

N-BEATS: Neural basis expansion analysis for interpretable time series forecasting

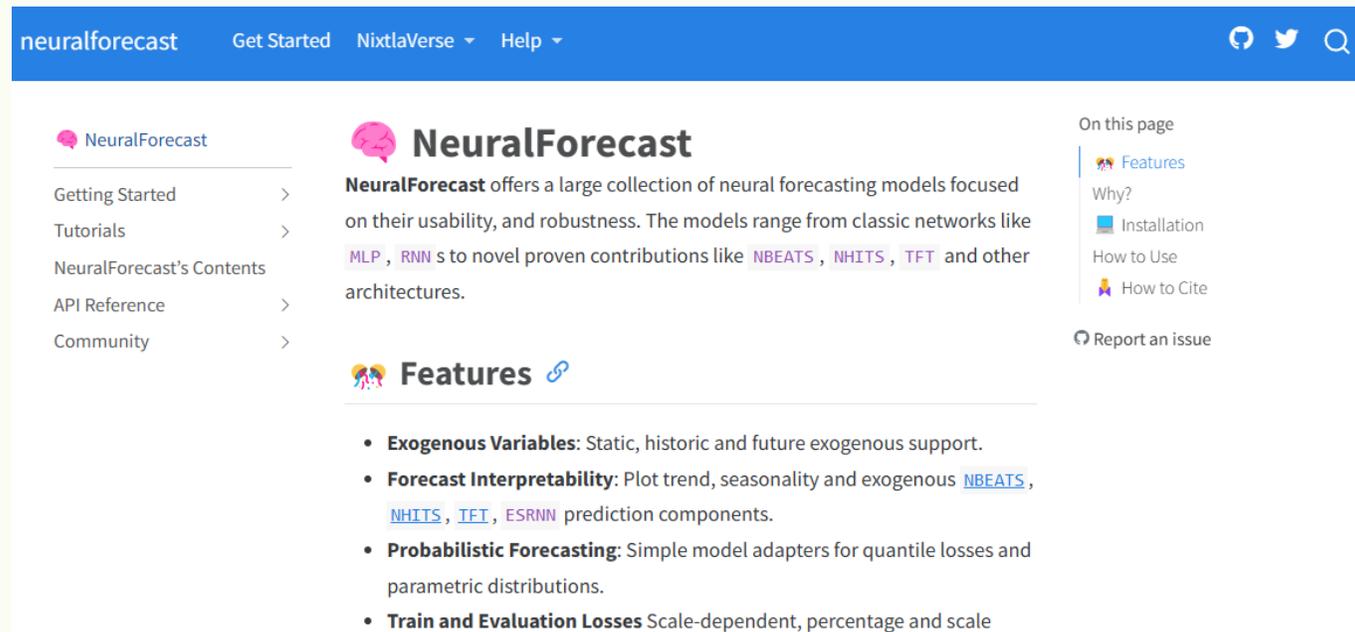
N-HiTS: Neural Hierarchical Interpolation for Time Series Forecasting

NTTコミュニケーションズ イノベーションセンター

木村大地

選定理由

- 最近 Transformer ベース以外の論文がこの読み会で流行っているみたい
- N-BEATS は MLPベース の時系列予測のベースライン的なところがあるし、拡張もいろいろ
- N-HiTS は AAAI23 のポスターで聞いて面白かったから
- いずれもライブラリ化 ([neuralforecast](#)) されているので実験がしやすい



N-BEATS: Neural basis expansion analysis for interpretable time series forecasting

- 著者・所属機関: Boris N. Oreshkin¹, Dmitri Carпов¹, Nicolas Chapados¹, Yoshua Bengio²
 - 1: Element AI, 2: Mila
- 投稿日付: 2020/2/20
- 投稿誌・学会: ICLR 2020
- 論文へのリンク: <https://arxiv.org/abs/1905.10437>

概要

- 単変量時系列予測モデルの提案
 - foreward/backward方向の残差ブロックと全結合層を積み重ねた構造
 - 時系列特有の特徴量設計や入力スケーリングに依存しない構造を目指す
 - 解釈性/様々なドメインで学習可能/学習が高速
 - M3, M4, TOURISMなど有名データセットでSOTA
 - 特にM4では、前年度の優勝手法よりも3%の改善

準備 -時系列データの問題設定-

csvファイル

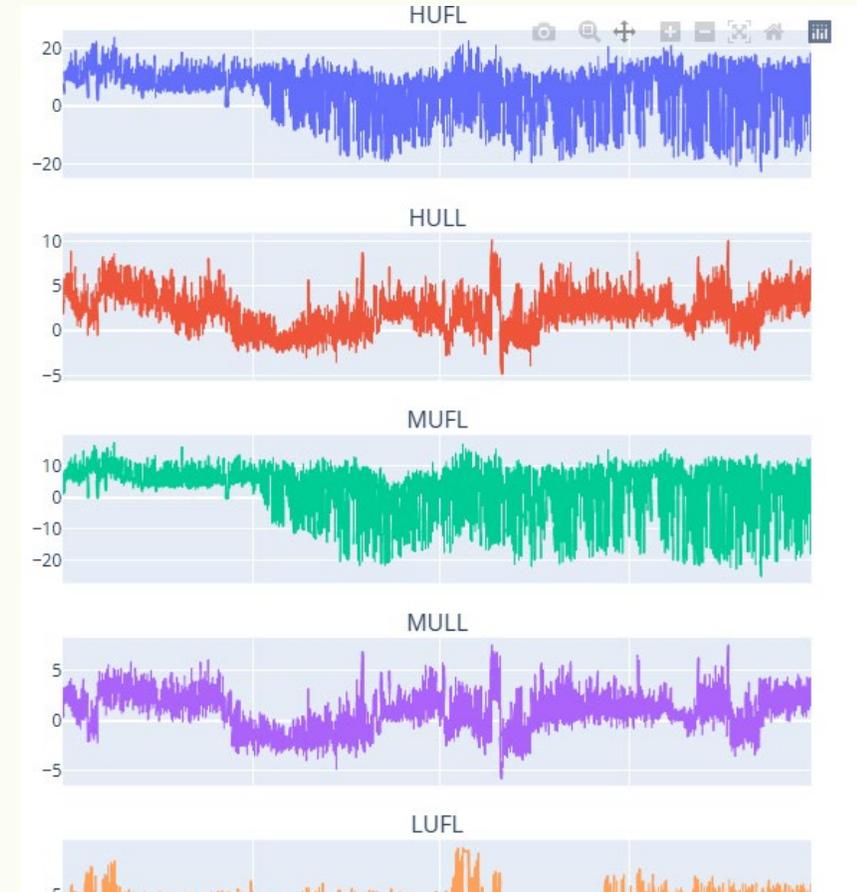
特徴量数(次元数, カラム数, etc...)

