Master's Thesis of Yejin Hwang

Time and Entity Adaptation on Panel Data Forecasting Via Meta Learning

패널 데이터 예측을 위한 시간 및 개체 적응에서의 메타러닝의 효과

February 2023

Graduate School of Data Science
Seoul National University
Data Science Major

Yejin Hwang

Time and Entity Adaptation on Panel Data Forecasting Via Meta Learning

Advisor Hyung−Sin Kim

Submitting a master's thesis of
Data Science

December 2022

Graduate School of
Data Science
Seoul National University
Data Science Major

Yejin Hwang

Confirming the master's thesis written by
Yejin Hwang
January 2023

Chair	Wen−Syan Li	(Seal)
Vice Chair	Hyung−Sin Kim	(Seal)
Examiner	Min−hwan Oh	(Seal)

Abstract

Keyword : Panel Data, Time-series, Meta learning, Few shot learning
Student Number : 2021-22944

Panel data refers to data with observations for multiple entities over time. The data is being used in diverse fields of research, including economics, energy, medical science, and physics. When dealing with panel data, researchers often encounter circumstances in which new entities are added. Researchers struggle to make prediction for these new entities owing to insufficient amount of data and distribution shift. Previous deep learning models lack generalizability to forecast the behavior of new entity in an unseen time, and none of the research addressed this challenge specifically. In this paper, we propose meta-learning based approach that enables model to extract general feature, or meta-knowledge across entities and times. The proposed approach can enhance the adaptability against unseen entity by leveraging this meta-knowledge and providing entity-specific few-shot adaptation. We designed unique task setting method for meta-learning that can well consider temporal characteristics of entity in panel data. We also suggest novel data split method which can represent the 3 different situations that can occur in panel data forecasting: existing entities in unseen time, unseen entity in existing time, and most importantly, unseen entity in unseen time. In evaluation on various panel data from broad range of domains, the results have demonstrated the effectiveness of meta-learning on panel data forecasting by achieving the performance improvement over conventional baseline models with most of the situations. Notably, our approach excelled the most in the situation of unseen entity and unseen time, which we are targeting on the most. It supports that our approach strengthens the model's generalizability to unseen data.

Table of Contents

Chapter 1. Introduction

1.1. Study Background

With the remarkable achievements of deep learning to process data and perform classification or regression tasks, deep learning technology has been actively studied in the fields of computer vision, natural language processing, and audio recognition. Recently, deep learning is expanding its applications to domains such as economics, manufacturing, medical, and climate study. The emergence of deep learning technology such as RNN networks and sequence to sequence learning to model sequential data for language processing and speech-to-text transcription, has enlarged the application field of deep learning to both univariate and multivariate time-series data as part of sequential data across different domains.

Compared to traditional methods which have mainly focused on domain specific prior knowledge to build parameteric models, modern methods with machine learning and deep learning try to learn temporal dynamics and representations from the data itself according to [1]. ML and DL struggled to outperform classical statistical approaches until the recent approach such as N-BEATS in [2] proved the effect of pure ML in time-series forecasting. Nonetheless, there still exists some persistent challenges such as non-stationarity and covariate shift in time-series forecasting, degrading performance in test time. Research in [3] has validated that meta learning can be a solution to overcome these challenges in time-series data.

Panel data is extended version of time-series data on entity or individual unit in specific domain. It is defined as cross-sectional time-series dataset which contains measurements of features over periods of time on multiple observed entities or units. Entities can be firms, households, countries, devices, demographic groups, or others depending on domains. Panel data is largely being used in areas like econometrics, social sciences, and medical study. It is complicated to analyze and forecast panel data because the model needs to capture both common and unique behaviors of entities considering temporal dynamics, handling the underlying challenges in time−series data at the same time. Conventional research dealing with panel data has been mainly focused on regression−based statistical model [4, 5]. However, it is difficult to capture nonlinear patterns and overcome non−stationarity with linear models. Additionally, data distribution differs not only by time axis, but also by entity. Therefore, the model can show limited performance without further adaptation and or technique to handle distribution shifts in case of unseen entity and timestep.

