https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

On using neural networks to predict mean Age-of-Air from long-lived tracers

Jan Kaumanns[1], Florian Voet[1], Felix Ploeger[1], Peter Preuße[1], Jörn Ungermann[1], and Michaela I. Hegglin[1]

[1]Institute of Climate and Energy Systems - Stratosphere (ICE-4), Forschungszentrum Jülich, Jülich, Germany

Correspondence: Jan Kaumanns (j.kaumanns@fz-juelich.de)

Abstract. Climate models predict changes in the Brewer-Dobson circulation under a changing climate, which could have profound effects on tracer distributions and the radiative budget. Age-of-Air is an important concept for describing transport in the stratosphere and to understand and quantify global atmospheric circulation patterns such as the Brewer-Dobson circulation. Being an unobservable quantity, it must be inferred from other, directly observable quantities such as long-lived trace gases. It is therefore essential to have accurate ways of determining Age-of-Air through observations. These observations are subject to measurement noise, which is a long-known source of uncertainty when deriving Age-of-Air, as such uncertainties can affect the derived Age-of-Air significantly. We present a novel approach of using neural networks to derive Age-of-Air from long-lived trace gases. Multi-layer perceptrons can be used to predict model Age-of-Air with accuracy as little as a single month. The networks can be optimally trained according to expected measurement uncertainties. An unsupervised autoencoder is presented which is capable of achieving similar predictability almost without relying on model Age-of-Air inputs. This study presents an overview of these new approaches and discusses their capabilities, their accuracies and precisions mostly from a technical perspective regarding input parameters, regularization and predictions outside the training domain. Our approach allows us to derive Age-of-Air with accuracy of up to a single month under considerable measurement noise and over a wide altitude range. This accuracy is even retained when predicting values from a completely different period.

1 Introduction

The Brewer–Dobson circulation (BDC) is the zonally averaged residual meridional overturning circulation of the stratosphere, characterized by slow vertical motion—upwelling in the tropics and downwelling in the extratropics—connected by relatively rapid quasi-horizontal poleward transport(Brewer, 1949; Holton et al., 1995; Butchart, 2014). An overview is shown in Fig. 1.

The general transport is commonly described by a shallow branch, extending approximately to the mid-stratosphere (around 20 km) and a deep branch, reaching up to the stratopause (around 50 km). In a broader sense, the meridional circulation in the mesosphere can be considered part of the deep branch. The air descends in the mid- to high-latitudes, where the circulation closes. It is one of the most influential circulations for distributing ozone and other trace gases, including those of tropospheric origin, and is important for the composition and consequently radiative properties of the atmosphere (Diallo et al., 2021; Poshyvailo-Strube et al., 2022). The timescale of the BDC is typically in the magnitude of less than 2 years in the shallow

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

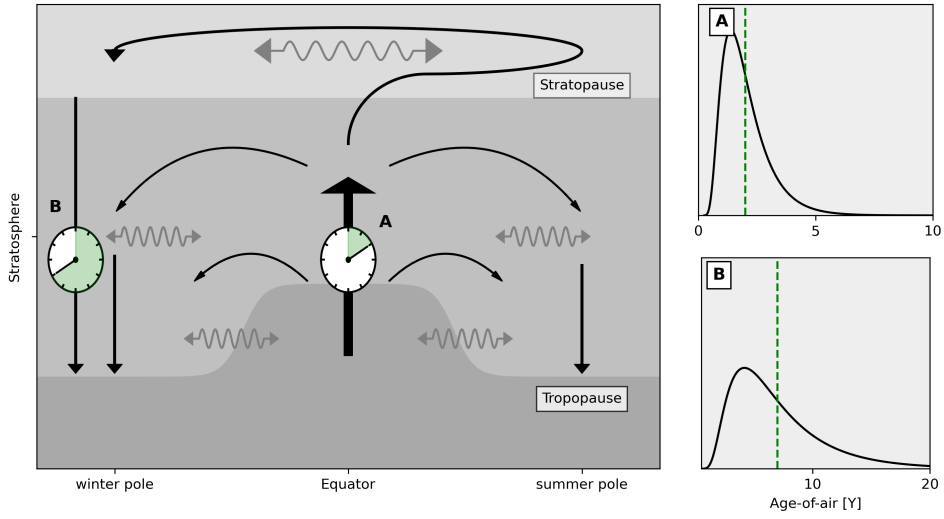

Figure 1. Illustration of the mean AoA concept. Air is uplifted from the troposphere in the tropics and transported polewards via either branch of the BDC (plain arrows). The two-sided wavy arrows indicate two-way mixing. Air may also be transported upwards as far as into the mesosphere, where it may reside for around one year Ray et al. (2017). The age spectra for both scenarios are sketched in the insets A and B, respectively. Adapated from ESA (2025).

branch (Lin and Fu, 2013) and up to 6 years for the deep branch (Poshyvailo et al., 2018).

Global climate models predict an acceleration of the BDC under increased greenhouse gas forcing, manifested by enhanced tropical upwelling, stronger extratropical downwelling, and shorter stratospheric transport times (Butchart et al., 2006; McLandress and Shepherd, 2009). This would lead to significant changes in the distribution of trace gas species. Observations of trace gas mixing ratios, however, are rather inconclusive, with some measurements suggesting no such long-term strengthening (Waugh, 2009; Engel et al., 2009) and more recent studies suggesting vertical dependencies (Hegglin et al., 2014) and also hemispheric differences (Stiller et al., 2017). Likewise, atmospheric reanalysis datasets — designed to provide optimal estimates of atmospheric state — exhibit inconsistent BDC trends over recent decades (Ploeger et al., 2019, 2021). It is therefore necessary to improve upon our understanding of the BDC and the wider circulation, particularly with the support of real measurements.

The BDC can quantitatively be characterized by the time which air takes from tropical entry to high and mid latitude subsidence. It is therefore directly linked to the stratospheric *Age-of-Air* (AoA), which refers to the time elapsed since an air parcel has left the troposphere (Hall and Plumb, 1994; Garny et al., 2024a). Different air parcels generally follow different transport pathways with different transit times.

Due to mixing processes occurring within the atmosphere, an air parcel is assigned a spectrum of AoA (Johnson et al., 1999; Ray et al., 2024), depending on the different pathways of the air parcels contributing to this air parcel after mixing. Generally

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

the first moment of the age spectrum is considered, as it is the closest to an observable quantity, and is often what is meant when AoA is mentioned. Here, we follow this example and will refer to this mean AoA unless otherwise noted. In Fig. 1 the air mass pathways and the mixing processes within the BDC are shown, along with example age spectra and their first moments at different locations. At location A (tropical tropopause region), most air parcels are transported directly from the tropopause within the tropical pipe, while only a small fraction reach this region via longer stratospheric pathways followed by horizontal mixing. Consequently, the age spectrum at location A is characterized by a pronounced peak at young transit times and a short tail, resulting in low mean AoA. In contrast, at location B (high latitudes in the winter hemisphere), air parcels arrive through a broader range of pathways, including rapid horizontal mixing of young air from the tropical pipe (shallow branch) and slow downwelling of older air (deep branch). As a result, the age spectrum at location B is more strongly skewed toward longer transit times, exhibiting a flatter peak and a longer tail compared to location A. An ideal clock tracer increases linearly in the boundary region and has no sources elsewhere and no sinks. Under these conditions, the mean AoA can be determined exactly from the stratospheric mixing ratio of this ideal clock tracer by the difference between the observation time and the matching boundary time (lag time; see, e.g., Hall and Plumb, 1994). Such ideal clock tracers can be implemented in models, but no such tracer exists in the real atmosphere. Therefore, AoA needs to be determined through different means.

The BDC, as a residual circulation, is too slow to be measured directly (e.g., via wind velocities). Instead, its characteristics are inferred from mixing ratios of long-lived trace gases (LLTs). Such LLTs have no stratospheric sources and stratospheric lifetimes which are long compared to the timescale of the BDC. Equally important is a linear (or at least well-known) change over time. To keep a consistent language we refer to all chemical species which have lifetimes of at least one year within the stratosphere as LLTs. Individual lifetimes of the species are not important for this study.

The LLT SF_6 is the most widely used tracer, and well-established methods exist to estimate AoA from its measurements (Maiss et al., 1996; Garny et al., 2024b). Although SF_6 has a very long atmospheric lifetime (estimated to be between 1910 and 2650 years; Vervalcke et al. (2026)), it is not entirely inert: it undergoes slow decomposition via dissociative electron attachment in the mesosphere (Loeffel et al., 2022). Garny et al. (2024a) demonstrated that this mesospheric sink introduces a substantial positive bias in AoA estimates for older air masses and proposed a correction method that is now widely used. The mesospheric sink correction is a crucial aspect when deriving AoA from SF_6. Other LLTs are also subject to other depletion mechanisms, such as the depletion of CH_4 through OH. AoA derived from an LLT can be viewed as a chemical age τ: the AoA that the observed mixing ratio would imply if the tracer had no sinks. Because real tracers do have sinks, τ generally overestimates the true AoA.

Volk et al. (1997) derived AoA from the correlations of different LLTs, with Voet et al. (2025) extending this approach to the correlations of many trace gas species beyond the classical LLTs. The core idea behind this approach is to construct lookup tables that relate AoA to the mixing ratios of different LLTs. These lookup tables are then used to derive individual estimates of AoA, from which then an uncertainty-weighted mean AoA is derived.

Remote sensing instruments on balloons, and especially on satellites, provide the densest spatial and temporal coverage of long-lived trace gas (LLT) measurements and are therefore particularly well suited for deriving AoA from LLTs. However, retrieving LLTs from raw remote sensing data is challenging and associated with substantial uncertainties (Stiller et al., 2008, 2012;

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

Haenel et al., 2015). Even after the LLT mixing ratios are obtained, the method by Voet et al. (2025) requires substantial preparatory effort, as each lookup table is only valid for a specific tracer, time period and spatial domain (i.e. altitude, latitude and longitude). Thus, a large number of these tables must be generated, which requires sufficiently large amounts of data. The choice of temporal and spatial discretization for constructing these lookup tables is also somewhat arbitrary, which may limit the method's general applicability.

We ask whether, rather than explicitly specifying the relationships between the LLTs and AoA through lookup tables, these relationships could be learned implicitly through the use of neural networks. How can neural networks be used to predict AoA from trace gas mixing ratios, and how accurate are such predictions? Neural networks are particularly suited to learn highly non-linear relationships between many inputs without requiring prior information of these relationships. Other advantages are excellent computational efficiency and generally high predictive accuracy. In this study we focus on simple neural network designs (in the following referred to as *architectures*) for maximal interpretability and inference. We use Multi-Layer Perceptrons (MLP) and Autoencoders (AE), both ordinary and probabilistic. We study the predictive capabilities of MLPs with respect to input parameters, regularizations and differences in datasets. Since MLPs are supervised machine learning methods, i.e. require some "true" AoA values to be predicted during training, we use simulations from the Chemical Lagrangian Model of the Atmosphere (CLaMS) for both training and validation. Observations are simulated by applying varying degrees of noise to the data. We also explore AEs, which are unsupervised neural networks and thus do not require "true" AoA values, as a way to derive a proxy for AoA and potentially a form of definition for AoA mostly without relying on model values for AoA.

This study is structured as follows: In Sect. 2 a brief overview is provided of the data and simulations used, the architectures of the neural networks employed, and the metrics by which network performance is judged. In Sect. 3 the results of several architectures are presented and discussed. Finally, a brief summary of our findings is given in Sect. 4.

2 Data and methods

In this section an overview is given over the simulations used and how the datasets for training were prepared from the model simulations. A brief overview of the architectures employed in Sect. 3 and the metrics applied to judge the predictive capabilities of the networks are given as well.

2.1 CLaMS simulations

In this study we use simulations of the Chemical Lagrangian model of the Stratosphere (CLaMS, McKenna et al., 2002b, a; Pommrich et al., 2014). Since CLaMS is a Lagrangian model, it simulates trace gas mixing ratios and mixing processes along trajectories, which allows it to resolve sharp gradients in the tracer fields because of reduced numerical diffusion and better separation between circulation and mixing (Charlesworth et al., 2020). The mixing can be directly controlled in a physics-informed way. The model was driven by ERA-5 reanalysis wind data provided by the European Centre for Medium-Range Weather Forecasts (ECMWF) Hersbach et al. (2020). The CLaMS model uses a hybrid vertical coordinate, which is a terrain following coordinate close to the surface and smoothly transitions into a potential temperature coordinate such that above

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

300 hPa it becomes a pure potential temperature coordinate Konopka et al. (2019). The lowest layer extends from the surface to around 1.5 km, and is the reference level for the ideal clock tracer used to determine the model AoA. The mixing is implemented by parcel creation and annihilation. The raw output of CLaMS consists of a point-cloud containing different trace gas mixing ratios along with the location of that air parcel. The simulation is initiated at 1st January 1979 and runs until the year 2023. The datasets used for the network training were created from the point-cloud output at 12:00 UTC of the respective day, with no interpolation performed. In the following simulation results for the years 2011, 2012, 2016 and 2021 are used, with a focus on the year 2011 (which is an arbitrary choice) and with 2012, 2016 and 2021 (one, five and ten years later) used for out-of-domain validation. In addition, data is interpolated onto an isosurface of potential temperature of 500 K and used for visualization of predicted results only. The LLTs used consist of CH_4, CFC – 11, CFC – 12, HCFC – 22, N_2O and SF_6, with the latitudes of the air parcels also used in some architectures.

2.2 Data preparation

We use preprocessed datasets created from the CLaMS simulations for model training and model validation. For each of the years 2011, 2012, 2016 and 2021 the simulation output of the first day of every month is considered. All data points with potential temperatures between 420 K and 600 K are selected. This corresponds to an altitude range of around 14 to 25 km. Conventional methods are considered reliable in this range (Kouznetsov et al., 2020; Loeffel et al., 2022; Voet et al., 2025, e.g.), such that we choose this range as well. All sets contain the LLTs CH_4, CFC – 11, CFC – 12, HCFC – 22, N_2O and SF_6 and may contain additional variables such as latitude and time. The selected sets are then normalized (per variable) to range (0, 1) to apply equal numerical weight to each variable. From each normalized set a subset of 50,000 random samples is created. The values in each dataset therefore are uniform, random samples across different altitudes, latitudes, and longitudes. These subsets are split into a training and a validation set, which contain 80% and 20% of the points, respectively. Sets containing several months may be created by merging their respective training and validation data sets, combining for instance the subset for January 2011 and the subset for February 2011 to create a subset for both months, with then 80,000 points in the training set and 20,000 points in the validation set.

Training data is exclusively used for training the neural networks. We utilize *early stopping*, i.e. termination of the training loop once the loss function used for training converges. The value of the loss function of a neural network after one training epoch here determines the training loss. An epoch refers to one pass through the available training data. The training data is generally shuffled after each epoch to mix up the order. Validation data is exclusively used for validating the network. In this way, no validation sample is seen by the model during training, so there is no direct overlap between the training and validation sets. Implicit correlations due to inherent similarity between training and validation data are inevitable and are ultimately even required for the usage of neural networks. Since the subsets are sampled randomly, we assume that the selected points represent the total data well, which visual inspection confirms. We separate both the training data and the validation data into input variables (features) and output variables (labels). While the features vary depending on the configuration of a particular neural network, the LLTs are always features. The labels are exclusively the model mean AoA. Thus, the predictive networks presented in the following provide predictions of AoA based on values of at least these six LLTs.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

In addition to the random samples of the simulation, we emulate observations by adding relative Gaussian white noise to the features, which are saved as new variables to the datasets. We consider noise levels of 1%, which roughly corresponds to the measurement precision of the in situ measurement technique AirCores which are used to measure LLTs (Laube et al., 2020), and 2%, 3% and 5%, which is the order of magnitude a limb viewing satellite instrument could achieve (Voet et al., 2025). At this point we would like to acknowledge that while white noise does capture the statistical uncertainty of an instrument, real instruments additionally produce systematic uncertainties, which may be more difficult for a neural network to learn and compensate for, but are specific to an individual instrument and are beyond the scope of this study. They are, however, not an insurmountable issue for neural networks if the general nature of these uncertainties is known.

