PROBTS: A UNIFIED TOOLKIT TO PROBE DEEP TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

A COMPONENTS OF THE PROBTS TOOLKIT

Data. The data module unifies varied data scenarios to facilitate thorough evaluation and implements standardized pre-processing techniques to ensure fair comparison. Moreover, we utilize a quantitative approach to visually delineate datasets' intrinsic characteristics, which employs decomposition to assess trends and seasonality in a time series and evaluate the similarity between data distribution and a Gaussian to depict the complexity of data distribution. Descriptions and statistics for each dataset are listed in Table 1, and a quantitative evaluation of their inherent properties is provided in Table 2. We attach the detailed quantitative calculation process in Appendix B.

Model. The modularized model module accommodates diverse neural network architectures, forecasting paradigms, and decoding schemes. Adhering to the decoupled model formulation from Section **??**, it enables the construction of various models by configuring the encoder f_ϕ and forecaster p_θ. For example, point estimation methods like DLinear centralize their design in the encoder, using a linear layer or identity mapping as the forecaster, with non-autoregressive decoding. In contrast, probabilistic models like TimeGrad incorporate general neural architectures in the encoder and advanced probabilistic techniques in the forecaster, employing autoregressive decoding.

Evaluator. The evaluator module integrates a diverse array of evaluation metrics such as Mean Absolute Error (MAE), Normalized Mean Absolute Error (NMAE), Mean Square Error (MSE), and Continuous Ranked Probability Score (CRPS), allowing for assessment of both point-level and distribution-level accuracies. We employ the NMAE metric for point-level evaluation to accommodate different scales of errors, and unlike previous studies (Rasul et al., 2021a; Tashiro et al., 2021) that used the CRPS_{sum} metric, we utilize CRPS for our analysis for a refined evaluation of each variate's probability distribution accuracy. A detailed list of evaluation metrics and their formal definitions can be found in Appendix C.

Implementation. To ensure the integrity of the results, `ProbTS` adheres to a standard implementation process, employing unified data splitting, standardization techniques, and adopting fair settings for hyperparameter tuning across all methods. We utilize reported optimal hyperparameters for models directly associated with specific datasets and conduct an extensive grid search to identify the most effective settings for those hyperparameters that were not available. Details regarding the experimental setup can be found in Appendix D.

B QUANTIFYING THE CHARACTERISTICS OF DATASETS

Trend & Seasonality To gain deeper insights into the dataset characteristics, we conducted a quantitative evaluation of trend and seasonality for each dataset, drawing upon methodologies outlined in the work of Wang et al. (2006). In particular, we employed a time series decomposition model expressed as:

$$y_t = T_t + S_t + R_t,$$

where T_t represents the smoothed trend component, S_t signifies the seasonal component, and R_t denotes the remainder component. In order to obtain each component, we followed the STL decomposition approach [1].

[1] `https://otexts.com/fpp2/stl.html`

Table 1: Dataset Summary.

Horizon	Dataset	#var.	range	freq.	timesteps	Description
Long-term	ETTh1/h2	7	$\mathbb{R}^+$	H	17,420	Electricity transformer temperature per hour
	ETTm1/m2	7	$\mathbb{R}^+$	15min	69,680	Electricity transformer temperature every 15 min
	Electricity	321	$\mathbb{R}^+$	H	26,304	Electricity consumption (Kwh)
	Traffic	862	(0,1)	H	17,544	Road occupancy rates
	Exchange	8	$\mathbb{R}^+$	Busi. Day	7,588	Daily exchange rates of 8 countries
	ILI	7	(0,1)	W	966	Ratio of patients seen with influenza-like illness
	Weather	21	$\mathbb{R}^+$	10min	52,696	Local climatological data
Short-term	Exchange	8	$\mathbb{R}^+$	Busi. Day	6,071	Daily exchange rates of 8 countries
	Solar	137	$\mathbb{R}^+$	H	7,009	Solar power production records
	Electricity	370	$\mathbb{R}^+$	H	5,833	Electricity consumption
	Traffic	963	(0,1)	H	4,001	Road occupancy rates
	Wikipedia	2,000	$\mathbb{N}$	D	792	Page views of 2000 Wikipedia pages

In the case of strongly trended data, the variation within the seasonally adjusted data should considerably exceed that of the remainder component. Consequently, the ratio $\text{Var}(R_t)/\text{Var}(T_t + R_t)$ is expected to be relatively small. As such, the measure of trend strength can be formulated as:

$$F_T = \max\left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(T_t + R_t)}\right).$$

The quantified trend strength, ranging from 0 to 1, characterizes the degree of trend presence. Similarly, the evaluation of seasonal intensity employs the detrended data:

$$F_S = \max\left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(S_t + R_t)}\right).$$

A series with F_S near 0 indicates minimal seasonality, while strong seasonality is indicated by F_S approaching 1 due to the considerably smaller variance of $\text{Var}(R_t)$ in comparison to $\text{Var}(S_t + R_t)$.

Tables 2 depict the results for each dataset. Notably, the ETT datasets and the Exchange dataset manifest conspicuous trends, whereas the Electricity, Solar, and Traffic datasets showcase marked seasonality. Additionally, the Exchange dataset stands out with distinctive features. Figure 2 illustrates that with shorter prediction windows, the Exchange dataset sustains comparatively minor fluctuations, almost forming a linear trajectory. This enables effective forecasting through a straightforward batch mean approach. As the forecasting horizon extends, the dataset appears a more pronounced trend while retaining minimal seasonality.

Data Distribution To analyze the influence of data distribution on model performance, we measured the similarity between each dataset's distribution and the Gaussian distribution. Specifically, we computed the Jensen–Shannon divergence (Nielsen, 2019) within a fixed-length sliding window for each variate. A window size of 30 was used for short-term datasets and 336 for long-term ones. The average of these calculations yielded the overall degree of conformity of each dataset to the Gaussian distribution. These results are summarized in Table 2.

