Contrastive Learning for Multivariate Time Series Forecasting

Zhenhua Zhuang [1]

Abstract

Multivariate time-series forecasting is a critical task for many real-world applications. It is a challenging problem that requires accounting for both the temporal relationships within each individual series and the relationships between the different series. Despite recent efforts to address these two types of correlations, most current methods focus solely on capturing temporal dependencies and rely on pre-existing assumptions about the relationships between the series.

In this paper, we propose C2Linear to further improve the accuracy of multivariate time-series forecasting. It captures inter-series correlations via contrastive regularization combining variance and covariance constraints which explores the mutual information between various series and decorrelates the embedding variables from each other. The extensive experiments on different datasets from real-world show the effectiveness and robustness of the method we proposed when compared with existing state-of-the-art methods.

1. Introduction

Time-series forecasting is an essential task in countless domains including medical monitoring, e-learning, energy and smart grid management, economics and finance. It helps people to make important decisions if the future evolution of events or metrics can be estimated accurately.

Existing forecasting models are designed focusing on exploring intra-series temporal patterns. In particular, a large amount of research works interest in Transformer-based modeling techniques are dedicated to the long time-series forecasting task, e.g., (Zhou et al., 2021; Wu et al., 2021; Zhou et al., 2022; Liu et al., 2022). Embarrassingly, LSTM-Linear (Zeng et al., 2023) questions the effectiveness of emerging favored Transformer-based solutions for the long-term time series forecasting problem via an simple yet powerful linear model. However, making accurate multivariate time series forecasting only based on intra-series patterns is challenging, as inter-series correlations are also need to be modeled.

Motivated by the above, in this paper, we propose a novel architecture to better model the inter-series correlations, named C2Linear. The **main contributions** of this paper are summarized as follows:

- To the best of our knowledge, C2Linear is the first work that explores inter-series correlations via *contrastive regularization*. The model is general for all multivariate time-series without pre-defined topologies.
- C2Linear achieves SOTA performances on nine public benchmarks of multivariate time-series forecasting. The experiments show the effectiveness and robustness of the method we proposed.

2. Related Work

Contrastive Learning. Contrastive learning (CL) is a representative self-supervised learning (SSL) method, which provides a promising paradigm to measure the dependency of input variables by calculating their mutual information. In past years, several classical works have proposed to utilize contrastive learning for computer vision, neural language processing and graph data. MoCo (He et al., 2020) builds a large dynamic dictionary with a queue and a movingaveraged momentum encoder. SimCLR (Chen et al., 2020) designs a simple contrastive learning framework with a composition of data augmentations and projectors for CL. VICReg (Bardes et al., 2022), which inspires our method, learns invariance to different views with a invariance term, avoids collapse of the representations with a variance preservation term,

[1]Nanjing Unversity.

Time Series Analysis Class Project, Autumn, 2022.

and maximizes the information content of the representation with a covariance regularization term.

Transformer-Based LTSF Solutions. There has been a significant amount of research on using transformer models to forecast long-term time series data. We summarizes several approaches to this problem. LogTrans (Li et al., 2019) utilizes convolutional self-attention layers with a LogSparse design to capture local information and reduce complexity. Informer (Zhou et al., 2021) employs ProbSparse self-attention with distillation techniques to extract the most important keys efficiently. Autoformer (Wu et al., 2021) incorporates ideas from traditional time series analysis methods, such as decomposition and auto-correlation. FED-former (Zhou et al., 2022) employs a Fourier enhanced structure to achieve linear complexity. Pyraformer (Liu et al., 2022) uses a pyramidal attention module with inter-scale and intra-scale connections to also achieve linear complexity. However, making accurate multivariate time series forecasting only based on intra-series patterns is challenging, as inter-series correlations are also need to be modeled.

