Robustness and reproducibility for data-driven weather forecasting

DE_398_EVIDEN-DE-Ethical-Machine-Learning

Issued by: Eviden / Author names: Sebastian Schmidt, Maram Akila, Afshin Sadeghi, Erik Prjadka, Djordje Benn-Maksimovic

Date: 02/02/2026

Ref: DE_398_D398.2.5.1_202503_Mandatory_White_Paper_2_v1.3.docx

Official reference number service contract: 2024/DE_398_EVIDEN

Status: Public

Funded by the European Union Destination Earth implemented by ECMWF esa EUMETSAT

This document has been produced in the context of the Destination Earth Initiative and relates to tasks entrusted by the European Union to the European Centre for Medium-Range Weather Forecasts implementing part of this Initiative.

This document is funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them. The European Centre for Medium-Range Weather Forecasts is not liable in respect of this document and gives no warranty for the information provided.

Change Log

Version	Date	Description
0.1	31.01.2025	Document Structure for Expert Briefing
1.0	17.03.2025	Version 1.0.
1.1	31.03.2025	Incorporated reviews as well as feedback from ECMWF experts
1.2	16.05.2025	Submitted version
1.3	12.12.2025	Updated references to other whitepapers of this series
1.4	02.02.2026	Minor corrections

Reviewers

Name	Organisation	Expertise
Anna Schmitz	Fraunhofer IAIS	Expert and PhD candidate for Trustworthy AI
Rebekka Görge	Fraunhofer IAIS	Expert and PhD candidate for Trustworthy AI
Maximilian Pintz	Fraunhofer IAIS	Expert and PhD candidate for Trustworthy AI

Contents

Glossary

AI	Artificial Intelligence
Aleatoric Uncertainty	Uncertainty from inherent randomness in the data
AIFS	Artificial Intelligence/Integrated Forecasting System
ECMWF	European Centre for Medium-Range Weather Forecasts
Ensemble Model	Method for Uncertainty quantification and improvement of accuracy combining multiple predictions
Epistemic Uncertainty	Uncertainty from lack of knowledge in modelling
EU	European Union
IFS	Integrated Forecasting System
ML	Machine Learning
NWP	Numerical Weather Prediction
RMSE	Root Mean Square Error
Robustification	Optimization to make a system more robust

1 Executive Summary

Data-driven models have shown great promise in weather and climate forecasting and are already being used to support traditional physics-based modelling. Their use introduces a range of new questions concerning the trustworthiness of forecasting systems accompanying new technical challenges such as increased dependency on (training) data. In this whitepaper we examine challenges to robustness and reproducibility in this context of data-driven weather forecasting.

The stronger dependency on data also deepens previously existing challenges regarding the uncertainty of predictions. While for physics-based models typically stemming from inherent unpredictability of weather dynamics and pre-existing challenges of accurately reflecting limited sensor data in a global model, it is now enhanced by further uncertainty stemming from ML-based modelling. This is linked to the robustness and replicability of the models, which was previously ensured by significant progress in numerical computations to allow perfect reproduction of outcomes and a direct physical grounding of parameter ranges and model design. Both are no longer given for machine learning. Despite these challenges, AI also offers new venues to address them, for instance computationally cheaper approximations to uncertainty distributions.

Central to this whitepaper is the discussion of robustness and reproducibility challenges along the AI lifecycle to precisely pinpoint where issues can arise, how they can be mitigated and what the current topics of research are.

This whitepaper further serves to give an overview of current strengths and weaknesses of data-driven forecasting in relation to robustness and reproducibility, presenting a structured approach to robustness from data to operation.

2 Introduction

Robustness is widely regarded as one of the cornerstones of Trustworthy AI [1]. The opening whitepaper of this series, where we have shed light on ethical questions regarding the adoption of data-driven weather forecasting and downstream systems, has reiterated this stance. Similar considerations have also led to robustness obligations for high-risk systems being included in the EU AI Act:

> *"High-risk AI systems shall be designed and developed in such a way that they achieve an appropriate level of accuracy, robustness, and cybersecurity, and that they perform consistently in those respects throughout their lifecycle"* (Art. 15.1).

While meteorological data-driven systems are not high-risk AI systems, these requirements can still guide voluntary codes of conduct for their development. Two aspects of this quote are particularly noteworthy: Firstly, robustness should be considered *by design* and addressed throughout the development process and secondly, the performance in those respects is important throughout the entire AI lifecycle. This mandates a close look at the different stages of the AI lifecycle which therefore serves as the underlying structure of this whitepaper.

To understand what these requirements entail, switching to a more technical perspective is necessary. Typically, robustness refers to model performance under changed conditions, for instance due to perturbations in initial conditions or parameters. This notion is naturally linked to an AI model's ability to generalize i.e., to perform well on previously unseen data, where it may need to withstand errors or attacks. From this angle, robustness determines if these models are applicable in a more general context outside the limitations of the training data or whether they simply memorize correlations for the dataset they are trained on. While physical descriptions e.g. of the atmospheric dynamics are applicable across different climate zones, this is a priori an open question for data-driven models, especially if they are trained on data with differing regional coverage. Similarly, atmospheric changes caused by climate change not represented in (historical) training data pose another challenge for model generalization. Whether data-driven weather and climate forecasting models can learn the underlying physical patterns, e.g. atmospheric dynamics, is a fundamental question for assessing model capabilities to transfer to unseen scenarios. Understanding and quantifying the robustness of a weather forecast is usually associated with uncertainty quantification. AI systems offer a lot of promise for facilitating and accelerating ensemble forecasts used for uncertainty quantification, which can also supplement NWP-models. Communicating resulting uncertainties is a challenge of its own with a wide reach – the communicated estimates can have great impact on economic decisions across different sectors.

Another focus of this whitepaper is the reproducibility of AI systems and their output, which also reflects in part the requirement of consistent performance in the above quote. This aspect captures robustness in view of the high degree of inherent stochasticity in ML and sometimes model hardware dependence. For example, bit-reproducibility, i.e. creating exactly the same output on multiple runs with the same input, represents perfect robustness in terms of absolutely consistent and deterministic

model behaviour. It is particularly desirable, because it facilitates model development and bug fixing and lightens the necessity of having to store model outputs for later use. On the other hand, the inherent stochasticity many types of models exhibit limits the possibility of achieving such bit-reproducibility. Even stronger notions than bit-reproducibility in a fixed setting could include exact reproduction across different compilers and hardware.

A trade-off is encountered here, regarding what degree of reproducibility is optimal regarding the trade-off between runtime performance and complexity of development and reproducibility. The question therefore becomes one of robustness to model stochasticity and hardware dependence as well as metrics and acceptance criteria to measure and set thresholds on this type of robustness.

This leads to another consideration – from an ethical point of view, robustness and reproducibility are key components for trust from both an internal and an external perspective. Reproducibility plays a particular role in the publication of scientific results, where it is part of good scientific practice for researchers to be able to reproduce and thereby verify crucial findings. This is not necessarily a question of bit-reproducibility but is rather linked to transparency in research.

This whitepaper is organized as follows. Section 3 introduces fundamental terminology and concepts for the discussion of robustness in AI systems. This includes definitions of robustness and reproducibility (Section 3.1), methods for uncertainty quantification (Section 3.2), as well as the distinct stages of the AI lifecycle (Section 3.3). We discuss these challenges and mitigations to them along the lifecycle stages Data Acquisition and Preparation (Section 4), Model Design and Training (Section 5) and Deployment and Operation (Section 6).

Section 4 covers topics in data collection, curation and transformation. A focus lies on challenges brought by data sparsity and bias and mitigations during data assimilation (i.e. transforming observations to grid-based data to be used for model initialization). Mitigations include practices for outlier detection, synthetic data augmentation and recent innovations to data-driven procedures.

Section 5 focuses on model development, including both aspects to consider when designing model architectures, such as reproducibility, and challenges arising from stochastic algorithms and challenges in model training, such as overfitting. Further, an in-depth look into uncertainty quantification and its relationship to model robustness is given with an introduction to different approaches in UQ.

Section 6 lastly focuses on challenges in model evaluation, integration into existing software, where new reproducibility issues can arise, and hardware frameworks and model operation, such as performance degradation due to drift. Mitigations and practices for validation and integration are presented and drift monitoring is offered as a way to deal with performance degradation.

While we mainly consider a technical perspective along the lifecycle, we offer a complementary, interaction-centred view in Section 7. First, we consider processes, where we address frameworks and structures to address robustness from a process-oriented perspective (7.1) along with international standardization (7.2) and lastly address the value of reproducibility for trust in research (7.3).

3 Robustness along the AI Lifecycle

Before addressing robustness challenges and mitigations along the AI lifecycle, this section serves to give an overview of what robustness and reproducibility entail. We will broadly introduce uncertainty and assessments of uncertainty as a prerequisite to deep dives in later sections. Finally, we will motivate and describe the AI lifecycle as the guiding element of this whitepaper.