1.2. Purpose of Research

This research concentrates on verifying the effect of meta−learning on short−term panel data forecasting by overcoming the concept drift and non−stationarity problem, extracting the meta behaviors or patterns of entities over time, and providing additional adaptation for specific time and entity from the meta knowledge. The proposed method is generalizable in both the time and entity axes, working well even in a situation to forecast newly added entity containing small number of historical data. This scenario can often occur during panel study, such as the situations when stock is newly listed in stock market, new household is added for measuring energy consumption, or when the global disease starts to break out in new country. The contributions of this research can be summarized as:

- First approach to apply meta−learning methodology and to verify its effect on panel data forecasting through experiments using diverse panel datasets.
- Novel task design and train/validation/test set split method to consider time and entity axes of panel data.

Chapter 2. Related Work

2.1. Panel data forecasting

Panel data, or longitudinal data refers to data with time series observations of multiple entities or individuals. Thus, the data involves at least two dimensions, a cross-sectional dimension and time series dimension, and more than two dimensions if the data contains multiple features. Standard regression model based on the assumption of linear relation between variables is typical model for panel data forecasting. Methodological literature on panel data analysis [6] suggested ways to decompose parameters into structural parameters which are invariant across time and entity, and incidental parameters which vary though cross-section units and time series observations. Modeling unobserved heterogeneity among entities and over time is one of the main challenges in panel data analysis. The model is referred to as random effects model when heterogeneity is assumed as random variables, and fixed effects model when assumed as fixed parameters, and mixed effects model when assumed as both according to [7]. [5] demonstrated the superiority of machine learning methods such as decision tree, and random forest over linear methods in panel data prediction. These methods have limited ability to capture the complex temporal dynamics in data.

One of the main reasons that researchers have not used ML or DL for panel data was due to lack of interpretability compared to statistical models. To improve the interpretability of neural network model, [8] proposed interpretable neural network model to predict individual's monthly employment status, which is panel data.

However, none of the research focused on building adaptive model which is generalizable to entity and timestep that were not seen when fitting model and does not handle the non-stationary characteristic of the data properly. The propose idea in this paper attempts to overcome these limitations by adopting meta-learning methodology.

2.2. Meta Learning

Meta−learning, or learning−to−learn, refers to the learning techniques for generalization and quick adaptation to new task. It is typically used to perform few−shot learning tasks in situations with limited data. When it is required to make prediction for entity with small number of time−series observations in panel data, few−shot learning can be effective. Meta−learning provides an advantage to improve learning performance for a new task, by being trained over multiple learning episodes, as explained in [9]. On top of that, Woo et al. [3] demonstrated that meta−learning gives additional benefit to avoid critical problem of conditional distribution shift occurred in time−series prediction due to distribution differences between train and test set. As the panel data is composed of timeseries, handling unstationarity is significant to achieve performance improvement as in time−series data. Designing task used in meta−training and testing to include data from nearby timestamps allows the model to utilize the locally stationary distribution.

Meta learning algorithms can be categorized as metric−based and optimization−based. The former targets on learning similarity between samples within the same class in embedding space while the latter aims to find optimal set of model parameters that are capable of adapting to each task with only a small number of gradient updating steps. Metric−based algorithms such as Siamese network from [10], prototypical network from [11] and matching network from [12] learn a metric or distance function over samples. Metric−based techniques are conceptually simple and can be fast when the number of tasks is small. However, they are unable to learn when tasks from meta−train time and meta−test time are distant. Additionally, when tasks become larger, the models are computationally expensive. Unlike metric−based approaches, optimization−based approaches can achieve good performance on wider range of task distributions. Finn et al. [13] presented model−agnostic meta−learning (MAML), which is the representative method of optimization based meta−learning. The method attained exceptional attention with its simplicity and powerful performance. Reptile from [14] is another optimization−based method, which saves time and computation with slight sacrifice on performance compared to MAML. Compared to image classification, there has not been a lot of research to apply meta−learning on time−series or panel data. In this research, MAML algorithm is used to validate the effect of meta−learning on panel data forecasting.