Outliers Outliers are data points that are significantly distant from the rest, which pose challenges in forecasting. We quantified outlier ratios from both global and local perspectives. The global view treats the entire dataset as a Gaussian distribution and identifies Z-score normalized values more than 3 standard deviations from the mean as outliers. The local perspective assesses outliers within a sliding window, following the same criterion. For short-term datasets, a window size of 30 is employed, while for long-term forecasting datasets, the window size is set to 336. We present the ratio of outliers in Table 3 for reference. From Table 3, we find that some datasets, such as Wikipedia-S, possess a high local ratio of outliers, which can have a large impact on short-term forecasting.

Data Visualization To offer a clearer insight into the characteristics of each dataset and the influence of varying forecasting horizons, we have illustrated instances of both short-term and long-term forecasting datasets in Figure 1 and Figure 2 respectively. Figure 1 reveals that in short-term scenarios, time series are primarily governed by local variations. On the other hand, as depicted in

Table 2: Quantitative assessment of intrinsic characteristics for each dataset. To eliminate ambiguity, we use the suffix "-S" and "-L" to denote short-term and long-term forecasting datasets, respectively. The JS Div denotes Jensen–Shannon divergence, where a lower score indicates closer approximations to a Gaussian distribution.

Dataset	**Exchange-S**	**Solar-S**	**Electricity-S**	**Traffic-S**	**Wikipedia-S**	**ETTm1-L**	**ETTm2-L**
Trend F_T	0.9982	0.1688	0.6443	0.2880	0.5253	0.9462	0.9770
Seasonality F_S	0.1256	0.8592	0.8323	0.6656	0.2234	0.0105	0.0612
JS Div.	0.2967	0.5004	0.3579	0.2991	0.2751	0.0833	0.1701
Dataset	**ETTh1-L**	**ETTh2-L**	**Electricity-L**	**Traffic-L**	**Weather-L**	**Exchange-L**	**ILI-L**
Trend F_T	0.7728	0.9412	0.6476	0.1632	0.9612	0.9978	0.5438
Seasonality F_S	0.4772	0.3608	0.8344	0.6798	0.2657	0.1349	0.6075
JS Div.	0.0719	0.1422	0.1533	0.1378	0.1727	0.1082	0.1112

Table 3: Ratio of outliers (%). The suffix "-S" denotes short-term forecasting datasets, while "-L" signifies long-term forecasting datasets.

Dataset	**Exchange-S**	**Solar-S**	**Electricity-S**	**Traffic-S**	**Wikipedia-S**	**ETTm1-L**	**ETTm2-L**
Local	0.1718	0.2228	0.1333	0.6595	1.5435	0.4126	0.4231
Global	0.0871	0.0002	0.4210	1.6890	1.1758	1.1079	1.8764
Dataset	**ETTh1-L**	**ETTh2-L**	**Electricity-L**	**Traffic-L**	**Weather-L**	**Exchange-L**	**ILI-L**
Local	0.4937	0.4707	0.1529	1.4352	0.5106	0.2021	1.2422
Global	1.2951	2.1929	0.4134	1.5885	0.8323	0.0066	1.5735

Figure 2, datasets like Traffic, Electricity, and ETT, under extended forecasting horizons, display enhanced seasonality and trends, making these series more predictable.

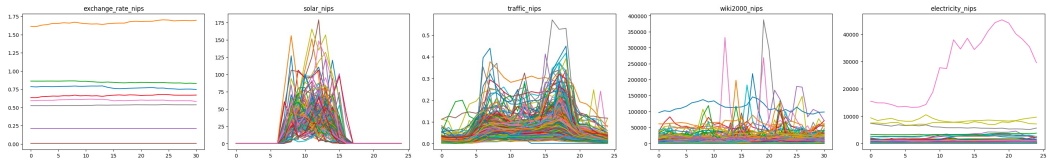


Figure 1: Time series samples extracted from the short-term forecasting dataset. The range of the x-axis is the pre-defined length of the prediction window in each dataset.

C Evaluation Metrics

The `ProbTS` toolkit incorporates a comprehensive range of metrics, spanning both point-level and distribution-level, to offer a nuanced and multifaceted evaluation of forecasting models.

C.1 Point-level Metrics

For point-level metrics, we primarily focused on several measures that are predominantly used in the branch devoted to optimizing neural network architecture design.

Mean Absolute Error (MAE) The Mean Absolute Error (MAE) quantifies the average absolute deviation between the forecasts and the true values. Since it averages the absolute errors, MAE is robust to outliers. Its mathematical formula is given by:

$$\text{MAE} = \frac{1}{K \times T} \sum_{i=1}^{K} \sum_{t=1}^{T} |x_{i,t} - \hat{x}_{i,t}|,$$

where K is the number of variates, L is the length of series, $x_{i,t}$ and $\hat{x}_{i,t}$ denotes the ground-truth value and the predicted value, respectively. For multivariate time series, we also provide the

Figure 2: Time series samples extracted from the long-term forecasting dataset. The x-axis spans the pre-defined prediction window lengths within each dataset, with prediction lengths set to $T \in \{24, 36, 48, 60\}$ for the ILI dataset and $T \in \{96, 192, 336, 720\}$ for the remaining datasets.

aggregated version:

$$\text{MAE}_{\text{sum}} = \frac{1}{T}\sum_{t=1}^{T}|x_t^{\text{sum}} - \hat{x}_t^{\text{sum}}|,$$

where x_t^{sum} and $\hat{x}_t^{\text{sum}}$ are the summation across the dimension K of $x_{i,t}$ and $\hat{x}_{i,t}$, respectively.

