
SURVEYS AND CROSSOVERS

This section provides surveys of the literature in investment management or short papers exemplifying advances in finance that arise from the confluence with other fields. This section acknowledges current trends in technology, and the cross-disciplinary nature of the investment management business, while directing the reader to interesting and important recent work.

A CASE STUDY IN AI ASSISTED RESEARCH: FORECASTING FINANCIAL VARIABLES WITH A FOUNDATION MODEL*

Sanjiv R. Das^{a,b}, Tarang Goyal^{a,c} and Mohini Yadav^{a,d}

Using Chronos-2, an open-source time-series foundation model, we evaluate pretrained time-series models for economic and financial forecasting with an emphasis on whether multivariate (MV) inputs improve accuracy relative to univariate (UV) baselines. This paper was developed using AI in multiple steps, described in the Appendix. The study covers two panels—7 tech equities prices and U.S. Treasury interest rates—as well as a combined panel, using rolling monthly evaluations for periods in the range 2000–2025. We vary input window lengths and forecast horizons and report RMSE and MAPE. For the Chronos-2 model, MV forecasts consistently outperform UV forecasts, with especially strong gains for interest rates and meaningful improvements for equities. Series-level comparisons show MV improvements in every case, and error dispersion is generally lower under MV inputs. We also provide parameter-heatmap and time-series visualizations. However, mixing time series across equity and interest rate markets reduces forecast accuracy, indicating that adding noisy context degrades model performance. The benefits are strongest when related series are modeled jointly under disciplined rolling protocols. This case study showcases how AI may be used for financial research.



^aSanta Clara University

^bE-mail: srdas@scu.edu

^cE-mail: taranggoyal2000@gmail.com

^dE-mail: monaydl3@gmail.com

*Das is a professor of Finance and Data Science, and Goyal and Yadav are graduate students in the Leavey School of Business at Santa Clara University.

1 Introduction

Time-series forecasting in finance remains a contested area for modern machine learning. While deep learning has transformed natural language processing and computer vision, its advantage in economic and financial forecasting is far less

clear. Recent comparative studies of U.S. Treasury yield curve forecasting report that classical econometric baselines and simple benchmarks can outperform many modern machine-learning models across long spans of daily data, with only a few deep learning approaches delivering competitive gains. These findings underscore an open question for practitioners: when, and under what conditions, do multivariate forecasting methods provide reliable improvements over univariate forecasting? Adding data for a forecast brings up the standard question of whether it introduces signal or noise.

Standard econometric models do not scale well as the number of joint series used for forecasting increases, even beyond two or three time series. As we will show here, this is not a constraint for transformer-based time series foundation models (TSFMs). The emergence of foundation models—large, self-supervised models trained on broad data and adapted to downstream tasks—has begun to reshape time-series forecasting. The promise of these models is not only scale, but also transfer: a single model can exploit shared structure across diverse series and potentially improve predictive accuracy in domains where data are limited or noisy. Yet evidence from forecasting competitions and applied finance (Singh *et al.*, 2025) suggests that pure deep-learning solutions often struggle unless they are hybridized or carefully engineered.

This paper evaluates whether foundation-model forecasting can deliver practical improvements in economic and financial settings, focusing on Chronos-2, an open-source time-series foundation model (GitHub: <https://github.com/amazon-science/chronos-forecasting>). Just as large language models generate a sequence of words, Chronos-2 is trained on thousands of time series, and generates forward sequences of data based on multivariate histories. Our central

hypothesis is that multivariate (MV) inputs, when properly chosen, aligned, and modeled, can help the model extract cross-series information and outperform univariate (UV) baselines. We test this hypothesis across two economically meaningful panels: the 7 tech equities prices and a set of U.S. Treasury interest rates, and we also study a combined panel to assess whether joint modeling improves results.

The contribution of the paper is as follows. First, we provide a systematic comparison of MV versus UV forecasting across equities and rates with a consistent rolling evaluation protocol. We find that the MV approach beats the UV one. Second, we examine whether MV gains differ by asset class and whether joint stock and interest rates modeling improves accuracy. In this case, we find that adding a different class of time series does not improve forecasting, highlighting that expanding the basis set for modeling does not always improve forecast accuracy. Indeed, there is a signal to noise tradeoff that applies here. Third, we provide in Appendix A the process by which AI was not only the subject of the paper, but was also used to produce the paper. Fourth, an additional goal of this paper is to introduce time series foundation models (TSFMs) to the readership of the journal. By providing all the implementation code in GitHub (see the Appendix), we show how easy it is to undertake multivariate forecasting via inference from a foundation model like Chronos-2 without having to do any model fitting.

1.1 Brief literature review

The literature on financial time-series forecasting spans classical econometrics, machine learning, and recent deep-learning approaches. Standard time-series tools and statistical benchmarks remain central in practice and are documented in foundational texts (Shumway and Stoffer, 2025). In fixed-income applications,

yield curve forecasting has been studied using no-arbitrage and factor-augmented approaches (Moench, 2006) as well as forecast combinations (Caldeira *et al.*, 2013). Machine-learning methods have been applied to yield curves with mixed results, including statistical ML approaches (Sambasivan and Das, 2017) and more recent comparative studies that find strong performance for classical econometric baselines relative to many ML and deep-learning models (Singh *et al.*, 2025). Related work on bond risk premia also highlights the potential and limitations of ML for term-structure prediction (Bianchi *et al.*, 2020; Hoogteijling *et al.*, 2021).