2.3 Neural network architectures

A neural network is a machine-learning model consisting of multiple interconnected nodes (neurons) following the example of biological brains that can "learn" a function or relation from provided training data. Neural networks are an increasingly relevant tool in both science and everyday life. The main advantage of a neural network is that no prior knowledge of the function that it shall learn is needed, neither in terms of parameterization, form or the relevance of the inputs provided. All information about the function is derived from the training data itself. All neural networks presented here were built on the `TensorFlow` framework Abadi et al. (2015) in Python.

2.3.1 Multi-Layer Perceptron

A Multi-Layer Perceptron (MLP) is a simple feed-forward neural network consisting of fully connected neurons, organized in layers, with non-linear activation functions. It *perceives* data and produces an output based on this perception. An MLP is a powerful tool for learning highly non-linear relationships between its inputs and the predicted output. In principle an MLP consists of an input layer, which accepts the training data, an output layer, which returns an output in the shape of the training output it is compared against, and a number of hidden layers of various sizes. Each neuron of a given layer usually is connected to every neuron of the following layer. A typical layout of an MLP is shown in Fig. 2a, which is a design used in this study: This particular design consists of an input layer with six neurons, accepting inputs consisting of six values, followed by three hidden layers with 300 neurons each and finally an output layer of size one. Inputs are processed in groups of samples called *batches*; each sample in a batch is a vector of six values. There vectors are individually passed through the network. This input is fed-forward into the first hidden layer, which consists of 300 neurons. Each connection between two neurons has a trainable weight, which determines the importance of the relationship between these neurons for the final prediction. Each neuron also has a trainable offset b, which does not depend on the individual connection. Both the individual weights w_i and the offset are adjusted during training such that the model output best matches the training output. Each raw output z of a neuron is then passed as a value y to every neuron in the next layer that is connected to it via:

$$z = \boldsymbol{w}^\top \boldsymbol{x} + b \tag{1}$$

$$y = \phi(z) \tag{2}$$

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

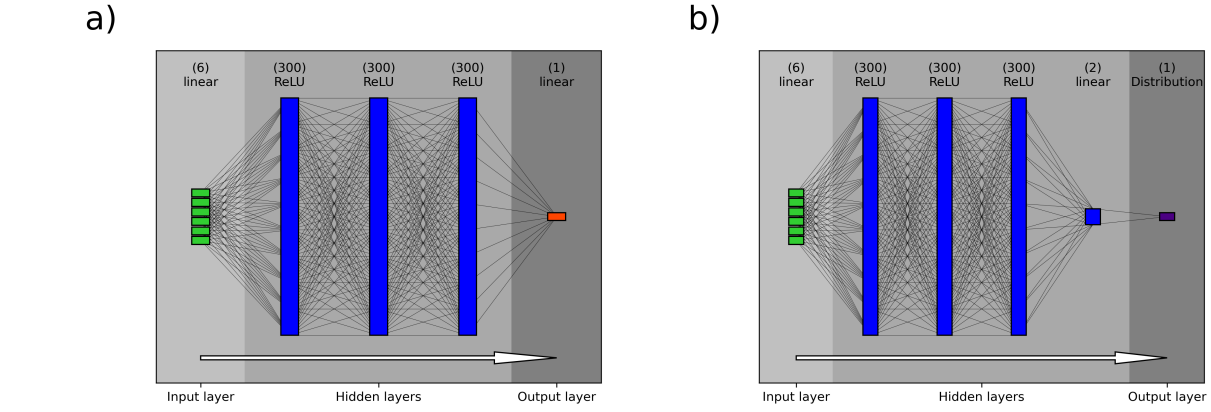

Figure 2. Sketch of a typical (supervised) MLP architecture. Panel a shows a regular MLP, panel b shows a probabilistic MLP. Shown is exemplarily the architecture for the six LLT training scenario. This architecture consists of 3 hidden layers containing 300 neurons each and an output layer of 1 neuron. The sizes shown are reduced compared to the true sizes for visualization purposes.

where $\boldsymbol{x}$ are the outputs from all connected neurons in the previous layer, $\boldsymbol{w}$ is the part of the weight matrix of the respective connection, b is the offset of the neuron and ϕ is an activation function such as the *Rectified Linear Unit* (ReLU), which is defined by:

$$\phi(z) = \max(0, z) \tag{3}$$

The input is passed successively through all hidden layers until it is passed to the output layer, which returns the final value. These values are then compared to the true values in the training dataset via a loss function L such as the *root-mean-square-error* (RMSE, see Eq. 8). The loss describes how well the true values were predicted. A large loss value indicates poor predictions, which necessitate large changes to the weights. A small loss subsequently indicates that the current state of the weights already yields good predictions. In each training step, i.e. for every batch, the values are passed through the network and the loss of the batch is calculated. How much each individual neuron contributes to a loss value (and therefore requires adjustment) is calculated via back-propagation. The weights w_i in W and b are updated according to:

$$\delta_{in} = \frac{\partial L}{\partial y}\frac{\partial \phi}{\partial z} \tag{4}$$

$$w_i \longleftarrow w_i - \eta \cdot \delta_{in} \cdot x_i \tag{5}$$

$$b \longleftarrow b - \eta \cdot \delta_{in} \tag{6}$$

where δ is referred to as the loss sensitivity at the neuron and η denotes the predefined learning rate. Passing every entry in the training data through the network during training is called an *epoch*. After each epoch the training data can be shuffled and new training batches can be created. Passing these new batches through the network still yields great benefit for the training process without risking overfitting. The number of epochs of course is limited and depends on the size of the network. As a

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

rule of thumb for a regression task neural network (used in this study) the training should provide between 20,000 and 50,000 weight updates. The number of epochs naturally depends on the batchsize. Since we are using a batchsize of 16 on datasets containing at least 40,000 samples we aim for 20 epochs. We employ an early stopping criterion if the training loss does not improve over the span of two epochs.

Rather than predicting a simple value, an MLP can be adjusted to predict a probability instead, see Fig. 2b. The final layer is replaced by a distributional layer, which returns a probability distribution (usually a Gaussian distribution), which requires a dense layer beforehand with a number of neurons corresponding to the number of parameters in the distribution layer. An overview into the advantages and mathematical foundation of probabilistic neural networks can be found in (Nix and Weigend, 1994; Gneiting and Raftery, 2007; Kendall and Gal, 2017). Implementations of these layers can be found in the *Tensorflow-Probability* package. A major difference for probabilistic MLPs is that the RMSE cannot (or should not) be used as the loss function L, as the predicted distribution cannot be reasonably compared to individual values with the RMSE. Instead, the *negative log-likelihood*:

$$\mathcal{L}(\theta) = -\sum_{i=1}^{N} \log p(y_i \mid x_i; \theta) \tag{7}$$

can be used, which expresses the likelihood that the true values y_i were realized from a given distribution given inputs x_i and parameters θ. Prediction uncertainty can be derived for regular MLPs by evaluating batches of similar (but distinct) inputs. A probabilistic MLP natively provides a measure of prediction uncertainty through the variance of the predicted distribution, since an ensemble of similar inputs then clusters around the predicted distribution. The robustness of these distributions, however, must be estimated similarly to single value predictions.

2.3.2 Autoencoder

An Autoencoder (AE) is an unsupervised neural network architecture which learns an encoding and its inverse decoding of some inputs, typically used for dimensionality reduction. It consists of two types of neural networks itself: an encoder and a decoder. Given a training input of size N, the encoder produces an output of size n, which is the latent dimension chosen for the encoder, with $N > n$. The output of the encoder is then fed into the decoder, which produces an output of the original size N. The aim during training is that the decoder accurately reproduces the training input from the output of the encoder. Thus the training inputs and outputs of an AE are identical. The decoder is the pseudo-inverse of the encoder. The main interest of an AE is the n-dimensional embedding the encoder produces. No prior information on such an embedding is required to be known. The constraint of an AE is that only such encodings can be determined which are invertible, such that the decoder can find the matching pseudo-inverse.

Encoder and decoder are usually MLPs. Due to the symmetry of the problem, encoder and decoder are typically mirror-symmetrical. Fig. 3 shows an AE as a combined model as used in Sect. 3.4. The architecture is similar to the previous MLPs. The left hand side of the network is the encoder, which returns a 1-D embedding of the 6-D encoder input, the right hand side of the network is the decoder, which transforms the 1-D embedding of the encoder back into a 6-D output, which is compared to the encoder input during training. A probabilistic AE is functionally identical to a regular one, but replaces the encoder

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

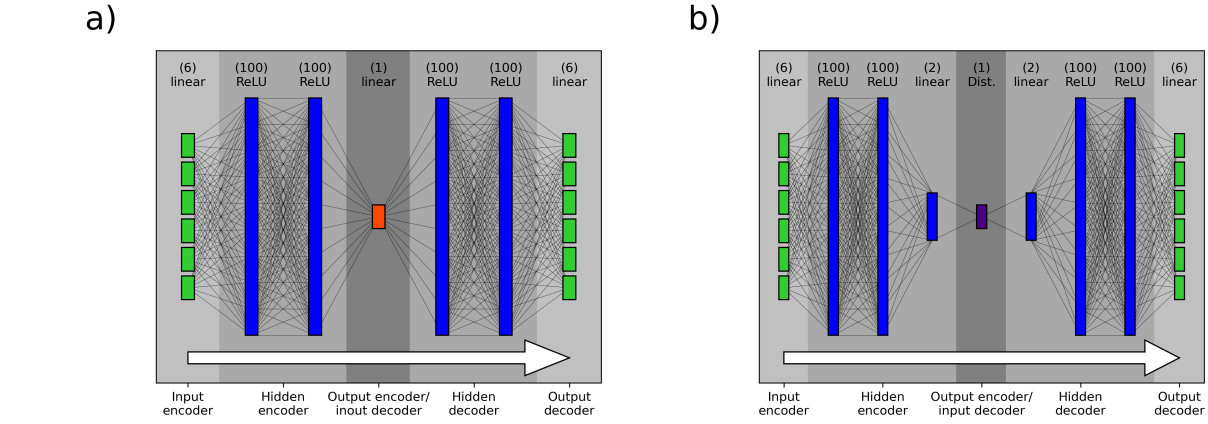

Figure 3. Sketch of the full AE architecture. The first half of the network is the encoder, which consists of two hidden layers with 100 neurons each and an output layer of one neuron. The second half of the network is the decoder, which takes the 1-D output of the encoder and attempts to return the original input. Its layers mirror that of the encoder. Shown is a regular design (panel a) and a probabilistic design (panel b).

output by a probabilistic layer. Instead of finding a deterministic (point) n-dimensional embedding, the probabilistic AE finds an embedding of distributions.

2.3.3 Regularization methods

Regularization encompasses a range of techniques designed to control model complexity – typically by adding penalty terms to the loss function or modifying the training scheme – in order to reduce overfitting and improve generalization in neural networks. While the goal of a neural network is to learn a general mapping from training data, it may instead learn spurious patterns contained in the training data. In such cases the network will fail to provide reasonable predictions for unseen data. Regularization introduces constraints that penalize such special patterns, thus favouring the more robust, generalizable patterns, at the cost of some training performance. This is particularly important for the prediction of AoA from LLTs, as the (supervised) neural networks must be trained on model data and are ultimately applied on observational data. Beside the natural discrepancies between model data and observational data, observational data may have intrinsic correlations between different tracers inherent to the measurement itself, such that these correlations may not be present in the model data. On the other hand, observations with high measurement uncertainties or poor sampling may have relevant correlations obfuscated, such that a neural network should not rely too heavily on these correlations. Several regularization methods exist. One method is dropout, where activations within the network are randomly set to zero during each training step, effectively preventing the model from relying too heavily on individual neurons or specific patterns. Another method is L_1 regularization, which adds a penalty to the loss function proportional to the absolute values of the model weights, encouraging sparsity and discouraging large parameter values associated with spurious patterns. In this study we employ data-based regularization by perturbing

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

the input features through the addition of random noise. This approach effectively smooths the data distribution and reduces sensitivity to the spurious patterns. A similar result can be achieved by including regularization layers in the architecture which do the same thing. Adding the noise beforehand has the added benefit that the perturbed features correspond directly to noisy measurements.

2.4 Evaluation metrics and concepts

In this study we use a variety of metrics to judge the predictive capabilities of our networks. In this section a brief overview of these core concepts with small examples is given for convenience: We differentiate between two types of uncertainties: *aleatoric* and *epistemic* uncertainty. The epistemic uncertainty of a problem describes uncertainty arising from an imperfect model or incomplete information, and can generally be reduced by acquiring more information (data) or improving the model. For example, a weather forecast may be epistemically uncertain if it is based on only a few temperature measurements (limited number and diversity of information), particularly if these are obtained using a low-precision instrument, such as a manually read analog thermometer (read-off uncertainty and limited instrument accuracy). The simple nature of the model used to predict weather from temperature alone is also a source of epistemic uncertainty. The aleatoric uncertainty of a problem is the inherent uncertainty due to the nature of the problem itself. A prediction of the weather is also aleatorically uncertain, if we assume that weather is somewhat chaotic in nature and does not strictly depend on particular measurable quantities. Regardless of how well the temperatures are known, the inherent variability of the weather makes perfect predictions impossible. When referring to aleatoric uncertainty, it is assumed that no feasible amount of information can render the problem strictly deterministic.

A predictive model $\mathcal{M}$ should be both *accurate* and *precise*: given a predicted value $Y_p = \mathcal{M}(X)$ and a corresponding reference value (or true value) Y_r, we consider $\mathcal{M}$ accurate if $\epsilon_1 = ||Y_p - Y_r||$ is sufficiently small, where $||\cdot||$ denotes a metric such as the Euclidean metric. The accuracy ϵ_1 is typically larger than zero, and reflects the aleatoric uncertainty of the prediction problem, i.e. the theoretical limit of how well an input X can be used to predict Y_r. We consider a model to be precise if the prediction of a small variation δ of an input $Y_\delta = \mathcal{M}(X-\delta)$ is close to the prediction Y_p, i.e. if $\epsilon_2 = ||Y_\delta - Y_p||$ is small as well. Mathematically the model is required to be continuous, which is generally the case in practice. An over-fitted model is highly accurate, but tends to have poor precision, since it does not reflect any X outside of its training domain. For a model which is both accurate and precise we get $||Y_\delta - Y_r|| \to \epsilon$, where ϵ is small. In the following we refer to the total predictive capability as the *performance* of a model, which is not an established expression, but is convenient. The average predictive capability of $\mathcal{M}$ can be measured, e.g., by the root-mean-square-error (*RMSE*), defined by:

$$\mathrm{RMSE}(A,B) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(A_i - B_i)^2} \tag{8}$$

where A, B denote the quantities which are compared against each other. In terms of the RMSE the accuracy of a model can be expressed as $\mathrm{RMSE}(Y_p, Y_r)$, the precision as $\mathrm{RMSE}(Y_\delta, Y_p)$ and the performance as $\mathrm{RMSE}(Y_\delta, Y_r)$.

