Probabilistic Inference for Spatiotemporal Dynamics

Dissertation
der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Jonathan Schmidt
aus Filderstadt

Tübingen
2025

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	28.01.2026
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Philipp Hennig
2. Berichterstatter:	Prof. Dr. Philipp Berens

Abstract

Simulation and inference are cornerstones of scientific inquiry. Numerous domains and applications can integrate advanced machine-learning methods for reasoning from prior mechanistic knowledge and data. Probabilistic approaches take into account the uncertainty associated with observations and the physical model. Bayesian inference, however, remains challenging in practice. In particular, high-dimensional settings that revolve around spatiotemporal dynamical systems, such as atmospheric models for predicting weather and climate, are computationally demanding. This thesis considers two different avenues for integrating large physical systems into a Bayesian inference pipeline. On the one hand, assuming linear dynamics governing the latent system state and capturing mechanistic and empirical uncertainty in tractable Gaussian noise models allows optimal state estimation via the Kalman filter. On the other hand, generative diffusion models that learn probabilistic representations of complex nonlinear dynamics from existing trajectories can be leveraged as prior models in Bayesian inverse problems. For both approaches, open conceptual and practical questions are addressed.

High-dimensional state spaces create a gap between the theoretical merits of the Kalman filter and its practicability. Concretely, the explicit modeling of pairwise errors that characterizes Gaussian models impedes their application to large-scale systems due to limited memory and compute. Low-dimensional representations of the pairwise error covariances aim to resolve this dilemma. Sampling-based approaches are broadly applicable but introduce stochasticity into the inference algorithm. For linear systems with a Gauss–Markov process-noise model, we develop a fully deterministic alternative. The proposed algorithm formulates prediction, update, and smoothing routines for optimal low-rank approximations of the covariance matrices and converges to the exact Kalman filter in the full-rank limit.

Especially in scientific applications, imposing mechanistic constraints on deep-learning models increasingly blurs the line between inference and simulation. Recent research on generative diffusion models establishes the model class as powerful statistical representations of physical processes. Their probabilistic nature and post-training conditioning make diffusion models suitable priors in flexible Bayesian inference models. We investigate and demonstrate this approach in the context of downscaling climate simulations. The proposed generative downscaling framework generates spatiotemporally coherent trajectories that estimate fine-scale processes obscured by simulating climate on coarse grids. Experiments on coarse weather and climate input illustrate that the sampled trajectories accurately estimate several atmospheric variables, simultaneously, on a fine spatiotemporal grid. Sampling multiple predictions provides structured uncertainty. Lastly, we use this generative downscaling framework for local risk assessment of renewable-energy shortages based on existing simulations of future climate.

Zusammenfassung

Simulation und Inferenz sind Grundpfeiler wissenschaftlicher Untersuchungen. Zahlreiche Bereiche und Anwendungen können fortgeschrittene Methoden des maschinellen Lernens einbeziehen, um Schlussfolgerungen aus bestehendem mechanistischem Wissen und Daten zu ziehen. Probabilistische Ansätze berücksichtigen die mit Beobachtungen und dem physikalischen Modell verbundene Unsicherheit. Bayes'sche Inferenz stellt jedoch in der Praxis eine Herausforderung dar. Insbesondere hochdimensionale Probleme, die räumlich-zeitliche dynamische Systeme einbeziehen, wie beispielsweise atmosphärische Modelle zur Vorhersage von Wetter und Klima, sind rechenaufwendig. Diese Dissertation betrachtet zwei verschiedene Zugänge zur Einbindung großer physikalischer Systeme in Bayes'sche Inferenz. Einerseits ermöglicht die Annahme linearer Dynamik im latenten Systemzustand und die Modellierung mechanistischer und empirischer Unsicherheit als Gauß'sches Rauschen eine optimale Zustandsschätzung durch den Kalman-Filter. Andererseits können generative Diffusionsmodelle, welche probabilistische Repräsentationen komplexer nichtlinearer Dynamik aus bestehenden Trajektorien lernen, als A-priori Modelle in Bayes'schen Inversionsproblemen genutzt werden. Für beide Ansätze werden hier offene konzeptionelle und praktische Fragen behandelt.

Hochdimensionale Zustandsräume schaffen eine Kluft zwischen den theoretischen Vorzügen des Kalman-Filters und seiner Anwendbarkeit. Konkret erschwert die explizite Modellierung von paarweisen Fehlern, welche Gauß'sche Modelle charakterisieren, deren Anwendung auf große Systeme aufgrund begrenzter Speicher- und Rechenkapazitäten. Niedrigdimensionale Darstellungen der paarweisen Fehlerkovarianzen zielen darauf ab, dieses Dilemma zu lösen. Randomisierte Ansätze finden breite Anwendung, führen jedoch zu Stochastizität im Inferenzalgorithmus. Für lineare Systeme mit einem Gauß–Markov-Modell für Prozessrauschen entwickeln wir eine vollständig deterministische Alternative. Der vorgestellte Algorithmus formuliert Vorhersage-, Filter- und Smoother-Routinen für optimale Niedrigrang-Approximationen der Kovarianzmatrizen und konvergiert zum exakten Kalman-Filter im Vollrang-Limit.

Insbesondere in wissenschaftlichen Anwendungen verschwimmt zunehmend die Grenze zwischen Inferenz und Simulation durch die Auferlegung mechanistischer Beschränkungen in Deep-Learning-Modellen. Jüngste Forschungen zu generativen Diffusionsmodellen etablieren diese Modellklasse als leistungsfähige statistische Repräsentationen physikalischer Prozesse. Aufgrund ihrer probabilistischen Natur und ihrer Konditionierung nach dem Training eignen sich Diffusionsmodelle als A-priori Modelle in flexiblen Bayes'schen Inferenzmodellen. Wir untersuchen und demonstrieren diesen Ansatz im Zusammenhang der Herunterskalierung („*Downscaling*“) von Klimasimulationen. Der vorgestellte generative Downscaling-Algorithmus erzeugt räumlich-zeitlich kohärente Trajektorien, die feinskalige Prozesse schätzen, welche durch die Klimasimulationen auf groben Gittern verschleiert werden. Experimente mit groben Wetter- und Klima-Inputs zeigen, dass die generierten Trajektorien mehrere atmosphärische Variablen gleichzeitig auf einem feinen räumlich-zeitlichen Gitter mit hoher Genauigkeit abschätzen. Das Generieren mehrerer Vorhersagen liefert strukturierte Unsicherheit. Schließlich verwenden wir dieses generative Downscaling-Framework für die lokale Risikobewertung von Engpässen bei erneuerbaren Energien auf der Grundlage existierender Simulationen des zukünftigen Klimas.

Acknowledgments

I want to express my deepest gratitude to Philipp Hennig. The fact that I looked forward to each of our conversations, and that I always left your office more encouraged, more confident, and more relaxed, is something that I have never stopped appreciating. Thank you for not only allowing me, but encouraging me to travel and communicate my research. Thank you for shaping me without pressure, and for accompanying me and guiding me through this journey.

I further want to thank Todd Ehlers for his support and for many valuable discussions. I am grateful to Philipp Berens for investing his time and effort to read and evaluate this thesis. To my collaborators, Jörg Nick, Filip Tronarp, Luca Schmidt, Felix Strnad, Nicole Ludwig, and Fabian Mockert, thank you, for making this research possible.

I would like to thank my colleagues in the MoML group for many inspiring discussions and for making it a pleasure to come to the office. I am especially thankful to everyone who helped improve the text through their valuable feedback. I am indebted and immensely grateful to Nathanael, Marvin, Nina, and Christian for their company and their insights, for shared coffees, travels, and a lot of shared laughter. Thank you, Nico, for showing me that and how I can become a good researcher and, also to Filip, for patiently teaching me so much of what I know today. I am grateful to Franzi, who guided me through all administrative processes and who always offered me a friendly ear when I needed it. I am also grateful to the International Max-Planck Research School for Intelligent Systems (IMPRS-IS) for their support and for organizing great events and workshops.

I want to thank my family, my mother Judith, my father Klaus, my brother David, and my sister Tabitha, for always being there for me. To Flo and Betty, who left Tübingen right after I started my PhD and who were still with me throughout the process, thank you for everything. Thank you, Sami, for sticking with me since primary school. I am deeply grateful that we have had (varyingly sensible) discussions for over 23 years now. Finally, to Laura, thank you for supporting and encouraging me, for challenging me and inspiring me and for being the kind and wonderful person that you are.

Jonathan Schmidt
Tübingen, October 28, 2025

Table of Contents

PART I

Introduction

> "In all science, error precedes the truth, and
> it is better it should go first than last."
> - Horace Walpole

1 Overview

Recent history has been shaped by rapid technological developments, particularly the increase in computational power and the capacity to acquire, store, distribute, and process vast amounts of data. Machine learning (ML) research continues to thrive under these conditions, not least due to data-driven ML models simultaneously benefitting from and driving this technological progress. The growing reservoir of learning algorithms comes with the enticing promise of automated extraction of knowledge from bulks of information that cease to be comprehensible by mere humans. However, in many scientific applications of ML, extensive prior knowledge is available. Taking into account the mechanistic nature of a data-generating process can introduce structural constraints that enable more robust and interpretable inference. Integrating knowledge about the structure of a problem into the design of algorithms can improve efficiency and reduce computational cost. The question of how to integrate different types of information into an inference model can be approached from different perspectives. The challenges inherent in high-dimensional settings, in particular, both motivate advancing existing algorithms and inspire developing and leveraging innovative methodology. In this light, this thesis will explore two complementary approaches to probabilistic modeling and inference for spatiotemporal dynamics.

`"We are drowning in information but starved for knowledge."`
`- John Naisbitt`

Sequences of data, *time series*, can be assumed to be generated through partially observing a latent, time-evolving state of a dynamical system. In the context of inference, assuming such an underlying process assigns a physical interpretation to the space between and beyond observations. Moreover, through mechanistic laws, knowledge about physical dependencies can be introduced—for instance, relationships between observed and unobserved quantities.

Dynamical systems are relevant in various applications in science and engineering, for example, in neuroscience [96, 100], disease modeling [93, 143], and atmospheric dynamics [14, 167, 17], to name only a few. The evolution of dynamical systems is commonly described by systems of *differential equations*. Through simulation, equations are materialized into state trajectories. This requires repeated numerical computations that move the system state through the (potentially vast) space of possible configurations. Many practical settings—such as atmospheric modeling for weather forecasting and climate simulation—assume dynamics that are of such conceptual and computational complexity that it is open to question how they are best integrated into an inference algorithm.

One approach is to simplify the model. While this can make inference tractable and more efficient, simplifying assumptions naturally introduce model error that should be accounted for. In fact, errors and uncertainty arise at various levels: in data collection (sensor noise, partial observations), modeling (simplifications, partial understanding), and even in computation (finite precision, numerical approximations). In his foundational book, Murphy [154] motivates probabilistic modeling as "the language used by most other areas of science and engineering" and, crucially, as "a unifying framework between these fields". The *Bayesian* framework phrases learning and reasoning under uncertainty as updating a *prior* belief over the current state of knowledge on new information. Observed quantities are linked to the unknown variables through an *observation* or *forward model*, often called *likelihood*. Together, prior and likelihood specify a joint model for the generative process that is assumed to underlie the observed variables. Upon encountering new information, the prior belief is updated by *conditioning* it on the incoming observations. Through this process, we obtain a *posterior* belief. Computing the posterior requires re-normalizing the joint distribution with respect to the marginal distribution of the data, which is called *evidence*:

$$\text{posterior} = \frac{\overbrace{\text{prior} \times \text{likelihood}}^{\text{joint model}}}{\text{evidence}} \, .$$

The computation of the evidence requires summing or integrating over the entire space of possible configurations of the unknown variables. For systems of many unknowns, this high-dimensional operation poses a prohibitive computational bottleneck and makes careful model and algorithm design necessary for practical applications of Bayesian methods.

For scientific applications, advanced ML tools offer promising approaches to efficiently represent complex and high-dimensional phenomena. Statistical inference algorithms are increasingly used for, or together with, scientific computation. In particular, structured *deep-learning* models [75]—such as, for instance, deep operator networks (DeepONets) [135], graph neural networks (GNNs) [183, 124], vision transformers (ViTs) [47, 156], and Fourier neural operators (FNOs) [118, 213]—already replace numerical simulation in certain prediction tasks. One notable landmark in this development is the operational deployment of ML-based weather forecasting systems by the European Centre for Medium-Range Weather Forecasts (ECMWF) [125, 50, 51]. Learning a statistical surrogate model for a physical system is attractive in practice for multiple reasons. One central aspect is the amortization of computational cost. While training a model is computationally intensive, once trained, it can be used as a cost-efficient substitute, ideally at a fraction of the cost of the original simulator. This is particularly advantageous when the mechanistic system needs to be queried repeatedly, for instance, within an inference loop. Moreover, a trained model can often be integrated easily into inference pipelines. For example, gradient-based optimization or sampling methods often rely on automatic differentiation [222, 78, 15]. When taking derivatives of numerical simulators is infeasible, differentiable statistical surrogate models can be a promising remedy.

1.1 Contributions

Incorporating mechanistic knowledge into an inference loop remains a challenge for large physical systems with high-dimensional state spaces. Such high-dimensional settings create a disparity between the theory of Bayesian inference and its practical applicability. This thesis aims to address challenges with two different approaches to probabilistic inference involving spatiotemporal dynamical systems. The first line of work investigates linear systems combined with Gaussian process-noise models that allow closed-form optimal state estimation. Simplifying model assumptions allow a principled treatment and make time-series inference more tractable. Nevertheless, exact inference becomes infeasible as the dimensionality of the problem increases, calling for algorithmic solutions tailored to the specific problem setting. The second line of work explores advanced ML approaches for Bayesian inference. By learning powerful and efficient representations of complex patterns from data, deep-learning models promise a flexible alternative to explicit (physical) modeling. *Probabilistic* representations can thereby serve as rich priors in Bayesian models. Thus promising to bridge the practicability-gap of probabilistic inference, modern ML models need to be developed, refined, and evaluated in the context of real-world applications. The contributions are outlined in the following.

Informing a mechanistic prior model with data

State-space models (SSMs) explicitly model a latent, time-evolving state, from which time-series are generated through partial, noisy observation. The goal is to obtain estimates about this latent process from the observed time series. In time-continuous formulations, differential equations provide deterministic models for the state dynamics [181]. Unknown forcings or uncertainty about the system dynamics are captured by a stochastic process. Linking the latent state to measurements via an observation model (likelihood) enables posterior inference over the state through Bayesian conditioning.

For general dynamics and noise models, SSMs share the computational challenges that are commonly entailed by Bayesian methods. Assuming a linear dynamics model and a Gaussian noise model, however, offers conceptual and practical benefits. In particular, Gauss–Markov (GM) priors that are observed through a linear Gaussian likelihood allow rigorous theoretical analysis and linear-in-time inference via the Kalman filter (KF) [105, 179]. However, the computational cost of KF inference scales prohibitively with the state dimension, which makes the exact algorithm unsuitable for very high-dimensional systems. In large-scale applications of Gaussian filtering, a common approximation constructs and propagates a moderately-sized ensemble of states instead of explicit densities [58, 31]. This implies a dimensionality reduction of the representation of pairwise errors and alleviates computational cost and memory requirements. The sampling also introduces stochasticity into the algorithm. Monte–Carlo approximations fail to recover the optimal state estimate obtained through the KF for finite ensemble sizes. For GM inference, a deterministic alternative can be constructed, which addresses this issue. Chapter 3 develops and evaluates a corresponding algorithm.

Disclaimer 1.1 Chapter 3 is based on the peer-reviewed conference publication with the following co-author contributions:

[185] Jonathan Schmidt, Philipp Hennig, Jörg Nick, and Filip Tronarp. "The Rank-Reduced Kalman Filter: Approximate Dynamical-Low-Rank Filtering In High Dimensions". In: *Advances in Neural Information Processing Systems (NeurIPS)*. vol. 36. 2023, pp. 61364–61376

	Ideas	**Experiments**	**Analysis**	**Writing**
J. Schmidt	40 %	80 %	60 %	45 %
P. Hennig	5 %	5 %	10 %	5 %
J. Nick	10 %	5 %	10 %	5 %
F. Tronarp	45 %	10 %	20 %	45 %

Informing a data-driven prior model with mechanistic constraints

This approach aligns with a broader trend in modern scientific ML aimed at integrating the structural constraints of physical systems into data-driven inference pipelines. Some data-driven ML models that are effectively used for scientific computation have already been mentioned above. Generative *diffusion models* (DMs) [194, 198, 94, 199] have gained particular interest due to their capability of learning *probabilistic* representations of complex spatial relationships. DMs learn to generate samples from complex target distributions. Equipped with a conditioning mechanism, DMs can function as rich prior beliefs in Bayesian inverse problems [66, 142]. For Gaussian likelihoods, the added complexity of conditioning the prior DM is negligible [39, 174].

We will encounter this model class in a concrete applied setting: statistical downscaling. The goal of downscaling is to estimate fine-scale weather dynamics from coarse climate simulations. We will mainly address two shortcomings that are frequently encountered in statistical approaches to downscaling: (i) considering only a single atmospheric variable, and (ii) operating on the coarse temporal grid of the climate simulations. Our approach is presented in Chapter 4 and phrases statistical downscaling as a Bayesian inverse problem to obtain probabilistic estimates.

Disclaimer 1.2 Chapter 4 is based on the peer-reviewed journal publication with the following co-author contributions:

[186] Jonathan Schmidt, Luca Schmidt, Felix M. Strnad, Nicole Ludwig, and Philipp Hennig. "A Generative Framework for Probabilistic, Spatiotemporally Coherent Downscaling of Climate Simulation". In: *npj Climate and Atmospheric Science* 8.1 (2025), p. 270

	Ideas	**Experiments**	**Analysis**	**Writing**
J. Schmidt	50 %	50 %	35 %	60 %
L. Schmidt	20 %	30 %	30 %	10 %
F. M. Strnad	10 %	10 %	15 %	15 %
N. Ludwig	10 %	5 %	10 %	10 %
P. Hennig	10 %	5 %	10 %	5 %

Climate simulations on coarse spatiotemporal grids provide an estimate for global long-term trends that allow an approximate, large-scale impact assessment. Downscaling is crucial for the assessment of *local* effects of future climatic conditions. These effects can affect various aspects of daily life, for example, power generated from renewable energy sources. The efficacy of wind turbines and photovoltaic power plants, naturally, depend crucially on wind speeds and solar irradiance. In particular, so-called *dark doldrums*, known as “Dunkelflauten” in German, can lead to prolonged energy shortages during periods of slow winds and low sunlight. Future climate projections provide means to simulate hypothetical future scenarios that allow to estimate the impact of Dunkelflaute events on the renewable-energy output. Local impact analysis is, however, impossible for climate simulations on coarse spatial grids. Additionally, unresolved fine-scale dynamics might lead to inaccurate estimates in wind- and solar-power generation. Building directly on the framework presented in Chapter 4, the follow-up study in Chapter 5 investigates whether probabilistic spatiotemporal downscaling can aid the risk assessment regarding future Dunkelflaute events. The generative approach allows a probabilistic treatment of this relevant inquiry, which accounts for uncertainty in estimating local weather patterns from coarse climate projections.

Disclaimer 1.3 Chapter 5 is based on the manuscript in preparation with the following co-author contributions:

[203] Felix Strnad, Jonathan Schmidt, Fabian Mockert, Philipp Hennig, and Nicole Ludwig. “Assessing the risk of future Dunkelflaute events for Germany using generative deep learning”. In: *arXiv preprint arXiv:2509.24788* (2025)

	Ideas	**Experiments**	**Analysis**	**Writing**
F. M. Strnad	40 %	55 %	50 %	60 %
J. Schmidt	25 %	30 %	5 %	10 %
F. Mockert	5 %	5 %	10 %	10 %
P. Hennig	5 %	5 %	5 %	5 %
N. Ludwig	25 %	5 %	30 %	15 %

For this preliminary work, **J. Schmidt** mainly contributed to the generative downscaling model (setup, extensions, training, writing) that underlies the analysis.

2 Background

2.1 Bayesian inference

Bayesian inference [18, 71, 22] provides a general framework for reasoning under uncertainty. At its core, it builds on elementary rules of probability calculus:

- *Conditional probability:*
$$p(\mathrm{X} \mid \mathrm{Y}) = \frac{p(\mathrm{X}, \mathrm{Y})}{p(\mathrm{Y})} \tag{2.1}$$
- *Sum rule:*
$$p(\mathrm{X}) = \int p(\mathrm{X}, \mathrm{Y}) \mathrm{dY} \tag{2.2}$$
- *Product rule:*
$$p(\mathrm{X}, \mathrm{Y}) = p(\mathrm{X})p(\mathrm{Y} \mid \mathrm{X}) = p(\mathrm{Y})p(\mathrm{X} \mid \mathrm{Y}) \tag{2.3}$$

Combining the above yields *Bayes' rule*:

$$p(\mathrm{X} \mid \mathrm{Y}) = \frac{p(\mathrm{X})p(\mathrm{Y} \mid \mathrm{X})}{p(\mathrm{Y})} = \frac{p(\mathrm{X})p(\mathrm{Y} \mid \mathrm{X})}{\int p(\mathrm{X})p(\mathrm{Y} \mid \mathrm{X}) \mathrm{dX}} \,. \tag{2.4}$$

In the context of inference problems, the densities in Bayes' rule are assigned a specific semantic meaning. Let us begin by considering X to be an *unknown*, or *latent*, quantity of interest and Y an *observed* quantity. The numerator in Bayes' rule is a joint distribution $p(\mathrm{X}, \mathrm{Y})$ over the latent and observed variables. This joint model is sometimes referred to as the "*generative model*", as it captures the process of generating observations Y from the unknown variables X. The joint factorizes into the product of a *prior* belief $p(\mathrm{X})$ over the unknown quantity and a *likelihood* $p(\mathrm{Y} \mid \mathrm{X})$, see the associated graphical model in Figure 2.1. We will see later (Sections 2.2.1 and 2.5) that additional structure can be imposed on a Bayesian model.

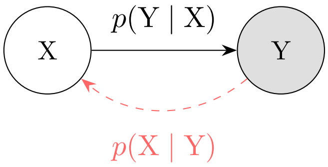


Figure 2.1: **Bayesian inference.** This graphical model depicts a factorization of the joint distribution $p(\mathrm{X}, \mathrm{Y}) = p(\mathrm{X})p(\mathrm{Y} \mid \mathrm{X})$. In the context of Bayesian inference, we call $p(\mathrm{X})$ the *prior* and $p(\mathrm{Y} \mid \mathrm{X})$ the *likelihood* and assume both to be known. The inverse problem yields the *posterior* $p(\mathrm{X} \mid \mathrm{Y})$ through conditioning the prior on observations.

The prior models the current state of information over the unknown quantity, for instance, previously processed evidence or prior knowledge. The likelihood is also often referred to as *observation* or *forward* model, as it defines how the observed variable relates to (or “is generated from”) the unknown variable. Accordingly, the *inverse problem* is to compute the conditional distribution over the unknown variable, the *posterior*, given the observations. Posterior inference amounts to normalizing the joint distribution with respect to the *evidence*, the marginal distribution over the observed variable Y. This is usually a computational bottleneck in practical applications of Bayesian inference due to intractable integrals over all possible configurations of X. The computational burden entailed by computing a normalized density motivates a diverse reservoir of tools that address and alleviate this practical issue. Notable approaches include variational methods [104, 101, 137, 109] and Monte–Carlo (sampling) methods [133, 171, 48, 26]. We now introduce *Gaussian* models for exact posterior inference.

2.2 Gaussian inference and the Kalman filter

The probability density function of the multivariate *normal* or *Gaussian* distribution is given by

$$p(\mathrm{X}) = \mathcal{N}(\mathrm{X}; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2}\,|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathrm{X}-\mu)^{\mathsf{T}}\Sigma^{-1}(\mathrm{X}-\mu)\right), \tag{2.5}$$

for $\mathrm{X} \in \mathbb{R}^n$, with $|\cdot|$ denoting the matrix determinant. The Gaussian distribution is fully parametrized by its first two moments: the *mean* $\mu \in \mathbb{R}^n$ and the symmetric and positive-definite (SPD) *covariance* matrix $\Sigma \in \mathbb{R}^{n\times n}$. The mean coincides with the mode and median of the Gaussian distribution and the covariance encodes the pairwise correlations between the components of X. A common way to quantify Gaussian uncertainty is to report the marginal error variances, which lie on the diagonal of the covariance matrix.

A linear Gaussian model for inference defines

1. a Gaussian prior belief over the unknown quantity $\mathrm{X} \in \mathbb{R}^n$

$$p(\mathrm{X}) = \mathcal{N}\big(\mathrm{X}; \mu^-, \Pi\big), \tag{2.6}$$

2. a linear Gaussian observation model that encodes the assumption that measurements $\mathrm{Y} \in \mathbb{R}^m$ are linear transformations of X, corrupted by additive Gaussian noise

$$p(\mathrm{Y} \mid \mathrm{X}) = \mathcal{N}(\mathrm{Y}; H\mathrm{X}, R), \tag{2.7}$$

for a measurement matrix $H \in \mathbb{R}^{m\times n}$ and measurement-error covariance $R \in \mathbb{R}^{m\times m}$.

For this model, the posterior distribution $p(\mathrm{X} \mid \mathrm{Y})$ is a Gaussian $\mathcal{N}(\mathrm{X}; \mu, \Sigma)$, with

$$\begin{aligned} S &= H \Pi H^{\mathsf{T}} + R, && (2.8)\\ K &= \Pi H^{\mathsf{T}} S^{-1}, && (2.9)\\ \mu &= \mu^{-} + K\big(\mathrm{Y} - H\mu^{-}\big), && (2.10)\\ \Sigma &= \Pi - K S K^{\mathsf{T}}. && (2.11) \end{aligned}$$

Furthermore, the marginal likelihood of Y is given by

$$p(\mathrm{Y}) = \mathcal{N}\big(\mathrm{Y}; H\mu^{-}, S\big). \tag{2.12}$$

2.2.1 State-space models

Linear Gaussian inference translates from static to dynamic models, providing a principled framework for time-series inference. A linear Gaussian *state-space model* (SSM) is constructed from

1. a *transition model* that defines the prior belief over the dynamics of the hidden state $\mathrm{X}(t) \in \mathbb{R}^n$. Consider a linear Itô stochastic differential equation (SDE) [159, 181]

$$\mathrm{dX}(t) = A\mathrm{X}(t)\mathrm{d}t + B\mathrm{dW}(t), \qquad \mathrm{X}(0) \sim \mathcal{N}(\mu_0, \Sigma_0) \tag{2.13}$$

driven by Brownian motion $\mathrm{W}(t) \in \mathbb{R}^q$ and subject to a Gaussian initial distribution, specified by initial mean $\mu_0 \in \mathbb{R}^n$ and covariance $\Sigma_0 \in \mathbb{R}^{n \times n}$. Since the *drift* matrix $A \in \mathbb{R}^{n \times n}$ and the *dispersion* matrix $B \in \mathbb{R}^{n \times q}$ are constant with respect to time, we call the transition model linear and time invariant (LTI). The LTI SDE in Equation (2.13) is solved by a Gaussian process (GP) with the Markov property—or Gauss–Markov (GM) process. In particular, for the solution process $\{\mathrm{X}(t); t \geq 0\}$, it holds that

   (i) any finite collection $\{\mathrm{X}(t_l)\}_{l=1}^{N}$ is jointly Gaussian distributed [168, 181], and
   (ii) $p(\mathrm{X}(t_l) \mid \{\mathrm{X}(t_k)\}_{k \neq l}) = p(\mathrm{X}(t_l) \mid \mathrm{X}(t_{l-1}))$ for all $k, l = 1, \dots, N$.

> **Remark 2.1** We will henceforth regularly use the short-hand notations
>
> $$\bullet_l := \bullet(t_l) \quad \text{and} \quad \bullet_{k:l} := (\bullet_k, \dots, \bullet_l)$$
>
> throughout this work, when considering finite evaluations of continuous (time-dependent) objects $\bullet$.

2. an *observation model* that relates measurements $\mathrm{Y}_l \in \mathbb{R}^m$ at times t_l linearly to X_l, through

$$\mathrm{Y}_l \mid \mathrm{X}_l \sim \mathcal{N}(H\mathrm{X}_l, R), \quad l = 1, \dots, N, \tag{2.14}$$

where (as in the static case, Equation (2.7)) $H \in \mathbb{R}^{m \times n}$ is a measurement matrix and $R \in \mathbb{R}^{m \times m}$ is the covariance matrix of the additive Gaussian observation noise.

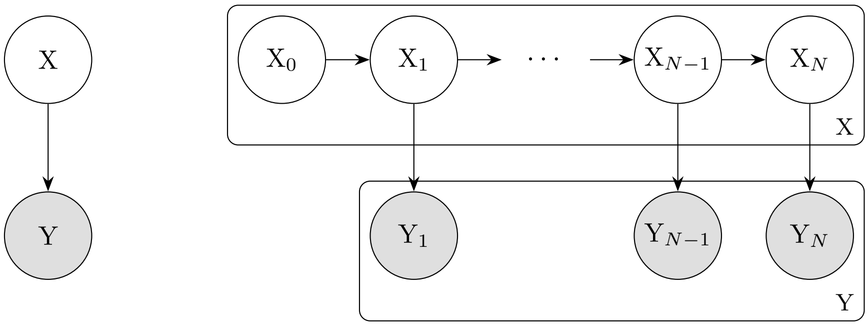


Figure 2.2: **Bayesian state estimation.** For time-series inference, we assume a latent time-evolving state X_l. The prior over the state dynamics is a Markov process, which means that the next state X_{l+1} is independent of all previous states $X_{0:l-1}$, given the present state X_l. Observations Y_l of the state at time t_l are conditionally independent of all past and future states and observations, given X_l.

The transition model Equation (2.13) admits an equivalent discrete formulation as the linear Gaussian transition density

$$p(X_l \mid X_{l-1}) = \mathcal{N}\big(X_l; \Phi_{l|l-1} X_{l-1}, Q_{l|l-1}\big), \tag{2.15}$$

with *transition matrix* $\Phi_{l|l-1} \in \mathbb{R}^{n \times n}$ and *process noise covariance* $Q_{l|l-1} \in \mathbb{R}^{n \times n}$. For LTI systems, the discretized transition model can be derived from Equation (2.13) in closed form, for instance, using the matrix-fraction decomposition [201, 181]. It is worth noting that $\Phi_{l|l-1}$ and $Q_{l|l-1}$ both solve matrix-valued linear differential equations on the open time interval $[t_{l-1}, t_l)$. More details follow in Chapter 3, for which this property is relevant.

Example 2.1 (Matérn covariance function) Gauss–Markov processes that are realizable as the solution to LTI SDEs of the form given in Equation (2.13) include the family of *Matérn processes* [181]. Let ν be a positive half-integer. The Matérn-ν covariance function

$$\mathcal{M}_\nu^{(\sigma,\ell)}(t,s) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{|t-s|}{\ell}\right)^\nu \mathrm{K}_\nu\left(\sqrt{2\nu}\frac{|t-s|}{\ell}\right), \tag{2.16}$$

allows to include prior knowledge over the smoothness of the process through ν and over its characteristic length scale through $\ell > 0$. The output scale is governed by $\sigma > 0$. Γ denotes the gamma function and K_ν the modified Bessel function of the second kind.

2.2.2 The Kalman filter

For the LTI SSM in Equations (2.13) and (2.14), Gaussian estimates of the hidden state $X(t)$ can be computed in closed form. The *Kalman filter* (KF) [105, 179] recursively computes conditional distributions $p(X_l \mid Y_{1:l})$ over the state at time t_l given all data points up to that time. The filtering recursions start at time t_0 from an initial Gaussian distribution

$$p(X_0) = \mathcal{N}(X_0; \mu_0, \Sigma_0), \tag{2.17}$$

with initial mean $\mu_0 \in \mathbb{R}^n$ and covariance $\Sigma_0 \in \mathbb{R}^{n \times n}$. For the subsequent time steps $l = 1, \ldots, N$, the filter follows a prediction-update scheme as follows:

1. **Predict:** Compute a predictive distribution (think: prior) over the state, given all *previous* measurements
$$p(\mathrm{X}_l \mid \mathrm{Y}_{1:l-1}) = \mathcal{N}\big(\mathrm{X}_l; \mu_l^-, \Pi_l\big), \tag{2.18}$$
with
$$\mu_l^- = \Phi_{l|l-1}\mu_{l-1} \tag{2.19}$$
$$\Pi_l = \Phi_{l|l-1}\Sigma_{l-1}\Phi_{l|l-1}^\mathsf{T} + Q_{l|l-1}. \tag{2.20}$$
The transition matrix $\Phi_{l|l-1}$ and process-noise covariance $Q_{l|l-1}$ are determined by the prior transition model, recall Equation (2.15).

2. **Update:** Compute the filtering distribution, updating the predicted state on the next observation Y_l at time t_l according to the measurement model Equation (2.14):
$$S_l = H\Pi_l H^\mathsf{T} + R, \tag{2.21}$$
$$K_l = \Pi_l H^\mathsf{T} S_l^{-1}, \tag{2.22}$$
$$\mu_l = \mu_l^- + K_l\big(\mathrm{Y}_l - H\mu_l^-\big), \tag{2.23}$$
$$\Sigma_l = \Pi_l - K_l S_l K_l^\mathsf{T} \tag{2.24}$$
$$= (I_n - K_l H)\Pi_l \tag{2.25}$$
$$= (I_n - K_l H)\Pi_l (I_n - K_l H)^\mathsf{T} + K_l R K_l^\mathsf{T}, \tag{2.26}$$
where I_n denotes the $n \times n$ identity matrix. The *Joseph form* of the covariance update (Equation (2.26)) avoids computing the difference between positive definite matrices and is thus less prone to numerical instabilities.

The marginal likelihood of the data sequence $\mathrm{Y}_{1:l}$ factorizes, via the *prediction error decomposition* [188], into
$$p(\mathrm{Y}_{1:l}) = p(\mathrm{Y}_1) \prod_{k=2}^{l} p(\mathrm{Y}_k \mid \mathrm{Y}_{1:k-1}), \tag{2.27}$$
where, in the case of Kalman filtering,
$$p(\mathrm{Y}_k \mid \mathrm{Y}_{1:k-1}) = \mathcal{N}\big(\mathrm{Y}_k; H\mu_k^-, S_k\big). \tag{2.28}$$

Smoothing

In the context of Gaussian filtering, *smoothing* means informing the filtering distribution at time t_l about all available data points, including those observed after t_l [110, 179]. Concretely, for a sequence of data points $\mathrm{Y}_{1:N}$, the smoother computes a Gaussian posterior $p(\mathrm{X}_l \mid \mathrm{Y}_{1:N})$ for all $l = 0, \ldots, N$. For the LTI SSM above, obtaining this posterior is again tractable and

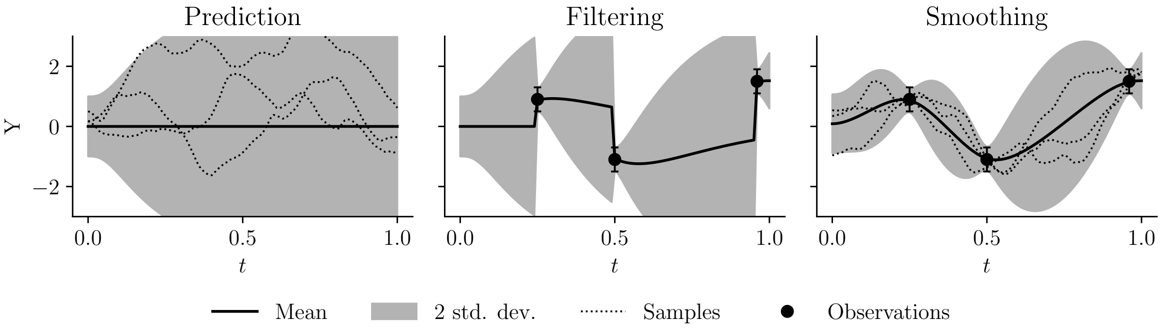


Figure 2.3: **Inference in Gauss–Markov models.** *Left:* Prediction with Matérn-3/2 prior dynamics. *Middle:* At three time points, the prior is updated on data. *Right:* The smoothed solution yields the posterior state estimate, having observed all data points. The Gaussian uncertainty around the mean state estimates (solid lines) is reported in terms of two marginal standard deviations (shaded areas) and three samples (dotted lines).

can be computed in closed form via a recursive algorithm [169]. The smoothing recursions can be written in terms of a *backward* Markov kernel, which is constructed from quantities that are already computed during the filtering recursions. Concretely, let

$$p(\mathrm{X}_l \mid \mathrm{X}_{l+1}) = \mathcal{N}\big(\mathrm{X}_l; \Psi_{l|l+1}\mathrm{X}_{l+1} + v_l, \Lambda_{l|l+1}\big) \tag{2.29}$$

be a Gaussian backwards kernel with a matrix $\Psi_{l|l+1} \in \mathbb{R}^{n\times n}$, a vector $v_l \in \mathbb{R}^n$, and $\Lambda_{l|l+1} \in \mathbb{R}^{n\times n}$ the transition covariance of the backward kernel. If we let

$$\Psi_{l|l+1} = \Sigma_l \Phi_{l+1|l}^{\mathsf{T}} \Pi_{l+1}^{-1}, \tag{2.30}$$

$$v_l = \mu_l - \Psi_{l|l+1}\mu_{l+1}^{-}, \tag{2.31}$$

$$\Lambda_{l|l+1} = \big(I_n - \Psi_{l|l+1}\Phi_{l+1|l}\big)\Sigma_l\big(I_n - \Psi_{l|l+1}\Phi_{l+1|l}\big)^{\mathsf{T}} + \Psi_{l|l+1}Q_{l+1|l}\Psi_{l|l+1}^{\mathsf{T}}, \tag{2.32}$$

then predicting *backwards in time* with Equation (2.29) computes the smoothing posterior [30]. The smoothing recursion starts at time t_N from

$$p(\mathrm{X}_N \mid \mathrm{Y}_{1:N}) = \mathcal{N}(\mathrm{X}_N; \mu_N^s, \Sigma_N^s) = \mathcal{N}(\mathrm{X}_N; \mu_N, \Sigma_N), \tag{2.33}$$

which coincides with the final-step filtering distribution. Then, for $l = N-1, N-2, \dots, 0$, the Gaussian smoothing posterior is

$$p(\mathrm{X}_l \mid \mathrm{Y}_{1:N}) = \mathcal{N}(\mathrm{X}_l; \mu_l^s, \Sigma_l^s), \tag{2.34}$$

with smoothing moments

$$\mu_l^s = \Psi_{l|l+1}\mu_{l+1}^s + v_l, \tag{2.35}$$

$$\Sigma_l^s = \Psi_{l|l+1}\Sigma_{l+1}^s\Psi_{l|l+1}^{\mathsf{T}} + \Lambda_{l|l+1}. \tag{2.36}$$

2.3 The Kalman filter in practice

Implementations of the KF in terms of the above equations face two practical challenges that involve the calculations of and with covariance matrices. The first is numerical stability and the second is computational cost. In the following, we outline how both challenges can be addressed using suitable representations of covariance matrices.

2.3.1 Square-root filtering for numerical stability

The SPD property of the covariance matrices in the KF can be lost due to finite-precision computations. These numerical instabilities can be detrimental to Gaussian state estimation in practice. The issue of numerical stability can be addressed through computations on square-root factorizations of covariance matrices.

Let $\Lambda \in \mathbb{R}^n$ be an SPD matrix and let a (left) matrix square root $\Lambda^{1/2}$ be given, such that $\Lambda = \Lambda^{1/2}\Lambda^{\mathsf{T}/2}$. The filtering recursions can be carried out entirely on the square roots of all involved covariance matrices without needing to compute the full matrix, which improves robustness of the filtering computations. Central to both the predict and update step in square-root filtering is the computation of sums of SPD matrices (cf. Equations (2.20), (2.21) and (2.26))

$$\Lambda \coloneqq MPM^{\mathsf{T}} + C, \tag{2.37}$$

where $M \in \mathbb{R}^{n\times p}$ is some matrix, and $P \in \mathbb{R}^{p\times p}$ and $C \in \mathbb{R}^{n\times n}$ are SPD matrices. Assume that we have $P = P^{1/2}P^{\mathsf{T}/2}$ and $C = C^{1/2}C^{\mathsf{T}/2}$ given as square-root factorizations, already. The goal is to compute the square-root factor $\Lambda^{1/2}$ of the above sum, directly, without constructing any full covariance matrix.

The square root of the Gramian MPM^{T} is given by

$$MPM^{\mathsf{T}} = MP^{1/2}P^{\mathsf{T}/2}M^{\mathsf{T}} = \left(MP^{1/2}\right)\left(MP^{1/2}\right)^{\mathsf{T}}. \tag{2.38}$$

Thus, $\left(MP^{1/2}\right) \in \mathbb{R}^{n\times p}$ is a matrix square root of the first summand.

The square-root factorization of the sum is obtained by noting that Equation (2.37) can be written as a product of block matrices. Let $G^{1/2} \coloneqq \left(MP^{1/2}\right)$. Then,

$$\Lambda = G^{1/2}G^{\mathsf{T}/2} + C^{1/2}C^{\mathsf{T}/2} = \begin{bmatrix} G^{1/2} & C^{1/2} \end{bmatrix} \begin{bmatrix} G^{\mathsf{T}/2} \\ C^{\mathsf{T}/2} \end{bmatrix} = \begin{bmatrix} G^{1/2} & C^{1/2} \end{bmatrix} \begin{bmatrix} G^{1/2} & C^{1/2} \end{bmatrix}^{\mathsf{T}}. \tag{2.39}$$

The block matrix $\begin{bmatrix} G^{1/2} & C^{1/2} \end{bmatrix} \in \mathbb{R}^{n\times(n+p)}$ is thus a matrix square root factor of Λ. An $n \times n$ square-root factor can be obtained by computing an orthogonal transformation. We illustrate this by means of the *singular-value decomposition* (SVD), which yields[1]

$$\begin{bmatrix} G^{1/2} & C^{1/2} \end{bmatrix} = UDV^{\mathsf{T}}, \tag{2.40}$$

---

[1] We consider the *thin* SVD here (see, e.g., Golub et al. [74, Section 2.4.3]).

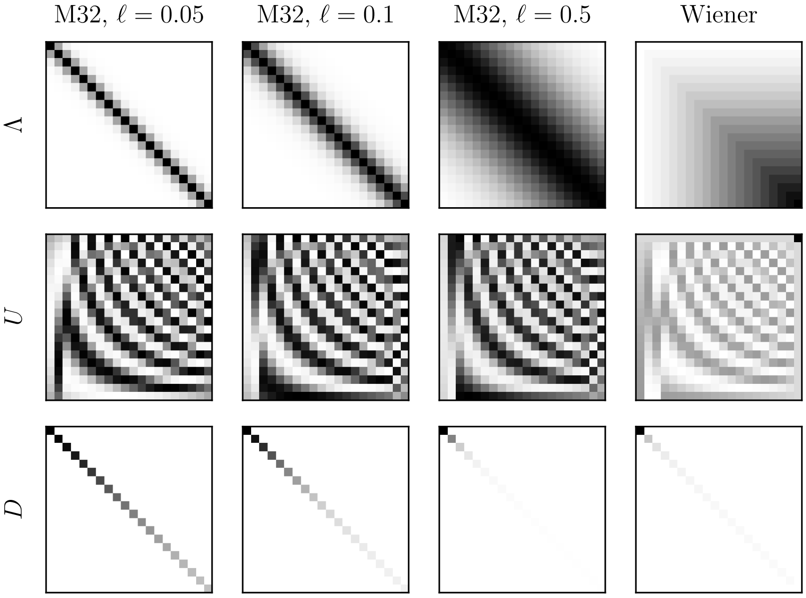


Figure 2.4: **Square-root factorizations of covariances.** White/black squares indicate low/high values. The first three columns correspond to Matérn-3/2 (M32) kernels with varying characteristic length scales $\ell = 0.05, 0.1, 0.5$; the fourth column to a Wiener kernel. The first row shows the full covariances (Λ). The second and third rows show U and D such that $\Lambda = (UD)(UD)^\mathsf{T}$ (cf. Equations (2.40) to (2.44)). For the Matérn covariance, the singular values on the diagonal of D decay at different rates, depending on the length scale ℓ.

where $U \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{(n+p)\times n}$ are orthogonal matrices, i.e., $U^\mathsf{T}U = UU^\mathsf{T} = I = VV^\mathsf{T} = V^\mathsf{T}V$. $D \in \mathbb{R}^{n\times n}$ is a diagonal matrix with the non-negative singular values as its diagonal elements. We have

$$\Lambda = \begin{bmatrix} G^{1/2} & C^{1/2} \end{bmatrix} \begin{bmatrix} G^{1/2} & C^{1/2} \end{bmatrix}^\mathsf{T} \tag{2.41}$$

$$= UDV^\mathsf{T} \left(UDV^\mathsf{T} \right)^\mathsf{T} \tag{2.42}$$

$$= UD \underbrace{V^\mathsf{T} V}_{=I} DU^\mathsf{T} \tag{2.43}$$

$$= (UD)(UD)^\mathsf{T}. \tag{2.44}$$

Thus, $UD \in \mathbb{R}^{n\times n}$ is also a valid, $n \times n$ matrix square root of $MPM^\mathsf{T} + C$.

2.3.2 Low-rank filtering for computational tractability

Through the conditional-independence assumptions encoded in SSMs (recall Equations (2.13) and (2.14)), the complexity of exact inference via the Kalman filter and smoother is *linear* in the number of time steps. However, the KF scales *cubically* in both state dimension n and measurement dimension m due to matrix-matrix products and solving linear systems involved in the prediction and update steps, see Equations (2.20) to (2.26). Storing the quadratic number of pairwise covariances poses an additional challenge in terms of memory requirements.

Low-rank filtering is an approximate variant of filtering that assumes that the covariance matrices involved in the Kalman filter can be approximated with a rank-r representation

$$\Lambda \approxeq \Lambda^{1/2}\Lambda^{\mathsf{T}/2}, \tag{2.45}$$

with $\Lambda^{1/2} \in \mathbb{R}^{n\times r}$ and $r \leq n$. If the approximation Equation (2.45) is good for small r, then computational complexity can be reduced while preserving accuracy. Formulating the filtering recursions entirely in terms of the low-rank factorized covariances—similar to square-root filtering—reduces both floating-point operations and memory requirements. To achieve this, two questions arise: first, how to construct the (initial) low-rank representation $\Lambda^{1/2} \in \mathbb{R}^{n\times r}$ of the involved covariances; and second, how to leverage the low-rank assumption to reduce computational complexity. For the latter, the low-rank factors need to be propagated through the filtering recursions. Crucially, the prediction and update steps must not require constructing the full covariance matrices. In the following, we exemplify the idea of filtering with low-rank covariances by outlining a naive sampling approach.