3.1 Terminology for Robustness and Reproducibility

Commonly robustness of an AI system refers to the ability to maintain performance under altered conditions. The most common examples are naturally arising [2] or adversarial perturbations [3] of the input data used for inference. Adversarial manipulations are the topic of an accompanying whitepaper and therefore only briefly mentioned here.

It is important to note the difference to simply considering accuracy. Systems might be highly accurate in terms of performance metrics, but unstable to changes in surrounding conditions, which might cause their performance to drop.

The prototypical example is robustness to variations in initial conditions. While in other ML-applications one can validate robustness to errors or small changes in initial conditions, in weather forecasting, it is often expected that small perturbations of input variables lead to larger changes in the output, especially for medium and long-range forecasting. Many of the systems in question describing the physical dynamics of weather are inherently chaotic in the sense that small perturbations to input variables can lead to larger changes along the forecast horizon, especially for medium and long-range forecasting. This shifts the focus from optimizing model and parameters until a certain robustness radius bound is achieved to trying to quantify inherent aleatoric uncertainty (see Section 3.2) for weather forecasting, underpinning the status of ensemble uncertainty quantification in the field. Robustness is only desirable as long as it does not restrict the model to accurately capture potentially unstable dynamics.

In addition to the common example above, the notion of robustness also applies to other alterations of conditions: Operating conditions, i.e. different operating environments, incomplete or biased data, robustness to the inherent stochasticity of the model, or changes in the distributions ought to be learned by the models. In this wider sense, when speaking of robustness in ML, it is important to specify what robustness refers to in specific cases.

The notion of reproducibility is similarly broad. For physical-models, there has been a debate about whether it is optimal to aim for bit-reproducibility, i.e. the exact reproducibility of the outcomes. Generally, bit-reproducibility has many advantages: It simplifies debugging and allows to reproduce data that would otherwise take up resources if it were stored. The latter is a point of debate in the weather forecasting community, as for NWP-based models the computational cost of rerunning models is relatively high, whereas for data driven models it could be desirable in principle. However, for data-driven models, achieving reproducibility is more difficult. Different sources of randomness come into play and randomized algorithms often outperform deterministic ones. This motivates a closer look at the weaker notions of reproducibility. One option would be to consider reproducibility of the findings, i.e.

while the outcomes do not have to be exactly equal, they should detect similar patterns and reproduce them, for example event onsets predicted by another model run.

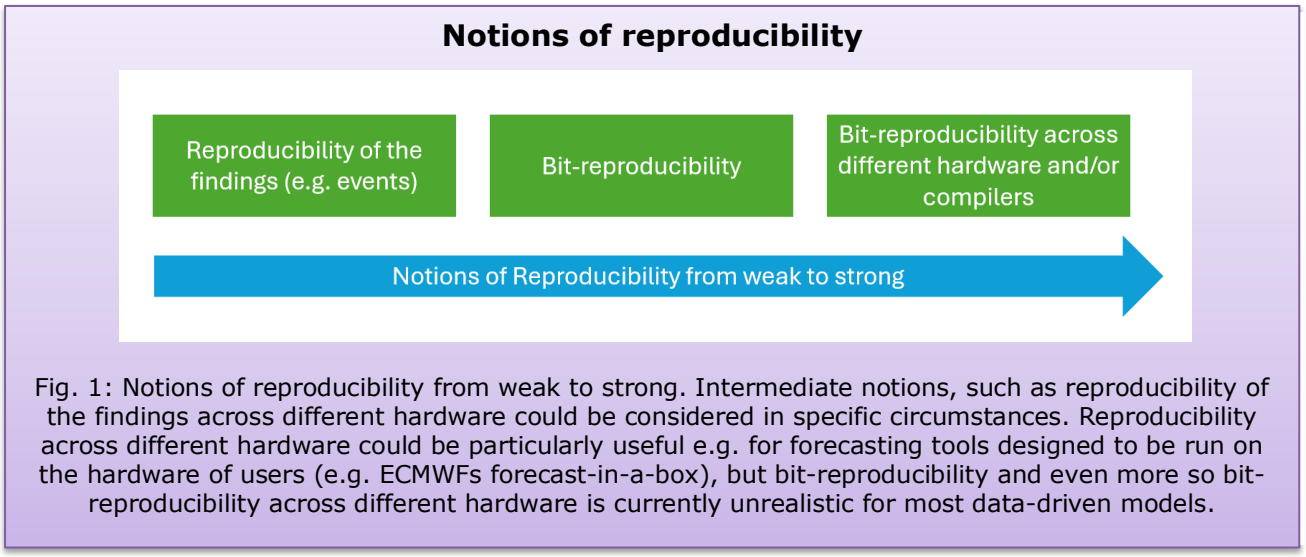

Fig. 1: Notions of reproducibility from weak to strong. Intermediate notions, such as reproducibility of the findings across different hardware could be considered in specific circumstances. Reproducibility across different hardware could be particularly useful e.g. for forecasting tools designed to be run on the hardware of users (e.g. ECMWFs forecast-in-a-box), but bit-reproducibility and even more so bit-reproducibility across different hardware is currently unrealistic for most data-driven models.

3.2 Uncertainty Quantification

When speaking of uncertainty, one commonly distinguishes between aleatoric and epistemic uncertainty. Aleatoric uncertainty refers to the inherent and thereby irreducible randomness of the processes underlying the forecasts, e.g. atmospheric variances or sensor noise. Epistemic uncertainty on the other hand arises through a lack of knowledge by the model, for example out-of-distribution inputs in inference, i.e. inputs in inference beyond the range of inputs for which the model was trained – high epistemic uncertainty could in such instances be fixed by retraining the model on a larger corpus of training data[1].

Measuring uncertainty of predictions can give an indication of model robustness, particularly in inference. Ensembles have been a mainstay in weather forecasting, including physics-based, hybrid and recently data-driven forecasting.

Multiple runs with different perturbations of initial conditions and parameters are meant to account for one or both types of uncertainty by gaining understanding on how differences in initial data or parameters propagate through the model, resulting in an ensemble of multiple varying forecasts. The greater the spread between ensemble members, the greater is the predicted uncertainty. It is thereby desirable to have the ensemble member spread to be approximately equal to the mean error of the ensemble.

[1] An example outside of weather forecasting would be an image recognition system in autonomous driving which has never seen an electric scooter in its training data due to their rapid emergence on the streets. Naturally the epistemic uncertainty regarding classification of this unseen object is high, including them in training data can however mitigate this issue.

Ensembles using NWP-models are typically expensive and runtime intensive (since NWP-predictions in general have high resource requirements and ensembles involve many runs of such models), whereas data-driven models are typically faster and less resource-demanding. This has led to an effort to try to replace NWP-based uncertainty quantification components by data-driven ensembles even in several NWP-based hybrid models, presenting an exemplary case for the promising future of data-driven forecasting.

In general, there are multiple approaches to uncertainty quantification for data-driven forecasting beyond ensembling, for which we will give an overview with advantages and disadvantages in Section 5, where we will also address the disentanglement of the two types of uncertainties (aleatoric and epistemic).

3.3 Stages of the AI lifecycle

Structuring the life cycle of AI systems in stages has become a ubiquitous methodology for AI developers across the globe. Originally introduced already in 1999 for data mining, CRISP-DM has become a foundational paradigm, on which many current process models for AI systems are building [4]. It details six phases from business understanding to deployment. While at the time it was common to see development and operation of IT systems as separate tasks, the faster development cycles and updates of online available software, especially Software-as-a-Service (SaaS), have led to a union of both development and operations into a joined DevOps paradigm around 2009. This was reflected specifically for data-driven systems AI by Machine Learning Operations (MLOps) around 2015 [5], which extends the development of AI systems by considerations on their operation. Specifically, MLOps focuses on update cycles, re-training (AI specific) and quality controls in short automation-assisted intervals, which has roughly been included in DevOps under the heading of Continuous Integration / Continuous Deployment (CI/CD). Generally, while MLOps is a set of practices focused around scalable and automated model deployment and operation, CRISP-DM gives a structured approach for the relevant considerations in data processing, starting from business understanding. In both paradigms, interactions between phases are present, e.g. when learnings during the development process inform needs for new data or when detected changes in the underlying data distributions trigger a relearning of the model.

Depending on the specific variant of the process model considered, the precise number of stages or sub-stages as well as their desired and expected outcomes may differ, for instance see [6] for an overview. For the purposes of this whitepaper, we will therefore consider a simplified version, adopted from the Fraunhofer IAIS AI Assessment Catalogue, which considers “Data”, “Development”, and “Operation” as different stages with separate challenges.