Chapter 3. Method

3.1. Task definition

Each entity is assumed to have different distribution and follow locally stationary distribution at nearby timestamps. Therefore, every task with different entity and timestamp is an independent task of data samples following similar distribution each.

$$\tau_i \sim P(\tau | e_m, t_n) \qquad (1)$$

In eq. (1), τ_i indicates task instance i, e_m and t_n denote the entity m and timestamp n. Tasks are grouped into disjoint sets, training meta-set S^{tr}, validation meta-set S^{val}, and test meta-set S^{test}. S^{tr} is used to train meta-learner, validation meta-set is used for model selection, and test meta-set is used to evaluate mode's generalizability performance. Validation meta-set is separated into 3 subsets $\{S^{val_1}, S^{val_2}, S^{val_3}\}$. S^{val_1} differs from S^{tr} in time axis, S^{val_2} differs in entity axis, and S^{val_3} differs in both axes. Similar to validation meta-set, test meta-set is also partitioned into 3 subsets: $\{S^{test_1}, S^{test_2}, S^{test_3}\}$ by time and entity axes. Model with highest f1 score of 3 validation sets is selected for test time to enable model to adapt well in all the test sets.

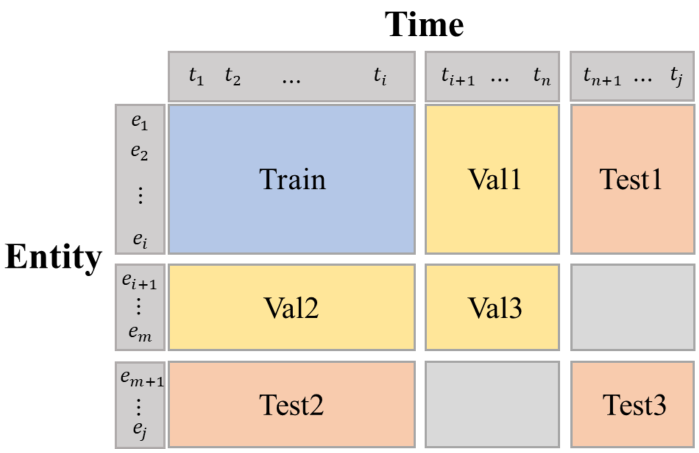

Figure 1: Meta-Training, Meta-Validation, Meta-Test sets

Each task instance τ_i is sampled from the distribution of eq. (1)

which is sampled from time−series of each entity. The task consists of support set D^{tr} and a query set D^{val}. Support set design differs depending on the type of problem. In classification task, support set D^{tr} includes the closest $N * K$ subsequence samples, K samples for each of the N classes to ensure that the model can learn dynamics of both classes evenly. In regression task, support set D^{tr} includes the N_s subsequence samples which are continuous. Each subsequence sample is composed of features in lookback window $\{x_{d-w}, \dots, x_{d-1}\}$ where w denotes the window size, and corresponding labels or target values within forecast horizon h from target day d, which are $\{y_d, \dots, y_{d+h-1}\}$. Query set D^{val} contains the subsequence sample that comes right after the last sample in support set. Fig 2 shows how support set and query set are decided in each task. Since subseries samples in a task are selected from nearby timeseries, model can effectively utilize locally stationary property instead of struggling from non−stationary problem.