Monte–Carlo approach

Both the construction and the propagation of low-rank filtering covariances can be approached through Monte–Carlo sampling. To this end, consider the state to be represented by an $n \times r$ matrix such that the columns represent randomly sampled realizations of the state distribution. Concretely, let

$$\mathrm{X}_l = \begin{bmatrix} X_l^{(1)} & \cdots & X_l^{(r)} \end{bmatrix} \in \mathbb{R}^{n\times r}, \quad X_l^{(i)} \sim \mathcal{N}(\mu_l, \Sigma_l) \in \mathbb{R}^n, \quad i = 1, \dots, r, \tag{2.46}$$

implicitly (via samples) represent the filtering distribution at time t_l. An approximation to the first moment is obtained through the sample mean

$$\mu_l \approx \overline{\mathrm{X}}_l \coloneqq \frac{1}{r}\sum_{i=1}^{r} X_l^{(i)}. \tag{2.47}$$

The matrix of state realizations also implies a square-root factor $\Sigma_l^{1/2}$ of the covariance:

$$\begin{aligned} \Sigma_l \approx \overline{\Sigma}_l &\coloneqq \frac{1}{r-1}\left(\mathrm{X}_l - \overline{\mathrm{X}}_l\right)\left(\mathrm{X}_l - \overline{\mathrm{X}}_l\right)^{\mathsf{T}} && (2.48)\\ &= \underbrace{\left(\frac{1}{\sqrt{r-1}}\left(\mathrm{X}_l - \overline{\mathrm{X}}_l\right)\right)}_{\Sigma_l^{1/2} \in \mathbb{R}^{n\times r}}\left(\frac{1}{\sqrt{r-1}}\left(\mathrm{X}_l - \overline{\mathrm{X}}_l\right)\right)^{\mathsf{T}}. && (2.49) \end{aligned}$$

Figure 2.5 visualizes Monte–Carlo approximations to a Matérn covariance for different r. In Equations (2.48) and (2.49), we assume the “broadcasted” difference to be defined as

$$\mathrm{X}_l - \overline{\mathrm{X}}_l = \begin{bmatrix} X_l^{(1)} - \overline{\mathrm{X}}_l & \cdots & X_l^{(r)} - \overline{\mathrm{X}} \end{bmatrix}. \tag{2.50}$$

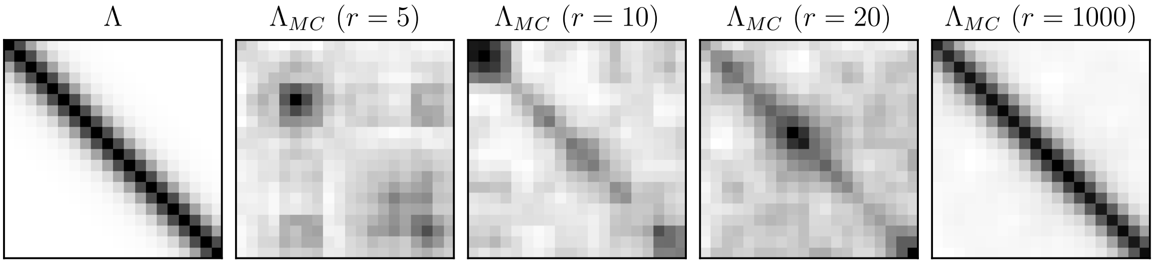


Figure 2.5: **Monte–Carlo low-rank approximations of a covariance matrix**. For a (full-rank) Matérn covariance $\Lambda \in \mathbb{R}^{n\times n}$ (cf. Figure 2.4, second column) and $n=20$, rank-r Monte–Carlo approximations $\Lambda \approx \Lambda_{MC} = \Lambda^{1/2}\Lambda^{\intercal/2}$ (cf. Equations (2.48) and (2.49)) are shown for $r = 5, 10, 20, 1000$.

The initial sample-state can, for instance, be drawn from an initial distribution

$$\mathrm{X}_0 \sim \mathcal{N}(\mu_0, \Sigma_0). \tag{2.51}$$

This leaves the question regarding the propagation of this sampled state through an approximate filtering routine. Consider the following approach:

1. **Predict:** Evaluating the transition model on each individual state realization (column) $X_l^{(i)}$ and adding sampled transition noise yields

$$\mathrm{X}_l^- = \Phi_{l|l-1}\mathrm{X}_{l-1} + \tilde{Q}_{l|l-1}, \tag{2.52}$$

$$\tilde{Q}_{l|l-1} := \begin{bmatrix} \zeta_{l|l-1}^{(1)} & \cdots & \zeta_{l|l-1}^{(r)} \end{bmatrix}, \qquad \zeta_{l|l-1}^{(i)} \sim \mathcal{N}\big(0, Q_{l|l-1}\big) \in \mathbb{R}^n. \tag{2.53}$$

   Note that, in order to sample process-noise realizations $\zeta_{l|l-1}^{(i)}$ *efficiently*, a low-rank factor $Q_{l|l-1}^{1/2} \in \mathbb{R}^{n\times r}$ of the process-noise covariance is required. As before, the predicted state implies a low-rank covariance matrix through

$$\Pi_l^{1/2} = \frac{1}{\sqrt{r-1}}\Big(\mathrm{X}_l^- - \overline{\mathrm{X}^-}_l\Big). \tag{2.54}$$

2. **Update:** The matrix of state realizations can be conditioned on noisy measurements Y [62, 58, 31]. Plugging in the low-rank factorization of the predicted covariance $\Pi^{1/2}$ yields

$$\mathrm{X}_l = \mathrm{X}_l^- + K_l\Big(\tilde{\mathrm{Y}}_l - H\mathrm{X}_l^-\Big), \tag{2.55}$$

   with

$$\tilde{\mathrm{Y}}_l = \begin{bmatrix} \mathrm{Y}_l + \xi_l^{(1)} & \cdots & \mathrm{Y}_l + \xi_l^{(r)} \end{bmatrix}, \qquad \xi_l^{(i)} \sim \mathcal{N}(0, R) \in \mathbb{R}^m, \tag{2.56}$$

$$\Upsilon_l = H\Pi_l^{1/2}, \tag{2.57}$$

$$K_l = \Pi_l^{1/2}\Upsilon^\intercal S_l^{-1}, \tag{2.58}$$

$$S_l = \Upsilon_l\Upsilon_l^\intercal + R. \tag{2.59}$$

This formulation still involves computing the inverse of the $m \times m$ matrix S_l and is thus still computationally intractable for high-dimensional measurements. However, using the *Sherman–Morrison–Woodbury formula* [74], the inverse of S_l can be written as [209]

$$\left(\Upsilon_l \Upsilon_l^\intercal + R\right)^{-1} = R^{-1} - R^{-1}\Upsilon_l \underbrace{\left(I_r + \Upsilon_l^\intercal R^{-1}\Upsilon_l\right)}_{\in\mathbb{R}^{r\times r}}^{-1} \Upsilon_l^\intercal R^{-1}, \tag{2.60}$$

which involves the inversion of a much smaller $r \times r$ matrix. Assuming that R^{-1} is available (for example assuming uncorrelated measurement error, which yields a diagonal R), computing Equation (2.60) scales cubically only in r.

Representing the filtering densities implicitly through a sampled state ensemble underlies the idea of a whole family of algorithms that we summarize under the term *ensemble Kalman filter* (EnKF) [59, 58, 60, 107, 173, 31]. Decades of research have produced various manifestations of the EnKF with different characteristics and properties. Notable variants include the *ensemble transform* Kalman filter (ETKF) [23] and the *ensemble adjustment* Kalman filter (EAKF) [8] that perform deterministic update steps. Tippett et al. [209] provide more details regarding ensemble-based square-root filters. Additionally, a comprehensive overview over the EnKF and related methods is provided by Carrassi et al. [31].

2.4 Dynamical systems

2.4.1 Temporal dynamics

For some time domain $[t_0, T] \subset \mathbb{R}$, let $u : [t_0, T] \to \mathbb{R}^n$ denote a time-varying state of a n-dimensional dynamical system. At time t, the components of $u(t)$ define a coordinate in the space of possible configurations of the system, often referred to as *phase space*. The temporal evolution of the system is governed by a system of *ordinary* differential equations (ODEs)

$$\dot{u}(t) = f(u(t), t) \tag{2.61}$$

where $\dot{u} = \frac{\mathrm{d}}{\mathrm{d}t}u$ denotes the temporal derivative. The *vector field* $f : \mathbb{R}^n \times [t_0, T] \to \mathbb{R}^n$ relates the *state* u to its time-derivative $\dot{u}$. In general, higher-order derivatives can enter the vector field. A first-order ODE can usually be obtained by tracking higher-order derivatives in a state vector [181] (cf. Equation (2.62)). Intuitively, f defines a direction for every location in phase space, in which the state at time t moves with an infinitesimal temporal increment. The goal is to determine the unknown function $u(t)$ that solves the ODE.

> **Example 2.2** (Pendulum) A simple pendulum, a weight hanging from a rod that is attached to a fixed anchor, often serves as an instructive example. At rest, the pendulum hangs perpendicularly to the floor. By swinging along a two-dimensional plane that intersects the rod and the anchor, the pendulum exhibits dynamics that are described by the temporal evolution of its angle ω (with respect to its resting position) and its current

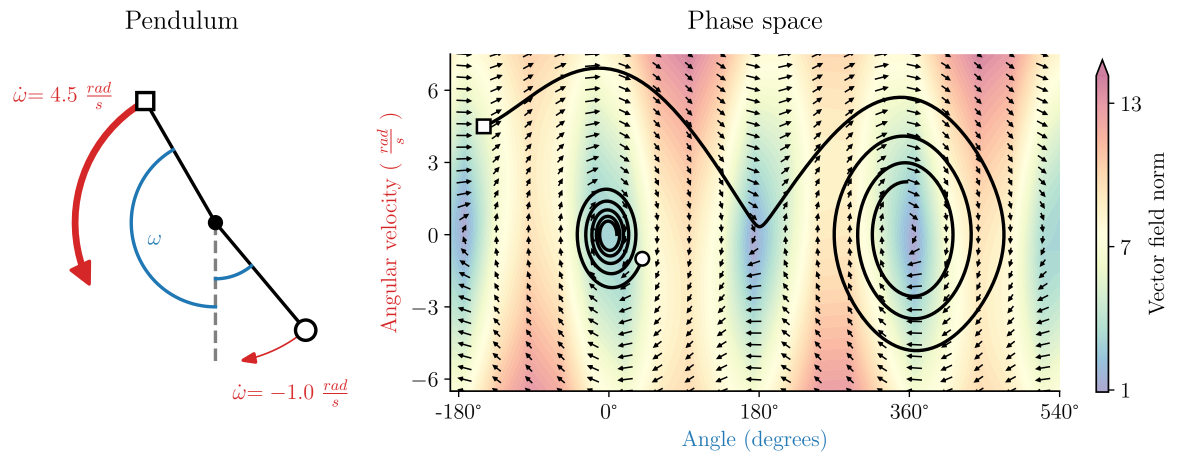


Figure 2.6: **Dynamics of a pendulum.** A pendulum system is fully described by its angle ω and angular velocity $\dot{\omega}$, which span a two-dimensional phase space. The dynamics depend on system parameters (rod length, (air-)resistance, gravitational constant; cf. Equation (2.62)) and initial conditions. Two trajectories illustrate different behaviors: the circular marker shows small oscillations around the stable equilibrium (0°), while the square marker demonstrates large-amplitude motion that overshoots 180° before settling into oscillations. The angle dimension repeats every 360° , making equilibria at 0° and 360° equivalent.

> angular velocity $\dot{\omega}$. The following IVP describes the dynamics of a specific pendulum model:
>
> $$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\omega(t)\\ \dot{\omega}(t)\end{bmatrix} = \begin{bmatrix}\dot{\omega}(t)\\ -\frac{g}{L}\sin(\omega(t)) - \gamma\dot{\omega}(t)\end{bmatrix}, \qquad \begin{bmatrix}\omega(0)\\ \dot{\omega}(0)\end{bmatrix} = \begin{bmatrix}\omega_0\\ \dot{\omega}_0\end{bmatrix}, \tag{2.62}$$
>
> with rod length L, gravitational constant g, (air-)resistance or dampening parameter γ, and initial configuration $[\omega_0 \quad \dot{\omega}_0]^\mathsf{T}$. Figure 2.6 provides a schematic of the pendulum together with two trajectories in phase space for two different initial conditions.

From Equation (2.61) it already becomes apparent that finding the solution $u(t)$ to a differential equation is not straightforward, as the unknown function itself appears as an argument to the vector field that defines the ODE. Approximate numerical integration, or *simulation*, finds the solution $u(t)$ to an *initial value problem* (IVP) (cf. Example 2.2)

$$\dot{u}(t) = f(u(t), t), \qquad u(t_0) = u_0 \in \mathbb{R}^n, \tag{2.63}$$

by starting from the initial condition u_0 and taking small steps that follow the directions of the vector field. The simplest such numerical integration scheme for ODEs is the *explicit Euler* solver, which, starting from $u(t_0) = u_0$, recursively computes the state on a sequence of time steps $(t_l)_{l=1}^N$ as

$$u(t_l) = u(t_{l-1}) + \Delta t_l \cdot f(u(t_{l-1}), t_{l-1}), \tag{2.64}$$

with *step size* $\Delta t_l = t_l - t_{l-1}$. Starting from the initial condition, this algorithm successively extrapolates linearly according to the direction given by the vector field f.

The discretization of the temporal domain poses a tradeoff between computational cost and accuracy. Large step sizes reduce the number of computations, as there are fewer points in the time domain on which the vector field has to be evaluated. Coarse time discretization, in turn, accumulates linearization error, which leads to inaccurate solution approximations or numerical instabilities. For details on numerical solvers for ODEs, the reader is referred to Hairer et al. [81, 82]. Probabilistic numerical ODE solvers that quantify the discretization error of the simulator compute a Gaussian posterior over the solution using approximate Kalman filtering [215, 120, 121, 119].

2.4.2 Spatiotemporal dynamics

Spatiotemporal dynamics are described by systems of *partial differential equations* (PDEs) [202, 57] of the form

$$\frac{\partial}{\partial t} u(t, x) = f(u(t, x), t, x, \mathcal{D}u(t, x)), \tag{2.65}$$

for time $t \in [t_0, T] \subset \mathbb{R}$ and space $x \in \Omega \subset \mathbb{R}^d$ and for some differential operator $\mathcal{D}$. The vector field that governs the PDE relates the time-derivative of the spatiotemporal system to its state and the spatial derivatives according to $\mathcal{D}$. The object of interest is the unknown function $u : [t_0, T] \times \Omega \to \mathbb{R}^n$ that solves a *boundary-value problem* (BVP) defined by the system in Equation (2.65) and initial conditions

$$u(t_0, x) = u_0(x), \quad x \in \Omega, \tag{2.66}$$

defined by a known function u_0 of x, and boundary conditions

$$\mathcal{B}u(t, \overline{x}) = u_b(\overline{x}), \quad \overline{x} \in \partial\Omega, \tag{2.67}$$

where u_b is a known function on the boundary $\partial\Omega$ of the spatial domain that defines the boundary conditions. $\mathcal{B}$ is a differential operator.

Spatial discretization

Discretizing the spatial domain x on a grid $\mathbb{X} := (x_1, \dots x_s)$ leaves time t as the only remaining independent variable. From the PDE in Equation (2.65) one obtains a large system of ODEs

$$\dot{u}(t) = f(u(t), t, Du(t)), \tag{2.68}$$

using the *method of lines* [184]. The matrix $D \in \mathbb{R}^{s \times s}$ approximates the differential operator $\mathcal{D}$ (Equation (2.65)) on $\mathbb{X}$ using, for example, finite differences [68]. The boundary conditions are discretized analogously. In this approach, each spatial grid node is tracked in the state vector, which can lead to intractably large systems. In particular, for fine spatial discretizations, s can be a large number. Moreover, multiple spatial dimensions let the number of grid points grow exponentially (see Figure 2.7). This poses an additional tradeoff between computational cost and accuracy, aside from the temporal discretization of the numerical integrator. For the

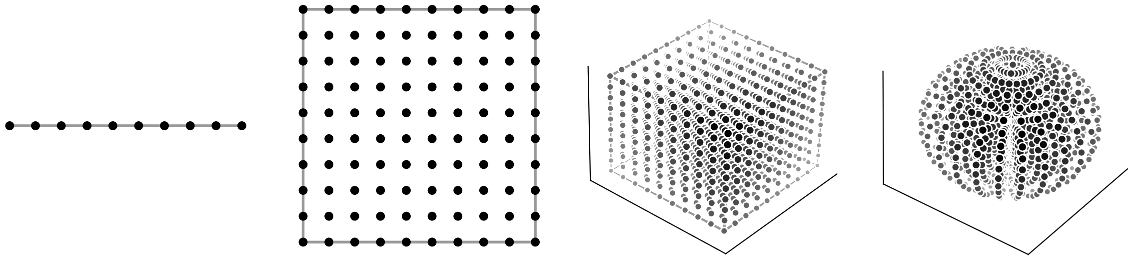


Figure 2.7: **Discretizing spatial domains in one, two, and three dimensions (left to right).** Let an equidistant grid discretize a spatial domain $\Omega \subset \mathbb{R}$ in one dimension on 10 nodes. For the same step size, the corresponding discretization of a two-dimensional domain $\Omega \subset \mathbb{R}^2$ requires $10^2 = 100$ nodes, and $10^3 = 1\,000$ nodes in three dimensions $\Omega \subset \mathbb{R}^3$. The state size for spatially discretized systems poses a computational challenge—in particular also for nonlinear spaces, like spheres and balls (*right*), which are relevant in atmospheric models.

simulation of spatiotemporal systems that exhibit fine-scale dynamics, simulation on coarse spatial grids can significantly limit the expressivity of the model.

Global climate (or *general circulation*) *models* (GCMs) [145, 13, 158] provide access to the analysis of future impact of climate change through the simulation of different hypothetical scenarios. Due to the complexity of the underlying spatiotemporal dynamics and the long time horizons, the computational resources required for climate projections are immense. Hence, climate models are usually simulated on coarse spatial grids with resolutions on the order of 100 km. Local assessment of climate conditions is thus impossible using only coarse GCM output. This motivates *downscaling* [140], which bridges the scale gap between coarse GCM projections and regional climate. *Dynamical* downscaling uses GCM boundary conditions to estimate local processes by simulating a physical model on a finer grid. *Statistical* downscaling offers a cost-efficient alternative. Estimating physically plausible fine-scale dynamics without an explicit physical model amounts to a complex and ill-posed inference problem. This motivates a probabilistic treatment and uncertainty quantification for statistical downscaling. In the following section, we introduce a class of generative models that will be used as dynamical prior models for probabilistic downscaling in Chapter 4.

2.5 Diffusion models

A *generative model* approximates the generative process that draws new samples from an unknown target distribution. Recently, a variety of methods have been developed that learn generative models for complex distributions from a finite set of available samples. Concretely, let a data set $\left\{\mathrm{X}^i\right\}_{i=1}^N$ be available and assume that $\mathrm{X}^i \sim p_0 \in \mathbb{R}^d$ are independent draws[2] from the unknown target distribution p_0. Prominent examples of generative models learn a parametric model $p^{(\theta)} \approx p_0$ from data, which *implicitly* represents the target distribution. The parameter vector θ can, for instance, consist of the weights of a deep neural network [75]. The

---

[2] The samples X^i can also be sequences $\mathrm{X}^i_{1:L}$. For notational simplicity, we omit the time index in the introduction. A specific class of sequential generative models will be introduced in Chapter 4.

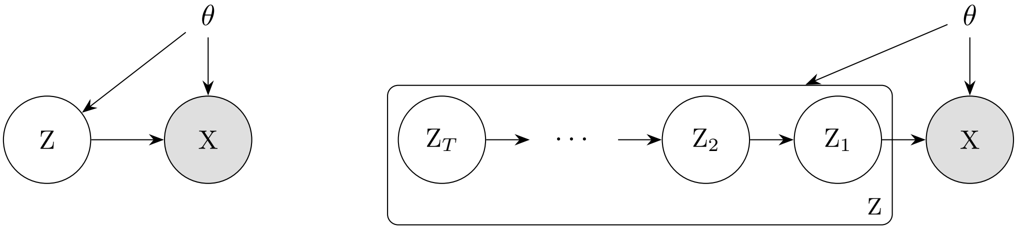


Figure 2.8: **Generative (latent-variable) models.** Left: a general latent-variable model that underlies some classes of generative models. The parameters θ of the generative process $p^{(\theta)}(\mathrm{X} \mid \mathrm{Z})$ are learned from samples from the unknown $p_0(\mathrm{X})$. *Right*: A feature of diffusion models is that the conditional $p(\mathrm{Z} \mid \mathrm{X}) = p(\mathrm{Z}_{1:T} \mid \mathrm{X})$ is known and modeled as a nonparametric Gauss–Markov process. At time $\tau = 0$, samples $\mathrm{Z}_0 =: \mathrm{X} \sim p^{(\theta)}(\mathrm{X} \mid \mathrm{Z}_{1:T}) \approx p_0(\mathrm{X})$ are drawn from the approximated data distribution.

term “implicit” means that, instead of approximating the density explicitly, the generative model learns to access the target distribution through sampling from a parametric distribution $p_0^{(\theta)}(\mathrm{X} \mid \mathrm{Z})$. Z is a latent variable that is assumed to underlie the generative process. Figure 2.8 shows a graphical model. This family of models includes generative adversarial networks (GANs) [76], variational auto-encoders (VAEs) [109], and *diffusion models* (DMs) [194, 198, 94, 199]. Over the past decade, these generative models have been extensively studied and developed. Compared to GANs and VAEs, DMs currently tend to be favored, likely due to their combining a simple and robust training objective with high-quality samples. Figure 2.9 shows the number of mentions of the terms “diffusion model”, “variational autoencoder”, and “generative adversarial network”, respectively, in titles or abstracts of papers uploaded to arXiv (`arxiv.org`) in the time period 2015–2024. After 2022, the mentions of DMs have increased significantly, while (according to this simple metric) research on GANs and VAEs has stagnated.

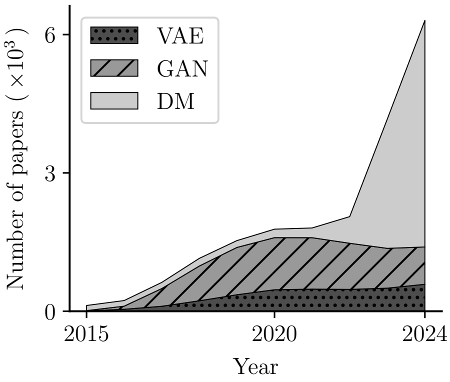


Figure 2.9: **Generative ML research.** This plot shows counts of mentions of three generative model classes—VAEs, GANs, and DMs—in titles or abstracts of arXiv submissions.

DMs revolve around two central components: a *diffusion process* and a *score function*. The eponymous diffusion process is a tractable noising process that gradually perturbs samples from the target distribution by injecting noise into the input signal. For DMs, the score function refers specifically to the gradient of the log-marginal distribution of the intermediate states of this perturbation process. Crucially, the score parametrizes a second, associated diffusion process that *reverts* the forward diffusion process. Intuitively, while the forward diffusion destroys all signal in data, the reversed *generative* process refines random noise into data.

Concretely, the diffusion process is, in the forward direction, defined as a GM process that solves

$$\mathrm{d}\mathrm{Z}(\tau) = A\mathrm{Z}(\tau)\mathrm{d}\tau + B\mathrm{d}\mathrm{W}(\tau), \tag{2.69}$$

where the drift $A \in \mathbb{R}$ and dispersion $B \in \mathbb{R}$, as well as the driving Brownian motion $\mathrm{W}(\tau) \in \mathbb{R}$ are assumed to be scalar. Notably, the framework generalizes to more general SDEs [199, 106, 227]—in particular, matrix-valued drift and dispersion can encode explicit correlation structure in the diffusion process. In practice, however, it is most common to model independent scalar, linear diffusion processes for each state component of $\mathrm{Z}(\tau) \in \mathbb{R}^d$, which conveniently makes the scalar case the standard case. Starting at $\tau = 0$, the diffusion process in Equation (2.69) transforms structured data, which is available as samples from the unknown target distribution p_0, into unstructured noise samples from a Gaussian distribution, which the diffusion process converges to as $\tau \to T$. The moments of this distribution depend on the choice of A, B, and T. Commonly, the noise distribution is an isotropic Gaussian distribution $p_T(\mathrm{Z}(T)) = \mathcal{N}\big(0,\, \sigma_T^2\big)$ for some variance σ_T^2 [199, 106].

For the Gauss–Markov process in Equation (2.69) that models the forward diffusion (data ⇝ noise) a corresponding *reverse* diffusion process (noise ⇝ data) [7, 199] is governed by

$$\mathrm{dZ}(\tau) = \left[A\mathrm{Z}(\tau) - B^2 \underbrace{\nabla_{\mathrm{Z}(\tau)} \log p_\tau(\mathrm{Z}(\tau))}_{\textit{score} \text{ of } p_\tau} \right] \mathrm{d}\tau + B \mathrm{d}\overleftarrow{\mathrm{W}}(\tau), \tag{2.70}$$

driven by reverse-time Brownian motion $\overleftarrow{\mathrm{W}}$. The dynamics of this generative process are parametrized by the *score function* $s(\mathrm{Z}(\tau)) \coloneqq \nabla_{\mathrm{Z}(\tau)} \log p_\tau(\mathrm{Z}(\tau))$ of the marginal p_τ, which is usually unknown. In practice, one commonly approximates the score function with a parametric model $s^{(\theta)}(\mathrm{Z}(\tau), \tau) \approx \nabla_{\mathrm{Z}(\tau)} \log p_\tau(\mathrm{Z}(\tau))$ such that the vector of parameters θ minimizes a *de-noising score matching* (DSM) objective [219, 198, 94, 199]

$$\theta = \arg\min_{\hat{\theta}} \mathbb{E}_{\mathrm{Z}(0) \sim p_0(\mathrm{Z}(0))} \mathbb{E}_{\tau \sim \mathrm{Unif}[0,T]} \mathbb{E}_{\mathrm{Z}(\tau) \sim p_\tau(\mathrm{Z}(\tau) | \mathrm{Z}(0))} [\ell_{\mathrm{DSM}}], \tag{2.71}$$

$$\text{with} \quad \ell_{\mathrm{DSM}} = \| s^{(\hat{\theta})}(\mathrm{Z}(\tau), \tau) - \nabla_{\mathrm{Z}(\tau)} \log p_\tau(\mathrm{Z}(\tau) \mid \mathrm{Z}(0)) \|_2^2. \tag{2.72}$$

A pivotal result by Vincent [219] shows the equivalence between the de-noising score matching objective in Equations (2.71) and (2.72) and the more general score-matching objective that depends directly on the inaccessible score of the marginal $\nabla_{\mathrm{Z}(\tau)} \log p_\tau(\mathrm{Z}(\tau))$. By choosing a linear Gaussian perturbation kernel $p(\mathrm{Z}(\tau) \mid \mathrm{Z}(0))$, the score-matching objective becomes tractable and efficient to evaluate. Recall that for a Gaussian Markov model Equation (2.69) an equivalent discretized formulation exists. Concretely, we have

$$p_\tau(\mathrm{Z}(\tau) \mid \mathrm{Z}(0)) = \mathcal{N}\big(\alpha(\tau)\mathrm{Z}(0),\, \sigma^2(\tau)\big). \tag{2.73}$$

The scalar transition function $\alpha : [0, T] \to \mathbb{R}$ and process-noise function $\sigma : [0, T] \to \mathbb{R}$ are derived from A and B. In practice, it is sufficient to directly select a "*noise schedule*" α and σ, instead of discretizing an SDE model [106]. In particular, it is common to set $\alpha(\tau) = \sqrt{1 - \sigma(\tau)^2}$, which is referred to as *variance-preserving* noise process. Forward

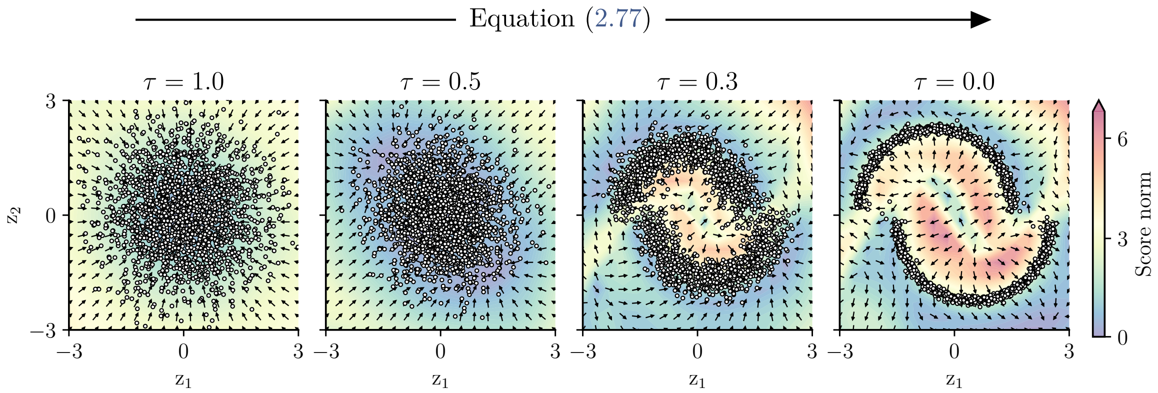


Figure 2.10: **Generative process of a diffusion model.** From left to right, this plot visualizes the vector field of the learned score model (arrows) and the samples $\mathrm{Z}(\tau) = [z_1(\tau), z_2(\tau)]^\intercal$ at $\tau = 1, 0.5, 0.3, 0$. Here, $T = 1$. The background color indicates the norm of the vector field.

simulation of Equation (2.69) amounts to sampling from Equation (2.73) and can be computed in a single step as

$$\mathrm{Z}(\tau) \mid \mathrm{Z}(0) = \alpha(\tau)\mathrm{Z}(0) + \sigma(\tau)\epsilon, \qquad \epsilon \sim \mathcal{N}(0, I_d) \in \mathbb{R}^d. \tag{2.74}$$

Furthermore, the closed-form score function

$$\nabla_{\mathrm{Z}(\tau)} \log p_\tau(\mathrm{Z}(\tau) \mid \mathrm{Z}(0)) = \nabla_{\mathrm{Z}(\tau)} \log \mathcal{N}\big(\alpha(\tau)\mathrm{Z}(0),\, \sigma^2(\tau)\big) \tag{2.75}$$

$$= -\frac{\epsilon}{\sigma(\tau)} \tag{2.76}$$

is directly accessible. It is common to learn a model for the noise-component $\epsilon^{(\theta)}(\mathrm{Z}(\tau), \tau) \approx \epsilon$, instead of learning the score-function directly [94, 157, 196]. However, different parametrizations of the score function exist that correspond to maximizing the same evidence lower bound under a variational-inference perspective [108]. In particular, DMs are recovered as a special (stochastic) case of the more general *flow-matching* (FM) framework [130, 131]. For simplicity, we will continue to base the introduction on a model $s^{(\theta)}$ for the score, which is linearly (affinely) related to other relevant parametrizations.

Sampling from the target distribution amounts to first drawing a noise vector $\mathrm{Z}(T) \sim p_T(\mathrm{Z}(T)) = \mathcal{N}\big(0,\, \sigma_T^2\big)$ and then simulating the reverse-time diffusion process Equation (2.70), plugging in the approximate score model $s^{(\theta)}$

$$\mathrm{dZ}(\tau) = \Big[A\mathrm{Z}(\tau) - B^2 s^{(\theta)}(\mathrm{Z}(\tau), \tau)\Big]\mathrm{d}\tau + B\mathrm{d}\overleftarrow{\mathrm{W}}(\tau), \tag{2.77}$$

from $T \to 0$—for example, using an Euler–Maruyama scheme [181, 199]. It is equivalent (in distribution) to solve the probability flow ODE associated with the generative diffusion process [199]. Figure 2.10 visualizes a sampling process (using the probability flow) in a two-dimensional setting, where $\mathrm{Z} = [z_1 \;\; z_2]^\intercal$. More efficient and robust sampling routines for DMs is subject of active research. Relevant methods include, but are not limited to,

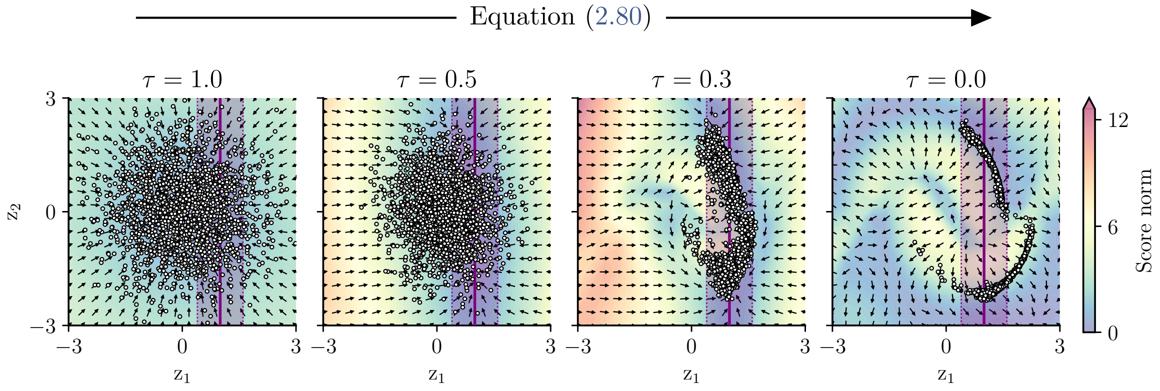


Figure 2.11: **Generative process of a posterior diffusion model.** From left to right, this plot visualizes the vector field of the *posterior* score model (arrows) and the samples $Z(\tau) = [z_1(\tau) \;\; z_2(\tau)]^\mathsf{T}$ at $\tau = 1, 0.5, 0.3, 0$. Here, $T = 1$. The background color indicates the norm of the vector field. The Gaussian observation model $\mathcal{N}\big(Y = 1; z_1(\tau), 0.3^2\big)$ is shown in purple; the shaded area corresponds to two standard deviations.

distillation methods [178], exponential integrators [227], and fine-tuning the score model to enable single-step generation [197]. A survey is provided by Shen et al. [191].

2.5.1 Posterior score for a tractable likelihood model

The score of the conditional distribution $p_\tau(Z(\tau) \mid Y) \propto p_\tau(Z(\tau))p(Y \mid Z(\tau))$ is given by

$$\nabla_{Z(\tau)} \log p_\tau(Z(\tau) \mid Y) = \nabla_{Z(\tau)}[\log p_\tau(Z(\tau)) + \log p(Y \mid Z(\tau))] \tag{2.78}$$

$$\approx \underbrace{\nabla_{Z(\tau)} \log p_\tau(Z(\tau))}_{\approx s^{(\theta)}(Z(\tau),\tau)} + \nabla_{Z(\tau)} \log p(Y \mid Z(\tau)). \tag{2.79}$$

Approximately sampling from a posterior $p(Z(0) \mid Y)$, for a likelihood model $p(Y \mid Z(\tau))$, thus amounts to simulating

$$dZ(\tau) = \Big[AZ(\tau) - B^2\Big(s^{(\theta)}(Z(\tau),\tau) + \nabla_{Z(\tau)} \log p(Y \mid Z(\tau))\Big)\Big]d\tau + B d\overleftarrow{W}(\tau). \tag{2.80}$$

If it is cheap to evaluate $\nabla_{Z(\tau)} \log p(Y \mid Z(\tau))$, conditional samples come at a negligible added (computational) complexity. Figure 2.11 visualizes the generative process for a diffusion model conditioned on the Gaussian likelihood

$$p\Big(Y \mid [z_1(\tau) \;\; z_2(\tau)]^\mathsf{T}\Big) = \mathcal{N}\big(Y = 1; z_1(\tau), 0.3^2\big). \tag{2.81}$$

This concept is related to *classifier guidance* [44] for conditional generation of images from class labels. For Gaussian and Poisson likelihoods, Chung et al. [39] develop *diffusion posterior sampling* (DPS) for approximate conditional sampling with a likelihood model $p(Y \mid Z(0))$. More details follow in Chapter 4.

PART II

Probabilistic Inference for Spatiotemporal Dynamics

"To err is human, but to really foul things up
you need a computer."
- Paul R. Ehrlich

3 Approximate Dynamical-Low-Rank Filtering In High Dimensions

Inference and simulation in the context of high-dimensional dynamical systems remain computationally challenging problems. Some form of dimensionality reduction is required to make the problem tractable in general. In this paper, we propose a novel approximate Gaussian filtering and smoothing method which propagates low-rank approximations of the covariance matrices. This is accomplished by projecting the Lyapunov equations associated with the prediction step to a manifold of low-rank matrices, which are then solved by a recently developed, numerically stable, dynamical low-rank integrator. Meanwhile, the update steps are tractable by noting that the covariance update only transforms the column space of the covariance matrix, which is low-rank by construction. The algorithm differentiates itself from existing ensemble-based approaches in that the low-rank approximations of the covariance matrices are deterministic, rather than stochastic. Crucially, this enables the method to reproduce the exact Kalman filter as the low-rank dimension approaches the true dimensionality of the problem. Our method reduces computational complexity from cubic (for the Kalman filter) to *quadratic* in the state-space size in the worst-case, and can achieve *linear* complexity if the state-space model satisfies certain criteria. Through a set of experiments in classical data-assimilation and spatiotemporal regression, we show that the proposed method consistently outperforms the ensemble-based methods in terms of error in the mean and covariance with respect to the exact Kalman filter. This comes at no additional cost in terms of asymptotic computational complexity.

3.1 Introduction

Spatiotemporal dynamical systems have always played an important role in the applied sciences, such as climate science, numerical weather prediction, or geophysics. In precisely these settings, one is often faced with massive amounts of data to be processed. At the same time the interactions are of such complex nature that it is imperative to include a notion of uncertainty in the model and in its outputs. Both these requirements have a reputation as being computationally expensive. Hence, sensible approximations are indispensable at a certain scale.

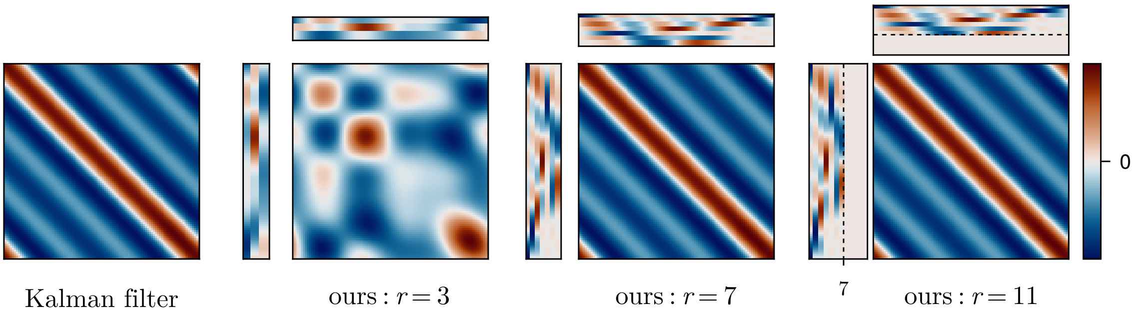


Figure 3.1: **Rank-r approximations to the true KF covariance (left) for increasing r.** The considered problem's true rank is $r^* = 7$ by construction, the state-dimension is $n = 1000$. The respective low-rank factors are shown above and left of their outer product. For $r \geq r^*$, the KF estimate is recovered. For $r > r^*$, the excess columns of the low-rank factor collapse (rightmost plot).

Concretely, we consider state-space models of the form

$$\mathrm{dX}(t) = A\mathrm{X}(t)\mathrm{d}t + B\mathrm{dW}(t), \tag{3.1a}$$

$$\mathrm{Y}(t_l) \mid \mathrm{X}(t_l) \sim \mathcal{N}(H\mathrm{X}(t_l), R), \quad l = 1, \ldots, N \tag{3.1b}$$

where $\mathrm{X} \in \mathbb{R}^n$ is the latent state, $\mathrm{Y} \in \mathbb{R}^m$ is the measurement process, and W is a standard Wiener process in $\mathbb{R}^q$. Equation (3.1a) defines the prior dynamics model via a linear time-invariant (LTI) stochastic differential equation (SDE) with drift matrix $A \in \mathbb{R}^{n \times n}$ and dispersion matrix $B \in \mathbb{R}^{n \times q}$. The linear Gaussian observation model Equation (3.1b) is specified via the measurement matrix $H \in \mathbb{R}^{m \times n}$ and measurement-noise covariance $R \in \mathbb{R}^{m \times m}$, where possible time-dependency has been omitted from the notation. While the probabilistic state-estimation problem may in principle be solved in closed form using the Kalman filter (KF) and smoother [105, 179], we will consider the case where the state dimension n and measurement dimension m are very large, making the covariance recursion computationally prohibitive in practice. Such settings include spatiotemporal Gaussian process regression [180, 182, 211, 33, 84], and data assimilation applications such as meteorology and oceanography [72, 59], numerical weather forecasting [85], geoscience [31], inverse problems [37, 204], and brain imaging [70]. While the Gaussian process (GP) literature offers many ways of tractable, approximate inference, approaches based on approximations of the underlying kernel function are not amenable to the present problem setting. Without a kernel to approximate, the problem is often solved by ensemble methods, which stochastically propagate low-rank approximations to the covariances in the filtering recursion. This stochasticity introduces unfavorable properties in these methods. To avert this shortcoming, we propose a novel low-rank filtering recursion, which is fully deterministic.

CONTRIBUTIONS

This text develops a method for efficient, approximate Gaussian filtering, including smoothing and marginal likelihood computation. The method employs the standard predict/correct formulation of the filtering problem, and can in principle be thought of as a square-root implementation of a Kalman filter, in that it represents all covariance matrices as their left square-root factors [77]. However, instead of square matrix square roots, the covariances are approximated by $(n \times r)$-dimensional low-rank factors, where $r \ll n$. We propose an approximate low-rank prediction-correction loop, together with the computation of the backwards Markov process representing the smoothing posterior, all while preserving this low-rank structure throughout time.

Crucial to this work are (i) the use of dynamic-low-rank approximation (DLRA) [115, 136, 36] to solve Lyapunov equations on the manifold of rank-r matrices in the prediction step and (ii) an efficient correction scheme that updates only the low-dimensional column space of the predicted covariance factor, which introduces no additional approximation error. The computational complexity of the proposed filter is asymptotically equal to those of the widely-used ensemble Kalman filters.

The method offers a deterministic alternative to existing ensemble methods. In particular, it recovers the exact Kalman filter estimate both in the full-rank limit $r = n$ and if the problem is in fact of rank $r < n$. The latter is demonstrated in Figure 3.1, in which a problem with $n = 1000$ state dimensions and a true rank of 7 is approximated with $r = 3, 7, 11$. As soon as r exceeds the true rank of the problem, the residual subspace dimensions collapse. The method will be detailed in Section 3.3 and evaluated on the basis of various experiments in Section 3.4. All source code is publicly available on GitHub.[1,2]

RELATED WORK

The state estimation problem defined by Equation (3.1) amounts to linear Gaussian state estimation and can be computed in closed form by Kalman filtering and smoothing. A modern overview can be found in Särkkä et al. [181]. While the exact filtering and smoothing recursions scale linearly with the number of time points N, they scale cubically in the state dimension n and the measurement dimension m, making their use computationally infeasible in many high-dimensional problems. A widely-used approximate filtering technique used to circumvent this computational burden are ensemble Kalman filters (EnKF) [59, 58, 60, 107, 173, 31]. The ensemble is a set of r randomly sampled state vectors of dimension n, which parametrize the mean and covariance via sample statistics. It is processed successively in a prediction-correction loop similar to the KF. The mean-centered ensemble can be regarded as a low-rank square-root factor of the sample covariance matrix. In Section 3.4 we compare our method to the fully stochastic version of the EnKF and to a variant that uses the deterministic update scheme of the ensemble transform Kalman filter (ETKF) [23]. The ETKF, in particular, is categorized as an ensemble-square-root filter [209].

In so far as it aims for computationally tractable Gaussian process regression, the proposed method is related to approaches using the Nyström method [224], inducing point methods [193, 210, 88], random Fourier features [165], and doubly-sparse variational Gaussian processes [2].

---

[1] `https://github.com/schmidtjonathan/RRKF.jl`

[2] `https://github.com/schmidtjonathan/RRKF_experiments/`

The literature offers more contributions related to the present work under the broader context of low-rank and dimensionality-reduction methods. Loiseau et al. [134], Netto et al. [155], and Vijayshankar et al. [218] approach low-rank modelling of dynamical systems based on measured data, whereas Farrell et al. [64] are concerned with low-rank approximations of a given model based on classical theory on linear time-invariant systems. Cressie et al. [42] develop a tractable hierarchical Bayesian model to approximate high-dimensional spatiotemporal dynamics in a reduced latent space and condition on measurements by transporting the low-dimensional diffusive process to the full space by a projection that is assumed to be known. In those cases, once a low rank model is obtained for the phenomena under study, state estimation becomes computationally tractable, due to the reduced size of the resulting model. These works build upon the assumption that the state of the system evolves approximately in a low-dimensional manifold, whereas our method works under the assumption that the *error* in the state estimate evolves approximately in a low-dimensional manifold.

3.2 Background

> **Remark 3.1** To de-clutter notation, in this chapter, we will use short-hand notation $\Phi_l = \Phi_{l|l-1}$ and $Q_l = Q_{l|l-1}$ (cf. Equation (2.15)) for the transition and process-noise covariance matrices.

The task of interest is computing the filtering densities $p(\mathrm{X}_l \mid \mathrm{Y}_{1:l})$ and smoothing densities $p(\mathrm{X}_l \mid \mathrm{Y}_{1:N})$ for $l = 0, \dots, N$. LTI SDEs of the form in Equation (3.1a) admit an equivalent discrete formulation in terms of linear Gaussian transition densities of the form

$$\mathrm{X}_l \mid \mathrm{X}_{l-1} \sim \mathcal{N}(\Phi_l \mathrm{X}_{l-1}, Q_l). \tag{3.2}$$

The transition matrix $\Phi(t) \in \mathbb{R}^{n \times n}$ and process-noise covariance matrix $Q(t) \in \mathbb{R}^{n \times n}$ solve the matrix differential equations $\dot{\Phi}(t) = A\Phi(t)$ and

$$\dot{Q}(t) = AQ(t) + Q(t)A^\intercal + BB^\intercal, \tag{3.3}$$

in the interval $[t_{l-1}, t_l)$ with initial conditions $\Phi(t_{l-1}) = I$ and $Q(t_{l-1}) = 0$, respectively [181]. Equation (3.3) is known as a *Lyapunov equation*, and will be of special importance in this work.