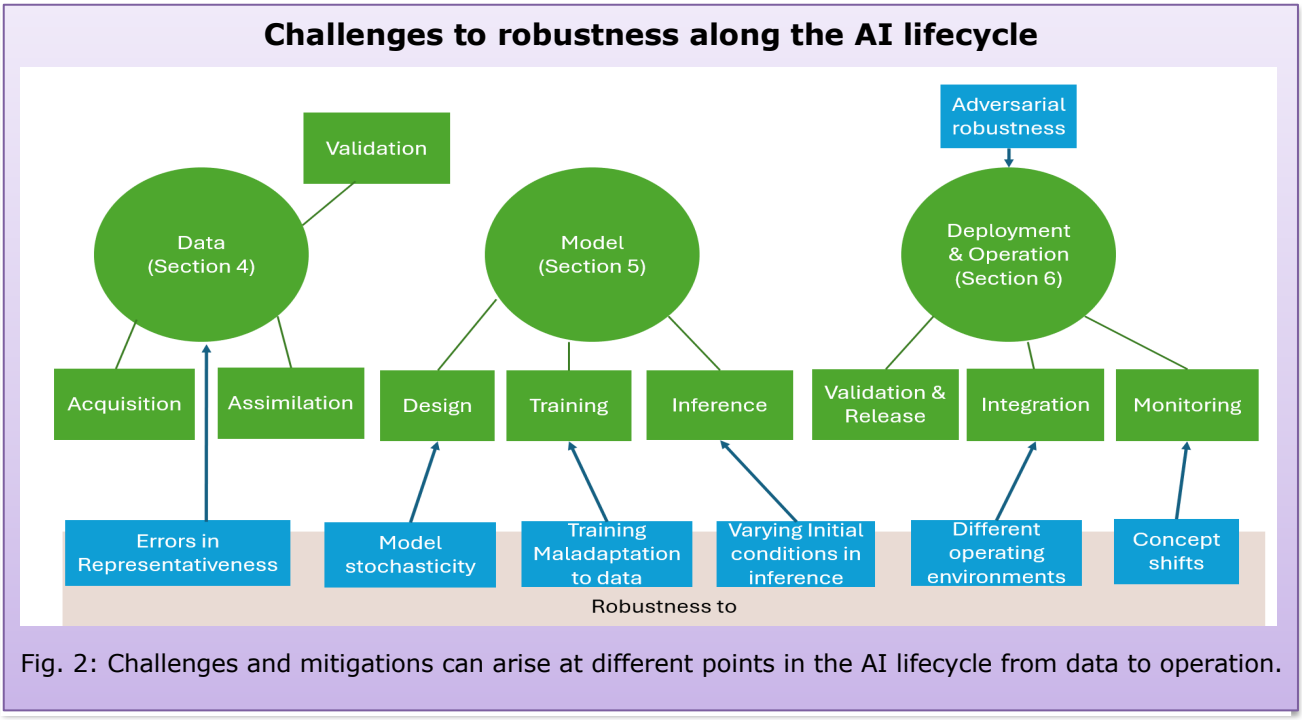

Fig. 2: Challenges and mitigations can arise at different points in the AI lifecycle from data to operation.

Data Acquisition, Understanding & Preparation

The typical first phase of the AI lifecycle[2] focuses on the data used to train the AI model, including acquisition, understanding and preparation.

Regarding AI systems for weather forecasting, this includes collecting data from various sources starting from satellites to weather station data. This data might come in different formats, resolutions, stem from different time points and include other variables depending on the source and must be unified to a format suitable for training (and inference). At the heart of this process are systems for data assimilation, i.e. combining observations from different sources into a coherent grid of variables, which is primarily used for forecasting initialization, but can in turn also be the starting point for training as it represents the most likely state of the Earth system.

Inference data used for actual forecasting undergoes an equivalent procedure of combining weather data into a grid understandable to the model, but instead of considering larger spans of historical data it is focused on (approximately) real-time and recent data for future predictions.

Model Development & Training

Model Development involves the choices of model architecture, optimization of hyperparameters and inclusion of safeguarding measures *by design*. This includes choices made in the training process (e.g. different optimizers) with impact on the robustness and reproducibility of the training process.

[2] For reasonable data acquisition, a clear objective and domain understanding is necessary, including the identification of trustworthiness requirements that need to be considered during development. In some frameworks, this is seen as the first lifecycle phase.

Typically, in weather forecasting, uncertainty quantification plays a significant role and is often part of the model design, as the additional information provided by, e.g., ensemble approaches is crucial for decision making based on weather forecasts. Approaches to uncertainty quantification with their advantages and disadvantages are discussed in Section 5.

Model Deployment & Operation

Upon finalization of the training-and optimization process, the next question is how the model can be integrated into existing structures and go operational.

Here on the one hand, a step of evaluation and validation should follow model development to ensure the trustworthiness. Following tests and validation, model deployment also depends on the hardware the model is deployed on. Hardware dependencies in code can result in performance differences in different operating environments.

While the model is in operation and regularly producing forecasts used in real-world applications, model performance degradation is possible if the distribution of input-data or the relationship between input and output the model has learned on training data changes in the real world.

Reproducibility along the AI lifecycle

Reproducibility plays a part at every stage of the lifecycle. Looking at the phase concerning data, the step of data assimilation is particularly relevant. There, reproducibility is given if the same observation data leads to the same data on the grid, which is then used for model training and inference.

Model training and parameter optimization often rely on inherently stochastic algorithms. The deterministic AIFS model for example is trained on ERA5-datasets using AdamW [7] as a stochastic optimization algorithm to arrive at model weights. Upon arriving at a model, one could ask for reproducibility in inference, i.e. producing the same results from the same input when making forecasts.

In deployment and operation, differences in software and hardware environments compared to training time can present challenges to reproducibility efforts.

4 Data Acquisition & Preparation

Models used in weather and climate forecasting depend on diverse and complex datasets collected from multiple sources. Unlike Numerical Weather Prediction (NWP) models, which rely on explicitly defined physical laws, data-driven models infer patterns and implicit rules from historical data. This leads to an even higher susceptibility to data-related challenges such as sparsity, inconsistency, bias, and distributional shifts, all of which can compromise predictive robustness. Challenges in data management, such as incomplete raw data, discrepancies between observational platforms, or uncorrected biases can propagate errors throughout the entire AI pipeline, causing less reliable predictions, potentially leading to inferior decision-making in sectors such as disaster prevention, agriculture, and energy planning, and ultimately reducing trust in data-driven forecasting systems. Addressing these challenges is therefore essential to ensure robust, fair, and reliable AI-based forecasting systems.

In this section, we examine the role of the data acquisition and preparation stage following the MLOps paradigm. This includes the collection, curation, transformation, and validation of data. We will explain these steps in the context of data-driven weather forecasting, describe common issues that can arise and strategies to mitigate those. Topics covered include how to manage data sparsity, handle inconsistencies across sources, detect and correct errors, address bias in training data, and enrich datasets to improve model robustness.

The most common ML-based forecasting models rely on a structured grid of variables. For these, the process of data assimilation, whereby new observations are merged with a short-term forecast from a previous time step to arrive at the best guess for the state of the Earth system, is central at this stage and fulfils several tasks along the steps of curation, transformation and validation. At the end of the section, we will also cover novel models that are trained directly on observational data, circumventing the need for computationally expensive data assimilation.

4.1 Data Collection

Weather forecasting relies on data collected from various sources such as satellites, radar systems, ground-based sensors, aircraft observations, weather balloons, and buoys. These diverse sources provide information on physical variables such as temperature, humidity, wind speeds, and atmospheric pressure, either by directly measuring the quantity or by measuring an indirectly related quantity – this is especially the case for radiances observed by satellites. They differ in spatial and temporal resolution, update frequency, and measurement precision. Satellites provide broad coverage but coarse vertical resolution, radar systems offer high-frequency updates with limited range, and ground-based sensors or radiosondes deliver high-accuracy local data but are unevenly and sparsely distributed, especially in remote regions such as the ocean or polar areas [8].

These observations form the foundation for predictive models – both physics-based and data-driven. However, spatial sparsity may create epistemic uncertainty, meaning that the model may lack sufficient observations to confidently predict outcomes in data-sparse regions. This is exacerbated by the fact that the most impactful events are rare extreme weather phenomena such as hurricanes and floods. Additionally, sensor

noise can result in measurement errors, and sensor failures, wrong metadata like geolocation, or variations in calibration between sources can result in missing or erroneous inputs, compromising forecasting accuracy [9], [10], [11].

Addressing Sparsity and Heterogeneity

These gaps in observations are addressed mainly through data assimilation, where missing values are estimated by merging a forecast from a previous timestep with new observations. We will cover this in more detail in section 4.3 below.