For equities and broader financial markets, surveys emphasize the breadth of deep-learning adoption but also the variability of performance across datasets and tasks (Sezer *et al.*, 2019). Empirical studies comparing ARIMA, LSTM, and Prophet illustrate that model choice can be highly data-dependent, with traditional statistical methods remaining competitive in many settings (Sunki *et al.*, 2024). The broader financial machine-learning literature argues for careful feature engineering, rigorous validation, and robust benchmarks to avoid overfitting and spurious gains (López de Prado, 2018). These themes motivate careful UV baselines and out-of-sample evaluation in this paper.

Recent advances in deep-learning forecasting include specialized architectures such as NHITS (Challu *et al.*, 2023) and, more recently, foundation models for time series. A recent review is available in Zhang and Zhang (2026). Chronos is a large, self-supervised forecasting model trained on diverse datasets and designed for transfer across domains (Ansari *et al.*, 2024). This paper contributes to the growing evidence on foundation models by testing whether MV inputs can systematically improve forecasting accuracy in economically important panels of equities and

rates, and by reporting results using consistent rolling protocols and multiple error metrics. In this paper we focus on the newest generation in this class of models, namely Chronos-2 (Ansari *et al.*, 2025).

2 Methodology

This section develops the research questions and the experimental framework including the Chronos-2 forecasting pipeline. We describe the data scope, model inputs and horizons, rolling evaluation design, and the artifacts produced for analysis and downstream use.

2.1 Research questions

The study addresses three related questions for the Chronos-2 model. First, we test whether multivariate (MV) forecasting yields more accurate predictions than univariate (UV) methods. By restricting attention within a single model, Chronos-2, we are able to focus specifically on this effect in the class of a foundation model. This validates the value of pretraining the model on thousands of time series. Second, we examine whether MV gains differ across asset classes by comparing stocks to interest rates. Third, we evaluate whether forecasting stocks and rates jointly improves accuracy relative to modeling each group separately.

2.2 Overview of the Chronos-2 model

Chronos-2 (Ansari *et al.*, 2025) extends the foundation-model approach to time-series forecasting by moving beyond purely univariate inputs toward a universal, zero-shot framework that can handle multivariate groups and covariate-informed tasks. The motivation is pragmatic: real forecasting problems often involve co-evolving series or external drivers, yet most pretrained models are limited to single-series histories. Chronos-2 addresses this gap by enabling

joint inference across related series without task-specific retraining, simplifying deployment and improving transfer to new domains.

Chronos-2 is just like a LLM for time-series sequences. Just as you provide a seed sequence of text (a prompt) to an LLM and it generates a textual continuation (the response), so it is with a TSFM like Chronos2—you provide a time series to the pretrained model and it generates a continuation, i.e., the forecast of the length you prespecify. As a visual example, see Figure 1.

Architecturally, Chronos-2 introduces a group-attention mechanism that supports in-context learning across sets of series. A “group” can represent variates of a multivariate series, multiple related targets, or targets plus covariates. The attention design enables efficient information sharing and allows the model to learn cross-series structure at inference time, which is crucial for multivariate forecasting and covariate conditioning.

To train these capabilities, Chronos-2 is built on synthetic multivariate constructions derived from univariate time series, enforcing diverse dependence structures during pretraining. Evaluation

is reported on multivariate and covariate-aware benchmarks such as fev-bench, GIFT-Eval, and Chronos Benchmark II, where Chronos-2 attains state-of-the-art performance, particularly on covariate-conditioned tasks. These design choices align with our use case: leveraging joint structure across equities and rates and assessing whether MV inputs yield systematic gains over UV baselines.

The methodology behind Chronos-2 centers on transforming time series forecasting into a universal, zero-shot task through a specialized transformer architecture and innovative training strategies. Unlike its predecessor model, which was primarily univariate, Chronos-2 leverages in-context learning (ICL) to handle multivariate data and covariates without needing task-specific fine-tuning. Many specific techniques are deployed in pre-training this model (see Figure 1 in the Chronos-2 technical report).

- (1) Tokenization and Robust Scaling. The pipeline begins by normalizing input data using a robust scaling scheme. This involves standardizing the series and applying a $\sinh^{-1}$ transformation to stabilize variance and reduce the impact of extreme outliers.

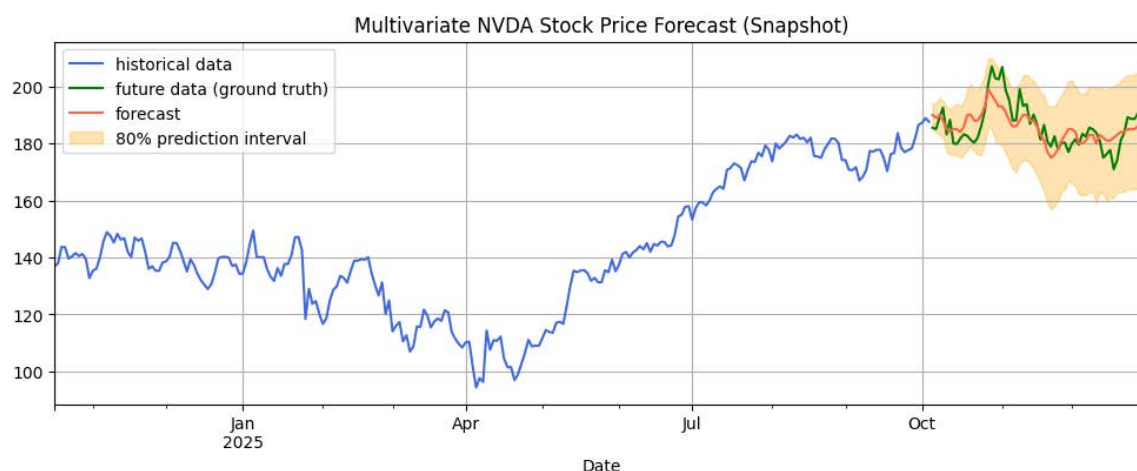


Figure 1 Forecast for NVDA. The input series is provided and the out-of-sample forecast is generated with error bounds. Here the forecast is plotted against the true outcome as well.

