Improving probabilistic wind speed forecasts with deep learning

Katharina Klein[1,2], Daniel Tolomei[1,2], Sjoerd Dirksen[1], Kirien Whan[2], Maurice Schmeits[2]

[1]Utrecht University, The Netherlands [2]Royal Netherlands Meteorological Institute (KNMI), The Netherlands

Contact: k.klein@uu.nl

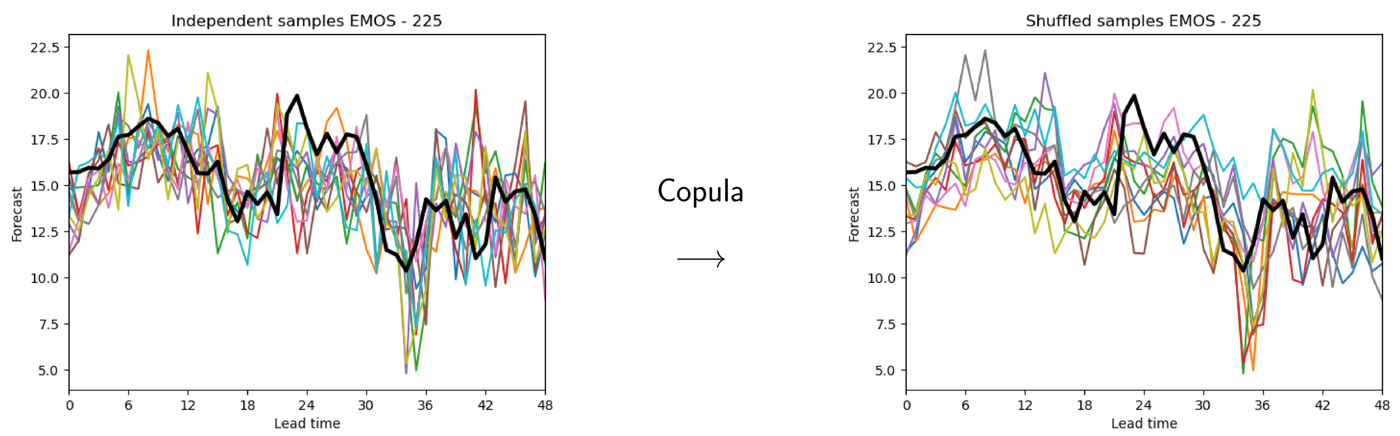
Motivation

Post-processing NWP model outputs is important:

- NWP forecasts often contain biases that need to be corrected.
- Quantifying the uncertainty in the forecasts allows us to make better inferences.

Classical approach:

- Process the forecasts for each variable, lead time and location independently.
- Reintroduce dependencies by applying an empirical copula method.