3.2.1 Square-root filtering

For numerically stable filtering and smoothing, all covariance matrices involved can be represented by matrix square roots (e.g. Cholesky factors). This is known as square-root filtering/smoothing [77]. Let $(M_1 \quad M_2)$ denote a wide block matrix in $\mathbb{R}^{d_1 \times (d_2 + d_3)}$, where $M_1 \in \mathbb{R}^{d_1 \times d_2}$ and $M_2 \in \mathbb{R}^{d_1 \times d_3}$ are some matrices. Given the filtering covariance at time t_{l-1}

as $\Sigma_{l-1} = \Sigma_{l-1}^{1/2}\Sigma_{l-1}^{\mathsf{T}/2}$ a square-root factorization of the predicted covariance $\Pi_l = \Pi_l^{1/2}\Pi_l^{\mathsf{T}/2}$ is given by

$$\Pi_l = \Phi_l \Sigma_{l-1} \Phi_l^{\mathsf{T}} + Q_l = \begin{pmatrix} \Phi_l \Sigma_{l-1}^{1/2} & Q_l^{1/2} \end{pmatrix} \begin{pmatrix} \Phi_l \Sigma_{l-1}^{1/2} & Q_l^{1/2} \end{pmatrix}^{\mathsf{T}}, \tag{3.4}$$

and, hence, $(\Phi_l \Sigma_{l-1}^{1/2} \quad Q_l^{1/2})$ is a matrix square-root of Π_l. Similarly, the correction step can be carried out entirely on square-root factors. For more details, see Grewal et al. [77].

3.2.2 Dynamical low-rank approximation

Consider an initial value problem with a matrix-valued flow field $F : \mathbb{R} \times \mathbb{R}^{u \times v} \to \mathbb{R}^{u \times v}$

$$\dot{M}(t) = F(t, M(t)), \qquad M(t_0) = M_0, \tag{3.5}$$

where u and v are potentially very large. Dynamic low-rank approximation (DLRA) methods [115, 36] efficiently compute low-rank factorizations $\Upsilon(t) = U(t)D(t)V^{\mathsf{T}}(t) \approx M(t)$ of Equation (3.5) by solving

$$\dot{\Upsilon}(t) = \mathcal{P}_r[\Upsilon(t)] \circ F(t, \Upsilon(t)), \qquad \Upsilon(t_0) = U_0 D_0 V_0^{\mathsf{T}}, \tag{3.6}$$

instead. The matrices $U_0 \in \mathbb{R}^{u \times r}$, $D_0 \in \mathbb{R}^{r \times r}$, and $V_0 \in \mathbb{R}^{v \times r}$ are an initial low-rank factorization of $\Upsilon(t_0) \approx M(t_0)$. $\mathcal{P}_r[\Upsilon(t)]$ denotes the projection operator onto the tangent space at $\Upsilon(t)$, where $\Upsilon(t)$ lies in the manifold of rank-r matrices. In the present work, we leverage this technique to obtain a low-rank factor $Q^{1/2}$ of the process-noise covariance at the next prediction location. Thus, we avoid much of the computational cost in the prediction step, otherwise caused by solving the full Lyapunov equation for Q in Equation (3.3), for example using matrix-fraction decomposition [201, 10]. Concretely, let $F(Q) = AQ + QA^{\mathsf{T}} + BB^{\mathsf{T}}$. Then, an approximate low-rank process-noise covariance matrix associated with the prediction step is obtained by integrating

$$\dot{Q}(t) = \mathcal{P}_r[Q(t)] \circ F(Q(t)), \qquad Q(t_{l-1}) = U_0 D_0^2 U_0^{\mathsf{T}} = 0 \tag{3.7}$$

from t_{l-1} to t_l. At the start of the filtering recursion t_0, an initial low-rank factorization is constructed with a random orthogonal matrix U_0 and $D_0 = 0$. For all subsequent steps the propagated orthogonal basis $U(t_l)$ can be reused. In this work, all mentions of dynamic low-rank integration always refer to the recently developed, numerically stable basis update & Galerkin (BUG) integrator [36], whose error bounds are independent of small singular values. More details are given in Section A.3.

3.3 Rank-reduced Kalman filtering

In this section, a method for approximate inference in Equation (3.1) is developed. The idea is to approximate the full filtering/smoothing covariance matrices of the latent state by an eigendecomposition truncated at the r-th largest eigenvalue. The main challenge is to thereby attain linear computational scaling in n and m under appropriate assumptions on the state-space model. Before proceeding with the filtering and smoothing recursions, it is instructive to examine inference in a static low-rank model.

3.3.1 Efficient inference in static models of low rank

Consider the following latent variable model

$$\mathrm{X} \sim \mathcal{N}\left(\mu, \Pi^{1/2}\Pi^{\mathsf{T}/2}\right), \qquad \mathrm{Y} \mid \mathrm{X} \sim \mathcal{N}(H\mathrm{X}, R), \tag{3.8}$$

where $\Pi^{1/2} \in \mathbb{R}^{n \times r}$ is of full column rank, $r \leq n$, and $R \in \mathbb{R}^{m \times m}$. Representing the latent state X as

$$\mathrm{X} = \mu + \Pi^{1/2}\mathrm{Z}, \qquad \mathrm{Z} \sim \mathcal{N}(0, I_{r \times r}), \tag{3.9}$$

reduces the problem to inference in the following model

$$\mathrm{Z} \sim \mathcal{N}(0, I_{r \times r}), \qquad \mathrm{Y} \mid \mathrm{Z} \sim \mathcal{N}\left(H\mu + H\Pi^{1/2}\mathrm{Z}, R\right). \tag{3.10}$$

An efficient inference scheme is given by the following proposition, which is proved in Section A.1.1.

Proposition 3.1 *Let* Z *and* Y *be two random variables governed by Equation* (3.10). *Assume* $r \leq m$ *and consider the following singular value decomposition* $(R^{-1/2}H\Pi^{1/2})^{\mathsf{T}} = UDV^{\mathsf{T}}$, *where* $U, D \in \mathbb{R}^{r \times r}$ *and* $V \in \mathbb{R}^{m \times r}$. *Then, defining the whitened residual* $e = R^{-1/2}(\mathrm{Y} - H\mu)$, *we get*

$$\mathrm{Y} \sim \mathcal{N}\left(H\mu, R^{1/2}(VD^2V^{\mathsf{T}} + I)R^{\mathsf{T}/2}\right), \tag{3.11a}$$

$$\mathrm{Z} \mid \mathrm{Y} \sim \mathcal{N}\left(U(I + D^2)^{-1}DV^{\mathsf{T}}e, U(I + D^2)^{-1}U^{\mathsf{T}}\right). \tag{3.11b}$$

Furthermore, let $|\cdot|$ *denote the matrix determinant. The marginal log-likelihood of* Y *is given by*

$$\begin{aligned}\log \mathcal{N}\left(\mathrm{Y}; H\mu, R^{1/2}(VD^2V^{\mathsf{T}} + I)R^{\mathsf{T}/2}\right) = &-\frac{m}{2}\log 2\pi - \log\left|R^{1/2}\right| - \frac{1}{2}\sum_{k=1}^{r}\log\left(1 + D_{kk}^2\right) \\ &- \frac{1}{2}\|e\|^2 + \frac{1}{2}e^{\mathsf{T}}VD(D^2 + I)^{-1}DV^{\mathsf{T}}e.\end{aligned} \tag{3.12}$$

The below corollary follows from the deterministic relationship between Z and X given by Equation (3.9).

Corollary 3.2 *Let* Y *and* X *be two random variables governed by the model Equation* (3.8) *and* $r \leq m$. *Then*

$$\mathrm{X} \mid \mathrm{Y} \sim \mathcal{N}(\mu + Ke, \Sigma), \tag{3.13}$$

> *where*
> $$K = \Pi^{1/2} U (I + D^2)^{-1} D V^{\mathsf{T}}, \qquad \Sigma = \Pi^{1/2} U (I + D^2)^{-1} U^{\mathsf{T}} \Pi^{\mathsf{T}/2}. \tag{3.14}$$
> *Moreover, a square-root of* Σ *is readily obtained by*
> $$\Sigma^{1/2} = \Pi^{1/2} U (I + D^2)^{-1/2}. \tag{3.15}$$

3.3.2 The rank-reduced filtering recursion

In this section, a low-rank prediction-correction recursion is developed for the purpose of obtaining the filtering densities and the logarithm of the marginal likelihood.

The prediction equations

Suppose the filtering distribution at time t_{l-1} is given by

$$p(\mathrm{X}_{l-1} \mid \mathrm{Y}_{1:l-1}) = \mathcal{N}\Big(\mathrm{X}_{l-1}; \mu_{l-1}, \Sigma_{l-1}^{1/2} \Sigma_{l-1}^{\mathsf{T}/2}\Big), \tag{3.16}$$

where $\Sigma_{l-1}^{1/2} \in \mathbb{R}^{n \times r}$. We begin by detailing how to compute a low-rank factor $\Pi_l^{1/2}$ of the predicted covariance. First compute the truncated singular value decomposition (SVD)

$$\Big(\Phi_l \Sigma_{l-1}^{1/2} \quad Q_l^{1/2}\Big) \approx \tilde{U}_l \tilde{D}_l \tilde{V}_l^{\mathsf{T}} \tag{3.17}$$

of the square-root factor (cf. Section 3.2.1), with $\tilde{U}_l \in \mathbb{R}^{n \times r}$, $\tilde{D}_l \in \mathbb{R}^{r \times r}$, and $\tilde{V}_l \in \mathbb{R}^{r \times r}$. This can be done in an optimal way by computing the full SVD and truncating it at the r-th largest singular value in $\mathcal{O}(nr^2)$ [74]. The process-noise covariance factor $Q_l^{1/2} \in \mathbb{R}^{n \times r}$ in Equation (3.17) is computed using DLRA as described in Section 3.2.2. The rank-reduced predicted moments at time t_l are thus given as

$$\mu_l^- = \Phi_l \mu_{l-1} \quad \in \mathbb{R}^n, \tag{3.18}$$
$$\Pi_l^{1/2} = \tilde{U}_l \tilde{D}_l \quad \in \mathbb{R}^{n \times r}. \tag{3.19}$$

The update equations

The low-rank update equations follow from Proposition 3.1 and Corollary 3.2:

$$\mu_l = \mu_l^- + K_l e_l \quad \in \mathbb{R}^n, \tag{3.20}$$
$$\Sigma_l^{1/2} = \Pi_l^{1/2} U_l (I + D_l^2)^{-1/2} \quad \in \mathbb{R}^{n \times r}, \tag{3.21}$$

with the whitened residual $e_l = R^{-1/2}(\mathrm{Y}_l - H\mu_l^-)$ and

$$(R^{-1/2} H \Pi_l^{1/2})^{\mathsf{T}} = U_l D_l V_l^{\mathsf{T}}, \qquad K_l = \Pi_l^{1/2} U_l (I + D_l^2)^{-1} D_l V_l^{\mathsf{T}}, \tag{3.22}$$

for $U_l, D_l \in \mathbb{R}^{r \times r}$ and $V_l \in \mathbb{R}^{m \times r}$.

> **Remark 3.2** Equation (3.21) corresponds to the ensemble transformation of the ETKF [23, Eq. (18)] when $\Pi^{1/2}$ is a normalized ensemble of sampled states as in Equation (2.54).

The marginal predictive log-likelihood of Y_l is given by

$$\begin{aligned}\log p(\mathrm{Y}_l \mid \mathrm{Y}_{1:l-1}) = -\frac{m}{2}\log 2\pi - \log\left|R^{1/2}\right| - \frac{1}{2}\sum_{k=1}^{r}\log\big((D_l)_{kk}^2 + 1\big) \\ - \frac{\|e_l\|^2}{2} + \frac{1}{2}e_l^\intercal V_l D_l (D_l^2 + I)^{-1} D_l V_l^\intercal e_l.\end{aligned} \tag{3.23}$$

Corollary 3.2—and by extension, the above filtering recursion—is only valid when $r \leq m$, which is exactly the intended usecase for the proposed method. For settings in which $m < r \leq n$ the correction step follows the square-root correction of a Kalman filter and is detailed in Section A.2.

3.3.3 Time complexity of the rank-reduced filtering recursion

This section analyzes the computational complexity of the proposed method. In the worst case, the method scales quadratically in the state dimension n and measurement dimension m. Under favorable conditions, which are met in many real-world applications, the rank-reduced Kalman filter obtains a computational complexity of $O(nr^2)$ as stated by Proposition 3.3, which is proven in Section A.1.2. We begin by specifying a set of assumptions for Proposition 3.3.

> **Assumption 3.1** *The maps $x \mapsto Ax$, $x \mapsto \Phi x$, and $x \mapsto BB^\intercal x$ can be evaluated in $\mathcal{O}(n)$.*

Assumption 3.1 is fulfilled naturally in many dynamical systems, in which local operators introduce sparsity into the dynamics, as, e.g., in finite-difference approximations of spatial differential operators. Another example is Kronecker structure in the system matrices arising in spatiotemporal GP regression by assuming the covariance function is separated into a product over the respective spatial and temporal components [180, 195, 84, 207].

> **Assumption 3.2** *The map $x \mapsto Hx$ can be evaluated in $\mathcal{O}(m)$.*

> **Assumption 3.3** *The map $x \mapsto R^{-1/2}x$ and the log-determinant $\log\left|R^{1/2}\right|$ can be evaluated in $\mathcal{O}(m)$.*

Throughout this work, we refer to the situation in which Assumption 3.1 or 3.2 do not apply as the “worst case”. Assumption 3.3 is taken for granted as the measurement-noise covariance R is often a diagonal matrix, which implies that sensor errors are uncorrelated. This is not only realistic but also commonly imposed in modelling. The above assumptions allow the following proposition.

> **Proposition 3.3** *Given Assumptions 3.1 to 3.3, the proposed method approximates the filtering densities and the marginal likelihood at a cost of* $\mathcal{O}(nr^2 + mr^2 + r^3)$.

The below corollary follows from the book-keeping in the proof of Proposition 3.3 in Section A.1.2.

> **Corollary 3.4** *When Assumption 3.1 or Assumption 3.2 are not satisfied, the worst-case complexity of the proposed low-rank filtering recursion is* $\mathcal{O}(n^2r + nm + m^2r)$.

Overall, the proposed filtering recursions achieve the same asymptotic complexity as the existing ensemble methods under the same set of assumptions. However, our method is entirely deterministic.

3.3.4 The rank-reduced smoothing recursion

It remains to obtain a recursion for the smoothing densities, which can be shown to be computationally tractable given that the filtering recursion is tractable. The smoothing densities are denoted by

$$p(\mathrm{X}_l \mid \mathrm{Y}_{1:N}) = \mathcal{N}(\mathrm{X}_l; \xi_l, \Lambda_l), \quad \xi_N = \mu_N,\ \Lambda_N = \Sigma_N. \tag{3.24}$$

It can be shown that the posterior process has a backward Markov representation [30], hence the smoothing marginals may be obtained by

$$p(\mathrm{X}_l \mid \mathrm{Y}_{1:N}) = \int b_{l,l+1}(\mathrm{X}_l \mid \mathrm{X}_{l+1}) p(\mathrm{X}_{l+1} \mid \mathrm{Y}_{1:N}) \mathrm{d}\mathrm{X}_{l+1}, \tag{3.25}$$

where we call $b_{l,l+1}$ the *backwards kernel.* Consequently, the problem consists of approximating $b_{l,l+1}$ such that the marginalization Equation (3.25) may be implemented in a computationally frugal manner.

APPROXIMATING THE BACKWARDS KERNEL

For the linear Gaussian case, the backward kernel [214]

$$b_{l,l+1}(\mathrm{X}_l \mid \mathrm{X}_{l+1}) = \mathcal{N}(\mathrm{X}_l; G_l \mathrm{X}_{l+1} + v_l, P_l), \tag{3.26}$$

is parametrized by the *smoothing gain* $G_l \in \mathbb{R}^{n \times n}$, the shift vector $v_l \in \mathbb{R}^n$, and the covariance of the backwards kernel $P_l \in \mathbb{R}^{n \times n}$. Let $(\cdot)^\dagger$ denote the Moore–Penrose pseudoinverse. Then

$$G_l = \Sigma_l \Phi_{l+1}^\intercal \Pi_{l+1}^\dagger, \quad v_l = \mu_l - G_l \mu_{l+1}^-, \quad P_l = (I - G_l \Phi_{l+1}) \Sigma_l (I - G_l \Phi_{l+1})^\intercal + G_l Q_{l+1} G_l^\intercal. \tag{3.27}$$

In the low-rank setting, the backwards kernel is efficiently approximated given the results from above. The approximate smoothing gain is a product of a tall, a small quadratic, and a wide matrix:

$$G_l \approx \Sigma_l^{1/2} \underbrace{\Sigma_l^{\mathsf{T}/2} \Phi_{l+1}^{\mathsf{T}} \left(\Pi_{l+1}^{1/2}\right)^{\dagger}}_{=:\Gamma_l \in \mathbb{R}^{r \times r}} \left(\Pi_{l+1}^{\mathsf{T}/2}\right)^{\dagger}. \tag{3.28}$$

During filtering, Γ_l is saved alongside the low-rank factors of the prediction and filtering covariances. We proceed to compute a low-rank representation of the backwards-transition covariance. Consider the following singular value decomposition, truncated at the r-th largest singular value,

$$\begin{pmatrix}(I - G_l \Phi_{l+1})\Sigma_l^{1/2} & G_l Q_{l+1}^{1/2}\end{pmatrix} \approx \widehat{U}_l \widehat{D}_l \widehat{V}_l^{\mathsf{T}}, \tag{3.29}$$

where $\widehat{U}_l \in \mathbb{R}^{n \times r}$, $\widehat{D}_l \in \mathbb{R}^{r \times r}$ diagonal, $\widehat{V}_l \in \mathbb{R}^{r \times r}$, and G_l as given in Equation (3.28). Then a rank-r factor of P_l is given as

$$P_l^{1/2} = \widehat{U}_l \widehat{D}_l \in \mathbb{R}^{n \times r}. \tag{3.30}$$

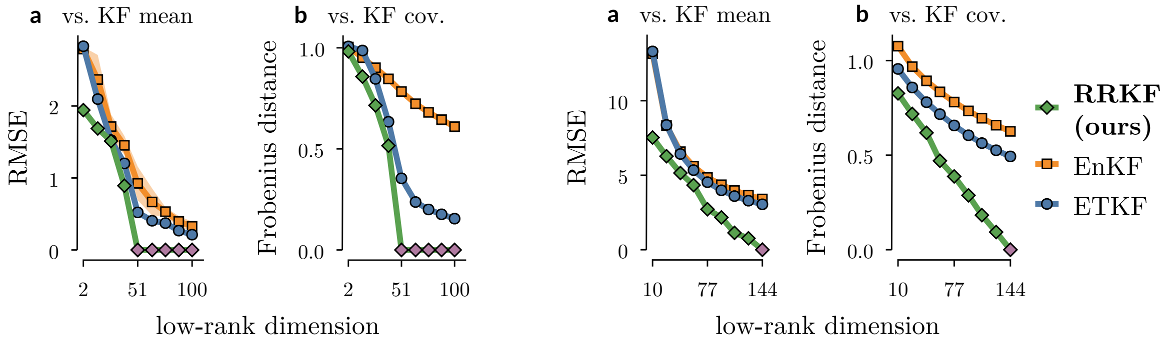
As for filtering, in the worst case, the cost for low-rank smoothing scales quadratically with the state dimension n. The following proposition, which is proved in Section A.1.3, states that this can be further reduced to linear complexity in n, given the assumptions on the dynamics model are satisfied.

> **Proposition 3.5** *Given Assumption 3.1, computing the smoothing density* $p(\mathrm{X}_l \mid \mathrm{Y}_{1:N})$ *costs* $\mathcal{O}(nr^2 + r^3)$.

Finally, it is useful to mention that realizations of the backwards process are posterior samples.

3.4 Experiments

This section evaluates the proposed method in different experimental settings. The chosen measure of quality is the distance of the approximate low-rank moments to the exact KF. We measure the mean deviations with the root-mean-squared error (RMSE) and the covariance deviations with the time-averaged relative Frobenius distances. The presented rank-reduced Kalman filter (RRKF) is compared to two ensemble-based methods that use stochastic transitions. The first baseline (EnKF) corresponds to the fully stochastic version of the ensemble Kalman filter, whereas the second baseline (ETKF) uses the deterministic update scheme of the ensemble transform Kalman filter. All EnKF and ETKF results are given in sample statistics over 20 runs. Section 3.4.1 shows that for a truly low-rank system, our method recovers the KF estimate up to numerical error as soon as r exceeds the true rank of the problem. Section 3.4.2 tests the method on a spatiotemporal GP regression problem with real data. After Section 3.4.3 evaluates the approximation quality for increasingly low-rank systems in a controlled experimental environment, Section 3.4.4 verifies the stated asymptotic cost of the method. Finally, a large-scale spatiotemporal regression problem is solved in



(a) **Linear advection dynamics ($r^* = 51$).**

(b) **London air-quality regression.**

Figure 3.2: **Performance of the low-rank filters on two different settings.** In the truly low-rank linear advection problem (a), the RRKF achieves the optimal estimate (◆) when r exceeds the true rank r^* of the problem. For the spatiotemporal air-quality regression (b), our method is consistently better.

Section 3.4.5. In all experiments, we compute the stationary mean and a low-rank factorization of the stationary covariance matrix of the prior and condition the stationary moments on the first measurement in the respective time-series dataset. More details on the experimental setups are given in Section A.4.

3.4.1 Linear advection model

The proposed algorithm is evaluated in a standard data-assimilation setup in which linear-advection dynamics with periodic boundary conditions are assimilated to a set of simulated data. The setup follows the description in Sakov et al. [177] and the experimental details are additionally detailed in Section A.4. This problem has rank 51 by construction. Figure 3.2a shows how the exact KF estimate is recovered by our method for $r = 51$, while the ensemble methods converge according to a Monte–Carlo rate. This is particularly clear for the covariance estimate (Figure 3.2a, **b**).

3.4.2 London air-quality regression

This experiment uses hourly data from the London air-quality network [99] between January 2019 and April 2019, which amounts to measurements at 72 spatial locations and 2159 points in time. The data used, together with its processing (except the train-test split), is the same as in the corresponding experiment by Hamelijnck et al. [84]. The model is a spatiotemporal Matérn-$3/2$ process with prior hyperparameters that maximize the marginal log likelihood. Figure 3.2b shows that—compared to the ensemble methods—our algorithm is consistently closer to the KF estimate and recovers it at $r = n$.

3.4.3 Spatiotemporal Matérn process with varying spatial lengthscale

Consider a spatiotemporal Gaussian process $\mathrm{X}(t) \sim \mathcal{GP}(0, k_t \otimes k_x)$ with covariance structure that is separable in time and space. Let both k_t and k_x be of the Matérn family with

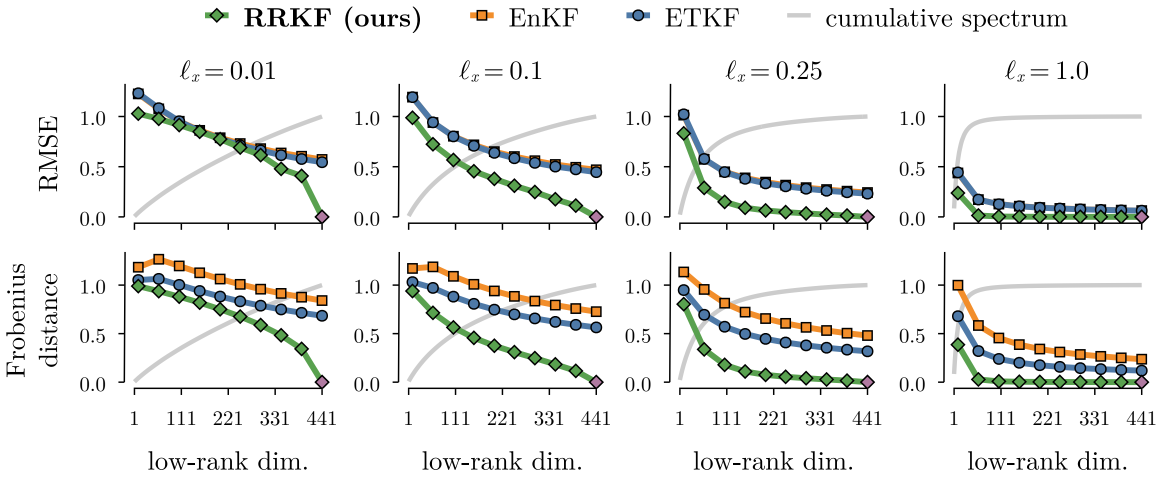


Figure 3.3: **How does the rate of spectral decay influence the low-rank approximation?** The larger the spatial length scale of a spatiotemporal Matérn model, the faster the spectrum decays and the faster all methods converge to the KF estimate. The first and second row compare the mean estimates and the covariance estimates, respectively. The RRKF estimate is consistently closer and recovers the KF estimate (◆) at $r = n$. The cumulative spectrum of the final-step KF covariance is shown in grey.

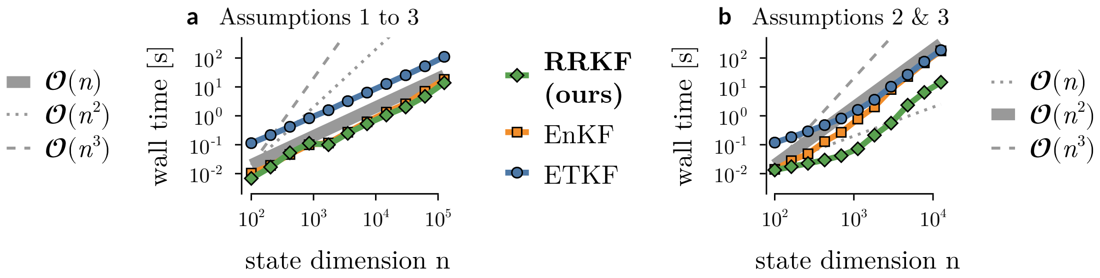


Figure 3.4: **Best-case and worst-case asymptotic complexities of the low-rank filters.** When all Assumptions 3.1 to 3.3 are satisfied (**a**), the elapsed time grows linearly with the state dimension n. If parts of the assumptions are not met (**b**), the cost scales quadratically with the respective dimension.

characteristic spatial and temporal lengthscales ℓ_t, ℓ_x and output scales σ_t^2, σ_x^2. Spatiotemporal Matérn processes can be translated to the formulation in Equation (3.1a), as detailed, for instance, by Solin [195]. This setting allows for varying the "low-rankness" by the choice of ℓ_x: the larger ℓ_x, the more information about a spatial point is prescribed by its surroundings. As ℓ_x approaches zero, the spatial kernel matrix gets more and more diagonal, encoding spatially independent states, and causing a barely decaying spectrum. The spectrum for large ℓ_x decays rapidly, making low-rank approximations more accurate for small r.

In the experiment, we consider a spatial domain $[0, 2] \times [0, 2] \subset \mathbb{R}^2$, which is subsampled at a uniformly-spaced grid with $\Delta_x = 0.1$ ($n = 21 \times 21 = 441$). Noisy observations of the full state trajectory, are drawn from a realization of the prior process. Figure 3.3 demonstrates that, for increasing spatial lengthscales, all methods converge faster to the true KF estimate. Crucially, however, our method is consistently closer to the optimal estimate and recovers it for $r = n$.

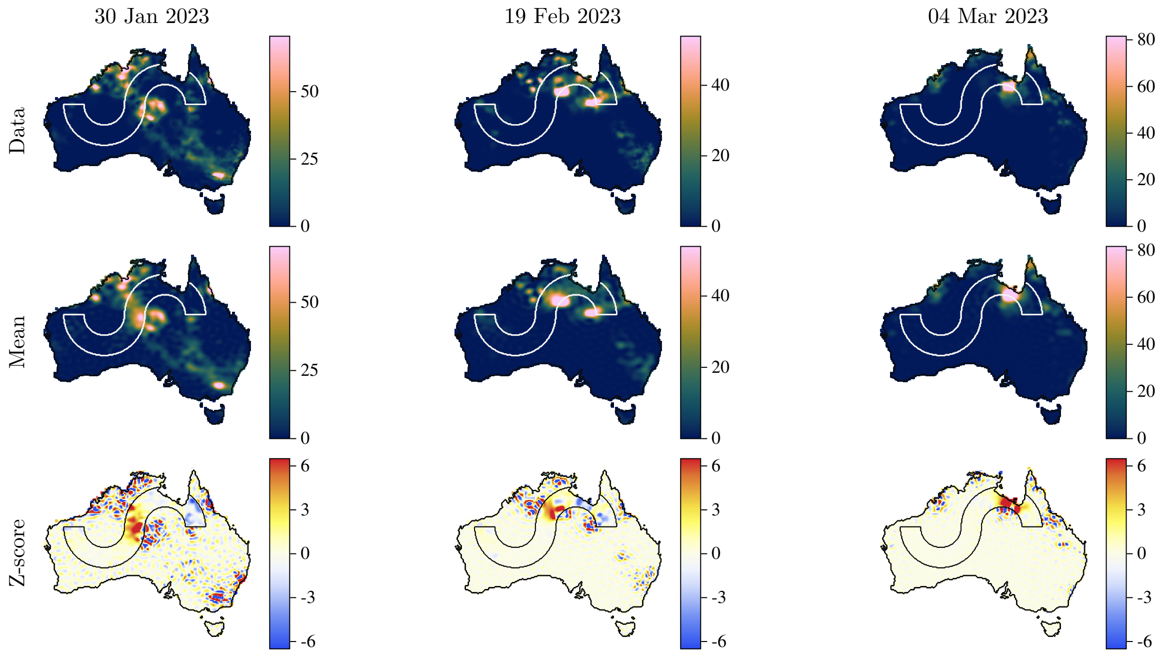


Figure 3.5: **Rainfall in Australia.** A spatiotemporal GP regression problem with $n = 30\,911$ state dimensions, solved with the RRKF with $r = 1000$. The columns are three different days in the time series. The rows show the data, mean estimate, and Z-scores, respectively.

3.4.4 Runtime

To demonstrate the asymptotic computational complexities given in Section 3.3.3, we investigate the runtimes on two different settings: The first fulfills both Assumptions 3.1 and 3.2, whereas the second only fulfills Assumption 3.2, which means that the cost of prediction is quadratic in the state dimension. Both problems are solved on a temporal grid of size $N = 100$ with the measurement dimension fixed at $m = 100$ and the low-rank dimension fixed at $r = 5$.

The problem from Section 3.4.1 serves as the best-case setting. The noise-free linear-advection dynamics amount to multiplication of the state with a circulant matrix, which is implemented efficiently using fast Fourier transform. The measurement operator amounts to array indexing. As subplot **a** in Figure 3.4 confirms, the computational cost grows linearly with the state dimension. As a worst-case example we employ a spatiotemporal Matérn process with a dense spatial kernel matrix (cf. Section 3.4.3). In this case, we expect the computation time to scale quadratically with the state dimension because $BB^\intercal$ is dense. This is verified by the right plot in Figure 3.4 (**b**).

3.4.5 Large-scale spatiotemporal GP regression on rainfall data

As a final experiment, a spatiotemporal GP regression problem is solved on a large gridded analysis data set measuring rainfall in Australia. The data is provided by the Australian Water Availability Project (AWAP) [1, 103]. The number of spatial data points is $n = 30\,911$, which rules out the cubic-in-n Kalman filter. The smoothing posterior is computed at $N = 40$ time points, from 25 January, 2023 through 5 March, 2023. The total number of observations is thus $n \cdot N = 1\,236\,440$. The spatiotemporal prior Matérn model is selected by maximizing the marginal log-likelihood of a lower-resolution data set. A set of 6160 spatial points is excluded from the data during filtering and smoothing in order to evaluate spatial interpolation performance. Thus, the measurement dimension is $m = n - 6160 = 24751$. Figure 3.5 shows the low-rank approximation to the GP posterior with $r = 1000$ low-rank dimensions. The non-observed spatial locations are framed in the horizontal-“S” shape. At measurement locations, the data is described well by the model, while a slight smoothing-out effect can be observed due to the high-frequency features of the model being truncated. At the evaluation points the interpolation aligns with the expectations regarding the simple Matérn model. The Z-score map shows $\frac{\xi_l - \mathrm{y}_l}{\lambda_l}$, where ξ_l and λ_l denote the smoothing mean and marginal standard deviation and y_l the data point, at time t_l. This highlights badly calibrated uncertainty estimates, wherever the model under-/overestimates the data and is divided by a small standard deviation, in blue/red. The Z-score distribution receives further attention in Section A.5. All in all, the RRKF achieves high-quality approximate estimates while compressing the state by a factor of $\frac{30\,911}{1000} \approx 30$.

3.5 Limitations

The main limitation of the proposed approximate probabilistic inference scheme is that the truncation to r dimensions cuts away covariance information, which is not accounted for. As a result, confidence *grows* as r shrinks, while estimates should arguably get more uncertain instead with increased compression. In existing low-rank filters, this issue is often counteracted manually by inflating the covariance matrices, on a per-application basis [31, Section 4.4] and calls for a more principled treatment. Finding a way to keep track of that residual uncertainty information is beyond the scope of this paper. It remains unclear how to preserve the stated asymptotic complexities while doing so. Further, all results herein are stated for linear dynamics and observation models. Extensions to non-linear models could include linearization of the according transitions, or cubature methods [179].

3.6 Conclusion

By building upon well-established knowledge about optimal compression, we have proposed an algorithm providing a principled way to balance approximation accuracy against computational cost in high-dimensional state estimation. It combines simple (truncated) singular value decompositions with recent numerical low-rank integrators of large matrix differential equations. We have offered both theoretical and empirical arguments for why it is desirable to use a deterministic algorithm when approximating large-scale probabilistic state-estimation problems.

4 Probabilistic Spatiotemporal Climate Downscaling

Local climate information is crucial for impact assessment and decision-making, yet coarse global climate simulations cannot capture small-scale phenomena. Current statistical downscaling methods infer these phenomena as temporally decoupled spatial patches. However, to preserve physical properties, estimating spatiotemporally coherent high-resolution weather dynamics for multiple variables across long time horizons is crucial. We present a novel generative framework that uses a score-based diffusion model trained on high-resolution reanalysis data to capture the statistical properties of local weather dynamics. After training, we condition on coarse climate model data to generate weather patterns consistent with the aggregate information. As this predictive task is inherently uncertain, we leverage the probabilistic nature of diffusion models and sample multiple trajectories. We evaluate our approach with high-resolution reanalysis information before applying it to the climate model downscaling task. We then demonstrate that the model generates spatially and temporally coherent weather dynamics that align with global climate output.

4.1 Introduction

Numerical weather and climate simulations based on discretized solutions of the Navier-Stokes equations are fundamental to understanding large-scale weather patterns, climate variability, and climate change. State-of-the-art numerical weather prediction (NWP) models, which primarily focus on atmospheric processes, can accurately resolve small-scale dynamics within the Earth system, providing fine-scale spatial and temporal weather patterns at resolutions on the order of kilometers [14]. However, the substantial computational resources required for these models render them impractical for simulating the extended time scales of multiple years and decades necessary to assess climatic changes. Moreover, even with substantial computational investment, global high-resolution models can still exhibit systematic biases and may fail to accurately reproduce observed climatic trends [151]. In contrast, Earth system models (ESMs), such as those included in the CMIP6 project [63], incorporate a broader range of processes—including atmospheric, oceanic, and biogeochemical interactions—while operating on coarser spatial scales. Typical grid resolutions for ESMs are approximately 1°, equivalent to around 100 km. This coarse resolution limits the ability of ESMs to fully capture small-scale processes. Key processes necessary to assess regional impact, for example, on wind turbines—such as local wind turbulence—occur at spatial and temporal scales that are too fine to be explicitly resolved in ESMs. Consequently, ESM data cannot be directly employed to evaluate changes at fine spatial scales, limiting their utility for localized impact assessment and decision-making.

Downscaling aims to provide regional climate information by estimating small-scale processes from coarse simulations of global models. Existing approaches to bridge this scale gap can be categorized into dynamical and statistical downscaling [140]. Dynamical downscaling employs high-resolution regional climate models (RCMs) that are nested within coarser global climate models (GCMs) to provide detailed projections for specific regions [127, 206]. Biases from the driving global models can be inherited by the RCMs, potentially limiting the accuracy of the downscaled results [170]. Additionally, the high computational cost of running RCMs restricts their use primarily to regional studies. Statistical downscaling methods use regression or weather generators [223], and, more recently, emulators based on machine learning (ML) techniques. In statistical downscaling, a functional or statistical relationship between large-scale climate variables (from GCMs) and local observations is established. Based on this relationship, statistical methods infer the local information from the coarse simulation, with ML-based approaches aiming to capture the mapping from coarse to fine-scale climate by learning the relationships from data [146, 16, 12, 11]. These models are usually computationally less expensive. However, not explicitly encoding physical laws can make them less robust in regions of low data density, risking physical inconsistencies.

One challenge that comes with downscaling arises from the fact that numerical models are inherently imperfect representations of the climate system. While ESMs are designed to generate accurate multi-decadal summary statistics, locally, the simulations differ from historical observational datasets. Climate models contain inaccuracies in parameterization, simplified process representations, uncertainties in the initial state of the system [192]. Even in the hypothetical case of perfect models, i.e., without epistemic uncertainty, forecasts are not deterministic due to the chaotic nature of the atmosphere. These variations lead to substantial discrepancies between models. Choosing a model, therefore, becomes a critical factor [46, 152]. Aside from other climate model biases [38], internal variability of the climate system leads to differences between projected and observed climate [102]. Individual climate simulations thus represent only one possible realization of the system with substantial uncertainty remaining. Due to this un-pairedness of ESM outputs and observational data, using supervised ML approaches, which rely on consistent simulation–observation pairs, remains challenging [91, 5]. Generative models have recently emerged as a promising solution. This model class is characterized by learning a representation of the training data distribution that allows the generation of novel samples. As self-supervised learning techniques, these models circumvent the need for data–label pairs by working solely on the target (output) distribution. Additionally, through the variability among generated samples, generative models provide structured uncertainty, which the ill-posed nature of most inference problems entails. In particular, diffusion models (DMs) [94, 199, 196, 174] have demonstrated superior performance over earlier approaches such as variational auto-encoders (VAEs) [109], generative adversarial networks (GANs) [76], and normalizing flows [45], particularly for structured data and image generation tasks [44, 29, 116].

In this work, we build on the score-based data assimilation (SDA) framework by Rozet et al. [174] and Rozet et al. [175] to phrase downscaling as a Bayesian-inference problem with a generative prior model. By construction, our model performs *joint* spatial and temporal downscaling on multiple variables, introducing stochasticity solely between samples. The predictions are thus coherent across the entire state space, avoiding sampling-induced inconsistencies between time steps and between interrelated atmospheric variables. Furthermore, the model training is separated from the task-specific inference (Figure 4.1 **I** & **II**), which makes the model flexible with respect to its input and the statistical relationship between input and output—allowing, for instance, downscaling climate simulations of varying spatiotemporal resolutions without retraining the model. The components of the algorithmic pipeline are outlined in Figure 4.1:

A The score model (Figure 4.1 🟦), which is the centerpiece of the generative diffusion model, is trained on high-resolution reanalysis data. The score-based diffusion model provides a statistical representation of local weather dynamics. This *"prior"* model gives access to samples from the distribution $p\left(\mathrm{X}_{\text{reanalysis}}\right)$, which represents our prior concept of the output space. Here, the output space includes the spatial region, the target resolution, and learned dynamics patterns for a set of selected atmospheric variables.

B The coarse ESM input Y_{ESM} is pre-processed; in particular, a bias-correction procedure can align the simulation with the reanalysis data in terms of its value distribution.

C The *"observation model"* $p\left(\mathrm{Y}_{\text{ESM}} \mid \mathrm{X}_{\text{reanalysis}}\right)$ (Figure 4.1 🟨) establishes the functional or statistical relationship between coarse climate output and fine reanalysis data. This assumes the ESM output to be a perfect prognosis [114, 138], based on which local climate is estimated. In the context of inverse problems, the observation model is often referred to as the "forward model", as it models how the partially observed quantity (Y_{ESM}) arises from the latent quantity of interest ($\mathrm{X}_{\text{reanalysis}}$)—usually by removing information.

D Accordingly, the inverse problem aims to predict the (much harder, underspecified) opposite direction: estimating the *"posterior"* $p\left(\mathrm{X}_{\text{reanalysis}} \mid \mathrm{Y}_{\text{ESM}}\right)$ requires adding information to the incomplete observations by conditioning the prior. In our case, the trained score model is coupled with the observation model to generate samples from the posterior distribution.

Perspectively, the algorithmic framework is versatile and flexible enough to be considered beyond downscaling. In the spirit of foundation models [128, 124, 162, 226], the trained generative model can be combined with other observation models in a similar zero-shot manner to solve various inference tasks on the target output space. Unlike approaches that require training the model directly on the conditioning information, the presented framework allows the formulation of explicit—and varying—functional or statistical relationships between observations and predictions.

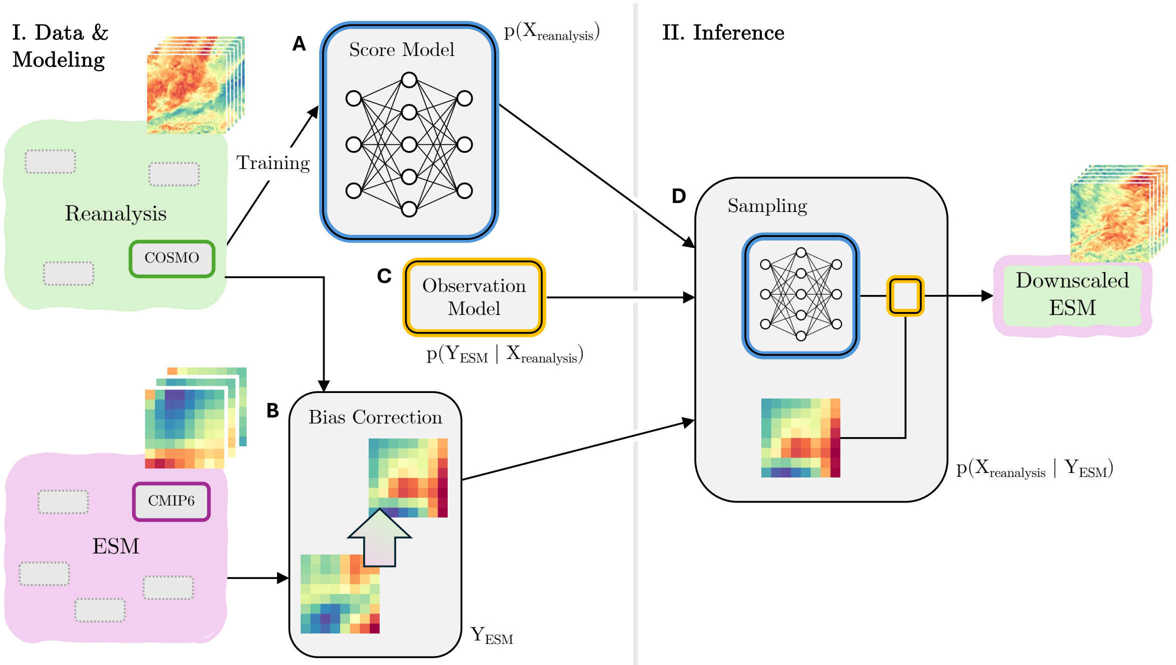


Figure 4.1: **Probabilistic pipeline for spatiotemporal downscaling of multiple variables.** This work introduces a probabilistic downscaling framework that jointly predicts fine-scale and spatiotemporally consistent time series for multiple variables from coarse ESM simulations. This schematic outlines the framework. Only one exemplary variable is shown for visual clarity. **A**: A score model is trained on sequences of reanalysis data. This is the centerpiece of the diffusion model that learns to reproduce the fine-scale spatial and temporal patterns. Note that ESM simulations are not part of the training process. **B**: From any ESM (e.g. CMIP6), select and pre-process an ensemble run (e.g. MPI-HR) for downscaling. A bias-correction step that mitigates distributional deviations between climate output and reanalysis data can be applied as a pre-processing step. **C**: This part establishes a relationship between coarse climate simulations and the (fine-scale) output space of the model. The observation model defines how the observed quantity (Y_{ESM}) is generated—or observed—from the inferred quantity ($X_{\mathrm{reanalysis}}$). The observation model is key to impose a constraint onto the generative model such that its samples adhere to the established relationship. **D**: The model generates time series that preserve the statistics of the coarse climate input. During the generative process, the trained score model (**A**) is conditioned, i.e., the predictions are informed by the conditioning information (**B**), such that they adhere to the relationship established by the observation model (**C**).

On a series of experiments, we demonstrate that the proposed model generates coherent time series of regional climate that are aligned with coarse input. The model predicts local weather dynamics, including extreme events such as winter storms. Sampling from the posterior distribution enables the generation of multiple weather trajectories crucial for assessing the internal variability and uncertainty of the downscaling problem, providing a comprehensive understanding of future weather scenarios. We include a simple quantile-mapping procedure as a pre-processing step to the ESM simulations. Other than that, the ESM simulations are taken as a perfect predictor and the mismatch to the observational data is not accounted for. We evaluate our downscaling framework on two different GCMs from CMIP6 in order to show that different mismatches between reanalysis and the respective ESM distributions remain. We then demonstrate that our model maintains the coarse, global properties of the respective ESM input while inferring local fine-scale weather patterns.

4.2 Results

The presented downscaling pipeline assumes a statistical relationship between a coarse numerical model prediction and a fine-scale reanalysis product [114, 138]. We begin by evaluating the methodological framework in an artificial setup, in which coarsened reanalysis data serve as the perfect prediction [141] and surrogate the ESM simulations. The coarse data is obtained from spatial area averages and by selecting a subset of the time steps. Concretely, the observation model establishes the following relationship between the downscaled predictions X and the coarse input Y:

$$\mathrm{Y}_t^{(i,j)} = \frac{1}{|\bar{i}| \cdot |\bar{j}|} \cdot \sum_{(i',j') \in \bar{i} \times \bar{j}} \mathrm{X}_t^{(i',j')}, \tag{4.1}$$

where (i, j) is a single point on the coarse spatial grid, which represents an area of multiple points $\bar{i} \times \bar{j}$ on the fine grid. For example, $\bar{i}/\bar{j}$ can be a local neighborhood around i/j in the longitude/latitude dimension. The observation model in Equation (4.1) assumes that the coarse information is provided as a snapshot at time points t. In our experiments, one coarse-grid point (i, j) encompasses a 16×16-area on the fine grid, i.e., $|\bar{i}| \cdot |\bar{j}| = 16^2 = 256$. Furthermore, the temporal resolution of the coarse grid is six hours, whereas the fine output time grid is resolved hourly. This setting enables a direct pairing between climate output and reanalysis data, which is not given in reality [141], allowing for evaluating the model's predictive performance and uncertainty quantification by comparing it to a ground truth. Having established the validity of the downscaling method on this artificial setup, we will then proceed to downscale climate output, assuming two different ESM simulations from the CMIP6 project to be the perfect predictors.