Non-Transformer-Based LTSF Solutions. Besides Transformers, other popular DNN architectures are also applied for time series forecasting. StemGNN (Cao et al., 2020) combines Graph Fourier Transform (GFT) which models inter-series correlations and Discrete Fourier Transform (DFT) which models temporal dependencies in an end-to-end framework without using pre-defined priors. CATN (He et al., 2022) learn inter-series hierarchical and grouped correlation via a tree-based deep learning approach and utilizes a multi-level learning mechanism to capture long-, short-range and cross temporal patterns for intra-series data. RGSL (Yu et al., 2022) contains a laplacian matrix mix-up module to discover implicit time-series pattern and fuse both the explicit graph and implicit graph in a dynamic convex fashion. SCINet (LIU et al., 2022) conducts sample convolution and interaction for temporal modeling and forecasting. LSTM-Linear (Zeng et al., 2023) questions the effectiveness of emerging favored Transformer-based solutions for the long-term time series forecasting problem via an simple yet powerful linear model.

3. Motivation

Multivariate time-series forecasting is a challenging problem as one needs to consider both intra-series temporal correlations and inter-series correlations simultaneously. Existing forecasting models (Zhou et al., 2021; Wu et al., 2021; Zhou et al., 2022; Liu et al., 2022; Zeng et al., 2023) are designed focusing on exploring intra-series temporal patterns. However, making accurate multivariate time series forecasting only based on intra-series patterns is challenging, as inter-series correlations are also need to be modeled (Cao et al., 2020).

Contrastive learning, a highly regarded self-supervised learning method, is widely acknowledged for its capacity to assess the interdependence of input variables through the calculation of mutual information (Liu et al., 2023). Also, experiments (Bardes et al., 2022) show that the contrastive learning term can learn the representation features better and stabilize the training process. For multivariate time series forecasting, the mutual information between various series is worth exploring and can benefit final forecasting results. Furthermore, contrastive learning methods can accurately illustrate the semantic correlations of inter-series and help to gain an appropriate representation of multivariate time series.

4. Methods

In this section, we first introduce the backbone of C2Linear model. We then describe the proposed contrastive regularization method in detail.

4.1. C2Linear Model

To simplify, we present the pure two linear layers model, named C2Linear, as our backbone. The basic formulation of directly regresses historical time series for future prediction via a weighted sum operation (as illustrated in Figure 1).

The mathematical expression is $Z_i = W_1 X_i$, $\hat{Y}_i = W_2 Z_i$, where $W_1 \in \mathbb{R}^{D \times L}$, $W_2 \in \mathbb{R}^{T \times D}$ is linear layer along the temporal axis. $\hat{Y}_i$ and X_i are the prediction and input for each i_{th} variate. Z_i is the corresponding representation embedding for each dim.

Like LSTM-Linear (Zeng et al., 2023), C2Linear is also a set of linear models. *Vanilla C2Linear* is a two-layers linear model. To handle time series across different domains (e.g., finance, traffic, and energy domains), we further introduce two variants with two preprocessing methods, named *CD2Linear*, which decomposes a raw data input into a trend component and a remainder component, and *CN2Linear* which first subtracts the input by the last value of the sequence to boost the performance when there is a distribution shift in the dataset.

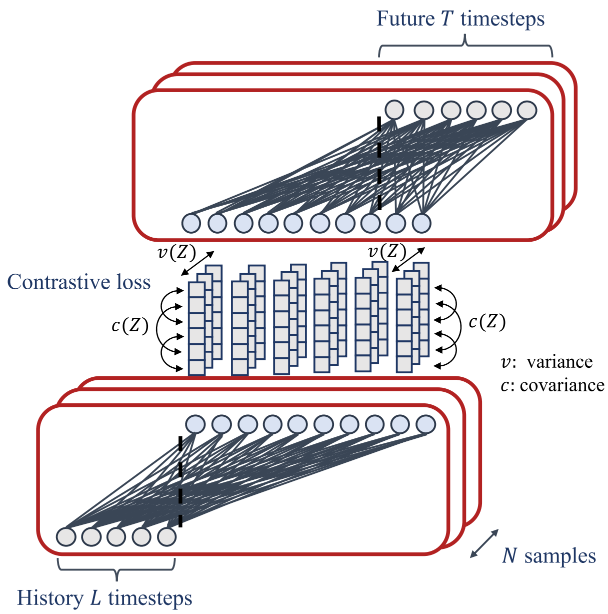

Figure 1. **C2Linear: joint linear embedding architecture with contrastive regularization, e.g. variance and covariance constraints.** Given a batch of history timestamps and are then encoded to producing the embeddings. To fully explore the mutual information between various series and decorrelate the variables from each other, the variance of each embedding variable over a batch is maintained above a threshold, and the covariance between pairs of embedding variables over a batch are attracted to zero.

