Spatially Coherent Postprocessing of Cloud Cover Ensemble Forecasts

Y. DAI^a AND S. HEMRI^b

^a *Seminar for Statistics, ETH Zurich, Zurich, Switzerland*

^b *Federal Office of Meteorology and Climatology MeteoSwiss, Zurich, Switzerland*

(Manuscript received 8 March 2021, in final form 20 September 2021)

ABSTRACT: Statistical postprocessing is commonly applied to reduce location and dispersion errors of probabilistic forecasts provided by numerical weather prediction (NWP) models. If postprocessed forecast scenarios are required, the combination of ensemble model output statistics (EMOS) for univariate postprocessing with ensemble copula coupling (ECC) or the Schaake shuffle (ScS) to retain the dependence structure of the raw ensemble is a state-of-the-art approach. However, modern machine learning methods may lead to both a better univariate skill and more realistic forecast scenarios. In this study, we postprocess multimodel ensemble forecasts of cloud cover over Switzerland provided by COSMO-E and ECMWF-IFS using (i) EMOS + ECC, (ii) EMOS + ScS, (iii) dense neural networks (dense NN) + ECC, (iv) dense NN + ScS, and (v) conditional generative adversarial networks (cGAN). The different methods are verified using EUMETSAT satellite data. Dense NN shows the best univariate skill, but cGAN performed only slightly worse. Furthermore, cGAN generates realistic forecast scenario maps, while not relying on a dependence template like ECC or ScS, which is particularly favorable in the case of complex topography.

SIGNIFICANCE STATEMENT: Often, statistical postprocessing is applied to the output from weather models to reduce systematic errors. However, traditional postprocessing approaches destroy the spatial patterns of the predictions of the weather model. For instance, cloud cover amounts at two neighboring points on a plain are more likely to be correlated with each other than with the amount on a nearby mountain. In traditional postprocessing, the spatial patterns of the weather model are often imposed on the postprocessed forecasts using the output from the weather model as a template. However, this approach is suboptimal for cloud cover. Therefore, we have applied a modern machine learning method, designed to produce realistic-looking images, to obtain postprocessed and physically realistic maps of cloud cover forecast scenarios.

KEYWORDS: Statistical techniques; Probability forecasts/models/distribution; Clouds; Ensembles; Postprocessing; Machine learning

1. Introduction

In recent years, neural networks have become increasingly powerful, and their rise has also influenced weather forecasting, especially the area of postprocessing (Shi et al. 2017; Rasp and Lerch 2018; Bremnes 2019). We will focus on postprocessing of cloud cover forecasts. The amount of cloud cover has a large influence on the intensity of solar radiation (Matuszko 2012), and has implications for both human health (Juzeniene et al. 2011) and energy supply (Palz 2013). Being able to accurately forecast cloud cover could also have benefits for observational astronomy (Ye and Chen 2013), and the presence or absence of clouds often highly affects how the weather is perceived.

Postprocessing of cloud cover forecasts using classical statistical methods, namely, multinomial and proportional odds

logistic regression, was performed by Hemri et al. (2016). More recently, Baran et al. (2021) applied several (mostly classical) machine learning techniques to the cloud cover postprocessing problem; multilayer perceptron neural networks, gradient boosting machines, and random forests. Both these approaches treat cloud cover as a discrete variable. Very recently, Dupuy et al. (2021) used a convolutional neural network for deterministic cloud cover postprocessing.

In this study, we attempt to probabilistically postprocess cloud cover as a continuous variable. The univariate approaches that are typically used for this, e.g., ensemble model output statistics (EMOS; Gneiting et al. 2005), tend to destroy the spatial dependence structures (Scheffzik and Möller 2018), which need to be reintroduced retrospectively; either from the raw ensemble as in ensemble copula coupling (ECC; Scheffzik et al. 2013), or from historical observations using the Schaake shuffle (ScS; Clark et al. 2004). However, both approaches are not optimal. The former relies on the assumption that the spatial dependence structure of the raw ensemble is consistent with the observed dependence structure. In particular over complex terrain, this assumption likely does not hold (Westerhuis et al. 2020). The latter implicitly assumes that the weather regimes of a forecast case and of the set of historical fields used to determine the spatial dependence structure are

Denotes content that is immediately available upon publication as open access.

Hemri's current affiliation: Department of Mathematics, University of Zurich, Zurich, Switzerland.

Corresponding author: Stephan Hemri, stephan.hemri@math.uzh.ch

DOI: 10.1175/MWR-D-21-0046.1

© 2021 American Meteorological Society. For information regarding reuse of this content and general copyright information, consult the [AMS Copyright Policy](#) ([www.ametsoc.org/PUBSReuseLicenses](#)).

similar. For a more elaborated ScS approach that addresses this issue, please refer to [Scheffzik \(2016\)](#).

We propose a multivariate method that is able to generate spatially realistic forecasts of cloud cover, without the need of any dependence template. This enables us to not depend on the (possibly faulty) raw ensembles, while not being restricted to the set of historical observations either. To this end, we use a conditional generative adversarial network (cGAN; [Mirza and Osindero 2014](#)), shown to be successful at generating realistic-looking images ([Karras et al. 2017](#); [Jones 2017](#); [Zhu et al. 2017](#); [Pathak et al. 2016](#)). This is achieved by training two neural networks simultaneously in a two-player game, where one tries to generate (fake) images that look real, and the other tries to identify whether an image is real or fake. Our goal will be to generate artificial images (forecasts) that are indistinguishable from the eventual cloud cover observations.

We will compare the realistic appearance as well as the accuracy of these forecasts with two univariate baselines that were optimized for performance in terms of the continuous ranked probability score (CRPS; [Matheson and Winkler 1976](#)): global EMOS (gEMOS) as a classical statistical method and a deep learning method (dense NN). ECC and ScS have been applied to both univariate baselines to obtain realistic forecast maps. Preliminary tests indicated that gEMOS and the proportional odd logistic regression approach by [Hemri et al. \(2016\)](#) perform equally well for the data at hand.

The remainder of this paper will be organized as follows. In [section 2](#), we will explain the dataset we used, [section 3](#) describes the methods mentioned above; and in [section 4](#), we will present our results, followed by a brief discussion in [section 5](#) and the conclusions in [section 6](#).

2. Data

We use a dataset with numerical weather forecasts from two different models, namely, COSMO-E and ECMWF-IFS (hereinafter “IFS”). COSMO-E ([Klasa et al. 2018](#)) provides numerical forecasts for more than 20 weather-related parameters, for each point on a $2\text{ km} \times 2\text{ km}$ grid over the Alpine region, with hourly resolution. The model is initialized twice per day (at midnight and noon, UTC), and forecasts are given for 0, 1, $\dots$, 120 h into the future. Each forecast consists of a control run and 20 additional ensemble members, obtained by perturbing the initial conditions within observation uncertainties.

