



Lossy compression by dimension reduction - Evaluation of methods applied to 2-d meteorological fields for data-based modeling

Uwe Ehret¹, Jieyu Chen², Fedor Scholz³, Sam Allen⁴, Sebastian Lerch⁵

¹Institute of Water and Environment, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

5 ²Institute of Statistics, Nanjing University of Information Science and Technology, Nanjing, China

³Stuttgart Center for Simulation Science, Cluster of Excellence EXC 2075, University of Stuttgart, Stuttgart, Germany

⁴Institute of Statistics, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

⁵Department of Mathematics and Computer Science, Marburg University, Marburg, Germany

Correspondence to: Uwe Ehret (uwe.ehret@kit.edu)

10 **Abstract.** Data sets in the geosciences continue to grow in size, and data compression methods can facilitate efficient data storage, transfer, and utilization. Related workflows in geoscience modelling typically involve data generation (e.g. via weather forecasting), followed by compression, transfer, decompression, extraction of local subsets, and integrations into downstream tasks such as hydrological modelling. Recently, data-driven models have been introduced into geoscientific workflows with great success, allowing input data sources and types to be used in much more flexible ways than in classical numerical models, 15 e.g. by directly using the compressed data. In this study, we therefore address the questions on 1) how various lossy compression algorithms compare in terms of general properties such as computational effort and their ability to extract spatial subsets directly from the compressed representation, and 2) how they compare in terms of compression efficiency and reproduction error. We focus on dimension reduction methods, as the resulting low-dimensional representations require fewer input channels in downstream data-driven models, thereby reducing computational costs and the risk of overfitting. We 20 compare several dimension reduction methods in an application to five years of hourly air temperature and rainfall data, in the form of 2-d gridded spatial fields spanning a 100,000 km² domain in central Germany. The methods we compare are: Block Averaging, Principal Component Analysis, an Autoencoder, and the Ramer-Douglas-Peucker algorithm. We measure compression by the number of distinguishable objects in the compressed representation, rather than by file size, keeping in mind the potential use of the compressed data as input to data-driven models. This approach directly relates dimension 25 reduction/compression to the number of input channels of a data-driven model, which is typically limited to avoid overfitting. Our results indicate that the methods differ substantially in terms of training and computational overhead, preservation of field-scale statistics, and the possibility to extract spatial subsets directly from the compressed representation. All methods demonstrated very good compression efficiency, with low reconstruction errors even for 99% compression. For the spatially smooth temperature fields, Principal Component Analysis performed best, followed by the Autoencoder. For the spatially 30 heterogeneous rainfall fields, the Autoencoder and the Ramer-Douglas-Peucker algorithm were most effective. The choice of the best method therefore depends on the specific application. This study provides a step towards a more efficient use of geoscientific data in downstream applications, either by directly using compressed data as input to data-driven models, or by efficiently extracting spatial subsets from compressed data, and using these subsets as input to local numerical models.

1 Introduction

35 Due to advances in observational methods (McCabe et al., 2017), higher model resolutions (Schär et al., 2020, Bauer et al., 2020), and increased computational power (Bauer et al., 2015), geoscientific data sets have grown considerably in size, and continue to do so (Graham and Shelton, 2013, Li et al., 2023). As a result, storing data for operational purposes and long-term archiving is a growing infrastructural challenge. This is particularly so for 2-d gridded meteorological fields, which serve as a test case in this study. To mitigate the challenges of data transfer and storage, many methods for lossless and lossy data 40 compression have been proposed and are used, such as the bzip2 and gzip data formats (lossless, Walters and Wong, 2023),



the GRIB data format of the World Meteorological Organization (lossless), ZFP (lossy, Lindstrom, 2014), and many others (Tintó Prims et al., 2024, and references therein; Duwe et al., 2020; Sayood, 2017). The efficiency of such methods is typically measured by comparing the size of the original and compressed files, and, for lossy compression, analysing the (dis-)agreement between the original and its reconstruction. Klöwer et al. (2021) demonstrate that, due to strong spatio-temporal correlations, atmospheric data can generally be greatly compressed without a significant loss of information, and a publicly available climate dataset that has undergone lossy compression is analysed and evaluated by Baker et al., (2016).

In the geosciences, a typical workflow in this context consists of compressing the original file, transferring or archiving it, decompressing it to its original format, possibly extracting subsets corresponding to regions of interest, and using the extracted data as input for downstream models. For example, output from ICON-RUC numerical ensemble weather forecasts is provided by the German Weather Service in grib format on their ftp servers (Mueller and Barleben, 2024). From there, the weather forecasts are downloaded and decompressed by, for example, operational hydrological services. River basin regions are then extracted from the decompressed file, and used as input to process-based hydrological forecasting models, such as LARSIM (Ludwig and Bremicker, 2006).

Traditionally, most downstream models are physical, spatial, and numerical representations of real-world domains and processes. The mapping of input data to model elements must therefore be coordinated by location, often requiring upscaling or downscaling to align the resolutions of the data and the model, and the data must be provided in the physical units required by the numerical process in which they are employed. Alternatively, data-driven models have recently been successfully applied in the context of geoscientific modelling, and (hydro-)meteorological modelling in particular (e.g. Gardner et al., 1998, Reichstein et al., 2019, Ben Bouallegue et al., 2024, Acuña Espinoza et al., 2024). These models are predominantly based on neural network architectures, such as Transformers (Vaswani et al., 2017) or Long Short-Term Memory (LSTM; Hochreiter and Schmidhuber, 1997) networks. In contrast to physical models, they are considerably more flexible in terms of the number, source, location and units of their input data, which partly explains their competitive performance in practical applications. While this additional flexibility makes data-driven models susceptible to overfitting, this can be mitigated by out-of-sample evaluation during model training, and by limiting the number of input channels of the models. For example, spatial rainfall forecasts are usually reduced to a single river basin-average rainfall value when used as input for hydrological LSTM models (see Yu et al., 2024 and references therein).

A natural question to ask is whether the traditional workflow of data compression, transfer, and decompression can be circumvented by directly using the compressed data as input to data-driven models, which would make the entire workflow considerably more efficient. The risk of overfitting will additionally be reduced if the dimensionality of the compressed data - and consequently the number of input channels for the data-driven model - is much smaller than that of the original data. The compression would then effectively replace existing preprocessing steps for dimension reduction, such as the basin-averaging described above. Many methods for achieving (lossy) data compression through dimension reduction exist. For reviews, see Carreira, 1997; Fodor, 2002; Lee and Verleyson, 2007; van der Maaten et al., 2009. For such methods, and for this purpose, the level of compression should be measured by the number of distinguishable objects in the compressed representation, rather than by its file size.

Using compressed data as input for data-driven models is an active field of research (Jia et al., 2022). In this context, the goal of this study is to compare several dimension reduction algorithms with respect to their suitability for use in data-driven geoscientific modelling workflows. We focus on 2-d gridded meteorological fields, which are regularly used as input for



downstream models, such as hydrological, agricultural, or ecological land surface models. In particular, we seek to answer two research questions:

- 85 • RQ 1: How do dimension reduction algorithms compare with respect to training effort (if any), and preservation of spatial location in the compressed representation? The latter aspect is important if spatial subsets need to be extracted directly from the compressed representation, without first reconstructing the entire field.
- RQ 2: How well do the dimension reduction algorithms perform in terms of reconstruction error at different compression levels?

90 We address RQ 1 and RQ 2 via an application of state-of-the-art dimension reduction methods to long, hourly time series of two important meteorological variables that exhibit contrasting behaviour: air temperature and rainfall. The former is spatially and temporally smooth, while the latter is intermittent and highly variable. We compare four methods that widely differ in their characteristics: Block Averaging (BA), Principal Component Analysis (PCA), the Ramer-Douglas-Peucker algorithm (RDP), and an Autoencoder (AE). We apply them at a range of compression levels (i.e. number of objects in the compressed
95 representation) and analyse the relationship between the compression level and the error of the reconstructed fields. The reconstruction error is evaluated using the Mean Absolute Error, Mean Square Error, and Multiscale Structural Similarity Index. We also analyse the effect of compression on field-scale statistics, such as the mean, standard deviation, and range of the data. Finally, we compare the methods with respect to their training requirements, and whether spatial subsets can be recovered directly from the compressed representation; both characteristics are important for their use in data-based modelling
100 workflows.

Many other studies have proposed using compressed data within statistical and machine learning models. The use of PCA to compress model input data dates back at least to Massy (1965), who used principal components as inputs to a linear regression model. More recently, several studies have applied this general framework to large climate data sets and machine learning models: for example, Garcia-Hinde et al. (2018) apply competing dimension reduction techniques to numerical weather model
105 output, and use the compressed representations as inputs to solar radiation forecasting models; Lerch and Polsterer (2022) leverage convolutional autoencoders to learn low-dimensional representations of spatial weather forecast fields, which are then used as inputs to data-driven post-processing models; Bonneau et al. (2024) forecast wheat yields using a random forest with predictors obtained by Functional Principal Component Analysis applied to large amounts of spatio-temporal climate data; Ritter et al. (2022) discuss a two-step forecasting approach whereby dimension reduction is used for predictor selection in a
110 normalizing flow-based weather forecasting model; and Huang and Hoefler (2022) propose a novel compression method that is subsequently used to determine the inputs used to train data-driven weather models. Our work differs in scope from these studies, in that we provide a more systematic comparison of different dimension reduction methods themselves, rather than focusing on a particular downstream application.

The remainder of the article is structured as follows. In the next section, we introduce the data to be compressed, the methods
115 used for dimension reduction, and the methods used to evaluate them. The results are presented and discussed in Sect. 3, before we conclude and suggest possible directions for future research in Sect. 4.

2 Data and Methods

2.1 Dimension reduction methods

In the following, we describe the dimension reduction methods used in this study. We use x and y to denote spatial coordinates, and $z(x, y)$ to represent a grid value at location (x, y) . The subscript $z_o(x, y)$ indicates original observed values, z_c indicates
120 values in the compressed representation, $z_r(x, y)$ indicates values reconstructed from the compressed to the original representation; note that the compressed representation z_c corresponds to a lower-dimensional latent space, and is thus not a



direct function of the spatial coordinates, unlike z_o and z_r , which are defined on the original spatial domain. N_g denotes the total number of gridded locations in the studied domain.