Normalized Mean Absolute Error (NMAE) The Normalized Mean Absolute Error (NMAE) is a normalized version of the MAE, which is dimensionless and facilitates the comparability of the error magnitude across different datasets or scales. The mathematical representation of NMAE is given by:

$$\text{NMAE} = \frac{1}{K \times T}\sum_{i=1}^{K}\sum_{t=1}^{T}\frac{|x_{i,t} - \hat{x}_{i,t}|}{|x_{i,t}|}.$$

Its aggregated version is:

$$\text{NMAE}_{\text{sum}} = \frac{1}{T}\sum_{t=1}^{T}\frac{|x_t^{\text{sum}} - \hat{x}_t^{\text{sum}}|}{|x_t^{\text{sum}}|}.$$

Mean Squared Error (MSE) The Mean Squared Error (MSE) is a quantitative metric used to measure the average squared difference between the observed actual value and forecasts. It is defined mathematically as follows:

$$\text{MSE} = \frac{1}{K \times T}\sum_{i=1}^{K}\sum_{t=1}^{L}(x_{i,t} - \hat{x}_{i,t})^2.$$

For multivariate time series, we also provide the aggregated version:

$$\text{MSE}_{\text{sum}} = \frac{1}{T}\sum_{t=1}^{L}(x_t^{\text{sum}} - \hat{x}_t^{\text{sum}})^2.$$

Normalized Root Mean Squared Error (NRMSE) The Normalized Root Mean Squared Error (NRMSE) is a normalized version of the Root Mean Squared Error (RMSE), which quantifies the average squared magnitude of the error between forecasts and observations, normalized by the expectation of the observed values. It can be formally written as:

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{K\times T}\sum_{i=1}^{K}\sum_{t=1}^{L}(x_{i,t} - \hat{x}_{i,t})^2}}{\frac{1}{K\times T}\sum_{i=1}^{K}\sum_{t=1}^{T}|x_{i,t}|}.$$

For multivariate time series, we also provide the aggregated version:

$$\text{NRMSE}_{\text{sum}} = \frac{\sqrt{\frac{1}{T}\sum_{t=1}^{L}(x_t^{\text{sum}} - \hat{x}_t^{\text{sum}})^2}}{\frac{1}{T}\sum_{t=1}^{T}|x_t^{\text{sum}}|}.$$

C.2 Distribution-level Metrics

Continuous Ranked Probability Score (CRPS) The Continuous Ranked Probability Score (CRPS) (Matheson & Winkler, 1976) quantifies the agreement between a cumulative distribution function (CDF) F and an observation x, represented as:

$$\text{CRPS} = \int_{\mathbb{R}} (F(z) - \mathbb{I}\{x \le z\})^2 dz,$$

where $\mathbb{I}x \le z$ denotes the indicator function, equating to one if $x \le z$ and zero otherwise.

Being a proper scoring function, CRPS reaches its minimum when the predictive distribution F coincides with the data distribution. When using the empirical CDF of F, denoted as $\hat{F}(z) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}\{X_i \le z\}$, where n represents the number of samples $X_i \sim F$, CRPS can be precisely calculated from the simulated samples of the conditional distribution $p_\theta(\boldsymbol{x}_t|\boldsymbol{h}_t)$. In our practice, 100 samples are employed to estimate the empirical CDF.

For multivariate time series, the aggregate CRPS, denoted as CRPS_{sum}, is derived by summing across the K time series, both for the ground-truth data and sampled data, and subsequently averaging over the forecasting horizon. Formally, it is represented as:

$$\text{CRPS}_{\text{sum}} = \mathbb{E}_t\left[\text{CRPS}\left(\hat{F}_{\text{sum}}(t), \sum_{i=1}^{K} x_{i,l}^0\right)\right].$$

D Implementation Details

D.1 Experiment settings

`ProbTS` is implemented using PyTorch Lightning (Falcon & The PyTorch Lightning team, 2019). During the training, we sample 100 batches per epoch and train for a maximum of 50 epochs, employing the CRPS metric as the monitor for checkpoint saving. We employ the Adam optimizer for all experiments, which are executed on single NVIDIA Tesla V100 GPUs using CUDA 11.3. In the evaluation phase, we sample 100 times to report the metrics on the test set.

D.2 Hyper-parameters

We carried out an extensive grid search for models, tuning hyperparameters individually for each method. Given the large number of models, we inlude only the partial hyperparameter settings in Table 4. All hyperparameter configurations identified for each model on every dataset will be accessible via a GitHub repository, to be open-sourced subsequent to the paper's publication.

Table 4: Hyperparameter settings for Electricity-S dataset.

Model	Hyperparameter
DLinear	learning_rate=0.01, kernel_size=3, f_hidden_size=40
PatchTST	learning_rate=0.0001, stride=3, patch_len=6, n_layers=3, n_heads=8, dropout=0.1, kernel_size=3, f_hidden_size=32
TimesNet	learning_rate=0.001, n_layers=2, num_kernels=6, top_k=5, f_hidden_size=64, d_ff=64
GRU NVP	learning_rate=0.001, f_hidden_size=40, num_layers=2, n_blocks=3, hidden_size=100, conditional_length=200
GRU MAF	learning_rate=0.001, f_hidden_size=40, num_layers=2, n_blocks=4, hidden_size=100, conditional_length=200
Trans MAF	learning_rate=0.001, f_hidden_size=32, num_heads=8, n_blocks=4, hidden_size=100, conditional_length=200
TimeGrad	learning_rate=0.001, f_hidden_size=128, num_layers=4, conditional_length=100, beta_end=0.1, diff_steps=100
CSDI	learning_rate=0.001, channels=64, emb_time_dim=128, emb_feature_dim=16, num_steps=50, num_heads=8, n_layers=4

E Additional Results and Experiments

E.1 Impact of Data Scale

To further explore critical characteristics of time-series forecasting, we have examed the correlation between model performance gains, relative to the baseline model (GRU), and dataset dimensions, length, and volume (see Table 5). However, our analysis does not identify a significant correlation between these factors and model performance.