The IFS ([Owens and Hewson 2018](#)) forecasts are also initialized at midnight and noon daily, and are given on an $18\text{ km} \times 18\text{ km}$ grid for 15 days into the future. The ensemble size is 51, consisting of a control run and 50 perturbed runs. Since the IFS forecasts become available later than those from COSMO-E, we use the IFS forecasts initialized 12 h earlier and adapt the lead times accordingly.

The cloud cover observations we use have been generated based on EUMETSAT satellite data using the CM SAF CFC algorithm ([Stöckli et al. 2019](#)). The resulting observation grid covers Switzerland at a 2-km resolution. Both the observations and the IFS forecasts are mapped onto the COSMO-E grid using nearest-neighbor matching. Hence, our study domain is confined to a $2\text{ km} \times 2\text{ km}$ grid over Switzerland. Because of

computational constraints, we have applied a subsequent 2×2 subsampling for dense NN and cGAN. Accordingly, all results shown in the following refer to the subsampled grid, which contains 59×90 grid points.

We use almost four years of forecasts from COSMO-E and IFS, initialized between May 2016 and February 2020, together with the corresponding observations. Two years (May 2016–April 2018) are used for training, one year (May 2018–April 2019) for validation, and 10 months (May 2019–February 2020) for testing. All models were trained using the training set, and the best models, i.e., the best model configuration and the best set of predictors, of each of our methods were selected using the validation set, after which only these were retrained using an extended training set (May 2016–April 2019). The results shown in [section 4](#) will be based on these models’ performance on the test set.

3. Methods

a. Univariate baselines

1) GEMOS

As mentioned in [section 1](#), gEMOS serves as a reference method. To tailor the nonhomogeneous Gaussian regression EMOS approach ([Gneiting et al. 2005](#)) to cloud cover, the bounded support $[0, 100]$ with point masses at 0% and 100% needs to be taken into account. To this end, we follow a suggestion by M. Scheuerer (2014, personal communication). Censoring the EMOS forecast distribution shifts probability masses below 0% (above 100%) to 0% (100%) cloud cover, leading to a censored normal distribution $\mathcal{N}_0^{100}(\mu, \sigma)$. Omitting any indices for spatial location, reference and lead times, μ is linked to the raw ensembles by

$$\mu = a_0 + a_1 \bar{f}_{\text{CLCT,CO}} + a_2 \bar{f}_{\text{CLCT,IFS}} + a_3 \bar{f}_{\text{T2M,CO}}, \quad (1)$$

and σ^2 by

$$\sigma^2 = b_0^2 + b_1^2 s_{\text{CLCT,CO}}^2 + b_2^2 s_{\text{CLCT,IFS}}^2, \quad (2)$$

where $\bar{f}_{\text{CLCT,CO}}$ and $\bar{f}_{\text{CLCT,IFS}}$ denote total cloud cover (CLCT) ensemble mean of COSMO-E and IFS, respectively. While the ensemble mean of COSMO-E near-surface temperature denoted by $\bar{f}_{\text{T2M,CO}}$ is used as additional predictor for the location parameter μ , the scale parameter σ^2 depends only on the CLCT ensemble variances of COSMO-E and ECMWF-IFS, denoted by $s_{\text{CLCT,CO}}^2$ and $s_{\text{CLCT,IFS}}^2$, respectively. The coefficients $a_0, a_1, a_2, a_3, b_0, b_1, b_2$ are estimated by CRPS minimization.

We fit gEMOS models separately for each season (DJF, MAM, JJA, SON), each initialization hour, and each lead time. However, spatially we use a global estimation approach, i.e., one set of gEMOS coefficients is estimated for all grid points. Despite its simplicity, the gEMOS approach outperformed a local EMOS alternative, i.e., fitting a separate model for each grid point, in terms of CRPS.

2) DENSE NN

The other reference method, the dense neural network (dense NN), was inspired by [Rasp and Lerch \(2018\)](#), and uses the following set of dynamic predictors: CLCT_mean, CLCT_var,

TABLE 1. List of NWP predictors from COSMO-E and IFS that we used, together with their definitions.

Definitions of used NWP predictors		
COSMO-E	CLCT	Total cloud area percentage
	CLCH	Cloud area percentage in high troposphere
	CLCM	Cloud area percentage in medium troposphere
	CLCL	Cloud area percentage in low troposphere
	HPBL	Height of planetary boundary layer
	T_2M	Air temperature at 2 m above ground
IFS-ENS	TCC	Total cloud area fraction
	HCC	Cloud area fraction in high troposphere
	MCC	Cloud area fraction in medium troposphere
	LCC	Cloud area fraction in low troposphere

CLCH_mean, CLCL_mean, TCC_mean, TCC_var, HCC_mean, HCC_var, MCC_mean, MCC_var, HPBL_mean, T_2M_mean, and T_2M_var. The meaning of the abbreviations can be found in Table 1; for example, CLCT_var means the ensemble variance of the total cloud area fraction predicted by COSMO-E. The static predictors—initialization hour, lead time, location, hour of day and month—are introduced to the network via embeddings (Mikolov et al. 2013), which proved successful in many different areas in deep learning. Originally, embeddings were designed for natural language processing, where every word in a finite vocabulary is mapped to a vector in a high-dimensional space, such that semantically similar words have similar vector representations (Pennington et al. 2014). This idea can be extended to any variable with a finite number of possible values, such as our static predictors. Doing so, more information can be captured than with a model relying solely on the raw predictors.

The embeddings of the static predictors are concatenated with the raw dynamic predictors, and fed into the eight subsequent dense (i.e., fully connected) layers of the network. The last layer outputs the 21 values that make up the ensemble forecast, which are clipped to the interval $[0, 100]$. Therefore, no assumptions are made on the distribution of the ensemble members, making this approach nonparametric. A single global dense NN is fitted for all seasons, initialization hours, lead times, and grid points. The ensemble CRPS is used as a loss function during training. An illustration of the model is given in Fig. 1. The hyperparameters were chosen using systematic trial-and-error, which means its performance could potentially be further improved by using, for example, randomized search or grid search. However, in our experience, the exact value of the hyperparameters used has a smaller influence on the performance than the choice of dynamic predictors.

b. Multivariate methods

1) ECC EXTENSION TO THE UNIVARIATE BASELINES

Being univariate postprocessing methods, both gEMOS and dense NN destroy the spatial dependence structure given by the raw ensemble members. Applying ECC (Scheffzik et al. 2013) allows reintroducing such a dependence structure by

reordering postprocessed forecast quantiles according to the rank order structure of the raw ensemble, here COSMO-E.