4.2.1 Evaluation of predictive distribution and uncertainty calibration

We first evaluate the basic capability of the model to adhere to the established statistical relationship between coarse and fine predictor (cf. Equation (4.1)) and to preserve the coarse-

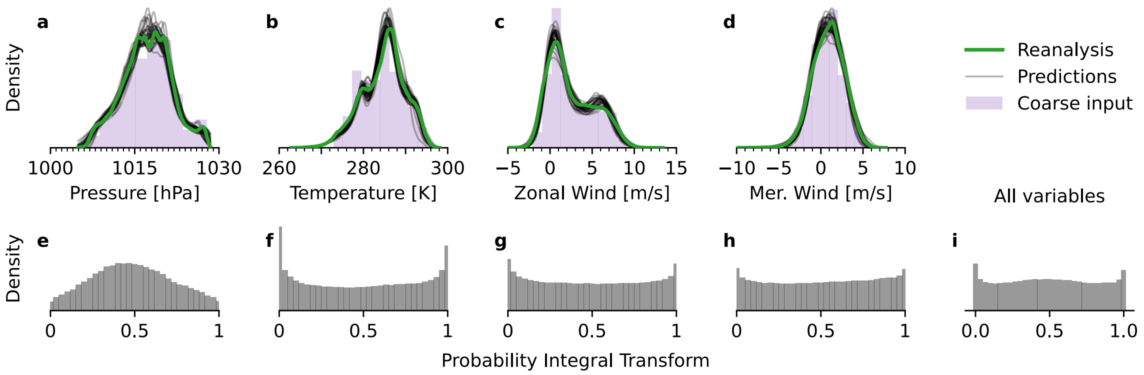


Figure 4.2: **Comparison of value distributions: reanalysis data, coarse input, predictions.** For a time range of 49 hours, this plot shows that the 1-hourly local predictions, which the model predicted from the coarse 6-hourly input, resemble the reanalysis data closely in distribution. *Top row:* Kernel-density estimations for value distributions of reanalysis data (green) and 30 predictions (black) for each variable separately (**a**–**d**). The prediction model was conditioned on coarse inputs (purple). The predicted samples align with the reanalysis data distribution, which is fully covered by the predictive uncertainty. *Bottom row:* The probability integral transform (PIT) demonstrates the uncertainty calibration of the model: for each variable separately (**e**–**h**) and overall (**i**). A PIT distribution that resembles a standard uniform distribution indicates that the reanalysis data and predictions likely come from the same distribution.

data value distribution in its predictions. Aggregating the values over temporal and spatial domains, we find that—per variable—the estimated densities of the predictions each align with the reanalysis data. The uncertainty induced by the sample spread covers the reanalysis distribution both near the modes and in regions of low and high quantiles. In particular, the probabilistic model introduces no systematic biases, like distribution shift, a tendency towards over- or under-predicting values, or mismatch in the tail regions. Instead, each sampled prediction captures the spread of the data distribution. Figure 4.2 **a**–**d** visualizes these findings using density estimations of the respective value distributions.

To assess the calibration of the predictive uncertainty, we compute the probability integral transform (PIT) [43, 205] for all values aggregated and for each variable separately. If the resulting distribution (Figure 4.2 **e**–**i**) resembles a standard uniform distribution on the interval $[0, 1]$, it is likely that the samples and the reanalysis data come from the same underlying distribution. We find that both wind-speed components (Figure 4.2 **g**,**h**) are well calibrated. The PIT reveals that the predictions for mean sea-level pressure (Figure 4.2 **e**) are slightly underconfident, over-predicting extreme values. For surface temperature (Figure 4.2 **f**), the opposite is the case: the predicted distribution is slightly too narrow and under-predicts tail events. Taken the joint distribution of variables together (Figure 4.2 **i**), the model is well calibrated.

4.2.2 Predicting local dynamics from coarse information

We require our downscaling model to augment the scarce information contained in the coarse input by adding nontrivial, local weather patterns and predicting complex temporal and spatial

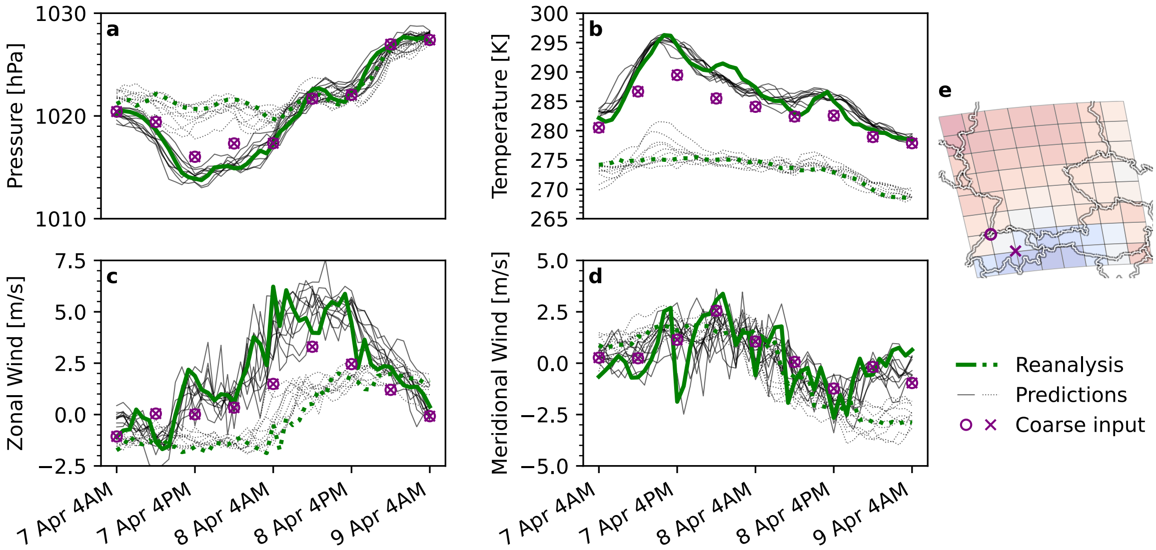


Figure 4.3: **Differences in local dynamics are inferred from coarse-grained observations.** This plot compares time series at two locations over 49 hours between reanalysis data (green), 30 predictions (black), and coarse input (purple) and shows that the model can accurately extract local dynamics from shared coarse information. Both fine-grid locations were selected to share a single point on the coarse grid (**e**). Each plot (**a**–**d**) shows one variable. Reanalysis and predicted weather trajectories are shown in solid and dotted lines for both locations, respectively. The conditioning information (purple circles and crosses) is a single value every six hours that is shared by both locations (**e**), at which the local weather dynamics are inferred. The predicted time series aligns with the reanalysis data at both locations. In particular, the uncertainty obtained through sampling multiple predictions covers the reanalysis data, and the individual samples mirror the local weather trajectories in their respective temporal structure.

dynamics. As described above, in our experimental setup, a single scalar measurement informs a six-hour window of 16×16 fine-grid locations, requiring the model to perform a mapping from a single node to $6 \times 16 \times 16 = 1\,536$ nodes. Hence, we need to assess whether the model sensibly incorporates prior knowledge, which it learned from data, in order to evaluate the plausibility of the added, generated information.

It is likely that—due to, for instance, varying environmental conditions—the dynamical patterns at distinct locations on the reanalysis grid differ substantially from each other, whereas the coarse-grid aggregation occludes these local variations, motivating the downscaling problem in the first place. The present input-output-paired setup allows us to investigate the true local variations that are lost through aggregation by comparing the reanalysis data at two distant locations within the 16-by-16 area that is encompassed by a single coarse-grid location. Accordingly, we can compare these ground-truth variations with our model predictions to assess whether the model infers local dynamics that align with the reanalysis data.

In Figure 4.3, we demonstrate that our model accurately predicts spatial and temporal variations in weather trajectories at two distinct locations, which share a single spatial observation. For this experiment, we selected two distant fine-grid locations near the Alps, for which we can expect substantial local variations (see purple circle and cross in Figure 4.3 **e**). We visualize the time series for the reanalysis data at both locations (solid and dotted green line) and the corresponding downscaled model predictions (solid and dotted black lines) alongside the coarsened input (purple circle and cross), which share a single value at every 6-hourly step. The visualization allows to identify the local variations that are occluded in the spatiotemporal aggregation by comparing the coarse observations (purple) to the fine ground-truth time series. Our model accurately predicts the ground-truth dynamics at both locations from the coarse information. Notably, the distinct features in temporal structure (e.g. smoothness and amplitude) of the predicted time series at both locations align with those found in the corresponding reanalysis data. As an illustrative example, the zonal wind speeds (Figure 4.3 **c**) show substantial variability between the two locations—both in their values and their temporal structure. The model captures these variations in its predictions, highlighting its ability to recover local-scale variability lost in the coarse-resolution observations. This is a promising indicator that the downscaling model has learned an accurate representation of local spatial and temporal patterns from the training data, which it blends into the coarse conditioning information to estimate coarsely informed local variations.

4.2.3 Spatiotemporally consistent weather trajectories: studying a winter storm

Statistical downscaling becomes particularly challenging during extreme events, which are rare and fall into the tails of the training distribution. These events often involve highly complex dynamics. However, extreme events are particularly interesting, as they often have the most significant societal and environmental impacts. As Figure 4.2 already demonstrated, the downscaling model is capable of accurately predicting high quantiles. Here, we supplement the evaluation of the aggregated value distributions with a qualitative assessment of the

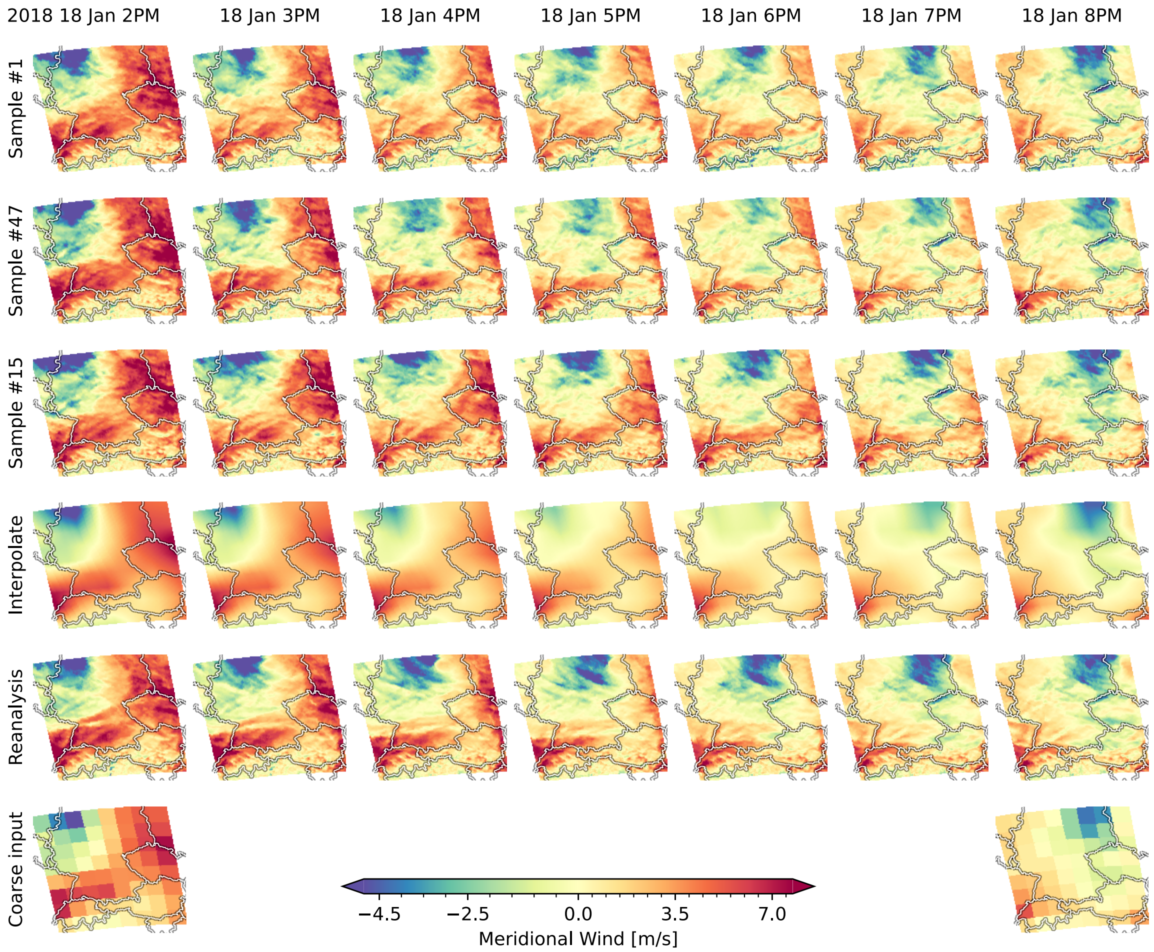


Figure 4.4: **Predicting high-resolution dynamics during a cyclone as an extreme event.** This plot shows meridional wind trajectories of three randomly selected model predictions (top three rows), spatiotemporal interpolation (fourth row), reanalysis data (fifth row), and coarse conditioning information (bottom row) during a cyclone ("Friederike", January 2018). The sign of the wind speed value defines its direction: negative values go southward, and positive values go northward. Time progresses from left to right hourly, starting 2018 January 18 at 02:00 PM and ending the same day at 08:00 PM. Between the first (2:00 PM) and the last (8:00 PM) visualized time point, no information is provided to the model. The top three rows show how different samples add missing spatial and temporal information, while the interpolation (fourth row) can only spread out the existing, coarse information. Each individual generated trajectory aligns visually with the coarse input, while the variation among the samples captures the uncertainty associated with the inference problem. Notably, the model does not introduce implausible “jumps” from one time step to the next but interpolates with spatially and temporally consistent dynamics.

spatiotemporal structure of the downscaled predictions during a winter storm. To this end, we consider the time range in which cyclone "Friederike" approached central Europe (including the modeled spatial region) from the west around January 18, 2018. Aside from the time period, the experimental setup remains unaltered from the above sections. Figure 4.4 exemplary shows the meridional wind speeds during the event. The visualization compares three randomly selected model predictions, a spatiotemporal interpolation of the coarse input, the ground-truth reanalysis data, and the coarse input. The model predictions are coherent in space and time and add different local variations to the coarse observations. A comparison to the spatiotemporal interpolation (Figure 4.4, fourth row) highlights the capability of the downscaling model to predict nontrivial fine-scale patterns. Figure B.9 visualizes anomalies—differences between the interpolation and a) downscaled predictions and b) reanalysis data. This visualization reveals the spatiotemporal patterns, which are lost by aggregating the reanalysis data, and allows a comparison to the disaggregated fine-scale structure predicted by our downscaling model. Additionally, Figure B.2a demonstrates a close match of power spectral densities between reanalysis data and model predictions, providing a more quantitative argument.

4.2.4 Downscaling ESM simulations

The above experiments evaluated the model in a setting that ensures that the large-scale predictor perfectly matches the statistics of the fine-scale reanalysis data. We proceed to downscaling climate-model outputs, using 6-hourly CMIP6 ESM simulations as conditioning information. As in the previous experiments, the four variables considered are downscaled spatially (by a factor of 16×16) and temporally (by a factor of 6) to align with the resolution of the reanalysis data. The same statistical relationship (Equation (4.1)) between coarse input and downscaled output is assumed in this experiment. Downscaling climate-model outputs, however, presents a greater challenge because there is no direct pairing between coarse- and fine-scale climate data, which renders the comparison of our downscaling predictions to a ground truth impossible. The goal of this experiment is to predict local patterns on a fine spatiotemporal grid, assuming two distinct perfect predictors given by two realizations from the distribution of coarse climate simulations. We evaluate whether our downscaling model—while adhering to the statistical relationship imposed between each prediction and different coarse ESM simulations (Figure 4.5)—is capable of simultaneously predicting local climate (Figure 4.6). Notably, climate model biases are not encoded directly and therefore not addressed by the downscaling model. As motivated by Volosciuk et al. [220], downscaling can be separated into, firstly, mitigating climate model biases and, secondly, bridging the gap between coarse and fine grids. We adopt this perspective, thereby focusing almost exclusively on the latter. As a pre-processing step, we apply a per-variable quantile-mapping bias-correction [139] to the ESM outputs.

Figure 4.5 **a**–**d** shows two model ensembles of the four variables from different climate models: the MPI-HR (purple) and the HadGEM model (yellow) for the year 2014 and, in comparison, the reanalysis data distribution (green curve). Comparing the distributions of the two ESMs, it becomes evident that, even in terms of their aggregated-value distribution, climate model

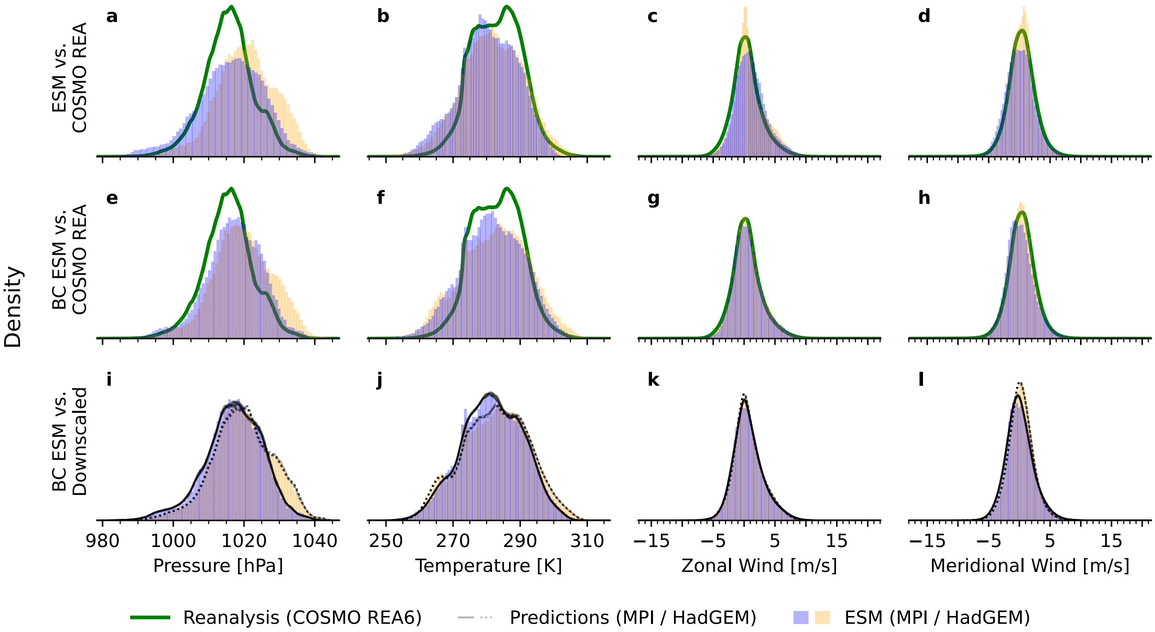


Figure 4.5: **Comparison of value distributions: ESM, de-biased ESM, reanalysis data, predictions.** This plot visualizes aggregated value distributions of reanalysis data (green), ESM simulations (two different models: MPI in purple, HadGEM in yellow), and downscaled predictions (black). The *top row* compares the raw, uncorrected ESM simulations with the reanalysis data to visualize the biases in the respective climate distributions. The *middle row* shows the bias-adjusted ESM simulations alongside the same reanalysis data to visualize the effect of the quantile-mapping procedure. The *bottom row* compares the value distribution of the bias-adjusted ESM simulations (same as middle row) with their downscaled counterpart (black). The eight solid/dotted lines correspond to downscaled predictions of the MPI/HadGEM model. Two things are demonstrated: Comparing the *top row* and the *middle row* shows that the distribution mismatch between both ESM ensembles and the reanalysis data (**a**–**d**) is mitigated through the bias-correction step (**e**–**h**). There is some mismatch remaining, likely due to the short evaluation period. Secondly, the *bottom row* shows for each variable separately (**i**–**l**) that the distribution of the downscaled climate trajectories aligns with the coarse model input. The considered time range is the year 2014. The model predicts 1-hourly steps starting January 01 at 06:00 AM and ending December 31 at 06:00 AM based on the corresponding 6-hourly coarse input. Visualizing two distinct CMIP6 ensembles (MPI and HadGEM) allows a comparison of the distributional mismatch a) between the respective climate outputs (purple vs. yellow) and b) between climate outputs and reanalysis data.

outputs can differ significantly from each other and, moreover, each individual ESM differs substantially from the reanalysis data. Figure 4.5 **e**–**h** visualizes the effect of the quantile-mapping bias adjustment, which adjusts the marginal distributions of each variable in the ESM simulations to better match those of the reanalysis data. As a result, the bias-corrected ESM simulations more closely align with the reanalysis data (green curve). A comparison of the uncorrected and bias-corrected distributions (Figure 4.5 **a**–**d** vs. **e**–**h**) reveals that some residual differences remain. Figure 4.5 **i**–**l** visualizes the distribution of eight downscaled predictions (black) for each of the two climate models. The downscaled predictions for MPI-HR (solid lines) and HadGEM (dotted lines) closely match the value distributions of their respective bias-corrected ESM simulations. Consequently, the downscaling model preserves the statistical properties of the coarse ESM input: the downscaled distributions are neither shifted nor skewed relative to their bias-corrected coarse counterparts, and the alignment extends to the tails of the distributions. As a result, any temporal changes or variations in the distribution shape present in the coarse input are faithfully retained in the downscaled output.

Figure 4.6 presents a qualitative assessment of the spatial and temporal progression of downscaled ESM simulations. The model output reproduces spatial patterns, which are, for example, consistent with geographical features, such as the Alps, and temporal patterns, like the day-night cycle, while introducing local weather dynamics consistent with the observations. Together, Figure 4.5 and Figure 4.6 show that the model can generate high-resolution weather trajectories that introduce complex local dynamics while preserving the value distribution of the respective ESM simulations.

4.3 Discussion

We introduced a probabilistic approach to joint spatial and temporal downscaling of multiple variables from climate to weather scale. The presented framework revolves around a generative diffusion model, which is trained to learn an implicit representation of the dynamical patterns in reanalysis data and serves as a probabilistic emulator for the forward dynamics model. By conditioning the forward model via an observation model on climate-model output, we can sample from a posterior downscaling distribution. Each sample drawn from this posterior adheres to the established statistical relationship between climate output and fine-scale weather and avoids inconsistencies between time steps and variables, which would lead to physically implausible behavior.

Our approach aligns with a broader trend [4, 86, 5, 90, 21, 187] of replacing computationally expensive simulations with statistical models that emulate Earth-system dynamics. Rampal et al. [166] give a topical overview and discussion regarding the use of ML methods for statistical downscaling. In particular, generative-modeling techniques, such as normalizing flows [79], GANs [91, 87], and diffusion models [5, 90, 212] have been emerging as a popular model class. Our model extends existing research [for example 187, 126, 5, 90, 3] by enforcing coherence across spatial, temporal, and variable dimensions. In particular, our predictions are sampled from a *joint* distribution, avoiding the disconnection that typically comes with

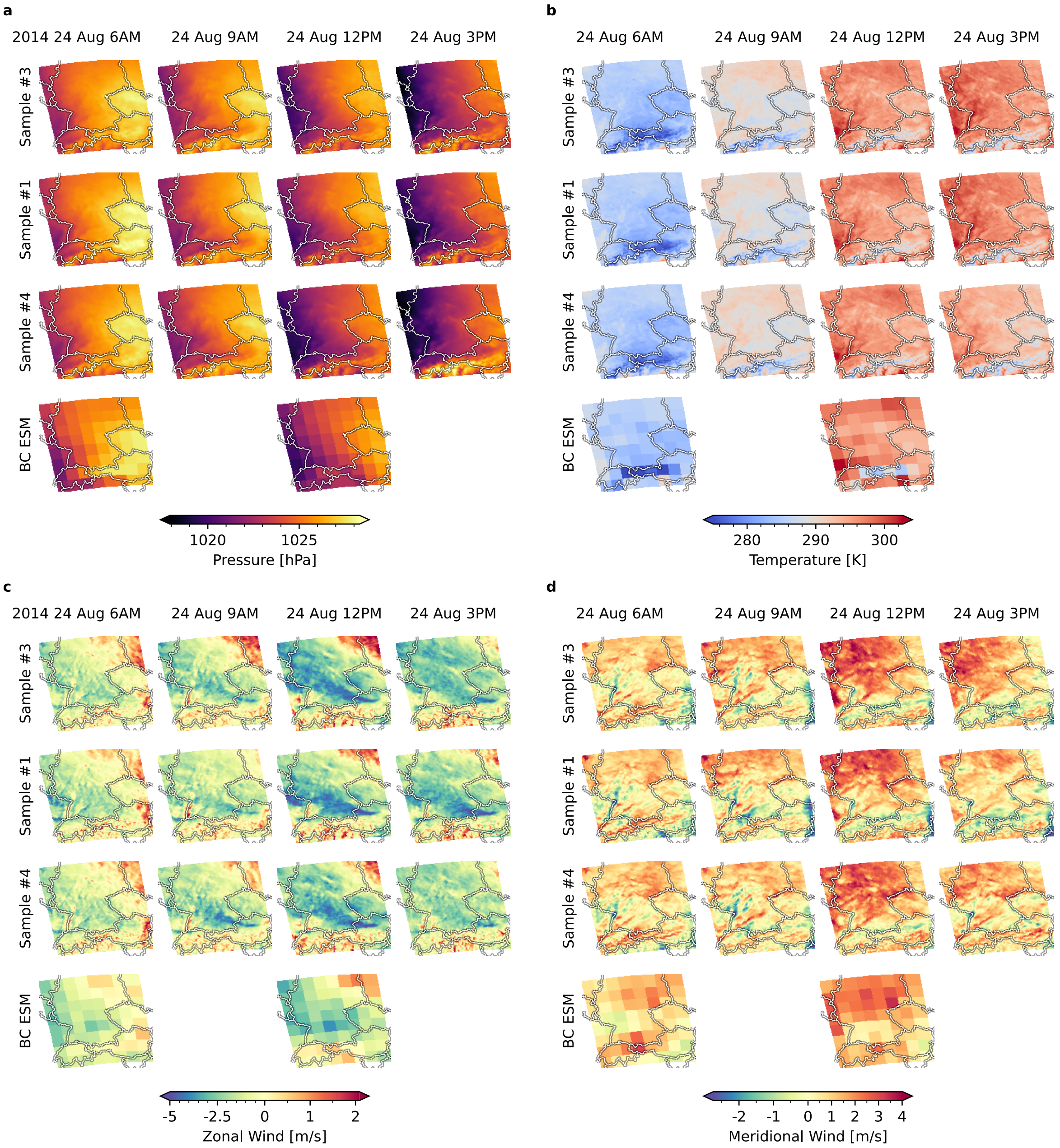


Figure 4.6: **Local climate information through downscaling.** The plot shows spatiotemporally downscaled predictions for the MPI-HR ensemble that is part of the CMIP6 simulations. Time progresses from left to right in three-hourly steps, starting on August 24, 2014, at 06:00 AM and ending the same day at 03:00 PM. The four variables, mean sea-level pressure (**a**), surface temperature (**b**), zonal (**c**) and meridional (**d**) wind-speed components, are downscaled jointly by the model. The top three rows show the progression of three randomly selected samples. The bottom row shows the corresponding conditioning information from the coarse bias-corrected (BC) ESM simulations. Where the observation is blank, the model interpolates in time, without any conditioning information.

sampling sequences of temporally independent states for each individual atmospheric variable. Inconsistencies between downscaled time steps and variables make predicted time series as a whole physically implausible and render the predictions unsuitable for downstream applications that require coherent estimates. We present our framework as one possible technique for such coherence-dependent applications, which could include, e.g., driving dynamical climate impact models and studies of compound events [229], which require inter-variable coherence. A recent related method by Srivastava et al. [200] treats the spatial and temporal domain jointly while focusing on a single variable (precipitation). For a chosen, fixed sequence length, their pipeline generates high-resolution estimates from coarse simulations by separating the task into, first, a deterministic statistical downscaling model, followed by a generative model that introduces probabilistic estimates of high-frequency patterns. A related model class are conditional weather generators [223, 166][140, Chapter 13], which require substantial expertise and resources to implement. By training an unconditional generative model, we differ from previous approaches that integrate the conditioning directly into the training process [5, 187, 212]. In contrast to Harder et al. [86], our method enables soft and uncertainty-aware constraints *post-training* through posterior inference. Hess et al. [90] exploit the iterative-denoising aspect of diffusion models and initialize the generative process at an only partially perturbed target state. This allows conditioning the model on large-scale information, which the model enhances by replacing the remaining noise with local patterns. Notably, this notion of "conditioning" lacks a clear probabilistic interpretation and is thus fundamentally different from how we use the term throughout this work. Further, existing work focuses on mapping between different reanalysis data sets [86, 225], whereas we explicitly developed a model that allows mapping different climate scenarios to finely resolved time series that predict local weather patterns. Our trained model can be readily re-used for mapping from different coarse predictors to the target resolution without re-training, simply by adapting the observation model and input data accordingly. Diffusion bridges have been proposed for unpaired downscaling of fluid dynamics, though their application to climate-model downscaling remains to be demonstrated [21].

The intended scope of our work is to demonstrate the score-based data assimilation framework by Rozet et al. [174] as an elegant and flexible technique for probabilistic downscaling and, perspectively, for other inference problems in the context of atmospheric dynamics. More than outperforming existing per-variable and per-time-step downscaling methods, or establishing our downscaling framework as a new, universally favored approach, we aim to provide a framework for downstream applications that

1. require spatiotemporally coherent time series,
2. consider multiple variables jointly,
3. benefit from flexibly encoding additional prior knowledge about the relation between input and output through the post-training conditioning.

The comprehensive feature set offered by our model complicates direct comparisons with existing methods. Nevertheless, we demonstrate that, according to standard statistical distance metrics, our framework maintains competitive performance when evaluated per time step and per variable against benchmark approaches (see Table B.1). Further, spectral density comparisons between predictions and reanalysis data are presented in Section B.2, and a concise assessment of predictive skill for wind-power generation is provided in Section B.4. Figure B.11 demonstrates that the model uses learned relationships between the jointly processed variables. Looking ahead, we anticipate that the expanded capabilities of the presented framework will broaden its applicability to a wider range of use cases requiring coherent, high-resolution climate information. Additionally, the modular structure of our approach (Figure 4.1) enables the straightforward integration of further, potentially complex, domain-specific knowledge via the observation model in future work. For example, established functional relationships between predicted climate variables and external forcings could be incorporated without retraining the generative model. This modularity allows the trained score model to serve as a reusable foundation for subsequent studies. The diffusion model, including trained weights, and the implementations of the methods and experiments are provided at `https://github.com/schmidtjonathan/Climate2Weather`.

The main limitation of the presented framework is the increased computational cost that comes with coherent predictions. While we have extended the original method by Rozet et al. [174] in terms of scalability, enabling it to process substantially longer trajectories, the temporal and spatial extent of the study is still limited by computational demands. Due to the simultaneous processing of the entire state space, memory requirements remain a limiting factor. This study limits itself to a small region (inspired by Langguth et al. [126]) with a diverse range of orography to establish and validate the methodological framework. However, we believe that long-range teleconnections that affect the sub-region under study are, to some extent, reflected in the regional predictions, given that the training data has been drawn from a Europe-wide reanalysis dataset. This argument is visually supported by embedding high-resolution predictions into a larger spatial context of reanalysis data in Figures B.3 to B.6. We conclude that the technique is likely most useful for smaller study regions, for which highly accurate predictions of local dynamics are desired.

In summary, our framework enables joint spatial and temporal downscaling of multiple climate variables, producing coherent, high-resolution scenarios from coarse climate simulations. By decoupling the learning of dynamical patterns from the conditioning on new inputs, the model offers a flexible and efficient tool for inference tasks in meteorology and climate science. This approach facilitates the use of long-term climate projections for local impact studies across multiple timescales.

4.4 Methods

4.4.1 Data

We analyze four interrelated atmospheric variables, namely the zonal (`uas`) and meridional (`vas`) components of near-surface (10 meter) wind speeds, surface (2 meter) air temperature (`tas`), and sea level pressure (`psl`) due to their crucial importance for understanding atmospheric dynamics. They evolve on different spatial and temporal scales, enabling insights into the performance of downscaling methods across different scales.

For training (2006-2013) and evaluation (2014) of the model, we use data from the Consortium for Small-Scale Modeling Reanalysis 6 (COSMO-REA6) data set, a high-resolution reanalysis product for the European domain developed by the German Weather Service (Deutscher Wetterdienst; DWD) [24] with a spatial resolution of approximately 6 km and hourly temporal resolution. It serves as the ground truth observational dataset in the perfect-predictor evaluation [141] of the presented methodological framework. The COSMO-REA6 data contains errors for some variables and lacks some observations for the years prior to 2006. In this work, only data from 2006 onwards was used.

We apply our downscaling model to historical model runs from two established general circulation models (GCMs) that are part of the sixth phase of the Coupled Model Intercomparison Project (CMIP6) [63], namely the higher-resolution Earth system model (ESM) of the Max–Planck Institute (MPI-ESM1.2-HR) [153] and the high-resolution configuration of the third Hadley Centre Global Environment Model in the global coupled configuration 3.1 (HadGEM3-GC3.1-LM) [9]. Both models have an approximately 100 km spatial and 6-hourly temporal resolution.

We restrict our analysis to a spatial subregion that is oriented at the benchmark region proposed by Langguth et al. [126]. The region includes parts of Germany, Switzerland, Austria, and the Czech Republic ($6° \mathrm{E} - 16° \mathrm{E}$, $46° \mathrm{N} - 52° \mathrm{N}$ (see Figure B.1) which results in 128×128 grid points per time point for the reanalysis dataset. We use data for the period 2006-2014, the time range in which the reanalysis and the historical GCM runs overlap. The GCM data is spatially re-gridded to a rotated latitude-longitude grid using bilinear interpolation to match the coordinates and grid type of the reanalysis dataset. As a pre-processing step, we apply a quantile-mapping procedure to the GCM data. This mitigates biases by adjusting the value distribution of the GCM to align better with the distribution of the reanalysis product. For this, we use the quantile-mapping implementation provided by the `python-cmethods` [189] Python package with default parameters.

4.4.2 Model

We approach statistical downscaling as a Bayesian-inference problem [71, 22]: An uncertain estimate for an unknown quantity (here: high-resolution weather) is obtained by first formulating a prior distribution that encodes known, assumed, or learned properties of the unknown quantity. The prior defines the output space of the prediction model. Through an observation model (or "likelihood"), the prior is conditioned on the available information (here: coarsely simulated climate output) to yield the posterior distribution. The posterior ideally captures the uncertainty that is both inherent in the prior and which arises from incomplete information and model mismatch.

For inference in the context of dynamical systems, it is useful to formulate a prior model that somehow represents a mechanistic system that we assume to underlie the unknown dynamics. The core of the presented methodological framework is a score-based diffusion model (DM)

[194, 94, 199, 108, 106], which constitutes the prior model of the framework. DMs are an instance of generative models that involve training a deep neural network on a finite set of data points, which can then be used to generate new samples from the underlying data distribution. The basic formulation of DMs involves two main components. Firstly, a forward *diffusion process* transports the data distribution to a known, tractable distribution, such as a standard normal distribution [94, 199]. In this process, structured data points (like RGB images, weather states, etc.) are successively perturbed until all signals are entirely replaced by noise. The pivotal insight is that a certain class of diffusion processes are known to have a reverse counterpart, providing a generative process that successively refines samples from the noise distribution into structured data. The reversal of the diffusion process requires access to the *score function*, which can be understood as a function that, for any given degree of perturbation during the reverse diffusion process, separates noise from the signal to remove it. An exemplary such de-noising process for sequences of meridional wind speeds is visualized in Figure B.10. In the following, we will give a brief overview of the general framework and the extensions to it necessary to obtain the results presented in this work.

The considered class of diffusion processes are Gauss–Markov processes that are the solution to linear stochastic differential equations of the form [181]

$$\mathrm{dZ}(\tau) = A\mathrm{Z}(\tau)\mathrm{d}\tau + B\mathrm{dW}(\tau), \quad \mathrm{Z}(0) \sim \mathcal{N}(\mathrm{Z}_0, \Sigma_0). \tag{4.2}$$

The state $\mathrm{X} \coloneqq \mathrm{Z}(0)$ at $\tau = 0$ is a data point $\mathrm{Z}_0 \sim \mathcal{D}$ selected from a data set $\mathcal{D}$. The data point is successively perturbed through the diffusion process and thus loses all structure as $\tau \to \mathrm{T}$. Note that τ is sometimes referred to as "time", which does not mean physical time but rather a continuous degree of perturbation of the initial state. Steps in physical time will later be denoted in the subscript, e.g., $\mathrm{Z}_{1:L}(\tau) \coloneqq (\mathrm{Z}_1(\tau), \dots, \mathrm{Z}_L(\tau))$ will denote a time series of length L perturbed according to τ. The drift A and dispersion B define functional properties of the forward process, which is driven by Brownian motion $W(\tau)$. At a final time step T (often $\mathrm{T} = 1$), the process converges to a Gaussian distribution such that $\mathrm{Z}(\mathrm{T}) \sim \mathcal{N}(0, \Pi)$, where the final-step covariance Π depends on the choice of A and B and is often modeled to be the identity matrix. From a result by Anderson [7], the reverse process of Equation (4.2) is known and given as

$$\mathrm{dZ}(\tau) = \left[A\mathrm{Z}(\tau) - BB^{\mathsf{T}} \underbrace{\nabla_{\mathrm{Z}(\tau)} \log p_\tau(\mathrm{Z}(\tau))}_{\text{score function}} \right] \mathrm{d}\tau + B\mathrm{d}\overleftarrow{\mathrm{W}}(\tau), \quad \mathrm{Z}(\mathrm{T}) \sim \mathcal{N}(0, \Pi). \tag{4.3}$$

This process is driven by reverse Brownian motion $\overleftarrow{W}(\tau)$ and depends on the gradient of the log-marginal density, called the *score*, of the diffused state $\mathrm{Z}(\tau)$ at every perturbation-time point τ. We call Equation (4.3) the *generative process* as it allows sampling unseen data points from the data distribution simply by

1. sampling a vector of independent Gaussian noise $\mathrm{Z}(\mathrm{T}) \sim \mathcal{N}(0, \Pi)$

2. numerically simulating Equation (4.3) backwards through time, starting from Z(T) and ending at a generated sample Z(0).

There is much existing and active research regarding the sampling algorithm used in step 2., which, in general, is slower and more complex when comparing diffusion models to other methods from the generative-model class [227, 197, 228]. Hess et al. [90] employ a recent extension to the diffusion-model framework that allows single-step sampling. This work uses a standard technique that solves an ordinary differential equation related to the marginal distribution of Equation (4.3) [196, 199, 106] following the original proposal by Rozet et al. [174].

In practice, the score of $p_\tau(\mathrm{Z}(\tau))$ is not accessible and has to be estimated. It is common practice to train a noise-dependent neural network $s_\theta(\mathrm{Z}(\tau), \tau)$ on a finite set of samples from the data distribution to approximate the true, intractable score function. Most practical approaches optimize a de-noising score-matching objective [219, 94, 196, 199, 108, 106]. In light of this, simulating Equation (4.3) is commonly perceived as iteratively de-noising the initial Gaussian random state Z(T) and successively refining it to a noise-free data point. For spatial data, a time-conditioned U-Net [172, 94] architecture has proven effective in modelling the score function. The U-Net architecture consists of multiple levels, each containing blocks of convolutional layers and skip connections. Along these levels, the spatial dimensionality of the input data is first successively reduced and then increased again. This encoder-decoder structure results in a bottleneck layer that forces the model to extract a limited set of meaningful features from the input. At each level, skip connections between equal-resolution blocks are introduced to facilitate the learning task by maintaining information throughout the encoding- and decoding process. This work uses a U-Net architecture with a total number of around 72 million parameters, including a self-attention mechanism at the 8×8-bottleneck layer [94, 44].

The framework used in the presented experiments leverages the work by Rozet et al. [174] that extends the basic DM framework, enabling robust sampling from a posterior distribution and generating arbitrary-length sequences of spatiotemporally coherent state trajectories. The idea behind score-based data assimilation (SDA) is to separate an inference task into two parts:

I. learning a generative model that generates spatiotemporal patterns from the target output space and

II. conditioning the prior model using a functional or statistical constraint (the observation model) that defines the specific inference task.

This separation aligns with the idea of Bayesian inference, in which a posterior estimate is obtained from combining a prior model (I.) with an observation model (II.) through conditioning. A high-level introduction of both parts follows below. For more details regarding the score-based data assimilation method, an extensive evaluation, and the corresponding code base, we refer to the original publication by Rozet et al. [174].

The generative model (I.) constitutes the prior in the present Bayesian-inference approach to statistical climate downscaling. As described above, the diffusion model requires a trained score model $s_\theta(\mathrm{Z}(\tau), \tau)$ to iteratively refine an initial random-noise sample into a generated data point (cf. Equation (4.3)). In our case, the generative model outputs uninformed sequences of high-resolution weather patterns. To generate sequential data, Rozet et al. [174] propose to learn a de-noising model on short fixed-length time windows. During training, the time window is flattened into the channel-dimension of the convolutional neural network that represents the score model s_θ. This allows the model to learn correlations between different variables at different points in time. In the sampling routine, each point in time is de-noised by computing the score in the context of a surrounding temporal context window of size $w = 2k + 1$. Rozet et al. [174] argue that many dynamical systems are (approximately) Markovian and in order to predict the current state, it is sufficient to regard the Markov blanket of order k around the current time step, instead of the entire trajectory. This yields output dynamics that are temporally coherent without the need for training the network on long sequences, which would not only significantly increase the computational complexity but also fix the desired sequence length L as a pre-determined hyperparameter, greatly limiting flexibility. Through an intricate convolution-like routine, the windows are assembled into trajectories of arbitrary length L. The corresponding pseudo-code is taken from Rozet et al. [174] and given in Algorithm 1.

Algorithm 1 Composing the sequential score function $s_\theta(\mathrm{Z}_{1:L}(\tau), \tau)$ (from Rozet et al. [174])

```
**Input**: sample $\mathrm{Z}_{1:L}(\tau)$ of sequence length $L$, perturbation level $\tau$;
    Markov order $k$; trained score model $s_\theta$.
**Output**: Score $s_{1:L} \approx \nabla_{\mathrm{Z}_{1:L}(\tau)} \log p_\tau(\mathrm{Z}_{1:L}(\tau))$ for every time step $l \in 1{:}L := (1, \ldots, L)$
function $s_\theta(\mathrm{Z}_{1:L}(\tau), \tau)$
    $s_{1:k+1} \leftarrow s_\theta(\mathrm{Z}_{1:2k+1}(\tau), \tau)[:k+1]$
    for $l = k + 2$ to $L - k - 1$ do
        $s_l \leftarrow s_\theta(\mathrm{Z}_{l-k:l+k}(\tau), \tau)[k+1]$
    end for
    $s_{L-k:L} \leftarrow s_\theta(\mathrm{Z}_{L-2k:L}(\tau), \tau)[k+1:]$
    return $s_{1:L}$
end function
```

For our experiments, we require scaling the original SDA framework to allow for the prediction of very long time horizons with thousands of steps. Unfortunately, sampling a spatiotemporally coherent trajectory in the described manner comes at an increased computational cost and, in particular, significant memory demands since the entire state trajectory must be held in memory. To make the approach scale to predicting 1-hourly weather dynamics for multiple years, we implement the spatiotemporal score function Algorithm 1 as a massively parallelized convolution operation through time that efficiently manages the available memory—optionally on multiple devices. This extension of the original implementation has proven indispensable for the problem scale. The corresponding Python implementation is provided at `https://github.com/schmidtjonathan/Climate2Weather`.

The generative model described above can generate random sequences of weather-like patterns. One can think of it as a "physical prior" in that it represents a statistical model for the general spatial and temporal patterns observed in high-resolution weather dynamics. Note that the physical laws are only implicitly represented by this model, through learning it from existing simulations, and no explicit mechanistic laws are encoded in the model. To tackle the downscaling task specifically, we have to inform the prior model about the climate constraints. To this end, we introduce the conditioning mechanism, which allows us to impose the constraints provided by the coarse ESM simulations onto the prior model.

We proceed to describe the conditioning mechanism (II.). While the unconditioned diffusion model generates samples $p(\mathrm{Z}(0))$, the goal is to sample from a posterior $p(\mathrm{Z}(0) \mid \mathrm{Y})$, instead. We denote the high-resolution output with $\mathrm{Z}(0)$ and the coarse input with Y. Here, we include the noise level τ of the sample $\mathrm{Z}(\tau)$ into the notation (in the parentheses), as it will become relevant in this section. Recall that $\mathrm{Z}(0)$ is the generated, noise-free sample. An observation model $p(\mathrm{Y} \mid \mathrm{Z}(0))$ relates the coarse input Y to $\mathrm{Z}(0)$. We consider a Gaussian observation model

$$p(\mathrm{Y} \mid \mathrm{Z}(0)) = \mathcal{N}(\mathrm{Y}; h(\mathrm{Z}(0)), R), \tag{4.4}$$

with an observation operator $h(\mathrm{Z}(0))$ that can be a linear or non-linear function [39, 174] and observation- or sensor noise defined by a positive-definite matrix R. In our case, h selects from the high-resolution state Z every six hours and computes an area averaging in space (see Equation (4.1)). For our experiments, the observation noise R is selected by running a simple random search on a range of plausible values. To accelerate model selection, we chose R based on the model's downscaling performance on a two-day window. The resulting observation model is used in all presented experiments. Sampling from $p(\mathrm{Z}(0) \mid \mathrm{Y})$ amounts to replacing the score function in Equation (4.3) by a posterior score function $\nabla_{\mathrm{Z}(\tau)} \log p_\tau(\mathrm{Z}(\tau) \mid \mathrm{Y})$. The posterior score is obtained through Bayes' rule by noting that

$$\nabla_{\mathrm{Z}(\tau)} \log p(\mathrm{Z}(\tau) \mid \mathrm{Y}) = \nabla_{\mathrm{Z}(\tau)} \log p_\tau(\mathrm{Z}(\tau)) + \nabla_{\mathrm{Z}(\tau)} \log p(\mathrm{Y} \mid \mathrm{Z}(\tau)), \tag{4.5}$$

The normalization constant is not dependent on Z and thus vanishes when taking the gradient. The prior score is modeled with the parametric score model $s_\theta(\mathrm{Z}(\tau), \tau)$. Note that the likelihood term $p(\mathrm{Y} \mid \mathrm{Z}(\tau))$ is defined on the perturbed state $\mathrm{Z}(\tau)$ and has to be approximated to be compatible with the observation model $p(\mathrm{Y} \mid \mathrm{Z}(0))$. For details, we refer the reader to [39, Section 3.1], who establish the concept of "Diffusion Posterior Sampling" and to its extension by Rozet et al. [174] (Section 3.2). Further, a mathematical description is given in Section B.6. To make the conditioning work efficiently at the problem scales considered in our experiments, we introduce an approximation in the conditioning mechanism that avoids the computation of gradients with respect to the score network. This is described and derived in Section B.6.

4.4.3 Experimental Setup

ESM projections are not directly paired to reanalysis data, so we begin by evaluating the generative downscaling model in an on-model setting. We set up an experimental setting that allows us to compare the model output to a ground truth to assess its predictive performance. Firstly, the model is trained on a subset (2006–2013) of the COSMO reanalysis data. A separate subset (2014) serves as a test set. Then, . . .