Mitigating Imbalance in Training Data & Synthetic Data Generation

A common approach to handling data imbalance in training datasets is through data resampling techniques, where underrepresented extreme weather events would be oversampled or weighted more heavily. This oversampling, however, would introduce a usually unwanted skew in the data that would reduce its physical soundness and reduce the applicability of models trained on the data as realistic predictors of weather.

Instead, to further enhance the reliability of data-driven models, these models benefit from specially adapted training methods that address data imbalance [12].

Increasing the size and diversity of training data by adding synthetic data from climate simulations to the training set has also been shown to improve robustness of models trained on the data, especially for extreme values [13]. The growing use of data-driven models, combined with the affordability of conducting multiple climate simulations under varying assumptions with them, makes them appealing for generating synthetic datasets. In that case, however, it is crucial to exercise caution to prevent model collapse and the learning of artificial, unphysical characteristics [14]. A balanced training dataset, combining real and synthetic simulated weather, helps ensure that data-driven models maintain realism and generalizability while minimizing biases.

4.2 Data Curation & Validation

Raw meteorological data must undergo systematic cleaning to remove gross errors like inconsistencies and redundancies that arise, for instance, due to duplicate measurements or incorrect metadata like geolocation, as well as wrong anomalous sensor readings. Without thorough validation, errors in input data propagate through AI models, leading to poor forecasting performance.

Consider, for example, a malfunctioning temperature sensor, which gives extreme values instead of an accurate reading for a single location. Assuming this anomaly is not treated in pre-processing, it would influence the forecast. While physics-based models have intrinsic constraints (such as energy conservation) that can somewhat mitigate the effect of single anomalous observations, operational experience shows that individual erroneous observations can still exert significant influence on NWP predictions [15]. For a data-driven forecasting system such an out-of-distribution input may have an even more severe impact on performance [16].

Outlier Detection

The exposure to potential errors in data necessitates dedicated quality control teams and effective monitoring mechanisms to identify and mitigate these observational errors proactively during operation. Rigorous validation procedures are required both on a continuous as well as a per incident basis, where specific events are analysed in detail and conclusions drawn. This includes calibrating for drifting sensor biases as well as the adjustment of reliability scores or blocklisting of unreliable data sources [17], [18]. By combining automated outlier detection methods – such as unsupervised algorithms that detect abrupt deviations or gradual drifts – with human oversight and expert judgment, it is possible to identify anomalies, assess their severity and causes accurately, and efficiently determine corrective actions. Usually, these quality control checks can happen before data assimilation – e.g. a comparison against climatology, the average weather variables for the location and time of year, as a baseline – as well as during data assimilation – e.g. the calculation of error probabilities for each observation. This integrated approach reduces manual intervention, enhances data reliability, and significantly strengthens the robustness of both data-driven and physics-based forecasting models [9], [11].

Ensuring Data Provenance, Traceability, and Governance

Maintaining data integrity and auditability also requires robust data lineage tracking and version control, ensuring that every transformation applied to a dataset is recorded and traceable. This improves reproducibility, allowing meteorologists and AI researchers to replicate models and verify model outputs. Additionally, governance frameworks establish transparency and accountability in how data is collected, processed and utilized in forecasting, ensuring that the same data can be used for forecasting, climate change research, and other use cases, making sure decision-making is based on verifiable and well-documented datasets [19], [20], [21].

4.3 Data Transformation

For structured grid weather forecasting models mainly two data transformations occur. On the one hand, remote sensing observations from satellites do not directly measure the desired physical variables, which must therefore be deduced from the measurements. On the other hand, the models require a complete grid of input values, which necessitates inferring these values from sparse and noisy observations that are not located at these grid points. The goal of this process of data assimilation is therefore to produce a physically consistent overall best guess of the Earth system state at the fixed grid points, the analysis, using numerical techniques such as Variational Data Assimilation (3D-Var, 4D-Var) and Ensemble Kalman Filters (EnKF). These methods iteratively adjust model states to align with observational data, ensuring consistency with physical laws. However, they require significant computational resources and can be sensitive to observation errors [8], [22], [23].

The data assimilation deals with several problems that arise in previous steps. By blending a forecast from a previous timestep with current observations, it interpolates missing data, and reconciles potentially inconsistent data from various sources into a single initial condition [24]. Nonetheless, to achieve an optimal initial condition we require a precise understanding of the uncertainties in both the observations and the

prior forecast state, as well as the relationship between the measured quantity (e.g. radiance) and the desired variable. This is a source of further epistemic uncertainty.

Optimizing Bias Correction

Both data assimilation and machine learning methods share fundamental similarities, notably their reliance on observational data to iteratively refine and correct model predictions. For instance, to address systematic errors either from the observations themselves, such as instrument anomalies or calibration issues, or from the parametrization of the relationship between observed and target quantities, typically parameters of a bias model are fitted to align with other observations that do not suffer from the same observational biases.

Direct from Observation to Prediction

An alternative, innovative strategy to avoid traditional data assimilation entirely is to use a model that predicts a future state end-to-end from observations. Two examples of such a system are Artificial Intelligence–Direct Observation Prediction (AI-DOP) [25] and GraphDOP [26], developed by ECMWF. They use ML models trained directly on historical observational datasets, enabling forecasts based solely on current observational data. This approach simplifies data preparation by removing the requirements to map raw observations onto structured model grids. Consequently, DOP models allow weather forecasting systems to directly leverage observational data in its native form, potentially improving data utilization and forecast skill. Recent advances demonstrate that DOP can produce accurate medium-range forecasts purely from observational data, representing a significant shift toward more streamlined and potentially more efficient weather prediction workflows [25].

However, while traditional data assimilation methods explicitly incorporate physical laws through numerical constraints, pure ML methods typically rely solely on statistical relationships learned from historical data. Physically constrained ML bridges this gap by explicitly embedding meteorological principles within data-driven models [27]. Harmonizing physics-based numerical weather prediction (NWP) models with such physically constrained ML approaches can ensure that ML predictions remain consistent with established meteorological knowledge [24], [27].

5 Training Process and Model Building

As explained in Section 3.2, the second stage of the AI lifecycle involves model creation and training process of AI models. This section focuses on robustness and reproducibility at this stage. With the recent accelerated evolution of machine learning, data-driven approaches integrate machine learning into weather forecasting more than ever, exemplified by models like ECMWF's AIFS. This transition to supplement traditional NWP forecasting with data-driven approaches introduces fundamental challenges to robustness and reproducibility in training and model building, that are inherent to data-driven models. For robustness such challenges encompass risks of overfitting, e.g. to historical reanalysis data like ERA5 during training, resulting in maladaptation to other climate regimes; reproducibility is impacted by, for instance, stochasticity in training workflows or hardware dependencies of the underlying implementation. Both can impact trust in ML-based forecasts, particularly for critical scenarios such as extreme weather events or long-term climate projections.

In the following we first look at various robustness and reproducibility challenges and their implications in weather forecasting. We then explore mitigations in the subsequent subsection. Uncertainty quantification (UQ) emerges as a tool to mitigate some of these challenges.

5.1 Challenges in the training process

Some of the robustness challenges of training data-driven forecasting models can be illustrated looking at the development and changes to it in ML and the meteorological community. Initial prediction models were treated as regression-type problems, which were optimized using the (appropriate) Mean-Squared Error (RMSE), trying to minimize average deviations to the historic forecasts. As an optimizer, typically a variant of stochastic gradient descent (SGD) was employed [28], incrementally optimizing against random subsets of the data. They often are computationally cheap and provide implicit regularization during training, which yields better generalizing solutions, despite or rather because they introduce stochasticity into the training process.

Using the RMSE optimization objective, it turned out that the tendency to provide mean results, while good "on average" can be especially detrimental to the prediction of extreme events. Coined as "double penalty" [29] predicting an extreme event at a location slightly shifted from the real location in time and space leads to strong error signals due to both the missed prediction and the wrong, i.e., misplaced one. It is thereby typically "beneficial" for the AI system not to predict such extreme events or outliers. This, in turn leads to overly smooth solutions, due to under-fitting, as explained further below.

Instead of directly predicting the outcome, second-generation models therefore try to predict the distribution of likely outcomes. For example, the approach in Diffusion-AIFS is training forecasting as a denoising task, others train using probabilistic loss functions, measuring differences between probability distributions instead of data points (e.g. KL-divergence, Ranked probability score). Each of the cases typically inherently includes a form of uncertainty quantification, the challenges and potentials of which are laid in the section on uncertainty quantification.

