X, s) + \\ + C \sum_{i=1}^L (s(\mathbf{x}'_i) - \hat{f}(\mathbf{x}'_i))^2,$$

Figure 1: Combined loss function

Hyper parameters

- L, how many random datapoints to sample from the domain. Between 0-N amount. Where N is the size of training data.
- C, the coefficient to the loss function. Between 0-1. Higher C, more value more the mse between surrogate prediction and base model prediction. Lower C, value more the likelihood between the surrogate model and the training data.

Training

To the train the model:

- sample L amount of datapoints
- get the basemodel predictions on the new sampled data
- define the surrogate model (gaussian process regression)

A. BAMOES pseudocode

In Algorithm 1 we provide a pseudocode for our BAMOES approach. We assume, that a surrogate $s(\mathbf{x})$ has a corresponding kernel $k_s(\mathbf{x}, \mathbf{x}')$:

Algorithm 1 BAMOES training

Require: Data (X, \mathbf{y}) , base model \hat{f} , sample size L , loss weight C , number of epochs M

- 1: Sample uniformly L points of interest:
- 2: $X' = \{\mathbf{x}'_i\}_{i=1}^L, x'_{ij} \sim U(\text{low} = \min(X_{.j}), \text{high} = \max(X_{.j}))$
- 3: Get base model predictions $\mathbf{y}' = \{\hat{f}(\mathbf{x}'_i)\}_{i=1}^L$
- 4: Define GPR surrogate model $s(\mathbf{x})$
- 5: **for all** epoch $\in \{1, \dots, M\}$
- 6: Compute the log likelihood $l(s) = \log p(\mathbf{y}|X, s)$
- 7: Compute surrogate predictions $\{s(\mathbf{x}'_i)\}_{i=1}^L$
- 8: Compute $e(s) = \sum_{i=1}^L (s(\mathbf{x}'_i) - \hat{f}(\mathbf{x}'_i))^2$
- 9: Calculate loss: $\text{loss}(s) = -(1 - C)l(s) + Ce(s)$
- 10: Update $k_s(\cdot, \cdot)$ to minimize loss via Adam for the gradients of $\text{loss}(s)$
- 11: **return** Surrogate $s(\mathbf{x})$

Figure 2: BAMOEs pseudocode