Since in our study domain, both clear and totally overcast sky are observed frequently, ensemble forecasts quite often exhibit ties at 0 and 100. Resolving these ties at random leads to unrealistically “noisy” maps of forecast scenarios. Scheuerer and Hamill (2018) introduced a modification to ECC for precipitation, which allows us to obtain physically realistic fields of postprocessed precipitation even in the presence of ties at zero. For this study, we have adapted this approach for cloud cover. More specifically, at each grid point, raw ensemble members forecasting 0 or 100% cloud cover are replaced by random draws from $\mathcal{U}(-100, 0)$ or $\mathcal{U}(100, 200)$, respectively. Then, in our standard implementation (denoted by ECC1e5 henceforth), the modified raw ensemble member fields are smoothed using a tricubic kernel with a support radius of 100 km. To verify the sensitivity with regard to the support radius, we have also applied ECC variants with smaller support radii of 10 and 50 km, denoted by ECC1e4 and ECC5e4, respectively. The smoothed modified raw ensembles are subsequently used as spatial dependence structure template for ECC reordering. Refer to Scheuerer and Hamill (2018) for details on the kernel smoothing approach.

2) ScS EXTENSION

As an alternative to ECC, we also apply ScS (Clark et al. 2004) to obtain a spatial dependence structure by reordering postprocessed forecast quantiles. ScS works in a similar way as ECC, but the rank order template is taken from past observations. Here, we use CM SAF observations for the hour of day of the forecast validation time from a symmetric period of 21 days around the same date of the year preceding the current year of interest.

3) CONDITIONAL GAN

We propose a multivariate postprocessing method that gives spatially realistic forecasts, without relying on ECC or ScS, with the use of generative adversarial networks. A generative adversarial network (GAN; Goodfellow et al. 2014) consists of two neural networks: a generator (G) and a discriminator (D). The discriminator network D is a classifier trained to determine whether its input comes from the (training) dataset, and the generator network G is trained to produce samples that D classifies as “from the actual dataset.” These networks are simultaneously trained and have conflicting interests. Ideally, G will ultimately be able to create artificial samples that are indistinguishable from those of the training dataset. The generator network G has (usually standard normal) random noise as input, which it uses to get variability in the output: every noise realization is mapped to an artificial sample from the training set, i.e., a realization from the distribution it learned. GANs are used for a wide variety of applications, from generating images of human faces (Karras et al. 2017) or creating artworks (Jones 2017), to numerous geoscientific applications (Leinonen et al. 2020; Gagne et al. 2019; Chen et al. 2019; Scher and Peßenteiner 2021). We will use GANs to generate artificial cloud cover observations, and use these as forecasts. Indeed, a forecast that is indistinguishable from the eventual observation, is by definition a good forecast.

FIG. 1. Illustration of the dense NN model. The batch size—indicated by None, as usual—is 8192, and the learning rate is 0.0001. The model was trained for 5500 batches, and the training time was about 9 h on two Intel Xeon Gold 6254 18-core 3.1-GHz CPUs.

Obviously, it is not sufficient for our forecasts to only look like an actual observation; they need to give meaningful information about future cloud cover scenarios. Thus, we will use conditional GANs (cGANs; [Mirza and Osindero 2014](#)), in which both G and D get an additional condition. cGANs can be applied to numerous nongenerative tasks, including domain translation ([Zhu et al. 2017](#)) and image infilling ([Pathak et al. 2016](#)). We will condition G and D on the forecasts from COSMO-E and ECMWF-IFS, meaning that G is trained to generate realistic observations given these numerical forecasts, and D tries to determine whether

an image is a real observation, given the same forecasts. In other words, G is attempting to learn the conditional distribution of cloud cover observations given the numerical weather forecasts, and can thus be interpreted as a post-processing tool. The following dynamic predictors are used: CLCT_mean, CLCT_var, CLCH_mean, CLCL_mean, TCC_mean, TCC_var, HCC_mean, MCC_mean, LCC_mean, LCC_var, HPBL_mean, and T_2M_mean. As in the dense NN case, we also give static predictors as additional input to both G and D . [Figure 2](#) illustrates the cGAN architecture for cloud cover postprocessing.

FIG. 2. The cGAN architecture for cloud cover postprocessing.

Both G and D are convolutional neural networks (Lecun et al. 1998), which are commonly used in such image processing tasks. The generator network G follows an encoder–decoder architecture (Ronneberger et al. 2015), where the dynamic predictors are encoded using convolution layers, and concatenated with the static predictor embeddings and random noise to make the output probabilistic. Subsequently, this is decoded into an artificial observation using transposed convolution layers. We also use leaky rectified linear unit (ReLU) activation functions (Maas et al. 2013), and batch normalization (Ioffe and Szegedy 2015) for stabilizing and accelerating the training. The last transposed convolution layer has a hyperbolic tangent (tanh) activation function to make sure the output is bounded. More details on the generator network G are given in Fig. 3. The attentive reader might notice that the images of the dynamic predictors and artificial observations are 60×92 , instead of the grid size 59×90 that we mentioned before. This is because both the number of rows and the number of columns in our images need to be divisible by four, due to our architecture with the (transposed) convolution layers.¹

The discriminator network D concatenates the (real or artificial) cloud cover observation with the dynamic predictors and static predictor embeddings, after which these images are encoded using two-dimensional convolution layers. We again use a leaky ReLU activation function, and dropout layers (Hinton et al. 2012) to prevent overfitting. Finally, there is a dense layer that outputs a single number, reflecting whether D thinks the cloud cover observation is real or fake. More details on the discriminator network D are given in Fig. 4.

¹ When applying the convolution layers for downsampling, both the number of rows and the number of columns are halved twice. Having an odd number of rows or columns is not a problem here, since the downsampled sizes are simply rounded. However, when upsampling using the transposed convolution layers, the number of rows and columns are doubled twice, which means the number of rows and columns in the artificial observation will always be divisible by 4. In order not to make it too easy for the discriminator, the sizes of the real observations need to be adapted accordingly.

As explained above, the output of G , which is a two-dimensional image, will be used as a postprocessed forecast for a fixed initialization time and lead time. Thanks to G 's output being probabilistic, we can obtain as many realizations as we desire, and use these as an ensemble forecast. The network D is only used as a training partner for G , but could also be leveraged to select the most realistic ones among the forecasts output by G . This would lead to more realistic-looking forecasts that tend to perform worse in terms of CRPS. On the other hand, forecasts with smaller CRPS tend to be more easily distinguishable from actual observations by the discriminator.²

Finally, note that during training, the forecasts given by G are not evaluated using CRPS or any other metric that compares them with the observation. Instead, during training time, G is optimized for fooling the discriminator, being rewarded—via binary cross-entropy loss—if D thinks the generated image is a real observation. In fact, G does not get to see any of the observations, and can only learn how to produce artificial observations via the feedback of D (who gets to see observations occasionally). After training, however, we can evaluate the forecasts given by G in the same way as we do for the other models. This will be done in the next section.

c. Verification measures

Besides CRPS, we use a number of other measures to evaluate the performance of the different methods. When performing probabilistic postprocessing, the objective is to “maximize sharpness subject to calibration” (Gneiting et al. 2005). Sharpness is a property of the forecast ensemble, measuring how concentrated the forecast distribution is, which can be assessed simply by computing the standard deviation of the ensemble members. Thus, we will use the average standard deviation of the ensemble members as a measure for evaluating sharpness. We assess calibration using rank histograms

² We have seen this in our results, but do not have a clear and theoretically backed explanation. Our hypothesis is that as the forecasts get more realistic, the spread decreases, leading to an increase in CRPS. Conversely, optimizing for CRPS leads to “blurry” forecasts that are easily distinguishable from observations.

