High-Dimensional Multivariate Forecasting with Low-Rank Gaussian Copula Processes

David Salinas [1], Michael Bohlke-Schneider [2], Laurent Callot [2], Roberto Medico [3], \
Jan Gasthaus [2]

[1]Naverlabs Europe, work done at Amazon Research

[2]Amazon Research

[3]Ghent University, work done at Amazon Research

July 18, 2019

Section 1

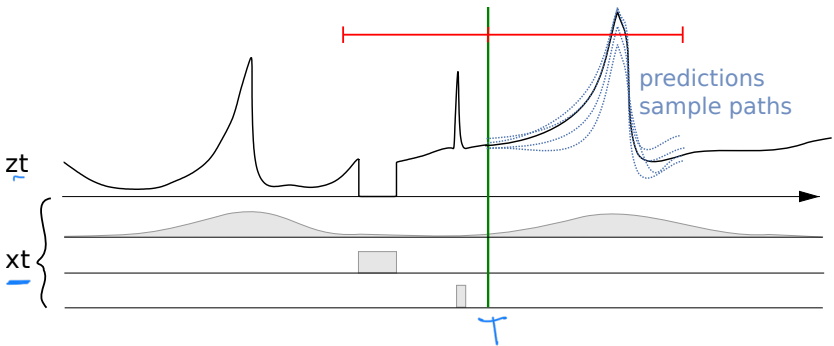
(Multivariate) Forecasting

General Setup

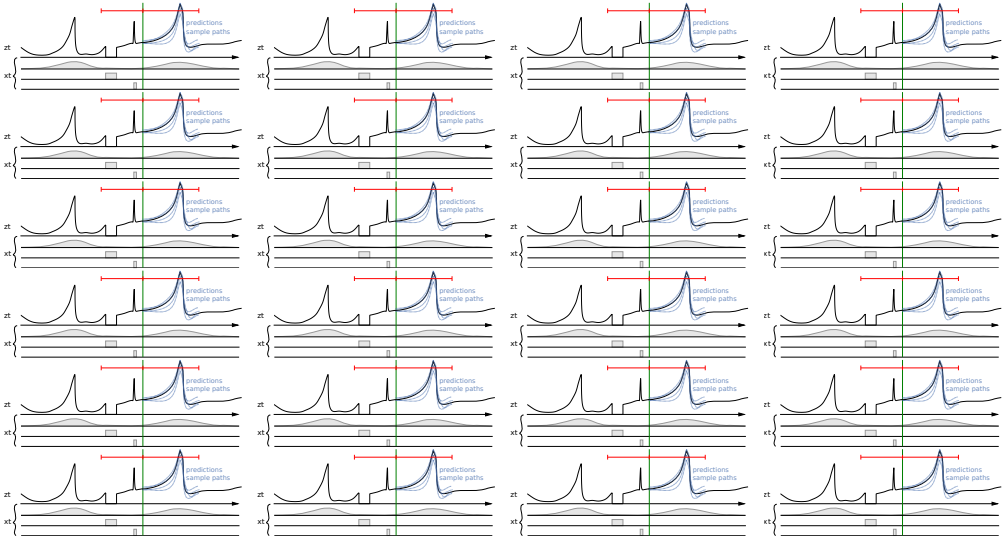
Predict the future behavior of a time series given its past

$$z_1, z_2, \ldots, z_T \Longrightarrow P(z_{T+1}, z_{T+2}, \ldots z_{T+\tau})$$

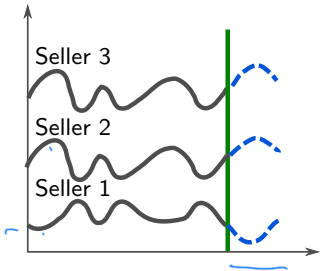
Earlier



Many times series are often available

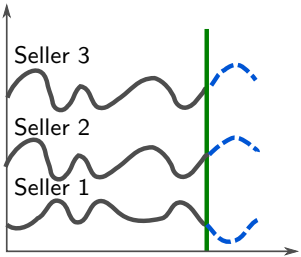


Multivariate time series



- Multivariate time series is sometimes used to refer to forecasting *independently* values of multiple time series
- In this paper, we are interested in forecasting *jointly* values of multiple series to model some potential noise relations
- If a future value of one series is higher, we would like other to be higher if positively correlated or lower if negative

Why modeling correlation matters



- Consider the time series of several seller sales, they may be negatively correlated (when competing) or positively correlated (selling in the same area with no competition)
- Neglecting this effect causes to underestimate the variance when considering sellers in junction
- If a future value of one series is higher, we would like other to be higher if positively correlated or lower if negative
- Applications: demand forecasting, finance, anomaly detection...