時間長 ↓

	A	B	C	D	E	F	G	H
1		気化器圧力_F	気化器圧力_S	気化器圧力_M	気化器圧力_N	気化器液面レ	気化器液面レ	気化器液
2	2020/04/13 00:00:00	127.983294	127.88392	18.728267	AUT	0.70020428	0.7	21876.9
3	2020/04/13 00:00:01	128.262272	127.807691	18.728267	AUT	0.7008654	0.7	21876.9
4	2020/04/13 00:00:02	127.899077	127.991074	19.0793245	AUT	0.69974009	0.7	21918.3
5	2020/04/13 00:00:03	127.622218	127.990697	18.7722276	AUT	0.69997	0.7	21892.1
6	2020/04/13 00:00:04	127.577243	127.903788	18.6051732	AUT	0.70093866	0.7	21885.6
7	2020/04/13 00:00:05	128.183048	127.934594	18.5832803	AUT	0.69965113	0.7	21927.6
8	2020/04/13 00:00:06	128.245161	127.901869	18.7103013	AUT	0.69944281	0.7	21894.1

一般にデータは等時間間隔でサンプリング

→
グラフ化



詳細は「ごちきか」参照: <https://gochikika.ntt.com>

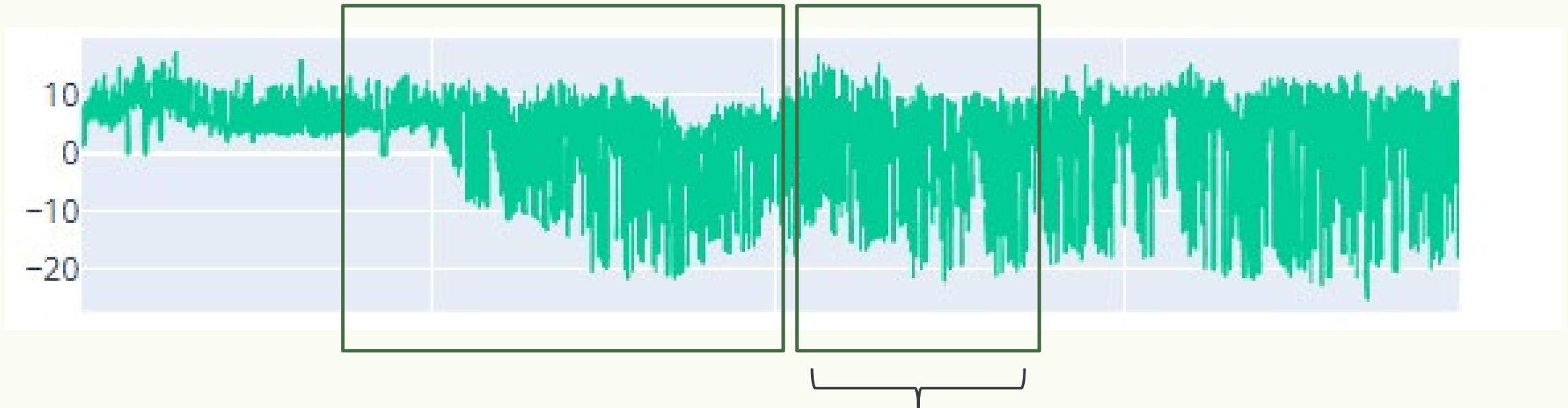
準備 -時系列データの問題設定-

入力ベクトル

(窓枠, lookback window,
Backcast period, etc...)

予測対象

(Forecast Period, etc...)

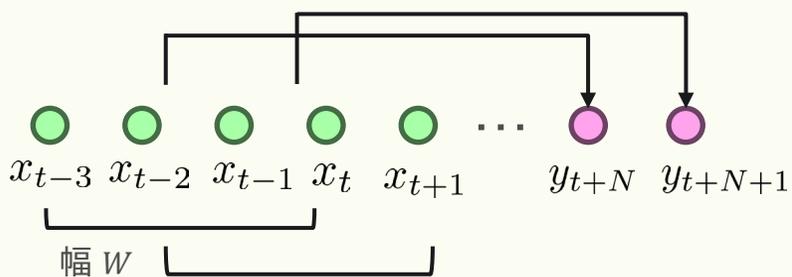


- 単変量: 特徴量数 = 1
- 多変量: 特徴量数 > 1

この長さを、窓幅とかHorizonとか

準備

Point



新しい観測値を得るたびに
N点先時刻を予測

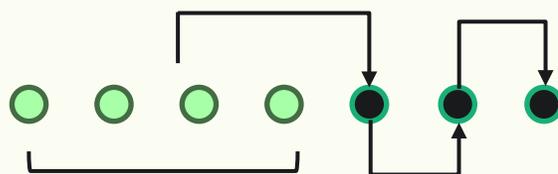
モデル: \mathcal{M}

$$y_{t+N} = \mathcal{M}(\mathbf{x}_{t-W+1:t})$$

$$y_{t+N+1} = \mathcal{M}(\mathbf{x}_{t-W:t+1})$$

⋮

Autoregressive



1期先予測結果を新しい入力値
として、さらに次の時刻を予測

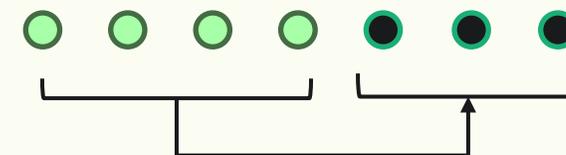
$$\hat{y}_{t+1} = \mathcal{M}(\mathbf{x}_{t-W+1:t})$$

$$\hat{y}_{t+2} = \mathcal{M}(\mathbf{x}_{t-W:t}, \hat{y}_t)$$

⋮

今回の問題設定

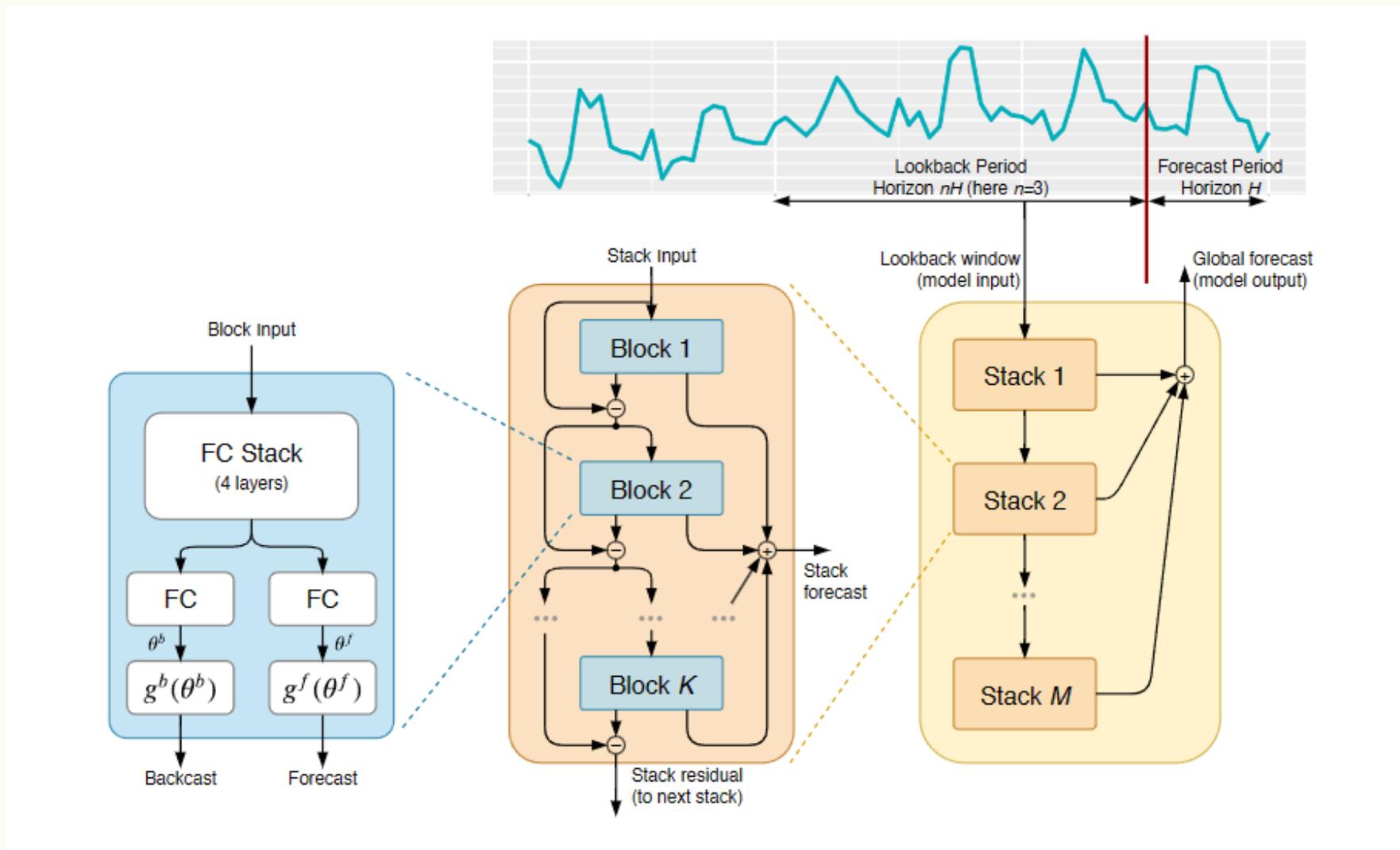
Non-autoregressive (generative)