Table 5: The correlation coefficient between the data volume and the relative performance improvement compared to the baseline model (GRU).

Model	DLinear		PatchTST		GRU NVP		TimeGrad		CSDI	
	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE
# Var.	0.2422	0.2422	-0.2676	-0.2676	-0.1856	-0.2136	-0.1665	-0.1793	-0.2315	-0.2592
# Total timestep	-0.1422	-0.1422	0.3821	0.3821	0.3072	0.3329	0.2860	0.2971	0.3542	0.3826
# Var. $\times$ Timestep	0.0162	0.0162	0.0166	0.0166	-0.0068	-0.0011	0.0082	0.0117	-0.0053	-0.0133

E.2 Statistical and Gradient Boosting Decision Tree Baselines

To enhance the empirical robustness of our study, we integrate classical statistical models, including ARIMA (Makridakis & Hibon, 1997) and ETS (Hyndman & Athanasopoulos, 2018), along with the Gradient Boosting Decision Tree (GBDT) model, XGBoost, into the ProbTS framework. The results in Table 6 clearly demonstrate the superior performance of deep learning methods over simple statistical baselines, emphasizing the importance of capturing non-linear dependencies for accurate forecasts. Notably, ARIMA and ETS exhibit varied performance across different data characteristics. ARIMA struggles with datasets like Solar, characterized by weak trending and strong seasonality, while ETS shows better adaptability. Conversely, in cases of strong trending and weak seasonality, as observed in the 'Wikipedia' dataset, ARIMA significantly outperforms ETS.

Utilizing the implementation from Elsayed et al. (2021), we find that XGBoost competes well, even surpassing neural network models in certain scenarios. However, for datasets with more complex distributions like 'Solar' and 'Electricity,' advanced probabilistic estimation methods demonstrate a substantial advantage over traditional learning methods and point estimation techniques. This highlights the adaptability and strength of advanced probabilistic methods in handling intricate forecasting scenarios.

E.3 Experiments on Univariate Datasets

In pursuit of a comprehensive analysis spanning univariate and multivariate scenarios, we examined a subset of M4 (Makridakis et al., 2020), M5 (Makridakis et al., 2022), and TOURISM datasets (Athanasopoulos et al., 2011)—crucial datasets for univariate time-series forecasting. Table 7 provides a quantitative assessment of the intrinsic characteristics of these new datasets, focusing on trending strength, seasonality, and data distribution complexity, as detailed in our paper.

Table 6: Results of statistical models and GBDT baseline on short-term forecasting datasets.

Model	Exchange Rate		Solar		Electricity		Traffic		Wikipedia	
	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE
ARIMA	0.009	0.009	1.000	1.000	0.164	0.164	0.461	0.461	0.348	0.348
ETS	0.011	0.011	0.580	0.580	0.121	0.121	0.413	0.413	0.685	0.685
ETS-prob	0.008	0.011	0.795	0.695	0.123	0.129	0.380	0.433	0.625	0.697
XGBoost	0.011	0.011	0.599	0.599	0.074	0.074	0.196	0.196	-	-
DLinear	$0.012_{.001}$	$0.012_{.001}$	$0.547_{.009}$	$0.547_{.009}$	$0.095_{.006}$	$0.095_{.006}$	$0.273_{.012}$	$0.273_{.012}$	$1.046_{.037}$	$1.046_{.037}$
PatchTST	$\underline{0.010_{.000}}$	$\mathbf{0.010_{.000}}$	$0.496_{.002}$	$0.496_{.002}$	$0.076_{.001}$	$0.076_{.001}$	$0.202_{.001}$	$0.202_{.001}$	$\underline{0.257_{.001}}$	$\mathbf{0.257_{.001}}$
TimesNet	$0.011_{.001}$	$0.011_{.001}$	$0.507_{.019}$	$0.507_{.019}$	$0.071_{.002}$	$0.071_{.002}$	$0.205_{.002}$	$0.205_{.002}$	$0.304_{.002}$	$0.304_{.002}$
GRU NVP	$0.016_{.003}$	$0.020_{.003}$	$0.396_{.021}$	$0.507_{.022}$	$0.055_{.002}$	$0.073_{.003}$	$0.161_{.006}$	$0.203_{.009}$	$0.282_{.003}$	$0.330_{.003}$
GRU MAF	$0.015_{.001}$	$0.020_{.001}$	$0.386_{.026}$	$0.492_{.027}$	$\underline{0.051_{.001}}$	$\underline{0.067_{.001}}$	$\underline{0.131_{.006}}$	$\underline{0.165_{.009}}$	$0.281_{.004}$	$0.337_{.005}$
Trans MAF	$0.011_{.001}$	$0.014_{.001}$	$0.400_{.022}$	$0.503_{.022}$	$0.054_{.004}$	$0.071_{.005}$	$\mathbf{0.129_{.004}}$	$\mathbf{0.165_{.006}}$	$0.289_{.008}$	$0.344_{.008}$
TimeGrad	$0.011_{.001}$	$0.014_{.002}$	$\mathbf{0.359_{.011}}$	$\mathbf{0.445_{.023}}$	$0.052_{.001}$	$\underline{0.067_{.001}}$	$0.164_{.091}$	$0.201_{.115}$	$0.272_{.008}$	$0.327_{.011}$
CSDI	$\mathbf{0.008_{.000}}$	$\underline{0.011_{.000}}$	$\underline{0.366_{.005}}$	$\underline{0.484_{.008}}$	$\mathbf{0.050_{.001}}$	$\mathbf{0.065_{.001}}$	$0.146_{.012}$	$0.176_{.013}$	$\mathbf{0.219_{.006}}$	$\underline{0.259_{.009}}$

Notably, these datasets, except for M4-Daily may exhibit fewer seasonal patterns, do not introduce particularly unique characteristics.