Previous work

- Very rich literature on the subject, most methods comes from Statistics/Econometry
- VAR, GARCH, multivariate ISSM, ...
- Limitation: quadratic number of parameters as dimension grows
- Modest dimensions (typically less than 100)
- Hypothesis to bypass this issue: homoskedasticity [Toubeau19], diagonal noise, ...

Multivariate Model

Instead of

$$z_1, z_2, \ldots, z_T \Longrightarrow P(z_{T+1}, z_{T+2}, \ldots z_{T+\tau})$$

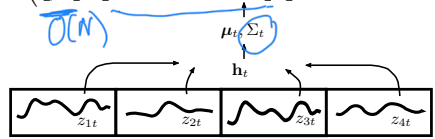
learn the joint distribution:

$$\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_T \Longrightarrow P(\mathbf{z}_{T+1}, \mathbf{z}_{T+2}, \ldots \mathbf{z}_{T+\tau})$$

where $\mathbf{z}_t \in \mathbb{R}^N$

Multivariate: a simple LSTM vectorial approach

$$\mathcal{N}\left(\begin{bmatrix}\mu_{1t}\\ \mu_{2t}\\ \mu_{3t}\\ \mu_{4t}\end{bmatrix}, \begin{bmatrix}l_{11t} & 0 & 0 & 0\\ l_{21t} & l_{22t} & 0 & 0\\ l_{31t} & l_{32t} & l_{33t} & 0\\ l_{41t} & l_{33t} & l_{34t} & l_{44t}\end{bmatrix} \begin{bmatrix}l_{11t} & 0 & 0 & 0\\ l_{21t} & l_{22t} & 0 & 0\\ l_{31t} & l_{32t} & l_{33t} & 0\\ l_{41t} & l_{33t} & l_{34t} & l_{44t}\end{bmatrix}^T\right)$$



- Learn an autoregressive model with an LSTM with state $\mathbf{h}_t$
- Input: all series observations $z_{1t}, \ldots, z_{Nt}$
- Output: Normal distribution of next time-step $p(\mathbf{z}_{t+1}|\mathbf{h}_t) \sim \mathcal{N}(\boldsymbol{\mu}_t, \Sigma_t)$

Multivariate Model

- The transition dynamics are parametrized using a LSTM-RNN.

$$\mathbf{h}_t = \varphi_{\theta_h}(\mathbf{h}_{t-1}, \mathbf{z}_{t-1}) \in \mathbb{R}^k.$$

- We can factorize **the joint distribution of the observations** as:

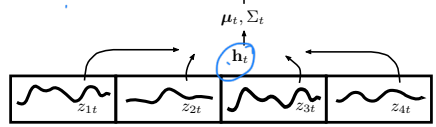
$$p(\mathbf{z}_1, \dots \mathbf{z}_{T+\tau}) = \prod_{t=1}^{T+\tau} p(\mathbf{z}_t | \mathbf{z}_1, \dots, \mathbf{z}_{t-1}) = \prod_{t=1}^{T+\tau} p(\mathbf{z}_t | \mathbf{h}_t).$$

- We train by minimizing the negative, Multivariate Gaussian **log-likelihood**:

$$-\log p(\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_T) = -\sum_{t=1}^{T} \log p(\mathbf{z}_t | \mathbf{h}_t).$$

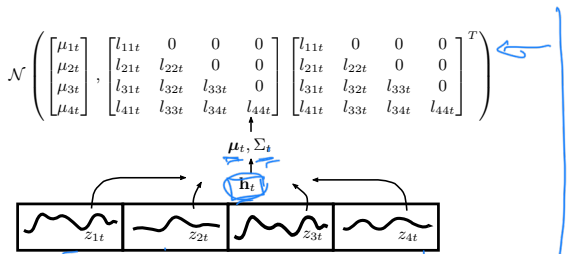
Multivariate Gaussian model: time and space complexity

$$\mathcal{N}\left(\begin{bmatrix}\mu_{1t}\\ \mu_{2t}\\ \mu_{3t}\\ \mu_{4t}\end{bmatrix}, \begin{bmatrix}l_{11t} & 0 & 0 & 0\\ l_{21t} & l_{22t} & 0 & 0\\ l_{31t} & l_{32t} & l_{33t} & 0\\ l_{41t} & l_{33t} & l_{34t} & l_{44t}\end{bmatrix}\begin{bmatrix}l_{11t} & 0 & 0 & 0\\ l_{21t} & l_{22t} & 0 & 0\\ l_{31t} & l_{32t} & l_{33t} & 0\\ l_{41t} & l_{33t} & l_{34t} & l_{44t}\end{bmatrix}^T\right)$$