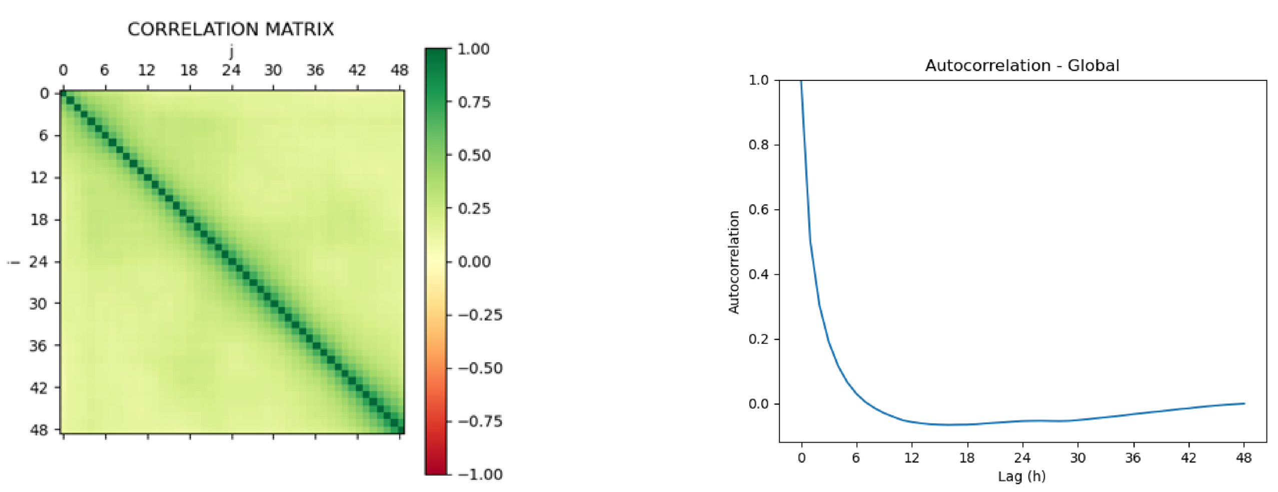
Skipping the two-step approach

We would instead like to model temporal dependencies explicitly.

> Can we derive an explicit joint distribution for wind speeds $(W_0, \ldots, W_T)$ from deterministic NWP forecasts?

Autocorrelation of Forecast Errors

Forecast errors at successive lead times are positively correlated.



The errors $W_t - w_t$ appear to form an AR(p) process

(1) $$(W_t - w_t) - \bar{\mu} = \phi_1(W_{t-1} - w_{t-1} - \bar{\mu}) + \ldots + \phi_p(W_{t-p} - w_{t-p} - \bar{\mu}) + \epsilon_t$$

where
- $\epsilon_t \sim N(0, \sigma^2)$ are independent error terms,
- w_t is the NWP forecast for wind speed at lead time t,
- W_t is the corresponding observed wind speed.

The parameter p of the process is expected to be small (correlations are strongest for lower lags). We derive the ARMOS(p) model by rewriting (1) and making the mean and variance parameters of the AR process time-dependent.

ARMOS(p) Model

> We model $W_0 \sim \mathcal{D}_0(\mu_0, \sigma_0^2)$ and
> $$W_t | W_{t-1}, \ldots, W_0 \sim \mathcal{D}_t\Big(\mu_t + \sum_{j=1}^{p} \phi_j (W_{t-j} - \mu_{t-j}), \sigma_t^2\Big)$$
> (assuming $W_{t-j} - \mu_{t-j} = 0$ if $t - j < 0$).

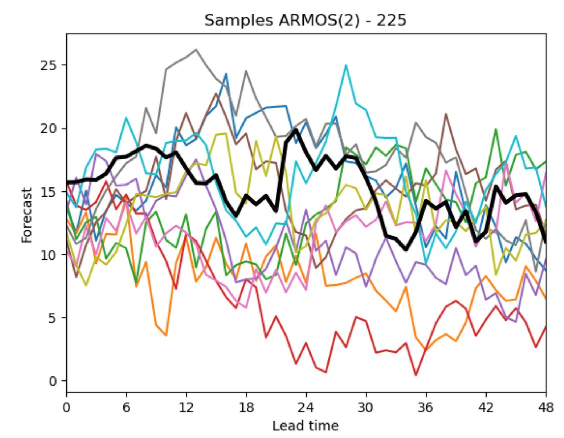
- Our model allows for distributions $\mathcal{D}_t$ other than the normal distribution, and for different choices in modeling the parameters.
- Here we choose a truncated normal distribution and the parameters are modeled as

$$\mu_t = a_{0,t} + a_{1,t} f_{1,t} + \ldots + a_{M,t} f_{M,t}$$
$$\sigma_t^2 = b_{0,t} + b_{1,t} S_t^2$$
$$\phi_1, \ldots, \phi_p \quad \text{free parameters}$$

where

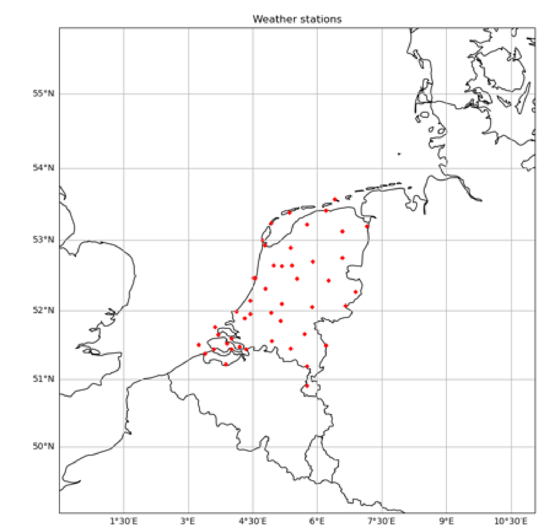
- $f_{1,t}, \ldots, f_{M,t}$ are NWP forecasts for different weather variables (wind speed at 10m, mean sea level pressure, ...)
- S_t^2 is the spatial variance of forecasts around the station

- We can subsequently sample from the fitted model without the need for a copula method.