Internal stochasticity

Stochasticity can exist both as part of the training process and as elements of the model itself. In terms of training, the commonly used optimizers, such as Stochastic Gradient Descent or its successors such as Adam, rely on random data selection to train neural networks and start their training on a random initial state. While such randomness, in principle, is controllable, this is technically challenging as often tasks during training are parallelized, e.g. for data loading or pre-processing, and for larger models might also be physically distributed across different server nodes. To add, machine learning typically favours speed over accuracy in terms of calculations as precise outcomes for weights, due to the ubiquitous randomness already in the idea of training. This is reflected by networks often using limited precision, often even using strongly quantized results in later inference, and usually do not control for noise introduced by mathematical operations on finite precision floating point numbers. This noise can depend on the concrete implementation of the operations and can differ between hardware architectures, potentially further decreasing reproducibility.

Outside of training, the architecture might also contain deliberate randomness. Examples include the previously mentioned diffusion processes, which use a random initial state for each inference. Such an auxiliary random input is also used in Generative Adversarial Networks (GANs) [30], which can be employed, e.g. to downscale weather data while avoiding excessive smoothing [31]. Further, some building blocks of architectures might possess their own randomness. For example, Monte Carlo Dropout, which randomly disables neurons during training or inference [32], is a stochastic technique for regularisation.

Maladaptation to training data

So far, we considered potential model flaws from an uncertainty-based perspective, considering how errors propagate and what is predictable. Another perspective is given by the bias-variance trade-off in Machine Learning. Depending on the complexity of the model in relation to the task the solution it produces can either possess bias or demonstrate variance, which are more commonly referred to as under- or overfitting, respectively. In the first case the model is biased towards specific types of solutions, for example a linear model trying to fit a non-linear problem, which lead to a systematic deviation between desired and expressed model behaviour. For example, the RMSE trained model example before was biased towards smooth solutions resulting in strong bias against extreme events. The other way around, an overly expressive or complex model might strongly specialise on the given data. Its results would then strongly depend on the limited data learned and models trained on varying data would produce strongly differing outcomes (variance). When tested on unseen data this often implies a lack of generalisation. Regularisation is one way to control model complexity and balance such trade-offs. As a remark, while this is common practice for most of Machine Learning and AI, currently emerging modern architectures are often complex enough to operate in the so-called double descent regime, for which the bias-variance trade-off no longer holds [33].

For neural networks training is further complicated as the loss landscape of such models over the learned parameters is highly non-convex [34] and has multiple minima of varying quality. That implies that optimization not only propagates randomly, as discussed above, but can also – depending on choice of optimization

algorithm[3] - randomly arrives at one of several minima. This has implications for both the optimization methods and reproducibility of the training as, although models at most different minima behave similarly performance-wise, they do not produce exactly the same outcomes and might even have different failure types.

Linked to these aspects and models' abilities to generalize is the question of what AI models learn during training. For previous NWP models this was directly encoded based on our physical understanding, with data-drive models however it is unclear to which extend it is reflected. While observation shows that physical constraints are reflected in AI [35], this is not a guaranteed fact. Explaining and investigating model decisions is therefore an important question in the field of Explainable AI (XAI) and one of the central topics of an accompanying whitepaper.

Here, instead, we briefly consider on what data the model was learned. In Section 4 we discussed that lacking data coverage may lead to epistemic uncertainty. Before, we gave the example of rare extreme events that are hard to fit; in a similar fashion if a model is trained pre-dominantly on data of a specific region, it might be enough for the model to achieve a good overall metric to fit this region at a disadvantage to others. This implies that data coverage can induce a bias, for instance towards regions with good coverage, implying good availability and quality of the data, the latter, e.g. with respect to the granularity of spatiotemporal measurements. It also can imply bias towards specific time frames considered, as we will discuss later in the operations section.

5.2 Potential Mitigations

In the following, we present possibilities to assess and mitigate the mentioned issues and challenges. A focus will thereby lie on uncertainty quantification as a tool to understand and quantify certain types of robustness.

Preventing maladaptation

Overfitting is an issue that can be tackled in multiple ways. One option for deep neural networks would be to include drop-out layers – this random disabling and enabling of neurons in training creates more flexibility in the model's behaviour in inference. This can be seen in the wider context of regularization, a range of techniques aiming to simplify ill-posed problems or avoid overfitting, e.g. by modifying loss functions to include penalty terms to discourage undesirable model behaviour.

Other approaches have included training models on more general climatic data, before fine-tuning on shorter-term (and sometimes regional) weather data [13]. While the primary motivation is to extend the amount of training data for regional models, where not a lot of historic data might be available, it also helps to prevent overfitting to overly specific training data.

[3] Optimizers such as AdamW include exploration of the loss landscape and can thereby typically avoid getting stuck in local minima.

Uncertainty quantification

Uncertainty quantification (UQ) as introduced in Section 3.2 measures uncertainty of predictions by a model and therefore adds actionable probabilistic insights to deterministic or probabilistic predictions. The provided insight through UQ is crucial in research and decision making in weather related applications [36]. For farming or energy supply, downstream users might depend on uncertainty quantifications to make insurance decisions. It addresses the challenges of propagation of errors and impact on forecasting horizon as well as internal stochasticity by allowing to measure impacts.

The two types of uncertainty defined in the Section 3.2 aleatoric and epistemic are differing by their source. In contrast to aleatoric uncertainty, epistemic uncertainty can be reduced upon additional complementary information. Considering these differences, usually scientists use different UQ methods to model them. Combining both types of uncertainty delivers predictive uncertainty, meaning that a UQ generated measurement could be translated to a confidence value for predictions of a model. In the following we will give an overview of these uncertainty quantification methods:

The most common approach to uncertainty quantification is to use some form of ensemble, i.e. using configurations of one or multiple models initialized on multiple instances of perturbed initial conditions. Fig. 3 shows a simple approach, using the same model across different perturbations of initial data. The goal is to understand the propagation of these slight changes in initial conditions throughout the model – mainly used to quantify the irreducible aleatoric uncertainties. In practice, while this method is easy to implement in principle, finding sensible perturbations that could simulate natural conditions is quite challenging. This is known as test-time augmentation (Fig. 2).

Example: How does uncertainty quantification work?

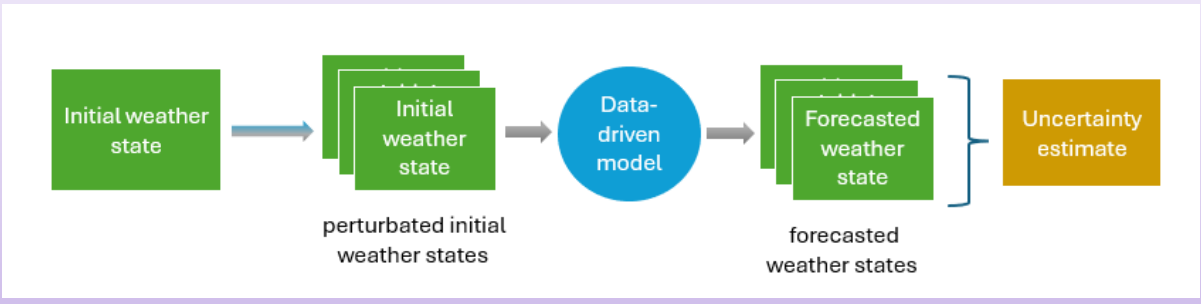

Fig. 3: Initial condition-based ensembling approach to uncertainty quantification

Similarly, to not only capture aleatoric, but also epistemic uncertainty, deep ensembles of different models can be considered. This is more expensive, as multiple models must be trained, and training is much more costly for ML-models compared to the relatively cheap inference. Combining the training of multiple models with perturbations to initial conditions in one ensemble gives an estimate of the combined aleatoric and epistemic uncertainty. Related, instead of training multiple models, one training run can be used by exploring the neighbourhood of the optimum in the parameter space and generating models based on parameter configurations close to this optimum. These ensemble methods are in principle like those used in physical ensembles – here perturbations of the initial conditions are also used to estimate aleatoric uncertainties [37]. The

advantage that data-driven models have is their comparatively low computational cost in inference, making ensemble predictions remarkably more efficient.

Bayesian approaches such as Bayesian neural networks inherently optimize model parameters and output as probability distributions. An advantage of these methods is their capability to both capture both aleatoric and epistemic uncertainty, while their cost remains reasonable compared to a deterministic neural network. They can be used in conjunction with most deep learning methods. However, predicting uncertainties accurately can be difficult and require tuning, in some cases the uncertainties tend to be overestimated.

Further approaches include encoder-decoder architectures such as variational autoencoders, the injection and shaping of noise or the use of drop-out layers in neural networks. While noise injection can enhance model generalization, excessive addition can degrade accuracy, and requires additional measures to ensure reproducibility [9], [17]. Choosing the correct method depends heavily on the context of the system, available computational resources and type of uncertainty to be assessed.

Disentanglement of aleatoric and epistemic uncertainty

Different approaches to uncertainty quantification have been introduced, measuring either aleatoric or epistemic uncertainty or both. By incorporating methods and variations only targeting one uncertainty type, disentangling them is possible (e.g. by considering variations in the initial conditions for aleatoric uncertainty and hyperparameter variations for epistemic uncertainty separately).