Table 7: Quantitative assessment of the intrinsic characteristics of the univariate datasets. The JS Div denotes Jensen–Shannon divergence, where a lower score indicates closer approximations to a Gaussian distribution.

Dataset	**M4-Weekly**	**M4-Daily**	**M5**	**TOURISM-Monthly**
Trend F_T	0.7677	0.9808	0.3443	0.7979
Seasonality F_S	0.3401	0.0467	0.2480	0.6826
JS Div.	0.5106	0.4916	0.6011	0.3291

Table 8 presents experimental results for representative methods, consistent with our initial observations. Probabilistic estimation methods like GRU NVP and TimeGrad excel on datasets with complex distributions (e.g., M4-Weekly and M5), while simpler point forecasting methods such as DLinear and PatchTST perform well on datasets with relatively simple data distribution, like TOURISM-Monthly. Both autoregressive and non-autoregressive decoding schemes show comparable performance in short-term forecasting, as discussed in the main paper."

Table 8: Results on M4, M5, and TOURISM datasets. We utilize a lookback window of 3H, with 'H' denoting the forecasting horizon.

Model	DLinear		PatchTST		GRU NVP		TimeGrad	
	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE
M4-Weekly	0.081	0.081	0.089	0.089	0.066	0.077	0.055	0.065
M4-Daily	0.034	0.034	0.035	0.035	0.030	0.038	0.026	0.032
M5	0.891	0.891	0.898	0.898	0.679	0.864	-	-
TOURISM-Monthly	0.168	0.168	0.136	0.136	0.171	0.223	0.152	0.191

E.4 Experiments on Synthetic Datasets

To enhance the rigor of the insights presented, we employ synthetic datasets, encompassing a baseline dataset and variants with pronounced trends, strong seasonality, and complex data distribution (see Table 9). Each dataset comprises series generated by combining trend, seasonality, noise, and anomaly components with controlled characteristics. Subsequent experiments on these synthetic datasets (refer to Table 10), using representative models, validate the empirical findings established on other datasets with ProbTS. Key observations include the declining performance of autoregressive decoding models, such as TimeGrad, in the presence of increasing trends, improved performance for models using autoregressive decoding with intensifying seasonality, and the competitive performance of probabilistic methods like CSDI in handling more complex data distributions.

Table 9: Quantitative assessment of intrinsic characteristics for synthetic datasets. The JS Div denotes Jensen–Shannon divergence, where a lower score indicates closer approximations to a Gaussian distribution.

Dataset	**Normal**	**Strong Trend**	**Strong Seasonality**	**Complex Distribution**
Trend F_T	0.105	0.554	0.105	0.064
Seasonality F_S	0.302	0.302	0.791	0.190
JS Div.	0.261	0.248	0.272	0.469

Table 10: Results on synthetic datasets. The look-back window and forecasting horizon are 30.

Model	Normal		Strong Trend		Strong Seasonality		Complex Distribution	
	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE	CRPS	NMAE
DLinear	0.013	0.013	**0.001**	**0.001**	0.014	0.014	0.301	0.301
PatchTST	**0.012**	**0.012**	**0.001**	**0.001**	**0.012**	**0.012**	0.275	**0.275**
TimeGrad	0.024	0.032	0.042	0.048	0.022	0.028	0.283	0.338
CSDI	0.013	0.014	0.010	0.007	0.020	0.027	**0.269**	0.301

E.5 Case Study

To intuitively demonstrate the distinct characteristics of point and probabilistic estimations, a case study was conducted on short-term datasets. Figure 3 illustrates that point estimation yields single-valued, deterministic estimates, in contrast to probabilistic methods, which model continuous data distributions as depicted in Figure 4. This modeling of data distributions captures the uncertainty in forecasts, aiding decision-makers in fields such as weather and finance to make more informed choices. It is also observed that while both methods align well with ground truth values in short-term forecasting datasets, they struggle to accurately capture outliers, particularly noted in the Wikipedia dataset.

E.6 Model Efficiency

For reference, detailed results regarding memory usage and time efficiency for five representative models on long-term forecasting datasets are provided here. Table 11 displays the computation memory of various models with a forecasting horizon set to 96. Additionally, Table 12 compares the inference time of these models on long-term forecasting datasets, illustrating the impact of changes in the forecasting horizon.

Table 11: Computation memory. The batch size is 1 and the prediction horizon is set to 96.

Metric	Dataset	DLinear	PatchTST	LSTM NVP	TimeGrad	CSDI
NPARAMS (MB)	ETTm1	0.075	2.145	1.079	1.233	1.720
	Electricity	0.076	2.146	3.680	3.472	1.370
	Traffic	0.078	2.149	15.926	8.298	1.390
	Weather	0.075	2.145	3.085	0.574	1.721
	Exchange	0.075	0.135	1.979	0.488	1.720
Max GPU Mem. (GB)	ETTm1	0.002	0.009	0.010	0.012	0.027
	Electricity	0.060	0.068	0.129	0.128	1.411
	Traffic	0.161	0.168	0.361	0.333	9.102
	Weather	0.004	0.012	0.021	0.012	0.070
	Exchange	0.002	0.002	0.013	0.008	0.030