2.1.1 Block averaging

Block averaging (BA) is a straightforward way to represent a spatial gridded field using fewer objects, and it therefore serves as a baseline in this study. BA consists of replacing a set of spatially neighbouring grid cells in a rectangular sub-domain (“block”) with their average value, according to Eq. (1),

$$z_c = \frac{1}{N_i \cdot N_j} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} z_o(x_i, y_j) \quad (1)$$

where N_i and N_j are the number of grid points in the block in the x - and y -directions, respectively. This is repeated for distinct, non-overlapping blocks that cover the original field, and the average z_c then serves as the representative value for all $N_i \cdot N_j$ grid points in the corresponding block. In our application, this reduces the amount of data points in each field from $N_g = 400 \times 250$ to $(400/N_i) \times (250/N_j)$. The only parameters of BA are N_i and N_j , which determine the block size; after these have been chosen, BA can be applied without training. Reconstruction of the original field from the compressed representation is straightforward, and consists of assigning to each original grid location the corresponding block average z_c . The reconstruction error results from the averaging procedure, and will increase with the block size. In this study, we compare a variety of block sizes.

2.1.2 Principal Component Analysis

Principal component analysis (PCA), proposed by Pearson (1901), is a linear method that reduces dimension via linear projections that are designed to maximize the amount of variance in the input data that is retained by the compressed data. PCA constructs orthogonal axes (principal components) as linear combinations of the input features, where the primary axis (the first principal component) captures the direction of maximum data variance. Subsequent axes (higher-order principal components) each capture the maximum remaining variance subject to orthogonality to all preceding axes. Since PCA achieves high-dimensional data compression by retaining the maximum proportion of total variance (Jolliffe and Cadima, 2016), it is particularly effective for structured spatiotemporal fields such as those considered here, where a large proportion of the variation is explained by a relatively small number of weather patterns. For the observation dataset z_o in this study, each gridded location (x, y) is treated as one input feature, and the dimensionality of the input data z_o is thus the total number of gridded locations in the studied domain, N_g . PCA yields exactly N_g principal components through the orthogonal transformation, and the compressed representation z_c is then defined by the first k principal components, where k is the compression level to be specified. Reconstruction error primarily arises from the orthogonal projection of the input data onto the truncated subspace spanned by the first k principal components, and this information loss decreases monotonically as k approaches the full data dimension. The orthogonality of the principal components guarantees an invertible transformation between the original and compressed features, which allows for data reconstruction z_r from the compressed representation z_c . The principal components are estimated from the training data and subsequently kept fixed when the transformation is applied to unseen test data. Once the principal components are identified, both obtaining the compressed representation and reconstructing the original field require only matrix operations, resulting in very low computational costs. We examine the reconstruction performance for different choices of k .

2.1.3 Ramer-Douglas-Peucker

The Ramer-Douglas-Peucker algorithm (RDP) was independently suggested by Ramer (1972) and Douglas and Peucker (1973) as an iterative procedure to approximate high-resolution 2-d polygons by subsets of their vertices. Developed in the field of



160 image processing, it is mainly used to compress images while preserving their main features. The RDP algorithm starts with a coarse approximation of the original polygon by connecting its end vertices by linear interpolation. In our study, the original polygon corresponds to the spatial field, and the vertices correspond to relevant grid points. From the original polygon, the vertex (i.e. grid point) with the largest difference between original value and the value approximated by the model is selected. The new vertex is included into the model, the linear interpolation is repeated, and the next vertex to include is identified. This procedure is iteratively repeated until either a maximum number of vertices k in the approximation, or a desired level of agreement ε between the original polygon and the approximation, is reached. The only parameter of the RDP algorithm is k or ε ; once this has been chosen, RDP can be applied without training. However, since the algorithm is an iterative process, its implementation can be computationally expensive, see Ramer (1972) for details. Particular advantages of the RDP algorithm are that the approximations are directly constructed from the original data, thus avoiding averaging losses whilst preserving extreme values. As already suggested by Ramer (1972), the algorithm easily extends to polygons in higher dimensions as long as linear interpolation between vertices is possible, and distances between vertices and the polygons can be calculated. For this study, the RDP compression consists of iteratively selecting a predefined number of points (x, y) from the original field z_o , where the number of points defines the compression level k . The selected set of k points $\{z_o(x_1, y_1), z_o(x_2, y_2), \dots, z_o(x_k, y_k)\}$ will be irregularly spaced, depending on the distribution of temperature or rainfall values in the field. Reconstruction of the original field uses the selected k points and triangulation onto the original grid locations, i.e. linear interpolation in the 3-dimensional (x, y, z) field, where z represents the value of temperature or rainfall. The reconstruction error results from the use of only k support points and will increase as k decreases. We again compare the results for different choices of k .

2.1.4 Autoencoder

An autoencoder (AE) is an artificial neural network designed to learn efficient representations of data (Goodfellow et al., 2016). The AE consists of two main components: an encoder that compresses the input into a latent space of lower dimensionality, and a decoder that attempts to reconstruct the input from this compressed representation. The encoder f progressively reduces the dimensionality of the input z_o through multiple layers, yielding the latent representation $z_c = f(z_o)$. The decoder g then produces a reconstruction $z_r = g(z_c) = g(f(z_o))$. The encoder and decoder are neural networks, whose parameters are estimated by minimising the reconstruction error over a training data set, with the error typically defined as the mean squared error of the reconstructed fields compared to the original fields. By training the encoder and decoder together in an end-to-end manner, the encoder learns to compress the input in a way that enables the decoder to better reconstruct the original data. While AEs can be used to compress any kind of data, they are most often used for gridded inputs like images or meteorological fields. In these cases, the components of the AE are typically convolutional neural networks (CNNs). The encoder CNN progressively reduces the spatial resolution of the input whilst increasing the number of feature channels. Each feature channel represents a different learned characteristic of the data. The data is typically then flattened and sent through a linear layer, resulting in a *bottleneck* that represents the compressed field. The decoder reverses this process using transposed convolutions (sometimes called deconvolutions). For spatially structured data, ConvNeXt-based AE architectures (Liu et al., 2022) are commonly employed since they often achieve better efficiency than standard CNNs while maintaining strong performance in terms of reconstruction error. ConvNeXt blocks use depth-wise convolutions, which process spatial patterns within each feature channel separately, followed by point-wise convolutions that mix information across channels. To stabilize training and enable effective learning in deeper networks, ConvNeXt incorporates two important techniques: layer normalization, which normalizes activations within each layer, and residual connections, which add the original input to the processed output with a learnable scaling parameter for each output channel. This combination of architectural features makes ConvNeXt particularly well-suited for learning compressed representations of spatially structured data. The main source of reconstruction error in AE compression stems from the information loss imposed by the bottleneck constraint. The quality of reconstruction depends on the size of the bottleneck and how well the model has learned the statistical patterns in the training



data. Unlike some other compression methods, AEs require extensive training on representative data to learn effective compression strategies. Once an AE is successfully trained, though, inference becomes computationally cheap.

2.2 Data

- 205 In this study, we apply the competing dimension reduction methods to high-resolution (1 hour, 1 km) gridded observations of air temperature 2m above ground (henceforth T2) and ground rainfall rate (henceforth RR). The observation fields are on a large (100,000 km²) rectangular domain in central Germany, and are studied over the five-year period 1 January 2019 00:00 – 31 December 2023 23:00 UTC. These choices were based on the following criteria:
- 210 • *Data type:* The data should be relevant for many meteorological and downstream applications, and they should represent a wide range of spatio-temporal characteristics such as seasonality, temporal intermittency, and spatial variability. T2 exhibits diurnal and seasonal variation, its short-term and short-range variability is typically small, and values follow a broadly symmetrical distribution with frequent intermediate values and few extremes. RR, on the other hand, is highly variable, both spatially and temporally, and follows a highly skewed distribution due to its intermittent nature. Both T2 and RR are key variables in weather forecasting, and are used in many downstream applications such as in hydrology or agriculture.
 - 215 • *Spatial domain and resolution:* As we expect that loss of detail through compression will be largest for highly-resolved spatially variable fields, we compare the methods using 1-km gridded data. The data covers a region of Germany in the mid-latitudes, which exhibits both strong seasonal variation and a varied topography, ranging from lowlands to mountainous areas (see Fig. 1). The rectangular region facilitates the application of compression algorithms, and we selected Germany because the German Weather Service (Deutscher Wetterdienst, henceforth DWD) offers free and open access to a wide range of nationwide, high-resolution meteorological products. The domain extends 400 km in the North-South direction and 250 km in East-West direction, and therefore comprises $N_g = 100,000$ grid points.
 - 220 • *Temporal domain and resolution:* Hourly data was selected for two reasons: Firstly, temporally highly resolved data typically show higher spatial variability than time-aggregated data, and we expect that compression losses will be larger for spatially variable data. Secondly, most operationally relevant meteorological products, such as ensemble weather forecasts, are produced hourly. The storage volume, and with it the potential benefit of data compression, is also much larger for high-resolution data than for e.g. daily products. As some compression methods require training, the 5-years of data are divided into three years (2019-2021) for training, and two years (2022-2023) for testing.
 - 225 • Overall, the data set comprises $N_t = 43,824$ time steps, split into $N_{t,train} = 26,304$ for training and $N_{t,test} = 17,520$ for testing. Please note that due to the strong temporal autocorrelation of meteorological fields, the samples in the validation set cannot be considered fully independent.
 - 230

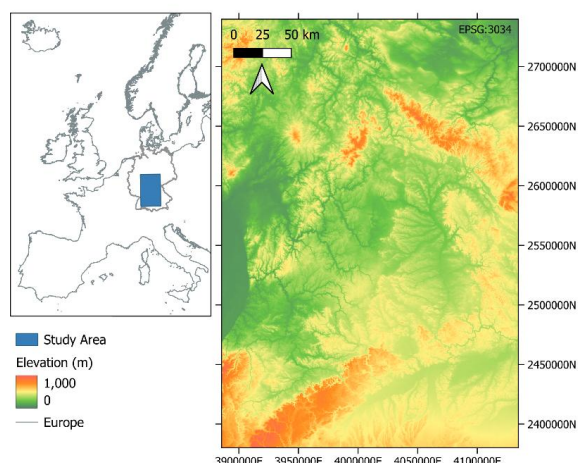


Figure 1. Topographic map of the study domain.

2.2.1 Air temperature data

The T2 air temperature data (in units [$^{\circ}\text{C}$]) was taken from the DWD “HOSTRADA” product. HOSTRADA comprises Germany-wide hourly station observations interpolated to a 1 km grid, for many relevant meteorological variables including air temperature, cloud cover, dew point, air pressure and radiation, but not rainfall. The HOSTRADA data set and interpolation methods are described in DWD (2024) and Krähenmann et al. (2018). The original data are available at https://opendata.dwd.de/climate_environment/CDC/grids_germany/hourly/hostrada/air_temperature_mean. The extracted and reformatted data are available in Ehret et al. (2026a). There were no missing data in the selected period and domain. The mean, minimum and maximum temperature in the data set (over all time steps and grid points) are 9.9°C , -23.9°C and 40.1°C , respectively. Spatial plots of T2 for selected times can be found in Fig. 5 and Fig. 6.