FIG. 3. Illustration of the generator model G . The batch size—indicated by None, as usual—is 256, and the learning rate is 0.0001. The model was trained for 340 epochs, and the training time was about 13 h using a GeForce RTX 2080 Ti GPU.

FIG. 4. Illustration of the discriminator model D . The batch size—indicated by None, as usual—is 256, and the learning rate is 0.0001. The model was trained for 340 epochs, and the training time was about 13 h using a GeForce RTX 2080 Ti GPU.

(e.g., [Anderson 1996](#)). To quantitatively assess deviations from a uniform rank histogram, we use a goodness-of-fit test statistic for uniformity. More specifically, we compute the test statistic T of a χ^2 as described, for example, in [Elmore \(2005\)](#):

$$T = \sum_{i=1}^k \frac{(o_i - f_i)^2}{f_i}, \quad (3)$$

where k denotes the number of bins, while o_i and f_i denote the observed and expected frequencies of observations to fall in bin i . Because of the large magnitude of this metric for large test sets, we will use $\log T$ as our calibration measure.

Finally, we introduce multivariate measures, used to evaluate the spatial correlation structure of the forecasts, namely, the energy score ([Gneiting and Raftery 2007](#)), the p -variogram score ([Scheuerer and Hamill 2015](#)), and three univariate quantities that are functions of the forecasts over the entire grid. The ensemble version of the energy score is given by

$$es = \frac{1}{M} \sum_{m=1}^M \|\mathbf{x}_m - \mathbf{y}\| - \frac{1}{2M^2} \sum_{n=1}^M \sum_{m=1}^M \|\mathbf{x}_n - \mathbf{x}_m\|, \quad (4)$$

where $\mathbf{x}_m = (x_m^1, \dots, x_m^L)$ denotes the vector of all forecasts of ensemble member m of length L equal to the number of grid points considered ([Scheffzik 2016](#)). The corresponding vector of observations is denoted by $\mathbf{y}$ and $\|\cdot\|$ is the Euclidean norm. The p -variogram score is computed as

$$vs = \sum_{k=1}^L \sum_{l=1}^L w_{kl} \left(|y_k - y_l|^p - \frac{1}{M} \sum_{m=1}^M |x_m^k - x_m^l|^p \right)^2, \quad (5)$$

where $w_{kl} \geq 0$ are nonnegative weights. In our case, they are equal to the inverse of the distance between the grid points corresponding to indices k and l if this distance is positive and smaller than 10 km; otherwise $w_{kl} = 0$. Following [Scheuerer and Hamill \(2015\)](#), we set p to 0.5 and refer to the score as the 0.5-variogram score henceforth. The univariate quantities comprise the CRPS of the spatial fraction of zero percent cloud cover (CRPS-f0), the CRPS of the spatial fraction of 100% cloud cover (CRPS-f100), and the average of the median pattern correlation (medPC). CRPS-f0 (CRPS-f100) refers simply to the ensemble CRPS for a forecast of the fraction of grid points with 0% (100%) cloud cover. We obtain pattern correlation ([Santer et al. 1993](#)) by computing Spearman's rank correlation coefficient ρ with respect to the CM SAF observations for each ensemble member. We then use the median of the ρ values of all ensemble members, called medPC henceforth, as a verification measure.

The multivariate measures are meant to quantify not only the forecast skill, but also how spatially realistic the forecasts are. Note that these metrics are computed separately for each initialization time and lead time, which means the temporal realistic appearance of the forecasts is not taken into account.

4. Results

The forecasts from our numerical models are either initialized at midnight or noon (UTC). Since preliminary tests showed no systematic difference between these two initialization hours, the results shown in this section will be based only

on numerical forecasts initialized at noon, and their corresponding postprocessed forecasts. [Figure 5](#) shows example forecasts for a subset of the models we compare, including IFS and COSMO-E, together with the corresponding observation. For each model, we show the worst and the best ensemble member in terms of energy score. For cGAN a best and a worst ensemble member are selected from an arbitrary sample of 21 cGAN forecast realizations.

Apparently, COSMO-E seems to fail to predict the observed band of cloud cover stretching from southwest to northeast. Both gEMOS and dense NN generate some clouds, but ECC does not provide any member with a spatial distribution of clouds similar to the observed field. In this example, applying ScS to gEMOS and dense NN leads to more realistic “best” scenarios. Likewise, cGAN seems to do a good job at producing a realistic “best” scenario. The last panel shows the topography of Switzerland, which can roughly be divided into three main geographic regions: the Jura in the northwest, the low-altitude Swiss Plateau, and the Alps with complex terrain.

We also computed the CRPS of all forecasts initialized in winter (DJF) and summer (JJA), grouped them by location (grid point) and averaged them; leading to a mean CRPS value for every model and location. Then for COSMO-E, dense NN, and cGAN, we computed the relative improvement of their mean CRPS with respect to that of gEMOS for every location separately, i.e., the continuous ranked probability skill score (CRPSS; [Winkler 1994](#)). These CRPSS values are shown on a map in [Fig. 6](#), together with the CRPS values of gEMOS.

Furthermore, we used Diebold–Mariano (DM) tests ([Diebold and Mariano 1995](#)) in order to assess the statistical significance of the CRPS-differences with respect to gEMOS. As the forecast horizon of each model run is 120 h and we consider only the noon initializations, the relevant forecast horizon for the (DM) tests is 5. For each season separately, we have computed DM test p values for each grid point and model comparison. The false discovery rate (FDR) is controlled and set to an overall nominal level of $\alpha_0 = 0.05$ by applying the Benjamini–Hochberg procedure as described in [Wilks \(2006\)](#). To account for multiple testing among the different model comparisons (COSMO-E versus gEMOS, dense NN versus gEMOS, and cGAN versus gEMOS), all p values of a specific season have been pooled together prior to applying the FDR correction. The grid points at which the difference is significant after these corrections are shown by a black dot in [Fig. 6](#).

We can see that gEMOS performs significantly better than COSMO-E, but seems to have some trouble in the Swiss plateau in winter. Moreover, the tests suggest that dense NN outperforms gEMOS in terms of CRPS. In particular for DJF, considerably more grid points show a significant improvement in CRPS compared to the expected 5% under the null hypothesis of equal CRPS after FDR correction. Also cGAN leads to a higher number of significant grid points than expected under the null hypothesis. However, for DJF, dense NN shows a more pronounced outperformance of gEMOS than cGAN.