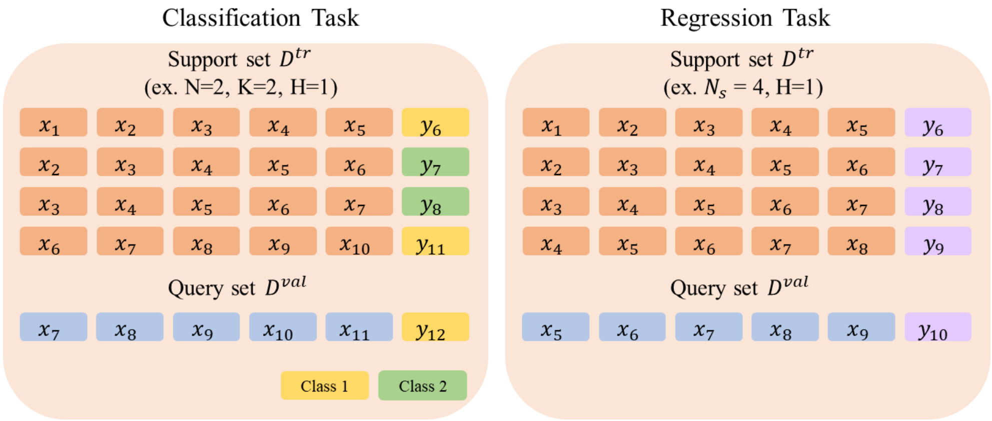

Figure 2: Support & Query set example

3.2. Model

Model Overview

I propose model framework based on model−agnostic meta−learning (MAML) algorithm from [13] to update model and task parameters. Meta−learning aims to make model which can quickly learn a new task from few unseen data by exposing and training model on many different tasks. The main point underlying the MAML algorithm is to optimize model's initial parameters (meta−learner) that allow model to adapt to new task only after few parameters update steps with a small amount of data. The intuition behind the approach is that there exist some meta−knowledge or representations than can be shared broadly with other tasks. The transferrable meta representations can help tasks to learn more rapidly. In panel data, this can be the undefined common characteristics among entities, and task−wise adaptation is beneficial to capture task−specific patterns for entity in an unseen time. MAML involves 2 training stages: one is inner loop to update task−wise parameters for task−specific adaptation and the other is outer loop to update model−parameters for meta−optimization. Any model that can encode temporal dynamics in sequences can be used to be optimized.

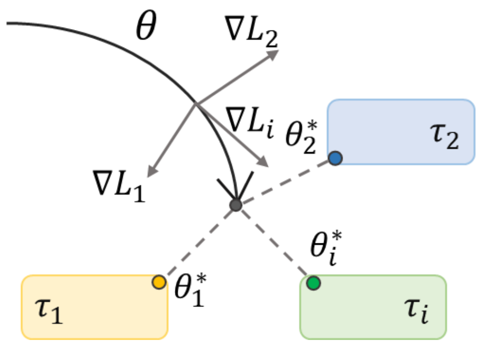

Figure 3: MAML algorithm for task adaptation in new entity

Inner-Loop

Inner-loop is where task-specific adaptation takes place. Starting from the optimized initial meta-parameters, task-wise parameters is updated on gradient descent update using the loss produced from support set in given task τ_i. The meta model parameters θ become θ'_i after inner adaptation as in equation (2). α is inner learning rate.

$$\theta_i' = \theta - \alpha \nabla_\theta L_{T_i}(f_\theta) \tag{2}$$

Outer-Loop

Outer-loop is to perform meta-optimization over model parameters θ. Query losses from adapted task-wise parameters across sampled tasks in a batch are used to update the model parameters as in equation(3), and the parameters are updated via stochastic gradient descent(SGD) with the outer learning rate β as in equation(4).

$$\min_\theta \sum_{T_i \sim p(T)} L_{T_i}\left(f_{\theta_i'}\right) = \sum_{T_i \sim p(T)} L_{T_i}(f_{\theta - \alpha \nabla_\theta L_{T_i}(f_\theta)}) \tag{3}$$

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_{T_i \sim p(T)} L_{T_i}(f_{\theta_i'}) \tag{4}$$

Loss functions

Cross-entropy in equation (6) is used to update gradient descent for classification task and mean-squared error in equation (5) is used for regression task in this research.

