POLA: Online Time Series Prediction by Adaptive Learning Rates

Paper ID: 3130

Institute for Infocomm Research I²R

Wenyu Zhang

Institute for Infocomm Research, Singapore

Online Time Series Prediction

Streaming Setting

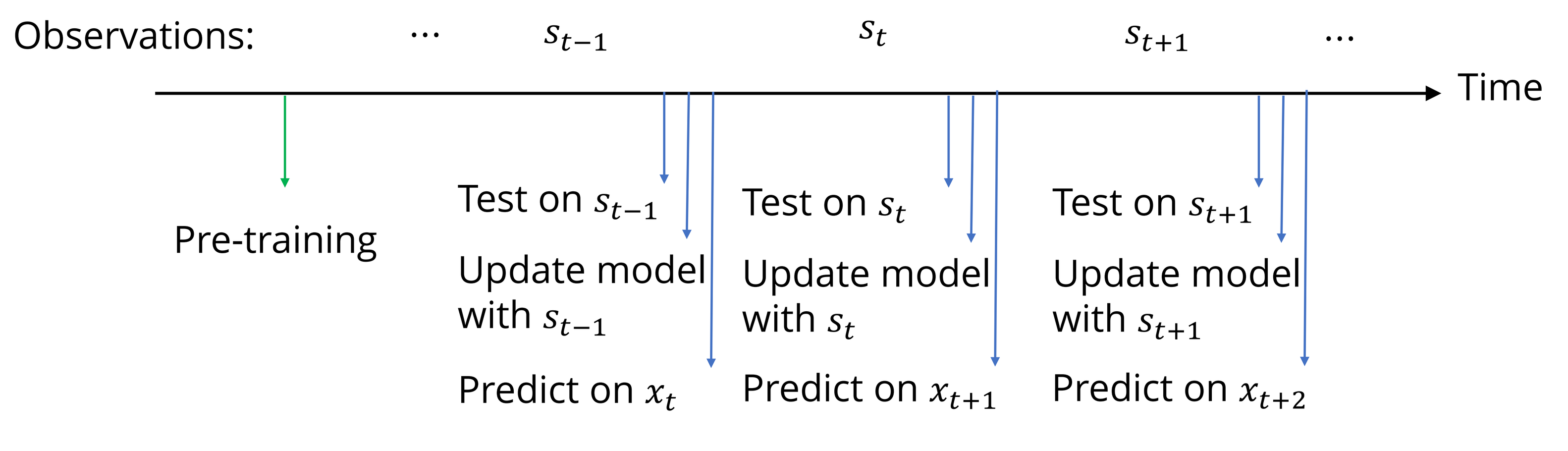
Data is collected by continuously monitoring a system. A key challenge for model deployments in dynamic environments is concept drift, where the joint distribution of predictor and response variables change across time. In order to avoid degrading performance, deployed models need to be updated when necessary.

Online Time Series Prediction

Unlike in the traditional online learning framework, temporal correlations means that observations cannot assumed to be independently and identically distributed.

Setup

- For process $\{Z_t\}$, observation $z_t \in \mathbb{R}^d$
- At time t, use a historical sequence x_t of length m, to predict an output forecast sequence y_t for the next n time steps
  - $x_t = [z_{t-m+1}, \dots, z_t]$
  - $y_t = [z_{t+1}, \dots, z_{t+n}]$
  - Denote prediction sample $s_t = (x_t, y_t)$
- Online batch size b