4.2. Standardization Method

We use the general standardization, i.e. z-score methods to remove the unit limit of the data so that indicators of different magnitudes can be compared.The mathematical expression is as follows,

$$X_t' = \frac{X_t - \mu}{\sigma}.$$

4.3. Contrastive Regularization

Inspired by VICReg (Bardes et al., 2022), a SOTA contrastive method, we introduce the contrastive regularization for training joint linear embedding architectures. The basic idea is to use a regularition function with two terms:

- **Variance**: maintains the standard deviation over a batch of each time serie variable of the embedding above a given threshold and forces the embedding vectors of samples within a batch to be different.

- **Covariance**: attracts the covariances over a batch between every pair of centered embedding variables towards zero. This term decorrelates the different time series variables of each embedding to maximize the mutual information between various time series.

The two terms are applied to the architecture separately, thereby preserving the information content of each embedding at a certain level and maximizing the mutual information between various time series. The variance term explicitly prevents a collapse due to a shrinkage of the time serie embedding vectors towards zero. The covariance term decorrelates the different time series variables of each embedding to maximize the mutual information between various time series. C2Linear is more generally applicable than most of the exsiting methods (Cao et al., 2020; He et al., 2022; Yu et al., 2022; LIU et al., 2022) because of fewer constraints on the architecture and fewer prior knowledge to use.

The time series are processed in batches, and we denote $Z = [z_1, \ldots, z_n]$ the batch composed of n vectors of dimension d. We denote by z^j the vector composed of each value at time serie j in all vectors in embedding Z. We define the variance regularization term v as a hinge function on the standard deviation of the time series embeddings along the batch dimension:

$$v(Z) = \frac{1}{d}\sum_{j=1}^{d} \max(0, \gamma - S(z^j, \epsilon)),$$

$$S(x, \epsilon) = \sqrt{\mathrm{Var}(x) + \epsilon},$$

γ is a constant target value for the standard deviation. This criterion encourages the variance inside the current batch to be equal to γ along each dimension, forcing the embedding vectors of samples within a batch to be different.

We define the covariance regularization term c as the sum of the squared off-diagonal coefficients of the covariance $C(Z)$ with a factor $1/d$:

$$C(Z) = \frac{1}{n-1}\sum_{i=1}^{n}(z_i - \bar{z})(z_i - \bar{z})^T.$$

$$c(Z) = \frac{1}{d}\sum_{i \neq j}[C(Z)]_{i,j}^2.$$

This term encourages the off-diagonal coefficients of $C(Z)$ to be close to 0, decorrelating the different dimensions of the embeddings and maximizing the mutual information between various time series.

The overall loss function is a weighted average of the variance, covariance regularization terms and MSE loss:

$$\ell_{total} = \mu v(Z) + \nu c(Z) + MSE(Y, \hat{Y}),$$

where μ and ν are hyper-parameters controlling the importance of each term.

5. Experiments

5.1. Datasets

We evaluate the performance of our proposed method on 9 popular datasets, including Weather, Traffic, Electricity, Exchange-Rate, ILI and 4 ETT datasets (ETTh1, ETTh2, ETTm1, ETTm2). These datasets have been extensively utilized for benchmarking and publicly available on (Zeng et al., 2023). The statistics of those datasets are summarized in Table 1.

- ETT (Electricity Transformer Temperature) (Zhou et al., 2021)[1] consists of two hourly-level datasets (ETTh) and two 15-minute-level datasets (ETTm). Each of them contains seven oil and load features of electricity transformers from July 2016 to July 2018.
- Traffic[2] describes the road occupancy rates. It contains the hourly data recorded by the sensors of San Francisco freeways from 2015 to 2016.
- Electricity[3] collects the hourly electricity consumption of 321 clients from 2012 to 2014.
- Exchange-Rate (Lai et al., 2018)[4] collects the daily exchange rates of 8 countries from 1990 to 2016.
- Weather[5] includes 21 indicators of weather, such as air temperature, and humidity. Its data is recorded every 10 min for 2020 in Germany.
- ILI[6] describes the ratio of patients seen with influenza-like illness and the number of patients. It includes weekly data from the Centers for Disease Control and Prevention of the United States from 2002 to 2021.