While the RMSE is a good metric to compare different architectures, it leaves out important information. The largest (or smallest) deviation between Y_r and Y_p indicates the largest (smallest) error that can result due to a prediction and can be more

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

important than a low RMSE. Also, Eq. 8 does not indicate whether the model overestimates or underestimates the true values. Full information of the predictive capabilities can only be derived from the full prediction statistic.

3 Results & discussion

Having introduced all necessary concepts of this study, we return to the originally posed question: can we use neural networks to accurately and precisely predict AoA from LLTs? We present experiments for different architectures and different configurations. In Sect. 3.1 – 3.3, we investigate the performance of different MLPs to predict AoA from LLT inputs. In particular, we strive to answer the following points:

- Given an input of only LLTs, what effect does measurement noise have on the predictions, and how much regularization is necessary to reduce over-fitting?
- Can the performance of a model be improved by explicitly including additional features such as latitude or time? If so, what combination of features yields the best performance?
- Can a model, which was trained on data of a limited time period, accurately predict AoA for different years? Traditional approaches require regular updates over time. How far into the future can a neural network retain its performance?

In Sect. 3.4, we investigate the capabilities of an AE to derive some measure of AoA. We investigate the encoding and correlate it to the model AoA. Lastly, we consider a calibration of the encoding and derive a methodology to accurately calibrate this encoding to a quantity reflecting the ideal model AoA.

3.1 Impact of measurement noise & regularization on predictions

We begin with a simple ordinary and probabilistic MLP using the architecture shown in Fig. 2a and b, respectively. These models consist of three hidden layers with 300 neurons each and effectively a single output. While the standard MLP directly outputs the predicted values (Y_p), the probabilistic MLP predicts Gaussian distributions instead. The means (expectation values) of these distributions are used as the predicted values (Y_p), while their standard deviations provide estimates of the uncertainty associated with the predictions. The features of either model consist of the six LLTs, the labels for these (and all following models) are the CLaMS model AoA. Both models are trained and validated on the dataset derived for the 1st August 2011. The LLT features without added noise are used during training. The validation is performed on every noise levels instead (0%, 1%, 2%, 3% and 5%, see Sect. 2.2).

Fig. 4 and Fig. 5 summarize the diagnostics mentioned in Sect. 2.4 for both models, respectively. In Fig. 4a the predicted AoA values are plotted against the true values of the CLaMS model. Note that these values were predicted based on input data without additional noise. A perfect model would predict the exact model values, such that all points would lie on the identity line (shown in gray). An imperfect (real) model deviates from the true model values. The average deviation between prediction and true CLaMS model values for this model is its accuracy represented by an RMSE(Y_r, Y_p) of around 15 days.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

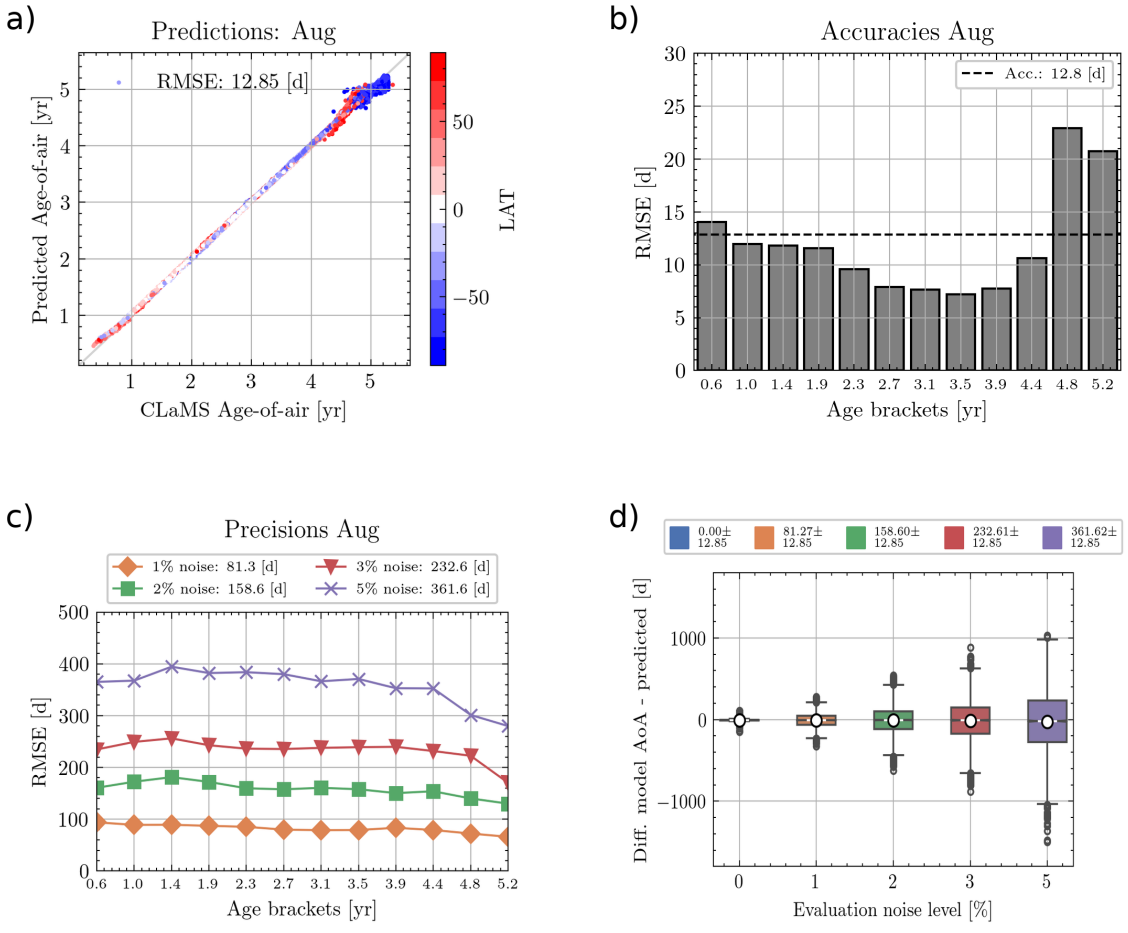

Figure 4. Diagnostics of the ordinary MLP. Panel a): Correlation between the predicted AoA based on noise-free features and the true CLaMS model AoA. The gray line indicates a perfect prediction (identity). Colors represent the corresponding latitude of each data point. Panel b): Prediction accuracy as a function of the predicted value. Here, accuracy is quantified via the root-mean-square error between the reference and predicted values, (RMSE(Y_p, Y_r)), reflecting the total systematic and random error relative to the CLaMS truth. The black dashed line indicates the mean accuracy. Panel c): Prediction precision as a function of the predicted value. Precision is quantified via the root-mean-square error between the noise-free and noise-perturbed predictions, (RMSE(Y_δ, Y_p)), isolating the sensitivity of the model to input noise. Precision is identically zero for noise-free inputs ('$Y_p = Y_\delta$') and hence not shown. Panel d): Statistical distribution of the differences between predicted (Y_p) and true values (Y_r). Boxplots show the median (white line), mean (white dot), 50% quantiles (boxes), whiskers (margins), and outliers (circles). Color-coding represents the validation noise level consistently across all panels. The legend displays the mean precision and accuracy for each specific validation noise level.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

This indicates that in principle the AoA can be accurately predicted from this set of LLTs. The latitude associated with each prediction (i.e. the latitude of the point from which the LLT values are taken) is indicated in color-code and does not show any significant systematic structure. Fig. 4b shows the accuracy represented by the RMSE(Y_r, Y_p) resolved by the predicted AoA, with the average RMSE indicated by the black, dashed line. While there is a small loss in accuracy (higher RMSE) towards higher AoA, the accuracy is fairly consistent across all values. Interesting is the response of the model for inputs with noise added, as is illustrated in Fig. 4c through the precision represented by the RMSE(Y_δ, Y_p). Shown are the errors for 1, 2, 3 and 5 percent input noise as a function of different values of AoA. While the model shows a minor increase in precision (lower RMSE) towards higher predicted AoA, each percent of noise added to the input increases the mean RMSE by around 100 days. The slight decrease in RMSE at high AoA for 5% noise (purple line, "x"-markers) is negligible relative to the year-scale mean error (≈364 days). The model shows the characteristic signs of overfitting, as small deviations in the inputs lead to progressive loss of precision. The poor precision indicates a very high epistemic uncertainty, which is the limiting factor of this model. The distribution of the difference between predicted and true values is summarized in Fig. 4d. While the core of the predictions shows no clear trend towards over- or underestimating, the width of the difference distribution increases significantly with increasing input noise, and statistical outliers show a strong bias towards overestimation for high input noise (5%, purple boxplot).

Similar diagnostics for the probabilistic MLP are shown in Fig. 5, with a few conceptual differences. Fig. 5a shows the correlation between the predicted values and the true model values. Since the probabilistic MLP predicts a Gaussian distribution, the predicted values are taken as the means of the predicted distributions. The accuracy (Fig. 5b) is again represented by the RMSE(Y_r, Y_p), which was calculated similarly to the ordinary MLP through an ensemble of predictions. However, since the negative-log-likelihood loss (see Eq. 7), which was used as the loss function of this model, optimizes the distribution means and standard deviations with respect to the true values, the predictions of similar inputs scatter around the true model values within an average standard deviation. We can thus interpret the standard deviation of the predicted distribution as a measure of the model's uncertainty for each prediction. This is illustrated in the predictions without noise (blue), as their average deviation (11 days) corresponds to the average standard deviation (9 days). A well trained model reflects its aleatoric uncertainty in its accuracy. Considering the predictions of the inputs with added noise (1% input noise, orange) we see that the aleatoric uncertainty (scatter in Fig. 5a) is no longer reflected by the average standard deviation (9.1 days).

Fig. 5c, d correspond to Fig. 4c, d and show comparable results. While the complete diagnostics provide detailed insights into a particular model, it is sufficient to focus on some diagnostics when comparing different models.

Since both ordinary and probabilistic MLPs are consistently very similar in their predictive capabilities, in the following results will be illustrated for the probabilistic type representatively.

To improve the generalisability of the architectures some form of regularization is used. In this study we apply regularization by training the models on noised inputs (data manipulation, see Sect. 2.3.3). The advantage of this form of regularization is that the noise added to the inputs can be correlated with measurement noise.

Next we train new MLPs (both regular and probabilistic) with the same architectures on the dataset of the same day (1st August 2011). Each model is now trained on noised inputs (1%, 2%, 3% and 5% noise, respectively), for a total of eight individual

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

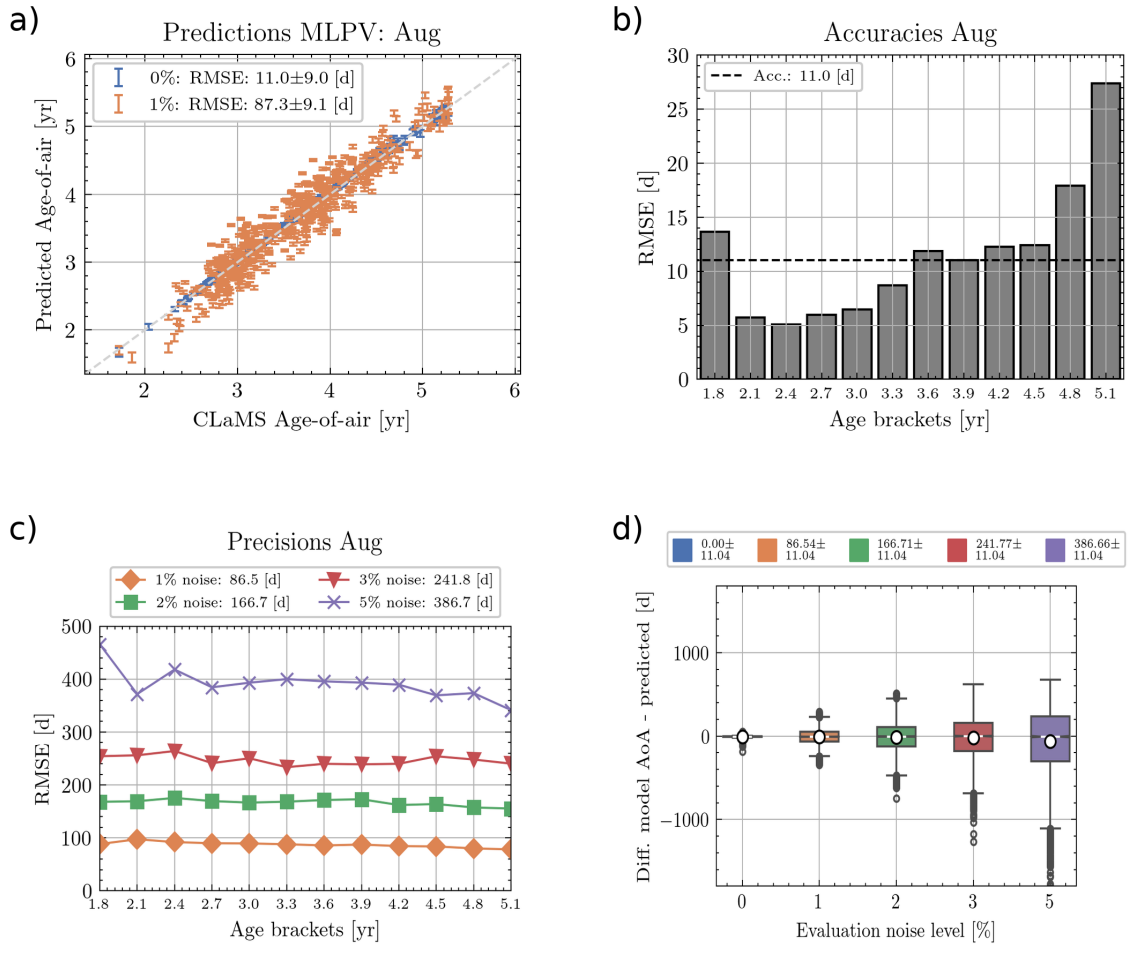

Figure 5. Diagnostics of the probabilistic MLP. Panel a): Correlation between predicted AoA based on noise-free features (blue) or features with added noise (orange) and the true model AoA. The standard deviation of the predicted distribution is shown as error bars and indicates the prediction confidence per predicted value. Panel b): Accuracy of the predictions as a function of the predicted value. The accuracy is represented by the RMSE between true and predicted values (RMSE(Y_p, Y_r)) similarly to Fig. 4b. Panel c): Precision of the predictions as function of predicted values. The precision is represented by the RMSE between between the noise-free and noise-perturbed predictions (RMSE(Y_δ, Y_p)) similarly to Fig. 4c. Panel d): Statistic of the differences between predicted (Y_p) and true values (Y_r), similarly to Fig. 4d.

340 models. Each model is validated the same way as the models before. Fig. 6 shows the distribution of the differences exemplarily for the probabilistic MLPs.