2.2.2 Rainfall data

The RR rainfall data (in units [mm/h]) was taken from the DWD “RADKLIM” product. RADKLIM comprises Germany-wide data derived from gauge-adjusted weather radar data mapped to a 1 km grid. The raw data and further information are available at Winterrath et al. (2018). The extracted and reformatted data are available in Ehret et al. (2026a). In the selected time period and domain, 3252 out of the 43824 hourly data sets (7.4%) contained missing values due to radar outage. These fields were omitted from further analyses. Among the remaining fields, 14251 (35.1%) showed zero rainfall, while the rest (64.9%) contained at least one grid point with rainfall. The mean, minimum and maximum grid rainfall values in the data set are 0.08 mmh^{-1} , 0 mmh^{-1} and 205 mmh^{-1} , respectively. Spatial plots of RR for selected times can be found in Fig. 8 and Fig. 9.

2.3 Experimental setup

To address research question RQ1, we analysed each dimension reduction method in terms of training effort (if any), preservation of the mean, variance and range of the original observation field, and additionally assess whether spatial subsets of the field can be directly reconstructed from the compressed representation (see Sect. 3.4).

To address research question RQ2, we set up dimension reduction experiments as follows. Firstly, parameters of the PCA and AE methods were learned from the 3 years of training data; the BA and RDP algorithms did not use data from the training period. Where necessary, parameters and hyperparameters for each method were also chosen or identified during training. We then analysed the performance of each method for a wide range of compression levels k , to explore the sensitivity of the



reconstruction error to the level of compression. The compression level k is defined by the number of distinguishable objects in the compressed representation, and it may range from 1 (strongest compression) to 100,000 (no compression). With a focus on analysing the reconstruction error under strong compression, we used the set $k = \{4, 8, 20, 40, 80, 160, 250, 500, 1000, 1500, 2000, 2500\}$, i.e. the number of objects in the compressed representation ranges from 0.4% to 2.5% of the original data size. The BA algorithm is an exception: here, averaging blocks had to be identified subject to the three constraints that for each compression level, all blocks should i) be equal in size and shape, ii) be as close to a square in shape as possible, and iii) together completely cover the gridded domain. This was fulfilled for compression levels $k = \{1, 8, 40, 160, 1000, 4000\}$, with the following block sizes [km²], extents in N-S direction [km] (i.e. N_i), and extents in E-W direction [km] (i.e. N_j),

265 $\{(100000, 400, 250), (12500, 100, 125), (2500, 50, 50), (625, 25, 25), (100, 10, 10), (25, 5, 5)\}$.

For PCA, each gridded field is first reshaped into a vector of length $N_g = 100,000$, representing one sample of the N_g features. The parameters of the PCA transformation, including the mean vector and the eigenvectors of the covariance matrix, are then learned based on all $N_{t,train}$ training samples. Let $z_{o,train} \in \mathbb{R}^{N_{t,train} \times N_g}$ denote the observation fields in the training data after reshaping, and $\mu_{train} \in \mathbb{R}^{N_g}$ denote the mean vector over all $N_{t,train}$ training samples. The principal components are then

275 computed as the eigenvectors of the covariance matrix $\frac{1}{N_{t,train}-1} (z_{o,train} - \mu_{train})^T (z_{o,train} - \mu_{train})$. The compressed representation of the $N_{t,test}$ training samples $z_{o,test} \in \mathbb{R}^{N_{t,test} \times N_g}$ with a compression level k is obtained by retaining the top k principal components, $z_c = W_k^T (z_{o,test} - \mu_{train})$, where $W_k^T \in \mathbb{R}^{N_g \times k}$ contains the first k eigenvectors. The reconstruction z_r is then generated by the inverse transform, $z_r = W_k^T z_c + \mu_{train}$.

For RDP, the algorithm was started with a minimal set of four vertices (here grid points) placed at the four corners of the

280 domain. This ensures that triangulation is done only by interpolation, and extrapolation is avoided.

For the AE, we used a ConvNeXt-based AE network architecture designed to accommodate the range of compression levels while maintaining computational efficiency. The encoder uses three stages of patch downsampling with factors of 5×5 , 5×5 and 2×2 , reducing the original 400×250 grid to 8×5 feature maps before flattening to the bottleneck. The number of initial feature channels was adjusted based on the bottleneck size: smaller bottlenecks (4-80 parameters) used 1 channel,

285 medium bottlenecks (160-500) used 2-7 channels, and larger bottlenecks (1000-2500) used 13-32 channels. This design ensures that the network capacity scales appropriately with the bottleneck size: Smaller bottleneck sizes require more aggressive compression, which translates into smaller latent spaces, which we achieve by having smaller networks. After the second 5×5 downsampling stage, the number of channels is doubled to increase the network's capacity as spatial resolution decreases. The overall architecture is shown in Fig. 2. Throughout the network, we used the SiLU (Sigmoid Linear Unit)

290 activation function (Hendrycks and Gimpel, 2016). Data preprocessing included standardization to improve training stability. For temperature data (T2), we applied z-score normalization using the training set statistics. For rainfall data (RR), we first applied a logarithmic transformation $\log(RR + 1)$ to handle the skewed distribution, followed by z-score normalization. The network was trained to predict standardized outputs, which were then transformed back to original units during reconstruction. Training was performed using the AdamW optimizer (Loshchilov, 2017) with a learning rate of 0.002, weight decay of 0.02,

295 and batch size of 256. To optimize hyperparameters, we first trained models on 2019-2020 data and validated on 2021 data to prevent overfitting and select the best configuration. After determining the optimal hyperparameters, we retrained each model on the complete 2019-2021 training dataset for 500 epochs to produce the final models used for testing. Training times ranged from 1.5 to 4 hours on an A100 GPU, depending on bottleneck size and associated network capacity. The number of trainable parameters varied with bottleneck size, from 2700 parameters for the smallest compression level to 13.7 million parameters

300 for the largest.

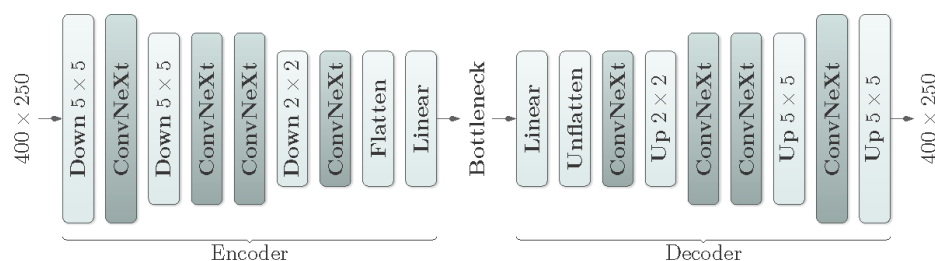


Figure 2. Architecture of the Autoencoder used in this study.

305 After training (if any), we applied all dimension reduction methods at all compression levels to the T2 and RR test data sets. Each application consisted of compressing the original field, and reconstructing it from the compressed representation. The information loss caused by the compression was then measured by a grid-by-grid comparison of the original and reconstructed fields using the performance measures described in the following section. The corresponding results are presented in Sect. 3.2 and Sect. 3.3.

310 2.4 Performance evaluation

We evaluated the performance of each compression method by visually comparing original and reconstructed fields at selected time steps, and by calculating reconstruction error aggregated across all time steps in the test period.

2.4.1 Visual comparison

315 While simple, a visual comparison allows for a descriptive (albeit subjective) assessment of the overall agreement of two fields, and can reveal patterns of disagreement that could remain unnoticed in aggregated error measures. For visual comparison, we selected pairs of fields from the test data with similar mean field values, but different field variabilities, expressed by the standard deviation of the field. This choice reflects the expectation that compression of highly variable fields is more challenging than that of smooth fields, and that the strengths and weaknesses of each compression method are best revealed by a comparison in the different cases. A scatterplot of the mean and standard deviation of each field in the test data set is shown in Fig. 3. The selected fields for both T2 and RR are indicated by red triangles. For T2 (subplot a), the fields have almost the same mean value of 19.6 °C, but very different standard deviations: 0.99 °C for 17 June 2022 8:00 and 6.11 °C for 20 June 2022 12:00. Spatial plots of the T2 fields are shown in Fig. 5 and Fig 6 in Sect. 3.2.

325 For RR (subplot b), the fields of 1 August 2023 3:50 and 16 August 2023 21:50 also have very similar mean values (1.60 mm/h and 1.58 mm/h, respectively), but very different standard deviations (1.41 mm h⁻¹ and 4.24 mm h⁻¹, respectively). Spatial plots of the RR fields are shown in Fig. 8 and Fig. 9 in Sect. 3.3. Please note that the timestamp at minute 50 is inherited from the cutoff times of the radar data.

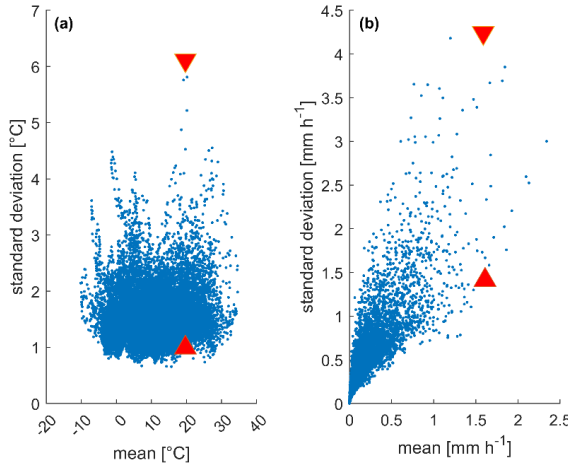


Figure 3. Scatterplot of standard deviation vs. mean value of each field in the testing period 2022–2023. Subplot (a) shows air temperature data T2, subplot (b) shows rainfall data RR. Red triangles indicate fields selected for visual display by similar mean and dissimilar standard deviation values. In subplot (a), this is 20 June 2022 12:00 (upper triangle) and 17 June 2022 8:00 (lower triangle). In subplot (b), this is 16 August 2023 21:50 (upper triangle) and 1 August 2023 3:50 (lower triangle).

2.4.2 Error measures

Separately for each variable, dimension reduction method, compression level, and time step, we quantified the (dis-)agreement of the original and reconstructed field with three well-established error measures: the Mean Absolute Error (MAE), Mean Squared Error (MSE), and Multiscale Structural Similarity Index (MSSIM). The MAE and MSE assess the reconstruction at each grid point separately, and are therefore insensitive to errors in the spatial structure of the fields. In contrast, the MSSIM is designed to compare large-scale aspects of the fields, and it is therefore capable of assessing misrepresentation of the dependencies between different grid points.