N期先までまとめて一度に予測
最近の流行り(?)

$$\hat{\mathbf{y}}_t = \mathcal{M}(\mathbf{x}_t)$$

アーキテクチャ



Block

Stackの基本構成要素(全結合MLP)

l 番目のブロックにおいて、入力を与えられたとき

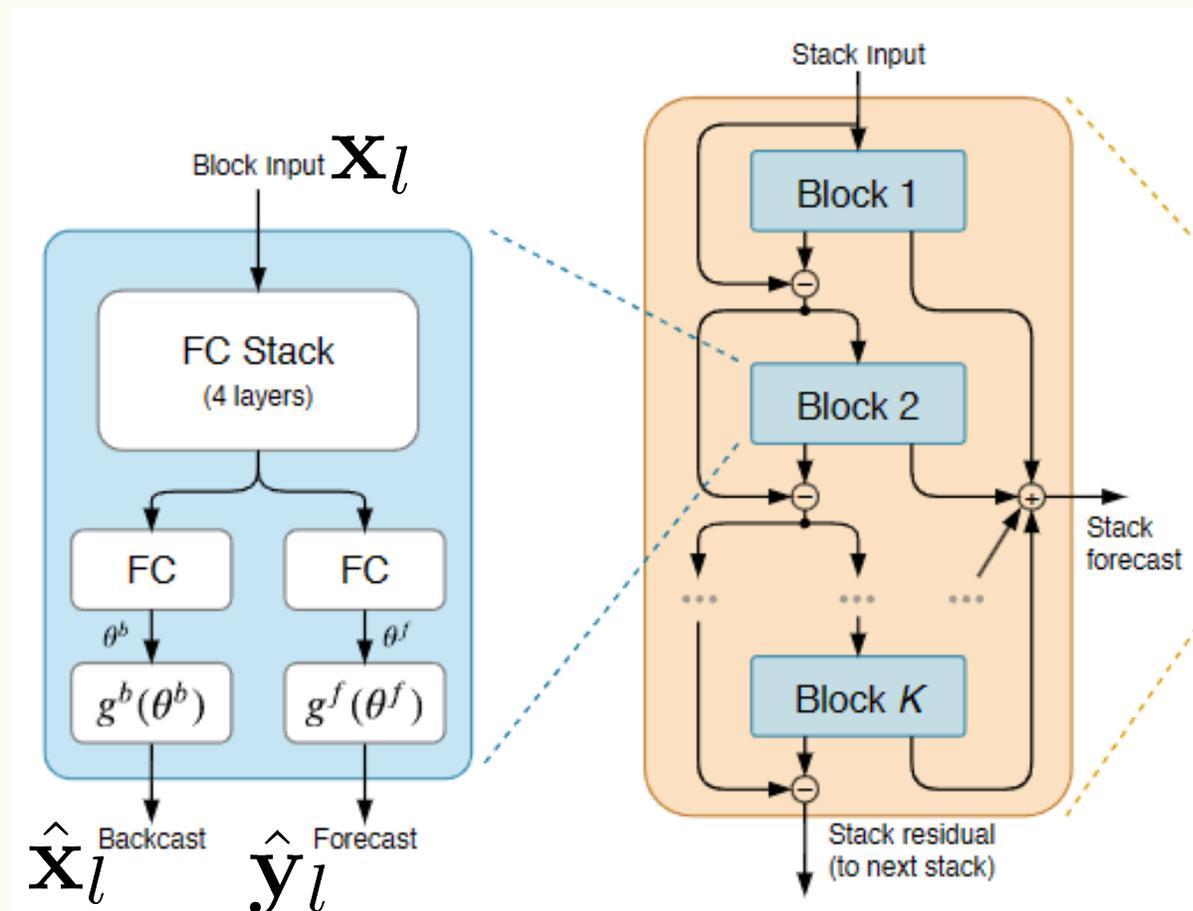
Backcast $\hat{\mathbf{X}}_l$

もともとの入力 \mathbf{x}_l を再構成

Forecast $\hat{\mathbf{y}}_l$

このブロックにおける H 期先予測

分岐したヘッド g^b, g^f は任意だがここでは線形結合



Doubly Residual Stacking

Backcast/Forecastの両方について残差構造

Backcast

$$\mathbf{X}_l = \mathbf{X}_{l-1} - \hat{\mathbf{X}}_{l-1},$$

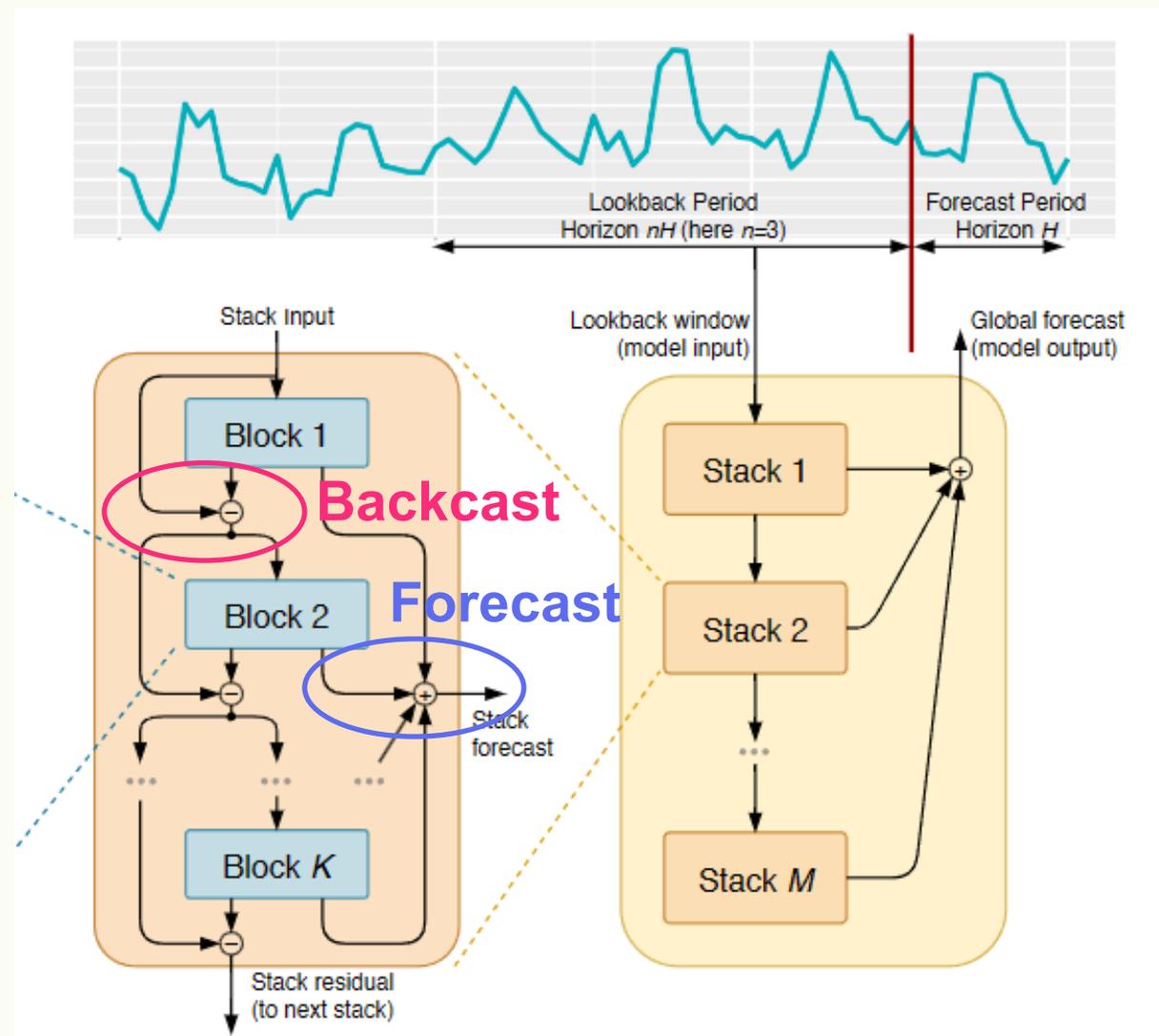
近似ができない(=表現しきれない)
信号成分は次のブロックへ

➡ 予測タスクが簡単になる

Forecast

各ブロックの部分予測を足し合わせる

$$\hat{y} = \sum_l \hat{y}_l$$



解釈性

ヘッドには任意性があり、時系列の特徴を陽に組み込むことも可能
出力される分解波形に制約をかける

トレンド: 単調変化/変化がなだらか

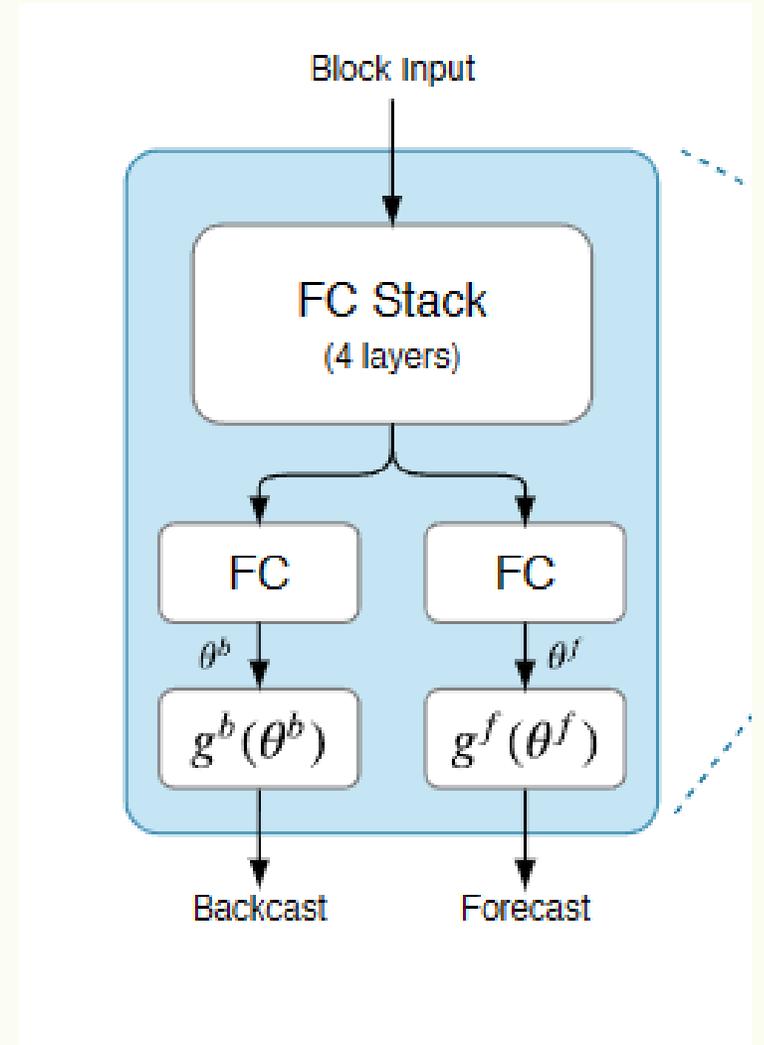
次数 $p \leq 3$ の多項式

$$\hat{y}_{s,l} = \sum_{i=0}^p \theta_{s,l,i}^f t^i$$

季節性/周期性

フーリエ級数

$$\hat{y}_{s,l} = \sum_{i=0}^{\lfloor H/2-1 \rfloor} \theta_{s,l,i}^f \cos(2\pi it) + \theta_{s,l,i+\lfloor H/2 \rfloor}^f \sin(2\pi it)$$



その他精度を上げるためのテクニック

アンサンブル

M4コンペの優勝手法はアンサンブルを使っており、提案手法でもアンサンブルを使用

- モデルの学習メトリクス4種類
 - sMAPE, MASE, MAPE, sMAPEの分母から予測 \hat{y} の項を抜いたもの