Regularizing the input features leads to vast improvement in the predictive capability of either architecture. The precision of predicting unmodified inputs (blue) has decreased significantly (from 11 days to between 27 and 55 days) as expected. The

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

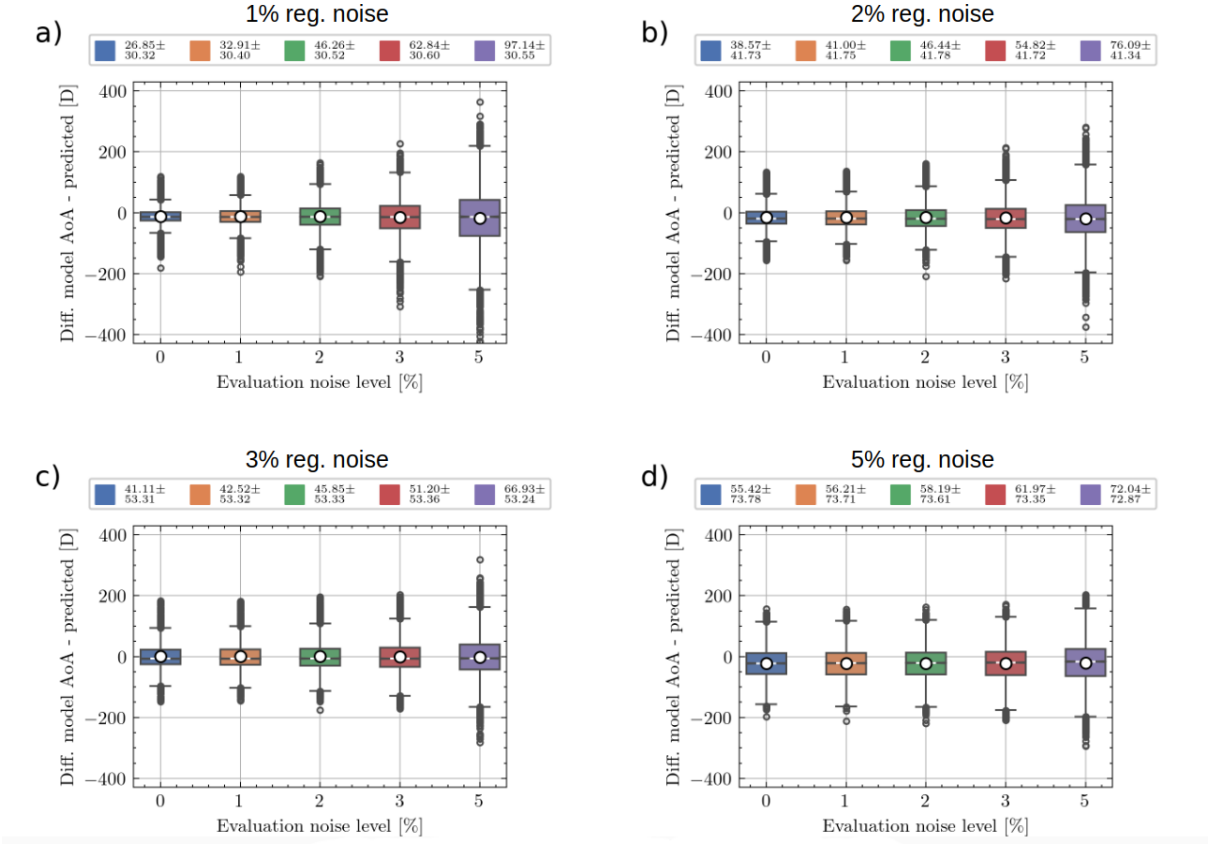

Figure 6. Difference diagnostic for probabilistic MLPs trained on different levels of feature noise. Panels a-d show the distribution of the differences for 1%, 2%, 3% and 5% regularization noise, respectively. The corresponding precision and accuracy of each validation is shown in the legends. The models show fairly constant performances for the different validations, since the accuracy is determined during training, and while it is not impossible for a model to adjust its accuracy for different predictions, the models fail to do so. The statistics are functionally identical to Fig. 4d and Fig. 5c.

precision of predicting inputs with added noise has improved significantly across all configurations, from up to around 500 days precision down to at most 97 days, depending on the regularization noise. The accuracy of the validations is comparable in size to the precision when evaluating a model on data with similar noise level to the regularization noise. For these validations the model's epistemic uncertainty reflects the aleatoric uncertainty; the model "recognizes" its prediction limits. Otherwise, the accuracy is rather constant across different validations. The difference distributions show a narrowing of the spread and a reduction of extreme outliers toward higher training data noise for all noise levels of the validation data. Predictions for most validations lie within two months of the true model values. A visible trend is that those models perform best if the noise level of the validation data matches that of the training data. An exception is the 5% validation noise, which is best predicted by a model with 3% regularization noise. It appears as if there is an upper limit for the regularization noise after which the variability

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

of the training data does not contribute to additional prediction improvements.

The very constant accuracy across different levels of epistemic uncertainty is a known issue, and subject to current debate and research (Ovadia et al., 2019; Foong et al., 2020; Kirsch, 2025, e.g.). It is sometimes known as the "epistemic collapse", where a model fails to recognize changes in epistemic uncertainty. Within the scope and goals of this study this issue must remain unresolved. We can, however, draw useful conclusions from this behavior: If one wants to use an MLP to predict AoA from real observations, the model should be regularized with noise comparable to the expected measurement noise. Since the regularization noise can be freely chosen, this particular choice would be expected to yield the best possible performance. Probabilistic MLPs would then also provide a realistic estimate for the epistemic uncertainty per prediction without the need to validate any ensembles (which require a ground truth). A small quirk visible in the validations is that for higher regularization noise the models show systematic under- or overestimation of AoA (Fig. 6c, d, respectively), with a slight asymmetry in the whiskers. While these effects are only weakly pronounced, they should be considered for specific applications of this methodology.

Another option is to use various noise levels simultaneously during training. If the model sees a wide range of measurement noise it is likely that it learns to accurately predict data with specific measurement noise. The resulting performance is basically the best of all worlds, and is used in the following considerations.

In summary, MLPs are capable of accurately predicting model AoA from LLT features. If the inputs are subject to noise, a regularization in the form of adding noise to the training features should be applied. Model performance lies around 1 month for low noise validations and around 3 months for high noise validations. Aleatoric uncertainty reflects the epistemic uncertainty well if the validation noise level is similar to the regularization noise level. Probabilistic MLPs perform similarly to regular MLPs with the benefit of providing an estimate of prediction uncertainty naturally. If an MLP is to be trained to predict noisy observations, the regularization noise should be of equal magnitude as the observation noise, if the observation noise does not exceed around 5%, in which case a lower regularization noise may yield better performance.

3.2 Feature selection & performance

So far we have only considered LLTs as features and evaluated models on a single dataset for the 1st August 2011. In general, any observable quantity can be used as feature for a neural network. The choice to include a particular feature is warranted if one can expect that this feature conveys information useful for the prediction. For example, considering the diagnostics for the probabilistic MLP with 1% regularization noise in Fig. 7a,b, we see some systematic structures in the predictions for the previous architectures:

Fig. 7a shows the difference between CLaMS model AoA and the AoA predicted from noised LLT features. In the southern hemisphere AoA is underestimated. In the northern hemisphere AoA is overestimated, with overestimation exceeding 3 months. The southern hemisphere shows stronger underestimation than the northern hemisphere. This suggests that explicitly adding latitude to the existing LLT features could improve model performance. So far it was also not shown if seasonal effects and variations are captured well by the previous models. Considering the evaluations on the other datasets of 2011 in Fig. 7b we see pronounced systematic patterns. While the performance is best around August, which is the month from which the training

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

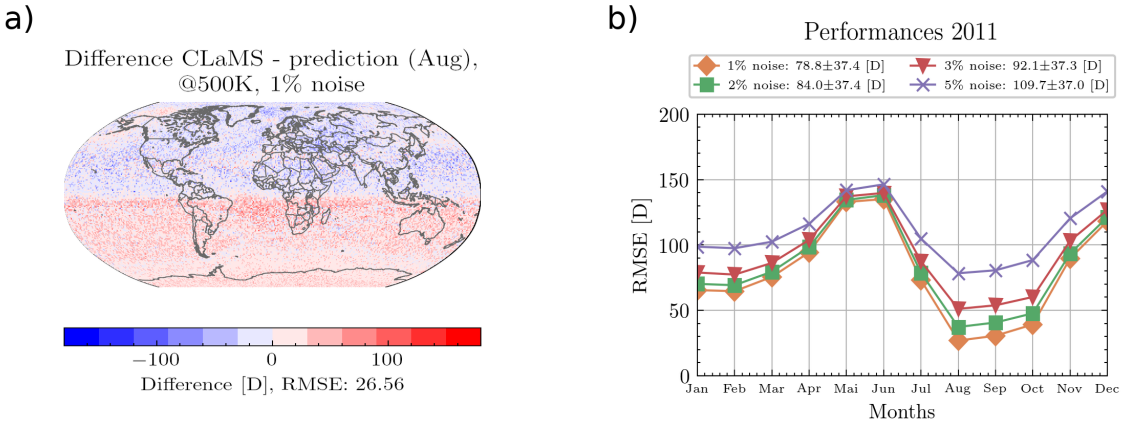

Figure 7. Diagnostics for particular model artifacts. Panel a): Spatial distribution of the differences between true CLaMS model and predicted AoA for a probabilistic MLP trained with 1% regularization noise. Shown is an isosurface of 500 K potential temperature, corresponding to an altitude range between 19 km and 22 km. Predicted are inputs with 1% simulated noise. Red regions indicate underestimation of AoA, blue regions indicate overestimation. White regions indicate good agreement between CLaMS model AoA and predicted AoA. The blur in the distribution stems from the inputs with added noise used to predict AoA. Panel b): Performance of the same model for different datasets of 2011. The performances are expressed in terms of RMSE(Y_δ, Y_r) for different validation noises. The precision and accuracy of each validation is shown in the legend.

data of the models were sampled, it diminishes for any other month, particularly for May and June. The trained models inherent the bias of only having "seen" data from a single month and therefore cannot accurately predict seasonal effects.

To remedy those shortcomings, we apply the following adaptations to the current training setup: the latitude coordinate can be included in the features, which transfers information about the zonal distribution of AoA. Instead of a single monthly dataset, multiple datasets may be used during training. Including all months during training is possible in principle, but would ultimately lead back to the original problem that the model cannot be trained on *all* possible data. This of course would improve prediction performance, but the aim should be to learn overarching patterns from limited amounts of data. For these reasons we include the months February, May, August and November in the training data, i.e. every third month and thus one month per season. We refer to such a dataset as a seasonal dataset. In addition to a seasonal dataset, time can be included in the features as well by numbering the months. In addition to the LLTs such a dataset contains a feature whose value indicates the month of any particular data point. This partitions the training data into each season, whose influence on the prediction is then determined by the model during training. A cyclic time coordinate was also tested, but we found this implies similarities between seasons that are not present in the data and yields worse performance. This might be due to the limited amount of months considered and may improve once larger, more complete datasets are considered. Since both latitude and time are known without significant uncertainty, we do not apply noise to these features. The resulting performances are shown in Fig. 8. Similarly to Sect. 3.1

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

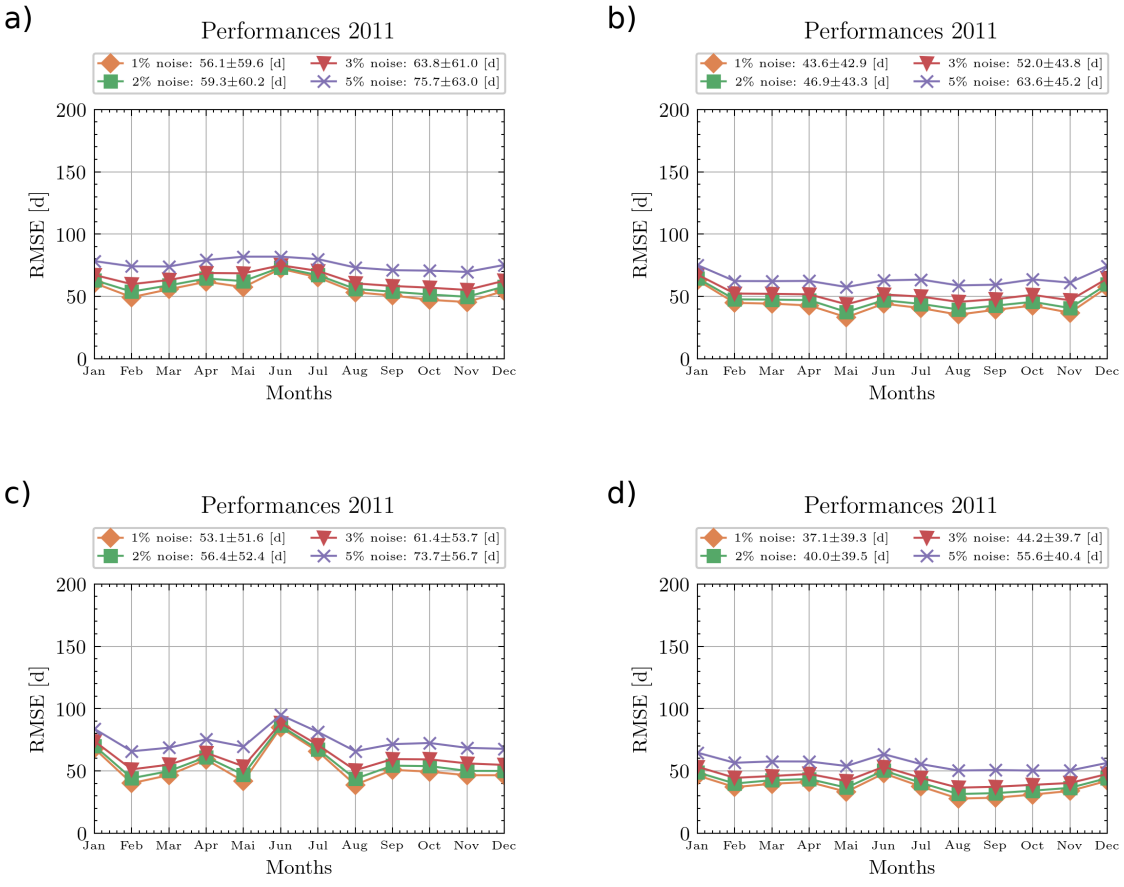

Figure 8. Annual performances of probabilistic MLPs trained on different features. All models were regularized with various noise where applicable, and therefore show reduced variability between validation noise levels.Panel a): Previous design as showed in Fig. 6, but trained on a seasonal dataset with implicit time dependence only. Panel b): MLP trained on a seasonal dataset which includes latitude as a feature. Panel c): MLP trained on a seasonal dataset which includes time explicitly as feature. Panel d): MLP trained on a seasonal dataset which includes both latitude and time as features.

the performance is represented by the RMSE, with lower RMSE values indicating higher performance. Training an MLP with a seasonal dataset (Fig. 8a) improves the performance by around 20%, and shows only small variations across the seasons. Including more training data (or volume) expectedly leads to higher performance. The model trained with explicit spatial knowledge (Fig. 8b) shows greatly improved performance by around 40% compared to the single-month model, but retains some systematic performance issues, particularly in December. The model trained with time as an explicit parameter shows the poorest overall performance of any of the adjusted designs (Fig. 8c), with an outstandingly poor performance in June. A potential reason for this performance loss by simply including more information is as follows: If there is only a weak

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

correlation between AoA and the time coordinate, the dataset including the time coordinate naturally has higher variability (with respect to AoA) than a dataset without it. Each feature also contributes through its correlation to other features; it acts as a covariate. Such covariates may show complex, incidental correlations to other features, which the neural network can pick up, but ultimately does not contribute to accurate predictions. If the time coordinate alone does not provide valuable information it can lead to reductions in performance. The exact reasons as to why the time coordinate decreases performance is currently unknown. However, combining both time and latitude in a seasonal training data yields a model with the highest performance so far, see Fig. 8d). While the systematic issues remain, particularly for June and December, the performance only exceeds 60 days for the highest validation noise. Whatever negative effect was introduced by the inclusion of the time coordinate is seemingly resolved by the inclusion of the latitude. It would seem as if the time coordinate only acts as a meaningful covariate to the latitude, which given seasonal structures is plausible. If we remove those months with poor performance from the total annual performance we would get an average annual performance around 1.5 months consistently. From these experiments it becomes clear that seasonality of AoA is only poorly captured by a simple MLP. While the performance is still high, one must recognize these variations in performance. One may consider training an additional, dedicated model for a time period where a more general model shows poor performance. The equivalent, regular MLPs yield very similar performance and are not shown here.