Example: $n = 1, b = 1$

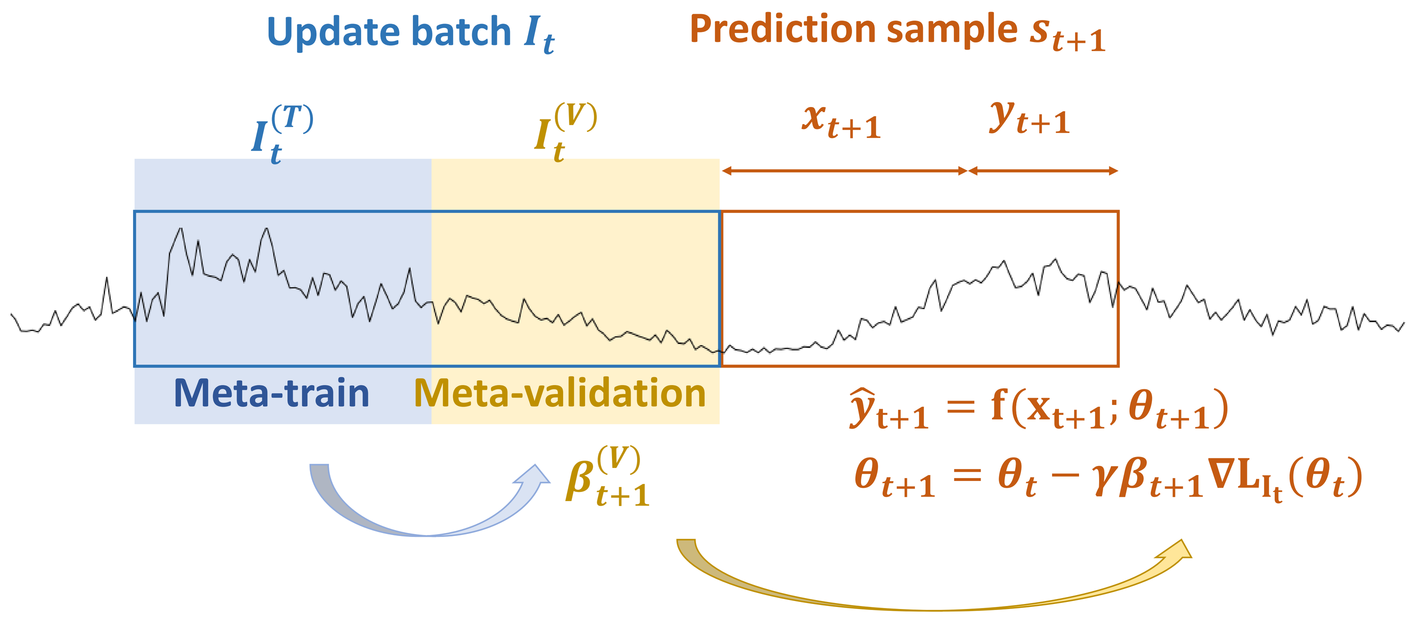
Proposed Approach: POLA

Main Contributions

- Adapt quickly in dynamic environments without overfitting to current system state or noisy samples, by
- Automatically scheduling the online learning rate of stochastic gradient descent (SGD) algorithm

Adaptive Learning Rate

- Maximum learning rate γ
- Learning rate factor $\beta_{t+1} \in [0,1]$
- Learning rate is small if the current update batch is not useful in helping the model adapt



Meta-Learn the Learning Rate Factor

- Split the update batch to a meta-training and meta-validation set
- Meta-learning sets are a proxy to the training and testing procedure
- Optimize the learning rate factor on the meta-learning sets

Bi-level Optimization in Meta-Stage

$$\theta_{t+1}^{(V)}, \beta_{t+1}^{(V)} = argmin_{\theta,\beta} L_{I_t^{(V)}}(\tilde{\theta})$$

subject to $\tilde{\theta} = argmin_{\theta}\{\gamma\beta L_{I_t^{(T)}}(\theta) - \frac{1}{2}||\theta - \theta_t||_2^2\}$

and approximate β_{t+1} with $\beta_{t+1}^{(V)}$

Implementation

1. POLA-FS: Search for $\beta_{t+1}^{(V)}$ in a finite set of candidates
2. POLA-GD: Optimizes for $\beta_{t+1}^{(V)}$ by gradient descent with learning rate η for k steps, while freezing all other model parameters

POLA-GD Learning Rate Factor Update

Step $i+1$: $\quad \alpha^{(i+1)} = \alpha^{(i)} - \eta\nabla_{\alpha^{(i)}} L_{I_t^{(V)}}(\theta_t - \gamma\sigma(\alpha^{(i)})\nabla L_{I_{t^{(T)}}}(\theta_t))$

$$\beta^{(i+1)} = \sigma(\alpha^{(i+1)})$$

where $\sigma(\alpha) = \frac{1}{1+e^{-\alpha}}$ is the sigmoid function

Datasets for POLA Evaluation

Sunspot

- Monthly sunspot number from January 1749 to July 2020
- Historical data length $m = 48$
- Forecast length $n = 5$

Household Power Consumption

- Daily power consumption (global active power, global intensity, voltage) from December 16, 2006 to November 26, 2010
- Historical data length $m = 28$
- Forecast length $n = 3$