In many cases, and especially during the development process of models it can make sense to disentangle these types of uncertainty, for example to support model optimization by accurately capturing and understanding epistemic uncertainty. High epistemic uncertainty can, e.g., point to weaknesses in hyperparameter optimization or training data selection or at least inform about sensitivities surrounding the selection of such parameters. If, as e.g. in [38] epistemic uncertainty is particularly high for some regional predictions, measures for retraining or fine-tuning on regional data could be undertaken.

6 Deployment and Operation

Within this section, we discuss as next lifecycle stage deployment and operation. Compared to CRISP-DM, we also include the evaluation phase here. Challenges in this lifecycle stage therefore fall in either of the following categories:

1) How can model robustness and reproducibility be validated with technical tests and quality gates?
2) Given the oftentimes stochastic nature of AI, how can reproducibility for such systems be defined and to what extent it can serve similar purposes to those of bit-reproducibility?
3) How can potential performance deterioration over time be monitored and mitigated?

6.1 Validation and quality gates

Bit-reproducibility is comparatively easy to verify – the models should produce exactly the same outcome on multiple re-runs. Since it currently seems difficult to achieve this with data-driven models (see Section 6.2), a relevant research question is, what level of reproducibility should be aimed for and how it can be measured. Here, as introduced in Section 3, it may make sense to weaken the conditions of bit-reproducibility and consider alternative approaches quantifying "sufficient" reproducibility.

For those ML-models, that are in principle deterministic in inference[4], developing benchmarks e.g. for event onset recognition, could be imaginable in order to quantify robustness to remaining stochastic elements in the code.

For probabilistic ensembles, this is more difficult, because different ensemble members are expected to behave differently. Reproducibility by fixing random seeds would be possible in theory, but it eliminates the useful variation that ensembles use to quantify uncertainties in the first place and therefore is pointless in practice. However, fixing random seeds could help save storage, as one of the discussion points regarding reproducibility in weather forecasting is whether it is desirable to store all model outputs or whether one should aim for making exact reproductions of outputs possible. So, while fixed random seeds are highly impractical for model operation, saving the values of the seeds to allow for re-runs could help mitigate this storage issue.

As for robustness, testing is not trivial, since often the behaviour of weather forecasting models is inherently sensitive to differences in initial conditions. This inherent aleatoric uncertainty should ideally be captured by uncertainty quantification, while trying to keep epistemic uncertainty as low as possible. Disentanglement of the two types of uncertainty can help address this issue. Monitoring epistemic uncertainty thereby can be a way to check model robustness but is also useful for systems that are still in development to test hyperparameter configurations and the effectiveness of robustifying measures.

Before release, not only the quantified uncertainties should be assessed, but more general testing is required. To assess the capabilities of new models and address the

[4] For example, models trained with RMSE to produce a single output such as the deterministic AIFS.

potential issues of overfitting and regional differences or biases in model performance, it is common practice in the ML-community to test the model on real-world data.

Example: Validation practices at ECMWF

Fig.4: Excerpt of a scorecard validating AIFS-single performance compared to the IFS, evaluated against analysis and observation data for different variables and metrics. Performance is very promising and often compares to or exceeds IFS but falls short of IFS in the stratosphere (50hPa - red areas). Graphic adapted from [7]

6.2 Integration into existing software and hardware structures

Currently, there exist no de-facto unique standard software and hardware structure for weather forecasting systems. Therefore, it is possible that the infrastructure for deployment is different or incompatible to the infrastructure during model training. Another scenario with the same root cause that challenges robustness and reproducibility of deployed models, is when a new model is integrated into an existing forecasting system. A particular challenge is that models can propagate earlier errors downstream, impacting dependent components (e.g. data pipelines, decision systems).

In terms of reproducibility, new sources of inaccuracy can arise in implementation and deployment on changing hardware. These challenges can e.g. arise through different compilers changing the order of operations in computations. Common ML-libraries such as TensorFlow or PyTorch – while allowing to fix random seeds – are often still non-deterministic in parts [39]. While there are deterministic environments offered, these often come with trade-offs in run-time and computational costs [39].

An example of this trade-off is differences in the implementation of matrix-multiplication, which is fundamental in training and inference of neural networks. While some may rely on fewer CPU operations and thereby support computational performance, this can come at the cost of slight deviations in floating-point calculations or rounding errors, which in turn can propagate through the AI system and lead to decreased reproducibility [40].

Publication of software in reproducible containers can support efforts to move towards reproducibility by providing a deterministic execution environment.

Similarly, however, underlying hardware can influence the order of operations, e.g. by use of parallelization. Currently, achieving full bit-reproducibility is considered very difficult, as without the use of advanced parallelization and stochasticity in forecasting, ML models would be set back significantly both in runtime performance and computational cost.

6.3 Performance degradation over time

Data-driven models have the inherent property that they heavily rely on the data they are trained on. While there is some evidence of generalizing abilities, changes in the underlying data distribution, e.g. due to climate change can lead to performance decrease. Commonly, we distinguish between data drift and concept drift.

Data drift refers to changes in the model input over time. In the case of weather forecasting this could be changes in the data assimilation procedure or differences in observed data. This can lead to performance decrease over time, as gradually the input data drifts further from the distribution the model was trained on.

Concept drift on the other hand refers to a change to the input-output relationship, i.e. the dynamics the model has learned have changed. One example could be changes in atmospheric dynamics due to climate change. For climate models, [41] suggests, while still showing overall good performance and robustness to climatic scenarios, that models tend to drift towards the training data compared to the initial conditions, which can be seen as a bias affecting forecasting quality. Currently there is a lot of research going on to examine this question of generalization across different climates and results are promising in the sense that at least there are indications that the drivers of atmospheric dynamics on timescales of weather forecasting are not fundamentally changing with climate change. Still, the current shortcomings make the questions of concept drift and data drift relevant.

Example: Data-drift in the form of a mean bias

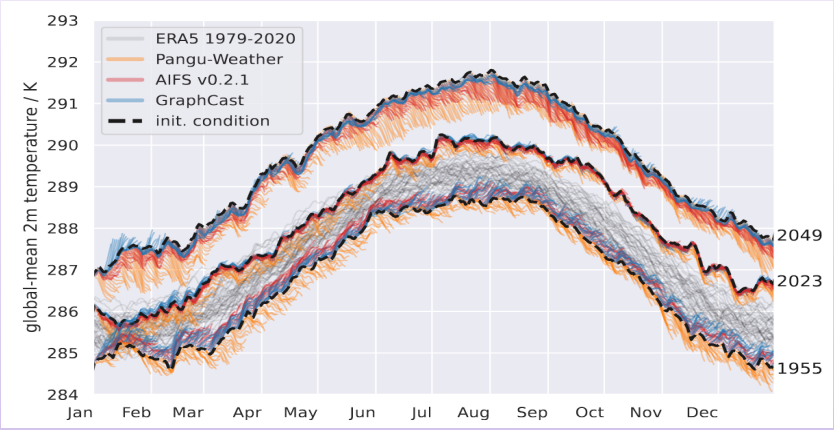

Fig.5: Mean bias refers to data-driven models' tendency to drift towards their training data, leading to a bias on predictions that could require monitoring and retraining. In this example, initializations for different models trained on data from 1979-2020 on historic, current and future conditions show drift towards the training data dynamics. Graphic taken from [41].

Data and concept drift often go together – changes in the atmospheric dynamics will likely results in changes to both the input-output-relationship the model has learned as well as the initial conditions themselves. It can naturally be difficult to distinguish between them from only the outputs themselves. Nonetheless, monitoring in operation is crucial and common practice in the weather forecasting community.

Typically, to detect data drift, statistical properties of the input data (i.e. the initial conditions) are considered [42]. Typically, at first summary statistics are monitored, i.e. means, variances or amount of datapoints within certain thresholds. While for smaller datasets, statistical tests are feasible, e.g. Kolmogorov-Smirnov-tests for the hypothesis that input datasets follow the same distribution, however for larger data sets other methods, such as measuring distances of the input dataset in inference to a reference dataset or previous input datasets in metrics such as the Wasserstein metric are more robust and advisable. Concept drift is measured on the model output – if the model performs worse in quality metrics over time, this indicates s an indication for potential concept drift.

Mitigation typically involves a retraining of the model to adapt either to the new distribution of input data (e.g. include more data of phenomena that were previously lacking complete coverage in training data) or the changes in input-output relationship (e.g. potentially fundamental climatic changes due to tipping points).

Continuously integrating new data into model training can support mitigating these issues and prevent drift-related deterioration of model performance from occurring.