In summary, predictions by an MLP for seasons not included in the training data are less precise. Including more training data improves upon this, as does including more features in the data. An outlier is an explicit time coordinate, which in our experiments only improves model performance if paired with latitude as a feature. The best performing model of our ensemble reaches average performance of around 1.5 months for the entire year of 2011.

3.3 Predicting the future

In the previous section we have discussed the performance of MLPs for predicting a complete year from limited training input. The best design yielded an average performance of around 1.5 months. However, we have only considered data sampled from one year so far. A useful prediction method should be transferable to other data significantly outside of its training domain. It is a well-established result from conventional approaches that models cannot, in general, be directly applied to data from substantially different distributions. If that was the case we would have access to a universal formula or algorithm which determines AoA for all cases. Since a neural network only learns a function as implicitly encoded within the training data, predictions of inputs from outside of that training domain are only possible if the function conveys an overarching structure for many months - or even years - into the future. Any changes in that function in the futureMcLandress and Shepherd (2009); Ploeger et al. (2021, e.g. any changes such as predicted by), could lead to significant performance losses.

So far no conventional method is capable of long term prediction of AoA without proper adjustments, i.e. relying on new observations. Following the results of the previous sections, we take the probabilistic MLP trained on a seasonal dataset of 2011 with all previously considered input features (see Fig. 8d) and evaluate it on the corresponding datasets created from data of the years 2012, 2016 and 2021. The performances are shown in Fig. 9: As expected, the performance for each year is reduced compared to the training year. However, the performance is much less reduced than one might have expected. For 2012

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

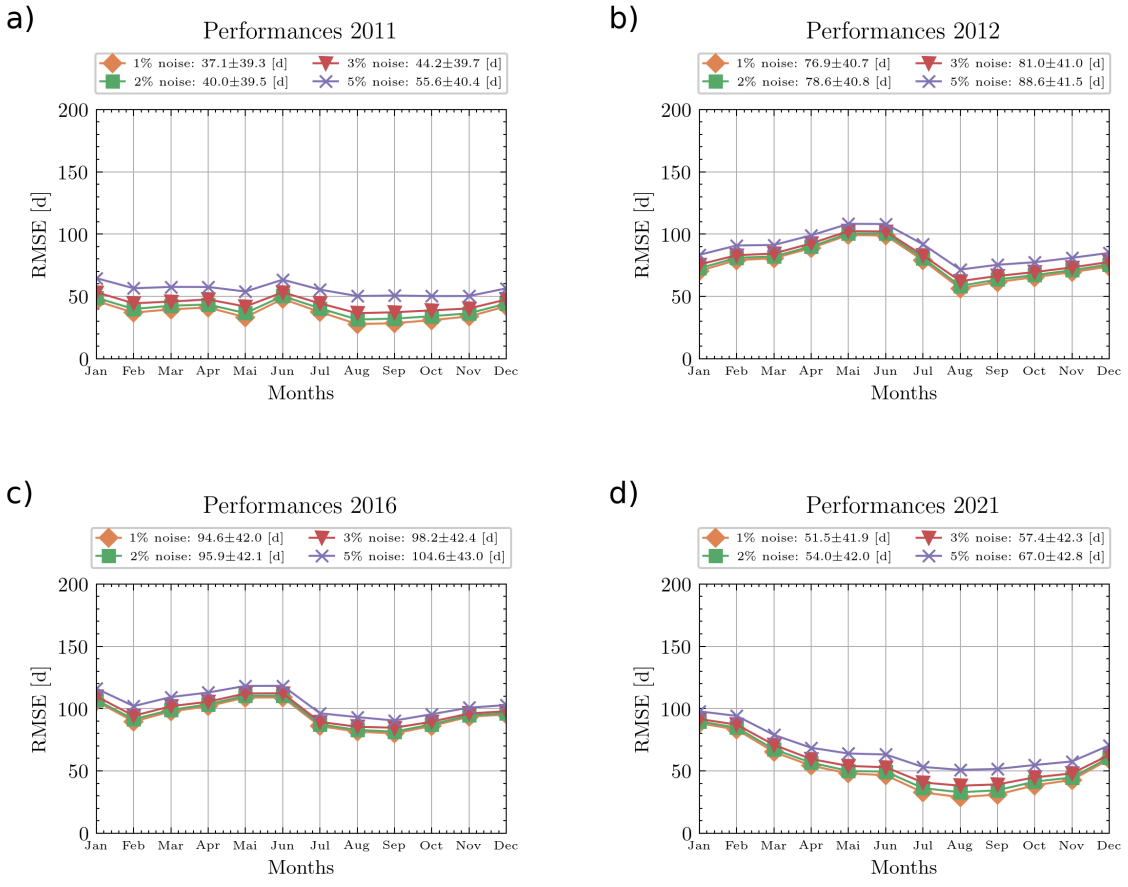

Figure 9. Performance evaluation of datasets from different years. The model is a probabilistic MLP trained on a seasonal dataset of 2011 with full features and various feature noise regularization. Shown are the years 2011, 2012, 2016 and 2021 in panel a, b, c, and d, respectively.

in Fig. 9b) pronounced systematic differences in performance are visible, with less performance in the boreal spring months. Average performance lies below 3 months, with (estimated) epistemic uncertainty of around 40 days (down from 60 days for 2011), indicating that the model actually does not recognize its shortcomings. 5 years later (Fig. 9c) the situation is comparable, with slightly decreased performances of just over 3 months, similar seasonal patterns and the same epistemic uncertainty. In Fig. 9d the performances for 2021, i.e. one decade after the training data, is shown. The model performs better for this year than the other two, with average performance comparable in size to the training year 2011, but with a new, seasonal pattern in its performance with high performing months in boreal autumn and low performing months in boreal winter.

We are unaware of any reason why this should be. Ultimately we must attribute this high performance to chance. Still, the model retains a high level of performance over many years far outside its training domain. Across all validated years the (estimated) epistemic uncertainty of the model remains close to 40 days, such that this estimate is likely inaccurate and most of

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

the uncertainty is aleatoric in nature. The model cannot intrinsically reconcile these issues. There is ultimately a limit to how different a dataset can be to be accurately predicted, even if the precision is surprisingly high. The regular MLP shows similar predictions and is not shown here.

Conventional methods require current data in order to determine the tropospheric mixing ratios. Thus data for 2021 would be needed if one wanted to determine AoA for that year. The MLPs show that they do not rely on such a thing and retain high performance even for those years which were not known during training. In this study a model trained on data from 2011 is still accurately predicting AoA from "measurements" of 2021. One reason why an MLP appears not to depend on this and is inherently capable of retaining its performance across many years lies in the preprocessing of the input data. The MLP is trained to map certain feature constellations to certain AoA values, or in the case of a probabilistic MLP to express the training inputs as distributions from which AoA is predicted. During preprocessing every dataset is normalized to range (0, 1) such that the absolute scale is lost, but the correlations are retained. While absolute values may change significantly over many years, the correlations between the normalized features stay approximately constant. The scaling of input data back to physically relevant quantities such as AoA is learned by the MLP from the training output. Any linear transformation which translates the mixing ratios observed in one year to those observed ten years after is implicitly accounted for during preprocessing. Thus the scaling of input data remains valid for future data.

In summary, an MLP trained on one specific year of data can still accurately predict AoA from data of other years. This makes it an interesting tool as it does not require additional calibration. However, the MLPs shown here are not capable of detecting more complex seasonal effects such that certain months may have less performance, or more performance in the case of 2021. An up-to-date model, i.e. one that is trained on more recent data, is generally preferable, but such models can be used to predict future values (and likely past values) with good performance such that frequent training of new models based on new data is likely not required.

3.4 Autoencoder

Up to this point we have presented *supervised learning* neural networks, i.e. such networks which require reference or *true values* during training. These networks are trained to find a function which best matches these true values to the training input, such as the LLTs considered in this study. In this configuration the performance is limited by the quality of all training data, which are usually taken from an atmospheric model or have to be derived from conventional methods and therefore are subject to their own complications.

Therefore, a long-standing goal is to become independent from climate models when deriving AoA, as individual models themselves carry their own inherent biases. There are well-established standard methods to infer AoA from measurements of approximate clock tracers, most commonly SF_6 (e.g. (Volk et al., 1997; Garny et al., 2024b)), but up to this point, no method is completely free of any model information (correction of mesospheric SF_6-sink, knowledge about ratio of moments).

Neural networks allow for an alternative approach to AoA in the form of Autoencoders (AE). An AE is a special *unsupervised learning* neural network, which is commonly used for dimensionality reduction. The general concept is described in Sect. 3.4. An AE derives a lower dimensional representation of a given set of input data without relying on any critical assumption on

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

that input data. If a 1-D encoding of the LLTs exists and if this encoding can be correlated to AoA, it may be possible to derive a definition of AoA based on this encoding of the LLT input. All information on the encoding is derived exclusively from the training input. Since the mixing ratios of all LLTs used in this study primarily depend on the transit time Γ within the atmosphere it is likely that sufficient information about Γ is indeed contained.

Generally the reduction of 6-D data to a 1-D encoding is highly non-trivial. However, in the case of the LLTs used here we demonstrated with the supervised MLPs that such a 1-D encoding between the 6-D LLTs and the 1-D model AoA most likely exists. A second argument for a likely success of the approach can be made by calculating the effective dimensionality (Wang, 2008) of the input data, which yields an effective dimension of around one and thus implies the existence of such a 1-D encoding.

In this study, we use simulated LLTs to demonstrate the feasibility of the approach in a single use case and discuss the implications of it. Given our previous results, we decided to design the AEs similarly to the MLPs as shown in Figure 10.

Similarly to the previous MLP studies, we consider a regular and a probabilistic AE design. The architectures for the *encoders* and *decoders* are shown in Fig. 10. Panels a and b show the encoder and decoder of the regular AE, and panels c and d show the encoder and decoder of the probabilistic AE, respectively. Since the probabilistic AE predicts a (single) distribution (see Sect. 2.3.1) an intermediate layer of size two is included in the encoder. Due to the symmetry of the problem, encoder and decoder are otherwise mirror-symmetrical. As described in Sect. 3.4, the encoder of such an AE finds a 1-D encoding of a set of LLT observations. The decoder can recreate the original inputs from this encoding to differing degrees depending on the training data. Within some precision limit the decoder is the inverse transformation of the encoder.

3.4.1 Simple calibration

In the following we present a probabilistic AE trained on the August 2011 dataset with 1% regularization noise and evaluated on the previously shown 500 K potential temperature isosurface on the 1st August 2011. The regular AE yields very similar results such that presenting its encoding yields very few additional benefits. We follow a two-step approach: We first motivate the correspondence between the 1-D embedding and AoA qualitatively by considering its spatial distribution. Afterwards, we derive ways to perform calibrations such that the encoding can be directly translated to AoA.

Fig. 11 show the results of the probabilistic AE. Fig. 11a shows the "true" CLaMS model AoA, Fig. 11b shows the raw, unprocessed encoding of the AE. As mentioned before the scale of the encoding is in principle arbitrary. Since the input was scaled to range (0, 1) the encoding lies in a similar range. The encoding is also ambiguous up to an overall sign (a factor of ± 1) due to the symmetry of the AE.

The spatial distribution of this embedding does follow the structure of the CLaMS model AoA very well. The main features such as the tropical pipe in the tropics, the polar vortex at the south pole and filamentary structures in the extra-tropics are all visible in the encoding similar to the CLaMS model AoA. When considering the correlation between the encoding and the CLaMS model AoA as shown in Fig. 12a we see that there is a large regime (for AoA < 4 years) where the embedding depends linearly on the CLaMS model AoA. Thus, a simple form of calibration can be performed by linearly scaling the encoding such that it corresponds closely to the CLaMS model AoA in this linear regime. This scaling could in principle be derived from a

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

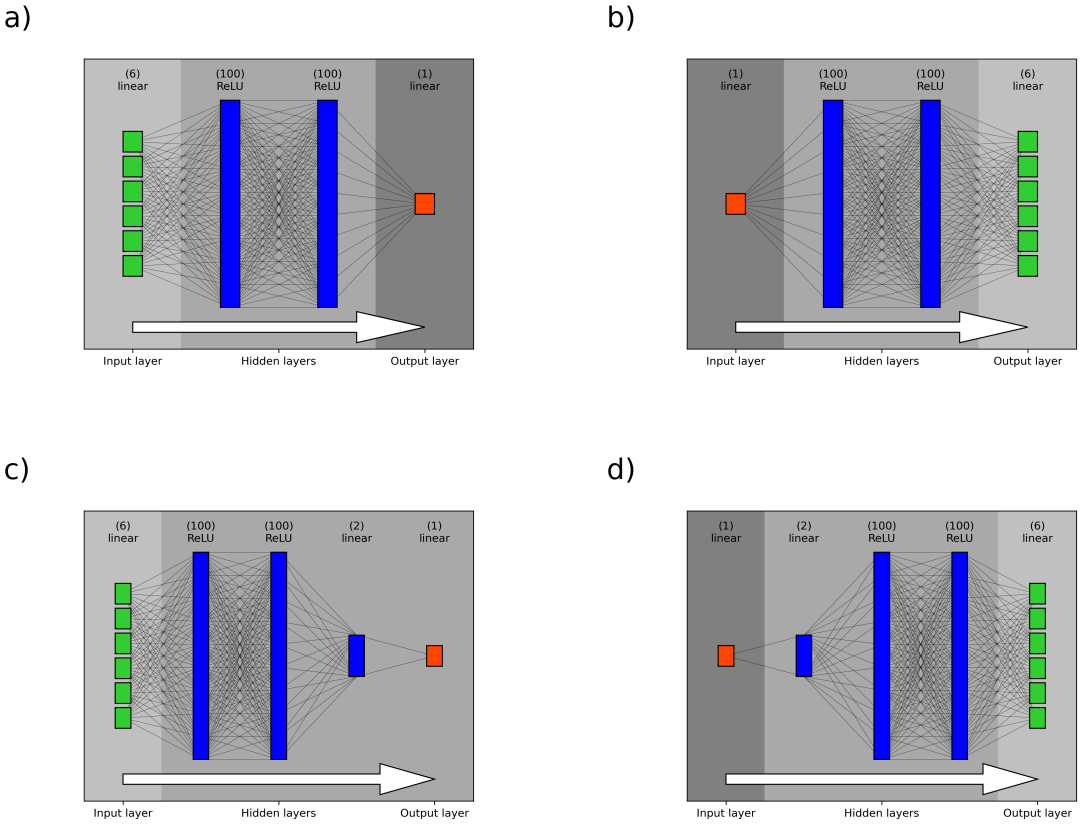

Figure 10. Architecture of the AEs. Panel a, c): Architecture of the regular and probabilistic encoder, respectively. The encoders take the six LLTs as features and returns a 1-D output. Panel b, d): Architecture of the corresponding decoder. The decoders take the 1-D output of the encoder and returns a 6-D output. During training, the decoder output is compared against the encoder input. In all panels the number of neurons shown in the hidden layers is reduced to 25 for visibility reasons.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