- Categorical covariates are converted to real-valued representations using target or ordinal encoding. After scaling, the data is augmented with meta features, including a relative time index and a binary mask that indicates observed versus missing values.
- (2) **Patch-Based Representation.** To process the data efficiently, Chronos-2 uses patching, where the continuous time series and its meta features are divided into non-overlapping segments of length P . These patches are mapped into high-dimensional embeddings via a residual network. A unique REG token is inserted between historical context and future patches to serve as an attention sink and separator.
 - (3) **Core Architecture: Time and Group Attention.** The model is an encoder-only transformer that alternates between two critical attention layers: (a) **Time Attention:** This layer aggregates information across temporal patches within a single time series, utilizing Rotary Position Embeddings (RoPE) to maintain temporal order. (b) **Group Attention:** This is the core innovation of Chronos-2. It allows the model to share information across different time series within a “group” at each patch index. A group can flexibly represent variates of a multivariate series, a target with its covariates, or even a collection of related univariate series for cross-learning.
 - (4) **Multivariate and Covariate Integration.** Chronos-2 manages different task types—univariate, multivariate, and covariate-informed—within a unified framework by assigning Group IDs. For multivariate tasks, all related variates share an ID, enabling the group attention layer to model their dependencies. When covariates are present, they are included in the group, and known future values are provided in the future input matrix W , allowing the model to condition its forecasts on them.
 - (5) **Training and Quantile Forecasting.** The model is trained in two stages to handle varying context lengths, starting at 2048 steps and extending to 8192. It uses a quantile regression objective to produce probabilistic forecasts across 21 quantiles (from 0.01 to 0.99), providing a comprehensive view of uncertainty and rare event risks. Remarkably, its advanced ICL capabilities are largely learned from synthetic datasets that impose complex multivariate structures on univariate base series.

In sum, the methodology of Chronos-2 involves an encoder-only transformer architecture that utilizes a novel group attention mechanism and robust scaling to perform zero-shot univariate, multivariate, and covariate-informed forecasting by sharing information across related time series in-context.

2.3 Implementation

The data consist of three panels: 7 tech stocks’ prices (AAPL, AMZN, GOOGL, MSFT, NFLX, NVDA, TSLA) ($K = 7$), Treasury interest rates for ten different maturities ($K = 10$), and a combined panel of all series ($K = 17$). This design supports both within-group and cross-group forecasting comparisons.

We use input window lengths $n \in \{126, 252, 504, 756\}$ trading days, and forecast horizons $m \in \{21, 63\}$ trading days to cover short to medium histories (0.5 to 3 years) and near- to mid-term prediction horizons (1 to 3 months).

The evaluation period spans 2000–2025 with monthly rolling forecasts to reflect realistic re-estimation and deployment cadence. Forecast accuracy is measured using RMSE (root mean squared error) and MAPE (mean absolute percentage error) to capture both scale-dependent and scale-free error behavior. Some of the

analyses will use the period from 2010–2025 as across stock prices and interest rates, this is the period for which there is no missing data. Given how difficult it is to forecast stock returns, we restrict the analysis to forecasting stock price series.

To assess temporal robustness, we compare performance across pre-2023 and post-2023 regimes using a training cutoff that separates the sample. We will also argue that this is an informal check on information leakage in the model.

3 Results

3.1 Forecasts within asset classes

One of the goals of the paper is to examine the extent of improvement when multivariate forecasting (MV) is used instead of univariate

Table 1 Average performance by dataset and mode.

Dataset	Mode	MAPE		RMSE	
		Mean	Std	Mean	Std
Rates	MV	0.0497	0.1673	0.0418	0.0445
Rates	UV	0.1233	0.2213	0.1865	0.1783
Stocks	MV	0.0706	0.0728	3.9619	9.0440
Stocks	UV	0.0844	0.0834	5.0395	11.5710

Lower mean error across all experiments (for values of input length $n = \{126, 252, 504, 756\}$, and forecast period $m = \{21, 63\}$) measured by MAPE and RMSE is better. Data for the rolling periods are from 2000 through 2025. Prediction periods begin 3 years after the start of the dataset.

forecasting (UV). Table 1 summarizes average performance by dataset and mode. Results are aggregated across all tickers, and in-sample and out-sample combinations. MV forecasts outperform UV for both rates and stocks, with large

Table 2 UV vs. MV comparison by series.

Dataset	Series	MAPE (MV)	MAPE (UV)	RMSE (MV)	RMSE (UV)	MAPE Imp.	RMSE Imp.
Rates	DGS3MO	0.2355	0.3186	0.0618	0.1227	0.0831	0.0609
Rates	DGS6MO	0.0833	0.1770	0.0455	0.1257	0.0937	0.0802
Rates	DGS1	0.0572	0.1326	0.0454	0.1424	0.0754	0.0970
Rates	DGS2	0.0334	0.1270	0.0413	0.1845	0.0936	0.1432
Rates	DGS3	0.0243	0.1181	0.0388	0.2068	0.0938	0.1680
Rates	DGS5	0.0171	0.0972	0.0359	0.2232	0.0801	0.1873
Rates	DGS7	0.0134	0.0827	0.0345	0.2278	0.0693	0.1933
Rates	DGS10	0.0114	0.0703	0.0344	0.2184	0.0589	0.1840
Rates	DGS20	0.0099	0.0572	0.0364	0.2120	0.0473	0.1756
Rates	DGS30	0.0113	0.0520	0.0442	0.2012	0.0407	0.1570
Stocks	AAPL	0.0599	0.0720	3.3016	4.1563	0.0121	0.8547
Stocks	AMZN	0.0567	0.0707	3.0825	4.0517	0.0140	0.9692
Stocks	GOOGL	0.0478	0.0635	3.1803	4.0070	0.0157	0.8267
Stocks	MSFT	0.0368	0.0466	4.2003	6.1519	0.0098	1.9516
Stocks	NFLX	0.1040	0.1145	1.7871	2.0192	0.0105	0.2321
Stocks	NVDA	0.0906	0.1107	1.5994	1.8420	0.0201	0.2426
Stocks	TSLA	0.1098	0.1242	13.6796	16.7870	0.0144	3.1074