5.2. Performance Comparison

Following previous works (Zhou et al., 2021; Wu et al., 2021; Zhou et al., 2022; Zeng et al., 2023), we use Mean Squared Error (MSE) and Mean Absolute Error (MAE) as the core metrics to compare performance.

Performance results. Table 2 reports the comparison of all mentioned Transformers and LTSF-Linears' performance in nine benchmarks. The best performing methods are bold, while the strongest Transformer baselines are underlined. We observe that:

- **C2Linear outperforms the state-of-the-art baselines in longer-term forecasting especially 336 and 720 across all datasets.** In particular, it achieves consistent improvements on MSE metric over the best LTSF baselines in longer-term forecasting ($T \in \{336, 720\}$). This clearly demonstrates that with the contrastive learning term, C2Linear can accurately illustrate the semantic correlations of inter-series and improves the forecasting power especially for longer time series.

- **C2Linear outperforms the state-of-the-art baselines in larger real-world datasets especially Weather, Traffic, and Electricity.** The three highlight datasets have many more number of time series, thus the results would be more stable and less susceptible to overfitting than other smaller datasets. Experiments show that compared to other datasets, C2Linear achieve higher performance in all forecasting steps ($T \in \{96, 192, 336, 720\}$) on them. Specifically, as the time series number increases, benefiting from the contrastive regularization term, C2Linear can capture inter-series correlations accurately, leading to higher representations and more improvements.

In summary, these results reveal that C2Linear captures inter-series correlations via contrastive regularization and shows the effectiveness and robustness on different datasets from real-world when compared with existing state-of-the-art methods.

Efficiency results. According to Table 3, as C2Linear is a linear model with four linear layers at most (*C2DLinear*), it costs much lower memory and fewer parameters and has a faster inference speed than existing Transformers. Although compared to LTSF-Linear, C2Linear occupies a little more memory and computing operations, the resources they use are in the same range, and C2Linear consistently outperforms exsiting SOTA solutions, so it is acceptable in practice.

[1] https://github.com/zhouhaoyi/ETDataset
[2] http://pems.dot.ca.gov
[3] https://archive.ics.uci.edu/ml/datasets/ElectricityLoadDiagrams20112014
[4] https://github.com/laiguokun/multivariate-time-series-data
[5] https://www.bgc-jena.mpg.de/wetter/
[6] https://gis.cdc.gov/grasp/fluview/fluportaldashboard.html

Datasets	ETTh1&ETTh2	ETTm1 &ETTm2	Traffic	Electricity	Exchange-Rate	Weather	ILI
Variates	7	7	862	321	8	21	7
Timesteps	17,420	69,680	17,544	26,304	7,588	52,696	966
Granularity	1hour	5min	1hour	1hour	1day	10min	1week

Table 1. The statistics of the nine popular datasets for the LTSF problem.