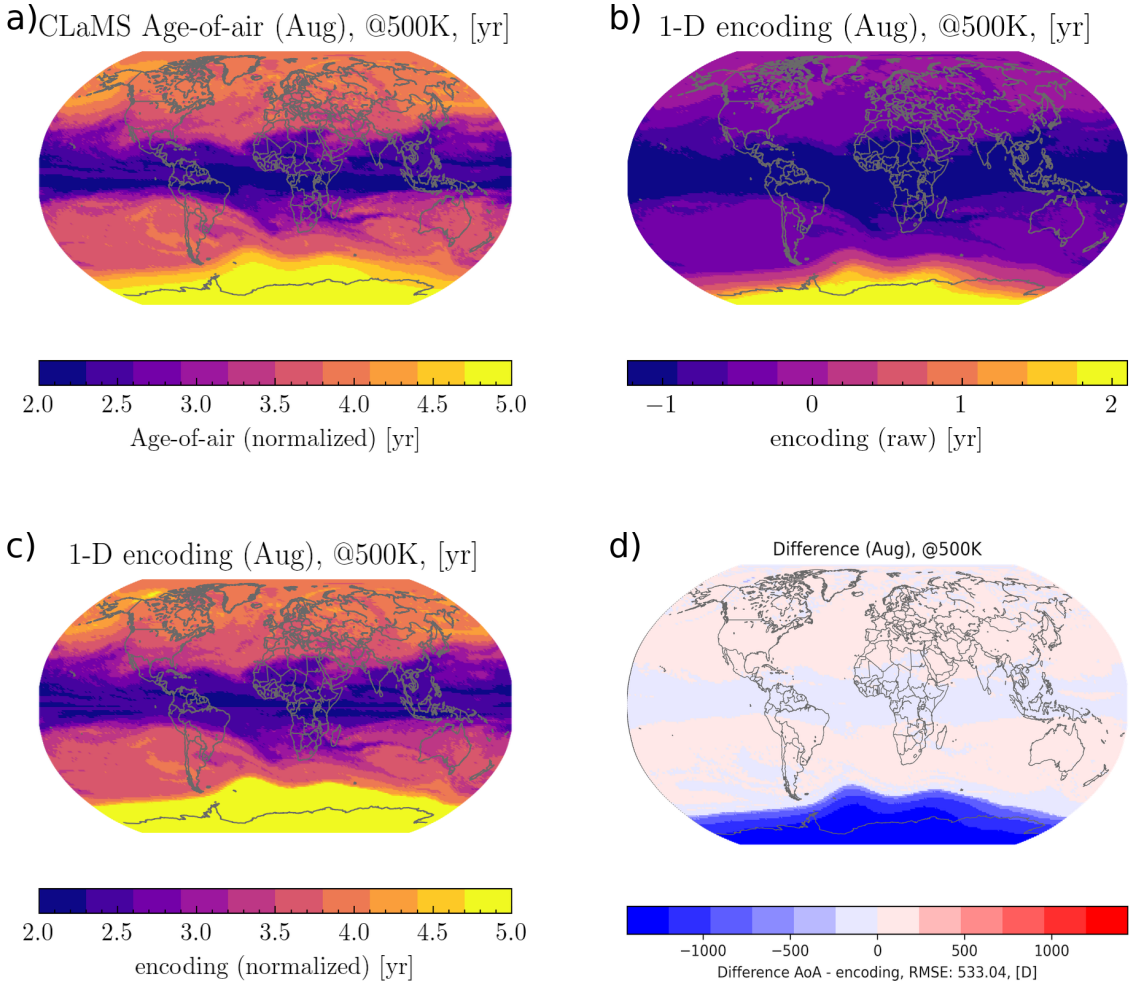

Figure 11. Spatial distribution of a probabilistic AE 1-D embedding of LLTs. Panel a): CLaMS model AoA values. Panel b): 1-D embedding, as provided by the encoder. The scale of the encoding follows the scale of the inputs. Panel c): 1-D embedding, linearly scaled to match the model value range based on the linear regime in Fig. 12a. The color-scale is limited to the value range found in the model AoA. Panel d): Difference between CLaMS model AoA and linearly scaled embedding values.

limited number of AoA values derived from in-situ measurements. The result of this simple calibration is shown in Fig. 11c and its difference to the CLaMS model AoA in Fig. 11d.

The calibrated encoding closely resembles the CLaMS model AoA almost everywhere, with the exception of the antarctic polar vortex, where the encoding massively overestimates the CLaMS model AoA. Particularly for younger ages (AoA < 4 years) the encoding follows even the mesoscale structure of the CLaMS model AoA to a high degree, with an RMSE in that region

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

of around 14.6 days. This strongly hints at a link between the encoding and the CLaMS model AoA. The encoding fails for higher AoA, which one may assume is due to vanishing mixing ratios of many LLTs for these ages. However, comparing the *decoding* of the *encoding*, i.e. the reconstructed LLT inputs, to the original inputs as shown in Fig. 12b, this is, in fact, unlikely a detectable issue. If vanishing mixing ratios would limit the ability to find an accurate encoding, the inverse transformation would likewise fail for high AoA or low mixing ratios. As shown in Fig. 12b this is not the case. While the reconstruction performance is lower towards low mixing ratios, it does not explain the issue with encoding high AoA. The encoding generally depends linearly on AoA except for aforementioned higher AoA, where the encoding is still unique, but follows an unknown relationship with AoA. Given that an AE can only derive information already included within the training inputs it is likely

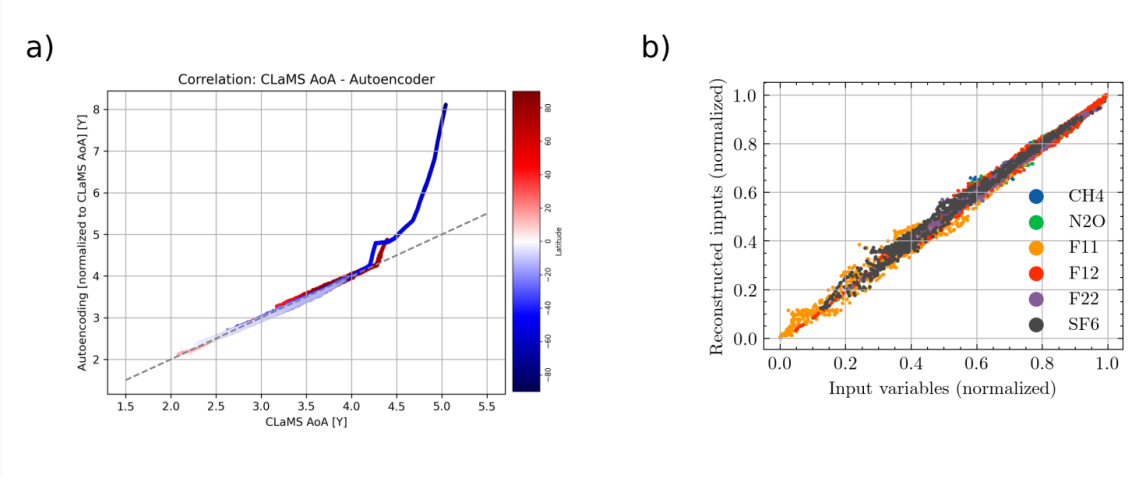

Figure 12. Relations between AE embedding and model values. Panel a): Correlation between CLaMS model AoA and linearly scaled 1-D embedding. The embedding is based on unmodified CLaMS values sampled from at 500 K potential temperature. The latitude of the encoded values is indicated in color-code. The embedding is scaled such that it corresponds to the CLaMS model AoA in the linear regime (< 4 years). Panel b): Correlation between the model LLTs used as features and the decoding of the embedding. The decoded values need no calibration.

that the encoder does not actually map to the transit time Γ, but rather to the apparent AoA τ, i.e. the age which the chemical composition would suggest if sinks are not considered. For low age all LLTs have at most weak stratospheric sinks such that there is a linear relationship between Γ and τ. It shall be noted that the exact strength of these sinks is not actually important at this point. Important is that these sinks are indeed captured by the AE, resulting in the correlation in Fig. 12a. However, without knowledge of this correlation (which requires knowledge of the true AoA) this embedding cannot be trivially translated into AoA. The encoding is therefore a good proxy for AoA for younger air. It can be translated to AoA through a simple linear function which needs to be calibrated. There is one piece of information contained in Fig. 12a which has not been exploited yet. If the relationship between the encoding and AoA would be known in the linear regime, could the non-linear regime be expressed in terms of the linear regime such that a complete calibration would be possible? In other words: if older air

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

apparently ages faster compared to young air, could the aging rate for old air be expressed in terms of the aging rate of younger air?

3.4.2 Advanced calibration

The task is to find a calibration which allows us to translate the encoding of LLTs associated with old AoA ($\Gamma > 4$) to the encoding of LLTs associated with young AoA ($\Gamma < 4$) such that both encodings can be expressed in terms of AoA. Let X be the LLT features and f be the encoding function of an AE. From Fig. 12a we know that for $\Gamma < 4$ the encoding $f(X)$ is linear as a function of Γ (which is in fact an implication of the training process itself). We thus separate X into two parts, namely

$$X_1 := X|_{\Gamma<4\,\mathrm{yr}} \tag{9}$$

$$X_2 := X|_{\Gamma\geq4\,\mathrm{yr}} \tag{10}$$

We can then train two separate instances of an AE on either X_1, X_2, whose encoder functions we denote f_1, f_2. The encoding $f_1(X_1)$ is shown as a function of Γ (which is presumed to be known) in Fig. 13a in dark blue. Expectedly, $\Gamma(X_1)$ is a linear function, which we can write as:

$$\Gamma(X_1) = m \cdot f_1(X_1) + b \tag{11}$$

with parameters m, b that can be derived via a least-square fit from Fig. 13a. Plotting $f_2(X_2)$ against Γ (Fig. 13a, pale blue) we see that $f_2(X_2)$ appears similarly linear in this case. Important here is that $f_2(X_2)$ is monotonous. Since the scales of f_1, f_2 are free and ultimately determined during training, both encoder functions have generally different scales. To express f_2 in terms of f_1 the relationship between the scales of both f_1 and f_2 must be determined, i.e. by comparing the encodings of the same X values. Fig. 13b shows the correlation between $f_1(X_2)$ and $f_2(X_2)$ in light blue. If this correlation function G is also monotonous there exists a transformation by which the encoding of one encoder can be expressed in units of another encoder. We see in Fig. 13b that this is indeed the case. We may thus express $f_2(X_2)$ in terms of f_1 and subsequently translate it to Γ using Eq. 11:

$$\Gamma(X_2) = m \cdot G(f_2(X_2)) + b \tag{12}$$

which is shown in Fig. 13c. This yields the calibrated encoding:

$$\Gamma(X) = \begin{cases} m \cdot f_1(X) + b & \text{for } X|_{\Gamma<4} \\ m \cdot G(f_2(X)) + b & \text{for } X|_{\Gamma\geq4} \end{cases} \tag{13}$$

which is shown in Fig. 13d. The correlation between $f_1(X_2)$ and $f_2(X_2)$ only needs to be monotonous, in which case G is obtained as a numerically tabulated (monotonic) function rather than a linear fit. Due to asymmetries between the hemispheres the process is performed for each hemisphere individually similarly to Voet et al. (2025). Naturally the split in encodings will likely introduce some small discontinuity, which will likely lead to the formation of systematic discrepancies. The calibration

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

thus depends on the linear parameters m, b , which need to be known, and the correlation between the encoders G, which in turn only depends on f_1, f_2 and X_2. The relationship between $f_2(X_2)$ and Γ is not required, and is instead being derived by expressing how one encoding changes compared to another in X_2. The cut-off point when the encoding becomes non-linear in Γ can either be derived from a single encoding (similar to Fig. 12a) or from a previous calibration. In order to become more independent from model information, the cut-off point can alternatively be estimated by comparing the encoding to a limited number of AoA-values derived from in situ measurements.

The spatial distribution of this calibrated encoding is shown in Fig. 14:

Similarly to the simple calibration in Fig. 11b the general spatial structures of AoA are captured in detail. Comparing the scale of the embedding to the model values we find much improved correspondence. There is still a systematic difference between the domains, i.e. between inside and outside of the Antarctic polar vortex, but the encoding now underestimates AoA slightly rather than vastly overestimating it. The largest discrepancy between encoding and CLaMS model AoA barely exceeds three months, and only in the region of the antarctic polar vortex, where the scale between encoding and model AoA was derived by expressing one autoencoder in terms of another. The RMSE for this calibration now lies around 30 days. Some systematic differences remain in the encoded representation, but they are generally not pronounced. Comparing the performance of the calibrated AE with that of the supervised MLPs presented earlier, we find that the AE performs similarly well for the day on which it was trained. The performance of the AE for other months can be similarly assessed as in Sect. 3.2 and is shown in Fig. 15. The performance of the AE is generally worse than a comparable supervised MLP, with average RMSEs between 2.5 and 4 months. While these performances are quite a bit lower, the AE does not require explicit knowledge of AoA beyond the calibration and is only weakly depending on the choice of regularization noise. The dependence on information from atmospheric models during calibration again might even be further reducible by comparing the encoding to a limited number of AoA-values derived from in situ measurements.

Whether the described calibration is universally valid is subject for debate. While there are no assumptions to our knowledge which can be made for AEs in general to guarantee the validity of the described calibration, over many initializations and for multiple different datasets we found the method to be consistent. We suspect that an intrinsic dimensionality of one may suffice if sufficiently many splits in the domain (here: split into four regions (northern/southern hemisphere, polar/non-polar vortex)) are made. Ultimately it should be decided individually for every use case whether the calibration is applicable.