1. . . . from the test set, we select an evaluation period and generate artificial coarse observations that represent ESM projections. The spatial resolution is reduced by a factor of 16 × 16 and the temporal resolution by a factor of 6 (see Equation (4.1)). Specifically,
    a) every 6th time step (hour) is selected from the ground truth,
    b) a patch-wise spatial averaging operation is applied throughout the spatial region at every time step. We compute the arithmetic mean for each 16 × 16 spatial patch to yield a single spatial observation.
2. With the observations from 1., the model predicts the underlying 1-hourly, high-resolution weather dynamics lost by coarsening the data. Multiple samples are drawn.
3. The samples are compared to the reanalysis data from the same time period and evaluated. Variations between the samples provide structured uncertainty.

As a second step, ESM (CMIP6) simulations replace the artificially coarsened reanalysis data for the final experiment. Instead of artificially spatiotemporally subsampled reanalysis data, we condition the score model on two different ensembles from the CMIP6 data set: the MPI-HR and the HadGEM3 runs. The experimental setup is adopted exactly from the on-model experiments, aside from the ESM conditioning information and the extended considered time horizon of one full year. Concretely, we downscale the ESM simulations for the considered spatial patch from January 1, 06:00 AM until December 31, 06:00 AM (8 736hours) for the year 2014, increasing the spatial resolution from 8 × 8 ($\sim$ 100km) to 128 × 128 ($\sim$ 6km) grid points. For the four high-resolution variables, this corresponds to a total of $4 \times 8\,736 \times (128 \times 128) = 572\,522\,496$ predicted values given conditioning information that is coarser by a factor of $(16 \times 16) \times 6 = 1\,536$. For this specific experiment, we used four NVIDIA A-100 GPUs, each of which generated two predictions in parallel. Generating one 1-year sample of hourly downscaled predictions for the 128 × 128-node region takes around two hours on a single NVIDIA A-100 device.

5 Assessing Regional Future Risks of Renewable-Energy Shortages

The European electricity power grid is transitioning towards renewable energy sources, characterized by an increasing share of off- and onshore wind and solar power. However, the weather dependency of these energy sources poses a challenge to grid stability, with so-called Dunkelflaute events—periods of low wind and solar power generation—being of particular concern due to their potential to cause electricity supply shortages. In this study, we investigate the impact of these events on the German electricity production in the years and decades to come. For this purpose, we adapt a recently developed generative deep learning framework to downscale climate simulations from the CMIP6 ensemble. We first compare their statistics to the historical record taken from ERA5 data. Next, we use these downscaled simulations to assess plausible future occurrences of Dunkelflaute events in Germany under the optimistic low (SSP2-4.5) and high (SSP5-8.5) emission scenarios.

Our analysis indicates that both the frequency and duration of Dunkelflaute events in Germany in the ensemble mean are projected to remain largely unchanged compared to the historical period. This suggests that, under the considered climate scenarios, the associated risk is expected to remain stable throughout the century.

5.1 Introduction

Meeting the European Union's Green Deal targets (net-zero for emissions by 2050; $\geq 55\%$ reduction by 2030) demands a rapid expansion of renewable energy sources, especially wind and solar [55]. Renewables already supplied 44.7% of EU electricity in 2023 and 62.7% in Germany, where wind and solar accounted for 31.5% and 13.8%, respectively [56, 40, 54]. This trajectory is expected to continue, with the share of renewables projected to reach up to 80% by 2030 [148]. As weather-dependent sources grow, the power grid becomes increasingly sensitive to meteorological variability. In particular, simultaneous lulls in both wind and solar output pose a serious challenge. These so-called "Dunkelflaute" (or "dark doldrums") events, typically occur in winter under prolonged high-pressure, cloudy, low-wind conditions. During such periods, most generation of renewable energy can get cut off at once. Germany experienced pronounced examples in November and December 2024: a week-long event in November cut renewables to 30% of supply and drove baseload prices to 145 €/MWh (compared to ∼40 €/MWh on average), while a December episode briefly pushed prices past 175 €/MWh [208]. These episodes demonstrate that Dunkelflaute events already cause acute supply shortfalls and extreme price volatility, underscoring the need to assess their future risk under climate change. Prior work has mainly characterized Dunkelflaute events under historical climate but not projected them into the future. For example, meteorological studies of Germany show that Dunkelflaute events occur mostly in winter, when solar irradiance is naturally low and winds—normally strong in this season—become stagnant. Dunkelflauten typically last a few days and recur several times per winter [149]. Similarly, Kittel et al. [112] examined historical events across Europe using aggregated time series per country. There is a growing number of studies highlighting how wind and solar lulls combine [111, 19], but they focus on past data and fixed portfolio effects, and thus do not necessarily allow conclusions about future scenarios under climate change.

To assess future Dunkelflaute risk, climate projections with high spatial and temporal resolution are needed to take into account local weather patterns and short-term variability. Future climate projections from output of global climate model (GCM) simulations are provided by the Coupled Model Intercomparison Project Phase 6 (CMIP6) extending to the year 2100 [63]. However, their native resolution, typically on the order of 100 km, remains too coarse for detailed, local impact assessments. Downscaling therefore refines GCM output either dynamically or statistically.

Dynamical downscaling uses high-resolution regional climate models (RCMs) driven by GCM output to simulate local climate processes based on an explicit physical model [73]. Databases, such as EURO-CORDEX, provide these high-resolution climate projections (12-50 km) for Europe [117]. There are first studies that applied such dynamical downscaling approaches to CMIP6 data to investigate future Dunkelflaute events [49]. Data from the EURO-CORDEX database is used to derive energy relevant variables by the Copernicus Pan-European Climate Database (PECD v4.2) [41], which provides wind and solar energy generation time series for past and future scenarios. However, different GCM-RCM pairings can yield divergent local trends even under the same scenario [113], particularly in complex terrain and coastal regions [150]. Moreover, the PECD remains constrained to selected GCM-RCM pairings, is largely deterministic, and often provides only monthly or aggregated fields rather than the sub-daily spatiotemporal data.

Statistical downscaling, on the other hand, establishes empirical relationships between large-scale predictors and local weather. While statistical methods are typically beneficial in terms of computational efficiency, they often fail to capture the complex spatiotemporal dependencies of multivariate weather fields [140, 217]. To address this, we adapt the generative deep-learning framework from Chapter 4, trained on high-resolution ERA5 reanalysis data [89], to learn joint local weather statistics and to conditionally map coarse CMIP6 fields to spatiotemporally consistent, turbine-relevant realizations. Crucially, the model's probabilistic sampling produces many plausible high-resolution scenarios, enabling uncertainty-aware assessment of Dunkelflaute occurrence, duration, and regional variability across the CMIP6 ensemble.

Table 5.1: Variable names in ERA5 and CMIP6.

Variable Name	**long**	**short**
10m zonal wind	`10m_u_component_of_wind`	`uas`
10m meridional wind	`10m_v_component_of_wind`	`vas`
Surface air temperature	`2m_temperature`	`tas`
Surface solar radiation downwards	`surface_solar_radiation_downwards`	`rsds`

5.2 Data and methods

5.2.1 Data

We use the ERA5 reanalysis dataset [89] from the Copernicus Climate Change Service (C3S) for the years 1979 to 2024 with a spatial resolution of 0.25° and a six-hourly temporal resolution. Six-hourly resolution was shown to be a suitable tradeoff against computational cost [53]. The variables used to compute the wind and solar power generation are given in Table 5.1. For future projections, we use a selection of the Coupled Model Intercomparison Project Phase 6 (CMIP6) dataset [63]: MPI-ESM1-2-HR, GFDL-ESM4, IPSL-CM6A-LR, MIROC6 and CANESM5. The selection is based on their spread in different climate sensitivities [147]. We look at two specific emission scenarios, the Shared Socioeconomic Pathways (SSPs) SSP2-4.5 (referred to as ssp245) and SSP5-8.5 (ssp585), which represent an optimistic low and a high emission scenario, respectively. The CMIP6 data is downloaded using the Pangeo catalogue[1]. A quantile mapping bias correction is applied to the CMIP6 data using coarsened ERA5 reanalysis as reference.

5.2.2 Estimation of Dunkelflaute events

Dunkelflaute events are generally characterized as periods when the combined wind and solar power generation falls below a specified threshold of installed capacity for an extended period [111]. For event identification, we adopt the methodology described by Mockert et al. [149], which relies on analyzing so-called *capacity factor* time series.

CAPACITY FACTOR TIME SERIES

The capacity factor (CF) is defined as the ratio

$$\mathrm{CF} = \frac{P_{\mathrm{generated}}}{P_{\mathrm{max}}} \tag{5.1}$$

of actual power output $P_{\mathrm{generated}}$ to the theoretical maximum output P_{max} over a given period. Here, P_{max} is determined by the installed capacity and the available wind speed or solar irradiance at each time step. A CF of 1 indicates operation at full rated capacity for the entire period, while CF < 1 reflects periods of reduced generation due to meteorological conditions, outages or maintenance. Pointwise and spatially averaged CFs are calculated using a modified

---

[1] `https://pangeo-data.github.io/pangeo-cmip6-cloud/pangeo_catalog.html`

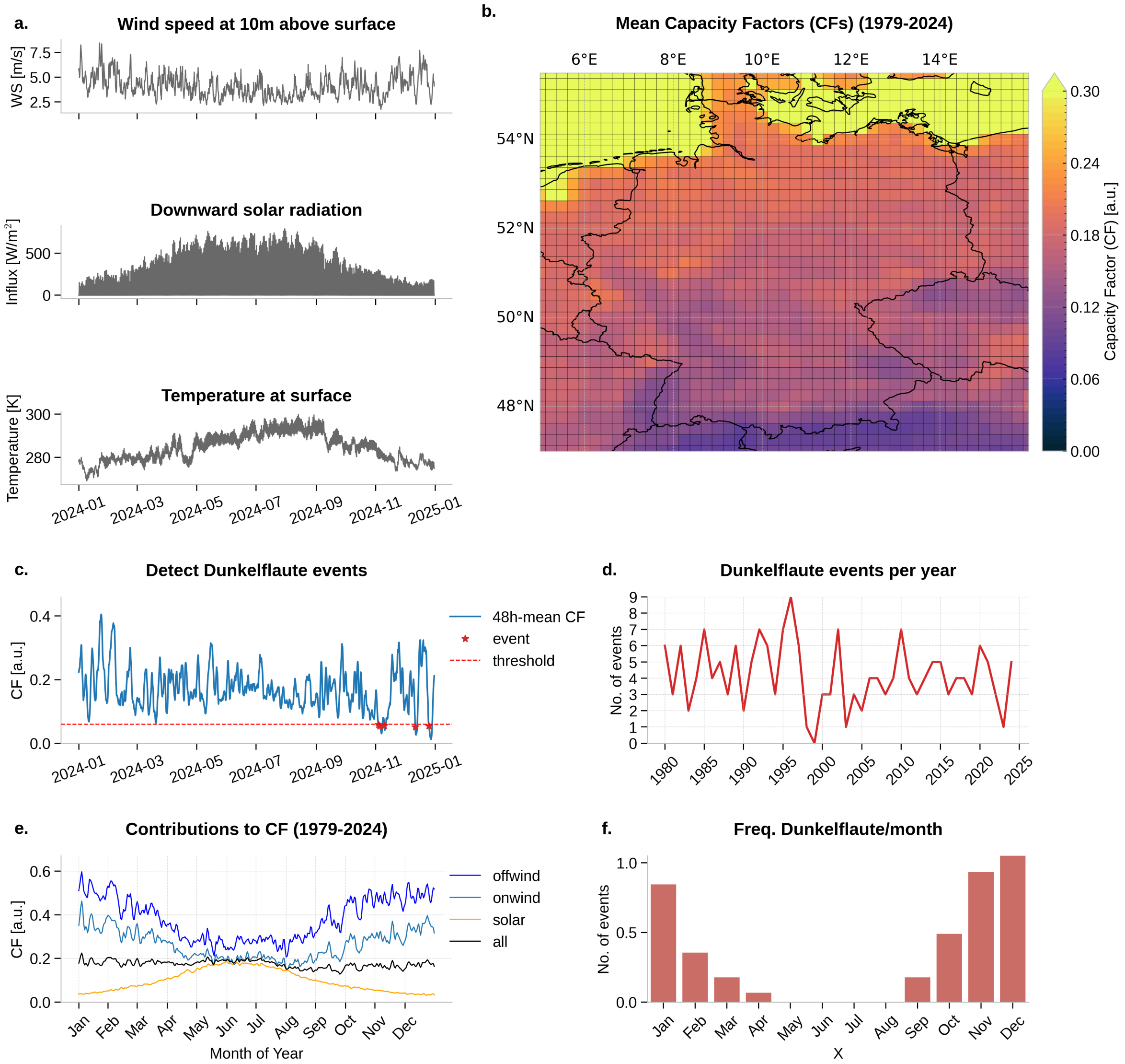


Figure 5.1: **Overview of the identification process for Dunkelflaute events in Germany.** Panel **a.** displays the time series used to calculate local capacity factors (CFs) for wind and solar energy in 2024, representative of the approach applied to the full reanalysis dataset. Panel **b.** presents the resulting local CFs for wind and solar energy across Germany, where the different sources (onshore, offshore, PV) are weighted by their relative fraction in 2024. The CF is a number between 0 and 1; [a.u.] stands for "arbitrary units". Panel **c.** highlights periods classified as Dunkelflaute events, marked by low wind and solar generation; the red line denotes the threshold of 0.06 used for event detection [149]. Panel **d.** illustrates the amount of significant low renewable generation periods identified in the ERA5 historical record. Panel **e.** shows the individual and weighted average CFs for onshore, offshore wind and solar energy, considering only grid cells within German borders (with partial cells weighted by their area within Germany, see Figure C.3). Panel **f.** summarizes the monthly frequency of Dunkelflaute events averaged over the period 1979–2024.

version of the `atlite` Python package [97], requiring only the variables listed above (Figure 5.1 **a.**, Table 5.1). Since, in this work the CF is computed from reanalysis or model data, it only reflects meteorological variability, excluding effects from maintenance or outages. For example, CF = 0.3 corresponds to an average output of 30% of capacity, not constant operation at that level. Thus, the CF serves as a robust metric for characterizing renewable intermittency. Local CF time series for onshore wind, offshore wind, and solar are computed per grid cell. The total renewable CF is then calculated as a weighted average of these, using current installed capacity shares, and this approach is applied consistently for both historical and future periods. For Germany in 2024, the installed capacities are: 57.7% solar (99.3 GW), 36.9% onshore wind (63.5 GW), and 5.4% offshore wind (9.2 GW) [28]. The local CFs are then averaged over Germany, considering only grid cells within German borders (weighted by their area within Germany, compare Figure C.3). From the `atlite` package we adopt the assumption that the spatial allocation of installed wind and solar capacities is proportional to the respective local capacity factors of each technology.

Dunkelflaute event detection

Following Mockert et al. [149], we define a Dunkelflaute event as a period during which the combined CF remains below a threshold of 0.06, averaged over 48 hours. This is shown exemplarily in Figure 5.1 **c.** for the year 2024 with the two events in November and December 2024 (red stars) mentioned above. Consecutive periods are merged into a single event, placed on the first date of occurrence. This procedure enables systematic identification of events/year (Figure 5.1 **d.**). The peak in the year 1997 aligns with previous studies [149, 112]. The same methodology is subsequently applied to the downscaled CMIP6 data in Section 5.2.3.

Dunkelflaute event statistics

The individual CFs for wind and solar energy have a clear seasonal cycle. While off- and onshore winds are, on average, higher in winter and lower in summer, the opposite holds for solar power (Figure 5.1 **e.**). Although this suggests that the average capacity remains approximately constant over the year (black line in Figure 5.1 **e.**), the majority of Dunkelflaute events occur in winter from November to January (Figure 5.1 **f.**). This is likely due to solar CFs not being sufficient to compensate for days with low wind speeds in winter, when daylight hours—and, hence, solar irradiance—is substantially lower. Consequently, while the average wind CFs may be higher in winter, the combined distribution of wind and solar CFs exhibits greater variability and lower tails in winter [129, 149].

5.2.3 Generative downscaling of CMIP6 simulations

This chapter builds directly on the method for probabilistic spatiotemporal downscaling proposed in Chapter 4. We will abbreviate the downscaling framework as GSD (Generative Spatiotemporal Downscaling) in the following.

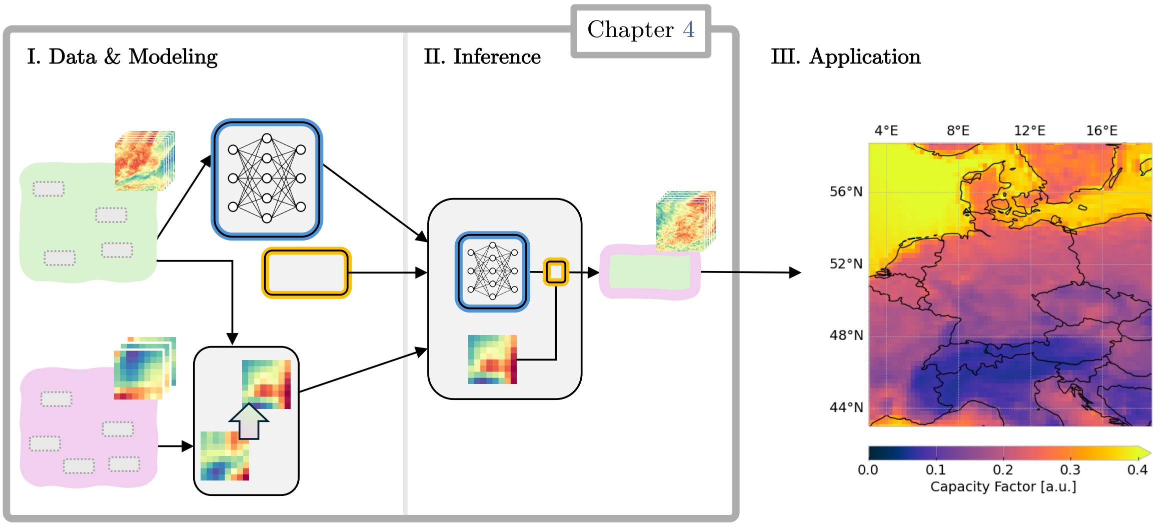


Figure 5.2: **Probabilistic pipeline for spatiotemporal downscaling of multiple variables.** This schematic illustrates the generative spatiotemporal downscaling framework (**I.** and **II.**) adapted from Chapter 4 and Schmidt et al. [186]. For clarity, only one representative variable is depicted. The generative model generates fine-scale downscaled time series from coarse climate simulations. These are used to calculate capacity factors for wind and solar energy (**III.**) on a fine spatial grid for regional assessment of the risk of Dunkelflaute events.

Panels **I.** and **II.** in Figure 5.2 show a sketch of this framework. GSD is built around a generative diffusion model (DM) [194, 198, 94, 196, 199, 108, 106]. Based on a finite data set, which is assumed to consist of independent draws from a shared probability distribution, DMs are trained to generate new, unseen data points by learning an implicit representation of the unknown data distribution and sampling from it. In GSD, a DM is trained on sequences of high-resolution reanalysis data (here: ERA5) and acts as a probabilistic model for the output space. During inference, i.e. *after* training, a conditioning mechanism allows for guiding the generated predictions to adhere to flexible boundary conditions provided by coarse climate simulations. The probabilistic formulation ensures that uncertainty associated with downscaling is reflected. Schmidt et al. [186] evaluate the model in an on-model setting (i.e. with available ground truth data) and, in a next step, apply it to downscaling output from different members of CMIP6. This work follows a similar approach.

We introduce a few modifications to GSD that preserve the core functionality of the original framework. DMs are special instance of *flow matching* (FM) models [130, 131]. The generative process in the DM formulation follows a stochastic diffusion path guided by the score function to reach regions of high data density. In contrast, the FM formulation allows for the construction of more general probability paths. In particular, linear probability paths have been shown to result in more stable training and sampling [130]. Linear probability paths are used for the generative model in this work.

GDS relies on training a fully unconditional generative model such that all guiding constraints are imposed *post-training*. While this allows a high degree of flexibility for downstream tasks, we found it advantageous to provide some conditioning information directly during the training. Specifically, we train the generative model on pairs of (i) sequences of reanalysis data and (ii) the corresponding time stamps. The time stamps only include the hour (here: 0:00, 06:00, 12:00, 18:00) and the month (1 through 12). The reasoning is that the hour of the day and the month capture information about the diurnal and the seasonal cycle. We train the conditional model by incorporating a learned embedding of the time stamps into the generative model [44]. This way, the model learns about periodic phenomena, directly, during training. Moreover, the hour-of-day information allows for generating temporally-aligned predictions even if the conditioning information aggregates samples longer than 24 hours.

After sampling, the output can be used for an application (Figure 5.2, **III.**). Here, we first perform a bias correction on the downscaled data using ERA5 as the reference to compensate for systematic biases introduced by temporal averaging in the input wind data (compare Figure C.1), which can underestimate sub-daily variability due to daily averaging [53] (see Figure C.1 **a.**, **b.**). Further details are provided in Section C.1. Next we use the downscaled data to estimate spatiotemporal wind- and solar-energy generation.

5.3 Results

5.3.1 Evaluating CF estimation from generative downscaling

Estimating local CFs from coarsened ERA5

We begin by evaluating the downscaling framework in an on-model setting, similar to the procedure in Chapter 4, conditioning the generative model on coarsened ERA5 data. Thereby, we assess the downscaling framework's ability to accurately reproduce the fine-scale spatial and temporal variability of wind (Figure 5.3, first column) and solar (Figure 5.3, second column) energy capacity factors in Germany. Further, we evaluate the model's performance during the critical winter months of November through January (Figure 5.3, third column), which are most relevant for Dunkelflaute events.

In this artificial setting, we find that (i) spatial downscaling is essential to reproduce spatial variability of wind energy (compare Figure 5.3 **d.** vs. Figure 5.3 **j.**,**m.**) and (ii) temporal downscaling is essential for resolving correct solar capacity factors (Figure 5.3 **h.**) due to the diurnal cycle. Without these steps, the coarse input data fails to capture the spatial heterogeneity for the critical winter months (Figure 5.3 **f.**) or is even unable to obtain a reasonable number of events (Figure 5.3 **i.**). We also see that the bias correction step is essential to avoid an underestimation of the CFs for specific regions (e.g. compare North-East Germany in Figure 5.3 **l.** vs. Figure 5.3 **c.** and Figure 5.3 **o.**). The importance of the bias corrected downscaling can also be shown exemplarily for a selected time period of the aforementioned Dunkelflaute events in November 2024 (Figure C.5). While the general trend is captured also by a coarser spatial and temporal resolution, the bias corrected downscaling is closest to the ground truth. The other methods tend to underestimate the CF and are therefore likely to overestimate the number of Dunkelflaute events.

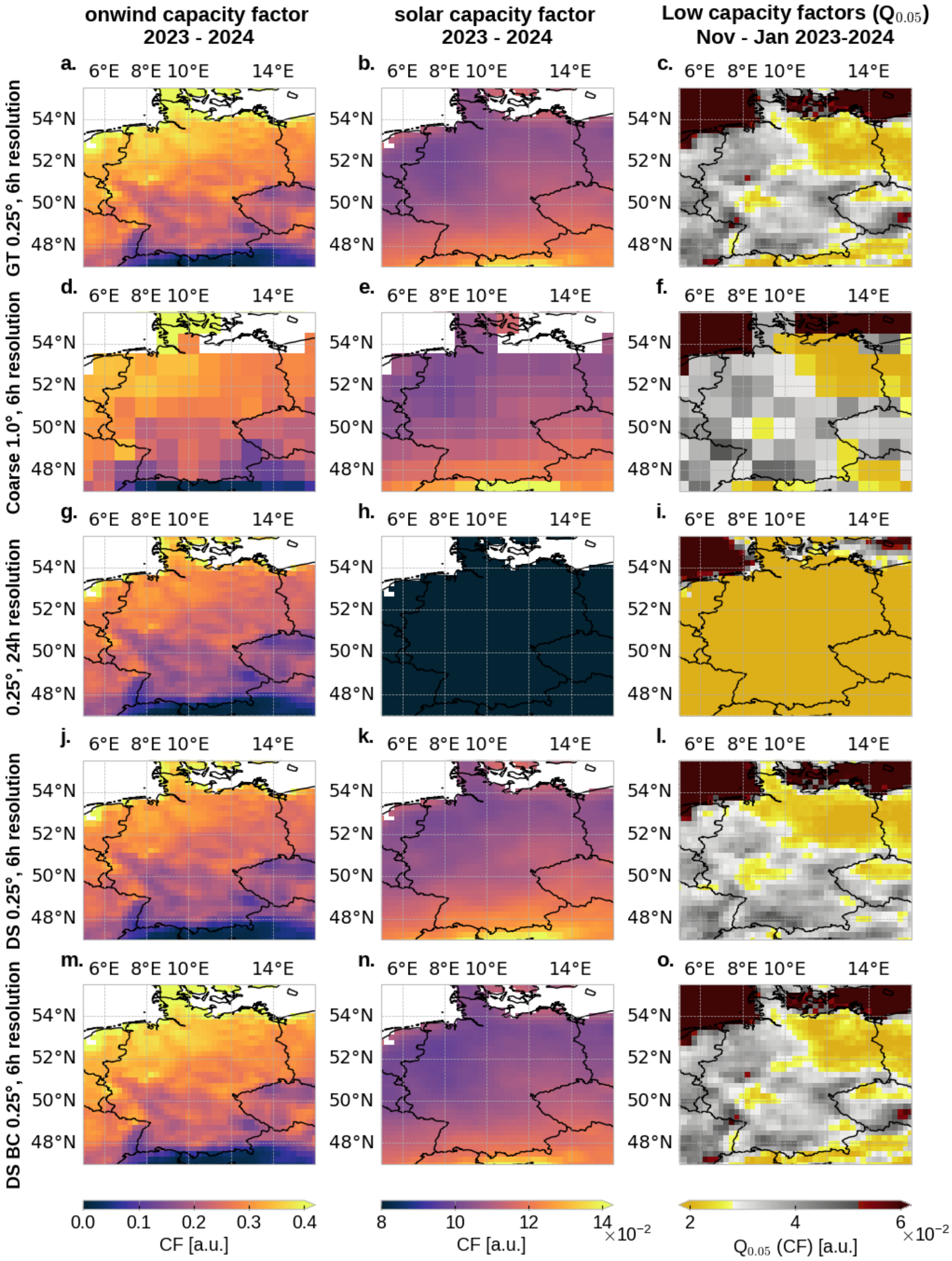


Figure 5.3: **Comparison of different spatial and temporal resolutions and the effect of bias correction.** The first column presents onshore wind energy CFs, the second column shows solar energy CFs, and the third column depicts the 0.05 quantile of the 48-hour moving average of CFs for November–January. The rows show (from *top* to *bottom*): CFs from ground truth (GT) ERA5 reanalysis data; a spatially coarse (1°) and a temporally coarse (24 h) resolution of the GT respectively; the uncorrected downscaling (DS) output; the bias-corrected downscaling (DS BC) results. The downscaling outputs are averaged over 10 samples drawn from the generative model.

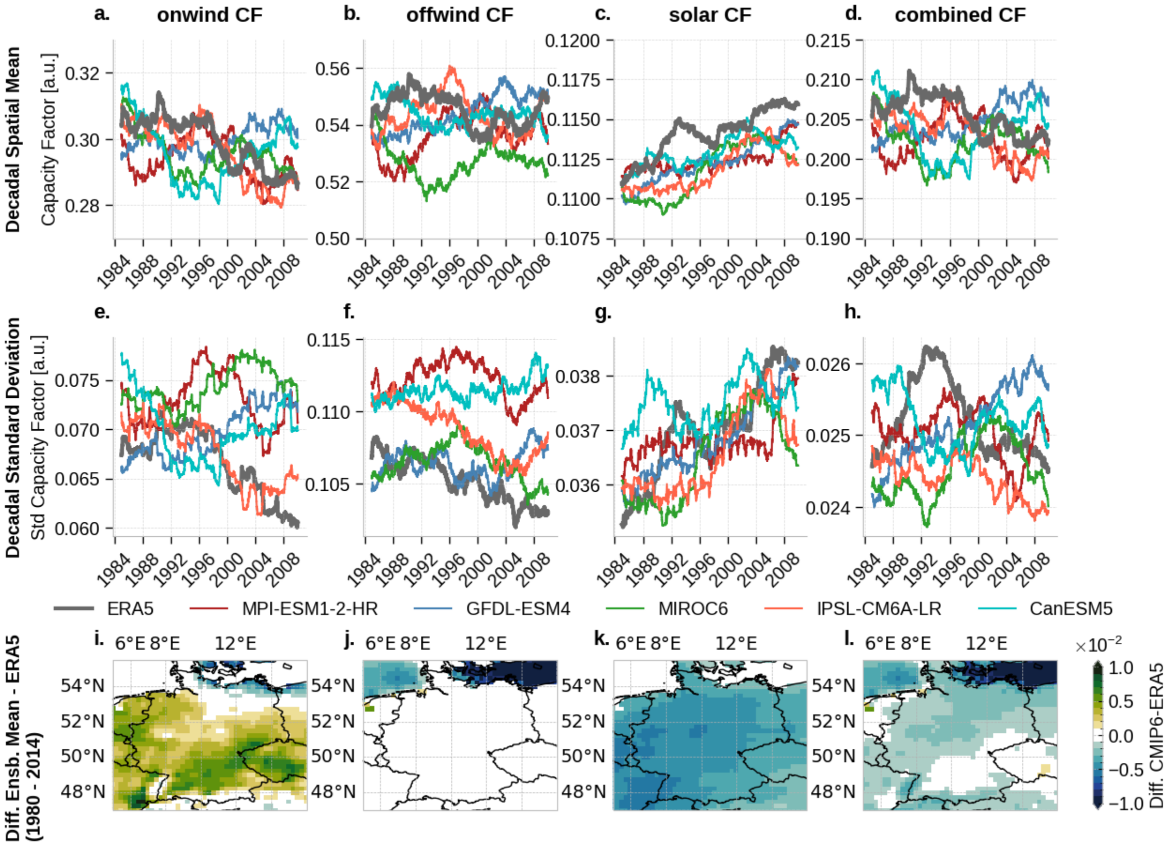


Figure 5.4: **Comparison of historical CFs obtained from CMIP6 to ERA5.** For the period 1980–2014, this figure presents a comparison between the ten-year moving average CFs calculated from the bias-corrected and downscaled ensemble mean of historical CMIP6 runs and those obtained from ERA5 reanalysis data. In the first row, the spatially weighted CF time series for Germany are shown, with individual CMIP6 models represented by colored lines and ERA5 reanalysis by grey lines. This provides a direct temporal comparison across models and the observational reference. The second row shows the respective decadal standard deviation per time point. The third row displays, for the considered spatial region, temporally averaged differences between the downscaled CMIP6 ensemble mean CFs and ERA5 CFs. Specifically, the first column corresponds to onshore wind energy, the second to offshore wind (restricted to grid cells with at least 50% water coverage), and the third to solar energy (restricted to land grid cells). The fourth column shows the weighted-mean CFs, in which individual sources are combined according to their current relative share in the German energy mix in 2024. The evaluation period covers January 1, 1980, to December 31, 2014.

Estimating local CFs from historical CMIP6

We compare the spatiotemporally averaged CFs for wind and solar energy in Germany. The CFs are computed on the downscaled historical CMIP6 runs and on ERA5 in the time range 1980–2014. The comparison is visualized in Figure 5.4. A direct, pairwise comparison between ERA5 and historical CMIP6 model runs is difficult, since ERA5 is an observation-constrained reanalysis that reconstructs the actual historical weather, while CMIP6 historical runs are free-running climate model simulations. Therefore, we use a ten-year moving average to smooth the time series and assess the estimation of temporal trends and variability. The results show that the time series from ERA5 and downscaled CMIP6 show comparable decadal trends and variability. The ten-year average of onwind CFs remains stable around the period from 1985–1998. In the period 1998–2010 it decreases by 2 percentage points (Figure 5.4 **a.**). Further, the decadal standard deviation is decreasing (Figure 5.4 **e.**). For the offshore wind CFs, the decadal variability is also decreasing (Figure 5.4 **f.**), although the CFs themselves have no clear trend (Figure 5.4 **b.**). Contrary to the onwind CFs, the solar CFs are increasing by 0.5 percentage points throughout the thirty-year period (Figure 5.4 **c.**) with an increasing variability. Even though solar CFs have the highest weighting in the combined CFs, the effect of the decrease in onshore wind CFs dominates the combined CFs. Over the thirty-year period, a decrease from 1998–2010 by 0.5 percentage points is visible. The downscaled historical CMIP6 runs reproduce decadal trends in solar CFs and show decadal spatial means for onshore and offshore wind CFs that remain within the ERA5 range. The decadal standard deviations of wind (Figure 5.4 **e.**, **f.**), solar (Figure 5.4 **g.**), and combined (Figure 5.4 **h.**) CFs are also consistent with those derived from ERA5.

A similar pattern is observed on the level of individual spatial grid cells. For onshore winds, the deviations per grid node (most nodes show only a very slight overestimation of less than 1%) are marginal for the German mainland (Figure 5.6 **i.**). The coastal region and the Baltic Sea present an exception from this with a more pronounced deviation between the downscaled and ERA5 CFs (Figure 5.6 **j.**). These biases are likely inherited from the coarse CMIP6 runs, which are known to be prone to inaccuracies at the land-sea transition [34, 6]. We find a slight underestimation for solar energy (Figure 5.6 **k.**) that is almost uniformly distributed over Germany, indicating a consistent climate model bias. Altogether, the results indicate a good agreement for the weighted mean of the different renewable energy sources according to their relative share in the German renewable energy mix (Figure 5.6 **l.**).

5.3.2 Historical and future projections of Dunkelflaute events

Evaluation on historical period

We first evaluate the estimated record of Dunkelflaute events in Germany during the historical period 1980–2014. To this end, we use the CF time series calculated on downscaled CMIP6 runs. The results are shown in panel **a.** of Figure 5.5. The number of events per year is shown as a moving average over a ten-year window to estimate decadal trends. In terms of both variability and number of events per year, the prediction of Dunkelflaute events based on ERA5 and downscaled CMIP6 runs are similar. None of the climate models exhibits a statistically significant trend[2], neither does ERA5. Starting around the year 2000, the ensemble mean closely aligns with the ERA5 data. Each individual CMIP6 model run yields different predictions. CANESM5 predicts a consistent decrease in the number of events and consistently predicts one event less than ERA5. MPI-ESM1-2-HR shows an increase over the historical record. GFDL-ESM4 and MIROC6 show some decadal fluctuations, while IPSL-CM6A-LR remains stable. The historical ERA5 record lies within the decadal variability

---

[2] using a Student's t-test with a 0.05 significance level

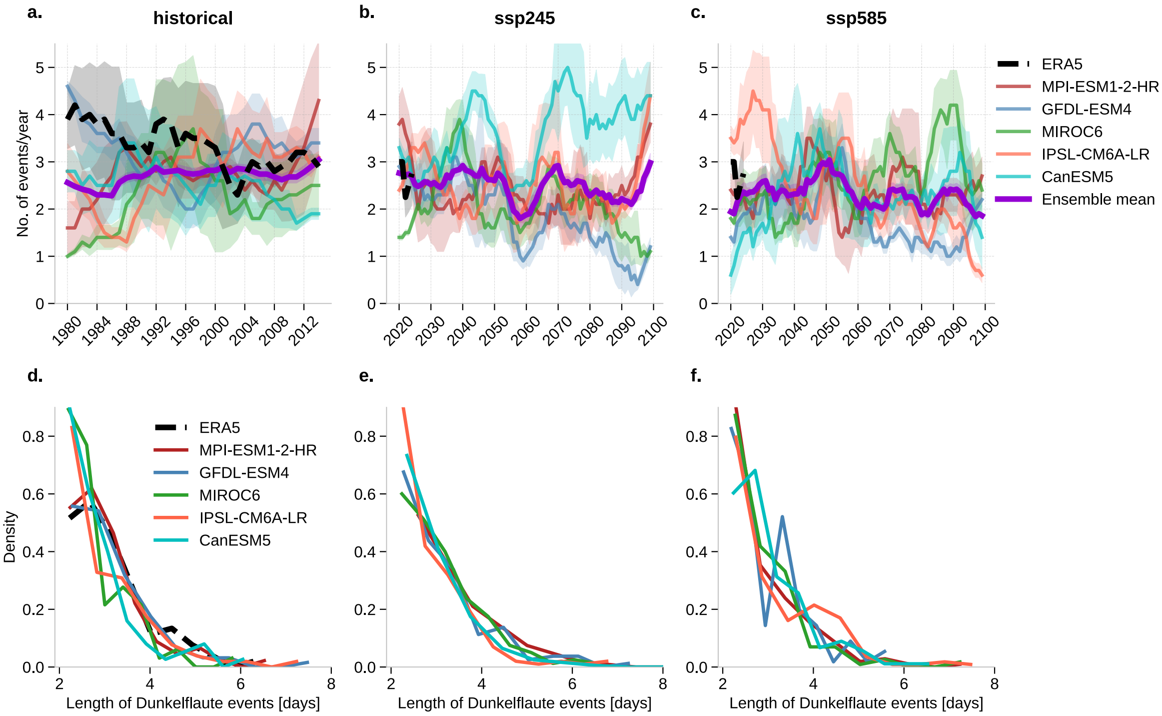


Figure 5.5: **Yearly Dunkelflaute events in Germany for historical and future scenarios.** The first row visualizes the estimated number of Dunkelflaute events over time. Panel (**a.**) visualizes the historical record and compares the climate-model runs to ERA5. The time until the year 2100 is compared considering both (**b.**) the optimistic-case (ssp245) and (**c.**) the worst-case (ssp585) emission scenarios. The number of events per year is the moving average over a 10-year window to estimate decadal trends. The second row shows the distribution of the duration of all measured events in the respective time period for historical record (**d.**) and future scenarios (**e.**, **f.**). The shaded areas correspond to the decadal standard deviation.

of the model ensemble. This indicates that the simulations provide plausible scenarios for past Dunkelflaute events. Next, we look at the distribution of the duration (in days) per event in the historical record (Figure 5.5 **d.**). We find that the downscaled CMIP6 runs show a similar distribution of event durations when compared to ERA5. Only the IPSL-CM6A-LR model slightly overestimates the durations. We argue that our approach provides a reasonable estimate of the frequency of event occurrences.

INVESTIGATION OF FUTURE PROJECTIONS

Next, we analyze projections for the years 2020–2100. Under the low-emission scenario (Figure 5.5 **b.**), most CMIP6 models start to diverge after the year 2080. The CANESM5 model is an exception in that it diverges from the other considered models at a much earlier point already. Towards the end of the projection period (year 2100), for the MPI-ESM1-2-HR and IPSL-CM6A-LR models, the number of predicted Dunkelflaute events increase significantly. In contrast, GFDL-ESM4 and MIROC6 predict a decreasing number of events. In the high-emission scenario (Figure 5.5 **c.**), the trend of the ensemble mean remains stable with no

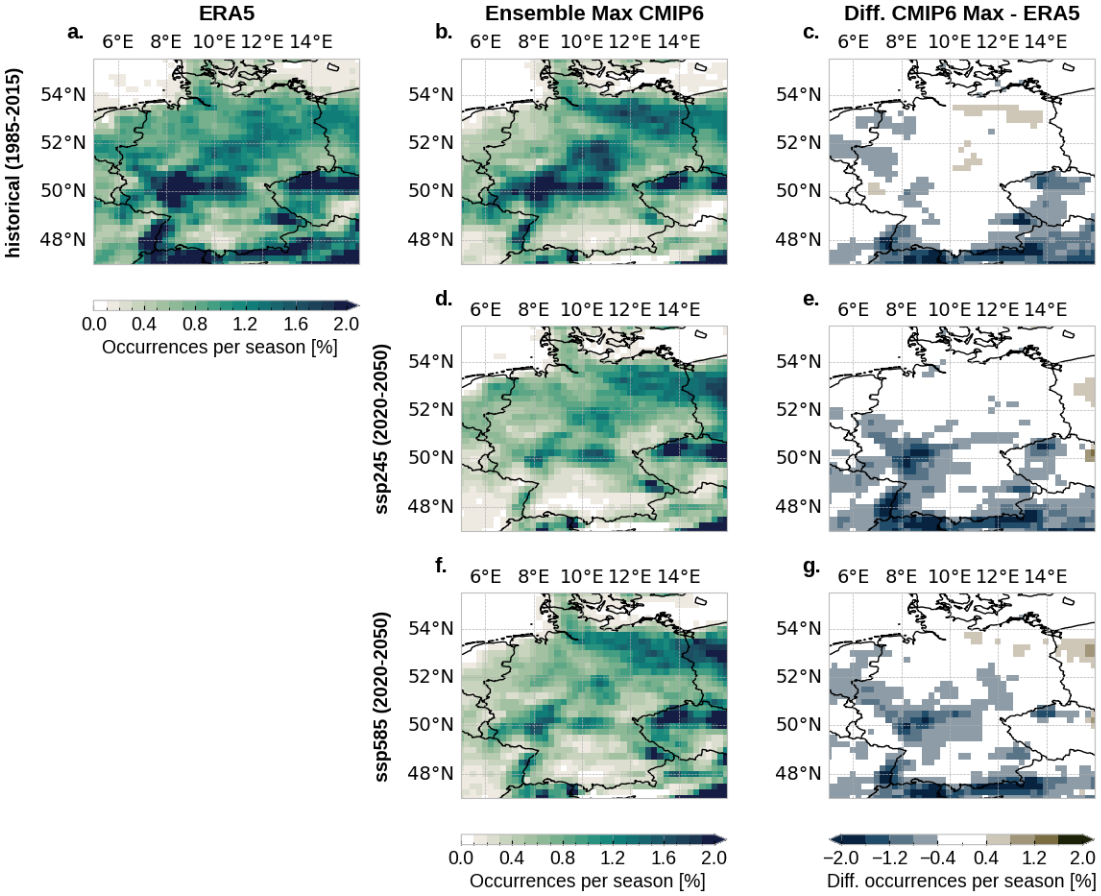


Figure 5.6: **Local risk of prolonged low CFs for different emission scenarios.** “Prolonged” means periods of at least 48 hours; CFs are considered as “low” for values below 6%. The figure shows the spatial distribution of low CFs per grid cell for Germany and surrounding regions. The first row presents the historical reference for ERA5 (**a.**) and the downscaled CMIP6 ensemble mean for 1985–2014 (**b.**, **c.**). The second and third row display projected occurrences for 2020–2050 under the ssp245 (**d.**, **e.**) and ssp585 (**f.**, **g.**) scenarios, respectively. The first column shows ERA5 for reference, the second column the CMIP6 ensemble maximum, and the third row the difference to the ERA5 reference period.

significant changes to around two to three events per year. The distribution of duration per event (Figure 5.5 **e.**, **f.**) remains largely unchanged compared to the historical record.

Local analysis of prolonged periods of low renewable-energy generation

While the ensemble mean of CMIP6 suggests no significant change in the overall number of Dunkelflaute events for Germany, notable local variations in the frequency of low capacity factor days may still occur. This spatial analysis is especially important for strategic planning of wind and solar installations. Fine-scale spatial estimates are needed for identifying regions, in which the risk of extended low renewable-energy generation may shift over the typical operational lifetime of assets (20-30 years). Further, local estimates highlight areas, for which a lower or higher total energy return can be expected under projected future climate conditions.

Estimating these future regional variations, taking the associated uncertainties into account, is crucial for informed investment decisions.

In Figure 5.6, we assess the occurrences of prolonged periods (≥ 48 hours) of low combined capacity factors ($<6\%$) for the near future (2020–2050) per spatial grid cell. The historical reference from ERA5 (Figure 5.6 **a.**) shows a heterogeneous spatial distribution across Germany, with a general concentration from South-West to North-East and over uplands and mountains. The CMIP6 ensemble mean for the historical period (Figure 5.6 **b.**) broadly replicates this pattern. Thereby, mountain regions such as the Alps (Figure 5.6 **c.**) pose an increased challenge for estimation. In particular, wind speeds tend to be biased towards lower values in these mountainous regions [150, 190, 123], resulting in more periods of low output. However, due to topography, wind turbines are rarely installed in such regions. For the future scenarios, analysis of the ensemble mean indicates that the spatial distribution of event frequency per grid cell remains largely unchanged (Figure 5.6 **d.**, **f.**) compared to the historical reference. This indicates no substantial alteration in local patterns under future projections. We find a consistent decrease in event frequency in the Hessian region (Figure 5.6 **e.**, **g.**) and a slight decrease in the North-East, near the Polish border. Importantly, the differences in event frequency among the ensemble mean, maximum, and minimum are comparable to the inter-model variability (Figure C.6 **j.**, **k.**, **l.**). These results indicate that the overall frequency of events in Germany is projected to remain similar in most regions, although localized changes may occur.

5.4 Discussion and Conclusion

The present study adapts a recently developed generative downscaling framework [186] to assess local changes in the risk of Dunkelflaute events projected by the CMIP6 ensemble. Our results indicate that there is a notable spread among the CMIP6 model runs. In the ensemble mean, the spatially-averaged frequency of periods of combined low wind- and solar-power generation in Germany is not expected to change substantially in the future. The analysis on individual spatial grid cells reveals that certain areas, particularly in the eastern parts of Germany and Poland, may experience a reduction in the wind speeds and might therefore become less attractive for newly installed wind power stations. In contrast Germany's South-West becomes more attractive in terms of renewable energies in the future, according to the present analysis. This agrees with related studies. CMIP6 projections consistently indicate a decline in mean near-surface wind speeds across most of Europe over the 21st century. Under SSP2-4.5, reductions of around 5-15% can be expected, and up to 20% under SSP5-8.5—particularly in northern and central regions [144]. For Germany, however, both raw CMIP6 output and downscaled analyses suggest only minor changes in mean wind-power output, with some small increases under SSP2-4.5 [52]. More critical for the energy system is variability. Climate change is expected to increase the likelihood of rare but multi-day wind droughts by 20-40% [164]. At the same time, winter-storm winds intensify in Northern Europe. Both effects pose challenges for grid stability and adequacy [132]. Notably, such effects were not examined in this study and remain subject to future research.

The downscaling framework presented here is a powerful tool for generating high-resolution climate projections that can be used to assess the potential impacts of climate change on renewable energy resources. However, this work focuses solely on the weather-induced likelihood of periods of combined low wind- and solar-power generation. This is only one aspect of the overall risk assessment for renewable energy systems. Other factors, such as the availability of storage technologies, grid infrastructure, and demand-side management strategies also play a crucial role in determining the overall resilience of the energy system [111]. Future work is therefore needed to integrate these aspects into a comprehensive risk assessment framework for renewable energy systems. One potential candidate would be the `PyPSA` package [27]. The package has already been used to simulate the European energy-grid's transition towards renewable energy sources [98, 69] and could be used to assess the impact of Dunkelflaute events on the grid's stability. These should then be coupled with local demand time series and storage capacities. To explore the grid stability, one could consider coupling the downscaled data with a grid-simulation model that simulates the flow of electricity through the grid.