Methods		C2Linear*		Linear		NLinear		DLinear		FEDformer		Autoformer		Informer		Pyraformer		*Repeat*	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Electricity	96	**0.139**$_1$	**0.232**$_1$	0.140	0.237	0.141	0.237	0.140	0.237	$\underline{0.193}$	$\underline{0.308}$	0.201	0.317	0.274	0.368	0.386	0.449	1.588	0.946
	192	**0.152**$_1$	0.249$_1$	0.153	0.250	0.154	**0.248**	0.153	0.249	$\underline{0.201}$	$\underline{0.315}$	0.222	0.334	0.296	0.386	0.386	0.443	1.595	0.950
	336	**0.168**$_1$	0.267$_1$	0.169	0.268	0.171	**0.265**	0.169	0.267	$\underline{0.214}$	$\underline{0.329}$	0.231	0.338	0.300	0.394	0.378	0.443	1.617	0.961
	720	**0.203**$_1$	0.300$_3$	**0.203**	0.301	0.210	**0.297**	**0.203**	0.301	$\underline{0.246}$	$\underline{0.355}$	0.254	0.361	0.373	0.439	0.376	0.445	1.647	0.975
Exchange	96	0.091$_3$	0.214$_3$	0.082	0.207	0.089	0.208	**0.081**	0.203	$\underline{0.148}$	$\underline{0.278}$	0.197	0.323	0.847	0.752	0.376	1.105	**0.081**	**0.196**
	192	**0.152**$_3$	**0.285**$_3$	0.167	0.304	0.180	0.300	0.157	0.293	$\underline{0.271}$	$\underline{0.380}$	0.300	0.369	1.204	0.895	1.748	1.151	0.167	0.289
	336	**0.289**$_3$	0.407$_3$	0.328	0.432	0.331	0.415	0.305	0.414	$\underline{0.460}$	$\underline{0.500}$	0.509	0.524	1.672	1.036	1.874	1.172	0.305	**0.396**
	720	0.846$_3$	0.697$_3$	0.964	0.750	1.033	0.780	**0.643**	**0.601**	$\underline{1.195}$	$\underline{0.841}$	1.447	0.941	2.478	1.310	1.943	1.206	0.823	0.681
Traffic	96	**0.402**$_2$	0.284$_2$	0.410	0.282	0.410	**0.279**	0.410	0.282	$\underline{0.587}$	$\underline{0.366}$	0.613	0.388	0.719	0.391	2.085	0.468	2.723	1.079
	192	**0.417**$_2$	0.292$_2$	0.423	0.287	0.423	**0.284**	0.423	0.287	$\underline{0.604}$	$\underline{0.373}$	0.616	0.382	0.696	0.379	0.867	0.467	2.756	1.087
	336	**0.433**$_2$	0.306$_2$	0.436	0.295	0.435	**0.290**	0.436	0.296	$\underline{0.621}$	0.383	0.622	$\underline{0.337}$	0.777	0.420	0.869	0.469	2.791	1.095
	720	**0.464**$_2$	0.310$_2$	0.466	0.315	**0.464**	**0.307**	0.466	0.315	$\underline{0.626}$	$\underline{0.382}$	0.660	0.408	0.864	0.472	0.881	0.473	2.811	1.097
Weather	96	**0.172**$_1$	**0.231**$_1$	0.176	0.236	0.182	0.232	0.176	0.237	$\underline{0.217}$	$\underline{0.296}$	0.266	0.336	0.300	0.384	0.896	0.556	0.259	0.254
	192	**0.215**$_1$	0.272$_1$	0.218	0.276	0.225	**0.269**	0.220	0.282	$\underline{0.276}$	$\underline{0.336}$	0.307	0.367	0.598	0.544	0.622	0.624	0.309	0.292
	336	**0.262**$_1$	0.314$_1$	**0.262**	0.312	0.271	**0.301**	0.265	0.319	$\underline{0.339}$	$\underline{0.380}$	0.359	0.395	0.578	0.523	0.739	0.753	0.377	0.338
	720	**0.322**$_1$	0.363$_1$	0.326	0.365	0.338	**0.348**	0.323	0.362	$\underline{0.403}$	$\underline{0.428}$	0.419	0.428	1.059	0.741	1.004	0.934	0.465	0.394
ILI	24	1.728$_2$	0.889$_2$	1.947	0.985	**1.683**	**0.858**	2.215	1.081	$\underline{3.228}$	$\underline{1.260}$	3.483	1.287	5.764	1.677	1.420	2.012	6.587	1.701