- Projecting $\mathbf{z}_t \in \mathbb{R}^N$ to $\mathbf{h}_t \in \mathbb{R}^k$: $O(Nk)$ parameters
- Projecting $\mathbf{h}_t$ to likelihood parameters of $p(\mathbf{z}_t|\mathbf{h}_t)$: $O(kN^2)$ parameters

Multivariate Gaussian model: time and space complexity



- Projecting $\mathbf{z}_t \in \mathbb{R}^N$ to $\mathbf{h}_t \in \mathbb{R}^k$: $O(Nk)$ parameters
- Projecting $\mathbf{h}_t$ to likelihood parameters of $p(\mathbf{z}_t|\mathbf{h}_t)$: $O(kN^2)$ parameters
- Hard to scale with large N

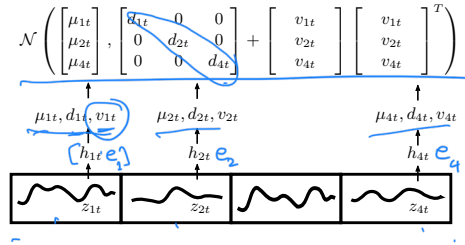
Issues with the Multivariate Gaussian Model

1. Large number of parameters, expensive to compute
2. All time series at training time need to be available during inference
3. Different scales of time series and non-Gaussian data

Section 2

Low-rank Gaussian Copula Process

Addressing efficiency issue: Low-rank Gaussian Process Parametrization



$$\mu_i(\mathbf{h}_{i,t}) = \tilde{\mu}(\mathbf{y}_{i,t}), \qquad d_i(\mathbf{h}_{i,t}) = \tilde{d}(\mathbf{y}_{i,t}), \qquad \mathbf{v}_i(\mathbf{h}_{i,t}) = \tilde{\mathbf{v}}(\mathbf{y}_{i,t}).$$

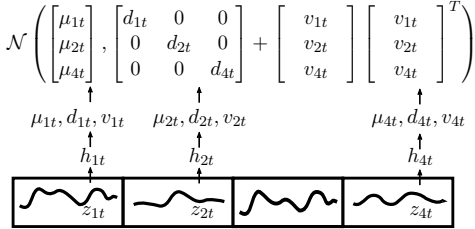
- $\mathbf{y}_{i,t} = [\mathbf{h}_{i,t}; \mathbf{e}_i]^T \in \mathbb{R}^{p\times 1}$, with $\mathbf{e}_i$ known features or learned embeddings.
- These shared functions can **parametrize a Gaussian Process** $g_t(\mathbf{y}_{i,t})$.
- $g_t \sim \mathsf{GP}(\tilde{\mu}(\cdot), k(\cdot,\cdot))$, with $k(\mathbf{y}, \mathbf{y}') = \mathbb{1}_{\mathbf{y}=\mathbf{y}'}\tilde{d}(\mathbf{y}) + \tilde{\mathbf{v}}(\mathbf{y})^T\tilde{\mathbf{v}}(\mathbf{y}')$.

Low-rank Approximation

$$\Sigma(\mathbf{h}_t) = \begin{bmatrix} d_1(\mathbf{h}_{1,t}) & & 0 \\ & \ddots & \\ 0 & & d_N(\mathbf{h}_{N,t}) \end{bmatrix} + \begin{bmatrix} \mathbf{v}_1(\mathbf{h}_{1,t}) \\ \dots \\ \mathbf{v}_N(\mathbf{h}_{N,t}) \end{bmatrix} \begin{bmatrix} \mathbf{v}_1(\mathbf{h}_{1,t}) \\ \dots \\ \mathbf{v}_N(\mathbf{h}_{N,t}) \end{bmatrix}^T = D_t + V_t V_t^T.$$

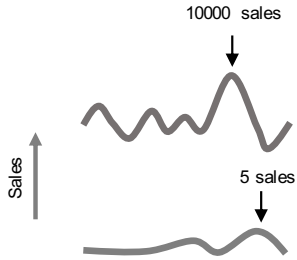
- $D_t \in \mathbb{R}^{N \times N}$ is diagonal. $V_t \in \mathbb{R}^{N \times r}$. $O(N \times r)$ parameters.
- $V_t V_t^T$ is a low-rank matrix with **rank hyperparameter** $r \ll N$.
- Low-rank likelihood evaluation only $O(Nr^2 + r^3)$.