$$L_{T_i}\left(f_{\theta_i}\right) = \sum_{x^{(j)}, y^{(j)} \sim T_i} \left\| f_{\theta_i}\left(x^{(j)}\right) - y^{(j)} \right\|_2^2 \tag{5}$$

$$L_{T_i}\left(f_{\theta_i}\right) = \sum_{x^{(j)}, y^{(j)} \sim T_i} y^{(j)} \log f_{\theta_i}\left(x^{(j)}\right) + \left(1 - y^{(j)}\right) \log\left(1 - f_{\theta_i}\left(x^{(j)}\right)\right) \tag{6}$$

Encoder

Any models that can effectively capture temporal dynamics in sequence data can be used as encoder. Models used in the experiments in this paper are Attention-LSTM(ALSTM) network from [15] to embed inputs considering temporal dependencies.

LSTM(Long short-term memory) in [16] learns a representation of the time-series by updating cell state and hidden state through time steps. ALSTM is LSTM network combined with attention mechanism to reflect importance of each time step effectively. Instead of only using the last hidden state, ALSTM takes all the hidden states to make encoded output. It computes the attention score α_i for each time step i, which is shown in equation (7). The embedded output h is computed as $h = \sum_i \alpha_i h_i$.

$$\alpha_i = \frac{\exp(h_i^T h_T)}{\sum_{j=1}^{T} \exp(h_j^T h_T)} \tag{7}$$

Chapter 4. Experiments

4.1. Experiment Settings

Data

In the experiments, two public panel data sets from different domains (economics and environment) are used to demonstrate the effect of meta learning. Because there has not been active research on forecasting panel data, I selected the panel data sets that are commonly used in multivariate time-series forecasting task without considering entity axis, aiming at short-term forecasting.

First dataset is the KDD17 [18], which includes stock price data of each firm. The dataset contains stock prices data of 50 companies from Jan-01-2007 to Jan-01-2016 in U.S markets. It has 11 features to express the trend of each stock, processed from historical prices (open, close, low, high, adj-close). The task is classification task to predict the movement of stock price. Labels are given by the movement of close prices, as the data instances with movement percent $\geq 0.55\%$ are labeled as positive, and $\leq -0.5\%$ are labeled as negative samples. Table 1 shows how KDD17 dataset is split.

Other 3 datasets are the benchmark datasets typically used for multivariate timeseries task [19]. Second dataset is the electricity consumption data to forecast the electricity consumption for client, which is regression task. The dataset is recorded with electricity consumption in kWh for 321 clients between 2012 and 2014, and the data is converted to hourly consumption in the experiment. Third dataset is Solar-Energy dataset, which contains the solar power production records in 2006. Records are sampled every 10 minutes from 137 plants in Alabama State. Last data set is exchange-rate dataset with daily exchange rates of eight countries from 1996 to 2016.

All the data sets are split into train, validation-time(val1), validation-entity(val2), validation-mix(val3), test-time(test1), test-entity(test2), and test-mix(test3). 60% of the data is used for training set both in time and entity axes, and 20% for validation set, and remaining 20% for test set.

Baselines

The performance of the proposed method is compared with ALSTM from [15] and Adv-ALSTM in [16] with KDD17. ALSTM [15] and was used as feature extractor for electricity consumption, solar-energy, and exchange rate dataset.

Evaluation metrics

The performances of the methods are evaluated with accuracy, defined as the $\frac{\text{\# of correct samples}}{\text{Batch size}} * \mathbf{100}$ for KDD17, and evaluated with MAE and MSE for electricity dataset.