In summary: AEs are unsupervised neural networks which can generate a 1-D representation of LLTs. During training no knowledge of the CLaMS model AoA is provided. The structure of this embedding is similar to the apparent AoA τ and related to the desired transit time Γ, represented in the ideal clock tracer of the CLaMS model AoA. Calibration of this embedding is linear for lower values of AoA. By dividing the encoding domain into hemispheres and low/high AoA the encoding seemingly becomes linear in Γ such that the different encodings can be combined into a single, meaningful estimate for AoA. The calibration can be performed such that only knowledge about the correlation between the encoding of low AoA and the low AoA itself are required. Thus high AoA can be accurately predicted without needing to know CLaMS model AoA for such low LLT values. The performance of this calibrated encoding is comparable to the supervised MLPs of around one month

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

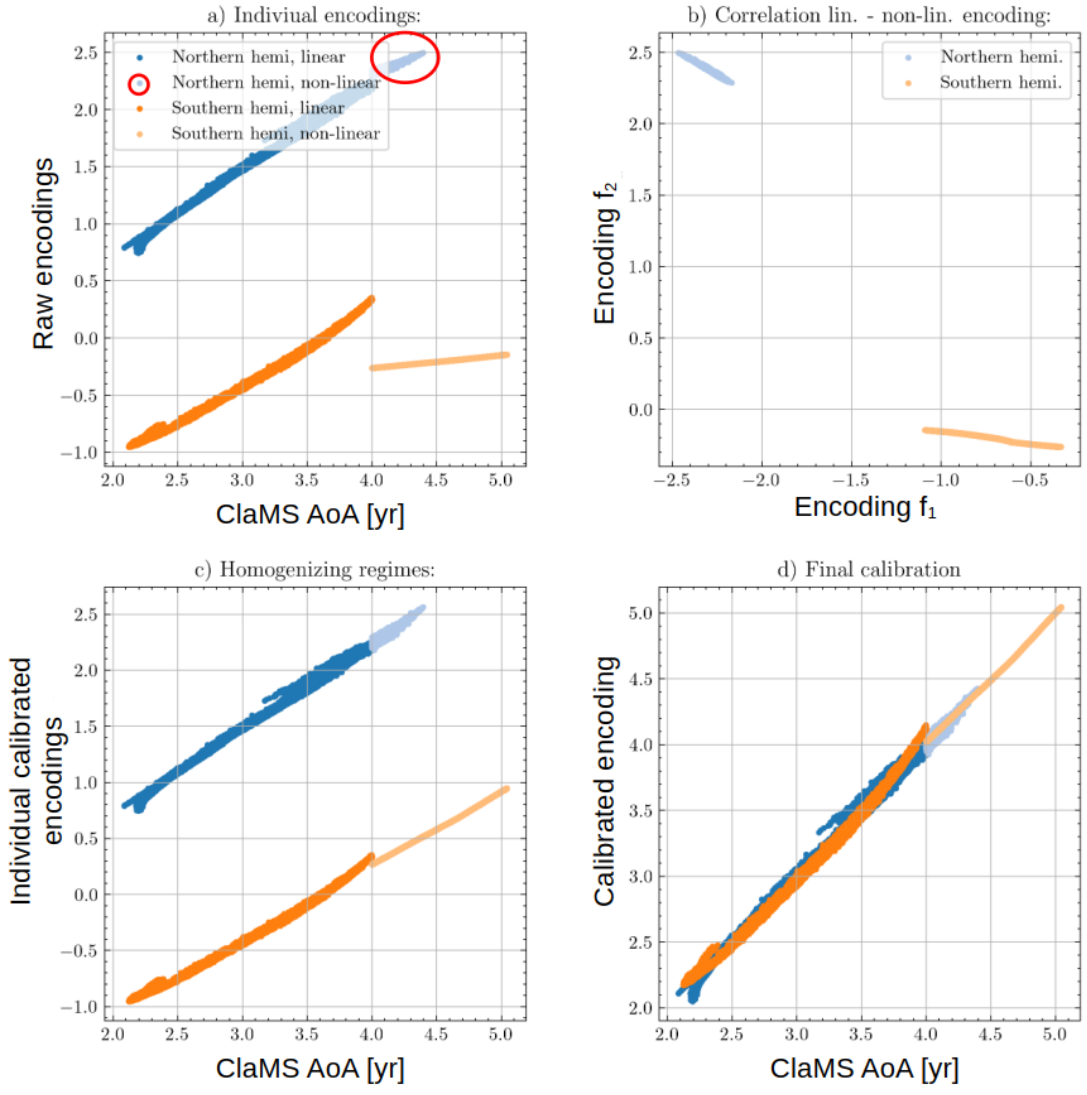

Figure 13. Outline for a calibration pipeline using four AEs. The steps follow from left to right. Panel a): raw, unmodified encodings for AEs trained on data of the respective domain. The scale of each encoding is ultimately free and depends on the individual training. The non-linear encoding of the northern hemisphere is visually highlighted by the red circle. Panel b): Correlation between the encodings of the non-linear domain by the linear encoder (trained on the linear domain) and the non-linear encoder (trained on the non-linear domain), referred in text as G. See Fig. 12a for a definition of either domain. Panel c): Calibrated encodings following Eq.11. The hemispheric split (and subsequent different scales) persist. Panel d): Unified calibration in terms of Γ, i.e removal of hemispheric split. The scales are derived from the linear regimes of either hemisphere.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

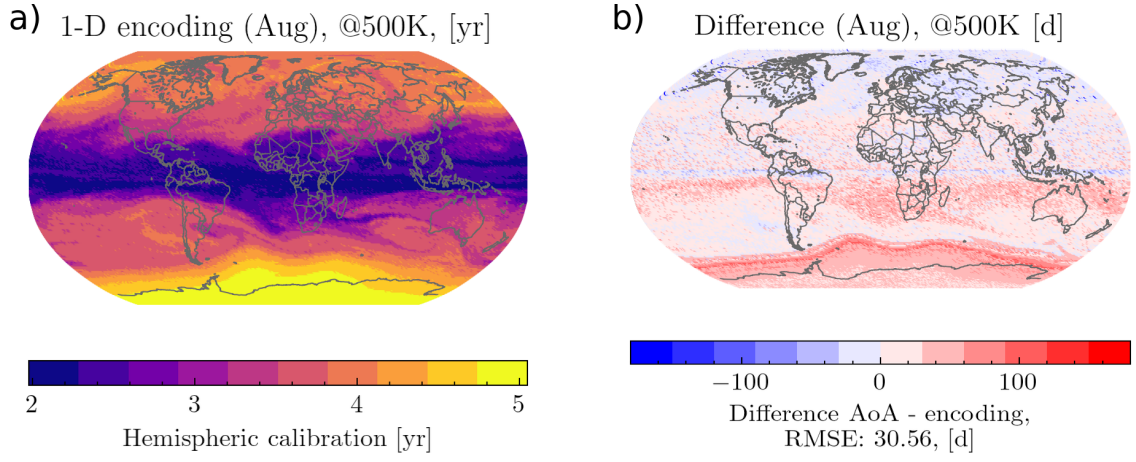

Figure 14. Calibration of AE embedding. Panel a): Spatial distribution of the calibrated embedding. Panel b): Difference between CLaMS model AoA and the calibrated embedding. The color-scale is consistent among similar plots.

outside of the polar vortex and around three months inside of it. Compared to supervised MLPs, the AE has fewer predictive capabilities at the cost of (partial) independence from CLaMS model AoA

4 Summary & Conclusion

The stratospheric AoA is an important quantity for understanding the broader atmospheric circulation, specifically with respect to the Brewer-Dobson circulation (BDC). It can be derived from observable quantities such as long-lived tracers, which act as proxies for an ideal clock tracer. Such methods are well established. It is easiest to obtain AoA through climate models by implementing an ideal clock tracer, but these tracers then describe the model circulation rather than the real one. Discrepancies between CLaMS model AoA and derived AoA are well-known and are increasingly important under a changing climate. While conventional methods require much supporting theory and calibration via measurements, neural networks can be trained to predict AoA from LLTs, which allow for precise and quick predictions. We present neural network architectures in the form of Multi-Layer Perceptrons (MLPs) and Autoencoders (AEs) for this specific task.

We find that given model values an MLP can predict AoA with high accuracy and precision. These MLPs retain high performance even when predicting inputs with large added noise similar to real life measurements. Using probabilistic MLPs allows us to obtain predictive uncertainties for individual predictions. If the MLP is calibrated with noise similar to the measurement noise, the uncertainty estimates are in good agreement with the observed prediction errors, indicating that the model provides well-calibrated estimates of the epistemic uncertainty. A regularization of 1-2% on its training inputs improves the robustness of an MLP when predicting data outside of its training domain, such as data from different months or years.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

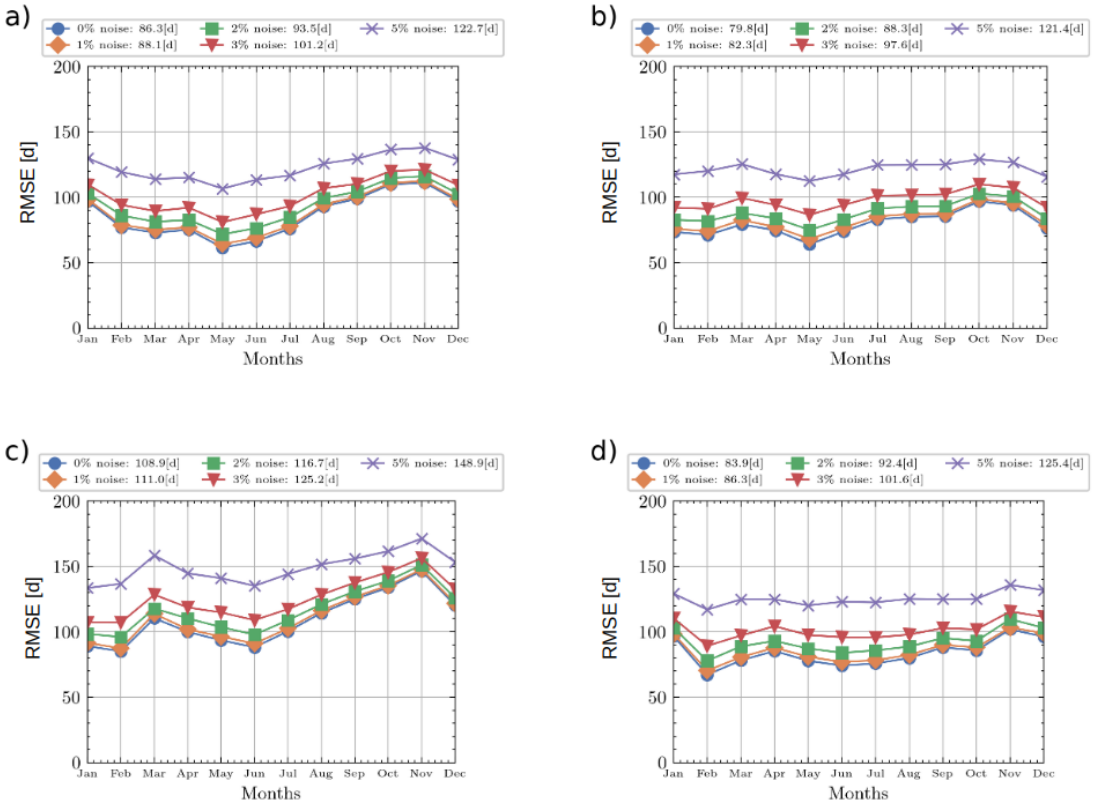

Figure 15. Performances of the probabilistic AE. Panels a - d show the performances of the years 2011, 2012, 2016 and 2021, respectively. Used is the same AE calibrated on the data of the original AE.

The approach of using LLTs can be further improved upon if other features such as spatial information like latitude or temporal information such as a monthly index is included. The monthly index is found to only contribute positively if used in conjunction with latitude. Since MLPs are trained on normalized inputs, they retain high performance even when predicting values a decade after its training data. Seasonal variability can be adequately attributed for if the training domain is expanded conservatively. Such an MLP which is trained extensively on a large amount of data is in principle feasible and could result in a powerful standard tool for AoA prediction. We found that a few distinct datasets could not be accurately predicted, hinting at physical irregularities within these datasets. Further investigation into the reasons is required.

We also introduced non-supervised neural networks in the form of AEs as an alternative approach to AoA determination. Learning from LLT inputs, our AE derived a 1-D embedding of its training inputs, which accurately reflected the mesoscale structures of the CLaMS model AoA, albeit in a random scale. We found this embedding most likely to reflect the apparent

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

AoA rather than the real AoA, which required some form of calibration. We derived a calibration method that only requires knowledge of AoA for those observations where AoA is linearly correlated to the chemical Age-of-Air. The calibration does rely on certain assumptions and unfortunately it must be decided for each AE individually if the calibration is feasible. We do find in our study that the calibration has been valid for many instances of AEs. The resulting predicted AoA is accurate within one month for the day it was trained, and only has accuracy of between 2.5 and 4 months for other datasets.

The calibration could be improved further if the encoding for young AoA-values was compared to a limited number of AoA values derived from in situ measurements (in-situ AoA). This could be done by replacing the true AoA-values of the model environment with in-situ AoA values at the corresponding day and location. It might then be possible to obtain the required parameters for the linear relation between encoding and young AoA values from a linear regression of encoding with respect to in-situ AoA rather than with respect to model-AoA. Given enough in-situ AoA values, even the cut-off value, where the linear relation begins to break down, could be determined independently of the model. Finally, even the model-based input features themselves could be replaced with observations. The upcoming satellite mission Stratosphere–Troposphere Response using Infrared Vertically-resolved light Explorer (Jaeglé et al., 2025, STRIVE), for example, could provide suitable datasets of input features. The logical next step would be to validate the here presented methodology on different climate models or on real observations.

Neural networks remain a powerful tool if deployed in small scale and show promise to advance atmospheric science in the future.

. Code availability

All code was written using the publicly available Python packages Tensorflow. The Python code used in this study is available upon request, with additional details or specific implementations provided as needed.

. Data availability

The CLaMS models are available in the Modular Earth Submodel System (MESSy) Git database at https://gitlab.dkrz.de/users/sign_in (MESSy Consortium, 2025, login required). Detailed information is available at https://messy-interface.org/licence/application (last access: 7th July 2026). The implementation of the CLaMS model into the MESSy framework is available at https://doi.org/10.5194/gmd-7-2639-2014 (Hoppe et al., 2014). ERA5 reanalysis data are available from the European Centre for Medium-Range Weather Forecasts (https://doi.org/10.24381/cds.adbb2d47, Hersbach et al., 2023). The CLaMS model data used for this paper may be requested from the corresponding author (j.kaumanns@fz-juelich.de). The sampled data used for this paper are available at Kaumanns et al. (2026).

. Author contribution

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

JK designed, implemented and tested the various architectures and designed the autoencoder calibration. FV set up and performed the CLaMS simulations. All authors contributed to the manuscript and helped improve it.

. Conflict of interest

The authors declare that there are no competing interests.

. We would like to acknowledge the Global Laboratory of the National Oceanic and Atmospheric Administration (NOOA) for providing the essential mixing ratio time series that served as the lower boundary conditions in the CLaMS simulation. We also would like to acknowledge the MIPAS team for supplying the SF_6 and AoA datasets, which were crucial for initializing the CLaMS simulations and defining the upper boundary condition for SF_6. We would like to express our appreciation to Nicole Thomas and Patrick Jöckel for their support and assistance in setting up the model simulations. Finally, we are deeply grateful for the computing time granted on the supercomputer JUWELS at the Jülich Supercomputing Centre (JSC), provided under the VSR project ID CLAMS-ESM, which was vital for the completion of the CLaMS simulations. Finally, we would like to acknowledge the Tensorflow community for providing the open access Python library, which were essential for creating our neural networks.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X.: TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, https://www.tensorflow.org/, software available from tensorflow.org, 2015.

Brewer, A. W.: Evidence for a World Circulation Provided by the Measurements of Helium and Water Vapour in the Stratosphere, Quart. J. Roy. Meteorol. Soc., 75, 351–363, 1949.

Butchart, N.: The Brewer-Dobson circulation, Rev. Geophys., 52, 157–184, https://doi.org/10.1002/2013RG000448, 2014.

Butchart, N., Scaife, A. A., Bourqui, M., de Grandpre, J., Hare, S. H. E., Kettleborough, J., Langematz, U., Manzini, E., Sassi, F., Shibata, K., Shindell, D., and Sigmond, M.: Simulations of anthropogenic change in the strength of the Brewer-Dobson circulation, Clim. Dyn., 27, 727–741, 2006.

Charlesworth, E. J., Dugstad, A.-K., Fritsch, F., Jöckel, P., and Plöger, F.: Impact of Lagrangian Transport on Lower-Stratospheric Transport Time Scales in a Climate Model, Atmos. Chem. Phys., 20, 15 227–15 245, https://doi.org/10.5194/acp-20-15227-2020, 2020.