The MAE is the field average of all absolute differences of the original and corresponding reconstructed grid values,

$$\text{MAE} = \frac{1}{N_g} \sum_{g=1}^{N_g} |z_o(x_g, y_g) - z_r(x_g, y_g)|. \quad (2)$$

The MAE range is $[0, \infty]$, with small values preferred.

The MSE is the field average of all squared differences of the original and corresponding reconstructed grid values,

$$\text{MSE} = \frac{1}{N_g} \sum_{g=1}^{N_g} (z_o(x_g, y_g) - z_r(x_g, y_g))^2. \quad (3)$$

The MSE range is $[0, \infty]$, with small values preferred.

The SSIM was proposed by Wang et al. (2002) to measure the agreement between an original and a distorted image as a combination of three different features of the images: loss of correlation, luminance distortion, and contrast distortion. Following the notation of Wang et al. (2002), we define the SSIM as

$$\text{SSIM} = \frac{\sigma_{or}}{\sigma_o \sigma_r} \cdot \frac{2\bar{o}\bar{r}}{(\bar{o})^2 + (\bar{r})^2} \cdot \frac{2\sigma_o \sigma_r}{\sigma_o^2 + \sigma_r^2}, \quad (4)$$

where $\bar{o}$ and $\bar{r}$ are the mean values of the original and distorted (here: reconstructed) fields, respectively, σ_o and σ_r are the corresponding standard deviations, and σ_{or} is the covariance of the original and reconstructed fields. The three components of the SSIM provide measures of the loss of correlation, luminance distortion, and contrast distortion, respectively, facilitating an analysis of where the errors in the reconstructed field arise. The SSIM range is $[-1, 1]$, where large positive values are



preferred. The SSIM was extended to a multiscale variant, MSSIM, by Wang et al. (2003), which is a weighted combination of SSIMs calculated for several levels of increasingly coarse-grained versions of the original and distorted images. The multiscale evaluation better approximates human perceptual assessment of the agreement between two images. To calculate the MSSIM, we used the function “multissim” implemented in Matlab 2024b with parameter values NumScales = 7 (default is 5), ScaleWeights = fspecial(“gaussian”, [1, numScales], 1) (default), Sigma = 1.5 (default), DynamicRange = 64 for T2 and 206 for RR. “NumScales” sets the number of scales used to calculate the MSSIM, “ScaleWeights” sets weights for each scale when calculating the overall error, “Sigma” sets the standard deviation of a gaussian function for weighting neighbourhood pixels around a pixel for estimating local statistics. DynamicRange is the range of possible values a variable can take, and we calculated it as the difference between the smallest and largest value of all time steps and fields, separately for T2 and RR. We tested several parameter combinations and found that the relative performance ranking of the dimension reduction methods was robust to these choices. We therefore kept the default values, and only increased NumScales to include low-resolution (64-by 64 km blocks for Scale 7) comparison of large-scale features.

3 Results and Discussion

3.1 Correlation of error measures

After having calculated the reconstruction error using the MAE, MSE and MSSIM for all variables, dimension reduction methods, and compression levels, we first analysed the similarities of the error measures by computing the pairwise correlation coefficients of the MAE with the MSE and MSSIM, respectively. For this, we used the negative value of the MSSIM to ensure that a positive dependency is indicated by a positive correlation coefficient. The results are shown in Table 1. For both variables, the correlations were consistently high across all dimension reduction methods and all compression levels. We therefore restrict the discussion of all further results in the main text to the MAE to avoid redundancies. We chose the MAE as it offers the straightforward interpretation of a natural distance on an interpretable scale. Results for MSE and MSSIM are shown in appendix A.

Table 1. Pairwise correlation coefficients [-] of reconstruction error in the test data measured by MAE (Mean Absolute Error), MSE (Mean Squared Error) and MSSIM (Multiscale Structural Similarity Index), for air temperature (T2) and rainfall (RR). The error was calculated for all compression levels: the values in the table are the average, minimum and maximum correlation coefficients over all compression levels.

	MAE, MSE		MAE, MSSIM	
	T2	RR	T2	RR
Block Averaging	0.96 [0.93, 0.97]	0.74 [0.68, 0.81]	0.93 [0.61, 0.99]	0.82 [0.76, 0.85]
Principal Component Analysis	0.97 [0.96, 0.97]	0.76 [0.74, 0.80]	0.94 [0.80, 0.97]	0.83 [0.80, 0.86]
Ramer-Douglas-Peucker	0.97 [0.95, 0.97]	0.79 [0.71, 0.86]	0.87 [0.44, 0.97]	0.83 [0.68, 0.88]
Autoencoder	0.97 [0.97, 0.98]	0.68 [0.64, 0.74]	0.95 [0.92, 0.98]	0.77 [0.72, 0.81]

3.2 Air temperature

For T2 and MAE, Fig. 4 shows the reconstruction error for all dimension reduction methods and compression levels up to 2500, i.e. up to 2.5 % of the original field size (see Fig. A01 for MSE and Fig. A02 for MSSIM). As expected, the reconstruction error generally decreases with increasing compression level, and is already fairly small at level 2500. This holds for all compression methods. All methods show an exponentially decreasing relationship between compression level and error:



starting at high compression, the reconstruction error quickly drops and then saturates with increasing compression level. The transition between the two regimes occurs at a comparatively low compression level for PCA (approx. level 200), and has been reached for all methods at compression level 1000. The latter is therefore a good compromise between error and degree of compression, and we use it for the display of selected spatial fields in Fig. 5 and Fig. 6. Across compression methods, PCA clearly results in the lowest reconstruction error at all compression levels, followed by the AE, BA and RDP. This suggests that fields exhibiting smooth spatial patterns and approximately Gaussian distributions, such as T2, are particularly suitable for PCA-based compression, since a small number of principal components can capture a large fraction of the total variability. Interestingly, the sophisticated recursive search algorithm of RDP to identify landmark points in fields does not pay off for compressing T2 fields, precisely because in generally smooth fields there are no distinct landmark points to select, and thus even a simple algorithm such as BA performs better than RDP.

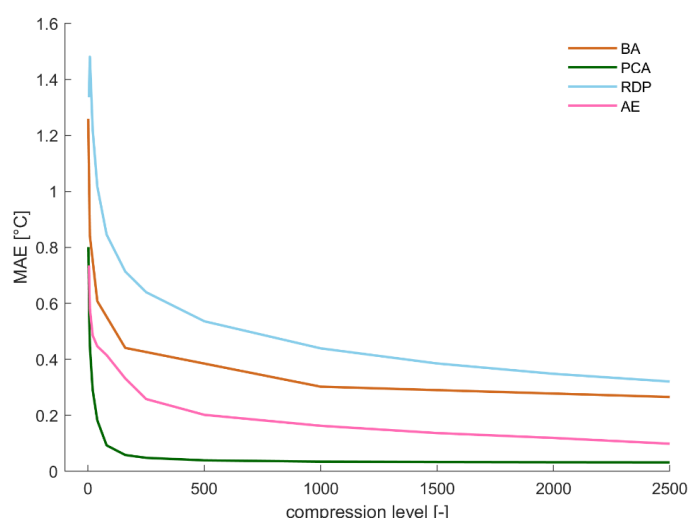


Figure 4. Mean Absolute Error (MAE) of the reconstructed field vs. compression level for air temperature T2 in the test period 2022–2023.

The compression level equals the number of objects in the compressed representation. Errors are shown for Block Averaging (BA), Principal Component Analysis (PCA), the Ramer-Douglas-Peucker algorithm (RDP), and an Autoencoder (AE).

In Fig. 5 and Fig. 6, exemplary observed spatial fields of T2 are shown together with their reconstructions using all dimension reduction methods at compression level 1000. Both fields have similar mean values, but differ in spatial variance: variance is low in Fig. 5, and high in Fig. 6 (see Fig. 3 and Table B1). The spatial temperature distribution of the observed field in Fig. 5 clearly reflects local topography such as river valleys and mountainous areas (see also Fig. 1), and PCA and the AE almost perfectly reproduce these patterns. BA shows the expected “blocky” appearance due to averaging (here for 10-by-10 km blocks), but correctly reproduces the large-scale features of the temperature distribution. Spurious artefacts appear in the RDP reconstruction. This is mainly the case for areas with low temperature gradients, as only few landmark points were selected there to support interpolation. Overall, the reconstructed PCA and AE fields appear to most closely resemble the original observed field, followed by the BA and RDP algorithms (this is supported by checks of the summary statistics in Table B1). In Fig. 6, a large-scale temperature gradient in the N-S direction due to frontal advection is present, plus some minor local patterns reflecting topography. As before, PCA and AE reproduce both the large- and small-scale patterns very well, while BA appears blocky but reproduces the large-scale pattern. RDP similarly captures the large-scale pattern, but local artefacts again appear due to insufficient support by interpolation points. Overall, PCA again performs best, followed by AE, BA and RDP (see Table B1).

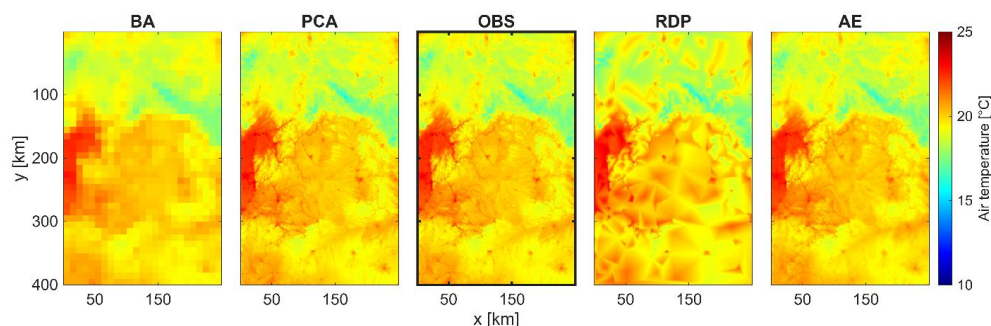


Figure 5. Gridded temperature T2 fields on 17 June 2022 8:00, a time point with comparatively low spatial variance. OBS is the observed field, BA, PCA, RDP and AE are fields reconstructed at compression level 1000, by Block Averaging, Principal Component Analysis, the Ramer-Douglas-Peucker algorithm and an Autoencoder.