	36	2.008$_2$	0.983$_2$	2.182	1.036	**1.703**	**0.859**	1.963	0.963	$\underline{2.679}$	$\underline{1.080}$	3.103	1.148	4.755	1.467	7.394	2.031	7.130	1.884
	48	**1.689**$_2$	**0.877**$_2$	2.256	1.060	1.719	0.884	2.130	1.024	$\underline{2.622}$	$\underline{1.078}$	2.669	1.085	4.763	1.469	7.551	2.057	6.575	1.798
	60	**1.816**$_2$	**0.910**$_2$	2.390	1.104	1.819	0.917	2.368	1.096	2.857	1.157	$\underline{2.770}$	$\underline{1.125}$	5.264	1.564	7.662	2.100	5.893	1.677
ETTh1	96	**0.370**$_2$	**0.392**$_2$	0.375	0.397	0.374	0.394	0.375	0.399	$\underline{0.376}$	$\underline{0.419}$	0.449	0.459	0.865	0.713	0.664	0.612	1.295	0.713
	192	0.409$_2$	0.421$_2$	0.418	0.429	0.408	**0.415**	**0.405**	0.416	$\underline{0.420}$	$\underline{0.448}$	0.500	0.482	1.008	0.792	0.790	0.681	1.325	0.733
	336	**0.424**$_2$	0.430$_2$	0.479	0.476	0.429	**0.427**	0.439	0.443	$\underline{0.459}$	$\underline{0.465}$	0.521	0.496	1.107	0.809	0.891	0.738	1.323	0.744
	720	**0.435**$_2$	0.454$_2$	0.624	0.592	0.440	**0.453**	0.472	0.490	$\underline{0.506}$	$\underline{0.507}$	0.514	0.512	1.181	0.865	0.963	0.782	1.339	0.756
ETTh2	96	**0.272**$_2$	**0.338**$_2$	0.288	0.352	0.277	**0.338**	0.289	0.353	$\underline{0.346}$	$\underline{0.388}$	0.358	0.397	3.755	1.525	0.645	0.597	0.432	0.422
	192	0.347$_2$	0.379$_2$	0.377	0.413	**0.344**	**0.381**	0.383	0.418	$\underline{0.429}$	$\underline{0.439}$	0.456	0.452	5.602	1.931	0.788	0.683	0.534	0.473
	336	**0.352**$_2$	**0.398**$_2$	0.452	0.461	0.357	0.400	0.448	0.465	$\underline{0.496}$	$\underline{0.487}$	0.482	0.486	4.721	1.835	0.907	0.747	0.591	0.508
	720	**0.389**$_2$	**0.428**$_2$	0.698	0.595	0.394	0.436	0.605	0.551	$\underline{0.463}$	$\underline{0.474}$	0.515	0.511	3.647	1.625	0.963	0.783	0.588	0.517
ETTm1	96	0.301$_3$	0.347$_3$	0.308	0.352	0.306	0.348	**0.299**	**0.343**	$\underline{0.379}$	$\underline{0.419}$	0.505	0.475	0.672	0.571	0.543	0.510	1.214	0.665
	192	0.338$_3$	**0.360**$_3$	0.340	0.369	0.349	0.375	**0.335**	0.365	$\underline{0.426}$	$\underline{0.441}$	0.553	0.496	0.795	0.669	0.557	0.537	1.261	0.690
	336	**0.361**$_3$	**0.380**$_3$	0.376	0.393	0.375	0.388	0.369	0.386	$\underline{0.445}$	$\underline{0.459}$	0.621	0.537	1.212	0.871	0.754	0.655	1.283	0.707
	720	**0.419**$_3$	**0.414**$_3$	0.440	0.435	0.433	0.422	0.425	0.421	$\underline{0.543}$	$\underline{0.490}$	0.671	0.561	1.166	0.823	0.908	0.724	1.319	0.729
ETTm2	96	**0.164**$_2$	**0.253**$_2$	0.168	0.262	0.167	0.255	0.167	0.260	$\underline{0.203}$	$\underline{0.287}$	0.255	0.339	0.365	0.453	0.435	0.507	0.266	0.328
	192	**0.219**$_2$	**0.290**$_2$	0.232	0.308	0.221	0.293	0.224	0.303	$\underline{0.269}$	$\underline{0.328}$	0.281	0.340	0.533	0.563	0.730	0.673	0.340	0.371
	336	0.278$_2$	0.332$_2$	0.320	0.373	**0.274**	**0.327**	0.281	0.342	$\underline{0.325}$	$\underline{0.366}$	0.339	0.372	1.363	0.887	1.201	0.845	0.412	0.410
	720	**0.367**$_2$	**0.383**$_2$	0.413	0.435	0.368	0.384	0.397	0.421	$\underline{0.421}$	$\underline{0.415}$	0.433	0.432	3.379	1.338	3.625	1.451	0.521	0.465