- 入力長を, 2H, 3H, ..., 7H
 - Hは予測長
- バギング

$$\text{sMAPE} = \frac{200}{H} \sum_{i=1}^H \frac{|y_{T+i} - \hat{y}_{T+i}|}{|y_{T+i}| + |\hat{y}_{T+i}|},$$

$$\text{MAPE} = \frac{100}{H} \sum_{i=1}^H \frac{|y_{T+i} - \hat{y}_{T+i}|}{|y_{T+i}|},$$

$$\text{MASE} = \frac{1}{H} \sum_{i=1}^H \frac{|y_{T+i} - \hat{y}_{T+i}|}{\frac{1}{T+H-m} \sum_{j=m+1}^{T+H} |y_j - y_{j-m}|},$$

$$\text{OWA} = \frac{1}{2} \left[\frac{\text{sMAPE}}{\text{sMAPE}_{\text{Naïve2}}} + \frac{\text{MASE}}{\text{MASE}_{\text{Naïve2}}} \right].$$

M4コンペでの指標

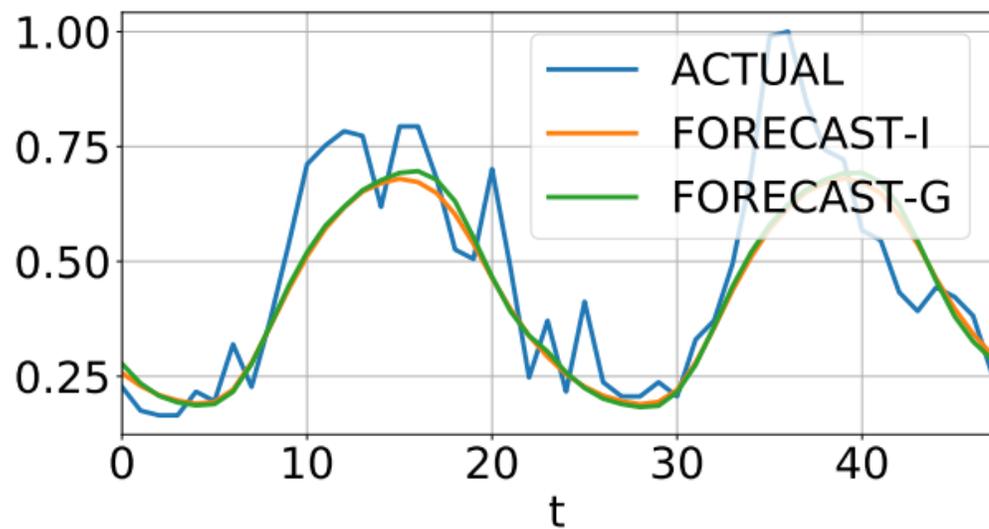
結果

M4 Average (100,000)			M3 Average (3,003)		TOURISM Average (1,311)	
	sMAPE	OWA		sMAPE		MAPE
Pure ML	12.894	0.915	Comb S-H-D	13.52	ETS	20.88
Statistical	11.986	0.861	ForecastPro	13.19	Theta	20.88
ProLogistica	11.845	0.841	Theta	13.01	ForePro	19.84
ML/TS combination	11.720	0.838	DOTM	12.90	Stratometrics	19.52
DL/TS hybrid	11.374	0.821	EXP	12.71	LeeCBaker	19.35
N-BEATS-G	11.168	0.797		12.47		18.47
N-BEATS-I	11.174	0.798		12.43		18.97
N-BEATS-I+G	11.135	0.795		12.37		18.52