7 Strategies towards robustness and reproducibility from a governance perspective

Following the presentation of challenges and approaches to robustness and reproducibility along the AI lifecycle, in this section we will focus on governance and processes from a higher-level perspective.

7.1 The process-oriented perspective

An integral part of the EU AI Act's requirements for high-risk AI systems addresses the establishment of governance structures and processes to achieve and consistently maintain desired performance, going beyond prescribing measurable features of an AI system. These requirements can serve as guidance for best practices and voluntary codes of conduct for systems that do not fall in the high-risk class. For example, a quality management system (Art. 17) with integrated risk management (Art. 9) can lay out procedures and systematic actions to help structure the design and validation process for an AI system and support documentation and versioning efforts.

There is a large number of systematic approaches available that lay out best practices, both from the perspective of AI assessment and AI development. Besides technical tests and mitigation measures, for the topic of reproducibility, versioning and documentation are paramount for both training data and models themselves, helping to understand and define suitable operating conditions and operational design domains, i.e. specifying permissible conditions, under which the model should be operated and performs adequately.

With increasing maturity, possibilities for automation arise, for example the automatic triggering of a retraining following the detection of data or concept drift. The MLOps framework as presented by Google [5] distinguishes between three levels of automation:

MLOps Level	Description
Level 0	Manual – Data preprocessing, model training and model deployment are done manually with separate code
Level 1	ML-Pipeline automation – Data preprocessing and model training are automated and can be triggered, e.g. when noticing data drift
Level 2	CI/CD-Pipeline automation – Validation, Integration and Deployment are further automated

Fig.5: Levels of Automation in Google's MLOps Framework

A higher degree of automation can drastically improve time for changes and time to restore in case of defects.

It can also support efforts to move towards reproducibility at different lifecycle stages by:

- automating versioning and meta-data generation
- reducing the influence of human errors or issues arising from incorrect software versioning and incompatibilities
- allowing the definition of quality gates in release management

7.2 The role of international standardization activities

Typically, recommendations by from standardization organizations are among the first places to look for systematic approaches for guidance on requirements analysis and development processes.

While due to the rapid emergence and fast progress of data-driven forecasting systems WMO guidelines are lacking on most related topics, some ISO technical reports exists which are applicable to AI systems beyond weather forecasting.

ISO/IEC TR 24029 [43] provides methodology to assess robustness properties of neural networks. While 24029-1 focuses on giving an overview of robustness assessment methods for a broad range of methodologies, e.g. classification models, particularly the aspects of interpolation stability could be useful when adapted to the scenario at hand as a way to measure robustness to inexhaustive training data sets.

24029-2 focuses on formal approaches to stability and ends with an interesting view of robustness along the AI lifecycle. Here, particularly relevant are sources of numerical accuracy in deployment, which has been addressed in Section 6 and some development methods, such as clearly defining the input domain (or ODD) and using an identification of relevant hidden features. We will cover these transparency related aspects in more detail in the accompanying whitepaper on explainability and resiliency against manipulations.

Currently under development is ISO/IEC AWI TS 25223 [44], an approved work item focusing on approaches to uncertainty quantification.

7.3 A scientific perspective on reproducibility and robustness

A question that has not yet been considered in detail yet, is the extent to which other researchers will be able to reproduce the forecasts and results. This fundamental question of reproducibility in research is not covered by the internal view of reproducibility. This question relates to the process-oriented perspective, since many aspects of this form of transparency are part of developers overarching strategies, e.g. designing code and products for openness, aiming for reproducibility of scientific results by providing the necessary tools and communicating models' strengths and weaknesses with respect to robustness and reproducibility to the general public.

Reproducibility in scientific research has been an issue in multiple scientific disciplines, in part due to publication practices that favour publishing positive vs. negative results.

With the recent emergence of ML-methods in scientific research, there are new fears of a reproducibility crisis accompanying this surge in particular methodologies [45].

While a broad view of transparency and explainability in weather forecasting will be given in the accompanying whitepaper, it makes sense to consider the aspects of transparency that are fundamentally linked to (external) reproducibility.

The biggest barriers in machine learning research outside of weather forecasting are limited access to data and code needed to reproduce experiments and results. In 2018, for published ML-research, only about one third of researchers shared data and 6% shared the code of the algorithms [46]. In the domain of weather forecasting, the landscape is looking quite different and open research is widely viewed as a guiding principle.

One such effort undertaken by ECMWF is Anemoi – a framework being developed by ECMWF and its member states, consisting of core modules for data preprocessing, modifications to model training, the models, inferences and graphs. The framework is available under permissive licenses and allows other researchers to not only reproduce results but also build own research on top of it by altering the modules without having to deep-dive into the intricacies of the lower levels of the code. This facilitates reproducible findings by lowering engagement thresholds, on the other hand the possibility remains to delve deeper into the functioning and code of the models.

Transparent communication plays an important role for robustness and uncertainty quantification as well. As knowledge of uncertainty can inform a wide range of decisions from holiday planning over crop yields to energy supply, communicating them to the general public is crucial and has been a challenge.

To adapt to the language and needs of users and mitigate potential human perception biases[5], uncertainty scales such as the scale used by the IPCC use a combination of plain language and probability estimates (“>99% - virtually certain, >90% - very likely, …).

[5] Such as the magnitude of events having an impact on their perceived likelihood [47].

8 Conclusion

Data-driven models offer great promise, although many challenges remain due to their rapid emergence. While some of these challenges in terms of robustness and reproducibility can be found in physical NWP-systems as well, and data-driven methods offer improvements, new challenges compared to traditional forecasting mainly arise through

i) the black-box character of AI models
ii) the increased reliance on data directly influencing model behaviour
iii) the use of new and different probabilistic methods

Aspects related to transparency and explainability, such as i) we will address in an accompanying whitepaper [b]. Regarding ii) and iii) it is noteworthy that these new challenges for data-driven systems have ethical implications.

Data quality becomes even more important compared to NWP-systems, as unmitigated data biases can directly generate discriminatory models exhibiting performance differences in predictions, if left unmitigated.

The increased stochasticity and trade-offs with performance and runtime makes bit-reproducibility difficult to achieve and less desirable, requiring alternative approaches to account for the gains in development and confidence in results that reproducibility can provide.

In line with the view on robustness brought forward by the EU AI Act and common paradigms in ML-development, these issues can arise at and propagate through different lifecycle stages. In a structured development process, these challenges can be tackled systematically, integrating the presented validation and mitigation approaches into existing trustworthiness frameworks.

As described in the introductory whitepaper of this series [a], this project is developing practical guidelines [c] that take a similar lifecycle view but extend ethical best practices for structured ML-development beyond robustness and reproducibility.

9 References

[a] DE398 Whitepaper - Adoption of Ethical Principles for Machine Learning in the Meteorological Domain, 29.01.2026

[b] DE398 Whitepaper - Explainability and Adversarial Vulnerability in AI-Based Weather Forecasting Systems, 02.02.2026

[c] DE398 Guidelines – Practical Guidelines on Ethical Machine Learning, 29.01.2026

[1] M. Poretschkin *et al.*, "Guideline for Trustworthy Artificial Intelligence -- AI Assessment Catalog," 2023, *arXiv*. doi: 10.48550/ARXIV.2307.03681.

[2] D. Hendrycks, K. Zhao, S. Basart, J. Steinhardt, and D. Song, "Natural Adversarial Examples," CVPR, 2021.

[3] C. Szegedy et al., "Intriguing properties of neural networks," in International Conference on Learning Representations, 2014. [Online]. Available: http://arxiv.org/abs/1312.6199

[4] C. Schröer, F. Kruse, and J. M. Gómez, "A Systematic Literature Review on Applying CRISP-DM Process Model," Procedia Computer Science, vol. 181, pp. 526–534, 2021, doi: 10.1016/j.procs.2021.01.199.

[5] K. Salama, J. Kazmierczak, and D. Schut, "Practitioners guide to MLOps: A framework for continuous delivery and automation of machine learning." Google Cloud, 2021.

[6] S. Gießelbach, "Data Science with Foundation Models : An Evidence-Based, Comprehensive Project Methodology," PhD Thesis, Rheinische Friedrich-Wilhelms-Universität Bonn, 2024. [Online]. Available: https://hdl.handle.net/20.500.11811/11566

[7] S. Lang et al., "AIFS -- ECMWF's data-driven forecasting system," Aug. 07, 2024, arXiv: arXiv:2406.01465. doi: 10.48550/arXiv.2406.01465.

[8] F. M. Mashao et al., "An Appraisal of the Progress in Utilizing Radiosondes and Satellites for Monitoring Upper Air Temperature Profiles," Atmosphere, vol. 15, no. 3, p. 387, Mar. 2024, doi: 10.3390/atmos15030387.