Data and Methods

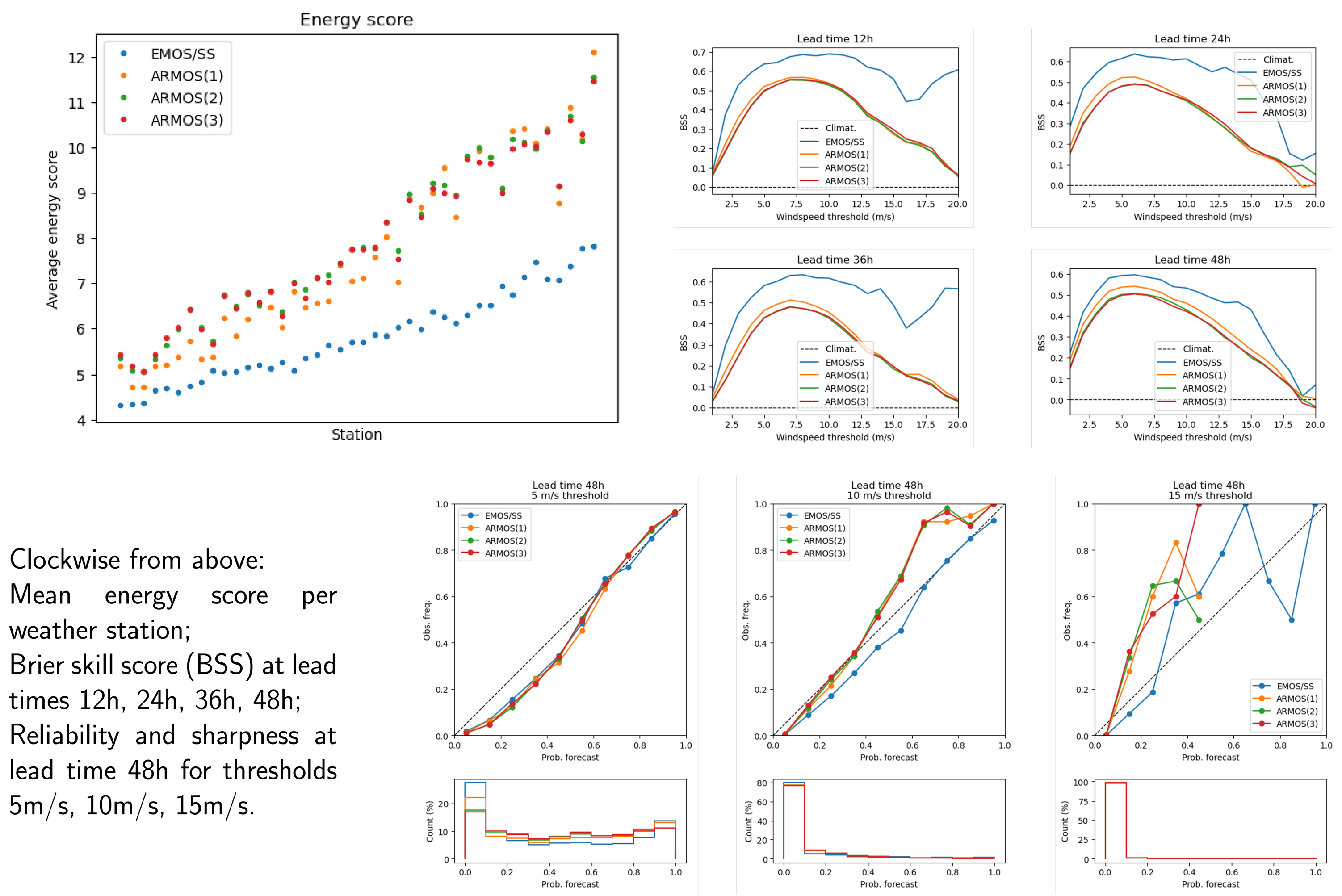
- NWP data from KNMI's Harmonie-Arome model (deterministic) for the extended winter period (October - April).
- 2.5km × 2.5km spatial resolution.
- Forecasts are initialized at 0000 UTC and range up to lead time +48h.
- Observations are available for 43 weather stations across the Netherlands.