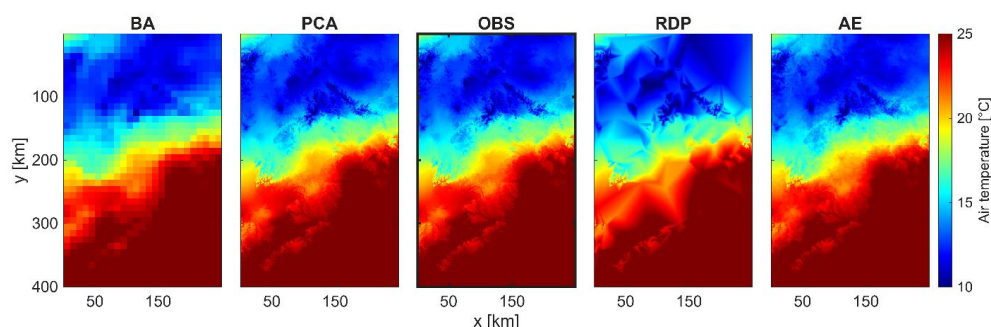


Figure 6. As in Fig. 5, but for 20 June 2022 12:00, a time point with comparatively high spatial variance.

3.3 Rainfall

As for T2 in the previous section, the reconstruction error measured by MAE of the RR fields is shown as a function of compression level in Fig. 7 (see Fig. A03 for MSE and Fig. A04 for MSSIM). When calculating the mean error over all time steps in the testing period, cases of zero total rainfall in the observed field were ignored. While this could implicitly favour methods that overpredict rainfall occurrence (Lerch et al., 2017), this is not an issue here: both BA and RDP correctly reconstruct zero rainfall fields by design, while PCA and AE reconstruct fields with very small but non-zero values (related field MAE is $8.6 \cdot 10^{-4} \text{ mm h}^{-1}$ for PCA, and $3.6 \cdot 10^{-4} \text{ mm h}^{-1}$ for AE). Fig. 7 reveals the same general relationship between compression level and error for RR as for T2: exponentially decreasing errors with increasing compression level, and a saturation of the error reduction beyond compression level 1000. However, differences appear in the ranking of the compression methods: for RR, the AE yields the lowest reconstruction error up to compression level 1000, beyond which the RDP algorithm performs best. The slightly increasing error of AE for compression levels beyond 1000 is likely due to overfitting and could possibly be eliminated by adjusting the size of the AE network architecture (see Sect. 2.2.4 and Sect. 2.3) or by introducing additional regularization techniques such as dropout. BA and PCA are ranked third and fourth place, respectively, but with little difference between them. The spatially intermittent nature of rainfall fields can apparently be captured well by the AE and RDP algorithms, and less so by BA and PCA.

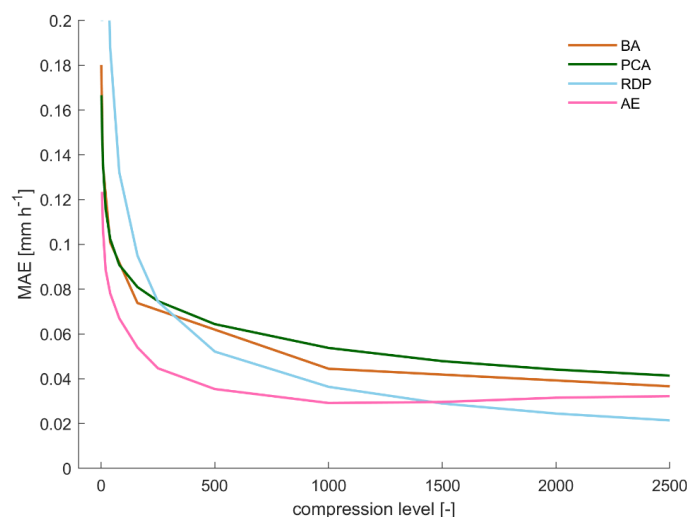


Figure 7. Mean Absolute Error (MAE) of reconstructed fields vs. compression level for rainfall RR in the test period 2022-2023, excluding all time steps where the observed field shows zero rainfall. The compression level equals the number of objects in the compressed representation. Errors are shown for Block Averaging (BA), Principal Component Analysis (PCA), the Ramer-Douglas-Peucker algorithm (RDP), and an Autoencoder (AE).

Spatial RR fields at two selected times are shown in Fig. 8 and Fig. 9. As for T2, both have similar mean values but differ in spatial variance (low for Fig. 8, high for Fig. 9, see Fig. 3 and Table B2). The observed rainfall in Fig. 8 shows widespread low-intensity rainfall plus some regions with higher-intensity rainfall cells. PCA and the AE reproduce both the large-scale features and the rainfall cells well; only some small-scale features of the high-intensity rainfall fields are lost. BA shows the expected blocky appearance, removing any spatial patterns at resolutions below the block size, but the large-scale rainfall pattern is preserved. For RDP, two effects stand out: firstly, in regions of low rainfall intensity and small rainfall gradients, artefacts appear due to insufficient local interpolation support; secondly, the spatial patterns of the high-intensity rainfall cells with pronounced spatial rainfall gradients are very well reproduced, as the majority of support points for interpolation is placed there. This characteristic of RDP is yet more present in Fig. 9. Here, a local region of high-intensity rainfall is present, and almost perfectly reproduced by RDP. AE also correctly identifies regions of zero rainfall, and reproduces the main features of the rainfall cell. For PCA, low-intensity artefacts appear in the dry region, while the main rainfall cell is reproduced in high detail. A possible explanation is that PCA reconstructs fields relative to the mean field estimated from the training data and uses only a truncated set of principal components. Consequently, sparse precipitation structures may not be accurately represented, which can lead to weak residual precipitation signals in regions that are dry in the original field. BA reproduces the zero rainfall regions and the spatial extent of the rainfall cells, but fails to reproduce the high-intensity regions within the rainfall cell due to the block-averaging. In terms of overall reconstruction error, the ranking of the compression methods for Fig. 8 is AE, PCA, BA, RDP, while it is AE, RDP, BA, PCA for Fig. 9 (always from best to worst). A comparison of the fields' summary statistics is available in Table B2.

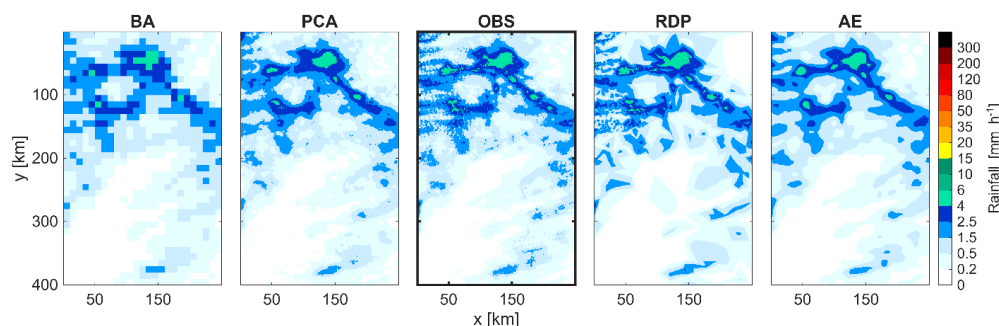


Figure 8. Gridded rainfall RR fields on 1 August 2023 3:50, a time point with comparatively low variance. OBS is the observed field, BA, PCA, RDP and AE are fields reconstructed at compression level 1000, by Block Averaging, Principal Component Analysis, the Ramer-Douglas-Peucker algorithm, and an Autoencoder.

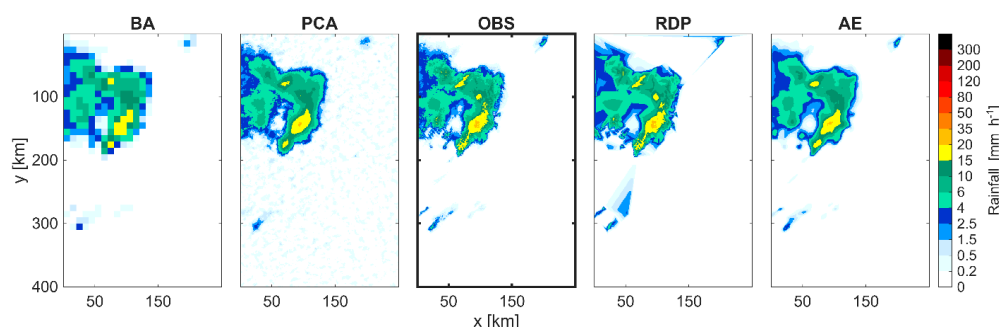


Figure 9. As in Fig. 8, but for 16 August 2023 21:50, a time point with comparatively high variance.

3.4 General properties of the dimension reduction methods

We now discuss the general properties of the dimension reduction methods, summarized in Table 2.

BA can be directly applied to a field without requiring training, and it is computationally cheap. Due to the averaging effect of *BA*, it preserves the field mean by design, but it will inevitably underestimate the field variability and data range. These effects are illustrated in the first column of Fig. 10 and Fig. 11 for the test data at compression level 1000. As each block is associated with a fixed spatial location, spatial subregions can be directly extracted from the compressed representation, though only at the level of the block resolution.

PCA is computationally very efficient in application, but the training can be computationally demanding for high-dimensional data, since it requires computing a large covariance matrix and performing eigenvalue decompositions. While *PCA* preserves the field mean by design, data with significant mean shifts compared to the training data set may compromise this property. The fraction of the standard deviation retained equals the square root of the preserved variance ratio, which depends on the compression level k , while data ranges as well as extreme values are dampened due to truncation in the principal component space. This effect is particularly visible for the rainfall data in Fig. 11 (second column). Notably, *PCA* enables location-aware partial reconstruction, in which a subset of data (e.g., individual grid points) can be reconstructed directly from compressed representations without reconstructing the entire field, leveraging feature-specific eigenvector components based on linear



weighting across principal components. However, PCA's linearity means it cannot capture nonlinear structures, which limits its ability to reconstruct complex spatial patterns such as rainfall fields.

RDP does not require training, but due to its recursive search for landmark points, it is computationally expensive in application. It does not guarantee preservation of field-scale statistics by design, but it appears to do so in practice for both T2 and RR fields (see the third column of Fig. 10 and Fig. 11). A particularly valuable property of RDP is that it preserves the data range, and hence the presence of extreme values, which distinguishes it from the other compression methods in this study.

The AE also exhibits several properties that distinguishes it from the other compression methods. Unlike BA and RDP, AE requires extensive and computationally expensive training on representative data. However, this training process allows the AE to learn complex, non-linear patterns specific to the data type being compressed. Once an AE is trained, its application is computationally inexpensive. Regarding preservation of statistical properties, AEs do not inherently preserve the field mean, standard deviation, or data range. The degree to which these properties are maintained depends on the bottleneck size and how well the model has learned the data distribution during training. It is particularly challenging for AEs to preserve the data range, since the reconstruction process through the decoder can even produce values outside of the original data range, especially for extreme values that were underrepresented in the training data. For the test data, this effect is visible in the fourth column and third row of Fig. 10 and Fig. 11, where dots above the 1:1 line are visible. A notable limitation of the AE approach is that partial reconstruction of field subsets is not directly possible from the compressed representation. The bottleneck contains a global, distributed encoding of the entire field, meaning that to extract information about any spatial subregion, the complete field must first be reconstructed through the decoder. While alternative AE architectures could potentially maintain spatial locality in the bottleneck, this would likely compromise compression efficiency.