Notes: Lower error values indicate better performance. Improvements are mean UV error minus mean MV error. Results are averaged across values of input length $n = \{126, 252, 504, 756\}$, and forecast period $m = \{21, 63\}$. Data for the rolling periods are from 2000 through 2025. Predictions periods begin 3 years after the start of the dataset.

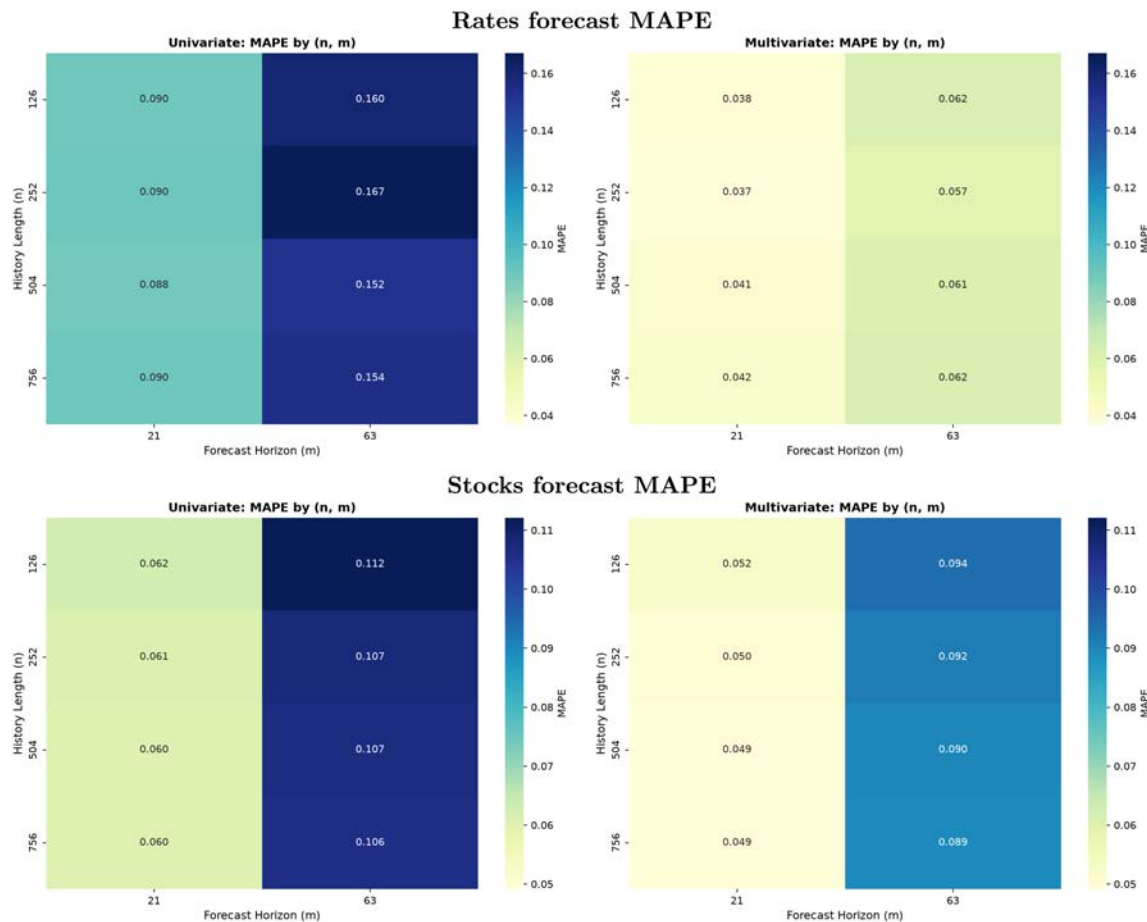


Figure 2 Parameter heatmap comparing MV and UV forecasting accuracy across window lengths and horizons (based on MAPE). The upper plot is for Rates and the lower one is for Stocks. In each plot, the left side is for the 21 working day forecast (1 month forecast) and the right side is for the 63 working day forecast (3 month forecast).

MAPE and RMSE reductions in rates and lesser gains in stocks. The standard deviation of the forecast error is also smaller for MV versus UV forecasts. Overall, the results suggest that multivariate models are easily implemented in Chronos-2 and provide robust forecasts.

Table 2 reports UV vs. MV results by series. MV improves both MAPE and RMSE for every rate and stock. This is a clear win for the MV forecasting model, suggesting that the information from additional related time series adds value when used in the realm of a pre-trained foundation time series model.

Figure 2 shows the breakdown of errors by the length n of the input time series, and the length m of the forecast series. For both rates and stocks, the forecast error is smaller for the MV approach than for the UV approach. As expected the forecast error is greater for the 63 day forecast than for the 21 day one. But the input series length n does not seem to matter, accuracy levels are not that different across input length, which varies from half a year to three years.