- Methods* are implemented by us; Other results are from LSTM-Linear (Zeng et al., 2023).
- The result for C2Linear is recorded as the best among the set of aforementioned linear models, and the subscripts 1, 2, and 3 represent *C2Linear*, *CN2Linear* and *CD2Linear* model respectively.

Table 2. Multivariate long-term forecasting errors in terms of MSE and MAE, the lower the better. Among them, ILI dataset is with forecasting horizon $T \in \{24, 36, 48, 60\}$. For the others, $T \in \{96, 192, 336, 720\}$. *Repeat* repeats the last value in the look-back window. The **best results** are highlighted in **bold** and the best results of Transformers are highlighted with a underline.

6. Conclusion and Future Work

Conclusion. In this paper, we propose a novel contrastive learning paradigm for multivariate time series forecasting, namely C2Linear, to capture inter-series correlations via contrastive regularization combining variance and covariance constraints, which explores the mutual information between various series. C2Linear outperforms existing approaches in a variety of real-world datasets.

Future work. Future works are considered in two directions. First, we will investigate method that combines contrastive learning and graph convolutional networks to depict the correlations among various time series more accurately. Second, we will look for its application to more latest and real-world scenarios, such as COVID-19 cases forecasting and stock price analysis.

References

Bardes, A., Ponce, J., and LeCun, Y. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022.

Cao, D., Wang, Y., Duan, J., Zhang, C., Zhu, X., Huang, C., Tong, Y., Xu, B., Bai, J., Tong, J., and Zhang, Q. Spectral temporal graph neural network for multivariate time-series forecasting. In *Proceedings of the*

Method	MACs	Parameter	Time	Memory
C2Linear	0.21G	319.1K	0.7ms	2115MiB
DLinear	**0.04G**	**139.7K**	**0.4ms**	**687MiB**
Transformer×	4.03G	13.61M	26.8ms	6091MiB
Informer	3.93G	14.39M	49.3ms	3869MiB
Autoformer	4.41G	14.91M	164.1ms	7607MiB
Pyraformer	0.80G	241.4M*	3.4ms	7017MiB
FEDformer	4.41G	20.68M	40.5ms	4143MiB

- × is modified into the same one-step decoder, which is implemented in Autoformer.
- * 236.7M parameters of Pyraformer come from its linear decoder.

Table 3. Comparison of practical efficiency of LTSF-Transformers under L=96 and T=720 on the **Traffic**. MACs are the number of multiply-accumulate operations. The inference time averages 5 runs.

34th International Conference on Neural Information Processing Systems, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. E. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

He, H., Zhang, Q., Bai, S., Yi, K., and Niu, Z. Catn: Cross attentive tree-aware network for multivariate time series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(4):4030–4038, Jun. 2022.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

Lai, G., Chang, W.-C., Yang, Y., and Liu, H. Modeling long- and short-term temporal patterns with deep neural networks. In *The 41st International ACM SIGIR Conference on Research amp; Development in Information Retrieval*, SIGIR '18, pp. 95–104, New York, NY, USA, 2018.

Li, S., Jin, X., Xuan, Y., Zhou, X., Chen, W., Wang, Y.-X., and Yan, X. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

LIU, M., Zeng, A., Chen, M., Xu, Z., LAI, Q., Ma, L., and Xu, Q. SCINet: Time series modeling and forecasting with sample convolution and interaction. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022.

Liu, S., Yu, H., Liao, C., Li, J., Lin, W., Liu, A. X., and Dustdar, S. Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *International Conference on Learning Representations*, 2022.

Liu, X., Zhang, F., Hou, Z., Mian, L., Wang, Z., Zhang, J., and Tang, J. Self-supervised learning: Generative or contrastive. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):857–876, 2023. doi: 10.1109/TKDE.2021.3090866.

Wu, H., Xu, J., Wang, J., and Long, M. Autoformer: Decomposition transformers with Auto-Correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, 2021.

Yu, H., Li, T., Yu, W., Li, J., Huang, Y., Wang, L., and Liu, A. Regularized graph structure learning with semantic knowledge for multi-variates time-series forecasting. In Raedt, L. D. (ed.), *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pp. 2362–2368. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track.

Zeng, A., Chen, M., Zhang, L., and Xu, Q. Are transformers effective for time series forecasting? 2023.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., and Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence*, volume 35, pp. 11106–11115, 2021.

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., and Jin, R. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *Proc. 39th International Conference on Machine Learning*, 2022.