The methodology presented here is general and can be applied to any region by training the generative downscaling framework on local high-resolution reanalysis data and conditioning on relevant coarse climate model outputs. This enables assessment of Dunkelflaute risks or similar meteorological events in different geographic contexts. The presented climate-change impact assessment uses the CMIP6 ensemble as input. These are tied to hard-coded emission pathways, the so-called shared socioeconomic pathways (SSPs) and representative concentration pathways (RCPs). These encode standardized scenarios describing different future trajectories of global societal development, such as, e.g., population growth, economic development, technological progress, combined with different levels of greenhouse gas emissions. Thus, these pathways offer only limited flexibility for "what-if" questions outside the existing SSP-RCP matrix. Future work could address this limitation by further developing our framework [186] into a conditional generative one that, for specified conditions (e.g., $+1.5\,^\circ\mathrm{C}$ warming), produces physically plausible coherent ensembles of meteorological fields and translates them into impact-ready variables. This could enable rapid, flexible, and high-fidelity generation of multiple hypothetical scenarios that are tailored to decision-making requirements.

PART III

Conclusion

"Physicists like to think that all you have to do is say, 'These are the conditions, now what happens next?'"
- Richard P. Feynman

6 Conclusion

6.1 Synthesis

Reasoning about unknown dynamics under uncertainty about the mechanistic model and the observed time series is highly relevant in numerous scientific domains. ML models and learning algorithms promise to revolutionize scientific inquiry through rapid processing of large amounts of data. Parallel to the trend of automated learning and reasoning, there is an argument to be made for *explicitly* integrating prior knowledge and uncertainty into inference problems. The term *"parallel"* is chosen consciously, here. It would be wrong to conclude that imposing structure and embracing flexibility are methodological or philosophical antitheses. Both lines of work presented in this thesis share the goal to efficiently estimate the state of an unknown, high-dimensional dynamical system, taking both mechanistic knowledge and empirical observations into account. Meanwhile, the respective approaches to the interaction of prior knowledge and constraints with flexible learning and inference routines are based on different perspectives.

Using GM processes as flexible prior beliefs over dynamics enables optimal state estimation from data through Kalman filtering and smoothing. However, storing, computing, and propagating quadratically-growing covariance matrices within the filtering and smoothing recursions make inference in GM models infeasible for high-dimensional systems. Chapter 3 addressed this computational issue by finding a low-dimensional representation of the covariance matrices. We showed that an initial *optimal* rank-r approximation of the state covariances can be propagated through the filtering recursions. If the pairwise errors evolve (approximately) on a rank-r^* subspace, the rank-reduced Kalman filter (RRKF) converges to the optimal state estimate as $r \to r^*$. Thereby, the asymptotic computational complexity of the KF is reduced by at least one order of magnitude. Additional structure in the model, such as the Kronecker structure of spatiotemporal GM priors, can be leveraged to further reduce the complexity.

One might argue that, despite their theoretical merits, linear systems fall short of describing complex, nonlinear dependencies that characterize, for instance, atmospheric dynamics. Integrating such spatiotemporal systems into an inference pipeline entails a two-fold challenge. Firstly, expensive numerical simulation risks amplifying the computational burden of inference. And secondly, a probabilistic representation of the dynamics of interest is required for reasoning under (model) uncertainty. In Chapter 4, we addressed both challenges at once using a conditioned generative diffusion model. We showed that, by first learning a powerful probabilistic representation of our target dynamics and then conditioning this physical prior on additional mechanistic constraints from climate simulations, statistical downscaling can be effectively approached as a Bayesian inverse problem. The generative surrogate model circumvents the need for expensive simulation of a global weather-forecasting model, while providing structured uncertainty through sampling. Chapter 5 leveraged the probabilistic prediction of spatiotemporal weather dynamics from climate projections for investigating the risk of Dunkelflaute events in different hypothetical future scenarios. We found that refining both the spatial and the temporal resolution of the considered atmospheric variables is useful for accurately estimating local wind and solar power generation.

6.2 Discussion and Outlook

This section addresses some open questions and outlines potential directions for future research based on the contributions presented in Part II.

Model uncertainty in the RRKF vs. the EnKF

Chapter 3 lays the groundwork for a methodological extension of low-rank Gaussian filtering. In this early stage, the RRKF is currently limited to linear dynamics priors with a GM process-noise model. Meanwhile, variants of the EnKF are used operationally in large-scale applications [31]. From the viewpoint of applicability, the comparison of both methods is thus arguably difficult. On the methodological level, however, comparing both methods in their role as approximate Gaussian filters sheds light on a noteworthy distinction concerning model uncertainty. Forecasting in ensemble filters amounts to propagating each ensemble member through a numerical simulator. The "pushed-forward" stochasticity from the state samples through the deterministic mechanistic system is then used for the approximate update step. This is an intuitive and effective approach to marrying mechanistic ("forecasting") information and observations ("analysis") and partly explains the broad applicability of ensemble filters. However, avoiding an explicit process-noise model confuses two distinct sources of uncertainty: *sampling error* that is introduced to reduce computational load is used as a surrogate for *model error* in the conditioning step. In practice, this byproduct of algorithmic stochasticity can be mitigated by "inflating" the low-rank covariance with a positive constant [31]. The RRKF, on the other hand, enables an efficient treatment of Gaussian process noise for high-dimensional linear systems. A natural next step towards a comparison of both methods in realistic applications is an extension of the RRKF to efficiently incorporate nonlinear dynamics.

Choosing a low-rank representation in the RRKF

The RRKF algorithm is developed based on the assumption that a prior rank-r factorization of the (initial) covariance is available. In practical settings, this might not be the case. In such instances, (i) how to choose the reduced dimension r and (ii) how to construct the initial low-rank factorization remain open questions.

(i) Currently, r is assumed to be a constant hyper parameter that is selected before running the filter, either from prior knowledge or through hyper-parameter tuning. *Rank-adaptive* versions of DLRA [35] could be investigated as a potential approach to tuning r during the filtering recursions.

(ii) In the presented experiments, the low-rank factorization of the initial covariance was computed from the full stationary covariance matrix of the prior process. One could consider randomized, or sketching methods for computing matrix factorizations [83, 216] to mitigate computational overhead prior to the filtering recursions.

Sketching methods for low-rank-plus-diagonal approximations [67] could, furthermore, play a role in follow-up work that addresses the over-confidence problem that arises from truncating eigen-directions of the covariance (cf. Section 3.5).

Artificial weather scenarios

In the downscaling framework presented in Chapter 4 and used in Chapter 5, the conditioning is entirely independent of model training. Choosing a forward model is what defines the inference task, for which the trained diffusion model provides a rich prior belief. Statistical downscaling is one example. By defining the forward model as destroying spatiotemporal information through aggregation (Equation (4.1)), samples from the posterior predict the associated (disaggregated) fine-scale weather dynamics. Notably, the trained generative prior can be *reused*, serving as a foundation model for different inference tasks, simply by replacing the forward model.

A promising application of this generative framework could be impact analyses based on artificial weather dynamics. Chapter 5 linked estimating the future risk of renewable-energy shortages to existing, yet immensely expensive, climate simulations through the downscaling-likelihood. Generating scenarios that are (weakly) constrained, for instance, on different average temperatures or local wind-speed conditions, could perspectively provide a cost-efficient alternative to relying on climate simulations.

On abandoning an explicit physical model for atmospheric dynamics

The generative prior model in Chapter 4 serves as a statistical surrogate for the latent dynamics. It is justified and necessary to bridge gaps between methodological ML research and domain-specific inquiries and applications. Atmospheric modeling is one domain that profits already from integrating modern ML tools [4, 161, 122, 25, 125]. In the process, it is important to consider that circumventing an explicit physical model comes at the risk of sacrificing guarantees and a certain degree of robustness.

In particular, extrapolation to system states that are not covered in the data set is, for a statistical model, of purely statistical nature. This is especially relevant for atmospheric models, considering climate change and the need to analyze very long time horizons. For this thesis, it is important to note that the DM framework presented in Chapter 4 and used for further analysis in Chapter 5 is dependent on conditioning throughout the considered time period. Without conditioning, the model quickly reverts to generating dynamics that resemble

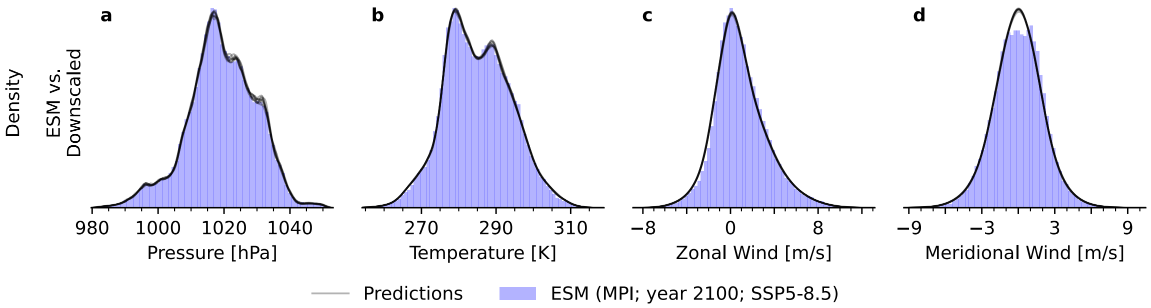


Figure 6.1: **Downscaled climate projections in the year 2100.** This plot visualizes densities of coarse climate input and the downscaled predictions (cf. Figure 4.5). Panels **a**–**d** show the four considered atmospheric variables. The coarse input are future climate projections, in the year 2100 under the high-emission scenario according to the shared socioeconomic pathway [160] SSP5-8.5. The densities of the predictions (black dotted lines) match the density of the future climate projections.

the training-data in distribution. For downscaling, a minimal requirement to our proposed framework is that it is robust to plausible distribution shifts in the coarse input, which are to be expected in future climatic conditions. Future analyses, like the one conducted in Chapter 5, rely on this robustness, which is additionally attested in Figure 6.1.

Naturally, no model is perfect. While physical laws are robust in theory, numerical simulation is fundamentally bound to a tradeoff between cost and accuracy. The efficacy of data-driven models heavily depends on the availability and quality of training data, and on potentially hard-to-interpret choices in modeling and training. The advancement of both philosophies and perspectives regarding their combination motivate the research in this thesis.

CONCLUSION

`"We are drowning in information, while starving for wisdom.`
`The world henceforth will be run by synthesizers, people able`
`to put together the right information at the right time, think`
`critically about it, and make important choices wisely."`
`- Edward Osborne Wilson`

High-dimensional dynamics systems continue to pose a challenge for probabilistic inference—both computational and conceptual. By advancing foundational methodology towards practical usability and exploring advanced ML models in the context of applications, this thesis acknowledges the merits of different approaches to addressing this worthwhile challenge.

Bibliography

[1] Australian Data Archive for Meteorology (ADAM). *Australian Water Availability Project.* http://www.bom.gov.au/jsp/awap/index.jsp. [Online; accessed May 2023]. 2023.

[2] Vincent Adam, Stefanos Eleftheriadis, Artem Artemev, Nicolas Durrande, and James Hensman. “Doubly Sparse Variational Gaussian Processes”. In: *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS).* Vol. 108. Proceedings of Machine Learning Research. 2020, pp. 2874–2884.

[3] Henry Addison, Elizabeth Kendon, Suman Ravuri, Laurence Aitchison, and Peter AG Watson. “Machine learning emulation of precipitation from km-scale regional climate simulations using a diffusion model”. In: *arXiv preprint arXiv:2407.14158* (2024).

[4] Rilwan A. Adewoyin, Peter Dueben, Peter Watson, Yulan He, and Ritabrata Dutta. “TRU-NET: a deep learning approach to high resolution prediction of rainfall”. In: *Machine Learning* 110.8 (2021), pp. 2035–2062.

[5] Michael Aich, Philipp Hess, Baoxiang Pan, Sebastian Bathiany, Yu Huang, and Niklas Boers. “Conditional diffusion models for downscaling & bias correction of Earth system model precipitation”. In: *EGUsphere* 2025 (2025), pp. 1–21.

[6] Mohammad Allouche, Elie Bou-Zeid, and Juho Iipponen. “The influence of synoptic wind on land–sea breezes”. In: *Quarterly Journal of the Royal Meteorological Society* 149.757 (2023), pp. 3198–3219.

[7] Brian D.O. Anderson. “Reverse-time diffusion equation models”. In: *Stochastic Processes and their Applications* 12.3 (1982), pp. 313–326.

[8] Jeffrey L. Anderson. “An Ensemble Adjustment Kalman Filter for Data Assimilation”. In: *Monthly Weather Review* 129.12 (2001), pp. 2884 –2903.

[9] Martin B Andrews, Jeff K Ridley, Richard A Wood, Timothy Andrews, Edward W Blockley, Ben Booth, Eleanor Burke, Andrea J Dittus, Piotr Florek, Lesley J Gray, et al. “Historical simulations with HadGEM3-GC3.1 for CMIP6”. In: *Journal of Advances in Modeling Earth Systems* 12.6 (2020), e2019MS001995.

[10] Patrik Axelsson and Fredrik Gustafsson. “Discrete-Time Solutions to the Continuous-Time Differential Lyapunov Equation With Applications to Kalman Filtering”. In: *IEEE Transactions on Automatic Control* 60.3 (2015), pp. 632–643.

[11] Hassen Babaousmail, Rongtao Hou, Gnim Tchalim Gnitou, and Brian Ayugi. “Novel statistical downscaling emulator for precipitation projections using deep Convolutional Autoencoder over Northern Africa”. In: *Journal of Atmospheric and Solar-Terrestrial Physics* 218 (2021), p. 105614.

[12] Rocío Balmaceda-Huarte, Jorge Baño-Medina, Matias Ezequiel Olmo, and Maria Laura Bettolli. “On the use of convolutional neural networks for downscaling daily temperatures over southern South America in a climate change scenario”. In: *Climate Dynamics* 62.1 (2024), pp. 383–397.

[13] Roger G Barry and Richard J Chorley. *Atmosphere, weather and climate.* 1st ed. Routledge, 2009.

[14] Peter Bauer, Alan Thorpe, and Gilbert Brunet. “The quiet revolution of numerical weather prediction”. In: *Nature* 525.7567 (2015), pp. 47–55.

[15] Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. “Automatic Differentiation in Machine Learning: a Survey”. In: *Journal of Machine Learning Research* 18.153 (2018), pp. 1–43.

[16] Jorge Baño-Medina, Rodrigo Manzanas, and José Manuel Gutiérrez. “On the suitability of deep convolutional neural networks for continental-wide downscaling of climate change projections”. In: *Climate Dynamics* 57.11 (2021), pp. 2941–2951.

[17] Stanley G Benjamin, John M Brown, Gilbert Brunet, Peter Lynch, Kazuo Saito, and Thomas W Schlatter. “100 years of progress in forecasting and NWP applications”. In: *Meteorological Monographs* 59 (2019), pp. 13–1.

[18] José M. Bernardo and Adrian F.M. Smith. *Bayesian theory.* Vol. 586. Wiley Online Library, 1994.

[19] Benjamin Biewald, Bastien Cozian, Laurent Dubus, William Zappa, and Laurens P. Stoop. “Evaluation of ‘Dunkelflaute’ event detection methods considering grid operators’ needs”. In: *Environ. Res.: Energy* 2.2 (2025), p. 025007.

[20] Sebastian Bischoff, Alana Darcher, Michael Deistler, Richard Gao, Franziska Gerken, Manuel Gloeckler, Lisa Haxel, Jaivardhan Kapoor, Janne K Lappalainen, Jakob H Macke, et al. “A practical guide to sample-based statistical distances for evaluating generative models in science”. In: *Transactions on Machine Learning Research* (2024).

[21] Tobias Bischoff and Katherine Deck. “Unpaired Downscaling of Fluid Flows with Diffusion Bridges”. In: *Artificial Intelligence for the Earth Systems* 3.2 (2024), e230039.

[22] Christopher M. Bishop. *Pattern Recognition and Machine Learning.* 1st ed. Information Science and Statistics. Springer, 2006. ISBN: 978-0-387-31073-2.

[23] Craig. H. Bishop, B. J. Etherton, and S. J. Majumdar. “Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects”. In: *Monthly weather review* 129.3 (2001), pp. 420–436.

[24] C. Bollmeyer, J. D. Keller, C. Ohlwein, S. Wahl, S. Crewell, P. Friederichs, A. Hense, J. Keune, S. Kneifel, I. Pscheidt, S. Redl, and S. Steinke. “Towards a high-resolution regional reanalysis for the European CORDEX domain”. In: *Quarterly Journal of the Royal Meteorological Society* 141.686 (2015), pp. 1–15.

[25] Boris Bonev, Thorsten Kurth, Ankur Mahesh, Mauro Bisson, Jean Kossaifi, Karthik Kashinath, Anima Anandkumar, William D Collins, Michael S Pritchard, and Alexander Keller. “FourCastNet 3: A geometric approach to probabilistic machine-learning weather forecasting at scale”. In: *arXiv preprint arXiv:2507.12144* (2025).

[26] Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of Markov Chain Monte Carlo.* CRC press, 2011.

[27] Thomas Brown, Jonas Hörsch, and David Schlachtberger. “PyPSA: Python for Power System Analysis”. In: *J. Open Res. Software* 6.1 (2018).

[28] Bundesnetzagentur. *Growth in renewable energy in 2024—installed capacities of solar and wind in Germany.* Press release, 8 January 2025. Accessed: 2025-07-21. 2025.

[29] Hanqun Cao, Cheng Tan, Zhangyang Gao, Yilun Xu, Guangyong Chen, Pheng-Ann Heng, and Stan Z. Li. “A Survey on Generative Diffusion Models”. In: *IEEE Transactions on Knowledge and Data Engineering* 36.7 (2024), pp. 2814–2830.

[30] O. Cappé, E. Moulines, and T. Ryden. *Inference in Hidden Markov Models.* Springer Series in Statistics. Springer, 2005. ISBN: 9780387402642.

[31] A. Carrassi, M. Bocquet, L. Bertino, and G. Evensen. “Data assimilation in the geosciences: An overview of methods, issues, and perspectives”. In: *Wiley Interdisciplinary Reviews: Climate Change* 9.5 (2018).

[32] C. Carrillo, A.F. Obando Montaño, J. Cidrás, and E. Díaz-Dorado. “Review of power curve modelling for wind turbines”. In: *Renewable and Sustainable Energy Reviews* 21 (2013), pp. 572–581.

[33] A. Carron, M. Todescato, R. Carli, L. Schenato, and G. Pillonetto. “Machine learning meets Kalman filtering”. In: *2016 IEEE 55th conference on decision and control (CDC).* IEEE. 2016, pp. 4594–4599.

[34] L. Cavaleri, G. Balsamo, A. Beljaars, L. Bertotti, S. Davison, J. Edwards, T. Kanehama, and N. Wedi. “ECMWF and UK Met Office Offshore Blowing Winds: Impact of Horizontal Resolution and Coastal Orography”. In: *J. Geophys. Res. Atmos.* 129.6 (2024), e2023JD039673.

[35] Gianluca Ceruti, Jonas Kusch, and Christian Lubich. “A rank-adaptive robust integrator for dynamical low-rank approximation”. In: *BIT Numerical Mathematics* 62.4 (2022), pp. 1149–1174.

[36] Gianluca Ceruti and Christian Lubich. “An unconventional robust integrator for dynamical low-rank approximation”. In: *BIT. Numerical Mathematics* 62.1 (2022), pp. 23–44.

[37] N. K. Chada, A. M. Stuart, and X. T. Tong. “Tikhonov regularization within ensemble Kalman inversion”. In: *SIAM Journal on Numerical Analysis* 58.2 (2020), pp. 1263–1294.

[38] D. Chen, M. Rojas, B.H. Samset, K. Cobb, A. Diongue Niang, P. Edwards, S. Emori, S.H. Faria, E. Hawkins, P. Hope, P. Huybrechts, M. Meinshausen, S.K. Mustafa, G.-K. Plattner, and A.-M. Tréguier. “Framing, Context, and Methods”. In: *Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change.* Cambridge, United Kingdom and New York, NY, USA: Cambridge University Press, 2021, pp. 147–286.

[39] Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. “Diffusion Posterior Sampling for General Noisy Inverse Problems”. In: *The Eleventh International Conference on Learning Representations (ICLR).* 2023.

[40] Jennifer Collins. “Germany’s electricity mix in 2024 ’cleanest ever’ — researchers”. In: *Clean Energy Wire* (2025). Accessed: 2025-07-21.

[41] Copernicus Climate Change Service (C3S). *Climate Data for Energy: Pan-European Climate Database (PECD) v4.0.* Dataset, Copernicus Climate Change Service. 2022.

[42] Noel Cressie, Tao Shi, and Emily L. Kang. “Fixed Rank Filtering for Spatio-Temporal Data”. In: *Journal of Computational and Graphical Statistics* 19.3 (2010), pp. 724–745.

[43] A. P. Dawid. “Present Position and Potential Developments: Some Personal Views: Statistical Theory: The Prequential Approach”. In: *Journal of the Royal Statistical Society. Series A (General)* 147.2 (1984), pp. 278–292.

[44] Prafulla Dhariwal and Alexander Quinn Nichol. “Diffusion Models Beat GANs on Image Synthesis”. In: *Advances in Neural Information Processing Systems (NeurIPS).* 2021.

[45] Laurent Dinh, David Krueger, and Yoshua Bengio. “NICE: Non-linear Independent Components Estimation”. In: *arXiv preprint arXiv:1410.8516* (2014).

[46] F.J. Doblas-Reyes, A.A. Sörensson, M. Almazroui, A. Dosio, W.J. Gutowski, R. Haarsma, R. Hamdi, B. Hewitson, W.-T. Kwon, B.L. Lamptey, D. Maraun, T.S. Stephenson, I. Takayabu, L. Terray, A. Turner, and Z. Zuo. “Linking Global to Regional Climate Change”. In: *Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change.* Cambridge, United Kingdom and New York, NY, USA: Cambridge University Press, 2021, pp. 1363–1512.

[47] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: *International Conference on Learning Representations (ICLR).* 2021.

[48] Arnaud Doucet, Nando De Freitas, Neil James Gordon, et al. *Sequential Monte Carlo methods in practice.* Vol. 1. 2. Springer, 2001.

[49] François Duchêne, Bert Van Schaeybroeck, Rafiq Hamdi, Joris Van den Bergh, Geert Smet, Steven Caluwaerts, and Piet Termonia. “Current-Day and Future Dunkelflaute Risks for Belgium”. In: *Journal of Applied Meteorology and Climatology* 63.11 (2024), pp. 1427–1440.

[50] ECMWF. *ECMWF’s AI Forecasts Become Operational.* Accessed: 2025-10-21. European Centre for Medium-Range Weather Forecasts. 2025. URL: https://www.ecmwf.int/en/about/media-centre/news/2025/ecmwfs-ai-forecasts-become-operational.

[51] ECMWF. *ECMWF’s Ensemble AI Forecasts Become Operational.* Accessed: 2025-10-21. European Centre for Medium-Range Weather Forecasts. 2025. URL: https://www.ecmwf.int/en/about/media-centre/news/2025/ecmwfs-ensemble-ai-forecasts-become-operational.

[52] Nina Effenberger and Nicole Ludwig. “Turbine location-aware multi-decadal wind power predictions for Germany using CMIP6”. In: *Environmental Data Science* 4 (2025), e38.

[53] Nina Effenberger, Nicole Ludwig, and Rachel H. White. “Mind the (spectral) gap: how the temporal resolution of wind data affects multi-decadal wind power forecasts”. In: *Environ. Res. Lett.* 19.1 (2023), p. 014015.

[54] Enerdata. *Renewables accounted for nearly 60% of Germany’s power generation in 2024.* Online news article. Accessed: 2025-07-21. 2025.

[55] European Commission. *The European Green Deal.* Press release. Accessed: 2025-07-21. 2019.

[56] Eurostat. “Renewables take the lead in power generation in 2023”. In: *Eurostat News* (2024). Accessed: 2025-07-21.

[57] Lawrence C Evans. *Partial differential equations.* Vol. 19. American mathematical society, 2022.

[58] G. Evensen. *Data assimilation: the ensemble Kalman filter.* Vol. 2. Springer, 2009.

[59] G. Evensen. “Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics”. In: *Journal of Geophysical Research: Oceans* 99.C5 (1994), pp. 10143–10162.

[60] G. Evensen. “The ensemble Kalman filter for combined state and parameter estimation”. In: *IEEE Control Systems Magazine* 29.3 (2009), pp. 83–104.

[61] Geir Evensen. “Formulating the history matching problem with consistent error statistics”. In: *Computational Geosciences* 25.3 (2021).

[62] Geir Evensen. “The ensemble Kalman filter: theoretical formulation and practical implementation”. In: *Ocean Dynamics* 53.4 (2003), pp. 343–367.

[63] Veronika Eyring, Sandrine Bony, Gerald A. Meehl, Catherine A. Senior, Bjorn Stevens, Ronald J. Stouffer, and Karl E. Taylor. “Overview of the Coupled Model Intercomparison Project Phase 6 (CMIP6) experimental design and organization”. In: *Geosci. Model Dev.* 9.5 (2016), pp. 1937–1958.

[64] Brian F. Farrell and Petros J. Ioannou. “State estimation using a reduced-order Kalman filter”. In: *Journal of the Atmospheric Sciences* 58.23 (2001), pp. 3666–3680.

[65] W Feller. “On the Theory of Stochastic Processes, with Particular Reference to Applications”. In: *First Berkeley Symposium on Mathematical Statistics and Probability.* 1949, pp. 403–432.

[66] Berthy T Feng, Jamie Smith, Michael Rubinstein, Huiwen Chang, Katherine L Bouman, and William T Freeman. “Score-based diffusion models as principled priors for inverse imaging”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2023, pp. 10520–10531.

[67] Andres Fernandez, Felix Dangel, Philipp Hennig, and Frank Schneider. “Sketching Low-Rank Plus Diagonal Matrices”. In: *arXiv preprint arXiv:2509.23587* (2025).

[68] Bengt Fornberg. “Generation of finite difference formulas on arbitrarily spaced grids”. In: *Mathematics of computation* 51.184 (1988), pp. 699–706.

[69] Martha Frysztacki and Tom Brown. “Modeling Curtailment in Germany: How Spatial Resolution Impacts Line Congestion”. In: *2020 17th International Conference on the European Energy Market (EEM).* IEEE, 2020, pp. 16–18.

[70] A. Galka, O. Yamashita, T. Ozaki, R. Biscay, and P. Valdés-Sosa. “A solution to the dynamical inverse problem of EEG generation using spatiotemporal Kalman filtering”. In: *NeuroImage* 23.2 (2004), pp. 435–453.

[71] Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis.* 3rd ed. Chapman and Hall/CRC, 2013. ISBN: 978-1-4398-4095-5.

[72] M. Ghil and P. Malanotte-Rizzoli. “Data assimilation in meteorology and oceanography”. In: *Advances in geophysics.* Vol. 33. Elsevier, 1991, pp. 141–266.

[73] Filippo Giorgi, Colin Jones, Ghassem R Asrar, et al. “Addressing climate information needs at the regional level: the CORDEX framework”. In: *World Meteorological Organization (WMO) Bulletin* (2009).

[74] Gene H. Golub and Charles F. Van Loan. *Matrix computations.* 4th ed. JHU press, 2013.

[75] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning.* MIT Press, 2016.

[76] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. “Generative Adversarial Nets”. In: *Advances in Neural Information Processing Systems (NeurIPS).* Vol. 27. 2014.

[77] M.S. Grewal and A.P. Andrews. *Kalman Filtering: Theory and Practice Using MATLAB.* Wiley, 2011.

[78] Andreas Griewank and Andrea Walther. *Evaluating Derivatives.* 2nd ed. Society for Industrial and Applied Mathematics, 2008.

[79] Brian Groenke, Luke Madaus, and Claire Monteleoni. “ClimAlign: Unsupervised statistical downscaling of climate variables via normalizing flows”. In: *Proceedings of the 10th International Conference on Climate Informatics*. 2020, pp. 60–66.

[80] Sabine Haas, Uwe Krien, Birgit Schachler, Stickler Bot, Velibor Zeli, Florian Maurer, Kumar Shivam, Francesco Witte, Sasan Jacob Rasti, Seth, and Stephen Bosch. *wind-python/windpowerlib: Update release*. Version v0.2.2. 2024.

[81] Ernst Hairer, Syvert Norsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Vol. 8. Springer-Verlag, 1993. ISBN: 978-3-540-56670-0.

[82] Ernst Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*. Vol. 14. Springer-Verlag, 1996.

[83] N. Halko, P. G. Martinsson, and J. A. Tropp. “Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions”. In: *SIAM Review* 53.2 (2011), pp. 217–288.

[84] O. Hamelijnck, W. Wilkinson, N. Loppi, A. Solin, and T. Damoulas. “Spatio-temporal variational Gaussian processes”. In: *Advances in Neural Information Processing Systems (NeurIPS)* 34 (2021), pp. 23621–23633.

[85] T. M. Hamill and C. Snyder. “A hybrid ensemble Kalman filter–3D variational analysis scheme”. In: *Monthly Weather Review* 128.8 (2000), pp. 2905–2919.

[86] Paula Harder, Alex Hernandez-Garcia, Venkatesh Ramesh, Qidong Yang, Prasanna Sattegeri, Daniela Szwarcman, Campbell Watson, and David Rolnick. “Hard-Constrained Deep Learning for Climate Downscaling”. In: *Journal of Machine Learning Research* 24.365 (2023), pp. 1–40.

[87] Lucy Harris, Andrew T. T. McRae, Matthew Chantry, Peter D. Dueben, and Tim N. Palmer. “A Generative Deep Learning Approach to Stochastic Downscaling of Precipitation Forecasts”. In: *Journal of Advances in Modeling Earth Systems* 14.10 (2022).

[88] James Hensman, Nicolò Fusi, and Neil D Lawrence. “Gaussian processes for Big Data”. In: *Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence (UAI)*. 2013, pp. 282–290.

[89] H. Hersbach, B. Bell, P. Berrisford, G. Biavati, A. Horányi, J. Muñoz Sabater, J. Nicolas, C. Peubey, R. Radu, I. Rozum, D. Schepers, A. Simmons, C. Soci, D. Dee, and J-N. Thépaut. *ERA5 hourly data on single levels from 1979 to present*. Accessed: 02-03-2022. 2018.

[90] Philipp Hess, Michael Aich, Baoxiang Pan, and Niklas Boers. “Fast, scale-adaptive and uncertainty-aware downscaling of Earth system model fields with generative machine learning”. In: *Nat. Mach. Intell.* 7 (2025), pp. 363–373.

[91] Philipp Hess, Markus Drüke, Stefan Petri, Felix M. Strnad, and Niklas Boers. “Physically constrained generative adversarial networks for improving precipitation fields from Earth system models”. In: *Nature Machine Intelligence* 4.10 (2022), pp. 828–839.

[92] Philipp Hess, Stefan Lange, Christof Schötz, and Niklas Boers. “Deep Learning for Bias-Correcting CMIP6-Class Earth System Models”. In: *Earth’s Future* 11.10 (2023), e2023EF004002.

[93] Herbert W. Hethcote. “The Mathematics of Infectious Diseases”. In: *SIAM Review* 42.4 (2000), pp. 599–653.

[94] Jonathan Ho, Ajay Jain, and Pieter Abbeel. “Denoising Diffusion Probabilistic Models”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 33. 2020, pp. 6840–6851.

[95] Jonathan Ho and Tim Salimans. “Classifier-Free Diffusion Guidance”. In: *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*. 2021.

[96] A. L. Hodgkin and A. F. Huxley. “A quantitative description of membrane current and its application to conduction and excitation in nerve”. en. In: *The Journal of Physiology* 117.4 (1952), pp. 500–544.

[97] Fabian Hofmann, Johannes Hampp, Fabian Neumann, Tom Brown, and Jonas Hörsch. “atlite: A lightweight Python package for calculating renewable power potentials and time series”. In: *Journal of Open Source Software* 6.62 (2021), p. 3294.

[98] Jonas Hörsch, Fabian Hofmann, David Schlachtberger, and Tom Brown. “PyPSA-Eur: An open optimisation model of the European transmission system”. In: *Energy Strategy Reviews* 22 (2018), pp. 207–215.

[99] Imperial College London. *Londonair—London air quality network (LAQN)*. `https://www.londonair.org.uk`. 2020.

[100] E.M. Izhikevich. “Simple model of spiking neurons”. In: *IEEE Transactions on Neural Networks* 14.6 (2003), pp. 1569–1572.

[101] Tommi S. Jaakkola and Michael I. Jordan. “Bayesian parameter estimation via variational methods”. In: *Statistics and Computing* 10.1 (2000), pp. 25–37.

[102] Shipra Jain, Adam A Scaife, Theodore G Shepherd, Clara Deser, Nick Dunstone, Gavin A Schmidt, Kevin E Trenberth, and Thea Turkington. “Importance of internal variability for climate model assessment”. In: *npj Climate and Atmospheric Science* 6.1 (2023), p. 68.

[103] D Jones, W Wang, and R Fawcett. “High-Quality Spatial Climate Data-Sets for Australia”. In: *Australian Meteorological and Oceanographic Journal* 58.04 (2009), pp. 233–248.

[104] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. “An introduction to variational methods for graphical models”. In: *Machine learning* 37.2 (1999), pp. 183–233.

[105] Rudolph Emil Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Transactions of the ASME–Journal of Basic Engineering* 82.Series D (1960), pp. 35–45.

[106] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. "Elucidating the Design Space of Diffusion-Based Generative Models". In: *Advances in Neural Information Processing Systems (NeurIPS).* Vol. 35. 2022, pp. 26565–26577.

[107] M. Katzfuss, J. R. Stroud, and C. K. Wikle. "Understanding the ensemble Kalman filter". In: *The American Statistician* 70.4 (2016), pp. 350–357.

[108] Diederik P. Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. "Variational Diffusion Models". In: *Advances in Neural Information Processing Systems (NeurIPS).* Vol. 34. 2021, pp. 21696–21707.

[109] Diederik P. Kingma and Max Welling. "Auto-encoding variational Bayes". In: *International Conference on Learning Representations (ICLR).* 2014.

[110] Genshiro Kitagawa. "Non-Gaussian statespace modeling of nonstationary time series". In: *Journal of the American statistical association* 82.400 (1987), pp. 1032–1041.

[111] Martin Kittel and Wolf-Peter Schill. "Measuring the Dunkelflaute: how (not) to analyze variable renewable energy shortage". In: *Environ. Res.: Energy* 1.3 (2024), p. 035007.

[112] Martin Kittel and Wolf-Peter Schill. "Quantifying the Dunkelflaute: An analysis of variable renewable energy droughts in Europe". In: *arXiv preprint arXiv:2410.00244* (2024).

[113] Erik Kjellström, Grigory Nikulin, Gustav Strandberg, Ole Bøssing Christensen, Daniela Jacob, Klaus Keuler, Geert Lenderink, Erik van Meijgaard, Christoph Schär, Samuel Somot, Silje Lund Sørland, Claas Teichmann, and Robert Vautard. "European climate change at global mean temperature increases of 1.5 and 2°C above pre-industrial conditions as simulated by the EURO-CORDEX regional climate models". In: *Earth Syst. Dyn.* 9.2 (2018), pp. 459–478.

[114] William H. Klein, Billy M. Lewis, and Isadore Enger. "Objective Prediction of Five-Day Mean Temperatures During Winter". In: *Journal of Atmospheric Sciences* 16.6 (1959), pp. 672 –682.

[115] O. Koch and C. Lubich. "Dynamical low-rank approximation". In: *SIAM Journal on Matrix Analysis and Applications* 29.2 (2007), pp. 434–454.

[116] Heejoon Koo and To Eun Kim. "A Comprehensive Survey on Generative Diffusion Models for Structured Data". In: *arXiv preprint arXiv:2306.04139* (2023).

[117] S. Kotlarski, K. Keuler, O. B. Christensen, A. Colette, M. Déqué, A. Gobiet, K. Goergen, D. Jacob, D. Lüthi, E. van Meijgaard, G. Nikulin, C. Schär, C. Teichmann, R. Vautard, K. Warrach-Sagi, and V. Wulfmeyer. "Regional climate modeling on European scales: a joint standard evaluation of the EURO-CORDEX RCM ensemble". In: *Geosci. Model Dev.* 7.4 (2014), pp. 1297–1333.

[118] Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. “Neural Operator: Learning Maps Between Function Spaces With Applications to PDEs”. In: *Journal of Machine Learning Research* 24.89 (2023), pp. 1–97.

[119] Nicholas Krämer, Nathanael Bosch, Jonathan Schmidt, and Philipp Hennig. “Probabilistic ODE Solutions in Millions of Dimensions”. In: *Proceedings of the 39th International Conference on Machine Learning.* Vol. 162. Proceedings of Machine Learning Research. 2022, pp. 11634–11649.

[120] Nicholas Krämer and Philipp Hennig. “Stable Implementation of Probabilistic ODE Solvers”. In: *Journal of Machine Learning Research* 25.111 (2024), pp. 1–29.

[121] Nicholas Krämer, Jonathan Schmidt, and Philipp Hennig. “Probabilistic Numerical Method of Lines for Time-Dependent Partial Differential Equations”. In: *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics.* Vol. 151. Proceedings of Machine Learning Research. 2022, pp. 625–639.

[122] Thorsten Kurth, Shashank Subramanian, Peter Harrington, Jaideep Pathak, Morteza Mardani, David Hall, Andrea Miele, Karthik Kashinath, and Anima Anandkumar. “Fourcastnet: Accelerating global high-resolution weather forecasting using adaptive Fourier neural operators”. In: *Proceedings of the platform for advanced scientific computing conference.* 2023, pp. 1–11.

[123] David C. Lafferty and Ryan L. Sriver. “Downscaling and bias-correction contribute considerable uncertainty to local climate projections in CMIP6”. In: *npj Clim. Atmos. Sci.* 6.158 (2023), pp. 1–13.

[124] Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, Alexander Merose, Stephan Hoyer, George Holland, Oriol Vinyals, Jacklynn Stott, Alexander Pritzel, Shakir Mohamed, and Peter Battaglia. “Learning skillful medium-range global weather forecasting”. In: *Science* 382.6677 (2023), pp. 1416–1421.

[125] Simon Lang, Mihai Alexe, Matthew Chantry, Jesper Dramsch, Florian Pinault, Baudouin Raoult, Mariana C. A. Clare, Christian Lessig, Michael Maier-Gerber, Linus Magnusson, Zied Ben Bouallègue, Ana Prieto Nemesio, Peter D. Dueben, Andrew Brown, Florian Pappenberger, and Florence Rabier. “AIFS – ECMWF’s data-driven forecasting system”. In: *arXiv preprint arXiv:2406.01465* (2024).

[126] Michael Langguth, Ankit Patnala, Sebastian Lehner, Markus Dabernig, Konrad Mayer, Irene Schicker, GeoSphere Austria, and Paula Harder. “A benchmark dataset for meteorological downscaling”. In: *International Conference on Learning Representations.* 2024.

[127] René Laprise. “Regional climate modelling”. In: *J. Comput. Phys.* 227.7 (2008), pp. 3641–3666.

[128] Christian Lessig, Ilaria Luise, Bing Gong, Michael Langguth, Scarlet Stadtler, and Martin Schultz. “AtmoRep: A stochastic model of atmosphere dynamics using large scale representation learning”. In: *arXiv preprint arXiv:2308.13280* (2023).

[129] Bowen Li, Sukanta Basu, Simon J. Watson, and Herman W. J. Russchenberg. “A Brief Climatology of Dunkelflaute Events over and Surrounding the North and Baltic Sea Areas”. In: *Energies* 14.20 (2021), p. 6508.

[130] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. “Flow Matching for Generative Modeling”. In: *The Eleventh International Conference on Learning Representations.* 2023.

[131] Yaron Lipman, Marton Havasi, Peter Holderrieth, Neta Shaul, Matt Le, Brian Karrer, Ricky T. Q. Chen, David Lopez-Paz, Heli Ben-Hamu, and Itai Gat. “Flow Matching Guide and Code”. In: *arXiv preprint arXiv:2412.06264* (2024).

[132] Alexander S. Little, Matthew D. K. Priestley, and Jennifer L. Catto. “Future increased risk from extratropical windstorms in northern Europe”. In: *Nat. Commun.* 14.4434 (2023), pp. 1–10.

[133] Jun S Liu. *Monte Carlo strategies in scientific computing.* Vol. 10. Springer.

[134] Jean-Christophe Loiseau, Bernd R Noack, and Steven L Brunton. “Sparse reduced-order modelling: sensor-based dynamics to full-state estimation”. In: *Journal of Fluid Mechanics* 844 (2018), pp. 459–490.

[135] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. “Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators”. In: *Nature Machine Intelligence* 3.3 (2021), pp. 218–229.

[136] C. Lubich and Ivan V. Oseledets. “A projector-splitting integrator for dynamical low-rank approximation”. In: *BIT Numerical Mathematics* 54.1 (2014), pp. 171–188.

[137] David JC MacKay. “Developments in probabilistic modelling with neural networks—ensemble learning”. In: *Neural Networks: Artificial Intelligence and Industrial Applications: Proceedings of the Third Annual SNN Symposium on Neural Networks, Nijmegen, The Netherlands, 14–15 September 1995.* Springer. 1995, pp. 191–198.

[138] D. Maraun, F. Wetterhall, A. M. Ireson, R. E. Chandler, E. J. Kendon, M. Widmann, S. Brienen, H. W. Rust, T. Sauter, M. ThemeSSl, V. K. C. Venema, K. P. Chun, C. M. Goodess, R. G. Jones, C. Onof, M. Vrac, and I. Thiele-Eich. “Precipitation downscaling under climate change: Recent developments to bridge the gap between dynamical models and the end user”. In: *Reviews of Geophysics* 48.3 (2010).

[139] Douglas Maraun. “Bias correction, quantile mapping, and downscaling: Revisiting the inflation issue”. In: *Journal of Climate* 26.6 (2013), pp. 2137–2143.

[140] Douglas Maraun and Martin Widmann. *Statistical Downscaling and Bias Correction for Climate Research.* Cambridge University Press, 2018.

[141] Douglas Maraun, Martin Widmann, José M. Gutiérrez, Sven Kotlarski, Richard E. Chandler, Elke Hertig, Joanna Wibig, Radan Huth, and Renate A.I. Wilcke. “VALUE: A framework to validate downscaling approaches for climate change studies”. In: *Earth’s Future* 3.1 (2015), pp. 1–14.

[142] Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. “A Variational Perspective on Solving Inverse Problems with Diffusion Models”. In: *The Twelfth International Conference on Learning Representations*. 2024.

[143] Maia Martcheva. *An introduction to mathematical epidemiology*. Vol. 61. Springer, 2015.

[144] Abel Martinez and Gregorio Iglesias. “Climate-change effects on wind resources in Europe and North America based on the Shared Socioeconomic Pathways”. In: *[Journal of Sustainable Development of Energy, Water and Environment Systems]* [12].[3] (2024), [1]–[16].

[145] Kendal McGuffie and Ann Henderson-Sellers. *A Climate Modelling Primer*. John Wiley & Sons, Ltd, 2005. ISBN: 9780470857618.

[146] J. Baño Medina, R. Manzanas, and J. M. Gutiérrez. “Configuration and intercomparison of deep learning neural models for statistical downscaling”. In: *Geoscientific Model Development* 13.4 (2020), pp. 2109–2124.

[147] Gerald A. Meehl, Catherine A. Senior, Veronika Eyring, Gregory Flato, Jean-Francois Lamarque, Ronald J. Stouffer, Karl E. Taylor, and Manuel Schlund. “Context for interpreting equilibrium climate sensitivity and transient climate response from the CMIP6 Earth system models”. In: *Sci. Adv.* 6.26 (2020).

[148] Edgar Meza. “German government approves draft law to accelerate renewables expansion”. In: *Clean Energy Wire* (2025). Accessed: 2025-07-21.

[149] Fabian Mockert, Christian M. Grams, Tom Brown, and Fabian Neumann. “Meteorological conditions during periods of low wind speed and insolation in Germany: The role of weather regimes”. In: *Meteorol. Appl.* 30.4 (2023), e2141.

[150] Julia Moemken, Mark Reyers, Hendrik Feldmann, and Joaquim G. Pinto. “Future Changes of Wind Speed and Wind Energy Potentials in EURO-CORDEX Ensemble Simulations”. In: *J. Geophys. Res. Atmos.* 123.12 (2018), pp. 6373–6389.

[151] J.-Y. Moon, J. Streffing, S.-S. Lee, T. Semmler, M. Andrés-Martínez, J. Chen, E.-B. Cho, J.-E. Chu, C. Franzke, J. P. Gärtner, R. Ghosh, J. Hegewald, S. Hong, N. Koldunov, J.-Y. Lee, Z. Lin, C. Liu, S. Loza, W. Park, W. Roh, D. V. Sein, S. Sharma, D. Sidorenko, J.-H. Son, M. F. Stuecker, Q. Wang, G. Yi, M. Zapponini, T. Jung, and A. Timmermann. “Earth’s future climate and its variability simulated at 9 km global resolution”. In: *EGUsphere* 2024 (2024), pp. 1–46.

[152] Sofia Morelli, Nina Effenberger, Luca Schmidt, and Nicole Ludwig. “Climate data selection for multi-decadal wind power forecasts”. In: *Environ. Res. Lett.* 20.4 (2025), p. 044032.

[153] Wolfgang A Müller, Johann H Jungclaus, Thorsten Mauritsen, Johanna Baehr, Matthias Bittner, R Budich, Felix Bunzel, Monika Esch, Rohit Ghosh, Helmut Haak, et al. "A higher-resolution version of the Max Planck Institute Earth system model (MPI-ESM1.2-HR)". In: *Journal of Advances in Modeling Earth Systems* 10.7 (2018), pp. 1383–1413.

[154] Kevin P. Murphy. *Probabilistic Machine Learning: An introduction.* MIT Press, 2022.

[155] Marcos Netto and Lamine Mili. "A robust data-driven Koopman Kalman filter for power systems dynamic state estimation". In: *IEEE Transactions on Power Systems* 33.6 (2018), pp. 7228–7237.

[156] Tung Nguyen, Rohan Shah, Hritik Bansal, Troy Arcomano, Romit Maulik, Rao Kotamarthi, Ian Foster, Sandeep Madireddy, and Aditya Grover. "Scaling transformer neural networks for skillful and reliable medium-range weather forecasting". In: *Advances in Neural Information Processing Systems (NeurIPS)* 37 (2024), pp. 68740–68771.

[157] Alexander Quinn Nichol and Prafulla Dhariwal. "Improved Denoising Diffusion Probabilistic Models". In: *Proceedings of the 38th International Conference on Machine Learning.* Vol. 139. Proceedings of Machine Learning Research. 2021, pp. 8162–8171.

[158] Greg O'Hare, John Sweeney, and Rob Wilby. *Weather, climate and climate change: human perspectives.* Routledge, 2014.

[159] Bernt Øksendal. "Stochastic differential equations". In: *Stochastic differential equations: an introduction with applications.* Springer, 2003, pp. 38–50.