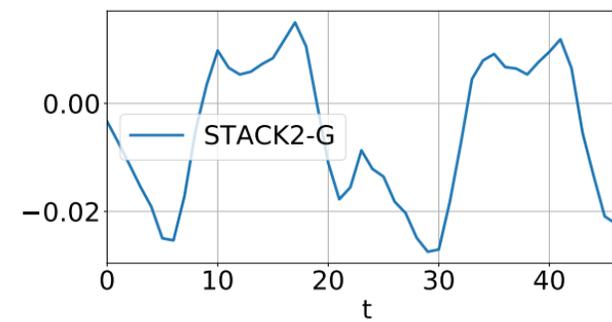
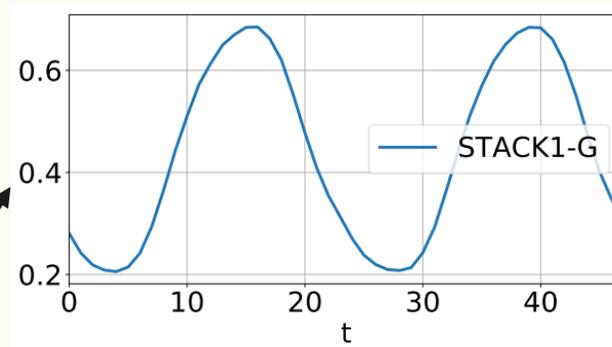
- G: Generic
- I: Interpretable
- I+G: アンサンブル

結果

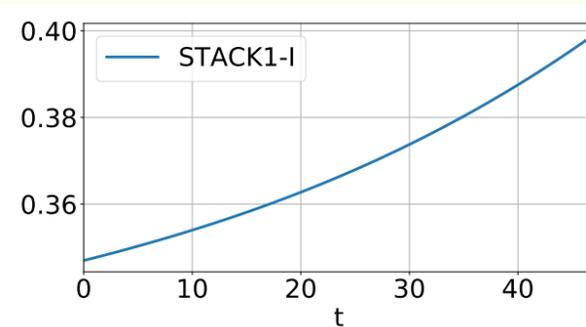
全体の予測結果(M4)



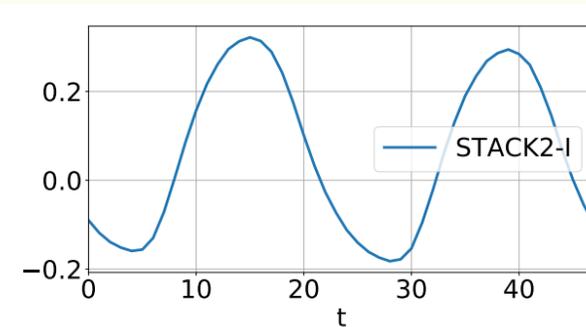
各stackの予測(generic)



Stack-Trend



Stack-Seasonal



N-HiTS: Neural Hierarchical Interpolation for Time Series Forecasting

- 著者: Cristian Challu¹, Kin G. Olivares¹, Boris N. Oreshkin², Federico Garza³, Max Mergenthaler-Canseco³, Artur Dubrawski¹
 - 1: Carnegie Mellon University, 2: Unity Technologies, 3: Nixtla
- 投稿日付: 2022/11/19
- 学会: AAI 2023
- 論文へのリンク: <https://arxiv.org/abs/2201.12886>

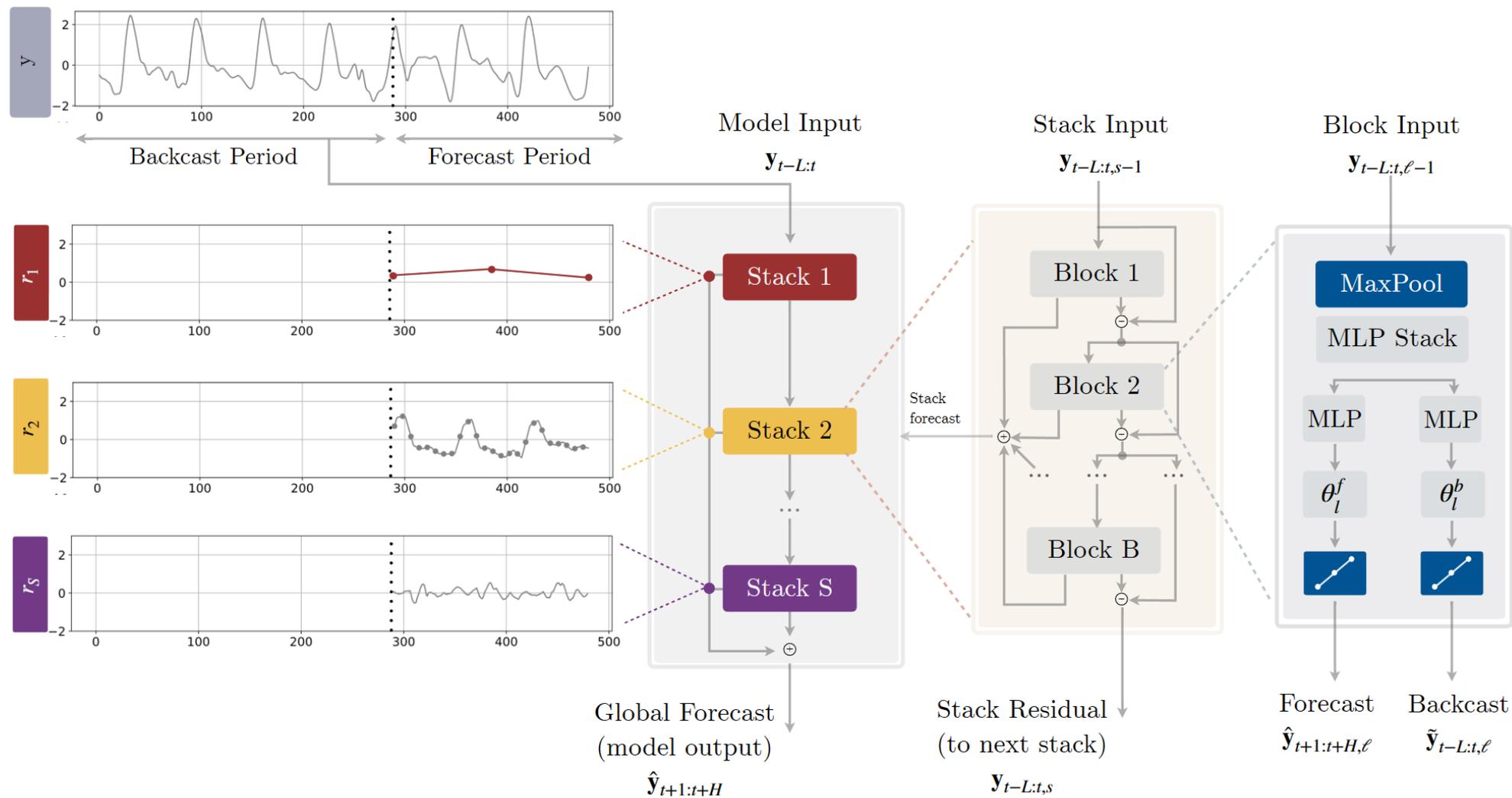
概要

- N-BEATSをベースとした長期時系列予測モデル
 - 階層的補間とマルチレート入力処理により、時系列の異なる周波数ごとに部分系列を出力
- Informerに比べて25%の精度改善。50倍の速度向上

Contributions

- **Multi-Rate Data Sampling:** FC層の前段にサブサンプリングレイヤーを組み込むことで、メモリ使用量と計算量を削減
- **Hierarchical Interpolation:** NNの予測次元を削減し、マルチスケール階層的補間により最終層の時間スケールを一致させることで、マルチステップ予測の滑らかさを担保する。
- **N-HiTS architecture:** 入力サンプリングレートと出力補間のスケールを、ブロック間で階層的に同期させる。それぞれのブロックが、時系列信号のそれぞれの周波数帯の予測に特化するようにさせる。