Table 2. Properties of the dimension reduction methods. BA is Block Averaging, PCA is Principal Component Analysis, RDP is the Ramer-Douglas-Peucker algorithm, AE is an Autoencoder. For preservation of field-scale statistics, bold letters indicate if they are guaranteed ("Yes") or violated ("No") by design.

Method	Training required	Computational effort for		Preservation of field-scale statistics			Partial extraction from compressed representation possible
		Training	Application	Mean	Standard deviation	Data range	
BA	No	--	Low	Yes	No	No	Yes
PCA	Yes	Medium	Low	Yes^a	No	No	Yes
RDP	No	--	High	Yes	Yes	Yes	Yes
AE	Yes	High	Low	Yes	Yes	No ^b	No

^a: Only guaranteed if global mean of training and application data match.

^b: AE can under- and overestimate the data range, BA and PCA only underestimate.

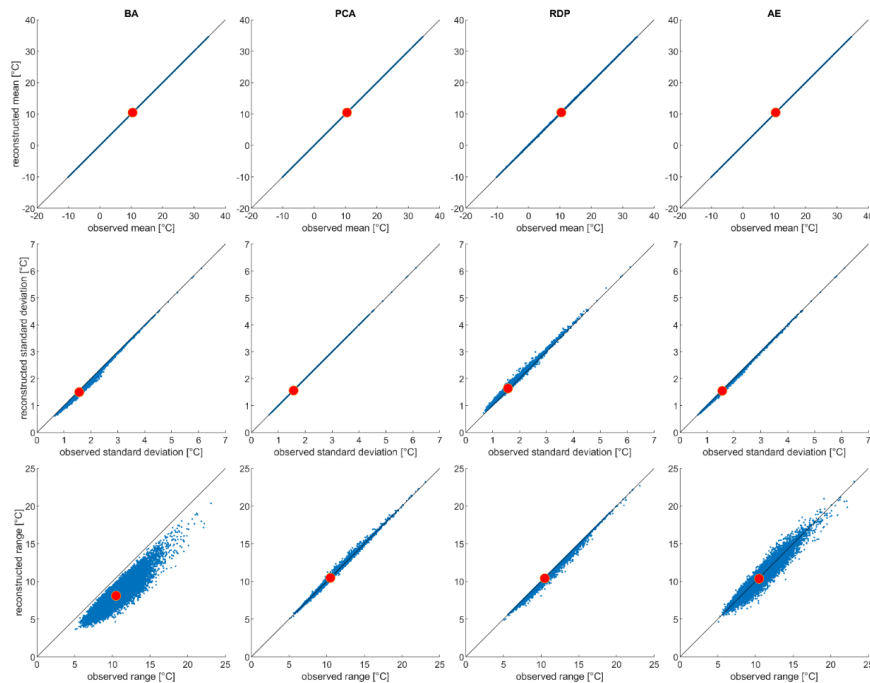


Figure 10. Scatterplots of reconstructed vs. observed temperature T2 field statistics in the test period 2022-2023 (blue dots). Top row is for the field mean, middle row is for standard deviation, bottom row is for the range. Columns BA, PCA, RDP and AE indicate fields reconstructed from Block Averaging, Principal Component Analysis, the Ramer-Douglas-Peucker algorithm and an Autoencoder, respectively, all at compression level 1000. The red dots indicate the mean of all field statistics. The black 1:1 line indicates perfect agreement between the observation and reconstruction; dots below the line indicate underestimation by the reconstruction, dots above overestimation.

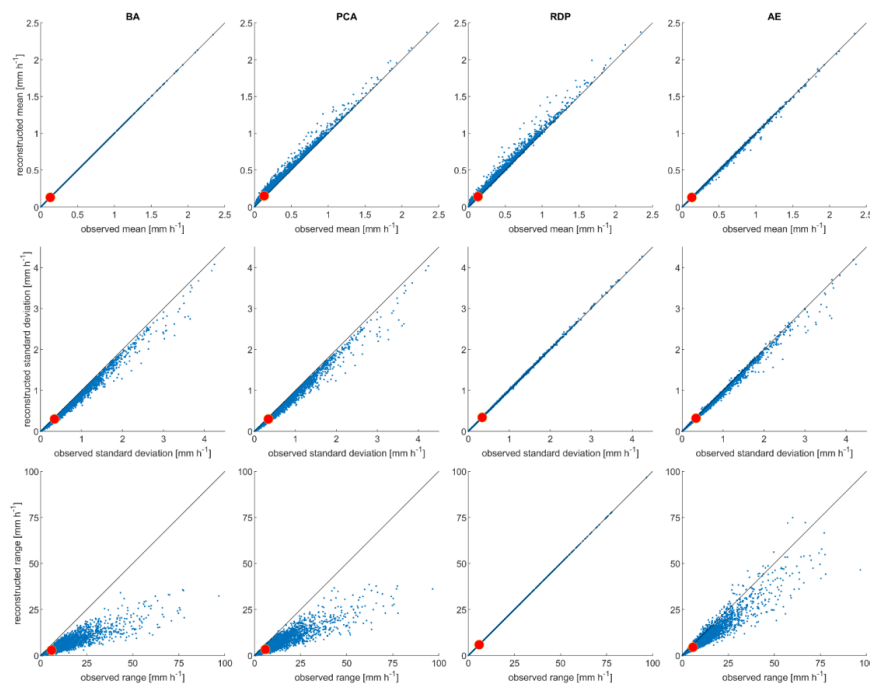


Figure 11. As in Fig. 10, but for rainfall fields RR.



515 4 Conclusions and Outlook

The continuous growth of file sizes in the geosciences demands efficient compression algorithms to facilitate workflows of data transfer, storage, and use in downstream applications. The recent rise of data-driven modelling approaches has relaxed the constraints on data use in the sense that the number and units of the data are no longer rigidly prescribed, as is the case in traditional physically based models, but can be selected to maximize the information content of the model input. This makes it feasible to use compressed representations of the data as model input. In spatial applications, subsets of the data corresponding to regions of interest can often be extracted directly from the compressed data. Therefore, the two research questions we addressed in this study were how different lossy compression algorithms compare in terms of general properties, such as computational effort and preservation of location under compression (RQ 1), and how they compare in terms of compression efficiency and reconstruction error (RQ 2). We focused on dimension reduction methods, because the resulting low-dimensional representations require fewer input channels when used in downstream data-driven models, which reduces computational efforts and the risk of overfitting. We studied RQ 1 and RQ 2 in an application to five years of hourly air temperature and rainfall fields on 2-d grids with a 1 km resolution across a 100,000 km² domain in central Germany. For compression, we tested Block Averaging (BA), Principal Component Analysis (PCA), the Ramer-Douglas-Peucker algorithm (RDP), and an Autoencoder (AE).

The main insights related to RQ 1 are: The methods differ substantially in terms of effort in training (PCA and AE require training, BA and RDP do not) and application (high for RDP, low for the others), preservation of field-scale statistics (BA and PCA preserve the mean by design, RDP and AE only empirically; RDP and AE preserve the standard deviation, BA and PCA reduce it; the data range is preserved by RDP, reduced by BA and PCA, and either reduced or inflated by AE), and partially in their ability to extract spatial subsets from the compressed representation (not possible for our AE, possible for all others). In conclusion, the choice of the best method should be made specifically for the task at hand.

Regarding RQ 2, the main results are: All test data were highly compressible. Even at a compression to 1% of the original number of grid cells, the reconstruction error was very low. For all methods, the error declined exponentially as a function of the compression level, with an initially fast decline followed by saturation at approximately 1% compression. For the spatially smooth temperature fields, PCA yielded the lowest reconstruction error at all compression levels, followed by AE, BA and RDP. For the spatially heterogeneous rainfall fields, AE and RDP achieved the best reconstruction performance. As for RQ 1, the conclusion is that the optimal choice of the compression method depends on the data.

In this study, we used dimension reduction methods for (lossy) compression, and measured compression by the reduction of the number of objects from the original to the compressed representation of the data, rather than by the decrease in file size, because fewer objects lend themselves better as input to the typically limited number of input channels of data-driven models.

But, of course, reducing the number of objects typically results in a 1:1 reduction of the storage size, so the compression methods proposed here can also be applied for the purpose of storage size reduction. Moreover, all of the methods used can be combined with other lossless or lossy methods if the objective is storage size reduction.

The four dimension reduction methods that we compared were chosen not only because they are the methods that are most widely used in practice, but also because they represent fundamentally different approaches to dimensionality reduction. This allowed us to relate the reconstruction errors of the methods to how the methods compress the data. For example, the RDP algorithm is well-suited to capturing “objects” in the fields, meaning it performs well when applied to heterogeneous rainfall fields, but relatively poorly when applied to smoother temperature fields. In contrast, due to the truncation of higher-order components, PCA reconstructions typically exhibit a smoothing effect, which makes the method particularly suitable for more homogeneous fields. Nonetheless, other compression methods have been proposed and future work could extend or adapt the comparison to include these methods. For example, kernel PCA (Schölkopf et al., 1997) is a non-linear extension of PCA that is similarly straightforward to implement in practice, while t-SNE (Maatens and Hinton, 2008) and UMAP (McInnes et al., 2018) present popular dimension reduction frameworks that are particularly adept at visualising high-dimensional data.



Alternative autoencoder architectures could also be studied. Kingma and Welling (2013) introduced variational autoencoders, which use generative models to not only compress and reconstruct data, but also to generate new samples. Shen and Meinshausen (2024) similarly introduce a probabilistic framework for dimension reduction based on generative models, and Chen et al., (2025) propose an approach to reduce the dimension of ensemble forecasts. These probabilistic approaches are not directly comparable to the methods considered herein, but could similarly be studied further in future work.

While we focused here on 2-d fields of air temperature and rainfall, further research could extend this study to other variables such as wind speed, 3-d spatio-temporal fields, or to achieve joint compression of multivariate data sets. As immediate next steps, we plan to further improve the PCA method by special treatment of zero rainfall through pre-transformation or exclusion, and to improve the AE method by adjusting the network architecture to mitigate the overfitting observed at high compression levels. More broadly, this study represents a step towards directly using data that were compressed by dimension reduction in downstream applications. Having characterized the properties and performance of different dimension reduction methods, the next step is to use the compressed data as input to data-driven models. If it is already known what the compressed data will be used for, then it additionally becomes possible to use the dimension reduction methods to directly optimize the downstream model performance, rather than employing a two-step procedure that first compresses the data, and then fits a model to the compressed data. We plan to test these approaches using large (Germany-wide) compressed meteorological fields as input to local (catchment-scale) data-based hydrological models.