The time series of MAPE UV and MV errors is shown in Figure 3. Across all experimental conditions, It is clear that the errors are greater for the

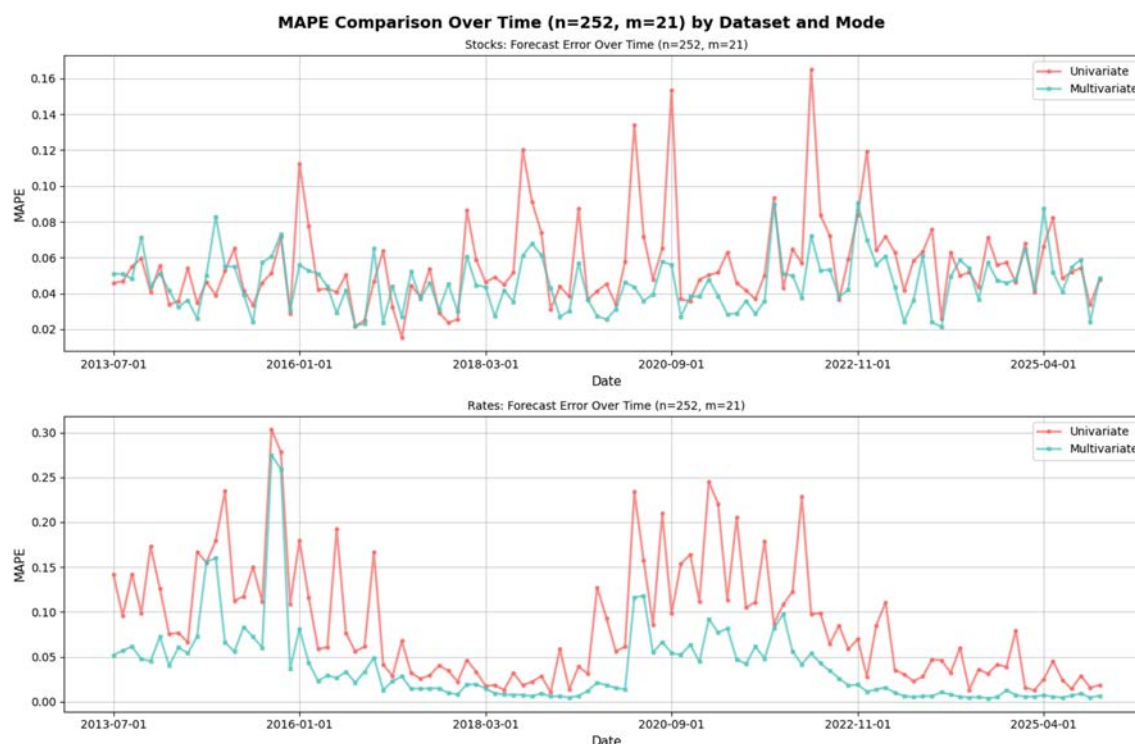


Figure 3 Time series of MV and UV MAPE errors over the evaluation period. This is for the dataset running from July 2010 through December 2025.

UV series for both rates and stocks, with this being more pronounced for interest rates. In both series the errors seem to be higher when the rates or stocks are at higher levels, even though these are percentage errors. Overall, Figures 2 and 3 show that the MV forecasts are more accurate than the UV ones.

3.2 Forecasts with combined asset classes

In this section we consider forecasts using data from both the Rates and Stocks datasets combined. This so-called small “world” model is assessed to see how cross-domain information may be used to improve forecasts. To be more specific, each rate and stock times series is now forecast using the combined information of rates and stock data. This assesses whether there is additional information that may be exploited across the two families of time series.

The combination of both datasets is only undertaken for the period July 2010 through December 2025, as all stock series are not consistently available through this entire period. Results are shown in Table 3. A comparison of the top half of the table with the bottom half shows that the additional data series from another market does not improve forecasts. If anything, this seems to slightly degrade the forecast quality, possibly adding noise instead of additional signal. In the realm of foundation models we can think of this as adding context that distracts the model from the domain in which the forecast is being made.

3.3 Leakage

The approach taken in this analysis is susceptible to leakage criticisms, i.e., that the forecasts are already in the pre-training data for the model.

Table 3 Combined dataset forecasts. UV vs. MV comparison by series.