アーキテクチャ



Multi-Rate Signal Sampling

それぞれのブロックの入力にMaxPoolingを追加

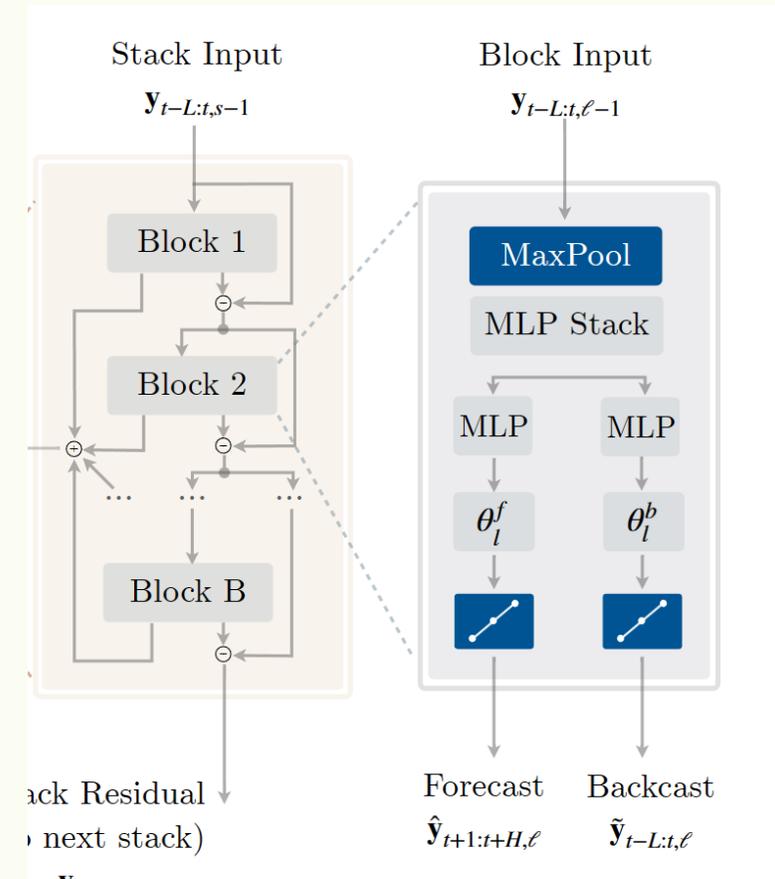
$$\mathbf{y}_{t-L:t,l}^p = \text{MaxPool}(\mathbf{y}_{t-L:t,l}, k_l)$$

k_l : カーネルサイズ

特定のスケールを持つ入力成分に注目することを期待

カーネルサイズが大きい=高周期(小さい時間スケール)を除去

加えて、Poolingで入力サイズも削減されるため、
メモリ使用量や計算量的にも有利、
学習パラメーターも減り過学習抑制



Hierarchical Interpolation

各Stack/Blockの出力次元 = 予測長 H

⇒ H が長くなるほど計算量も上がる

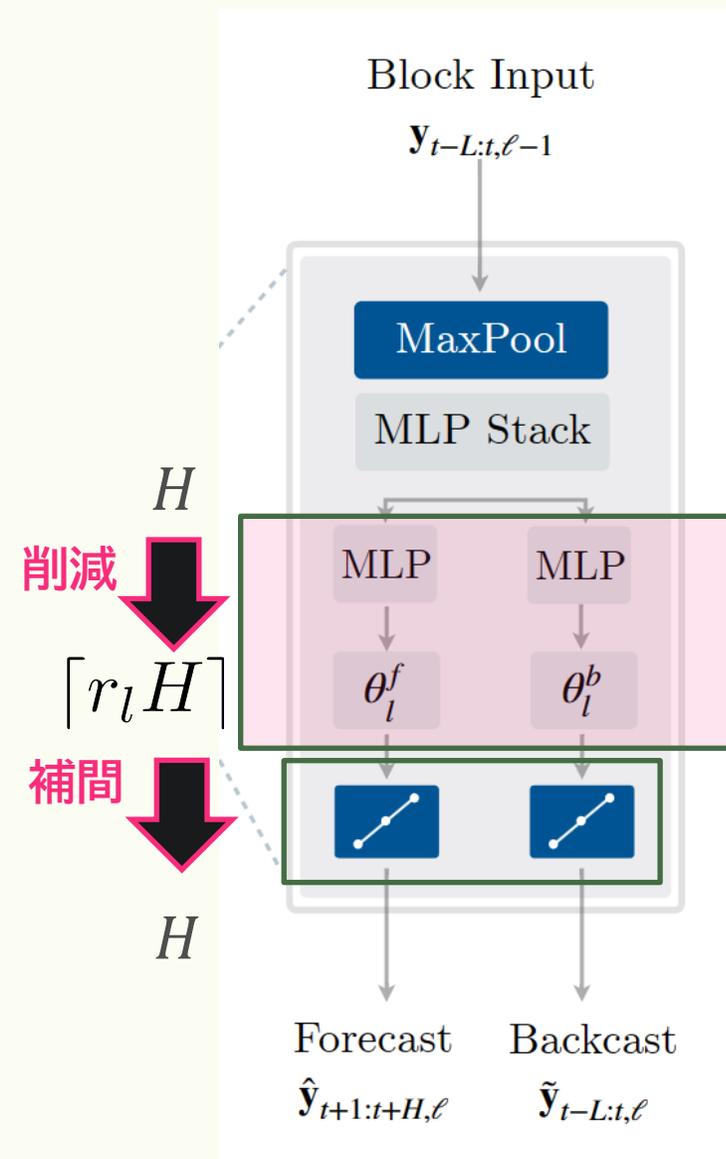
Temporal interpolation

expressiveness ratio, r_l をもちいて、出力パラメータ数を制限

$$|\theta_l^f| = \lceil r_l H \rceil$$

ただし、予測系列 \hat{y} 自体の長さは H 必要なので、関数 g で補間

$$\hat{y}_{\tau,l} = g(\tau, \theta_l^f), \quad \forall \tau \in [t+1, \dots, t+H]$$



※特に論文中では言及がないが、Backcastもたぶん同様

Hierarchical Interpolation

補間関数 g として、

- 近傍点をそのまま使う
- 線形
- 二次多項式

を論文中では提案

例えば線形補完:

$$g(\tau, \theta) = \theta[t_1] + \left(\frac{\theta[t_2] - \theta[t_1]}{t_2 - t_1} \right) (\tau - t_1) \quad (4)$$