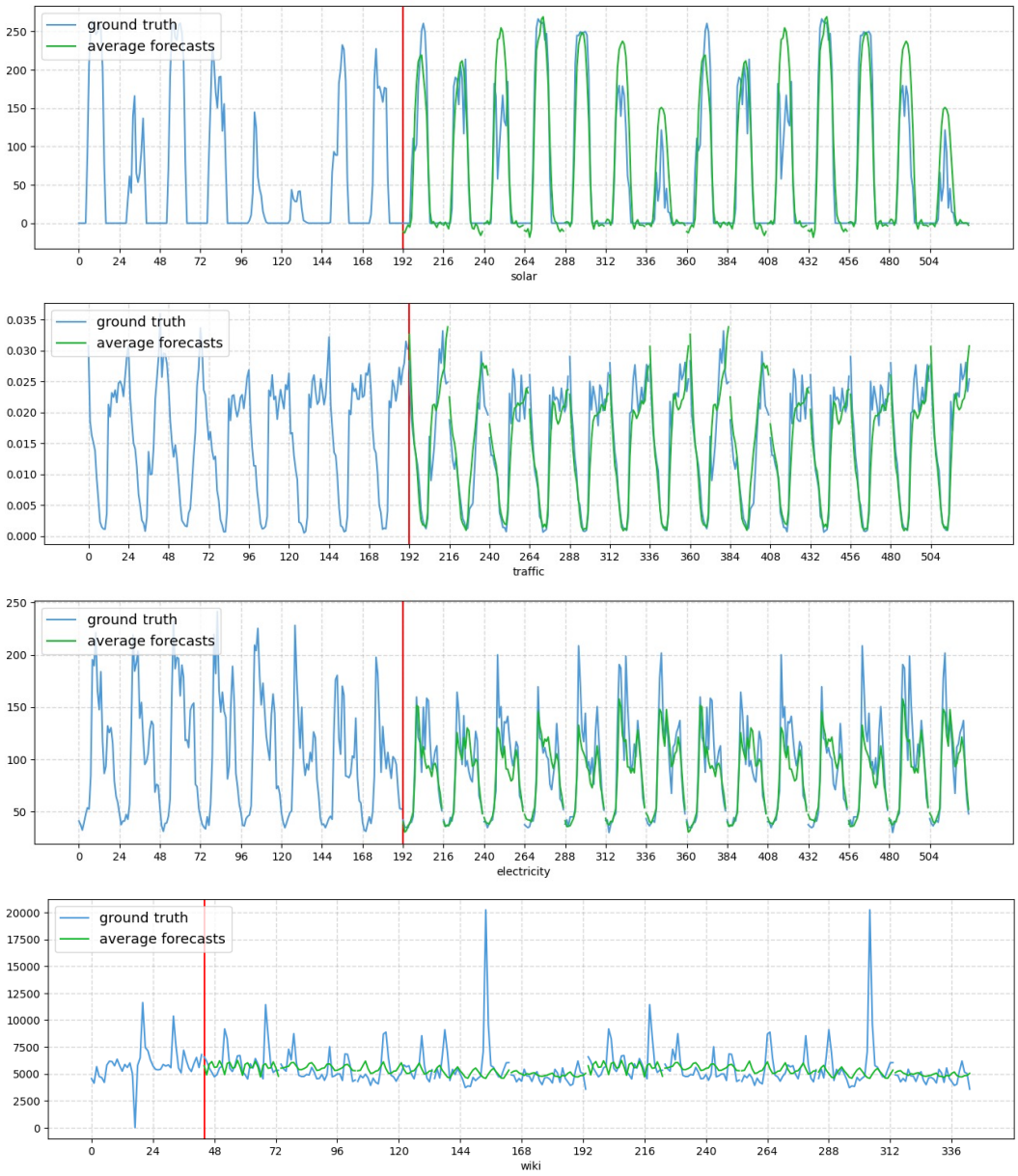


Figure 3: Point forecasts from the PatchTST model and the ground-truth value on short-term forecasting datasets.

F Further Discussion on Cross-channel Interactions

We compile a summary table (Table 13) delineating how models from each branch address the multivariate aspect. Despite a thorough investigation, we have not identified a clear pattern linking the modeling of cross-channel interactions to overall model performance. A notable trend is the prevalent use of a channel-mixing approach in most studies. However, findings are diverse; models like DLinear and PatchTST suggest that processing channels independently can yield superior results, while others like CSDI indicate that explicit modeling of cross-channel interactions offers significant advantages. This diversity underscores the ongoing exploration of the impact of cross-channel interactions on forecasting performance.