In [Table 2](#), we show the scores introduced in [section 3c](#) for the different methods, averaged over the test set. Calibration measured as the log of the t -test statistic for uniformity of the

FIG. 5. Best and worst forecast member in terms of energy score compared to the CM SAF cloud cover observations of IFS, COSMO-E, gEMOS ECC1e5, gEMOS Schaake1e5, dense NN ECC1e5, dense NN Schaake1e5, and cGAN, respectively, initialized at 1200 UTC 28 Dec 2019, with lead time 22. The last two panels show the CM SAF cloud cover observation and the elevation of the points on our grid.

rank histograms, is best for cGAN followed by gEMOS. COSMO-E and dense NN seem to be considerably worse calibrated. In contrast, dense NN leads to the sharpest post-processed forecasts in terms of mean ensemble standard deviation, but also cGAN tends to be sharper than gEMOS.

For the multivariate measures (0.5-variogram, energy score, CRPS-f0, CRPS-f100, and medPC), we used Diebold–Mariano tests in the same way as described before, in order to conduct a significance test for the score differences with respect to gEMOS. For these eight tests, we used Walker's criterion to control for multiple testing, meaning that individual tests with p value smaller than 0.0073 are regarded as significant. The p values given in Table 2 indicate that the 0.5-variogram score of cGAN and the ScS variants of gEMOS and dense NN are significantly better than that of gEMOS ECC1e5, and that the difference between COSMO-E and gEMOS ECC1e5 as well as the difference between dense NN ECC1e5 and gEMOS ECC1e5 are not significant. Decreasing the ECC kernel size for gEMOS reordering, does not lead to an improvement in 0.5-

variogram score. In terms of energy score, COSMO-E performs significantly worse than gEMOS, whereas dense NN and cGAN both perform significantly better. Replacing ECC by ScS does not lead to any relevant change in energy score. CRPS-f0 and CRPS-f100 suggest that dense NN performs significantly worse than gEMOS ECC1e5 in terms of predicting the fraction of grid points with zero (one hundred) percent cloud cover. This may be related to rather poor calibration of dense NN, which leads to an underprediction of the fraction of grid points with zero (one hundred) percent cloud cover. However, in terms of CRPS-f100 cGAN exhibits a significantly lower score than gEMOS ECC1e5. Finally, medPC is rather low for gEMOS with ECC. Note that medPC is a positively oriented score, i.e., the larger the value the better. COSMO-E, gEMOS with ScS, dense NN with either ECC or ScS, and cGAN outperform gEMOS ECC1e5 in terms of medPC.

Let us now have a more detailed look at the temporal evolution of the forecast performance in terms of CRPS. Figure 7 shows the spatially averaged CRPS values of the different

FIG. 6. Mean CRPS of (first row) gEMOS, together with mean CRPSS of (second row) COSMO-E, (third row) dense NN, and (fourth row) cGAN with respect to gEMOS, based on all forecasts initialized at noon in (left) winter or (right) summer. Stippling shows the grid points at which the difference is statistically significant (pointwise DM tests, Benjamini–Hochberg adjusted for an FDR of 0.05). Ratios of significant values are provided in parentheses for (all grid points/grid points with positive CRPSS/grid points with negative CRPSS).

postprocessed forecasts against lead time and hour of day, stratified by season. To minimize the effect of the daily cycle of the performances of COSMO-E and IFS, for each lead time and hour of day, we use the raw ensemble with lower CRPS as reference. As an additional reference, the CRPSS of COSMO-E relative to a simple climatological forecast is shown as well.

The climatological forecast refers to 21 equidistant quantiles from the empirical distribution of all observed values for the same valid time of the 3 preceding years in a moving window of ± 2 weeks. It appears that COSMO-E outperforms the climatological forecast at all lead times. The gap in skill between gEMOS and dense NN or cGAN tends to decrease with

TABLE 2. Forecast skill, sharpness, calibration, and multivariate evaluation for COSMO-E, gEMOS, dense NN, and cGAN. Walker corrected significance level is 0.0073, and the best values are indicated in bold.

Method	CRPS	CRPSS	Sharpness	Calibration	0.5-variogram	Energy	CRPS-f0	CRPS-f100	medIPC
COSMO-E	15.28	-0.140	24.36	6.619	5.425 ($p \approx 0.9539$)	21.24 ($p < 10^{-4}$)	0.085 ($p \approx 0.2807$)	0.106 ($p \approx 0.6589$)	0.215 ($p \approx 0.0001$)
gEMOS ECC1e5	13.40	0.000	27.36	5.170	5.457 (reference)	19.52 (reference)	0.079 (reference)	0.103 (reference)	0.191 (reference)
gEMOS ECC1e4	13.40	0.000	27.36	5.170	5.887 ($p < 10^{-4}$)	19.56 ($p \approx 0.0005$)	0.081 ($p \approx 0.0018$)	0.108 ($p \approx 0.0003$)	0.186 ($p < 10^{-4}$)
gEMOS ECC5e4	13.40	0.000	27.36	5.170	5.479 ($p \approx 0.0057$)	19.55 ($p \approx 0.0050$)	0.080 ($p \approx 0.0080$)	0.106 ($p \approx 0.0003$)	0.188 ($p < 10^{-4}$)
gEMOS Schaake1e5	13.40	0.000	27.36	5.170	5.105 ($p < 10^{-4}$)	19.52 ($p \approx 0.8661$)	0.078 ($p \approx 0.1260$)	0.103 ($p \approx 0.6534$)	0.213 ($p < 10^{-4}$)
dense NN ECC1e5	12.98	0.031	25.93	6.759	5.526 ($p \approx 0.1115$)	19.01 ($p < 10^{-4}$)	0.125 ($p \approx 0.0001$)	0.142 ($p < 10^{-4}$)	0.200 ($p \approx 0.1014$)
dense NN Schaake1e5	12.98	0.031	25.93	6.759	5.142 ($p < 10^{-4}$)	19.02 ($p < 10^{-4}$)	0.117 ($p \approx 0.0001$)	0.135 ($p < 10^{-4}$)	0.227 ($p < 10^{-4}$)
cGAN	13.03	0.028	26.02	4.996	4.855 ($p < 10^{-4}$)	19.03 ($p < 10^{-4}$)	0.074 ($p \approx 0.0265$)	0.093 ($p \approx 0.0011$)	0.222 ($p < 10^{-4}$)

increasing lead time. Moreover, gEMOS skill exhibits a stronger daily cycle than dense NN or cGAN in autumn and winter.

5. Discussion

In line with the results by Baran et al. (2021), overall the machine learning approaches proved to outperform gEMOS, which we have used as classical reference approach. Furthermore, their results suggested that adding auxiliary predictors, e.g., other weather variables from the atmospheric model, to the post-processing model is more beneficial when using machine learning approaches compared to their classical counterparts. Though not shown here, our study confirms this in that adding auxiliary predictors to dense NN or cGAN improved skill, whereas adding noncloud predictors other than near-surface temperature to gEMOS typically decreased out-of-sample skill. Even though adding topography as static predictor to gEMOS did not improve forecast skill, we suppose that near-surface temperature provides some information related to topography that is beneficial to gEMOS. Moreover, we could not find any additional predictor that consistently improves fog and low stratus forecasts in winter. Tests using, e.g., pressure differences parallel and perpendicular to the Alps, relative humidity, or planetary boundary layer height did not improve out-of-sample skill of gEMOS.