Dataset	Series	MAPE (MV)	MAPE (UV)	RMSE (MV)	RMSE (UV)	MAPE Imp.	RMSE Imp.
Rates	DGS3MO	0.2304	0.2849	0.0597	0.1052	0.0545	0.0455
Rates	DGS6MO	0.0937	0.1714	0.0500	0.1179	0.0777	0.0679
Rates	DGS1	0.0609	0.1289	0.0475	0.1363	0.0680	0.0888
Rates	DGS2	0.0366	0.1216	0.0426	0.1727	0.0850	0.1301
Rates	DGS3	0.0257	0.1111	0.0378	0.1883	0.0854	0.1505
Rates	DGS5	0.0185	0.0959	0.0341	0.2025	0.0774	0.1684
Rates	DGS7	0.0144	0.0843	0.0321	0.2086	0.0699	0.1765
Rates	DGS10	0.0128	0.0759	0.0325	0.2045	0.0631	0.1720
Rates	DGS20	0.0116	0.0628	0.0353	0.1993	0.0512	0.1640
Rates	DGS30	0.0122	0.0573	0.0396	0.1923	0.0451	0.1527
Stocks	AAPL	0.0500	0.0621	5.6015	7.1124	0.0121	1.5109
Stocks	AMZN	0.0462	0.0617	5.2798	7.0294	0.0155	1.7496
Stocks	GOOGL	0.0411	0.0545	4.8211	6.0685	0.0134	1.2474
Stocks	MSFT	0.0305	0.0453	6.6967	10.1334	0.0148	3.4367
Stocks	NFLX	0.0792	0.0902	3.0946	3.4715	0.0110	0.3769
Stocks	NVDA	0.0743	0.0980	2.7598	3.3253	0.0237	0.5655
Stocks	TSLA	0.1090	0.1258	16.6789	20.7258	0.0168	4.0469
Combined	DGS3MO	0.2392	0.2849	0.0655	0.1052	0.0457	0.0397
Combined	DGS6MO	0.1002	0.1714	0.0543	0.1179	0.0712	0.0636
Combined	DGS1	0.0671	0.1289	0.0518	0.1363	0.0618	0.0845
Combined	DGS2	0.0392	0.1216	0.0440	0.1727	0.0824	0.1287
Combined	DGS3	0.0256	0.1111	0.0374	0.1883	0.0855	0.1509
Combined	DGS5	0.0189	0.0959	0.0346	0.2025	0.0770	0.1679
Combined	DGS7	0.0145	0.0843	0.0329	0.2086	0.0698	0.1757
Combined	DGS10	0.0126	0.0759	0.0327	0.2045	0.0633	0.1718
Combined	DGS20	0.0121	0.0628	0.0367	0.1993	0.0507	0.1626
Combined	DGS30	0.0129	0.0573	0.0414	0.1923	0.0444	0.1509
Combined	AAPL	0.0519	0.0621	5.6524	7.1124	0.0102	1.4600
Combined	AMZN	0.0470	0.0617	5.1538	7.0294	0.0147	1.8756
Combined	GOOGL	0.0399	0.0545	4.5781	6.0685	0.0146	1.4904
Combined	MSFT	0.0318	0.0453	6.7621	10.1334	0.0135	3.3713
Combined	NFLX	0.0786	0.0902	3.0877	3.4715	0.0116	0.3838
Combined	NVDA	0.0761	0.0980	2.8012	3.3253	0.0219	0.5241
Combined	TSLA	0.1114	0.1258	16.8843	20.7258	0.0144	3.8415

Notes: Lower error values indicate better performance. Improvements are mean UV error minus mean MV error. Results are averaged across values of input length $n = \{126, 252, 504, 756\}$, and forecast period $m = \{21, 63\}$. Data are for the period July 2010 through December 2025, the period when there is complete data for all stocks and rates. For comparison, we show the results when forecasts are based on the individual datasets for Rates and Stocks, as well as when forecasts are based on the combination of both datasets. Predictions periods begin 3 years after the start of the dataset. (As expected the UV forecast errors do not change).

One reason for using the Chronos class of models is that they were not trained on stock market and interest rate datasets. The development of Chronos-2 (Ansari *et al.*, 2025) emphasizes that for generalist pretrained models, the quality and diversity of training data are often more influential than the specific model architecture. While existing large-scale datasets provide a foundation, they are frequently limited to univariate series. To address this and foster robust in-context learning, Chronos-2 integrates established corpora like GIFT-Eval with a heavy reliance on synthetic data. This is achieved through two distinct methods: the TSI approach (Bahrpeyma *et al.*, 2021), which blends randomized trend, seasonality, and noise components, and the TCM approach (Runge *et al.*, 2023), which uses temporal causal models and autoregression to generate complex, graph-based data patterns. The original Chronos univariate forecasting model (Ansari *et al.*, 2024) uses a large number of datasets, including some macroeconomic data from FRED (Godahewa *et al.*, 2021) and some financial data in the datasets used in the Makridakis competition (Makridakis *et al.*, 2020). However, the datasets do not explicitly include any of the series we use for forecasting in this paper. And most certainly,

the training data was not post-2023, so the results from that period do not suffer from leakage.

Nevertheless, if there were large amounts of leakage, because of training on time series that are in this paper or correlated to the ones used in this paper, then (i) we should see little difference between the MV and UV forecasts, and (ii) the pre-2023 forecasts should in fact be more accurate than the post-2023 ones, where the data are not likely to be correlated to or included in the pre-training dataset. First, we see that there is a great difference in the MV and UV forecasts. This is apparent in Tables 1, 2, and 3. With leakage there should be no material difference. Second, we look at the forecast accuracy pre-2023 and post-2023. Contrary to what we would expect with leakage, the post-2023 forecasts are more accurate, even though data from this period could not possibly be in the training data. This is shown for both rates and stocks in Figure 4. This effect is more pronounced for rates than for stocks, where there is no clear difference. These results are unsupported if the model was contaminated by leakage from pre-training data. This is suggestive, though not conclusive, that the results are not impacted by leakage and that foundation models of this type