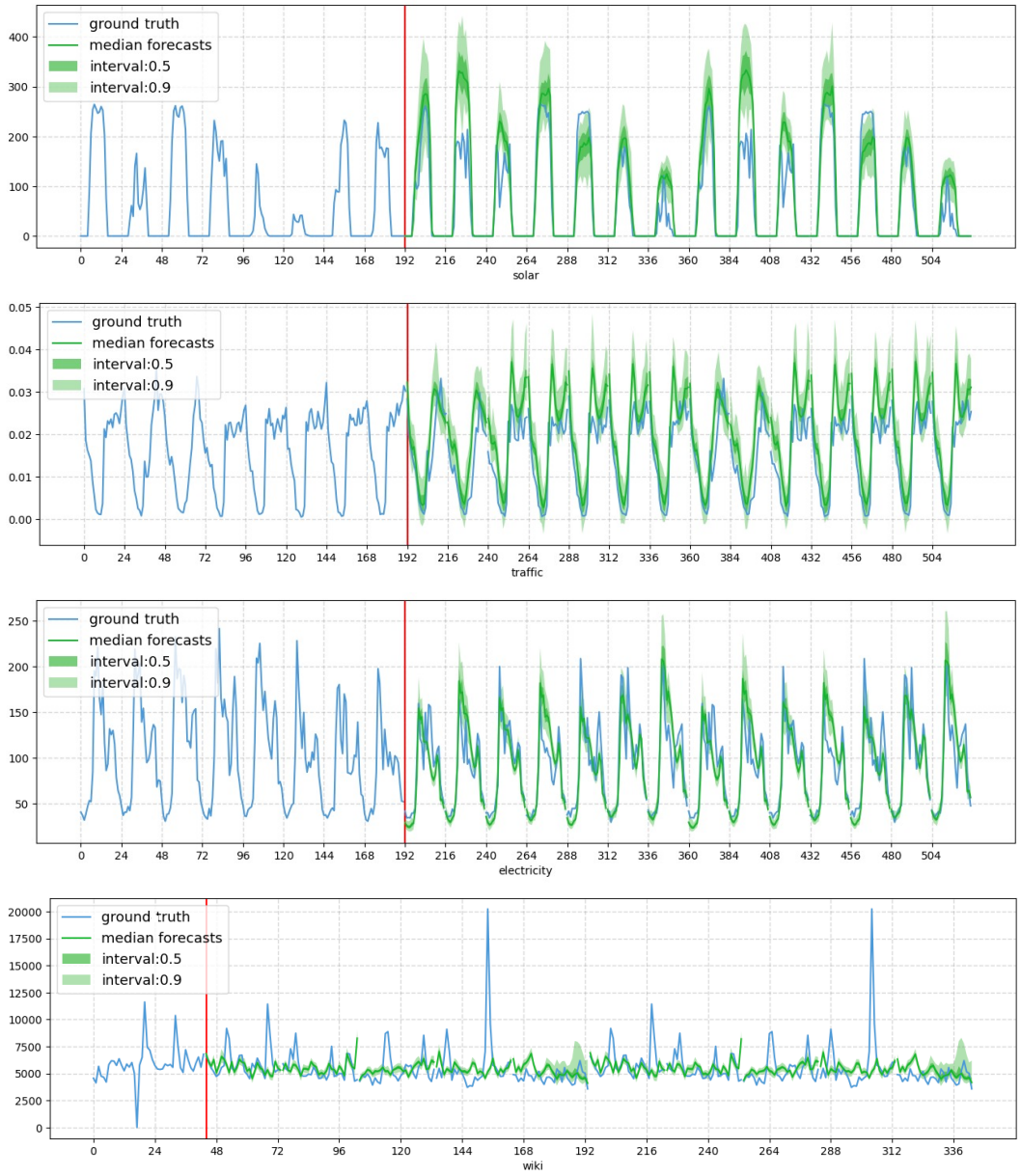


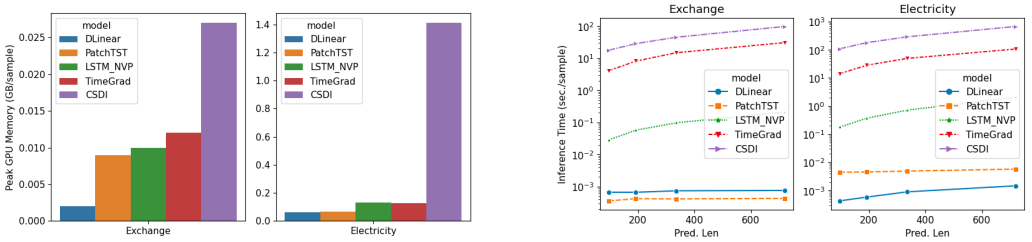
Figure 4: Forecasting intervals from the TimeGrad model and the ground-truth value on short-term forecasting datasets.

REFERENCES

George Athanasopoulos, Rob J Hyndman, Haiyan Song, and Doris C Wu. The tourism forecasting competition. *International Journal of Forecasting*, 27(3):822–844, 2011.

Marin Bilos, Kashif Rasul, Anderson Schneider, Yuriy Nevmyvaka, and Stephan Günnemann. Modeling Temporal Data as Continuous Functions with Stochastic Process Diffusion. In *In Proc. of ICML*, pp. 2452–2470, 2023.

Cristian Challu, Kin G. Olivares, Boris Oreshkin, Federico Ramirez, Max Canseco, and Artur Dubrawski. NHITS: Neural Hierarchical Interpolation for Time Series Forecasting. In *In Proc. of AAAI*, pp. 6989–6997, 2023.



(a) Computational memory.

(b) Inference time.

Figure 5: Comparison of computational efficiency. The forecasting horizon is set to 96 for calculating memory usage.

Table 12: Comparison of inference time (sec./sample).