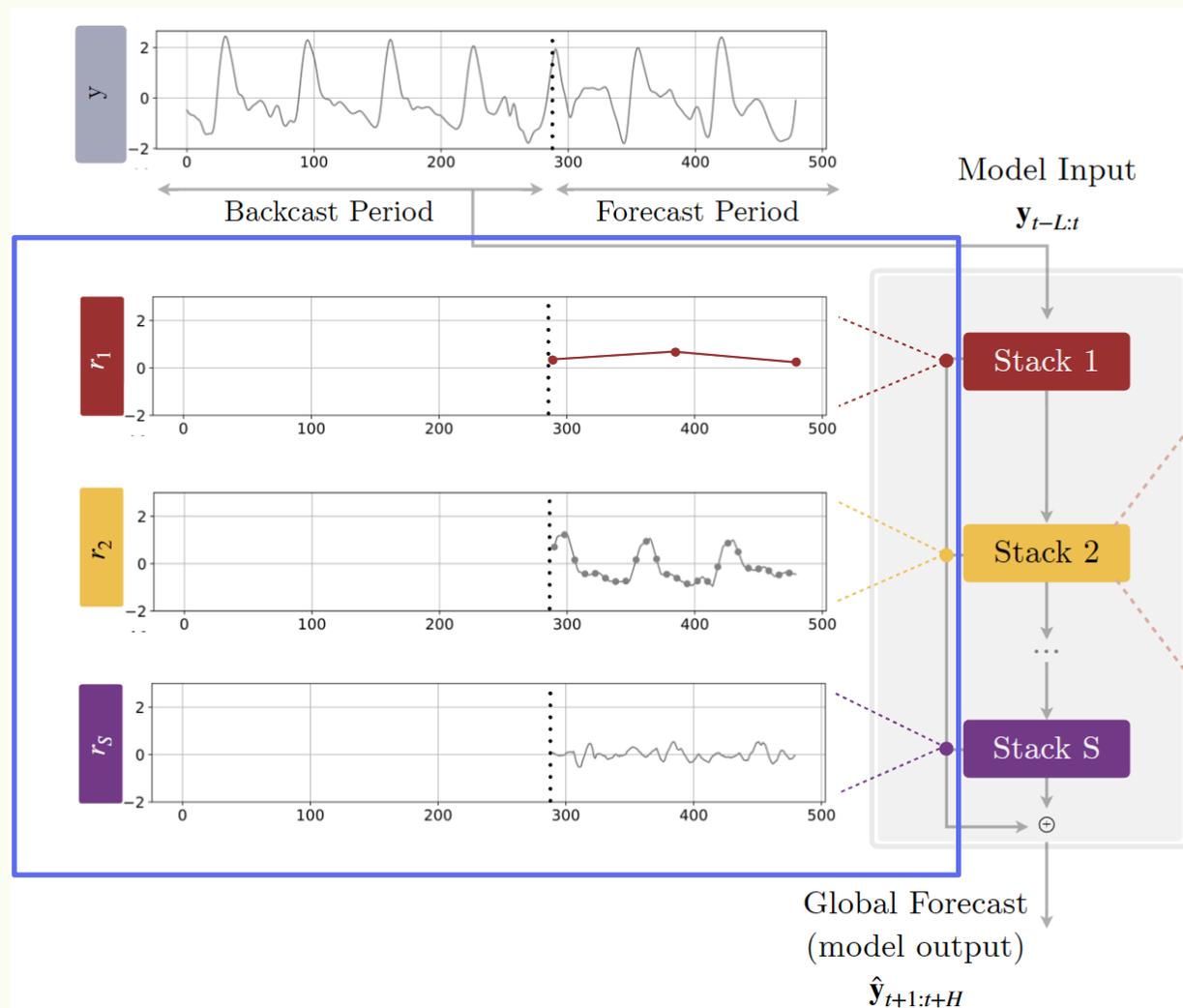
$$t_1 = \arg \min_{t \in \mathcal{T}: t \leq \tau} \tau - t, \quad t_2 = t_1 + 1/r_\ell.$$

入力に近いBlockに、小さい r_l と大きい k_l を設定



より補間が強く効いた低粒度な信号

階層が深くなるほど細かい変動を捉える



結果

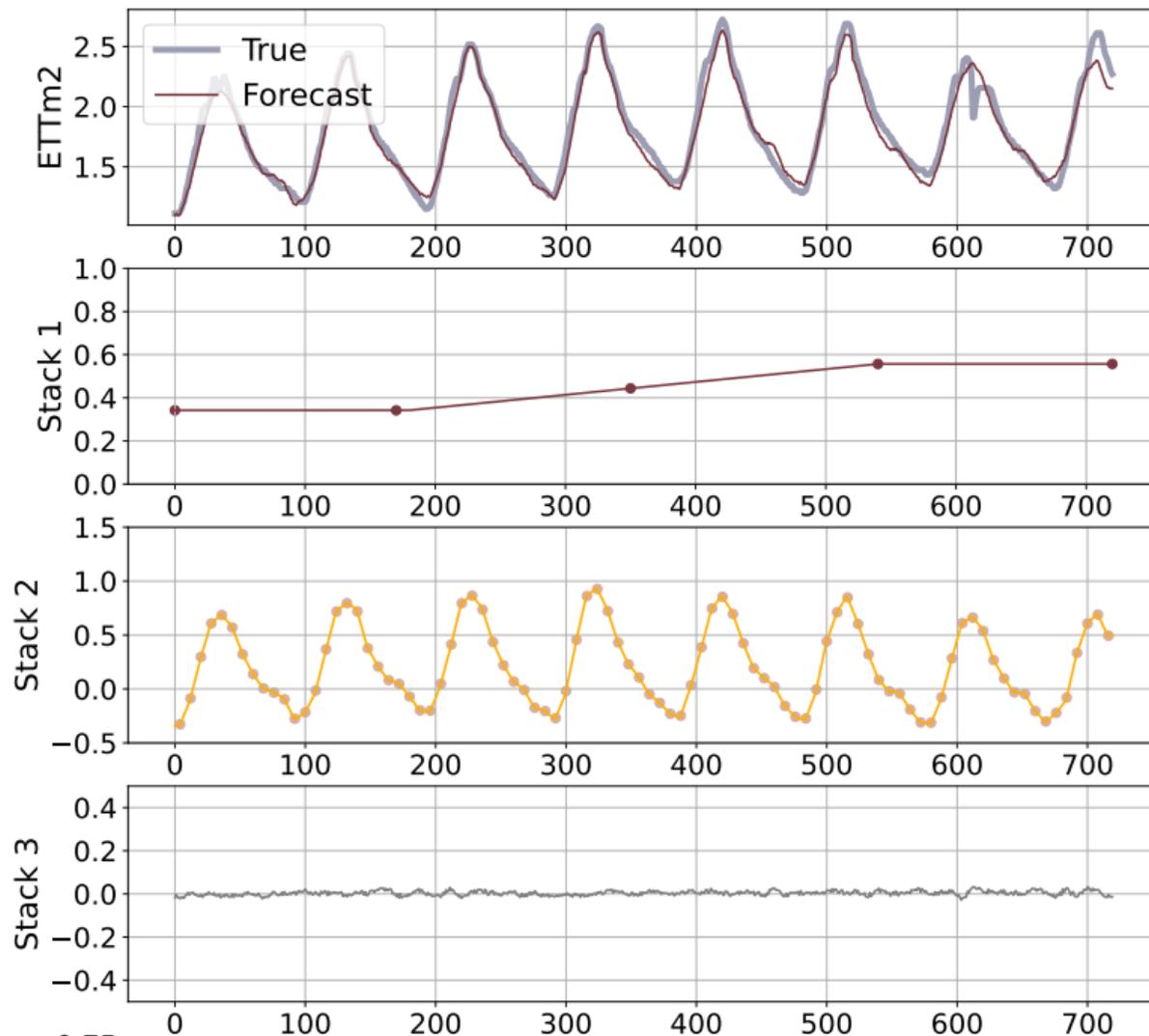
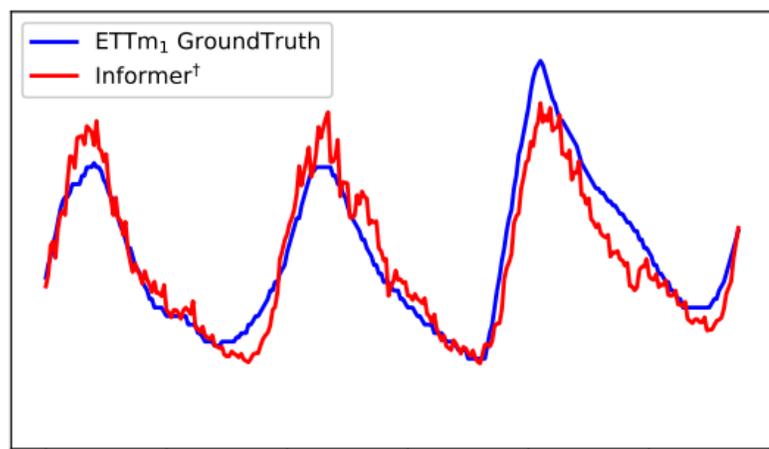
多くのデータセットでSOTA. MSEで平均14%, MAEで平均16%の改善