Code and data availability. The current version of all code related to this study is available from <https://doi.org/10.35097/3en5z1293bn3fkh2> (Ehret et al., 2026b) under the licence CC BY-SA 4.0. All data related to this study are available from <https://doi.org/10.35097/cbmr0yj6qb06fc1n> (Ehret et al., 2026a) under the licence CC BY-SA 4.0.

Author contributions. Original idea for the study by UE. Design of experiments by all authors. Coding and computing: UE, JC, FS. Analysis of results by UE. Discussion of results and writing by all authors. Funding was acquired by UE and SL.

Competing interests. The contact author has declared that none of the authors has any competing interests

Acknowledgements. JC, SA, SL and FS gratefully acknowledge support by the Vector Stiftung through the Young Investigator Group "Artificial Intelligence for Probabilistic Weather Forecasting", and the project "Zukunftsfähige Modellierung für die Geowissenschaften". JC gratefully acknowledges support within project T4 "Development of a deep learning prototype for operational probabilistic wind gust forecasting" of the Transregional Collaborative Research Center SFB/TRR 165 "Waves to Weather" funded by the German Research Foundation (DFG). All authors acknowledge the support of the state of Baden-Württemberg through bwHPC. UE would like to thank the people of Baden-Württemberg, who, through their taxes, provided the basis for this research.

Financial support. The article processing charges for this open-access publication were covered by the Karlsruhe Institute of Technology (KIT) publishing fund.



595 Appendix A: Reconstruction errors MSE and MSSIM for air temperature and rainfall

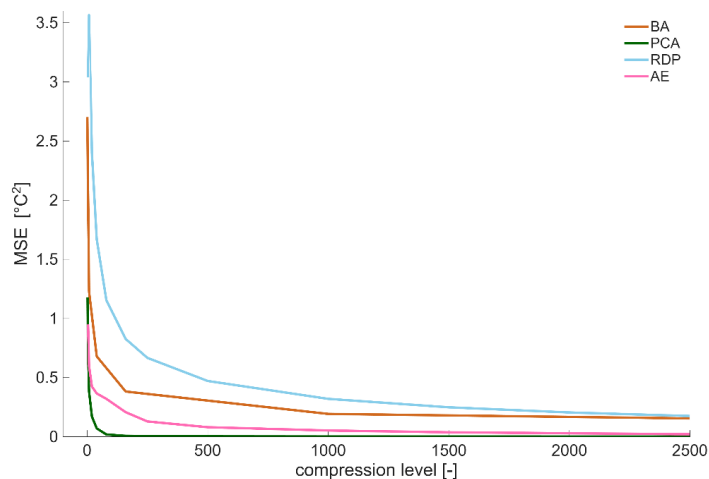


Figure A01. Mean Squared Error (MSE) of the reconstructed field vs. compression level for air temperature T2 in the test period 2022-2023. The compression level equals the number of objects in the compressed representation. Errors are shown for Block Averaging (BA), Principal Component Analysis (PCA), the Ramer-Douglas-Peucker algorithm (RDP), and an Autoencoder (AE).

600

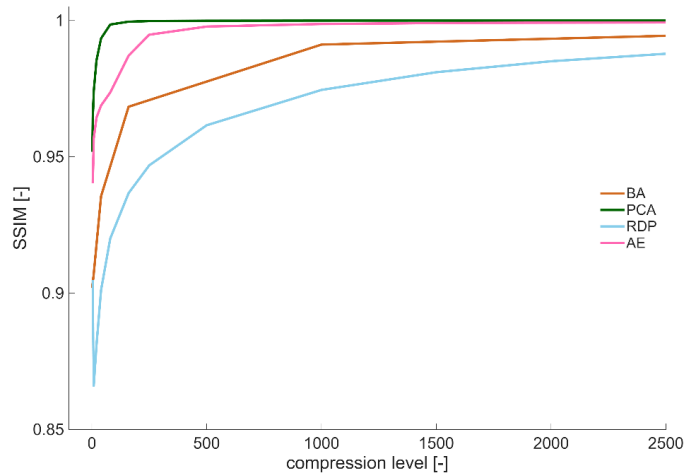
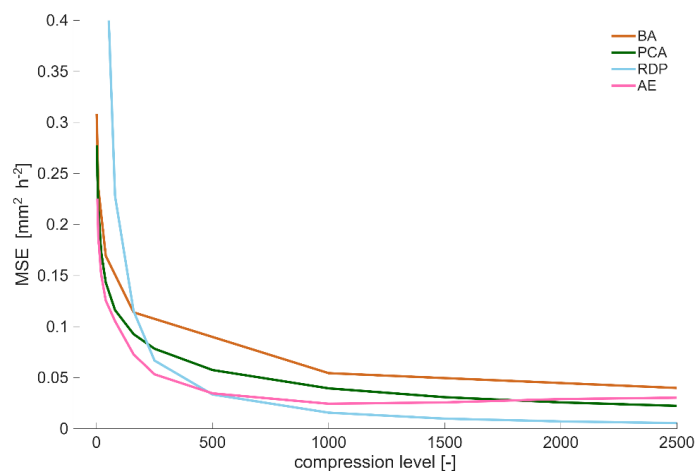


Figure A02. As in Fig. A01, but for Multiscale Structural Similarity Index (MSSIM).



605 **Figure A03.** As in Fig. A01, but for rainfall RR.

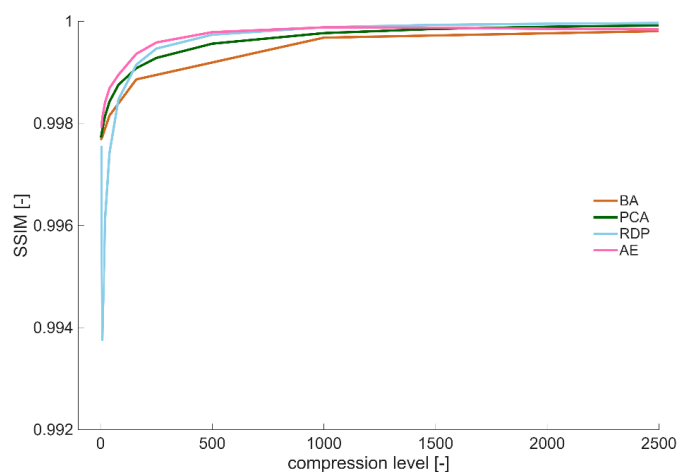


Figure A04. As in Fig. A01, but for Multiscale Structural Similarity Index (MSSIM) and rainfall RR.

Appendix B: Field statistics for selected time steps

610 **Table B1.** Statistics of air temperature fields T2 at two time steps: mean and standard deviation of the observed fields, and reconstruction error at compression level 1000 of Block Averaging (BA), Principal Component Analysis (PCA), the Ramer-Douglas-Peucker (RDP) algorithm, and an Autoencoder (AE). The smallest error is printed bold.

Date/Time (UTC)	Observed field		Mean Absolute Error MAE [°C]			
	Mean [°C]	Standard deviation [°C]	BA	PCA	RDP	AE
17 June 2022 8:00	19.68	0.99	0.24	0.03	0.32	0.11
20 June 2022 12:00	19.69	6.11	0.31	0.05	0.49	0.20

Table B2. As in Table B1, but for rainfall fields RR.



615

Date/Time (UTC)	Observed field		Mean Absolute Error MAE [mm h ⁻¹]			
	Mean [mm/h]	Standard deviation [mm/h]	BA	PCA	RDP	AE
1 August 2023 3:50	1.60	1.41	0.27	0.24	0.30	0.21
16 August 2023 21:50	1.58	4.24	0.36	0.45	0.26	0.24

References

- Acuña Espinoza, E., R. Loritz, M. Álvarez Chaves, N. Bäuerle, and U. Ehret: To bucket or not to bucket? Analyzing the performance and interpretability of hybrid hydrological models with dynamic parameterization, *Hydrol. Earth Syst. Sci.*, 28(12), 2705-2719, 2024.
- Baker, A. H., Hammerling, D. M., Mickelson, S. A., Xu, H., Stolpe, M. B., Naveau, P., Sanderson, B., Ebert-Uphoff, I., Samarasinghe, S., De Simone, F., Carbone, F., Gencarelli, C. N., Dennis, J. M., Kay, J. E., and Lindstrom, P.: Evaluating lossy data compression on climate simulation data within a large ensemble, *Geosci. Model Dev.*, 9, 4381-4403, 10.5194/gmd-9-4381-2016, 2016.
- Bauer, P., Thorpe, A., and Brunet, G.: The quiet revolution of numerical weather prediction, *Nature*, 525, 47–55, doi: 10.1038/nature14956, 2015.
- Bauer, P., Quintino, T., Wedi, N., Bonanni, A., Chrut, M., Deconinck, W., Diamantakis, M., Düben, P., English, S., Flemming, J., Gillies, P., Hadade, I., Hawkes, J., Hawkins, M., Iffrig, O., Kühnlein, C., Lange, M., Lean, P., Marsden, O., and Zanna, C.: The ECMWF Scalability Programme: Progress and Plans, 10.21957/gdit22ulm, 2020.
- Ben Bouallègue, Z., Clare, M. C. A., Magnusson, L., Gascón, E., Maier-Gerber, M., Janoušek, M., Rodwell, M., Pinault, F., Dramsch, J. S., Lang, S. T. K., Raoult, B., Rabier, F., Chevallier, M., Sandu, I., Dueben, P., Chantry, M., and Pappenberger, F.: The Rise of Data-Driven Weather Forecasting: A First Statistical Assessment of Machine Learning–Based Weather Forecasts in an Operational-Like Context, *Bull. Amer. Meteorol. Soc.*, 105, E864-E883, <https://doi.org/10.1175/BAMS-D-23-0162.1>, 2024.
- Bonneu, F., Makowski, D., Joly, J., and Allard, D.: Machine learning based on functional principal component analysis to quantify the effects of the main drivers of wheat yields, *European Journal of Agronomy*, 159, 127254, <https://doi.org/10.1016/j.eja.2024.127254>, 2024.
- Carreira-Perpinán, Miguel A.: A review of dimension reduction techniques, University of Sheffield Technical Report, CS-96-09 9, 1-69, 1997.
- Chen, J., Höhle, K., and Lerch, S.: Learning low-dimensional representations of ensemble forecast fields using autoencoder-based methods, *arXiv preprint arXiv:2502.04409*, 2025.
- Douglas, D., Peucker, T.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. In: *The Canadian Cartographer*. Bd. 10, Nr. 2, 1973, ISSN 0008-3127, S. 112–122, 1973.
- Duwe, K., Lüttgau, J., Mania, G., Squar, J., Fuchs, A., Kuhn, M., Betke, E., and Ludwig, T.: State of the art and future trends in data reduction for high-performance computing, *Supercomp. Front. Inn.*, 7, <https://doi.org/10.14529/jsfi200101>, 2020.
- DWD
https://opendata.dwd.de/climate_environment/CDC/grids_germany/hourly/hostrada/DESCRIPTION_gridsgermany_hourly_hostrada_en.pdf, 2024.
- Ehret, U., Chen, J., Scholz, F., Allen, S., and Lerch, S.: Data for lossy compression by dimension reduction for 2-d meteorological fields. Karlsruhe Institute of Technology. DOI: 10.35097/cbmr0yj6qb06fc1n, 2026a