Dataset	KDD17	
	Time	Entity
Train	(07/01/01~14/12/31)	35
Val1	(15/01/01~15/12/31)	35
Val2	(07/01/01~14/12/31)	10
Val3	(15/01/01~15/12/31)	10
Test1	(16/01/01~16/12/31)	35
Test2	(07/01/01~14/12/31)	5
Test3	(16/01/01~16/12/31)	5

Table 1: Train set, Validation sets, Test sets split for KDD17

Hyperparameter Settings

The proposed method TEAP is implemented with Pytorch and the Adam optimizer is used in outer loop update. In the experiment with KDD17, hidden size in ALSTM is 32, window size is 15, inner learning rate is fixed to 0.01, and initial outer learning rate is set to 0.005. 5 gradient updates take place in each inner loop. Hyperparameters used for the experiment in table 3 are shown in table 4.

Hyperparameter	hidden	Inner−lr	Outer−lr	n−inner−step	horizon	Lookback window
Electricity	64	0.0001	0.009	10	3	168
Solar−Energy	64	0.001	0.0001	10	3	160
Exchange−rate	64	0.0001	0.00001	10	3	179

Table 4: Hyperparameter settings on Electricity, Solar-Energy, Exchange-rate datasets

4.2. Experiment Results

Performance Comparison

Table 2 and Table 3 show the performances of baselines and proposed method on KDD17 and electricity dataset respectively. As can be seen in table 4, the proposed method TEAP outperforms baselines with KDD17. It is especially notable that the performance gap is larger between proposed model and baselines with test3, compared to test1 and test2. Also, the model showed better prediction results with test3 than test1 in solar energy and exchange rate dataset. Heterogeneity between tasks from meta-train and meta-test are largest with test3, and smallest with test2. Outstanding performance in test3 implies that the meta-learning effectively enhances the model's generalizability to completely new data. I attribute the performance improvement of TEAP to capturing meta-knowledge across entities and times using meta-learning and task-wise adaptation for each entity.

Method	ALSTM	Adv-ALSTM	TEAP
Metric	Accuracy	Accuracy	Accuracy
Test1	51.26	51.53	52.66
Test2	54.18	54.07	53.44
Test3	49.48	49.10	**54.84**

Table 2: Performance comparison on KDD17

Dataset	Electricity		Solar-Energy		Exchange-rate	
Metric	MSE	MAE	MSE	MAE	MSE	MAE
Test1	0.1079	0.2481	0.0508	0.1519	0.0541	0.2250
Test2	0.1064	0.2441	0.0586	0.1630	0.0025	0.0413
Test3	0.1123	0.2460	0.0430	0.1430	0.0158	0.1071

Table 3: TEAP's Performance of 3 test sets on Electricity, Solar-Energy, Exchange- rate datasets

Among 3 test sets (test1, test2, test3), this research is primarily focused on test3, representing the different entity and time from train set. Therefore, test 3 is used in the fallowing experiments.

Ablation Study

Effect of varying window during training in KDD17

Unlike conventional models that could only be trained with certain window, TEAP allows model to learn dynamic patterns from varying window sizes during training time by task−based learning.
It is shown that robust window training can raise performance and achieve efficiency since it does not need to be trained multiple times for each window size and model can learn various temporal patterns. Figure 4 verifies the effectiveness of robust window training compared to baselines. When TEAP is trained window sizes of [5,10,15,20] with KDD17, it outperforms baselines especially when window size is large.