It might not come as a surprise that dense NN tends to have a slightly better univariate performance than cGAN, since it was optimized for CRPS directly and not for the realistic appearance of its forecasts. Given its model complexity, it was also expected that dense NN would perform better than gEMOS. However, it is good to see that cGAN's generator can produce forecasts that are better than those from gEMOS, while seeing ground-truth observations only indirectly via the discriminator during training. Since forecast performance of cloud cover is rather low (Haiden et al. 2015, 2018), gEMOS and dense NN tend to regress to the mean, in particular at longer lead times. By design, cGAN models do not regress to the mean. As regression to the mean is desirable when predictability is low, cGAN models may experience a stronger decay in forecast skill with increasing lead time compared to dense NN or gEMOS.

In the second row of Fig. 6, we saw that gEMOS is able to significantly outperform COSMO-E at all grid points except those in the Swiss Plateau in winter. This could be caused by fog usually occurring in lower-altitude areas in winter, which is challenging to predict (Hewson 2019). In the last two rows of the figure, we can see that dense NN and cGAN seem to do a slightly better job at this. However, all considered methods fail to predict fog in a reliable way. As predicting fog for a typical anticyclonic day in winter, would literally mean inventing clouds, while having no cloud signal in the raw ensembles, postprocessing approaches would need to be able to switch between fog and other regimes. Lerch and Thorarinsdottir (2013) developed such a regime-switching EMOS approach for wind speed. Subject to being fed with all necessary predictors, dense NN and cGAN may be able to model regime switching straightforwardly.

FIG. 7. Mean CRPSS of gEMOS, dense NN, and cGAN with respect to the direct model output (DMO) of the best numerical model (either COSMO-E or IFS), averaged over all forecasts initialized at noon in (top) winter, (middle) summer, or (bottom) autumn; shown against (left) lead time or (right) hour of day. Results for spring are omitted, because of scarcity of reliable data. Color bar 1 indicates the CRPSS of COSMO-E vs IFS. From this information, one can deduce which of these two models is the “best numerical model.” Color bar 2 indicates the CRPSS of COSMO-E vs a climatological forecast.

A bit in contrast to the last paragraph, the average CRPS(S) results for winter shown in Fig. 7 indicate a particularly low CRPSS of the postprocessing models relative to the raw ensembles in the early morning, which then increases again in the following hours. This daily pattern of CRPSS may be related to fog and low stratus. Westerhuis et al. (2020) show that the COSMO model dissipates fog and low stratus clouds in winter too quickly. Hence, it is likely that fog and low stratus cloud forecasts of COSMO-E are best in terms of CRPS in the early morning and increasingly deteriorate over the day. To some extent, postprocessing may be able to correct for this erroneous dissipation leading to an increase in CRPSS.

Moreover, we need to take into account that the forecasts from dense NN and cGAN are less explainable and interpretable than

those from gEMOS, due to the nature of deep learning. Tests also showed that cGAN is less data-efficient (i.e., requires more training data) than the other two methods.

As pointed out already, the main advantage of using cGAN lies in its ability to generate physically coherent spatial scenarios without any dependence template. Furthermore, as suggested by the 0.5-variogram score, cGAN outperforms ECC-based approaches in terms of multivariate verification. Nevertheless, the energy score does not confirm this. According to recent simulation studies by Ziel and Berk (2019) and Lerch et al. (2020), the energy score exhibits at least as good multivariate discrimination ability as the p -variogram score. However, since we are working with real data, we cannot control the marginal distributions, which makes it difficult to discern the univariate

(i.e., location and dispersion) errors from the multivariate (i.e., erroneous correlation structure) contributions to the multivariate scores. For instance, a spatially constant additive bias leads to a deterioration of the energy score, while the p -variogram score does not change. Moreover, cGAN exhibits the lowest CRPS-f0 (not significant) and CRPS-f100 scores. Nevertheless, applying ScS instead of ECC leads to an improvement in terms of multivariate verification for both gEMOS and dense NN.

A drawback of cGAN—this also holds for our naive ScS implementation—compared to ECC is that intervariable and temporal dependencies are not captured. We did try out taking the time dimension into account using a recurrent cGAN—a recurrent version of cGAN with similar architectures (using LSTM layers). The first results looked promising, but we abandoned the approach due to its computational intensity. Using ECC also on cGAN predictions, one could indeed incorporate temporal dependencies. However, this would make the cGAN scenarios less realistic in space.

6. Conclusions

We showed that the cloud cover postprocessing problem can successfully be tackled using generative adversarial networks. Our cGAN method turned out to be capable of considerably outperforming the numerical model COSMO-E in terms of CRPS, and had better overall forecast skill than the classical statistical method gEMOS as well. cGAN was not able to achieve better CRPS than dense NN, but produced spatially realistic forecasts without any need for a spatial dependence template. Not only can this be judged with the human eye; it is reflected in the multivariate metrics as well. Interestingly, also dense NN with ScS spatial dependence structure performs well in terms of multivariate verification. Simply put, both cGAN and ScS obtain spatial patterns from historical CM SAF observations. Our experience with optimizing the cGAN model also suggests that having strong forecast skill and spatially realistic forecasts at the same time is only possible to a certain extent; a trade-off between these two desires needs to be found.

To find this trade-off, one could think of a hybrid approach between dense NN and cGAN, in which cGAN's discriminator could be used to train dense NN to give more realistic forecasts; which might be an interesting area for further exploration. Other potential topics for further research include: (i) a more appropriate postprocessing scheme for fog, (ii) a theoretical assessment of the properties of different multivariate scores for doubly censored variables, and (iii) an extension of cGAN to spatiotemporal scenarios (Qin and Gao 2020).

Acknowledgments. First, we are grateful to the editor and the three anonymous reviewers. Their valuable comments helped to improve the quality of this paper considerably. We would also like to thank D. Nerini for raising the idea of using cGANs for postprocessing of cloud cover ensemble forecasts. We are grateful to A. Duguay-Tetzlaff and R. Stöckli for generating and providing the CM SAF data. We would also like to thank M. Maathuis for supervising the master thesis of Y. Dai, on which important parts of this article are based.

Moreover, we are indebted to J. Bhend, C. Spirig, J. Rajczak, L. Moret, M. Liniger, and R. Furrer for fruitful discussions and valuable inputs. We would also like to thank P. Falke for helping us with extracting the forecast data. Furthermore, the authors wish to thank the Swiss National Supercomputing Centre (CSCS) for providing technical infrastructure.