[9] C. Shyalika, R. Wickramarachchi, and A. P. Sheth, "A Comprehensive Survey on Rare Event Prediction," ACM Comput. Surv., vol. 57, no. 3, pp. 1–39, Mar. 2025, doi: 10.1145/3699955.

[10] A. Fawzy, A. Tahir, M. Galster, and P. Liang, "Exploring Data Management Challenges and Solutions in Agile Software Development: A Literature Review and Practitioner Survey," 2024, arXiv. doi: 10.48550/ARXIV.2402.00462.

[11] M. Dahoui, "Use of machine learning for the detection and classification of observation anomalies," 2023, doi: 10.21957/N64MD0XA5D.

[12] W. Xu, K. Chen, T. Han, H. Chen, W. Ouyang, and L. Bai, "ExtremeCast: Boosting Extreme Value Prediction for Global Weather Forecast," Aug. 16, 2024, arXiv: arXiv:2402.01295. doi: 10.48550/arXiv.2402.01295.

[13] C. Bodnar et al., "A Foundation Model for the Earth System," Nov. 21, 2024, arXiv: arXiv:2405.13063. doi: 10.48550/arXiv.2405.13063.

[14] I. Shumailov, Z. Shumaylov, Y. Zhao, N. Papernot, R. Anderson, and Y. Gal, "AI models collapse when trained on recursively generated data," Nature, vol. 631, no. 8022, pp. 755–759, 2024, doi: 10.1038/s41586-024-07566-y.

[15] F. Hamilton, T. Berry, and T. Sauer, "Correcting Observation Model Error in Data Assimilation," Chaos: An Interdisciplinary Journal of Nonlinear Science, vol. 29, no. 5, p. 053102, May 2019, doi: 10.1063/1.5087151.

[16] J. Liu et al., "Towards Out-Of-Distribution Generalization: A Survey," Jul. 27, 2023, arXiv: arXiv:2108.13624. doi: 10.48550/arXiv.2108.13624.

[17] Y. Yang et al., "A Survey on Diffusion Models for Time Series and Spatio-Temporal Data," 2024, arXiv. doi: 10.48550/ARXIV.2404.18886.
[18] N. Bormann, D. Dee, and M. Dahoui, "Observational bias correction in data assimilation and an overview of satellite data monitoring," Mar. 13, 2024. [Online]. Available: https://events.ecmwf.int/event/375/contributions/4263/attachments/2312/4041/Bormann_TC_2024_Bcor_SAT.pdf
[19] European Commission. Joint Research Centre., Robustness and explainability of Artificial Intelligence: from technical to policy solutions. LU: Publications Office, 2020. Accessed: Feb. 20, 2025. [Online]. Available: https://data.europa.eu/doi/10.2760/57493
[20] F. Rafii, H. Gonzalez-Velez, and A. E. Chis, "Automatic FAIR Provenance Collection and Visualization for Time Series," in Companion of the 2023 ACM/SPEC International Conference on Performance Engineering, Coimbra Portugal: ACM, Apr. 2023, pp. 331–336. doi: 10.1145/3578245.3585026.
[21] C. Lacagnina et al., "Quality Management Framework for Climate Datasets," CODATA, vol. 21, no. 1, p. 10, Apr. 2022, doi: 10.5334/dsj-2022-010.
[22] L. Chen, B. Han, X. Wang, J. Zhao, W. Yang, and Z. Yang, "Machine Learning Methods in Weather and Climate Applications: A Survey," Applied Sciences, vol. 13, no. 21, p. 12019, Nov. 2023, doi: 10.3390/app132112019.
[23] ECMWF, "Fact sheet: Earth system data assimilation." [Online]. Available: https://www.ecmwf.int/en/about/media-centre/focus/2020/fact-sheet-earth-system-data-assimilation
[24] P. Manshausen et al., "Generative Data Assimilation of Sparse Weather Station Observations at Kilometer Scales," 2024, arXiv. doi: 10.48550/ARXIV.2406.16947.
[25] T. McNally et al., "An update on AI–DOP: skilful weather forecasts produced directly from observations," 2025, doi: 10.21957/TMI6Y913DC.
[26] M. Alexe et al., "GraphDOP: Towards skilful data-driven medium-range weather forecasts learnt and initialised directly from observations," Dec. 20, 2024, arXiv: arXiv:2412.15687. doi: 10.48550/arXiv.2412.15687.
[27] M. Bonavita, A. Geer, P. Laloyaux, S. Massart, and M. Chrust, "Data assimilation or machine learning?" [Online]. Available: https://www.ecmwf.int/en/newsletter/167/meteorology/data-assimilation-or-machine-learning
[28] I. Amir, T. Koren, and R. Livni, "SGD generalizes better than GD (and regularization doesn't help)," in Conference on Learning Theory, PMLR, 2021, pp. 63–92.
[29] L. Llorenç, T. Haiden, J. Schröttle, and R. Forbes, "Scale-dependent verification of precipitation and cloudiness at ECMWF." [Online]. Available: https://www.ecmwf.int/en/newsletter/174/earth-system-science/scale-dependent-verification-precipitation-and-cloudiness
[30] I. Goodfellow et al., "Generative adversarial networks," Commun. ACM, vol. 63, no. 11, pp. 139–144, Oct. 2020, doi: 10.1145/3422622.
[31] N. Rampal, P. B. Gibson, S. Sherwood, G. Abramowitz, and S. Hobeichi, "A Reliable Generative Adversarial Network Approach for Climate Downscaling and Weather Generation," J Adv Model Earth Syst, vol. 17, no. 1, p. e2024MS004668, Jan. 2025, doi: 10.1029/2024MS004668.
[32] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning," PMLR, vol. PMLR 48, pp. 1050–1059, Oct. 2016, doi: 10.48550/arXiv.1506.02142.
[33] M. Belkin, D. Hsu, S. Ma, and S. Mandal, "Reconciling modern machine-learning practice and the classical bias–variance trade-off," Proceedings of the National Academy of Sciences, vol. 116, no. 32, pp. 15849–15854, 2019.

[34] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets," Advances in neural information processing systems, vol. 31, 2018.
[35] A. Bihlo, "A generative adversarial network approach to (ensemble) weather prediction," Neural Networks, vol. 139, pp. 1–16, 2021.
[36] C. Bülte, N. Horat, J. Quinting, and S. Lerch, "Uncertainty quantification for data-driven weather models," Mar. 20, 2024, arXiv: arXiv:2403.13458. doi: 10.48550/arXiv.2403.13458.
[37] C. Bülte, N. Horat, J. Quinting, and S. Lerch, "Uncertainty quantification for data-driven weather models," Mar. 20, 2024, arXiv: arXiv:2403.13458. doi: 10.48550/arXiv.2403.13458.
[38] M. C. A. Clare, M. Sonnewald, R. Lguensat, J. Deshayes, and V. Balaji, "Explainable Artificial Intelligence for Bayesian Neural Networks: Towards trustworthy predictions of ocean dynamics," May 2022, Oceanography. doi: 10.1002/essoar.10511239.1.
[39] A. Desai, M. Abdelhamid, and N. R. Padalkar, "What is Reproducibility in Artificial Intelligence and Machine Learning Research?," Apr. 29, 2024, arXiv: arXiv:2407.10239. doi: 10.48550/arXiv.2407.10239.
[40] O. E. Gundersen and S. Kjensmo, "State of the Art: Reproducibility in Artificial Intelligence," AAAI, vol. 32, no. 1, Apr. 2018, doi: 10.1609/aaai.v32i1.11503.
[41] T. Rackow et al., "Robustness of AI-based weather forecasts in a changing climate," Sep. 27, 2024, arXiv: arXiv:2409.18529. doi: 10.48550/arXiv.2409.18529.
[42] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under Concept Drift: A Review," IEEE Trans. Knowl. Data Eng., pp. 1–1, 2018, doi: 10.1109/TKDE.2018.2876857.
[43] ISO/IEC JTC 1/SC 42, "ISO/IEC TR 24029-1:2021 Artificial Intelligence (AI) — Assessment of the robustness of neural networks." ISO/IEC.
[44] ISO/IEC JTC 1/SC 42, "ISO IEC AWI TS 25223 Information technology — Artificial intelligence — Guidance and requirements for uncertainty quantification in AI systems."
[45] E. Gibney, "Could machine learning fuel a reproducibility crisis in science?," Nature, vol. 608, no. 7922, pp. 250–251, Aug. 2022, doi: 10.1038/d41586-022-02035-w.
[46] H. Semmelrock et al., "Reproducibility in Machine Learning-based Research: Overview, Barriers and Drivers," 2024, arXiv. doi: 10.48550/ARXIV.2406.14325.
[47] A. G. Patt and D. P. Schrag, "Using Specific Language to Describe Risk and Probability," Climatic Change, vol. 61, no. 1/2, pp. 17–30, 2003, doi: 10.1023/A:1026314523443.