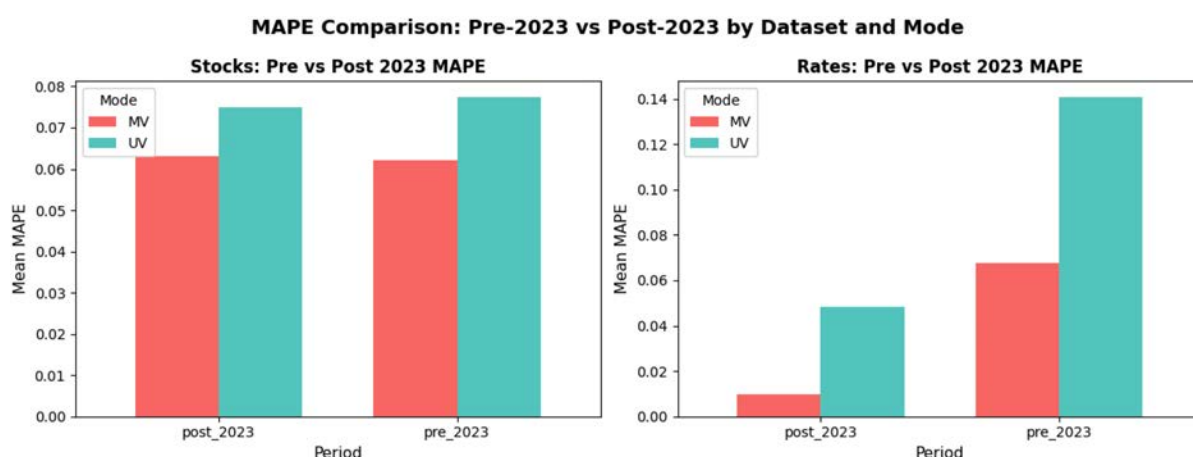


Figure 4 Comparison of forecast accuracy pre-2023 and post-2023. Data used is for the period July 2010 through December 2025.

do better when using multivariate series that come from the same contextual domain as the series that is being forecasted.

3.4 *Limits of the analysis*

While this paper aims to showcase both, AI-assisted research and the use of Chronos-2, it does not aim to be an exhaustive analysis of foundation models. It is merely illustrative as an example of forecasting using pretrained transformers. There are several directions in which a comprehensive analysis can proceed to extend this work in greater detail.

There are several limitations. The study focuses on a single foundation model and a specific set of assets. The evaluation uses fixed window and horizon choices; alternative regimes, higher-frequency data, or different rolling cadences may alter results. While we report standard error metrics, practitioners may, depending on their specific use case, undertake further examination of tail risk, directional accuracy, and economic value to add depth to their analyses. The empirical design is narrow, using a small set of equities and Treasury yields, so generalizability is uncertain. In short, the evidence supports only a model-specific claim about Chronos-2, not a general conclusion about financial forecasting.

The equity results are based on price levels rather than returns, which weakens the interpretation of any forecasting gain for investing or risk management. But given efficient markets, forecasting returns is not theoretically an interesting exercise. The determination of possible data leakage is suggestive, but is useful for all forecasting with foundation pretrained models. The evaluation is limited to statistical error metrics like RMSE and MAPE, and can be extended to other metrics such as directional accuracy, trading performance, Sharpe ratio, or utility gains.

4 **Concluding Discussion**

This crossover article uses AI assistance to create a paper about AI, and all steps in the production of the paper are detailed in the Appendix. It evaluates Chronos-2 as a foundation-model approach to economic and financial forecasting, with a specific focus on whether multivariate inputs improve accuracy relative to univariate baselines. Across equities and U.S. Treasury rates, MV forecasts consistently outperform UV forecasts, with especially strong gains in the rates panel and meaningful improvements in equity forecasts, using MAPE and RMSE metrics. The results indicate that cross-series information from the same domain is valuable and that foundation models can capitalize on it when inputs are properly structured.

The findings also speak to the ongoing debate on the role of modern ML in financial forecasting. Consistent with prior evidence that classical benchmarks remain hard to beat, our results show that gains are not automatic; rather, they arise when models leverage multivariate structure and are evaluated with disciplined rolling protocols. This suggests that the most practical benefit of foundation models may come from their ability to transfer information across related series, rather than from raw model complexity alone. In addition, foundation models can accommodate many input series, which is difficult to do with traditional time-series econometric models.

Future work can extend the framework in three directions. One path is to broaden the universe to a larger “world” panel that includes macroeconomic, commodity, and international series to test transfer at scale. Another is to integrate uncertainty calibration, for example via conformal prediction, to improve the reliability of forecast intervals. Finally, hybrid designs that combine strong econometric baselines with

foundation-model embeddings may offer robust performance while preserving interpretability.

Appendix A AI-Driven Workflow to Generate the Paper

This appendix outlines how the paper was produced with the assistance of Artificial Intelligence.

Detailed Workflow Step-by-Step

Each step below was followed by human review, if AI was involved.

- (1) **Conceptualization:** Students are provided an initial outline.
- (2) **Initial Coding:** Students prepare a Google Colab notebook with assistance from Gemini.
- (3) **Refinement:** The professor updates the notebook, modularizing the code and updating logic, also utilizing Gemini for coding assistance.
- (4) **Data Preparation:** Results are finalized in Colab, consisting of printed dataframes, plots, and textual outputs.
- (5) **Environment Setup:** A $\text{\LaTeX}$ shell for the paper is established in OpenAI Prism (<https://openai.com/prism/>).
- (6) **Outlining:** The structure is set up with blank sections: Introduction, Methodology, Results, and Concluding Discussion (including relevant subsections).
- (7) **Table Conversion:** Printed dataframes from Colab are fed into Gemini with prompts to convert them into $\text{\LaTeX}$ tables. These are then integrated into the Results section.
- (8) **Visuals:** Figures are added to the project folder, and the corresponding $\text{\LaTeX}$ code for figure inclusion is written.
- (9) **Drafting Results:** Prism is tasked with drafting text for the Results section to explain the tables and figures. Notably, the

underlying model (GPT-5.2) inferred results directly from the tables to generate the write-up.