Model	pred len	DLinear	PatchTST	LSTM NVP	TimeGrad	CSDI
ETTm1	96	0.0003 ± 0.0000	0.0003 ± 0.0000	0.0352 ± 0.0007	4.1067 ± 0.0504	16.3280 ± 0.0747
	192	0.0003 ± 0.0000	0.0003 ± 0.0000	0.0697 ± 0.0020	7.8979 ± 0.0403	25.8378 ± 0.3124
	336	0.0003 ± 0.0000	0.0003 ± 0.0000	0.1221 ± 0.0044	13.6197 ± 0.1023	39.8832 ± 0.2157
	720	0.0004 ± 0.0000	0.0003 ± 0.0000	0.2603 ± 0.0020	28.6074 ± 1.1346	86.1862 ± 0.1863
Electricity	96	0.0004 ± 0.0000	0.0045 ± 0.0001	0.1783 ± 0.0006	13.8439 ± 0.0054	388.3150 ± 0.2155
	192	0.0006 ± 0.0000	0.0046 ± 0.0000	0.3700 ± 0.0010	27.6683 ± 0.0368	659.4284 ± 0.2003
	336	0.0008 ± 0.0000	0.0049 ± 0.0000	0.7157 ± 0.0028	48.4456 ± 0.0279	-
	720	0.0015 ± 0.0000	0.0057 ± 0.0000	2.0785 ± 0.0186	104.1473 ± 0.1465	-
Traffic	96	0.0010 ± 0.0001	0.0102 ± 0.0000	0.3695 ± 0.0022	31.7644 ± 0.0101	-
	192	0.0013 ± 0.0000	0.0106 ± 0.0000	0.8287 ± 0.0094	63.5832 ± 0.0060	-
	336	0.0020 ± 0.0000	0.0114 ± 0.0001	1.6945 ± 0.0026	111.4147 ± 0.0169	-
	720	0.0039 ± 0.0000	0.0137 ± 0.0000	5.0963 ± 0.0018	258.1274 ± 0.6088	-
Weather	96	0.0002 ± 0.0000	0.0004 ± 0.0000	0.0800 ± 0.0016	4.1261 ± 0.0812	37.8984 ± 0.0782
	192	0.0003 ± 0.0000	0.0004 ± 0.0000	0.1568 ± 0.0008	8.2913 ± 0.5544	62.0223 ± 0.2329
	336	0.0003 ± 0.0000	0.0004 ± 0.0000	0.2482 ± 0.0297	14.2391 ± 0.4891	96.8704 ± 0.2258
	720	0.0003 ± 0.0000	0.0005 ± 0.0000	0.5447 ± 0.0249	29.4407 ± 0.3519	216.6044 ± 0.4253
Exchange	96	0.0006 ± 0.0000	0.0004 ± 0.0000	0.0284 ± 0.0001	4.1069 ± 0.0981	17.8655 ± 0.1282
	192	0.0007 ± 0.0000	0.0004 ± 0.0000	0.0563 ± 0.0008	8.1576 ± 0.0911	28.5456 ± 0.0873
	336	0.0007 ± 0.0000	0.0004 ± 0.0000	0.0966 ± 0.0007	14.4593 ± 0.4466	44.9733 ± 0.3820
	720	0.0007 ± 0.0000	0.0004 ± 0.0000	0.2085 ± 0.0046	30.1443 ± 0.5378	97.7417 ± 0.2606
ILI	24	0.0002 ± 0.0000	0.0008 ± 0.0001	0.0080 ± 0.0001	1.0427 ± 0.0190	12.4038 ± 0.1681
	192	0.0002 ± 0.0000	0.0008 ± 0.0000	0.0121 ± 0.0003	1.5762 ± 0.0282	12.7187 ± 0.1344
	336	0.0002 ± 0.0000	0.0008 ± 0.0000	0.0155 ± 0.0002	2.1344 ± 0.0660	12.7386 ± 0.1868
	720	0.0002 ± 0.0000	0.0008 ± 0.0000	0.0196 ± 0.0004	2.5787 ± 0.0594	12.5407 ± 0.0481

Shereen Elsayed, Daniela Thyssens, Ahmed Rashed, Hadi Samer Jomaa, and Lars Schmidt-Thieme. Do we really need deep learning models for time series forecasting? *arXiv preprint arXiv:2101.02118*, 2021.

William Falcon and The PyTorch Lightning team. PyTorch Lightning, March 2019. URL `https://github.com/Lightning-AI/lightning`.

Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

Spyros Makridakis and Michele Hibon. Arma models and the box–jenkins methodology. *Journal of forecasting*, 16(3):147–163, 1997.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The m4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1): 54–74, 2020.

Table 13: Summary of how existing models handle multivariate time series.

Model	Research branch	Process channels independently
Customized neural architectures	N-BEATS (Oreshkin et al., 2020)	✓
	N-HiTS (Challu et al., 2023)	✓
	Autoformer (Wu et al., 2021)	✗
	Informer (Zhou et al., 2021)	✗
	LTSF-Linear (Zeng et al., 2023)	✗/ ✓
	PatchTST (Nie et al., 2023)	✗/ ✓
	TimesNet (Wu et al., 2023)	✗
Probabilistic estimation	DeepAR (Salinas et al., 2020)	✓
	GP-copula (Salinas et al., 2019)	✗
	LSTM NVP (Rasul et al., 2021b)	✗
	LSTM MAF (Rasul et al., 2021b)	✗
	Trans MAF (Rasul et al., 2021b)	✗
	TimeGrad (Rasul et al., 2021a)	✗
	CSDI (Tashiro et al., 2021)	✗
	SPD (Bilos et al., 2023)	✗

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The m5 competition: Background, organization, and implementation. *International Journal of Forecasting*, 38(4):1325–1336, 2022.

James E Matheson and Robert L Winkler. Scoring Rules for Continuous Probability Distributions. *Management science*, 22(10):1087–1096, 1976.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *In Proc. of ICLR*, 2023.

Frank Nielsen. On the Jensen–Shannon symmetrization of distances relying on abstract means. *Entropy*, 21(5):485, 2019.

Boris N. Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. In *In Proc. of ICLR*, 2020.

Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive Denoising Diffusion Models for Multivariate Probabilistic Time Series Forecasting. In *In Proc. of ICML*, pp. 8857–8868, 2021a.

Kashif Rasul, Abdul-Saboor Sheikh, Ingmar Schuster, Urs M. Bergmann, and Roland Vollgraf. Multivariate Probabilistic Time Series Forecasting via Conditioned Normalizing Flows. In *In Proc. of ICLR*, 2021b.

David Salinas, Michael Bohlke-Schneider, Laurent Callot, Roberto Medico, and Jan Gasthaus. High-dimensional Multivariate Forecasting with Low-rank Gaussian Copula Processes. In *In Proc. of NeurIPS*, pp. 6824–6834, 2019.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. DeepAR: Probabilistic Forecasting with Autoregressive Recurrent Networks. *International Journal of Forecasting*, 36 (3):1181–1191, 2020.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. CSDI: Conditional Score-based Diffusion Models for Probabilistic Time Series Imputation. In *In Proc. of NeurIPS*, pp. 24804–24816, 2021.

Xiaozhe Wang, Kate Smith, and Rob Hyndman. Characteristic-based Clustering for Time Series Data. *Data mining and knowledge Discovery*, 13:335–364, 2006.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. In *In Proc. of NeurIPS*, pp. 22419–22430, 2021.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In *In Proc. of ICLR*, 2023.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are Transformers Effective for Time Series Forecasting? In *In Proc. of AAAI*, pp. 11121–11128, 2023.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond Efficient Transformer for Long Sequence Time-series Forecasting. In *In Proc. of AAAI*, pp. 11106–11115, 2021.