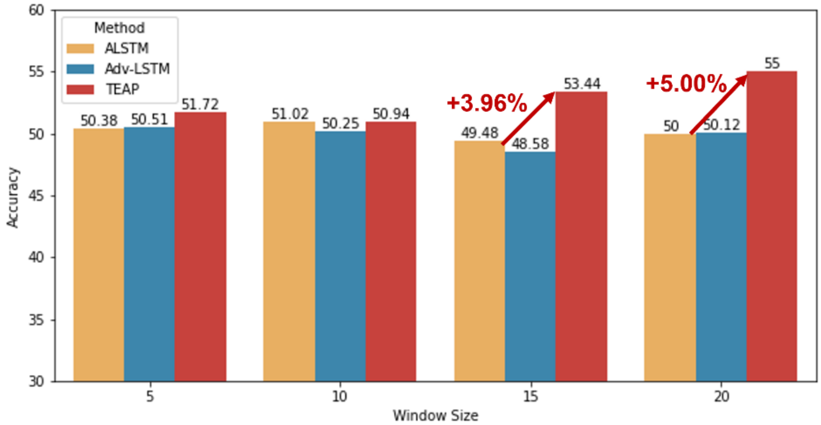

Figure 4: Performance comparison by window sizes

Effect of number of support samples (K shots) in KDD17

TEAP’s performance showed tendency to rise as the number of support samples per each class K increases. Figure 5 showed it achieves performance gain over 5%p when K = 10, and 2%p only when K = 1. Test3’s performance improvement by increasing K can be interpreted that it can adapt better given more data since test3 involves data which is different from train set in both time and entity axes.

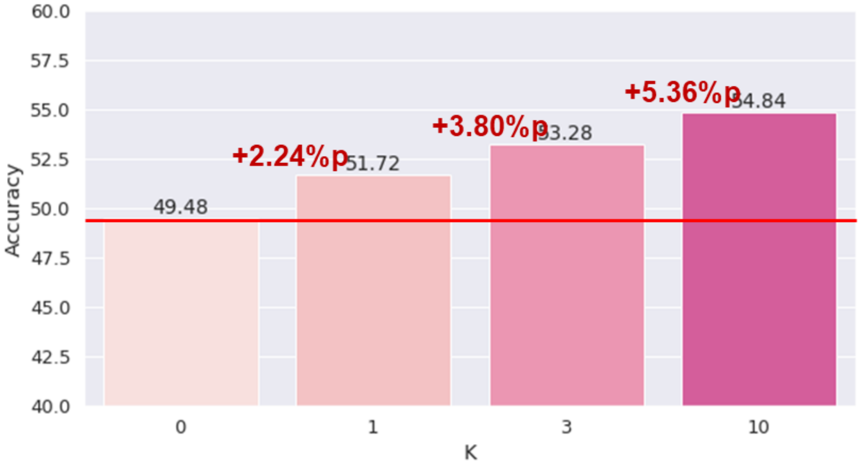

Figure 5: Performance change by K shot

Chapter 5. Conclusion

In this paper, I propose a novel methodology using deep learning based on meta-learning to perform panel data forecasting. Deep learning has not been heavily used for panel data forecasting so far, as it has been considered difficult to model common characteristics across entities and over time, as well as unique patterns of each entity at the same time. The proposed method can effectively discover meta-knowledge across entities and adapt rapidly to new task with few samples in sequence by task-based learning. Researchers can often face the problem of data scarcity while analyzing panel data when the new entity is added. The proposed method is especially powerful in such circumstances, outperforming the baseline models that do not use meta-learning. Also, the non-stationarity in data makes forecasting even harder in test time since the distribution can be different from train time. The proposed method mitigates distribution shift problem by splitting tasks into shorter subsequences in nearby timesteps. I demonstrated the method's effectiveness especially at the situation of unseen time and unseen time using 4 panel datasets from different domains of economics and energy. I expect that the method be utilized in other areas such as medical science, or physics.

Bibliography

1] Lim, Bryan, and Stefan Zohren. "Time-series forecasting with deep learning: a survey." Philosophical Transactions of the Royal Society A 379.2194 (2021): 20200209.

[2] Oreshkin, Boris N., et al. "N-BEATS: Neural basis expansion analysis for interpretable time series forecasting." arXiv preprint arXiv:1905.10437 (2019).

[3] Woo, Gerald, et al. "DeepTIMe: Deep Time-Index Meta-Learning for Non-Stationary Time-Series Forecasting." arXiv preprint arXiv:2207.06046 (2022).

[4] Baltagi, Badi H. "Forecasting with panel data." Journal of forecasting 27.2 (2008): 153-173.

[5] Chen, James Ming. "An introduction to machine learning for panel data." International Advances in Economic Research 27.1 (2021): 1-16.