[160] Brian C. O'Neill, Elmar Kriegler, Kristie L. Ebi, Eric Kemp-Benedict, Keywan Riahi, Dale S. Rothman, Bas J. van Ruijven, Detlef P. van Vuuren, Joern Birkmann, Kasper Kok, Marc Levy, and William Solecki. "The roads ahead: Narratives for shared socioeconomic pathways describing world futures in the 21st century". In: *Global Environmental Change* 42 (2017), pp. 169–180.

[161] Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadenesheli, et al. "Fourcastnet: A global data-driven high-resolution weather model using adaptive Fourier neural operators". In: *arXiv preprint arXiv:2202.11214* (2022).

[162] Ilan Price, Alvaro Sanchez-Gonzalez, Ferran Alet, Tom R. Andersson, Andrew El-Kadi, Dominic Masters, Timo Ewalds, Jacklynn Stott, Shakir Mohamed, Peter Battaglia, Remi Lam, and Matthew Willson. "Probabilistic weather forecasting with machine learning". In: *Nature* 637 (2025), pp. 84–90.

[163] S. Pulkkinen, D. Nerini, A. A. Pérez Hortal, C. Velasco-Forero, A. Seed, U. Germann, and L. Foresti. "Pysteps: an open-source Python library for probabilistic precipitation nowcasting (v1.0)". In: *Geoscientific Model Development* 12.10 (2019), pp. 4185–4219.

[164] Meng Qu, Lu Shen, Zhenzhong Zeng, Bolei Yang, Huiru Zhong, Xinrong Yang, and Xi Lu. "Prolonged wind droughts in a warming climate threaten global wind power security". In: *Nat. Clim. Change* 15 (2025), pp. 842–849.

[165] Ali Rahimi and Benjamin Recht. “Random Features for Large-Scale Kernel Machines”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2007.

[166] Neelesh Rampal, Sanaa Hobeichi, Peter B. Gibson, Jorge Baño-Medina, Gab Abramowitz, Tom Beucler, Jose González-Abad, William Chapman, Paula Harder, and José Manuel Gutiérrez. “Enhancing Regional Climate Downscaling through Advances in Machine Learning”. In: *Artificial Intelligence for the Earth Systems* 3.2 (2024), p. 230066.

[167] David A Randall, Cecilia M Bitz, Gokhan Danabasoglu, A Scott Denning, Peter R Gent, Andrew Gettelman, Stephen M Griffies, Peter Lynch, Hugh Morrison, Robert Pincus, et al. “100 years of Earth system model development”. In: *Meteorological Monographs* 59 (2019), pp. 12–1.

[168] CE. Rasmussen and CKI. Williams. *Gaussian Processes for Machine Learning.* MIT Press, 2006.

[169] Herbert E Rauch, F Tung, and Charlotte T Striebel. “Maximum likelihood estimates of linear dynamic systems”. In: *AIAA journal* 3.8 (1965), pp. 1445–1450.

[170] Mark D. Risser, Stefan Rahimi, Naomi Goldenson, Alex Hall, Zachary J. Lebo, and Daniel R. Feldman. “Is Bias Correction in Dynamical Downscaling Defensible?” In: *Geophys. Res. Lett.* 51.10 (2024), e2023GL105979.

[171] Christian P Robert and George Casella. *Monte Carlo statistical methods.* 2nd ed. Vol. 2. Springer, 2004.

[172] O. Ronneberger, P.Fischer, and T. Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI).* Vol. 9351. LNCS. 2015, pp. 234–241.

[173] M. Roth, G. Hendeby, C. Fritsche, and F. Gustafsson. “The Ensemble Kalman filter: a signal processing perspective”. In: *EURASIP Journal on Advances in Signal Processing* 2017.1 (2017), pp. 1–16.

[174] François Rozet and Gilles Louppe. “Score-based Data Assimilation”. In: *Advances in Neural Information Processing Systems (NeurIPS).* Vol. 36. 2023, pp. 40521–40541.

[175] François Rozet and Gilles Louppe. “Score-based Data Assimilation for a Two-Layer Quasi-Geostrophic Model”. In: *ArXiv* abs/2310.01853 (2023).

[176] Evan Ruzanski and V Chandrasekar. “Scale filtering for improved nowcasting performance in a high-resolution X-band radar network”. In: *IEEE transactions on geoscience and remote sensing* 49.6 (2011), pp. 2296–2307.

[177] Pavel Sakov and Peter R. Oke. “A Deterministic Formulation of the Ensemble Kalman Filter: An Alternative to Ensemble Square Root Filters”. In: *Tellus A: Dynamic Meteorology and Oceanography* 60.2 (2008), p. 361.

[178] Tim Salimans and Jonathan Ho. “Progressive Distillation for Fast Sampling of Diffusion Models”. In: *International Conference on Learning Representations.* 2022.

[179] S. Särkkä. *Bayesian Filtering and Smoothing.* Cambridge University Press, 2013.

[180] S. Särkkä and J. Hartikainen. “Infinite-dimensional Kalman filtering approach to spatio-temporal Gaussian process regression”. In: *Artificial Intelligence and Statistics*. PMLR. 2012, pp. 993–1001.

[181] S. Särkkä and A. Solin. *Applied Stochastic Differential Equations*. Cambridge University Press, 2019.

[182] S. Särkkä, A. Solin, and J. Hartikainen. “Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing: A look at Gaussian process regression through Kalman filtering”. In: *IEEE Signal Processing Magazine* 30.4 (2013), pp. 51–61.

[183] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. “The Graph Neural Network Model”. In: *IEEE Transactions on Neural Networks* 20.1 (2009), pp. 61–80.

[184] William E Schiesser. *The numerical method of lines: integration of partial differential equations*. Elsevier, 2012.

[185] Jonathan Schmidt, Philipp Hennig, Jörg Nick, and Filip Tronarp. “The Rank-Reduced Kalman Filter: Approximate Dynamical-Low-Rank Filtering In High Dimensions”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 36. 2023, pp. 61364–61376.

[186] Jonathan Schmidt, Luca Schmidt, Felix M. Strnad, Nicole Ludwig, and Philipp Hennig. “A Generative Framework for Probabilistic, Spatiotemporally Coherent Downscaling of Climate Simulation”. In: *npj Climate and Atmospheric Science* 8.1 (2025), p. 270.

[187] Luca Schmidt and Nicole Ludwig. “Wind Power Assessment based on Super-Resolution and Downscaling—A Comparison of Deep Learning Methods”. In: *arXiv preprint arXiv:2407.08259* (2024).

[188] Fred Schweppe. “Evaluation of likelihood functions for Gaussian signals”. In: *IEEE transactions on Information Theory* 11.1 (1965), pp. 61–70.

[189] Benjamin T. Schwertfeger. *btschwertfeger/python-cmethods: v2.3.0*. Version v2.3.0. 2024.

[190] Cheng Shen, Jinlin Zha, Zhibo Li, Cesar Azorin-Molina, Kaiqiang Deng, Lorenzo Minola, and Deliang Chen. “Evaluation of global terrestrial near-surface wind speed simulated by CMIP6 models and their future projections”. In: *Ann. N.Y. Acad. Sci.* 1518.1 (2022), pp. 249–263.

[191] Hui Shen, Jingxuan Zhang, Boning Xiong, Rui Hu, Shoufa Chen, Zhongwei Wan, Xin Wang, Yu Zhang, Zixuan Gong, Guangyin Bao, et al. “Efficient Diffusion Models: A Survey”. In: *Transactions on Machine Learning Research (TMLR)* (2025).

[192] DM Smith, R Eade, AA Scaife, L-P Caron, G Danabasoglu, TM DelSole, T Delworth, FJ Doblas-Reyes, NJ Dunstone, L Hermanson, et al. “Robust skill of decadal climate predictions”. In: *npj Climate and Atmospheric Science* 2.1 (2019), p. 13.

[193] Edward Snelson and Zoubin Ghahramani. “Sparse Gaussian processes using pseudo-inputs”. In: *Advances in Neural Information Processing Systems (NeurIPS)* 18 (2005), pp. 1257–1264.

[194] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. “Deep Unsupervised Learning using Nonequilibrium Thermodynamics”. In: *Proceedings of the 32nd International Conference on Machine Learning*. Vol. 37. Proceedings of Machine Learning Research. Lille, France, 2015, pp. 2256–2265.

[195] Arno Solin. “Stochastic Differential Equation Methods for Spatio-Temporal Gaussian Process Regression”. English. Doctoral thesis. Aalto University, School of Science, 2016, 68 + app. 72. ISBN: 978-952-60-6711-7 (electronic), 978-952-60-6710-0 (printed).

[196] Jiaming Song, Chenlin Meng, and Stefano Ermon. “Denoising Diffusion Implicit Models”. In: *International Conference on Learning Representations*. 2021.

[197] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. “Consistency Models”. In: *Proceedings of the 40th International Conference on Machine Learning*. Vol. 202. Proceedings of Machine Learning Research. 2023, pp. 32211–32252.

[198] Yang Song and Stefano Ermon. “Generative Modeling by Estimating Gradients of the Data Distribution”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 32. 2019.

[199] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. “Score-Based Generative Modeling through Stochastic Differential Equations”. In: *International Conference on Learning Representations*. 2021.

[200] Prakhar Srivastava, Ruihan Yang, Gavin Kerrigan, Gideon Dresdner, Jeremy J McGibbon, Christopher S. Bretherton, and Stephan Mandt. “Precipitation Downscaling with Spatiotemporal Video Diffusion”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2024.

[201] R.F. Stengel. *Optimal Control and Estimation*. Dover Publications, 1994.

[202] Walter A Strauss. *Partial differential equations: An introduction*. John Wiley & Sons, 2007.

[203] Felix Strnad, Jonathan Schmidt, Fabian Mockert, Philipp Hennig, and Nicole Ludwig. “Assessing the risk of future Dunkelflaute events for Germany using generative deep learning”. In: *arXiv preprint arXiv:2509.24788* (2025).

[204] A. M. Stuart. “Inverse problems: a Bayesian perspective”. In: *Acta numerica* 19 (2010), pp. 451–559.

[205] Sean Talts, Michael Betancourt, Daniel Simpson, Aki Vehtari, and Andrew Gelman. “Validating Bayesian inference algorithms with simulation-based calibration”. In: *arXiv preprint arXiv:1804.06788* (2018).

[206] Francisco J. Tapiador, Andrés Navarro, Raúl Moreno, José Luis Sánchez, and Eduardo García-Ortega. “Regional climate models: 30 years of dynamical downscaling”. In: *Atmos. Res.* 235 (2020), p. 104785.

[207] Will Tebbutt, Arno Solin, and Richard E. Turner. “Combining pseudo-point and state space approximations for sum-separable Gaussian processes”. In: *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*. Vol. 161. Proceedings of Machine Learning Research. 2021, pp. 1607–1617.

[208] Timera Energy. *Impact of German Dunkelflaute on Flex Asset Value*. `https://timera-energy.com/blog/impact-of-german-dunkelflaute-on-flex-asset-value/`. Accessed: 2025-07-21. 2025.

[209] M. K. Tippett, J. L. Anderson, C. H. Bishop, T. M. Hamill, and J. S. Whitaker. “Ensemble square root filters”. In: *Monthly weather review* 131.7 (2003), pp. 1485–1490.

[210] Michalis Titsias. “Variational learning of inducing variables in sparse Gaussian processes”. In: *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR. 2009, pp. 567–574.

[211] M. Todescato, A. Carron, R. Carli, G. Pillonetto, and L. Schenato. “Efficient spatio-temporal Gaussian regression via Kalman filtering”. In: *Automatica* 118 (2020), p. 109032.

[212] Elena Tomasi, Gabriele Franch, and Marco Cristoforetti. “Can AI be enabled to perform dynamical downscaling? A latent diffusion model to mimic kilometer-scale COSMO5.0_CLM9 simulations”. In: *Geosci. Model Dev.* 18.6 (2025), pp. 2051–2078.

[213] Alasdair Tran, Alexander Mathews, Lexing Xie, and Cheng Soon Ong. “Factorized Fourier Neural Operators”. In: *The Eleventh International Conference on Learning Representations*. 2023.

[214] Filip Tronarp, Nathanael Bosch, and Philipp Hennig. “Fenrir: Physics-Enhanced Regression for Initial Value Problems”. In: *Proceedings of the 39th International Conference on Machine Learning*. Vol. 162. Proceedings of Machine Learning Research. 2022, pp. 21776–21794.

[215] Filip Tronarp, Hans Kersting, Simo Särkkä, and Philipp Hennig. “Probabilistic solutions to ordinary differential equations as nonlinear Bayesian filtering: a new perspective”. In: *Statistics and Computing* 29.6 (2019), pp. 1297–1315.

[216] Joel A Tropp, Alp Yurtsever, Madeleine Udell, and Volkan Cevher. “Practical sketching algorithms for low-rank matrix approximation”. In: *SIAM Journal on Matrix Analysis and Applications* 38.4 (2017), pp. 1454–1485.

[217] Thomas Vandal, Evan Kodra, and Auroop R Ganguly. “Intercomparison of machine learning methods for statistical downscaling: the case of daily and extreme precipitation”. In: *Theoretical and Applied Climatology* 137 (2019), pp. 557–570.

[218] Sanjana Vijayshankar, Saleh Nabi, Ankush Chakrabarty, Piyush Grover, and Mouhacine Benosman. “Dynamic mode decomposition and robust estimation: Case study of a 2D turbulent Boussinesq flow”. In: *2020 American Control Conference (ACC)*. IEEE. 2020, pp. 2351–2356.

[219] Pascal Vincent. “A Connection Between Score Matching and Denoising Autoencoders”. In: *Neural Computation* 23.7 (2011), pp. 1661–1674.

[220] C. Volosciuk, D. Maraun, M. Vrac, and M. Widmann. “A combined statistical bias correction and stochastic downscaling method for precipitation”. In: *Hydrology and Earth System Sciences* 21.3 (2017), pp. 1693–1719.

[221] Z. Wang, E.P. Simoncelli, and A.C. Bovik. “Multiscale structural similarity for image quality assessment”. In: *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*. Vol. 2. 2003, 1398–1402 Vol.2.

[222] R. E. Wengert. “A simple automatic derivative evaluation program”. In: *Commun. ACM* 7.8 (1964), pp. 463464.

[223] D. S. Wilks. “Multisite downscaling of daily precipitation with a stochastic weather generator”. In: *Climate Research* 11.2 (1999), pp. 125–136.

[224] Christopher Williams and Matthias Seeger. “Using the Nyström method to speed up kernel machines”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2001.

[225] Christina Winkler, Paula Harder, and David Rolnick. “Climate Variable Downscaling with Conditional Normalizing Flows”. In: *arXiv preprint arXiv:2405.20719* (2024).

[226] Shangshang Yang, Congyi Nai, Xinyan Liu, Weidong Li, Jie Chao, Jingnan Wang, Leyi Wang, Xichen Li, Xi Chen, Bo Lu, Ziniu Xiao, Niklas Boers, Huiling Yuan, and Baoxiang Pan. “Generative assimilation and prediction for weather and climate”. In: *arXiv preprint arXiv:2503.03038* (2025).

[227] Qinsheng Zhang and Yongxin Chen. “Fast Sampling of Diffusion Models with Exponential Integrator”. In: *The Eleventh International Conference on Learning Representations*. 2023.

[228] Hongkai Zheng, Weili Nie, Arash Vahdat, Kamyar Azizzadenesheli, and Anima Anandkumar. “Fast Sampling of Diffusion Models via Operator Learning”. In: *Proceedings of the 40th International Conference on Machine Learning*. Vol. 202. Proceedings of Machine Learning Research. 2023, pp. 42390–42402.

[229] Jakob Zscheischler, Seth Westra, Bart J. J. M. van den Hurk, Sonia I. Seneviratne, Philip J. Ward, Andy Pitman, Amir AghaKouchak, David N. Bresch, Michael Leonard, Thomas Wahl, and Xuebin Zhang. “Future climate risk from compound events”. In: *Nature Climate Change* 8.6 (2018), pp. 469–477.

Note on the Use of Artificial Intelligence

Large language models (LLMs) and other artificial intelligence (AI) tools were used in creating this thesis. My (Jonathan Schmidt) personal use of AI tools is documented below and was discussed with my supervisor (Prof. Dr. Philipp Hennig).

TEXT

I used AI for identifying spelling and grammar errors and for refining wording and individual formulations. I translated the abstract to German with the help of *DeepL Translator*[1]. Early in the writing process, I discussed the thesis structure and storyline with AI chat interfaces. AI had no determining influence on the content or structure of the thesis. In particular, the text contains no full sentences or paragraphs that were AI-generated.

CODE

AI tools assissted in creating, editing, and documenting code for plots and figures (Python and TikZ). Figure 2.9 was primarily created using an AI coding agent, including the arXiv API queries and visualization. Most plots in Chapter 2 and Chapter 4 involved AI assistance. All AI-generated code was reviewed and edited for correctness and efficiency.

RESEARCH

The research presented was conducted without AI tools; AI supported literature search only. Research questions, project structures, and experimental designs were developed without AI assistance.[2] Evaluation, analysis, and interpretation of research findings was carried out without AI assistance. No idea-generation tools were used.

Tübingen, ______________________

Jonathan Schmidt (doctoral candidate)	Prof. Dr. Philipp Hennig (supervisor, first reviewer)

---

[1] `https://www.deepl.com/en/translator`

[2] Research was conducted collaboratively with co-authors.

PART IV

Appendix

A Appendix for Chapter 3

A.1 Proofs

A.1.1 Proof of Proposition 3.1

Proof. By standard results on Gaussian conditioning and marginalization

$$\mathrm{Z} \mid \mathrm{Y} \sim \mathcal{N}(K_{\mathrm{Z}}(\mathrm{Y} - H\mu), \Sigma_{\mathrm{Z}}), \tag{A.1a}$$

$$\mathrm{Y} \sim \mathcal{N}(H\mu, S), \tag{A.1b}$$

where

$$\Sigma_{\mathrm{Z}}^{-1} = I + (R^{-1/2} H \Pi^{1/2})^{\mathsf{T}} R^{-1/2} C \Pi^{1/2}, \tag{A.2a}$$

$$K_{\mathrm{Z}} = \Sigma_{\mathrm{Z}} (R^{-1/2} H \Pi^{1/2})^{\mathsf{T}} R^{-1/2}, \tag{A.2b}$$

$$R^{-1/2} S R^{-\mathsf{T}/2} = R^{-1/2} H \Pi^{1/2} (R^{-1/2} H \Pi^{1/2})^{\mathsf{T}} + I. \tag{A.2c}$$

Substituting for the singular value decomposition of $(R^{-1/2} H \Pi^{1/2})^{\mathsf{T}}$ gives the conditional covariance

$$\Sigma_{\mathrm{Z}} = \big(I + U D^2 U^{\mathsf{T}}\big)^{-1} = \big(U(I + D^2) U^{\mathsf{T}}\big)^{-1} = U(I + D^2)^{-1} U^{\mathsf{T}}, \tag{A.3}$$

the Kalman gain

$$K_{\mathrm{Z}} = U(I + D^2)^{-1} U^{\mathsf{T}} U D V^{\mathsf{T}} R^{-1/2} = U(I + D^2)^{-1} D V^{\mathsf{T}} R^{-1/2}, \tag{A.4}$$

and the marginal measurement covariance matrix

$$R^{-1/2} S R^{-\mathsf{T}/2} = V D^2 V^{\mathsf{T}} + I. \tag{A.5}$$

This gives the result on marginalization and conditioning. To obtain the expression for the marginal likelihood, which is given by

$$\mathcal{N}(\mathrm{Y}; H\mu, S) = -\frac{m}{2} \log 2\pi - \frac{1}{2} \log |S| - \frac{1}{2} (\mathrm{Y} - H\mu)^{\mathsf{T}} S^{-1} (\mathrm{Y} - H\mu). \tag{A.6}$$

The log determinant is given by

$$\log |S| = \log \Big| R^{1/2} (V D^2 V^{\mathsf{T}} + I) R^{\mathsf{T}/2} \Big| = 2 \log \Big| R^{1/2} \Big| + \log \Big| V D^2 V^{\mathsf{T}} + I \Big|. \tag{A.7}$$

Let V_k be the k-th column vector of V then it is an eigenvector of $VD^2V^\mathsf{T} + I$ with eigenvalue $D_{kk}^2 + 1$. Furthermore, any vector in the orthogonal complement to the column space of V is also an eigenvector with eigenvalue 1. Therefore,

$$\log \left| VD^2V^\mathsf{T} + I \right| = \sum_{k=1}^{r} \log\bigl(D_{kk}^2 + 1\bigr). \tag{A.8}$$

It remains to obtain the desired expression for the quadratic form. Start by inserting the expression for S

$$\begin{aligned}(\mathrm{Y} - H\mu)^\mathsf{T} S^{-1} (\mathrm{Y} - H\mu) &= (\mathrm{Y} - H\mu)^\mathsf{T} R^{-\mathsf{T}/2} (VD^2V^\mathsf{T} + I)^{-1} R^{-1/2} (\mathrm{Y} - H\mu) \\ &= e^\mathsf{T} (VD^2V^\mathsf{T} + I)^{-1} e.\end{aligned} \tag{A.9}$$

Now $e^\mathsf{T}(VD^2V^\mathsf{T} + I)^{-1}e = e^\mathsf{T}b$, where we define the vector b such that

$$(VD^2V^\mathsf{T} + I)b = e. \tag{A.10}$$

Multiplying from the left by V^T gives

$$(D^2 + I)V^\mathsf{T}b = V^\mathsf{T}e, \tag{A.11}$$

therefore,

$$V^\mathsf{T}b = (D^2 + I)^{-1}V^\mathsf{T}e. \tag{A.12}$$

Inserting this in the original definition of b gives

$$VD^2(D^2 + I)^{-1}V^\mathsf{T}e + b = e, \tag{A.13}$$

from which it follows that

$$e^\mathsf{T}(VD^2V^\mathsf{T} + I)^{-1}e = e^\mathsf{T}b = \|e\|^2 - e^\mathsf{T}VD(D^2 + I)^{-1}DV^\mathsf{T}e. \tag{A.14}$$

□

A.1.2 Proof of Proposition 3.3

In the following, the computational complexity of the low-rank filtering recursion is analyzed in detail, which proves Proposition 3.3 and Corollary 3.4. In the best case, the cost of the proposed filtering algorithms scales linearly in the state dimension n and the measurement dimension m. For this to hold, we assume that

(a) the maps $x \mapsto Ax$, $x \mapsto \Phi x$, and $x \mapsto BB^\mathsf{T}x$ can be evaluated in $\mathcal{O}(n)$,

(b) the map $x \mapsto Hx$ can be evaluated in $\mathcal{O}(m)$, and

(c) the map $x \mapsto R^{-1/2}x$ and the log-determinant $\log \left| R^{1/2} \right|$ can be evaluated in $\mathcal{O}(m)$.

Table A.1: Time complexity of reduced-rank filtering: The prediction step

Step			
Eq.	Operation	Best case	Worst case
(A.15)	$AK(t)$	$\mathcal{O}(nr)$	$\mathcal{O}(n^2r)$
	$K(t)(U_0^\mathsf{T}(A^\mathsf{T}U_0))$	$\mathcal{O}(nr^2)$	$\mathcal{O}(n^2r + nr^2)$
	$BB^\mathsf{T}U_0$	$\mathcal{O}(nr)$	$\mathcal{O}(n^2r)$
(A.16)	$\mathrm{QR}(K(t+h))$	$\mathcal{O}(nr^2)$	$\mathcal{O}(nr^2)$
(A.17)	$M = U_h^\mathsf{T}U_0$	$\mathcal{O}(nr^2)$	$\mathcal{O}(nr^2)$
(A.18)	$D(t_0) = MD_0M^\mathsf{T}$	$\mathcal{O}(r^3)$	$\mathcal{O}(r^3)$
(A.19)	$U_h^\mathsf{T}(A(U_hD(t)))$	$\mathcal{O}(nr^2)$	$\mathcal{O}(n^2r + nr^2)$
	$(D(t)U_h^\mathsf{T})(A^\mathsf{T}U_h)$	$\mathcal{O}(nr^2)$	$\mathcal{O}(n^2r + nr^2)$
	$U_h^\mathsf{T}(BB^\mathsf{T}U_h)$	$\mathcal{O}(nr^2)$	$\mathcal{O}(n^2r + nr^2)$
(A.20)	$D_l = D_l^{1/2}D_l^{\mathsf{T}/2}$	$\mathcal{O}(r^3)$	$\mathcal{O}(r^3)$
(A.21)	$\Phi_l\mu_{l-1}$	$\mathcal{O}(n)$	$\mathcal{O}(n^2)$
(A.22)	$\Phi_l\Sigma_{l-1}^{1/2}$	$\mathcal{O}(nr)$	$\mathcal{O}(n^2r)$
	$(\Phi_l\Sigma_{l-1}^{1/2} \quad Q_l^{1/2}) \approx \tilde{U}_l\tilde{D}_l\tilde{V}_l^\mathsf{T}$	$\mathcal{O}(nr^2)$	$\mathcal{O}(nr^2)$

We refer to the situation in which (a) or (b) do not apply as the “worst case”. Assumption (c) is taken for granted as the measurement-noise covariance R is often a diagonal matrix, which implies that sensor errors are uncorrelated. This is not only realistic but also commonly imposed in modelling.

Proof. The best-case cost of the approximate filtering scales linear in the state dimension n and the measurement dimension m. In the worst case, the complexity scales quadratically in n and m.

PREDICTION

We begin by analyzing the cost for the approximate integration of the low-rank process-noise covariance $Q^{1/2}$. This amounts to the cost of the dynamical low-rank approximation (DLRA) algorithm for a symmetric Lyapunov equation. Section A.3 gives a detailed description of how this algorithm is used as part of the RRKF recursions. Let the matrix-valued flow field be $F(Q) = AQ + QA^\mathsf{T} + BB^\mathsf{T}$. Given an initial factorization $Q_0 \approx \Upsilon_0 = U_0D_0^2U_0^\mathsf{T}$, with $U_0 \in \mathbb{R}^{n\times r}$ and $D_0 \in \mathbb{R}^{r\times r}$, the cost for integrating this matrix equation using DLRA is as follows:

1. **K-step**
    a) Flow-field evaluation

$$F(t, K(t)U_0^\mathsf{T})U_0 = A(K(t)U_0^\mathsf{T})U_0 + K(t)U_0^\mathsf{T}A^\mathsf{T}U_0 + BB^\mathsf{T}U_0 \tag{A.15a}$$

$$= AK(t) + K(t)(U_0^\mathsf{T}(A^\mathsf{T}U_0)) + BB^\mathsf{T}U_0, \tag{A.15b}$$

b) QR factorization of tall $n \times r$ matrix

$$\mathrm{QR}(K(t+h)), \tag{A.16}$$

c) Compute

$$M = U_h^\mathsf{T} U_0. \tag{A.17}$$

2. **S-step**

a) Compute initial $D(t_0)$

$$D(t_0) = M D_0 M^\mathsf{T}, \tag{A.18}$$

b) Flow-field evaluation

$$U_h^\mathsf{T} F(t, U_h D(t) U_h^\mathsf{T}) U_h = U_h^\mathsf{T} \left(A U_h D(t) U_h^\mathsf{T} + U_h D(t) U_h^\mathsf{T} A^\mathsf{T} + B B^\mathsf{T} \right) U_h \tag{A.19a}$$

$$= U_h^\mathsf{T} (A(U_h D(t))) + (D(t) U_h^\mathsf{T})(A^\mathsf{T} U_h) + U_h^\mathsf{T} (B B^\mathsf{T} U_h). \tag{A.19b}$$

The above steps 1. and 2. are repeated according to how many DLRA integration steps are performed in the respective prediction step. Therefore, their cost has to be multiplied by that (typically small) constant.

3. Using DLRA integration we obtain a low-rank factorization of the process-noise covariance matrix $Q_l \approx \Upsilon_l = U_l D_l U_l^2$. It remains to compute a matrix square root $Q_l^{1/2} = U_l D_l^{1/2}$ and thus a matrix square root of $D_l \in \mathbb{R}^{r \times r}$

$$D_l = D_l^{1/2} D_l^{\mathsf{T}/2}. \tag{A.20}$$

Now, having obtained $Q_l^{1/2}$, we proceed to

4. predict the mean

$$\mu_l^- = \Phi_l \mu_{l-1}, \tag{A.21}$$

5. and the low-rank factor $\Pi_l^{1/2}$ of the predicted covariance matrix. This amounts to building the rank-$2r$ square-root factor and truncating it at its r-th largest singular value (i.e. , a truncated SVD of a tall $n \times 2r$ matrix):

$$\begin{pmatrix} \Phi_l \Sigma_{l-1}^{1/2} & Q_l^{1/2} \end{pmatrix} \approx \tilde{U}_l \tilde{D}_l \tilde{V}_l^\mathsf{T}. \tag{A.22}$$

The computational complexities of the respective steps can be found in Table A.1.

Table A.2: Time complexity of reduced-rank filtering: The correction step

Step			
Eq.	Operation	Best case	Worst case
(A.23)	$R^{-1/2}(H\Pi_l^{1/2})$	$\mathcal{O}(mr)$	$\mathcal{O}(m^2r)$
	$(R^{-1/2}H\Pi_l^{1/2})^\top = U_l D_l V_l^\top$	$\mathcal{O}(mr^2)$	$\mathcal{O}(mr^2)$
(A.24)	$e_l = R^{-1/2}(\mathrm{Y}_l - H\mu_l^-)$	$\mathcal{O}(m)$	$\mathcal{O}(mn + m^2)$
(A.25)	$\Pi_l^{1/2}(U_l((I + D_l^2)^{-1}(D_l(V_l^\top e_l))))$	$\mathcal{O}(mr + nr^2)$	$\mathcal{O}(mr + nr^2)$
(A.26)	$\Pi_l^{1/2}(U_l(I + D_l^2)^{-1/2})$	$\mathcal{O}(nr^2)$	$\mathcal{O}(nr^2)$
(3.23)	$\log\left\lvert R^{1/2}\right\rvert$	$\mathcal{O}(m)$	$\mathcal{O}(m)$
	$\sum_{k=1}^{r}\log\big((D_l)_{kk}^2 + 1\big)$	$\mathcal{O}(r)$	$\mathcal{O}(r)$
	$\lVert e_l\rVert^2$	$\mathcal{O}(r)$	$\mathcal{O}(r)$
	$(e_l^\top V_l)(D_l((D_l^2 + I)^{-1}D_l))(V_l^\top e_l)$	$\mathcal{O}(mr)$	$\mathcal{O}(mr)$

CORRECTION STEP

Now, the cost of the correction step is analyzed. First, we compute an SVD of a wide $r \times m$ matrix

$$(R^{-1/2}H\Pi_l^{1/2})^\top = U_l D_l V_l^\top. \tag{A.23}$$

Given this decomposition and the whitened residual

$$e_l = R^{-1/2}(\mathrm{Y}_l - H\mu_l^-), \tag{A.24}$$

we proceed to

1. update the mean

$$\Delta\mu_l = K_l e_l = \Pi_l^{1/2}(U_l((I + D_l^2)^{-1}(D_l(V_l^\top e_l)))), \tag{A.25}$$

2. and compute the low-rank factor of the filtering covariance

$$\Sigma_l^{1/2} = \Pi_l^{1/2}(U_l(I + D_l^2)^{-1/2}). \tag{A.26}$$

Finally, the marginal predictive log-likelihood $\log p(\mathrm{Y}_l \mid \mathrm{Y}_{1:l-1})$ is computed as in Equation (3.23).

The computational complexities of the respective steps can be found in Table A.2. Together, Tables A.1 and A.2 prove the stated asymptotic complexities of the low-rank filtering recursion.

□

A.1.3 Proof of Proposition 3.5

This section analyzes the computational complexity of the low-rank smoothing recursion in detail, which proves Proposition 3.5.

Proof. The best-case cost of the approximate smoothing scales linear in the state dimension n. In the worst case, the complexity is quadratic in n.

Approximate backwards kernel

Factorizations of $\Sigma_{l-1}^{1/2}$ and $\Pi_l^{1/2}$ are already given from filtering. For the smoothing gain, it remains to compute

$$\Gamma_l = \Sigma_l^{\mathsf{T}/2}(\Phi_{l+1}^{\mathsf{T}}(\Pi_{l+1}^{1/2})^{\dagger}). \tag{A.27}$$

Then, we proceed to compute the shift vector

$$v_l = \mu_l - G_l \mu_{l+1}^{-} = \mu_l - \Sigma_l^{1/2}(\Gamma_l((\Pi_{l+1}^{\mathsf{T}/2})^{\dagger}\mu_{l+1}^{-})), \tag{A.28}$$

and the low-rank factor $P_l^{1/2}$ of the backwards-process-noise covariance matrix. Therefore, a truncated SVD of a tall $n \times 2r$ matrix is computed:

$$\begin{pmatrix} (I - G_l \Phi_{l+1})\Sigma_l^{1/2} & G_l Q_{l+1}^{1/2} \end{pmatrix} \approx \widehat{U}_l \widehat{D}_l \widehat{V}_l^{\mathsf{T}}. \tag{A.29}$$

Backwards prediction (smoothing)

Smoothing amounts to consecutive predictions with the backwards kernel. For this prediction, the low-rank factor of the process-noise covariance matrix does not come from DLRA, but can be directly computed from the smoothing gain and the process-noise covariance matrix (see Equations (3.29) and (3.30)).

The smoothing mean is therefore given as

$$\xi_l = G_l \xi_{l+1} + v_l = \Sigma_l^{1/2}(\Gamma_l((\Pi_{l+1}^{\mathsf{T}/2})^{\dagger}\xi_{l+1})) + v_l, \tag{A.30}$$

and the low-rank factor of the smoothing covariance as $\Lambda_l^{1/2} = \bar{U}_l \bar{D}_l$, where

$$\begin{pmatrix} G_l \Lambda_{l+1}^{1/2} & P_l^{1/2} \end{pmatrix} \approx \bar{U}_l \bar{D}_l \bar{V}_l^{\mathsf{T}} \tag{A.31}$$

is the truncated SVD of a tall $n \times 2r$ matrix.

The computational complexities of the respective steps can be found in Table A.3, which shows that—in the best case—the cost of approximate low-rank smoothing never exceeds $\mathcal{O}(nr^2 + r^3)$, as stated by Proposition 3.5. It also shows that in the worst case, the complexity is quadratic in n. □

Table A.3: Time complexity of reduced-rank smoothing

Step			
Eq.	Operation	Best case	Worst case
(A.27)	$\Gamma_l = \Sigma_{l-1}^{\mathsf{T}/2}(\Phi_l^{\mathsf{T}}(\Pi_l^{1/2})^{\dagger})$	$\mathcal{O}(nr^2)$	$\mathcal{O}(n^2r + nr^2)$
(A.28)	$G_l\mu_{l+1}^{-} = \Sigma_l^{1/2}(\Gamma_l((\Pi_{l+1}^{\mathsf{T}/2})^{\dagger}\mu_{l+1}^{-}))$	$\mathcal{O}(nr + r^2)$	$\mathcal{O}(nr + r^2)$
(A.29)	$G_l\Phi_{l+1}\Sigma_l^{1/2} = \Sigma_l^{1/2}(\Gamma_l((\Pi_{l+1}^{\mathsf{T}/2})^{\dagger}(\Phi_{l+1}\Sigma_l^{1/2})))$	$\mathcal{O}(nr^2 + r^3)$	$\mathcal{O}(n^2r + nr^2 + r^3)$
	$G_lQ_l^{1/2} = \Sigma_{l-1}^{1/2}(\Gamma_l((\Pi_l^{\mathsf{T}/2})^{\dagger}Q_l^{1/2}))$	$\mathcal{O}(nr^2 + r^3)$	$\mathcal{O}(nr^2 + r^3)$
	$((I - G_l\Phi_{l+1})\Sigma_l^{1/2} \quad G_lQ_{l+1}^{1/2}) \approx \widehat{U}_l\widehat{D}_l\widehat{V}_l^{\mathsf{T}}$	$\mathcal{O}(nr^2)$	$\mathcal{O}(nr^2)$
(A.30)	$G_l\xi_l = \Sigma_{l-1}^{1/2}(\Gamma_l((\Pi_l^{\mathsf{T}/2})^{\dagger}\xi_l))$	$\mathcal{O}(nr + r^2)$	$\mathcal{O}(nr + r^2)$
(A.31)	$G_l\Lambda_{l+1}^{1/2} = \Sigma_l^{1/2}(\Gamma_l((\Pi_{l+1}^{\mathsf{T}/2})^{\dagger}\Lambda_{l+1}^{1/2}))$	$\mathcal{O}(nr^2 + r^3)$	$\mathcal{O}(nr^2 + r^3)$
	$(G_l\Lambda_{l+1}^{1/2} \quad P_l^{1/2}) \approx \bar{U}_l\bar{D}_l\bar{V}_l^{\mathsf{T}}$	$\mathcal{O}(nr^2)$	$\mathcal{O}(nr^2)$

A.2 Efficient inference for the case r > m

This section gives an inference scheme that can be used in the correction step in case the low-rank dimension r exceeds the measurement dimension m at a time point t_l. This is not generally the designated use case of the algorithm, since in most applications, we assume $r \ll m \leq n$. However, if at a given time point t_l, there are less than usual measurements available (e.g. due to missing data at that time), it is still useful to have an inference scheme in place for this case. This is provided by the following proposition.

Proposition A.1 *Let $m < r \leq n$ and $\Pi_l^{1/2} \in \mathbb{R}^{n \times r}$ and $R^{1/2} \in \mathbb{R}^{m \times m}$. An approximate update is then obtained according to the standard Kalman filter update rules:*

$$S_l = H\Pi_l^{1/2}(H\Pi_l^{1/2})^{\mathsf{T}} + R^{1/2}R^{\mathsf{T}/2}, \tag{A.32a}$$

$$K_l = \Pi_l^{1/2}(H\Pi_l^{1/2})^{\mathsf{T}}S^{\dagger}, \tag{A.32b}$$

$$\Delta\mu_l = K_l(\mathrm{Y}_l - H\mu_l), \tag{A.32c}$$

$$\Sigma_l = \Pi_l - K_lS_lK_l^{\mathsf{T}}. \tag{A.32d}$$

The marginal measurement covariance, S_l, is obtained by a singular value decomposition

$$U_l^sD_l^s(V_l^s)^{\mathsf{T}} = \Big(H\Pi_l^{1/2} \quad R^{1/2}\Big), \tag{A.33a}$$

$$S = U_l^s(D_l^s)^2(U_l^s)^{\mathsf{T}}, \tag{A.33b}$$

where the matrix on the right-hand side of the first equation is $m \times (r+m)$. Hence, $U_l^s \in \mathbb{R}^{m \times (r+m)}$, and $D_l^s, V_l^s \in \mathbb{R}^{(r+m)\times(r+m)}$. The Moore–Penrose pseudoinverse $S_l^{\dagger}$ is then given by

$$S_l^{\dagger} = U_l^s(D_l^s)^{-2}(U_l^s)^{\mathsf{T}}. \tag{A.34}$$

The gain matrix is thus obtained as

$$\widetilde{K}_l = (H\Pi_l^{1/2})^\mathsf{T} U^s (D^s)^{-1}, \tag{A.35a}$$

$$K_l = \Pi_l^{1/2} \widetilde{K}_l (D^s)^{-1} (U^s)^\mathsf{T}, \tag{A.35b}$$

where $\widetilde{K}_l \in \mathbb{R}^{r\times(r+m)}$*. The updated covariance is obtained by*

$$\begin{aligned} \Sigma_l &= \Pi_l - K_l S_l K_l^\mathsf{T} \\ &= \Pi_l - \Pi_l^{1/2} \widetilde{K}_l (D_l^s)^{-1} (U_l^s)^\mathsf{T} U_l^s D_l^s (U_l^s)^\mathsf{T} (\Pi_l^{1/2} \widetilde{K}_l (D_l^s)^{-1} (U_l^s)^\mathsf{T})^\mathsf{T} \\ &= \Pi_l - \Pi_l^{1/2} \widetilde{K}_l (\Pi_l^{1/2} \widetilde{K}_l)^\mathsf{T} \\ &= \Pi_l^{1/2} \big(I - \widetilde{K}_l \widetilde{K}_l^\mathsf{T}\big) \Pi_l^{\mathsf{T}/2}. \end{aligned} \tag{A.36}$$

Consider the singular value decomposition of $\widetilde{K}_l$

$$\widetilde{K}_l = U_l^k D_l^k (V_l^k)^\mathsf{T}, \tag{A.37}$$

where $U_l^k, D_l^k \in \mathbb{R}^{r\times r}$ *and* $V_l^k \in \mathbb{R}^{r\times r+m}$*. The covariance update in low-rank form is then given by*

$$\begin{aligned} \Sigma^{1/2} &= \Pi^{1/2} \big(I - U_l^k (D_l^k)^2 (U_l^k)^\mathsf{T}\big)^{1/2}, \\ &= \Pi^{1/2} \big(U_l^k (U_l^k)^\mathsf{T} - U_l^k (D_l^k)^2 (U_l^k)^\mathsf{T}\big)^{1/2}, \\ &= \Pi^{1/2} U_l^k \big(I - (D_l^k)^2\big)^{1/2}. \end{aligned} \tag{A.38}$$

The marginal predictive log likelihood $p(\mathrm{Y}_l \mid \mathrm{Y}_{1:l-1}) = \mathcal{N}\big(H\mu_l^-, S_l\big)$ *can be cheaply evaluated given the factorization of* S_l *from Equation (A.33).*

A.3 Dynamical-low-rank approximation algorithm for Lyapunov equations

Following Ceruti et al. [36], we give the procedure for one DLRA integration step, adapted to our specific case of a symmetric Lyapunov equation

$$F(t, Q(t)) = AQ(t) + Q(t)A^\mathsf{T} + B(t)B(t)^\mathsf{T}. \tag{A.39}$$

Let an initial rank-r factorization $Q_0 \approx \Upsilon_0 = U_0 D_0 U_0^\mathsf{T}$, with an orthogonal matrix $U_0 \in \mathbb{R}^{n\times r}$ and a symmetric matrix $D_0 \in \mathbb{R}^{r\times r}$, be given at time t_0. For a temporal step size h, a rank-r factorization $Q(t+h) \approx \Upsilon_h = U_h D_h U_h^\mathsf{T}$ at the next integration step $t_0 + h$ is computed as follows.

K-step: Update $U_0 \to U_h$.
Integrate from $t = t_0$ to $t_0 + h$ the $n \times r$ matrix differential equation

$$\dot{K}(t) = F(t, K(t)U_0^\mathsf{T})U_0, \qquad K(t_0) = U_0 D_0. \tag{A.40}$$

Orthogonalize $K(t+h)$ by computing a QR decomposition, yielding the orthogonal matrix U_h. Then, compute the $r \times r$ matrix $M = U_h^\mathsf{T} U_0$.

S-step: Update $D_0 \to D_h$.
Integrate from $t = t_0$ to $t_0 + h$ the $r \times r$ matrix differential equation

$$\dot{D}(t) = U_h^\mathsf{T} F(t, U_h D(t) U_h^\mathsf{T}) U_h, \qquad D(t_0) = M D_0 M^\mathsf{T}, \tag{A.41}$$

and set $D_h = D(t+h)$

> **Remark A.1** The terms "K-step" and "S-step" can be confusing due to conflicting notation conventions in the Kalman-filtering and dynamical-low-rank-approximation literature, respectively. The matrix K and the term "S-step" are not related to the Kalman gain K and the marginal measurement covariance matrix S from the Kalman filter update step described in Sections A.2 and 3.3.2.

The $n \times r$ and $r \times r$ matrix differential equations in the K-step and S-step can be solved using a standard numerical integrator, e.g., a Runge-Kutta method or an exponential integrator.

It is useful to note that the matrix equation in the S-step itself is a symmetric Lyapunov equation:

$$\begin{aligned}
\dot{D}(t) &= U_h^\mathsf{T} F(t, U_h D(t) U_h^\mathsf{T}) U_h && \text{(A.42a)}\\
&= U_h^\mathsf{T} \Big(A U_h D(t) U_h^\mathsf{T} + U_h D(t) U_h^\mathsf{T} A^\mathsf{T} + B B^\mathsf{T} \Big) U_h && \text{(A.42b)}\\
&= (U_h^\mathsf{T} A U_h) D(t) + D(t) (U_h^\mathsf{T} A U_h)^\mathsf{T} + U_h^\mathsf{T} B B^\mathsf{T} U_h && \text{(A.42c)}\\
&= A_D D(t) + D(t) A_D^\mathsf{T} + B_D B_D^\mathsf{T}, && \text{(A.42d)}
\end{aligned}$$

where the final step defines the parameters of the Lyapunov equation for $D(t)$ as $A_D = U_h^\mathsf{T} A U_h$ and $B_D = U_h^\mathsf{T} B$. Since $D(t) \in \mathbb{R}^{r \times r}$ is small, this equation can be solved exactly with little computational cost using matrix-fraction decomposition [201, 10].

A.4 Details on experimental setups

This section gives more details regarding the experimental setups used in Section 3.4. For all results of the proposed RRKF algorithm, only a single DLRA step was used in the prediction step to compute the low-rank factorization of the process noise covariance matrix $Q^{1/2}$.

A.4.1 Linear advection model

This section provides more details on the data assimilation setting that is considered in Section 3.4.1.

We consider a spatial grid of $n = 1024$ uniformly spaced points and a temporal grid of 800 uniformly spaced time points. Unit step sizes $\Delta t = \Delta x = 1$ are assumed on the spatiotemporal grid. To generate an initial ground-truth state we sample an initial sinusoidal curve $\psi(0)$ according to the description by Sakov et al. [177]

$$\psi_i(0) = \sum_{k=0}^{25} a_k \sin\left(\frac{2\pi k}{1000} i + \varphi_k\right), \tag{A.43}$$

where $i = 1, \ldots, 1024$ is an index into the spatial grid and $a_k \sim \mathrm{Unif}(0, 1)$, $\varphi_k \sim \mathrm{Unif}(0, 2\pi)$. The initial state is normalized as described by Sakov et al. [177]. To generate a ground-truth trajectory from this initial state, the linear-advection dynamics $\frac{\partial \psi}{\partial t} = -\alpha \frac{\partial \psi}{\partial x}$ are simulated on the finite spatial grid. We assume constant unit velocity $\alpha = 1$ and periodic boundary conditions in space. As data, 10 equidistant state components are observed every 5 time steps. Each observation is corrupted by additive Gaussian noise.

An initial ensemble of size $r = n$ is built by successive sampling according to Equation (A.43). The exact sampling process follows the more detailed description by Evensen [61, Section 3.1]. From this $\mathbb{R}^{n \times n}$ ensemble matrix, the sample covariance matrix is computed and used as the initial covariance for the Kalman filter. The initial factorization of the RRKF is obtained by a spectral decomposition of the sample covariance matrix truncated at the r-th largest eigenvalue. Selecting the r first ensembles serves as an initial ensemble for the EnKF and the ETKF.

Figure 3.2a shows the deviations of the individual low-rank approximations to the optimal KF estimate on the described data assimilation problem.