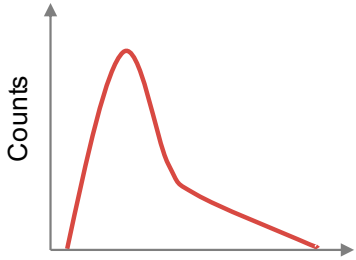
Low-rank Gaussian Process Parametrization complexity



- Projecting $\mathbf{z}_{it} \in \mathbb{R}$ to $\mathbf{h}_{it} \in \mathbb{R}^k$: $O(k)$ parameters
- Projecting $\mathbf{h}_{it}$ to likelihood parameters of $p(\mathbf{z}_t|\mathbf{h}_t)$: $O(kr)$ parameters where r is the rank ($r = 3$ in the figure)

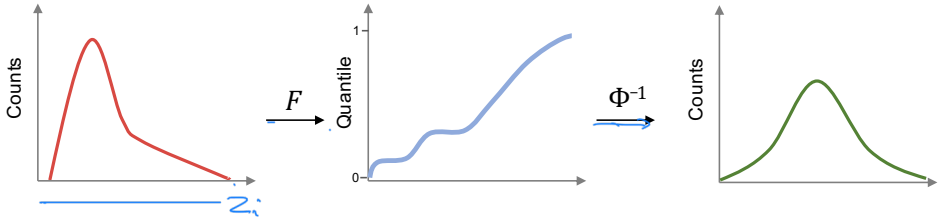
Scaling and non-Gaussian Data





Addressing Issue 3: Gaussian Copula

$$C(F_1(z_1), \ldots, F_d(z_N)) = \phi_{\boldsymbol{\mu},\Sigma}(\Phi^{-1}(F_1(z_1)), \ldots, \Phi^{-1}(F_N(z_N)))$$