H.	N-HiTS (Ours)		N-BEATS		FEDformer		Autoformer		Informer		LogTrans		Reformer		DilRNN		ARIMA		
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	
ETTm2	96	0.176	0.255	0.184	0.263	0.203	0.287	0.255	0.339	0.365	0.453	0.768	0.642	0.658	0.619	0.343	0.401	0.225	0.301
	192	0.245	0.305	0.273	0.337	0.269	0.328	0.281	0.340	0.533	0.563	0.989	0.757	1.078	0.827	0.424	0.468	0.298	0.345
	336	0.295	0.346	0.309	0.355	0.325	0.366	0.339	0.372	1.363	0.887	1.334	0.872	1.549	0.972	0.632	1.083	0.370	0.386
	720	0.401	0.413	0.411	0.425	0.421	0.415	0.422	0.419	3.379	1.388	3.048	1.328	2.631	1.242	0.634	0.594	0.478	0.445
ECL	96	0.147	0.249	0.145	0.247	0.183	0.297	0.201	0.317	0.274	0.368	0.258	0.357	0.312	0.402	0.233	0.927	1.220	0.814
	192	0.167	0.269	0.180	0.283	0.195	0.308	0.222	0.334	0.296	0.386	0.266	0.368	0.348	0.433	0.265	0.921	1.264	0.842
	336	0.186	0.290	0.200	0.308	0.212	0.313	0.231	0.338	0.300	0.394	0.280	0.380	0.350	0.433	0.235	0.896	1.311	0.866
	720	0.243	0.340	0.266	0.362	0.231	0.343	0.254	0.361	0.373	0.439	0.283	0.376	0.340	0.420	0.322	0.890	1.364	0.891
Exchange	96	0.092	0.202	0.098	0.206	0.139	0.276	0.197	0.323	0.847	0.752	0.968	0.812	1.065	0.829	0.383	0.45	0.296	0.214
	192	0.208	0.322	0.225	0.329	0.256	0.369	0.300	0.369	1.204	0.895	1.040	0.851	1.188	0.906	1.123	0.834	1.056	0.326
	336	0.301	0.403	0.493	0.482	0.426	0.464	0.509	0.524	1.672	1.036	1.659	1.081	1.357	0.976	1.612	1.051	2.298	0.467
	720	0.798	0.596	1.108	0.804	1.090	0.800	1.447	0.941	2.478	1.310	1.941	1.127	1.510	1.016	1.827	1.131	20.666	0.864
TrafficI	96	0.402	0.282	0.398	0.282	0.562	0.349	0.613	0.388	0.719	0.391	0.684	0.384	0.732	0.423	0.580	0.308	1.997	0.924
	192	0.420	0.297	0.409	0.293	0.562	0.346	0.616	0.382	0.696	0.379	0.685	0.390	0.733	0.420	0.739	0.383	2.044	0.944
	336	0.448	0.313	0.449	0.318	0.570	0.323	0.622	0.337	0.777	0.420	0.733	0.408	0.742	0.420	0.804	0.419	2.096	0.960
	720	0.539	0.353	0.589	0.391	0.596	0.368	0.660	0.408	0.864	0.472	0.717	0.396	0.755	0.423	0.695	0.372	2.138	0.971
Weather	96	0.158	0.195	0.167	0.203	0.217	0.296	0.266	0.336	0.300	0.384	0.458	0.490	0.689	0.596	0.193	0.245	0.217	0.258
	192	0.211	0.247	0.229	0.261	0.276	0.336	0.307	0.367	0.598	0.544	0.658	0.589	0.752	0.638	0.255	0.306	0.263	0.299
	336	0.274	0.300	0.287	0.304	0.339	0.380	0.359	0.395	0.578	0.523	0.797	0.652	0.064	0.596	0.329	0.360	0.330	0.347
	720	0.351	0.353	0.368	0.359	0.403	0.428	0.419	0.428	1.059	0.741	0.869	0.675	1.130	0.792	0.521	0.495	0.425	0.405
ILI	24	1.862	0.869	1.879	0.886	2.203	0.963	3.483	1.287	5.764	1.677	4.480	1.444	4.400	1.382	4.538	1.449	5.554	1.434
	36	2.071	0.934	2.210	1.018	2.272	0.976	3.103	1.148	4.755	1.467	4.799	1.467	4.783	1.448	3.709	1.273	6.940	1.676
	48	2.134	0.932	2.440	1.088	2.209	0.981	2.669	1.085	4.763	1.469	4.800	1.468	4.832	1.465	3.436	1.238	7.192	1.736
	60	2.137	0.968	2.547	1.057	2.545	1.061	2.770	1.125	5.264	1.564	5.278	1.560	4.882	1.483	3.703	1.272	6.648	1.656

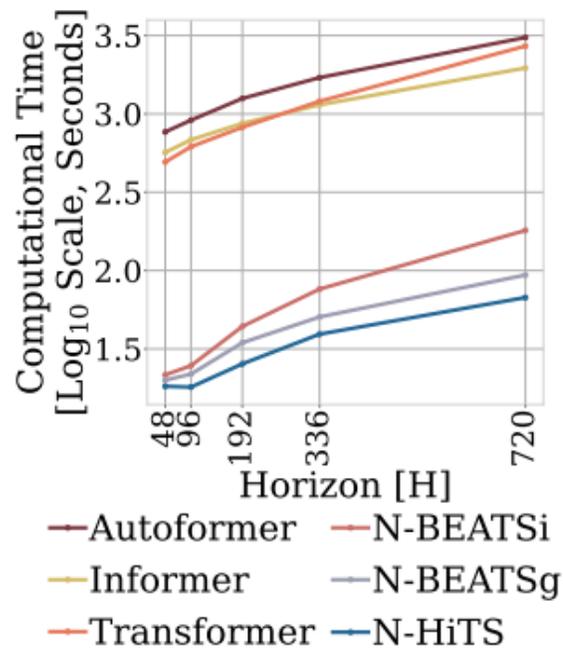
結果

- 滑らかな長期時系列予測が可能
 - 表現能力が適度に制限されているため
 - 例えばInformerは高い表現能力で、ギザギザした予測をしてしまう
- 階層ごとにどの成分を予測したか解釈可能

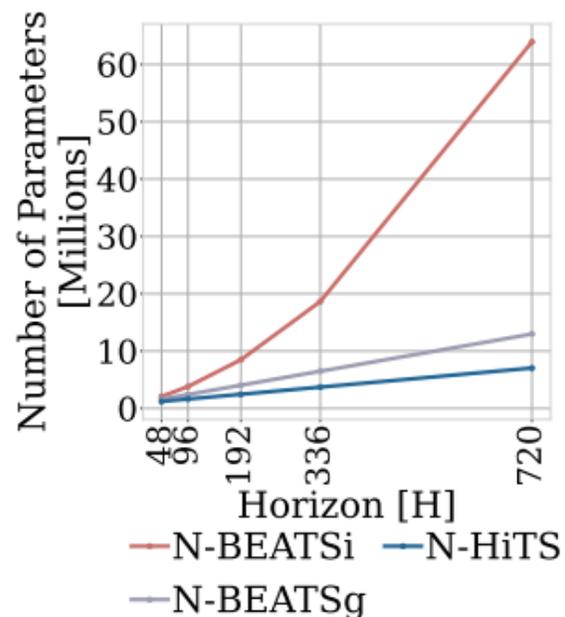
参考: Informerより引用



結果



(a) *Time Efficiency*



(b) *Memory Efficiency*

そもそも200点先とかの予測を必要とする場面が少ない気もするため、
実用上はN-BEATSの計算時間でもよさそう

まとめ

- N-BEATS, N-HiTSというMLPベースの階層型時系列予測モデルを紹介
 - Backcast/Forecastをおこなうブロックを残差を取りながら積み上げることで、長周期～短周期成分を各層で予測。これにより各タスク分解が行われ、予測精度が向上
 - また、予測に対する解釈性も高い。
- 特にN-BEATSはベースラインとして使われ、拡張も容易で多くの後続研究があるため、押さえておくといいかもしれない
- ライブラリ実装もあり(neuralforecast)
- ただしいずれも単変量しか扱えない
 - N-BEATSx という後続研究では多変量入力(外生変数)に対応した