- (10) **Methodology:** The Chronos-1 and Chronos-2 papers, along with experiment descriptions from the Colab notebook, are uploaded to Gemini. Gemini generates the Methodology section, which is then moved to the draft.
- (11) **Specific Subsections:** A subsection on data leakage is added; the reasoning is provided by the authors while the LLM handles the prose.
- (12) **Literature Review (Part 1):** A Bib $\text{\TeX}$ file from Zotero is uploaded. Prism is prompted to generate a literature review based on the citations as they relate to the Methodology and Results.
- (13) **Introduction and Literature Review (Part 2):** Previously published related papers are uploaded. Prism generates the Introduction and updates the literature review for context.
- (14) **Conclusions:** Using the full paper as context, Prism generates the Concluding Discussion.
- (15) **Abstract:** Prism is prompted to synthesize an abstract for the completed draft.
- (16) **Human Editing:** Final human-led editing passes are performed to improve narrative flow, reformat the $\text{\LaTeX}$ layout, and ensure technical accuracy.

Self Review and Revision Process

Gemini was also utilized to generate a mock referee report (`review.pdf`). This report informed a response document (`responses.md`) created with the following prompt:

“Please read the file ‘review.pdf’ which is a referee report on the paper ‘chronos2.pdf’—then suggest changes and improvements to the paper in a document titled ‘response.md’ that outlines the changes in response to cited sections of ‘review.pdf’ as a response note to the review.”

Final edits were made to the manuscript based on these AI-suggested improvements.

The code and files for the paper have been made public on GitHub: <https://github.com/srdas/time-series-fm/tree/main>.

References

- Ansari, A. F., Shchur, O., Küken, J., Auer, A., Han, B., Mercado, P., Rangapuram, S. S., Shen, H., Stella, L., Zhang, X., Goswami, M., Kapoor, S., Maddix, D. C., Gueron, P., Hu, T., Yin, J., Erickson, N., Desai, P. M., Wang, H., Rangwala, H., Karypis, G., Wang, Y., and Bohlke-Schneider, M. (2025). Chronos-2: From Univariate to Universal Forecasting. arXiv:2510.15821 [cs].
- Ansari, A. F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S. S., Arango, S. P., Kapoor, S., Zschiegner, J., Maddix, D. C., Wang, H., Mahoney, M. W., Torkkola, K., Wilson, A. G., Bohlke-Schneider, M., and Wang, Y. (2024). Chronos: Learning the Language of Time Series, arXiv: 2403.07815.
- Bahrpeyma, F., Roantree, M., Cappellari, P., Scriney, M., and McCarren, A. (2021). “A Methodology for Validating Diversity in Synthetic Time Series Generation,” *MethodsX* **8**, 101459.
- Bianchi, D., Büchner, M., and Tamoni, A. (2020). “Bond Risk Premiums with Machine Learning,” *The Review of Financial Studies* **34**(2), 1046–1089.
- Caldeira, J., Moura, G. V., and Santos, A. A. (2013). “Predicting the Yield Curve Using Forecast Combinations,” Publication Title: SSRN Electronic Journal.
- Challu, C., Olivares, K. G., Oreshkin, B. N., Ramirez, F. G., Canseco, M. M., and Dubrawski, A. (2023). “NHITS: Neural Hierarchical Interpolation for Time Series Forecasting,” *Proceedings of the AAAI Conference on Artificial Intelligence* **37**(6), 6989–6997.
- Godahewa, R., Bergmeir, C., Webb, G. I., Hyndman, R. J., and Montero-Manso, P. (2021). “Monash Time Series Forecasting Archive,” arXiv:2105.06643 [cs].
- Hoogteijling, T., Martens, M. P. E., and Wel, M. v. d. (2021). “Forecasting Bond Risk Premia Using Stationary Yield Factors”.
- López de Prado, M. (2018). *Advances in Financial Machine Learning*, Wiley.
- Makridakis, S., Spiliotis, E. and Assimakopoulos, V. (2020). “The M4 Competition: 100,000 time-series and 61 Forecasting Methods,” *International Journal of Forecasting* **36**(1), 54–74.
- Moench, E. (2006). “Forecasting the Yield Curve in a Data-Rich Environment: A No-Arbitrage Factor-Augmented VAR Approach,” *SSRN Electronic Journal*.
- Runge, J., Gerhardus, A., Varando, G., Eyring, V., and Camps-Valls, G. (2023). “Causal Inference for Time Series,” *Nature Reviews Earth and Environment* **4**, 487–505, ADS Bibcode: 2023NRvEE...4..487R.
- Sambasivan, R. and Das, S. (2017). “A Statistical Machine Learning Approach to Yield Curve Forecasting,” In *2017 International Conference on Computational Intelligence in Data Science (ICCIDS)*.
- Sezer, O. B., Gudelek, M. U., and Ozbayoglu, A. M. (2019). “Financial Time Series Forecasting with Deep Learning,” *A Systematic Literature Review 2005–2019*, arXiv:1911.13288 [cs].
- Shumway, R. H. and Stoffer, D. S. (2025). *Time Series Analysis and Its Applications: With R Examples* (Springer Texts in Statistics. Cham: Springer Nature Switzerland).
- Singh, A., Ogunfunmi, T., and Das, S. (2025). “Yield Curve Forecasting Using Machine Learning and Econometrics: A Comparative Analysis,” *Journal of Investment Management* **23**(4), 37–58.
- Sunki, A., SatyaKumar, C., Surya Narayana, G., Koppera, V., and Hakeem, M. (2024). “Time Series Forecasting of Stock Market Using ARIMA, LSTM and FB prophet,” *MATEC Web of Conferences* **392**, 01163.
- Zhang, O. and Zhang, Z. (2026). “A Review of Large Language Models for Stock Price Forecasting from a Hedge-Fund Perspective,” arXiv:2605.05211 [q-fin].