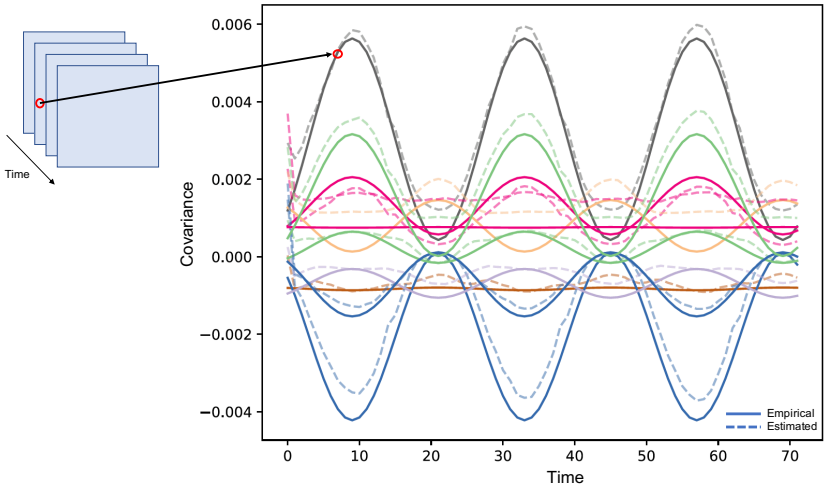


- F_i is the empirical CDF of z_{it}

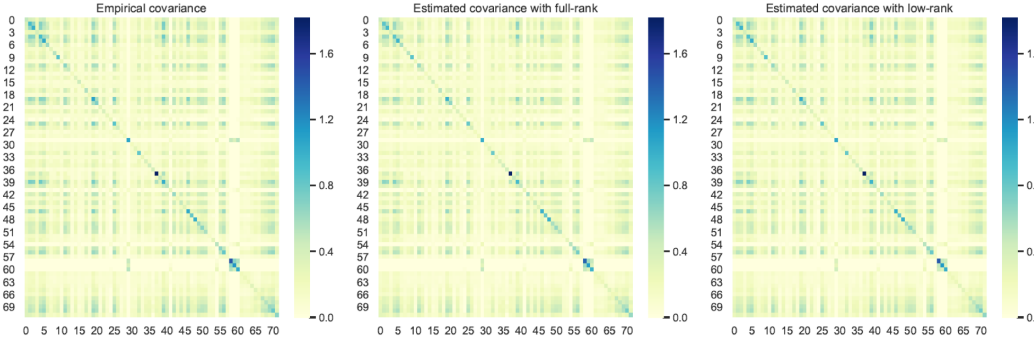
Section 3

Experimental Results

Model recovers Covariances of Artificial Dataset



Covariance Structure is Preserved by Low-rank Approximation



Results on Real-World Datasets

	CRPS-Sum					
dataset estimator	exchange	solar	elec	traffic	taxi	wiki
VAR	0.010+/-0.000	0.524+/-0.001	0.031+/-0.000	0.144+/-0.000	0.292+/-0.000	3.400+/-0.003
GO-GARCH	0.020+/-0.000	0.869+/-0.000	0.278+/-0.000	0.368+/-0.000	-	-
Vec-LSTM-ind	0.009+/-0.000	0.470+/-0.039	0.731+/-0.007	0.110+/-0.020	0.429+/-0.000	0.801+/-0.029
Vec-LSTM-ind-scaling	0.008+/-0.001	0.391+/-0.017	0.025+/-0.001	0.087+/-0.041	0.506+/-0.005	**0.133+/-0.002**
Vec-LSTM-fullrank	0.646+/-0.114	0.956+/-0.000	0.999+/-0.000	-	-	-
Vec-LSTM-fullrank-scaling	0.394+/-0.174	0.920+/-0.035	0.747+/-0.020	-	-	-
Vec-LSTM-lowrank-Copula	**0.007+/-0.000**	**0.319+/-0.011**	0.064+/-0.008	0.103+/-0.006	0.326+/-0.007	0.241+/-0.033
GP	0.011+/-0.001	0.828+/-0.010	0.947+/-0.016	2.198+/-0.774	0.425+/-0.199	0.933+/-0.003
GP-scaling	0.009+/-0.000	0.368+/-0.012	**0.022+/-0.000**	**0.079+/-0.000**	**0.183+/-0.395**	1.483+/-1.034
GP-Copula	**0.007+/-0.000**	**0.337+/-0.024**	**0.024+/-0.002**	**0.078+/-0.002**	**0.208+/-0.183**	**0.086+/-0.004**

CRPS-sum accuracy comparison (lower is better, best two methods are in bold).

Results on Real-World Datasets

baseline	architecture	data transformation	distribution	CRPS ratio	CRPS-Sum ratio	num params ratio
VAR	-	-	-	10.0	10.9	35.0
GO-GARCH	-	-	-	7.8	6.3	6.2
Vec-LSTM-ind	Vec-LSTM	None	Independent	3.6	6.8	13.9
Vec-LSTM-ind-scaling	Vec-LSTM	Mean-scaling	Independent	1.4	1.4	13.9
Vec-LSTM-fullrank	Vec-LSTM	None	Full-rank	29.1	44.4	103.4
Vec-LSTM-fullrank-scaling	Vec-LSTM	Mean-scaling	Full-rank	22.5	37.6	103.4
Vec-LSTM-lowrank-Copula	Vec-LSTM	Copula	Low-rank	1.1	1.7	20.3
GP	GP	None	Low-rank	4.5	9.5	1.0
GP-scaling	GP	Mean-scaling	Low-rank	2.0	3.4	1.0
GP-Copula	GP	Copula	Low-rank	1.0	1.0	1.0

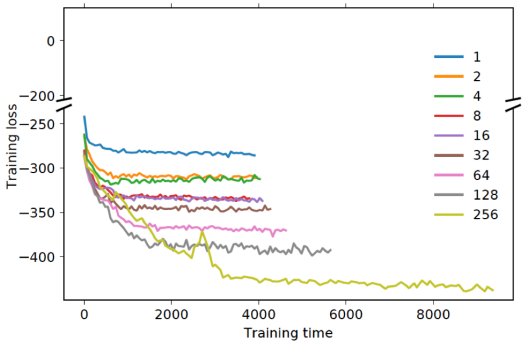
Baselines summary and average ratio compared to GP-Copula for CRPS, CRPS-Sum.

Conclusion

Low-Rank Gaussian Copula Processes

- Scalable, multivariate, global deep learning model
- Explicitly models the joint distribution up to 2000 dimensions
- Lower number of parameters and better accuracy than state of the art methods
- Opens the door to new applications that require modeling of correlations: risk minimizing portfolios, seller competition, anomaly detection

Effect of Low-rank Approximation



rank	test NLL	train NLL
1	-291.4+/-8.2	-288.9+/-8.2
2	-306.2+/-6.7	-304.8+/-5.7
4	-319.3+/-4.9	-312.1+/-3.5
8	-333.6+/-7.7	-330.2+/-6.3
16	-334.8+/-4.9	-337.5+/-4.
32	**-341.8+/-6.8**	-345.2+/-17.0
64	-338.5+/-10.9	-360.5+/-10.7
128	-326.6+/-20.1	-393.7+/-26.1
256	-238.0+/-38.4	**-423.1+/-20.7**