Data availability statement. ECMWF-IFS data are publicly available and can be downloaded from ECMWF's online archive system MARS <https://www.ecmwf.int/en/forecasts/datasets/archive-datasets>. Due to MeteoSwiss' data policy, COSMO-E data are not publicly available, but can be requested for research purposes through MeteoSwiss' customer service <https://www.meteoswiss.admin.ch/home/form/customer-service.html>. The EUMETSAT/CM SAF data we have used correspond to the prototype version 2.0 to be released and made publicly available in 2022. The previous version is already available through the CM SAF webpage www.cmsaf.eu. When exploiting EUMETSAT/CM SAF data you are kindly requested to acknowledge this contribution accordingly and make reference to the CM SAF, e.g. by stating "The work performed was done (i.a.) by using data from EUMETSAT's Satellite Application Facility on Climate Monitoring (CM SAF)."

REFERENCES

- Anderson, J. L., 1996: A method for producing and evaluating probabilistic forecasts from ensemble model integrations. *J. Climate*, **9**, 1518–1530, [https://doi.org/10.1175/1520-0442\(1996\)009<1518:AMFPAE>2.0.CO;2](https://doi.org/10.1175/1520-0442(1996)009<1518:AMFPAE>2.0.CO;2).
- Baran, A., S. Lerch, M. El Ayari, and S. Baran, 2021: Machine learning for total cloud cover prediction. *Neural Comput. Appl.*, **33**, 2605–2620, <https://doi.org/10.1007/s00521-020-05139-4>.
- Bremnes, J. B., 2019: Ensemble postprocessing using quantile function regression based on neural networks and Bernstein polynomials. *Mon. Wea. Rev.*, **148**, 403–414, <https://doi.org/10.1175/MWR-D-19-0227.1>.
- Chen, H., X. Zhang, Y. Liu, and Q. Zeng, 2019: Generative adversarial networks capabilities for super-resolution reconstruction of weather radar echo images. *Atmosphere*, **10**, 555, <https://doi.org/10.3390/atmos10090555>.
- Clark, M., S. Gangopadhyay, L. Hay, B. Rajagopalan, and R. Wilby, 2004: The Schaake shuffle: A method for reconstructing space–time variability in forecasted precipitation and temperature fields. *J. Hydrometeorol.*, **5**, 243–262, [https://doi.org/10.1175/1525-7541\(2004\)005<0243:TSSAMF>2.0.CO;2](https://doi.org/10.1175/1525-7541(2004)005<0243:TSSAMF>2.0.CO;2).
- Diebold, F. X., and R. S. Mariano, 1995: Comparing predictive accuracy. *J. Bus. Econ. Stat.*, **13**, 253–263, <https://doi.org/10.2307/1392185>.
- Dupuy, F., and Coauthors, 2021: ARPEGE cloud cover forecast postprocessing with convolutional neural network. *Wea. Forecasting*, **36**, 567–586, <https://doi.org/10.1175/WAF-D-20-0093.1>.
- Elmore, K. L., 2005: Alternatives to the chi-square test for evaluating rank histograms from ensemble forecasts. *Wea. Forecasting*, **20**, 789–795, <https://doi.org/10.1175/WAF884.1>.
- Gagne, D., H. Christensen, A. Subramanian, and A. Monahan, 2019: Machine learning for stochastic parameterization: Generative adversarial networks in the Lorenz '96 model. *J. Adv. Model. Earth Syst.*, **12**, e2019MS001896, <https://doi.org/10.1029/2019MS001896>.