[6] Hsiao, Cheng. Analysis of panel data. Cambridge university press, 2022.

[7] Hsiao, Cheng. "Panel data analysis—advantages and challenges." Test 16.1 (2007): 1-22.

[8] Yang, Yucheng, and Zhong Zheng. "Interpretable neural networks for panel data analysis in economics." arXiv preprint arXiv:2010.05311 (2020).

[9] Hospedales, Timothy, et al. "Meta-learning in neural networks: A survey." IEEE transactions on pattern analysis and machine intelligence 44.9 (2021): 5149-5169.

[10] Koch, Gregory, Richard Zemel, and Ruslan Salakhutdinov. "Siamese neural networks for one-shot image recognition." ICML deep learning workshop. Vol. 2. 2015.

[11] Snell, Jake, Kevin Swersky, and Richard Zemel. "Prototypical networks for few-shot learning." Advances in neural information processing systems 30 (2017).

[12] Vinyals, Oriol, et al. "Matching networks for one shot learning." Advances in neural information processing systems 29 (2016).

[13] Finn, Chelsea, Pieter Abbeel, and Sergey Levine. "Model-agnostic meta-learning for fast adaptation of deep networks." International conference on machine learning. PMLR, 2017.

[14] Nichol, Alex, Joshua Achiam, and John Schulman. "On first-order meta-learning algorithms." arXiv preprint arXiv:1803.02999 (2018).

[15] Li, Hao, Yanyan Shen, and Yanmin Zhu. "Stock price prediction using attention-based multi-input LSTM." Asian conference on machine learning. PMLR, 2018.

[16] Nelson, David MQ, Adriano CM Pereira, and Renato A. De Oliveira. "Stock market's price movement prediction with LSTM neural networks." 2017 International joint conference on neural networks (IJCNN). Ieee, 2017.

[17] Bai, Shaojie, J. Zico Kolter, and Vladlen Koltun. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling." arXiv preprint arXiv:1803.01271 (2018).

[18] Xu, Yumo, and Shay B. Cohen. "Stock movement prediction from tweets and historical prices." Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018.

[19] Lai, Guokun, et al. "Modeling long-and short-term temporal patterns with deep neural networks." The 41st international ACM SIGIR conference on research & development in information retrieval. 2018.

초 록

Keyword : 패널데이터, 시계열, 메타러닝, 퓨샷러닝
Student Number : 2021-22944

본 연구는 여러 개체들을 복수의 시간대에서 관측하여 얻은 데이터인 패널 데이터의 예측에 도움을 줄 수 있는 메타러닝의 효과를 입증한다. 메타러닝을 기반으로 한 제안 기법은 모델로 하여금 개체와 시간 축에서의 메타 지식을 효과적으로 추출할 수 있도록 하여 개체와 시간 축에 대한 모델의 적응성을 강화한다. 또한 태스크 별로 파라미터 최적화가 이루어질 수 있도록 하여 특정 개체의 특정 시간대에서의 개별 패턴을 추가적으로 학습할 수 있도록 하여 모델이 공통 특성과 개별 특성을 고루 학습할 수 있도록 한다. 기업 별 주가 데이터와 클라이언트 별 에너지 소비량 데이터셋을 사용한 실험을 통해 메타 러닝을 통한 학습이 새로운 개체와 시간대에서의 성능 개선에 도움이 됨을 보여주었다. 본 연구는 패널 데이터 예측 시 딥 러닝의 활용 가능성을 보여주며, 패널 데이터의 개체와 시간 축을 모두 고려한 새로운 태스크 구성 기법과 데이터셋 분리 방법을 제안한다는 점에서 가치를 지닌다. 특히 데이터의 양이 한정되어 있는 새로운 개체에 대한 예측을 수행해야 하는 상황에서 모델이 빠르고 효과적으로 학습할 수 있도록 한다는 점에서 유용하게 활용될 수 있다.