Diallo, M., Ern, M., and Ploeger, F.: The advective Brewer–Dobson circulation in the ERA5 reanalysis: climatology, variability, and trends, Atmos. Chem. Phys., 21, 7515–7544, https://doi.org/10.5194/acp-21-7515-2021, 2021.

Engel, A., Möbius, T., Bönisch, H., Schmidt, U., Heinz, R., Levin, I., Atlas, E., Aoki, S., Nakazawa, T., Sugawara, S., Moore, F., Hurst, D., Elkins, J., Schauffler, S., Andrews, A., and Boering, K.: Age of stratospheric air unchanged within uncertainties over the past 30 years, Nature Geoscience, 2, 28–31, https://doi.org/10.1038/ngeo388, 2009.

ESA: Earth Explorer 11 candidate mission CAIRT: report for mission selection, Tech. rep., European Space Agency, ESTEC, Keplerlaan 1, 2200 AG Noordwijk, The Netherlands, https://doi.org/10.5281/zenodo.15606819, 2025.

Foong, A. Y., Burt, D. R., Li, Y., and Turner, R. E.: On the expressiveness of approximate inference in Bayesian neural networks, in: Advances in Neural Information Processing Systems (NeurIPS), vol. 33, pp. 15 897–15 908, 2020.

Garny, H., Eichinger, R., Laube, J. C., Ray, E. A., Stiller, G. P., Bönisch, H., Saunders, L., and Linz, M.: Correction of stratospheric age of air (AoA) derived from sulfur hexafluoride (SF_6) for the effect of chemical sinks, Atmos. Chem. Phys., 24, 4193–4215, https://doi.org/10.5194/acp-24-4193-2024, 2024a.

Garny, H., Ploeger, F., Abalos, M., Bönisch, H., Castillo, A. E., von Clarmann, T., Diallo, M., Engel, A., Laube, J. C., Linz, M., Neu, J. L., Podglajen, A., Ray, E., Rivoire, L., Saunders, L. N., Stiller, G., Voet, F., Wagenhäuser, T., and Walker, K. A.: Age of stratospheric air: progress on processes, observations, and long-term trends, Rev. Geophys., 62, e2023RG000 832, https://doi.org/https://doi.org/10.1029/2023RG000832, 2024b.

Gneiting, T. and Raftery, A. E.: Strictly Proper Scoring Rules, Prediction, and Verification, J. Am. Stat. Assoc., 102, 359–378, https://doi.org/10.1198/016214506000001437, 2007.

Haenel, F., Stiller, G. P., von Clarmann, T., Funke, B., Eckert, E., Glatthor, N., Grabowski, U., Kellmann, S., Kiefer, M., Linden, A., and Reddmann, T.: Reassessment of MIPAS age of air trends and variability, Atmos. Chem. Phys., 22, 13 161–13 176, https://doi.org/10.5194/acp-15-13161-2015, 2015.

Hall, T. M. and Plumb, R. A.: Age as a diagnostic of stratospheric transport, J. Geophys. Res., 99, 1059–1070, 1994.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

Hegglin, M. I., Plummer, D. A., Shepherd, T. G., Scinocca, J. F., Anderson, J., Froidevaux, L., Funke, B., Hurst, D., Rozanov, A., Urban, J., von Clarmann, T., A.Walker, K., Wang, H. J., Tegtmeier, S., and Weigel, K.: Vertical structure of stratospheric water vapour trends derived from merged satellite data, Nature Geosci., 7, 768–776, https://doi.org/10.1038/NGEO2236, 2014.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horànyi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., Simmons, A., Soci, C., Abdalla, S., Abellan, X., Balsamo, G., Bechtold, P., Biavati, G., Bidlot, J., Bonavita, M., De Chiara, G., Dahlgren, P., Dee, D., Diamantakis, M., Dragani, R., Flemming, J., Forbes, R., Fuentes, M., Geer, A., Haimberger, L., Healy, S., Hogan, R. J., Hólm, E., Janisková, M., Keeley, S., Laloyaux, P., Lopez, P., Lupu, C., Radnoti, G., de Rosnay, P., Rozum, I., Vamborg, F., Villaume, S., and Thépaut, J.-N.: The ERA5 global reanalysis, Q. J. R. Meteorol. Soc., 146, 1999–2049, https://doi.org/10.1002/qj.3803, 2020.

Holton, J. R., Haynes, P. H., McIntyre, M. E., Douglass, A. R., Rood, R. B., and Pfister, L.: Stratosphere-Troposphere Exchange, Rev. Geophys., 33, 403–439, 1995.

Hoppe, C. M., Hoffmann, L., Konopka, P., Grooß, J.-U., Ploeger, F., Günther, G., Jöckel, P., and Müller, R.: The implementation of the CLaMS Lagrangian transport core into the chemistry climate model EMAC 2.40.1: application on age of air and transport of long-lived trace species, Geosci. Model Dev., 7, 2639–2651, https://doi.org/10.5194/gmd-7-2639-2014, 2014.

Jaeglé, L., Wang, J., Oman, L., DeLand, M., and Team, S. S.: The STRIVE Earth System Explorer Mission Concept, EGU General Assembly 2025, presentation EGU25-7598, https://doi.org/10.5194/egusphere-egu25-7598, 2025.

Johnson, D. G., Jucks, K. W., Traub, W. A., Chance, K. V., Toon, G. C., Russell III, J. M., and McCormick, M. P.: Stratospheric age spectra derived from observations of water vapor and methane, J. Geophys. Res., 104, 21 595, https://doi.org/10.1029/1999JD900363, 1999.

Kaumanns, J., Voet, F., Ploeger, F., Preusse, P., Ungermann, J., and Hegglin, M. I.: https://doi.org/10.5281/zenodo.21333750, 2026.

Kendall, A. and Gal, Y.: What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?, in: Advances in Neural Information Processing Systems (NeurIPS), vol. 30, pp. 5574–5584, Curran Associates, Inc., 2017.

Kirsch, A.: (Implicit) Ensembles of Ensembles: Epistemic Uncertainty Collapse in Large Models, Trans. Mach. Learn. Res., https://arxiv.org/abs/2409.02628, 2025.

Konopka, P., Tao, M., Ploeger, F., Diallo, M., and Riese, M.: Tropospheric mixing and parametrization of unresolved convective updrafts as implemented in the Chemical Lagrangian Model of the Stratosphere (CLaMS v2.0), Geosci. Model Dev., 12, 2441–2462, https://doi.org/10.5194/gmd-12-2441-2019, 2019.

Kouznetsov, R., Sofiev, M., Vira, J., and Stiller, G.: Simulating age of air and the distribution of SF6 in the stratosphere with the SILAM model, Atmos. Chem. Phys., 20, 5837–5859, https://doi.org/10.5194/acp-20-5837-2020, 2020.

Laube, J. C., Elvidge, E. C. L., Adcock, K. E., Baier, B., Brenninkmeijer, C. A. M., Chen, H., Droste, E. S., Grooß, J.-U., Heikkinen, P., Hind, A. J., Kivi, R., Lojko, A., Montzka, S. A., Oram, D. E., Randall, S., Röckmann, T., Sturges, W. T., Sweeney, C., Thomas, M., Tuffnell, E., and Ploeger, F.: Investigating stratospheric changes between 2009 and 2018 with halogenated trace gas data from aircraft, AirCores, and a global model focusing on CFC-11, Atmos. Chem. Phys., 20, 9771–9782, https://doi.org/10.5194/acp-20-9771-2020, 2020.

Lin, P. and Fu, Q.: Changes in various branches of the Brewer–Dobson circulation from an ensemble of chemistry climate models, J. Geophys. Res., 118, 73–84, https://doi.org/https://doi.org/10.1029/2012JD018813, 2013.

Loeffel, S., Eichinger, R., Garny, H., Reddmann, T., Fritsch, F., Versick, S., Stiller, G., and Haenel, F.: The impact of sulfur hexafluoride (SF6) sinks on age of air climatologies and trends, Atmos. Chem. Phys., 22, 1175–1193, https://doi.org/10.5194/acp-22-1175-2022, 2022.

Maiss, M., Steele, L., Francey, R. J., Fraser, P. J., Langenfelds, R. L., Trivett, N. B., and Levin, I.: Sulfur hexafluoride — A powerful new atmospheric tracer, Atmos. Environment, 30, 1621–1629, https://doi.org/https://doi.org/10.1016/1352-2310(95)00425-4, 1996.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

McKenna, D. S., Grooß, J.-U., Günther, G., Konopka, P., Müller, R., Carver, G., and Sasano, Y.: A new Chemical Lagrangian Model of the Stratosphere (CLaMS) 2. Formulation of chemistry scheme and initialization, J. Geophys. Res., 107, https://doi.org/10.1029/2000JD000113, 2002a.

McKenna, D. S., Konopka, P., Grooß, J.-U., Günther, G., Müller, R., Spang, R., Offermann, D., and Orsolini, Y.: A new Chemical Lagrangian Model of the Stratosphere (CLaMS) 1. Formulation of advection and mixing, J. Geophys. Res., 107, https://doi.org/10.1029/2000JD000114, 2002b.

McLandress, C. and Shepherd, T. G.: Simulated anthropogenic changes in the Brewer-Dobson circulation, including its extension to high latitudes, J. Clim., 22, 1516–1540, https://doi.org/10.1175/2008JCLI2679.1, 2009.

Nix, D. A. and Weigend, A. S.: Estimating the mean and variance of target distributions at the same time, in: Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN), vol. 6, pp. 3981–3985, https://doi.org/10.1109/ICNN.1994.341681, 1994.

Ovadia, Y., Fertig, E., Ren, J., Lakshminarayanan, B., Nowozin, S., Sculley, D., and Snoek, J.: Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift, in: Advances in Neural Information Processing Systems (NeurIPS), vol. 32, 2019.

Ploeger, F., Legras, B., Charlesworth, E., Yan, X., Diallo, M., Konopka, P., Birner, T., Tao, M., Engel, A., and Riese, M.: How robust are stratospheric age of air trends from different reanalyses?, Atmos. Chem. Phys., 19, 6085–6105, https://doi.org/10.5194/gmd-12-2441-2019, 2019.

Ploeger, F., Diallo, M., Charlesworth, E., Konopka, P., Legras, B., Laube, J. C., Grooß, J.-U., Günther, G., Engel, A., and Riese, M.: The stratospheric Brewer–Dobson circulation inferred from age of air in the ERA5 reanalysis, Atmos. Chem. Phys., 21, 8393–8412, https://doi.org/10.5194/acp-21-8393-2021, 2021.

Pommrich, R., Müller, R., Grooß, J.-U., Konopka, P., Ploeger, F., Vogel, B., Tao, M., Hoppe, C. M., Günther, G., Spelten, N., Hoffmann, L., Pumphrey, H.-C., Viciani, S., D'Amato, F., Volk, C. M., Hoor, P., Schlager, H., and Riese, M.: Tropical troposphere to stratosphere transport of carbon monoxide and long-lived trace species in the Chemical Lagrangian Model of the Stratosphere (CLaMS), Geosci. Model Dev., 7, 2895–2916, https://doi.org/10.5194/gmd-7-2895-2014, 2014.

Poshyvailo, L., Müller, R., Konopka, P., Günther, G., Riese, M., Podglajen, A., and Ploeger, F.: Sensitivities of modelled water vapour in the lower stratosphere: temperature uncertainty, effects of horizontal transport and small-scale mixing, Atmos. Chem. Phys., 18, 8505–8527, https://doi.org/10.5194/acp-18-8505-2018, 2018.

Poshyvailo-Strube, L., Müller, R., Fueglistaler, S., Hegglin, M. I., Laube, J. C., Volk, C. M., and Ploeger, F.: How can Brewer–Dobson circulation trends be estimated from changes in stratospheric water vapour and methane?, Atmos. Chem. Phys., 22, 9895–9914, https://doi.org/10.5194/acp-22-9895-2022, 2022.

Ray, E. A., Moore, F. L., Elkins, J. W., Rosenlof, K. H., Laube, J. C., Röckmann, T., Marsh, D. R., and Andrews, A. E.: Quantification of the SF_6 lifetime based on mesospheric loss measured in the stratospheric polar vortex, J. Geophys. Res. Atmos., 122, 4626–4638, https://doi.org/10.1002/2016JD026198, 2017.

Ray, E. A., Moore, F. L., Garny, H., Hintsa, E. J., Hall, B. D., Dutton, G. S., Nance, D., Elkins, J. W., Wofsy, S. C., Pittman, J., Daube, B., Baier, B. C., Li, J., and Sweeney, C.: Age of air from in situ trace gas measurements: insights from a new technique, Atmos. Chem. Phys., 24, 12 425–12 445, https://doi.org/10.5194/acp-24-12425-2024, 2024.

Stiller, G. P., von Clarmann, T., Höpfner, M., Glatthor, N., Grabowski, U., Kellmann, S., Kleinert, A., Linden, A., Milz, M., Reddmann, T., Steck, T., Fischer, H., Funke, B., López-Puertas, M., and Engel, A.: Global distribution of mean age of stratospheric air from MIPAS SF_6 measurements, Atmos. Chem. Phys., 8, 677–695, 2008.

https://doi.org/10.5194/egusphere-2026-4183
Preprint. Discussion started: 30 July 2026
© Author(s) 2026. CC BY 4.0 License.

Stiller, G. P., von Clarmann, T., Haenel, F., Funke, B., Glatthor, N., Grabowski, U., Kellmann, S., Kiefer, M., Linden, A., Lossow, S., and López-Puertas, M.: Observed temporal evolution of global mean age of stratospheric air for the 2002 to 2010 period, Atmos. Chem. Phys., 12, 3311–3331, https://doi.org/10.5194/acp-12-3311-2012, 2012.

Stiller, G. P., Fierli, F., Ploeger, F., Cagnazzo, C., Funke, B., Haenel, F., Reddmann, T., Riese, M., and von Clarmann, T.: Shift of subtropical transport barriers explains observed hemispheric asymmetry of decadal trends of age of air, Atmos. Chem. Phys., 17, 11 177–11 192, 2017.

Vervalcke, S., Errera, Q., Eichinger, R., Reddmann, T., Chabrillat, S., Op de beeck, M., Stiller, G., and Mahieu, E.: Atmospheric lifetime of sulphur hexafluoride (SF_6) and five other trace gases in the BASCOE model driven by three reanalyses, Atmos. Chem. Phys., 26, 391–409, https://doi.org/10.5194/acp-26-391-2026, 2026.

Voet, F., Ploeger, F., Laube, J., Preusse, P., Konopka, P., Grooß, J.-U., Ungermann, J., Sinnhuber, B.-M., Höpfner, M., Funke, B., Wetzel, G., Johansson, S., Stiller, G., Ray, E., and Hegglin, M. I.: On the estimation of stratospheric age of air from correlations of multiple trace gases, Atmos. Chem. Phys., 25, 3541–3565, https://doi.org/10.5194/acp-25-3541-2025, 2025.

Volk, C. M., Elkins, J. W., Fahey, D. W., Dutton, G. S., Gilligan, J. M., Loewenstein, M., Podolske, J. R., Chan, K. R., and Gunson, M. R.: Evaluation of source gas lifetimes from stratospheric observations, J. Geophys. Res., 102, 25 543–25 564, 1997.

Wang, X.: A scale-based approach to finding effective dimensionality in manifold learning, Electronic Journal of Statistics, 2, 127–148, 2008.

Waugh, D. W.: The age of stratospheric air, Nature Geoscience, 2, 14–16, 2009.