- Gneiting, T., and A. E. Raftery, 2007: Strictly proper scoring rules, prediction, and estimation. *J. Amer. Stat. Assoc.*, **102**, 359–378, <https://doi.org/10.1198/016214506000001437>.
- , —, A. H. Westveld, and T. Goldman, 2005: Calibrated probabilistic forecasting using ensemble model output statistics and minimum CRPS estimation. *Mon. Wea. Rev.*, **133**, 1098–1118, <https://doi.org/10.1175/MWR2904.1>.
- Goodfellow, I. J., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, 2014: Generative adversarial networks. <https://arxiv.org/abs/1406.2661>.
- Haiden, T., R. Forbes, M. Ahlgrimm, and A. Bozzo, 2015: The skill of ECMWF cloudiness forecasts. *ECMWF Newsletter*, No. 143, ECMWF, Reading, United Kingdom, 14–19, <https://doi.org/10.21957/lee5bz2g>.
- , M. Janousek, J. Bidlot, R. Buizza, L. Ferranti, F. Prates, and F. Vitart, 2018: Evaluation of ECMWF forecasts, including the 2018 upgrade. ECMWF Tech. Memo. 831, ECMWF, 54 pp., <https://www.ecmwf.int/sites/default/files/elibrary/2018/18746-evaluation-ecmwf-forecasts-including-2018-upgrade.pdf>.
- Hemri, S., T. Haiden, and F. Pappenberger, 2016: Discrete post-processing of total cloud cover ensemble forecasts. *Mon. Wea. Rev.*, **144**, 2565–2577, <https://doi.org/10.1175/MWR-D-15-0426.1>.
- Hewson, T., 2019: Use and verification of ECMWF products in member and co-operating states (2018). ECMWF Tech. Memo. 840, 40 pp., <https://www.ecmwf.int/sites/default/files/elibrary/2019/18911-use-and-verification-ecmwf-products-member-and-co-operating-states-2018.pdf>.
- Hinton, G. E., N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, 2012: Improving neural networks by preventing co-adaptation of feature detectors. <https://arxiv.org/abs/1207.0580>.
- Ioffe, S., and C. Szegedy, 2015: Batch normalization: Accelerating deep network training by reducing internal covariate shift. <https://arxiv.org/abs/1502.03167>.
- Jones, K., 2017: GANGogh: Creating art with GANs. Accessed 16 July 2020, <https://towardsdatascience.com/gangogh-creating-art-with-gans-8d087d8f74a1>.
- Juzeniene, A., and Coauthors, 2011: Solar radiation and human health. *Rep. Prog. Phys.*, **74**, 066701, <https://doi.org/10.1088/0034-4885/74/6/066701>.
- Karras, T., T. Aila, S. Laine, and J. Lehtinen, 2017: Progressive growing of GANs for improved quality, stability, and variation. <https://arxiv.org/abs/1710.10196>.
- Klasa, C., M. Arpagaus, A. Walser, and H. Wernli, 2018: An evaluation of the convection-permitting ensemble COSMO-E for three contrasting precipitation events in Switzerland. *Quart. J. Roy. Meteor. Soc.*, **144**, 744–764, <https://doi.org/10.1002/qj.3245>.
- Lecun, Y., L. Bottou, Y. Bengio, and P. Haffner, 1998: Gradient-based learning applied to document recognition. *Proc. IEEE*, **86**, 2278–2324, <https://doi.org/10.1109/5.726791>.
- Leinonen, J., D. Nerini, and A. Berne, 2020: Stochastic super-resolution for downscaling time-evolving atmospheric fields with a generative adversarial network. *IEEE Trans. Geosci. Remote Sens.*, **59**, 7211–7223, <https://doi.org/10.1109/TGRS.2020.3032790>.
- Lerch, S., and T. L. Thorarinsdottir, 2013: Comparison of non-homogeneous regression models for probabilistic wind speed forecasting. *Tellus*, **65A**, 21206, <https://doi.org/10.3402/tellusa.v65i0.21206>.
- , S. Baran, A. Möller, J. Groß, R. Schefzik, S. Hemri, and M. Graeter, 2020: Simulation-based comparison of multivariate ensemble post-processing methods. *Nonlinear Processes Geophys.*, **27**, 349–371, <https://doi.org/10.5194/npg-27-349-2020>.
- Maas, A. L., A. Y. Hannun, and A. Y. Ng, 2013: Rectifier nonlinearities improve neural network acoustic models. *Proc. 30th Int. Conf. on Machine Learning*, Atlanta, GA, ICML, 6 pp.
- Matheson, J. E., and R. L. Winkler, 1976: Scoring rules for continuous probability distributions. *Manage. Sci.*, **22**, 1087–1096, <https://doi.org/10.1287/mnsc.22.10.1087>.
- Matuszko, D., 2012: Influence of the extent and genera of cloud cover on solar radiation intensity. *Int. J. Climatol.*, **32**, 2403–2414, <https://doi.org/10.1002/joc.2432>.
- Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, 2013: Distributed representations of words and phrases and their compositionality. <https://arxiv.org/abs/1310.4546>.
- Mirza, M., and S. Osindero, 2014: Conditional generative adversarial nets. <https://arxiv.org/abs/1411.1784>.
- Owens, R. G., and T. Hewson, 2018: ECMWF Forecast User Guide. ECMWF, Reading, United Kingdom, p16559, <https://doi.org/10.21957/mlcs7h>.
- Palz, W., 2013: *Solar Power for the World: What You Wanted to Know about Photovoltaics*. CRC Press, 800 pp.
- Pathak, D., P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, 2016: Context encoders: Feature learning by inpainting. <https://arxiv.org/abs/1604.07379>.
- Pennington, J., R. Socher, and C. Manning, 2014: GloVe: Global vectors for word representation. *Proc. 2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, Association for Computational Linguistics, 1532–1543, <https://doi.org/10.3115/v1/D14-1162>.
- Qin, C., and X. Gao, 2020: Spatio-temporal generative adversarial networks. *Chin. J. Electron.*, **29**, 623–631, <https://doi.org/10.1049/cje.2020.04.001>.
- Rasp, S., and S. Lerch, 2018: Neural networks for postprocessing ensemble weather forecasts. *Mon. Wea. Rev.*, **146**, 3885–3900, <https://doi.org/10.1175/MWR-D-18-0187.1>.
- Ronneberger, O., P. Fischer, and T. Brox, 2015: U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*, N. Navab et al., Eds., Springer, https://doi.org/10.1007/978-3-319-24574-4_28 234–241.
- Santer, B., T. Wigley, and P. Jones, 1993: Correlation methods in fingerprint detection studies. *Climate Dyn.*, **8**, 265–276, <https://doi.org/10.1007/BF00209666>.
- Schefzik, R., 2016: A similarity-based implementation of the Schaaek shuffle. *Mon. Wea. Rev.*, **144**, 1909–1921, <https://doi.org/10.1175/MWR-D-15-0227.1>.
- , and A. Möller, 2018: Ensemble postprocessing methods incorporating dependence structures. *Statistical Postprocessing of Ensemble Forecasts*, S. Vannitsem, D. S. Wilks, and J. W. Messner, Eds., Elsevier, 91–125, <https://doi.org/10.1016/B978-0-12-812372-0.00004-2>.
- , T. L. Thorarinsdottir, and T. Gneiting, 2013: Uncertainty quantification in complex simulation models using ensemble copula coupling. *Stat. Sci.*, **28**, 616–640, <https://doi.org/10.1214/13-STS443>.
- Scher, S., and S. Peßenteiner, 2021: Technical note: Temporal disaggregation of spatial rainfall fields with generative adversarial networks. *Hydrol. Earth Syst. Sci.*, **25**, 3207–3225, <https://doi.org/10.5194/hess-25-3207-2021>.
- Scheuerer, M., and T. M. Hamill, 2015: Variogram-based proper scoring rules for probabilistic forecasts of multivariate quantities. *Mon. Wea. Rev.*, **143**, 1321–1334, <https://doi.org/10.1175/MWR-D-14-00269.1>.
- , and —, 2018: Generating calibrated ensembles of physically realistic, high-resolution precipitation forecast fields

- based on GEFS model output. *J. Hydrometeor.*, **19**, 1651–1670, <https://doi.org/10.1175/JHM-D-18-0067.1>.
- Shi, X., Z. Gao, L. Lausen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, 2017: Deep learning for precipitation nowcasting: A benchmark and a new model. *Proc. 31st Conf. on Neural Info. Process. Syst. (NIPS 2017)*, Long Beach, CA, NIPS, 5617–5627.
- Stöckli, R., J. S. Bojanowski, V. O. John, A. Duguay-Tetzlaff, Q. Bourgeois, J. Schulz, and R. Hollmann, 2019: Cloud detection with historical geostationary satellite sensors for climate applications. *Remote Sens.*, **11**, 1052, <https://doi.org/10.3390/rs11091052>.
- Westerhuis, S., O. Fuhrer, J. Cermak, and W. Eugster, 2020: Identifying the key challenges for fog and low stratus forecasting in complex terrain. *Quart. J. Roy. Meteor. Soc.*, **146**, 3347–3367, <https://doi.org/10.1002/qj.3849>.
- Wilks, D., 2006: On “field significance” and the false discovery rate. *J. Appl. Meteor. Climatol.*, **45**, 1181–1189, <https://doi.org/10.1175/JAM2404.1>.
- Winkler, R. L., 1994: Evaluating probabilities: Asymmetric scoring rules. *Manage. Sci.*, **40**, 1395–1405, <https://doi.org/10.1287/mnsc.40.11.1395>.
- Ye, Q., and S. Chen, 2013: The ultimate meteorological question from observational astronomers: How good is the cloud cover forecast? *Mon. Not. Roy. Astron. Soc.*, **428**, 3288–3294, <https://doi.org/10.1093/mnras/sts278>.
- Zhu, J.-Y., T. Park, P. Isola, and A. A. Efros, 2017: Unpaired image-to-image translation using cycle-consistent adversarial networks. *Proc. IEEE Int. Conf. on Computer Vision*, Venice, Italy, IEEE, 2223–2232, <https://doi.org/10.1109/ICCV.2017.244>.
- Ziel, F., and K. Berk, 2019: Multivariate forecasting evaluation: On sensitive and strictly proper scoring rules. <https://arxiv.org/abs/1910.07325>.