- Ehret, U., Chen, J., Scholz, F., Allen, S., and Lerch, S.: Code for lossy compression by dimension reduction for 2-d meteorological fields. Karlsruhe Institute of Technology. DOI: 10.35097/3en5z1293bn3fkh2, 2026b.
- Fodor, I. K.: A survey of dimension reduction techniques, No. UCRL-ID-148494, Lawrence Livermore National Lab.(LLNL),
655 Livermore, CA (United States), 2002.
- García-Hinde, O., Terrén-Serrano, G., Hombrados-Herrera, M. Á., Gómez-Verdejo, V., Jiménez-Fernández, S., Casanova-Mateo, C., Sanz-Justo, J., Martínez-Ramón, M., and Salcedo-Sanz, S.: Evaluation of dimensionality reduction methods applied to numerical weather models for solar radiation forecasting, *Engineering Applications of Artificial Intelligence*, 69, 157-167, <https://doi.org/10.1016/j.engappai.2017.12.003>, 2018.
- 660 Gardner, Matt W., and Dorling, S. R.: Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences, *Atmospheric environment*, 32.14-15, 2627-2636, 1998.
- Goodfellow, I., Bengio, Y., and Courville, A.: *Deep Learning*, MIT Press, <http://www.deeplearningbook.org>, 2016.
- Graham, Mark, and Shelton, T.: Geography and the future of big data, big data and the future of geography, *Dialogues in Human geography*, 3.3, 255-261, doi:10.1177/2043820613513121, 2013.
- 665 Hendrycks, D., and Gimpel, K.: Gaussian Error Linear Units (GELUs). <http://arxiv.org/abs/1606.08415v5>, 2016.
- Hinton, G. E., and Salakhutdinov, R. R.: Reducing the Dimensionality of Data with Neural Networks, *Science*, 313, 504-507, doi:10.1126/science.1127647, 2006.
- Hochreiter, S., and Schmidhuber, J.: Long short-term memory, *Neural computation* 9.8, 1735-1780, 1997.
- Huang, L., and Hoefler, T.: Compressing multidimensional weather and climate data into neural networks,
670 10.48550/arXiv.2210.12538, 2022.
- Jia, W., Sun, M., Lian, J., and Hou, S.: Feature dimensionality reduction: a review, *Complex & Intelligent Systems*, 8, 2663-2693, 10.1007/s40747-021-00637-x, 2022.
- Jolliffe, I. T., and Cadima, J.: Principal component analysis: a review and recent developments, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374, 20150202, doi:10.1098/rsta.2015.0202, 2016.
- 675 Kingma, D. P., and Welling, M.: Auto-encoding variational bayes, *arXiv preprint arXiv:1312.6114*, 2013.
- Klöwer, M., Razinger, M., Dominguez, J. J., Düben, P. D., and Palmer, T. N.: Compressing atmospheric data into its real information content, *Nature Computational Science*, 1, 713-724, 10.1038/s43588-021-00156-2, 2021.
- Krähenmann, S., Walter, A., Brien, S., Imbery, F., and Matzarakis, A.: High-resolution grids of hourly meteorological variables for Germany, *Theor. Appl. Climatol.*, 131, 899-926, 10.1007/s00704-016-2003-7, 2018.
- 680 Lee, J. A., and Verleysen, M.: *Nonlinear dimensionality reduction*, Springer Science & Business Media, 2007.
- Lerch, S., Thorarindottir, T., Ravazzolo, F., and Gneiting, T.: Forecaster's Dilemma: Extreme Events and Forecast Evaluation. *Statist. Sci.* 32 (1) 106 - 127, <https://doi.org/10.1214/16-STS588>, 2017.
- Li, X., Feng, M., Ran, Y., Su, Y., Liu, F., Huang, C., Shen, H., Xiao, Q., Su, J., Yuan, S., and Guo, H.: Big Data in Earth system science and progress towards a digital twin, *Nature Reviews Earth & Environment*, 4, 319-332, 10.1038/s43017-
685 023-00409-w, 2023.
- Lindstrom, P.: Fixed-rate compressed floating-point arrays, in: *IEEE Transactions on Visualization and Computer Graphics*, vol. 20 (12), pp. 2674-2683, <https://doi.org/10.1109/TVCG.2014.2346458>, 2014.
- Liu, Z., Mao, H., Wu, C. Y., Feichtenhofer, C., Darrell, T., and Xie, S.: A ConvNet for the 2020s, 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 11966-11976, 2022.
- 690 Loshchilov, I., and Hutter, F.: Decoupled Weight Decay Regularization, <http://arxiv.org/abs/1711.05101v3>, 2017.
- Ludwig, K., and Bremicker, M.: *The Water Balance Model LARSIM - Design, Content and Applications*, Freiburger Schriften zur Hydrologie, Institut für Hydrologie, Uni Freiburg i. Br., 2006.
- Maaten, L., and Hinton, G.: Visualizing data using t-SNE, *Journal of Machine Learning Research* 9, 2579-2605, 2008.



- Massy, W. F.: Principal components regression in exploratory statistical research, *Journal of the American Statistical Association*, 60.309, 234-256, 1965.
- McCabe, M. F., Rodell, M., Alsdorf, D. E., Miralles, D. G., Uijlenhoet, R., Wagner, W., Lucieer, A., Houborg, R., Verhoest, N. E. C., Franz, T. E., Shi, J., Gao, H., and Wood, E. F.: The future of Earth observation in hydrology, *Hydrol. Earth Syst. Sci.*, 21, 3879-3914, 10.5194/hess-21-3879-2017, 2017.
- McInnes, L., Healy, J., and Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction, *arXiv preprint arXiv:1802.03426*, 2018.
- Müller, R. and Barleben, A.: Data-Driven Prediction of Severe Convection at Deutscher Wetterdienst (DWD): A Brief Overview of Recent Developments, *Atmosphere*, 15, 499, doi:10.3390/atmos15040499, 2024.
- Pearson, K.: On lines and planes of closest fit to systems of points in space, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2, 559-572, 1901.
- Ramer, U.: An iterative procedure for the polygonal approximation of plane curves, *Computer Graphics and Image Processing*, 1, 244-256, [https://doi.org/10.1016/S0146-664X\(72\)80017-0](https://doi.org/10.1016/S0146-664X(72)80017-0), 1972.
- Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., and Prabhat: Deep learning and process understanding for data-driven Earth system science, *Nature*, 566, 195-204, 10.1038/s41586-019-0912-1, 2019.
- Rittler, N., Graziani, C., Wang, J., and Kotamarthi, R.: A deep learning approach to probabilistic forecasting of weather, *arXiv preprint arXiv:2203.12529*, 2022.
- Sayood, K.: Introduction to Data Compression, *The Morgan Kaufmann Series in Multimedia Information and Systems*, Morgan Kaufmann, San Francisco, 790 pp., ISBN 978-0128094747, 2017.
- Schär, C., Fuhrer, O., Arteaga, A., Ban, N., Charpillot, C., Di Girolamo, S., Hentgen, L., Hoefler, T., Lapillonne, X., Leutwyler, D., Osterried, K., Panosetti, D., Rüdisühli, S., Schlemmer, L., Schulthess, T. C., Sprenger, M., Ubbiali, S., and Wernli, H.: Kilometer-Scale Climate Models: Prospects and Challenges, *Bull. Amer. Meteorol. Soc.*, 101, E567-E587, <https://doi.org/10.1175/BAMS-D-18-0167.1>, 2020.
- Schölkopf, B., Smola, A., and Müller, K.: Kernel principal component analysis, *International conference on artificial neural networks*, Berlin, Heidelberg: Springer Berlin Heidelberg, 1997.
- Shen, X., and Meinshausen, N.: Distributional principal autoencoders, *arXiv preprint arXiv:2404.13649*, 2024.
- Tintó Prims, O., Redl, R., Rautenhaus, M., Selz, T., Matsunobu, T., Modali, K. R., and Craig, G.: The effect of lossy compression of numerical weather prediction data on data analysis: a case study using enstools-compression 2023.11, *Geosci. Model Dev.*, 17, 8909-8925, 10.5194/gmd-17-8909-2024, 2024.
- Van Der Maaten, L., Postma, E. O., and Jaap Van Den Herik, H.: Dimensionality reduction: A comparative review, *Journal of machine learning research* 10.66-71, 2009.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I.: Attention is all you need, *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, California, USA, 2017.
- Walters, M. S. and Wong, D. C.: The impact of altering emission data precision on compression efficiency and accuracy of simulations of the community multiscale air quality model, *Geosci. Model Dev.*, 16, 1179-1190, <https://doi.org/10.5194/gmd-16-1179-2023>, 2023.
- Wang, Z., and Bovik, A. C.: A universal image quality index, *IEEE Signal Processing Letters*, 9, 81-84, 10.1109/97.995823, 2002.
- Wang, Z., Simoncelli, E. P., and Bovik, A. C.: Multiscale structural similarity for image quality assessment, *The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers 2003*, 1398-1402 Vol. 1392, 2003.



- 735 Winterrath, T., Brendel, C., Hafer, M., Junghänel, T., Klameth, A., Lengfeld, K., Walawender, E., Weigl, E., and Becker, A.:
Radarklimatologie aus angeeichten Niederschlagsstundensummen Version 2017.002 Gerasterte Niederschlagswerte für
Deutschland version v2017.02, doi:10.5676/DWD/RADKLIM_RW_V2017.002, 2018.
- Yu, Z., Li, D., Zhang, Z., Luo, W., Liu, Y., Wang, Z., and Yuan, L.: Lossy compression of Earth system model data based on
a hierarchical tensor with Adaptive-HGFDR (v1.0), Geosci. Model Dev., 14, 875-887, 10.5194/gmd-14-875-2021, 2021.
- 740 Yu, Q., Tolson, B. A., Shen, H., Han, M., Mai, J., and Lin, J.: Enhancing long short-term memory (LSTM)-based streamflow
prediction with a spatially distributed approach, Hydrol. Earth Syst. Sci., 28, 2107-2122, 10.5194/hess-28-2107-2024,
2024.