We compare the ARMOS(p) model to a variant of EMOS combined with Schaake-Shuffle. Models are trained with the log score (LS) as loss function, and evaluated additionally on the energy score (ES) and variogram score (VS) to assess marginal calibration and correlation structure respectively, as well as with regard to several univariate verification methods.

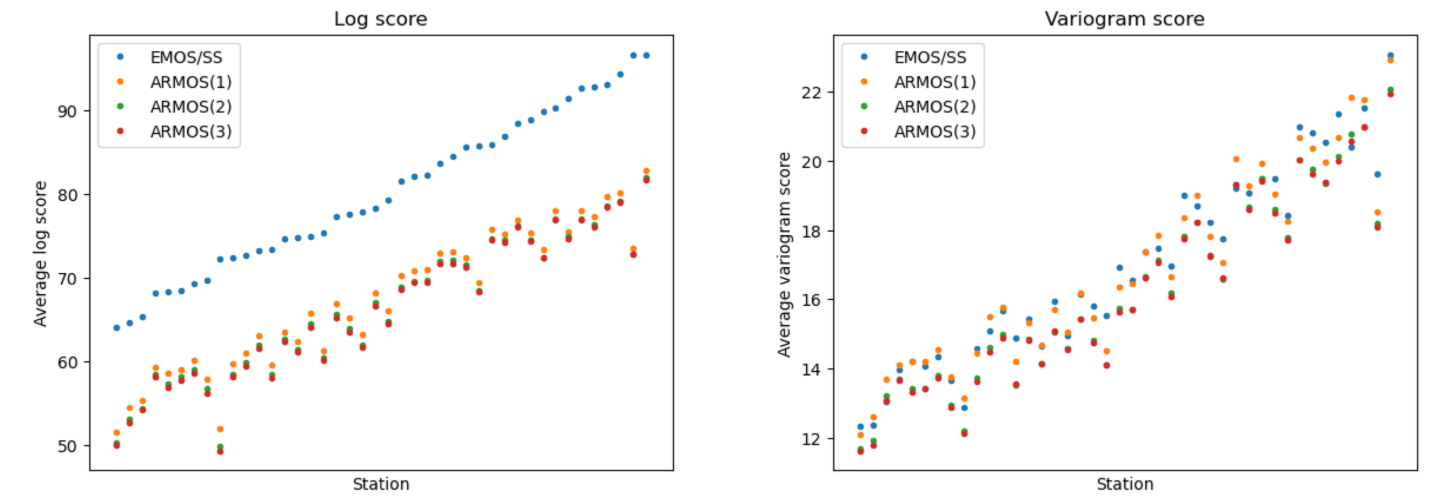
Preliminary Results

As expected for the marginal calibration, the EMOS approach outperforms the ARMOS(p) model.



Clockwise from above:
Mean energy score per weather station;
Brier skill score (BSS) at lead times 12h, 24h, 36h, 48h;
Reliability and sharpness at lead time 48h for thresholds 5m/s, 10m/s, 15m/s.

However, the ARMOS(p) model achieves improved results for the log score and the variogram score, indicating that the correlation structure is captured well.



Outlook and Ongoing Work

- Improve the model by including more or different NWP variables to derive model parameters.
- Adjust the loss function to improve marginal calibration.
- Model the parameters as the output of a neural network. Spatial and temporal dependencies from the (grid) input can be extracted through convolutional, LSTM and dense layers.

References:
(1) *Multivariate Postprocessing of Temporal Dependencies with Autoregressive and LSTM Neural Networks.* Tolomei, D. MA thesis, Utrecht University (2022).
(2) In preparation: *Multivariate Post-processing of Temporal Dependencies with Autoregressive and LSTM Neural Networks.* Klein, K., Tolomei, D., Dirksen, S., Schmeits, M., Whan, K.