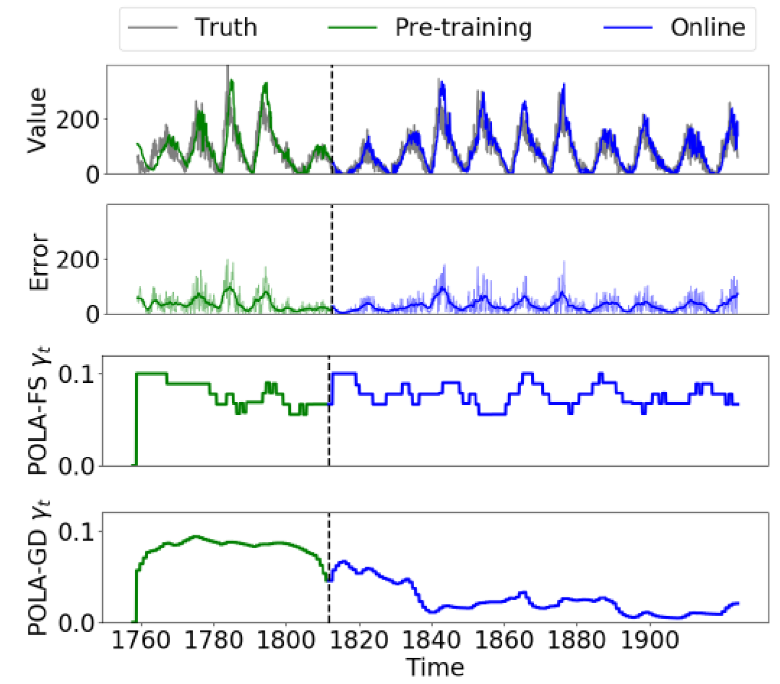
Experimental Results

Online Prediction Performance (Prequential):

Holt-Winters: exponential smoothing
OR-ELM: online recurrent extreme learning machine

Recurrent neural network
- Pre-trained: no model update in online phase
- FTL: Follow-The-Leader retraining every b steps
- MAML: meta-learning pre-training
- Online-SGD: constant learning rate
- Online-RMSprop: element-wise adaptive learning rate
- WG: adapts SGD learning rate based on whether current sample is outlier or change point

Sunspot Dataset



RNN

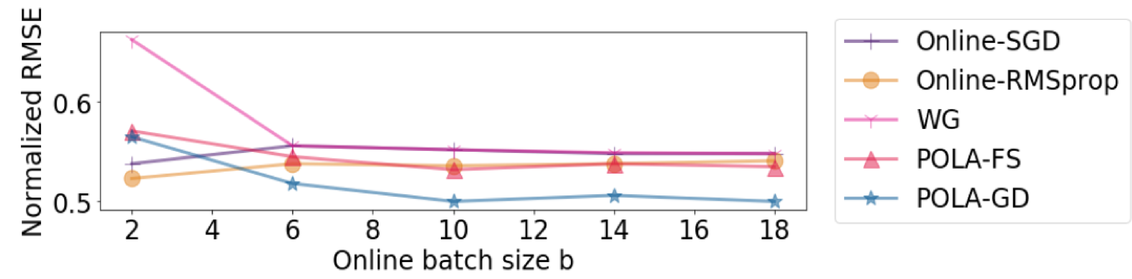
METHOD	NORMALIZED RMSE	
	Sunspot	Power
Holt-Winters	0.991	NA
OR-ELM	0.822	NA
Pre-trained	0.572	0.816
FTL*	0.572	0.820
MAML	1.295	1.023
Online-SGD	0.552	0.775
Online-RMSprop	0.536	0.809
WG	0.552	NA
POLA-FS	0.532 ± 0.002	**0.769** ± 0.003
POLA-GD	**0.500** ± 0.002	0.773 ± 0.005

LSTM & GRU

MODEL	METHOD	NORMALIZED RMSE	
		Sunspot	Power
LSTM	Online-SGD	0.532	0.821
	Online-RMSprop	0.517	**0.794**
	WG	0.532	NA
	POLA-FS	0.534 ± 0.009	0.802 ± 0.005
	POLA-GD	**0.512** ± 0.006	0.806 ± 0.071
GRU	Online-SGD	0.526	**0.768**
	Online-RMSprop	0.521	0.786
	WG	0.526	NA
	POLA-FS	0.508 ± 0.002	0.769 ± 0.003
	POLA-GD	**0.489** ± 0.002	**0.768** ± 0.003

Sensitivity Analysis:

Online Batch Size



POLA-GD Gradient Descent Hyperparameters

# STEPS (k)	LEARNING RATE (η)	NORMALIZED RMSE	
		Sunspot	Power
1	0.1	0.515	0.773
2	0.1	0.504	0.772
3	0.1	0.500	0.773
1	0.01	0.525	0.777
2	0.01	0.520	0.776
3	0.01	0.516	0.775

References

T. Guo, Z. Xu, X. Yao, H. Chen, K. Aberer and K. Funaya, "Robust Online Time Series Prediction with Recurrent Neural Networks," *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*.

J. Park and J. Kim, "Online Recurrent Extreme Learning Machine and Its Application to Time-Series Prediction," *2017 International Joint Conference on Neural Networks (IJCNN)*.

Anusha Nagabandi, I. Clavera, Simin Liu, Ronald S. Fearing, P. Abbeel, S. Levine and Chelsea Finn, "Learning to Adapt in Dynamic, Real-World Environments through Meta-Reinforcement Learning," *2019 arXiv*.

Geoffrey Hinton, Nitish Srivastava and Kevin Swersky, "Neural Network for Machine Learning."