A.4.2 London air-quality regression

The experimental setup and the data used in Section 3.4.2 is provided by Hamelijnck et al. [84]. The model is selected via the log-likelihood estimation of the *exact* Kalman filter. The RMSE of the Kalman filter mean to the test data is 9.96791 (cf. Hamelijnck et al. [84]). For $r = n$ the RRKF also obtains this RMSE up to numerical error, as shown in Figure 3.2b.

A.4.3 Spatiotemporal Matérn process with varying spatial lengthscale

Section 3.4.3 evaluates the low-rank approximation quality in approximate on-model spatiotemporal GP regression with varying strength of interaction between the state components. In this experiment, we consider a spatiotemporal GP with separable covariance structure, as described in Section 3.4.3 with hand-picked hyperparameters. The time domain is a uniform grid on the interval $[0.1, 10]$ with step size $\Delta t = 0.1$. The spatial domain is a uniform grid on $[0, 2] \times [0, 2]$ with step size $\Delta x = 0.1$.

Over the experiment, the spatial lengthscale is varied in order to evaluate low-rank approximation quality given how much correlation between the respective state components is encoded in the prior. The smaller the spatial lengthscale, the less interactions between the spatial point and the worse the low-rank approximation is expected to be for small r. We test the values $\ell_x \in \{0.01, 0.1, 0.25, 1.0\}$. For each of those values, we proceed as follows. First, a realization of the prior dynamics is drawn from the spatiotemporal Matérn process. At every temporal grid point, the entire state vector of the prior draw at that point is corrupted by additive Gaussian noise to generate data. Then, the Kalman filter estimate and the respective low-rank filtering estimates are computed for increasing values of r. Figure 3.3 shows for increasing spatial lengthscales and varying r (i) the resulting RMSE of the approximate filter means to the Kalman filter mean and (ii) the time-averaged Frobenius distance between the approximate filter covariance matrices to the Kalman filter covariance matrix.

A.4.4 Runtime

This section details the respective experimental setups used to evaluate the best-case and worst-case computational complexity of the approximate low-rank filters with respect to the state dimension n, as described in Section 3.4.4. We analyze the asymptotic computational complexity with respect to the state dimension n both for the linear-in-n case (Figure 3.4, left) and for the quadratic-in-n case (Figure 3.4, right). The elapsed wall time is always measured using the BenchmarkTools.jl software package[1] with default settings, which computes sample statistics over multiple runs automatically to prevent distortions by background processes. The setups for both analyzed cases differ only marginally from already encountered setups.

LINEAR-IN-n CASE

We use a setting that is similar to what is described in Sections A.4.1 and 3.4.1. The state dimension is varied in order to evaluate computational complexity with respect to n. The measurement dimension is fixed at $m = 100$ and the low-rank dimension at $r = 5$. The temporal domain is a grid on the interval $[0, 100]$ with unit step size $\Delta t = 1$. Noisy observations are generated from the ground truth trajectory at every 5 time points.

The discretized linear advection dynamics with periodic boundary conditions amount to multiplication with a circulant matrix. The shift direction depends on the advection velocity, which we chose as $\alpha = 1$. Multiplication with a circulant matrix can be carried out in linear complexity via multiplication in the Fourier space. Since there is no process noise assumed, the matrix $BB^\intercal$ is the zero matrix. Hence, Assumption 3.1 is satisfied by the dynamics model.

The measurement operator selects $m = 100$ uniformly spaced spatial points from the state vector and the measurement noise covariance matrix is a diagonal matrix. Hence, the measurement model fulfills Assumptions 3.2 and 3.3.

---

[1] `https://github.com/JuliaCI/BenchmarkTools.jl`

QUADRATIC-IN-n CASE

Here, we use a setting that is similar to what is described in Sections A.4.3 and 3.4.3. The measurement dimension is fixed at $m = 100$ and the low-rank dimension at $r = 5$. The temporal domain is a grid on the interval $[0, 10]$ with step size $\Delta t = 0.1$. The spatial domain are n equidistant points in an interval in $\mathbb{R}$, where n varies. The generated data are noisy observations from a ground-truth realization of the prior at 20 time points, each measuring $m = 100$ state components. We evaluate the runtime of solving a spatiotemporal regression problem with this prior and the generated data.

Assumption 3.1 is *not* fulfilled by the dynamics model in this setting, in that evaluating the map $x \mapsto BB^\mathsf{T} x$ costs $\mathcal{O}(n^2)$. By modeling spatial diffusion with a dense spatial kernel matrix, the inhomogeneity in the Lyapunov equation for the process-noise covariance matrix is also a dense matrix. Concretely, $BB^\mathsf{T} = (\widetilde{B}\mathcal{K}_x^{1/2})(\widetilde{B}\mathcal{K}_x^{1/2})^\mathsf{T}$, where $\widetilde{B}$ is the temporal dispersion matrix and $\mathcal{K}_x$ denotes the spatial kernel matrix. Since $\mathcal{K}_x$ is dense, so is BB^T, which violates a part of Assumption 3.1. The cost of the low-rank prediction step thus scales quadratically in the state dimension n.

The observation model is a sparse selection operator that projects the state onto $m = 100$ points and the measurement noise covariance is a diagonal matrix. Hence, Assumptions 3.2 and 3.3 are still fulfilled by the observation model.

Figure 3.4 demonstrates that the theoretical complexities are empirically verified.

A.4.5 Large-scale spatiotemporal GP regression on rainfall data

This section provides further details on the large-scale low-rank approximation to spatiotemporal GP regression, described in Section 3.4.5.

We select the prior model via the log-likelihood estimate of the RRKF, as given in Equation (3.23). For the model selection, the data set is distinct from the data used in the filtering/smoothing problem (Figure 3.5). We use the time period from 25 January, 2010 through 5 March, 2010 and each data point is downsampled to a lower spatial resolution using cubic-spline interpolation. This reduces the state dimension to $n = 4350$ spatial points during model selection. Further, for model selection the low-rank dimension is set to $r = 200$.

A.5 Z-scores of (approximate) Gaussian state estimates

This sections investigates a limitation of the proposed algorithm, which is addressed in Section 3.5. When truncating covariance information between the state components, we expect this to be reflected by a *higher* uncertainty in the resulting estimate. However, the algorithm does not account for that missing information and tends to return overconfident estimates for small values of r.

We analyze the distribution of the Z-scores of Gaussian state estimates on a Gaussian model. We expect the absolute values of the Z-scores to be distributed according to a Chi(1) distribution. A spatiotemporal GP regression problem in an on-model setting, similar to the

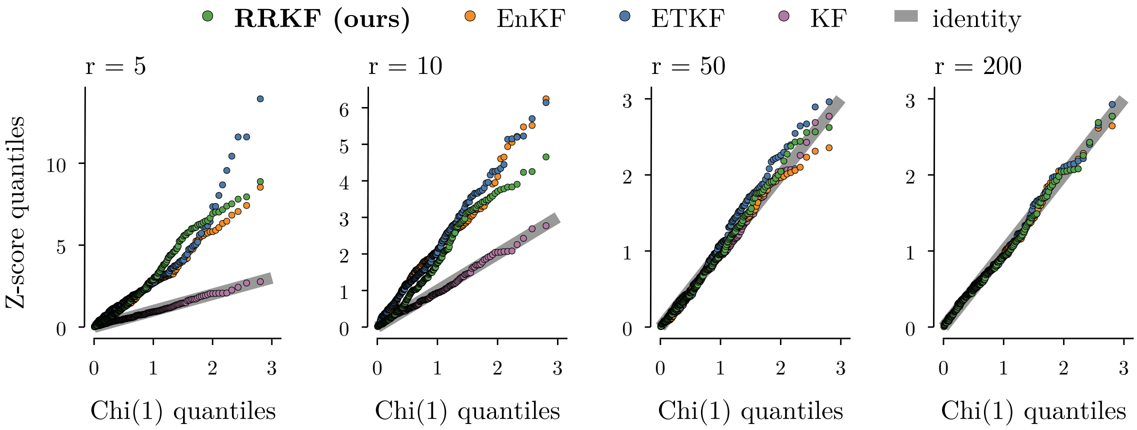


Figure A.1: **Z-score distribution for the low-rank filters with varying low-rank dimensions r.** In this on-model GP regression problem, we assume the Z-scores to be Chi(1) distributed. For the approximate low-rank filters, the Z-score distribution has too much mass in the high regimes, indicating overconfident estimates. For $r = n$, the Z-scores of the RRKF and the KF align.

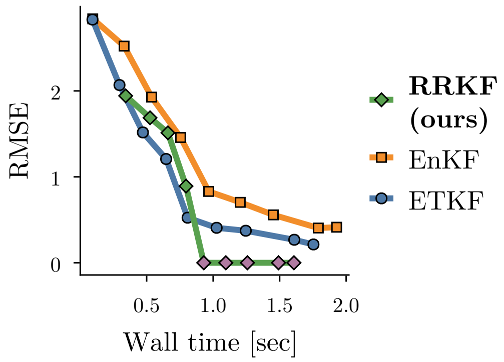
setting described in Section A.4.3, is solved to investigate this. A ground truth realization is drawn from a spatiotemporal Matérn-1/2 process in the temporal domain $[0, 50]$ with a step size of $\Delta t = 0.1$. Data is generated by adding zero-mean Gaussian noise to $m = 150$ state components at 100 randomly sampled time points. The spatial grid is a uniformly spaced grid in the interval $[0, 20]$ with spatial step size of $\Delta x = 0.1$.

Figure A.1 visualizes the distribution of the vector of Z-scores $\frac{\mu_N - x_N^\star}{\sigma_N}$, where μ_N and σ_N are the final-step filtering/smoothing mean and standard deviation, respectively. $x_N^\star$ is the ground-truth state at time t_N. The Z-scores are computed for increasing values of r. The first $r = 50$ eigenvalues of the final-step Kalman filter covariance matrix account for around 97.8% of the spectrum. No covariance inflation [31, Section 4.4] is used for the ensemble methods.

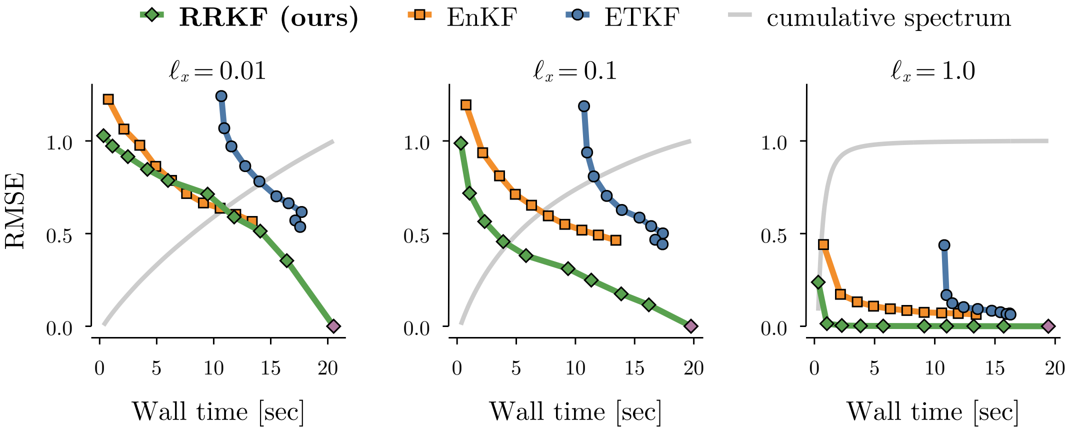
It becomes apparent that—especially for very small values of r—there are significantly more high Z-scores than expected. This indicates that too many states are estimated poorly and divided by a small standard deviation. In their naïve implementation, all examined low-rank algorithms exhibit this behavior. For the RRKF, it is left for future work to find a principled solution to account for missing covariance information in a computationally efficient manner.

A.6 Error as a function of raw computation time

In addition to the previous anaylsis of the error in the approximate filtering estimate with varying low-rank dimensions, Figures A.2a and A.2b show the error as a function of wall-clock computation times for the linear-advection model (Section 3.4.1) and the spatiotemporal Matérn model (Section 3.4.3), respectively.



(a) **Linear advection dynamics (true rank = 51).**



(b) **London air-quality regression.**

Figure A.2: **Error of the low-rank filters as a function of wall-clock computation time.** The raw computational expenses are comparable to those of the ensemble methods. The lower-left corner is the optimal setting with low error and low computation time. All methods approach this region of the plot for faster decays of the spectrum, while the RRKF performs better most of the time.

B

Appendix for Chapter 4

B.1 Spatial region: coarse and fine grid

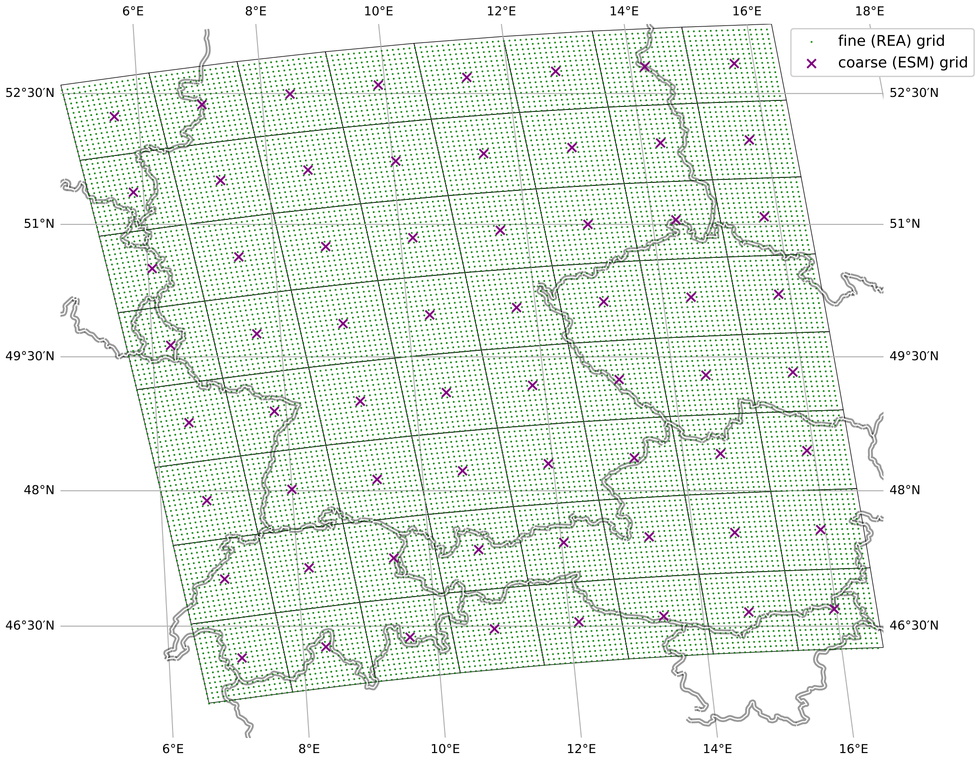
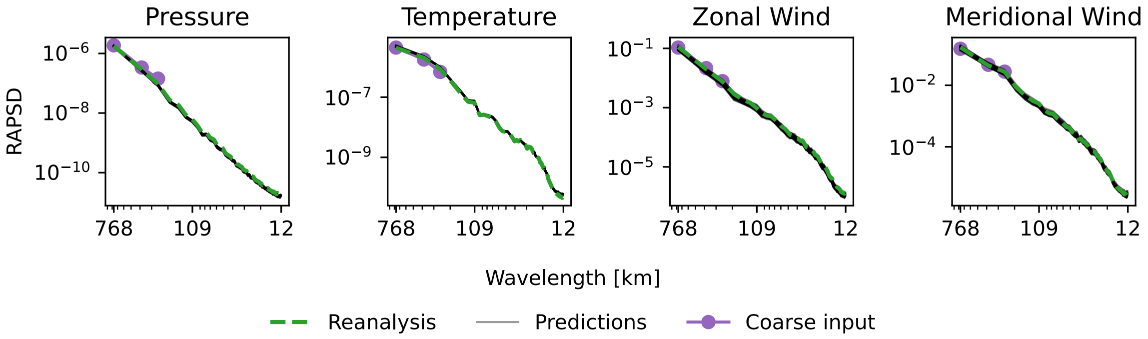


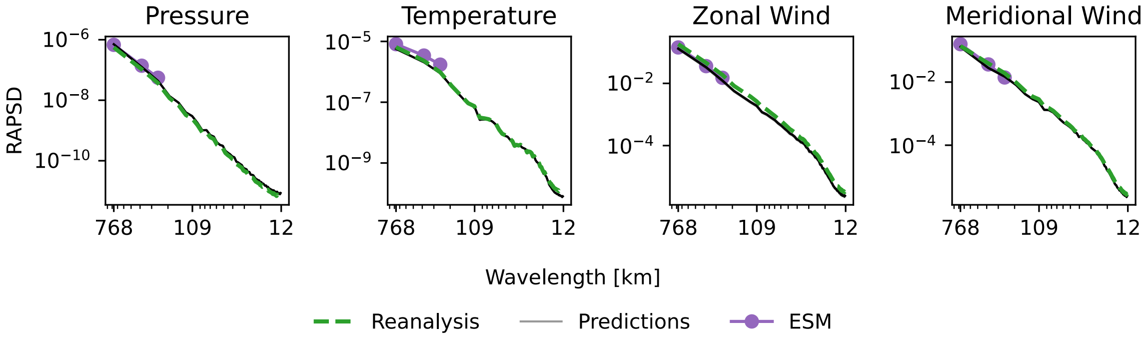
Figure B.1: This plot shows the spatial region considered in this study. The coarse 8 × 8-node grid is marked with purple crosses. The fine 128 × 128-node grid is marked with green dots. Each coarse-grid node lies in the center of a corresponding 16 × 16-patch of high-resolution grid nodes. Both grids span exactly the same area.

B.2 Spatial patterns on multiple scales

To assess the performance of the downscaling model across multiple spatial length scales, we show that the radially averaged power spectral densities (RAPSD) [176] of the predictions align with the reanalysis data in Figure B.2. The RAPSD is computed by averaging the power spectrum over all directions of the same wavenumber in Fourier space. The quantity is commonly used in the context of weather dynamics, especially when estimating precipitation [87, 92, for example]. We use the open-source `pysteps` package by Pulkkinen et al. [163] to compute the RAPSD. The reported RAPSD values are averages over the considered time period.



(a) RAPSD for reanalysis data, coarse input, and predicted fine-grained reanalysis data.



(b) RAPSD for reanalysis data, CMIP6 simulations, and downscaled CMIP6 simulations.

Figure B.2: This plot shows the RAPSD for reanalysis data, coarse input, and the corresponding downscaled predictions. Subplot (a) mirrors the experimental setup of the on-model experiment that predicts high-resolution reanalysis data during the cyclone "Friederike" in January 2018. Subplot (b) covers the CMIP6 downscaling setup as described in the Methods section.

B.3 Embedding the predicted region in an extended spatial context

We embed the high-resolution predictions during the cyclone "Friederike" (c.f. Figure 4.4) into a larger spatial context in order to learn about how long-distance interconnections between the studied spatial region and its surroundings are captured by the model. We argue that it is likely that our statistical downscaling model is able to capture the global connectedness of weather dynamics that are contained in the reanalysis data it is trained on. Concretely, Figures B.3 to B.6 visualize the spatiotemporal dynamics of downscaled samples, reanalysis data, and conditioning input. Thereby, the spatial region is extended beyond the one considered in this study. Reanalysis data is used to fill in the regions outside of the predicted patch. Clearly visible or implausible transitions at the edges of the patch would indicate that the model predictions do not align with the surrounding spatial context. We find, however, that there is a smooth transition from outside the predicted area to its interior for the generated high-resolution predictions. The generated local dynamics seamlessly fit into the more surrounding context of reanalysis data.

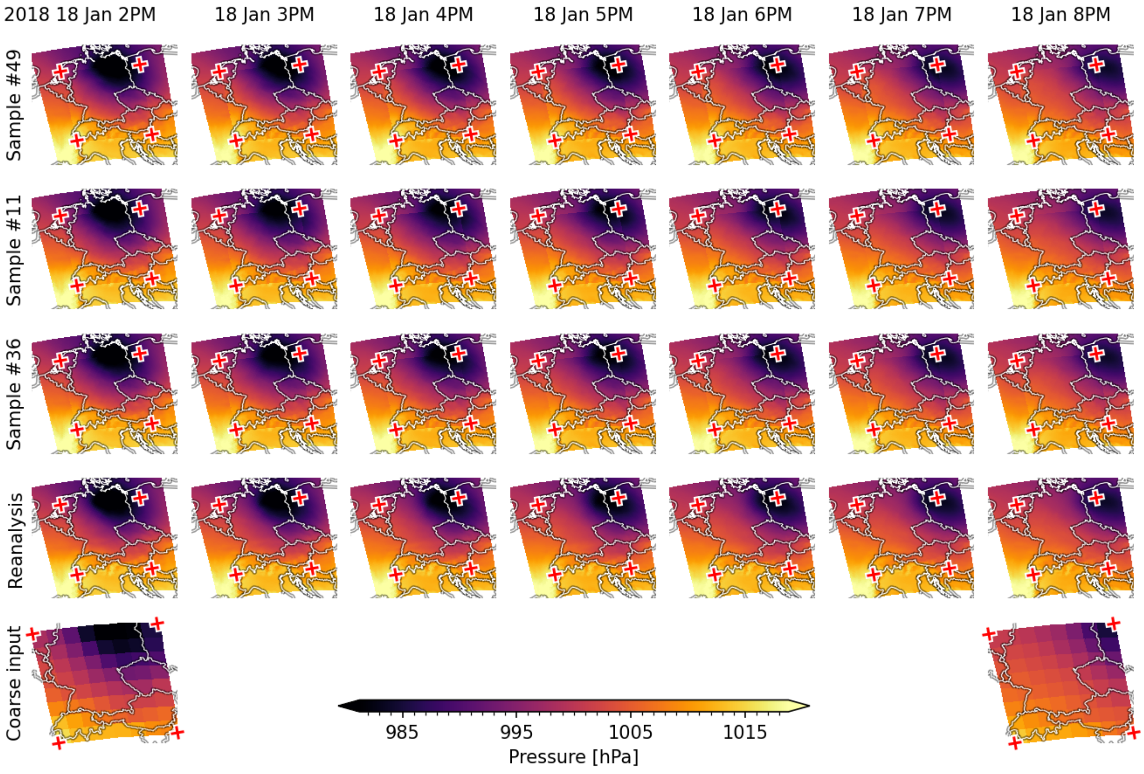


Figure B.3: Predicting mean sea-level pressure during the cyclone "Friederike" (c.f. experiment above). Outside of the prediction range, which lies in the region indicated by red crosses, reanalysis data is filled in to give an impression as to how the predictions fit into more global dynamics.

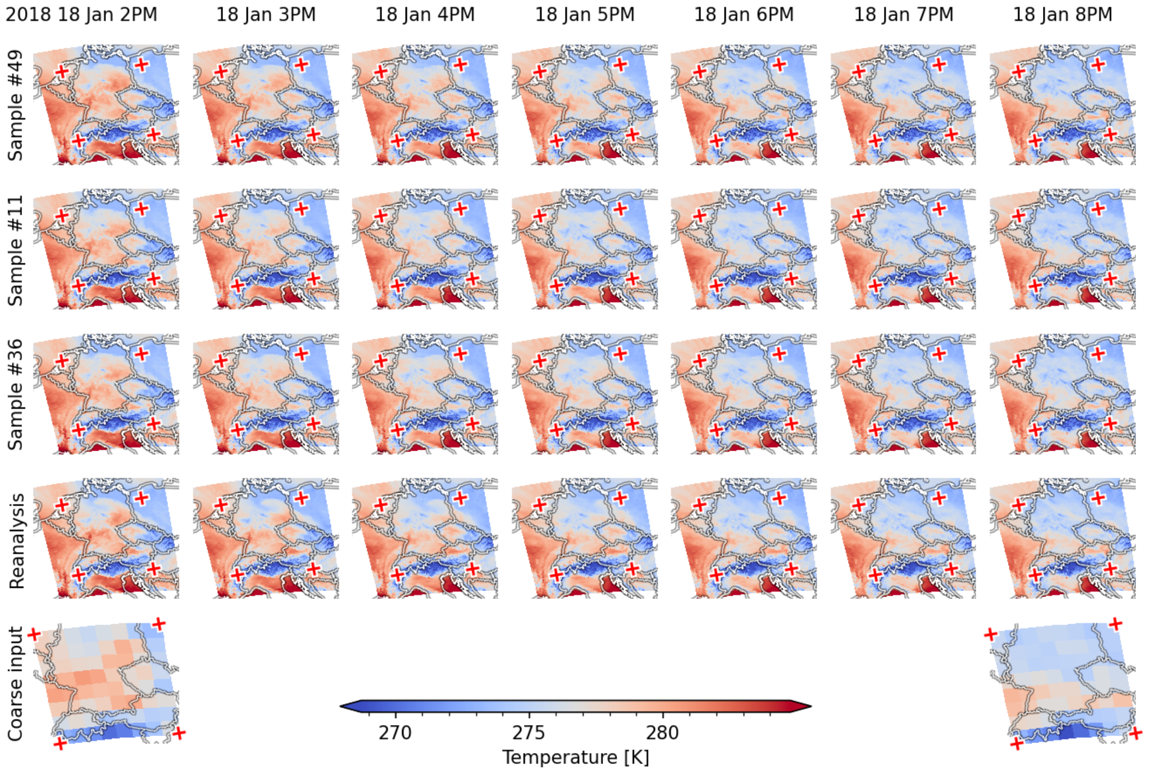


Figure B.4: Predicting surface air temperature during the cyclone "Friederike" (c.f. experiment above). Outside of the prediction range, which lies in the region indicated by red crosses, reanalysis data is filled in to give an impression as to how the predictions fit into more global dynamics.

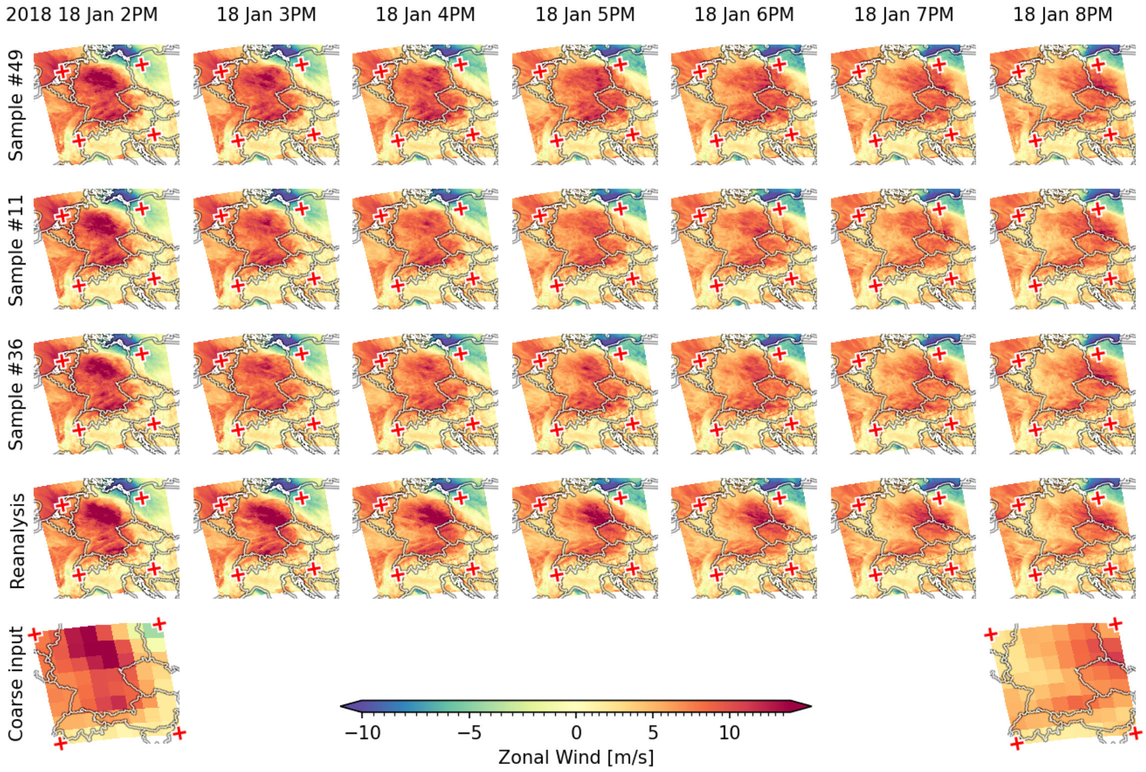


Figure B.5: Predicting zonal wind speed during the cyclone "Friederike" (c.f. experiment above). Outside of the prediction range, which lies in the region indicated by red crosses, reanalysis data is filled in to give an impression as to how the predictions fit into more global dynamics.

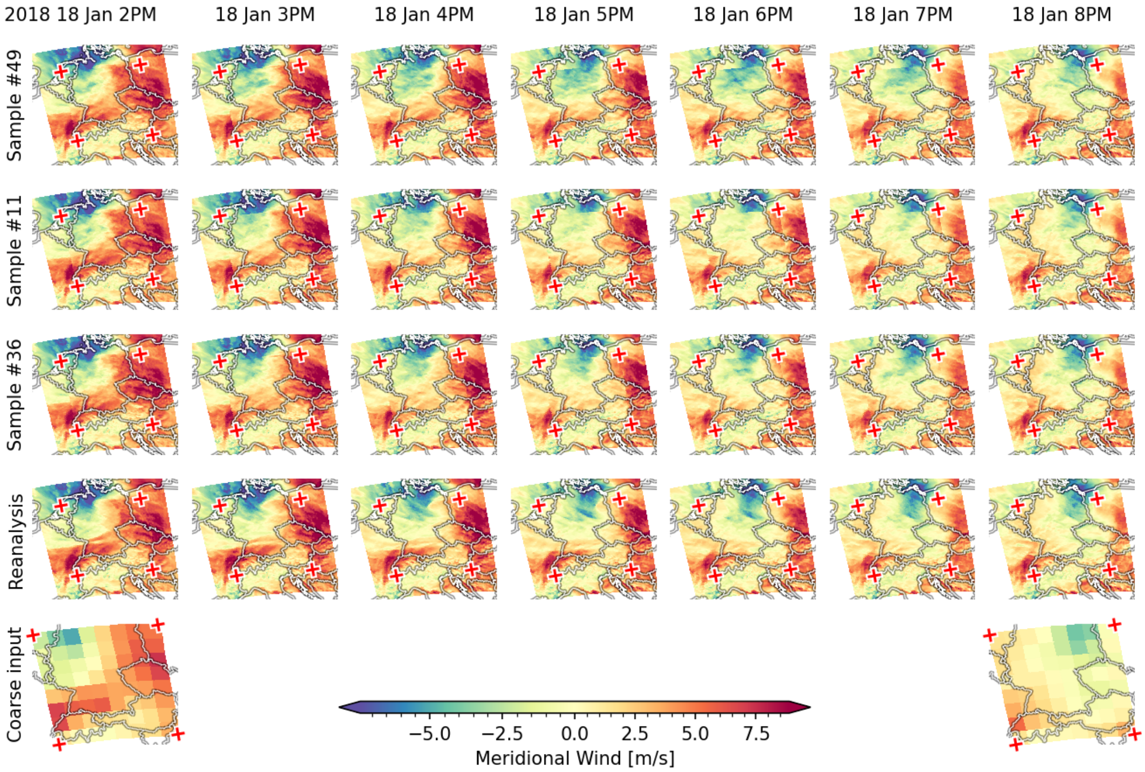


Figure B.6: Predicting meridional wind speed during the cyclone "Friederike" (c.f. experiment above). Outside of the prediction range, which lies in the region indicated by red crosses, reanalysis data is filled in to give an impression as to how the predictions fit into more global dynamics.

B.4 Wind power prediction

A crucial motivation for spatiotemporal downscaling climate simulations to the weather scale is downstream tasks that require future local weather patterns. We provide an exemplary evaluation of the estimated generated wind powers as computed from the CMIP6 simulation, the downscaled predictions, and the reanalysis data. We find that, when computing the spatial average of generated wind powers, the wind-speed and wind-power predictions of the ESM are matched by the downscaled samples in distribution (Figure B.7). Comparing single locations reveals that the ESM locally sometimes over-predicts or under-predicts the wind power generated following the reanalysis data. For both cases, we consistently find that our model corrects the respective over- and under-estimation for multiple randomly selected locations. We compare estimated densities of the wind-speed values (Figure B.7 **a**) and use this distribution to derive the amount of wind power generated from the respectively predicted wind speeds (Figure B.7 **b**). For that, we first compute the wind-power curve for a range of wind-speed values (0m/s to 30m/s), using the open-source package `windpowerlib` [80] and an arbitrary wind-turbine model (turbine type: "E-115/3000", hub-height: 100m). Then, we weigh this power curve with the density of predicted wind speeds to obtain the predicted wind power for each wind speed, taking into account how likely this wind speed value is to occur in the predictions. Finally, we compare the accumulated generated wind power over time. We show the comparison between two pairs of locations in Figure B.8.

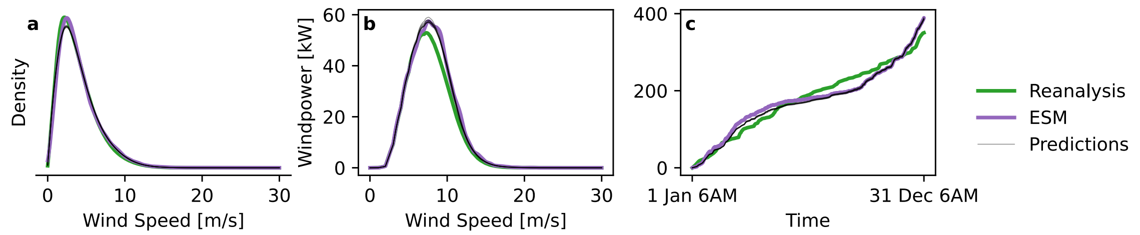
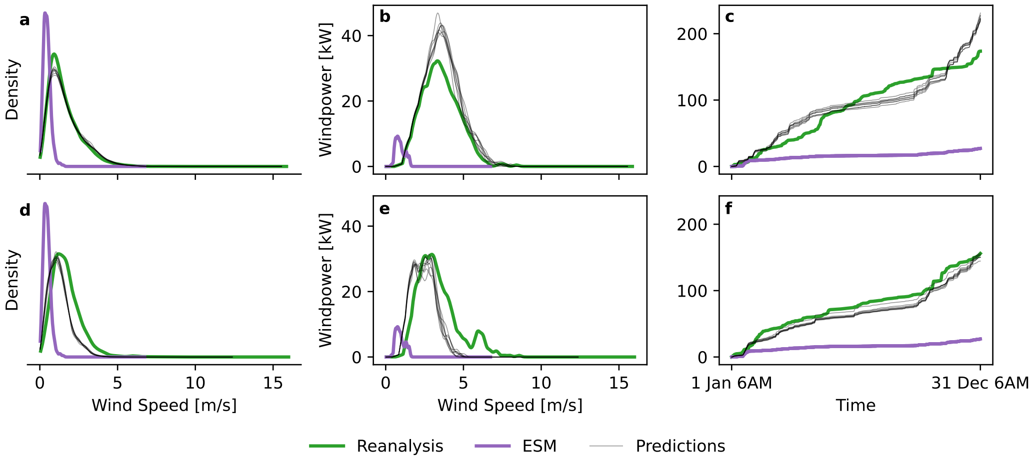
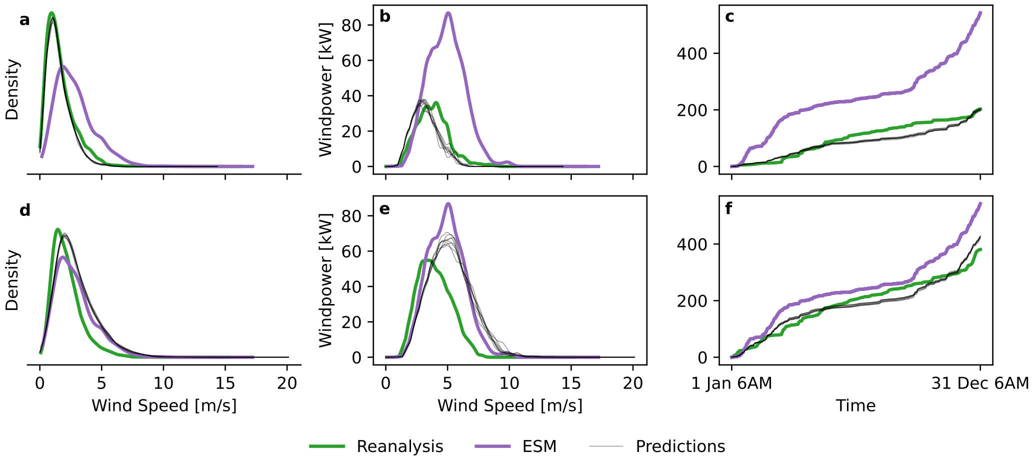


Figure B.7: **Estimated generated wind power over time, averaged over the spatial domain.** This plot analyzes the local wind-speed predictions via the generated wind-power derived from the wind speeds. The full spatial region is considered over the year 2014. Subplot **a** visualizes a kernel-density estimate of the aggregated wind speeds for reanalysis data (green), ESM simulations (purple), and downscaled predictions (black). In **b**, we plot the generated wind power for the entire range of wind speeds from 0m/s to 30m/s, weighted by the density of the estimated wind speeds from **a**. We use a wind-power curve of the form Carrillo et al. [32, Fig. 1]. This estimates the wind power that is generated for the respective wind speeds and accounts for the frequency at which these wind speeds occur. Finally, subplot **c** shows the cumulative sum of the generated wind powers over time. All wind-power values are normalized by the number of time steps on the respective temporal grids to align the wind-power scales. From the black lines aligning closely with the purple line, we conclude that the downscaled predictions preserve the aggregated wind-power generation as simulated by the ESM.



(a) Compare generated windpower at two distinct locations that share a single node on the coarse ESM grid. In this instance, the ESM underpredicts the generated windpower.



(b) Compare generated windpower at two distinct locations that share a single node on the coarse ESM grid. In this instance, the ESM overpredicts the generated windpower.

Figure B.8: **Local estimates of wind power over time.** This plot investigates two pairs of locations that are each constrained by a single spatial node on the coarse climate grid. The layout of each row of the subplots (a) and (b) mirrors Figure B.7; each row corresponds to a single location on the fine-resolution grid, respectively.

B.5 Quantitative evaluation

We first apply quantile-mapping to bias-correct the climate patches. These frames are used as input for the different downscaling algorithms. Our model is compared to the following two benchmark approaches.

Benchmark 1: Interpolation

The first benchmark consists of a bias-correction step, followed by bi-linearly interpolating the coarse data to a finer spatial grid. This two-step procedure is called Bias Correction Spatial Disaggregation (BCSD) [217] and is both simple and prominently used in the downscaling literature. The reanalysis data is first remapped to match the coarse resolution of the climate model grid. The bias correction step then uses quantile mapping applied to the entire spatial domain. To account for seasonal variations, our implementation uses a moving average (with a window size of 25 days) centered around each calendar day when computing the quantiles. The temporal pooling ensures robust statistics by considering similar days from the seasonal cycle together.

Benchmark 2: Conditioned frame-to-frame diffusion model

While our approach leverages a score-based approach, i.e., all states/trajectories are generated simultaneously in a non-autoregressive manner, we compare it to a conventional frame-by-frame modeling approach using a denoising diffusion implicit model (DDIM) [196]. We use a time-conditional U-Net backbone, incorporating residual blocks and self-attention layers. The conditioning is then performed by bi-linearly upsampling the bias-corrected climate data to match the high-resolution data and concatenating them along the channel dimension. To predict the middle frame, we condition on a sequence of three frames [187].

Metrics

We use an approximate (sliced) two-dimensional Wasserstein distance [20] as a metric to compare two high-dimensional probability distributions, defined as

$$\mathrm{W}(P_{\mathrm{pred}}, P_{\mathrm{ref}}) = \inf_{\gamma \in \Pi(P_{\mathrm{pred}}, P_{\mathrm{ref}})} \mathbb{E}_{(x,y)\sim\gamma}[\|x-y\|_1], \tag{B.1}$$

where $\Pi(P_{\mathrm{pred}}, P_{\mathrm{ref}})$ is the set of couplings, that is, probability distributions whose marginals are P_{pred} and P_{ref}. This variant makes the traditional Wasserstein distance computationally feasible for high-dimensional probability distributions [20]. In the sliced variant, the high-dimensional data is projected onto a one-dimensional line for which a one-dimensional Wasserstein distance can be efficiently computed. This slicing process is performed repeatedly for multiple different slices, and the result is averaged to obtain a reliable metric.

The Mean Energy Log Ratio (MELR) is used to evaluate the preservation of potentially highly varying physical patterns in the downscaled spatial patches. The MELR is a metric that is derived from the radially averaged power spectral density (Section B.2) as

$$\mathrm{MELR} = \sum_k \left| \log\left(\frac{E_{\mathrm{pred}}(k)}{E_{\mathrm{ref}}(k)} \right) \right|, \tag{B.2}$$

where the energy of the predicted field $E_{\mathrm{pred}}(k)$ is compared to that of the reference field $E_{\mathrm{ref}}(k)$.

We use the structural similarity index measure (SSIM) [221], which takes into account perceptual properties of local structures when quantifying the similarity between two spatial data points. The SSIM is defined by sliding a window $W_k(x, y)$ of size $k \times k$ along the spatial data point that computes for two $k \times k$-patches x and y

$$W_k(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{x,y} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{B.3}$$

where μ_x, σ_x and μ_y, σ_y are the respective average and standard-deviations of the values in the respective patches x and y. Analogously, σ_{xy} is the covariance between the patches. Finally, c_1 and c_2 are constants to make the computation more robust. The SSIM takes on values between 0 and 1, whereby higher values indicate more structural similarities between the compared data points. An SSIM of 1 can only be attained when both data points are identical. In Table B.1 we report the SSIM using a window size of $k = 15$.

Table B.1: **Quantitative evaluation of downscaling methods.** We compare the performance of our model, based on score-based data assimilation (SDA), to a conditional denoising diffusion implicit model (DDIM) and to bias-correction spatial disaggregation (BCSD). We report the sliced Wasserstein-1 distance (Sliced W1) over time, temporally averaged energy log ratio (MELR), and structural similarity index (SSIM) for the four variables: mean sea-level pressure (`psl`), surface (2m) air temperature (`tas`), surface (10m) zonal `uas`, and meridional (`vas`) wind speeds. The values are reported as mean ± standard deviation over the generated predictions. Note that, since the benchmark methods are not capable of temporal downscaling, the quantitative evaluation is performed on the 6-hourly observation grid. Even though the method proposed in this work is under the additional constraint of generating temporally consistent 1-hourly trajectories, it beats both benchmarks in most cases. Between all approaches, the best performance per metric and variable is highlighted in **bold**.

Method **Metric**	var	**SDA (ours)**	**BCSD**	**DDIM**
Sliced W1 ↓	`psl`	0.3028 ± 0.0012	**0.2900** ± 0	0.3526 ± 0.0536
	`tas`	0.3526 ± 0.0005	0.3999 ± 0	**0.2894** ± 0.0999
	`uas`	**0.1998** ± 0.0015	0.2523 ± 0	0.4055 ± 0.0586
	`vas`	**0.2348** ± 0.0033	0.2702 ± 0	0.3553 ± 0.0371
MELR ↓	`psl`	**1.0256** ± 0.0031	1.1625 ± 0	1.0874 ± 0.1022
	`tas`	**0.3470** ± 0.0018	1.1312 ± 0	0.3932 ± 0.0187
	`uas`	**1.4622** ± 0.0025	2.4141 ± 0	1.5598 ± 0.0770
	`vas`	1.4555 ± 0.0096	2.4042 ± 0	**1.3317** ± 0.0664
SSIM ↑	`psl`	0.8925 ± 0.0002	**0.9027** ± 0	0.8691 ± 0.0074
	`tas`	**0.8524** ± 0.0002	0.7066 ± 0	0.8382 ± 0.0094
	`uas`	**0.1351** ± 0.0008	0.0844 ± 0	0.1182 ± 0.0061
	`vas`	**0.1322** ± 0.0009	0.0973 ± 0	0.1185 ± 0.0068

B.6 Approximating the gradient of the observation model

Using Bayes' rule, we obtain the posterior score function

$$\nabla_{\mathrm{Z}(\tau)}[\log p_\tau(\mathrm{Z}(\tau) \mid \mathrm{Y})] = \nabla_{\mathrm{Z}(\tau)}[\log p_\tau(\mathrm{Z}(\tau)) + \log p(\mathrm{Y} \mid \mathrm{Z}(\tau))] \tag{B.4}$$

$$= \underbrace{\nabla_{\mathrm{Z}(\tau)}[\log p_\tau(\mathrm{Z}(\tau))]}_{\approx s_\theta(\mathrm{Z}(\tau),\tau)} + \nabla_{\mathrm{Z}(\tau)}[\log p(\mathrm{Y} \mid \mathrm{Z}(\tau))]. \tag{B.5}$$

This illustrates that the conditioning mechanism, which involves only a simple addition of the gradient of the log-observation model, is independent of the trained score model. However, as detailed by Chung et al. [39], the posterior score requires relating the measurement Y to the diffused state $\mathrm{Z}(\tau)$. Chung et al. [39] propose to approximate

$$p(\mathrm{Y} \mid \mathrm{Z}(\tau)) = \int p(\mathrm{Y} \mid \mathrm{Z}(0), \mathrm{Z}(\tau))\ p(\mathrm{Z}(0) \mid \mathrm{Z}(\tau))\ \mathrm{dZ}(0) \tag{B.6}$$

$$= \int p(\mathrm{Y} \mid \mathrm{Z}(0))\ p(\mathrm{Z}(0) \mid \mathrm{Z}(\tau))\ \mathrm{dZ}(0) \tag{B.7}$$

$$= \mathbb{E}_{\mathrm{Z}(0)\sim p(\mathrm{Z}(0)\mid\mathrm{Z}(\tau))}[p(\mathrm{Y} \mid \mathrm{Z}(0))] \tag{B.8}$$

$$\approx p\Big(\mathrm{Y} \mid \hat{\mathrm{Z}}(0) := \mathbb{E}_{\mathrm{Z}(0)\sim p(\mathrm{Z}(0)\mid\mathrm{Z}(\tau))}[\mathrm{Z}(0)]\Big). \tag{B.9}$$

Intuitively, the last step pulls the expectation into the conditioning, which is not equivalent in general, and thereby approximates $p(\mathrm{Y} \mid \mathrm{Z}(\tau)) \approx p(\mathrm{Y} \mid \hat{\mathrm{Z}}(0))$. The quantity $\hat{\mathrm{Z}}(0)$ is a posterior-mean estimate for the noise-free data point underlying the diffused state $\mathrm{Z}(\tau)$ at the current diffusion-time τ in the generative process. In the following, we detail how $\hat{\mathrm{Z}}(0)$ is computed.

Linear stochastic differential equations, like the diffusion process defined in Equation (4.2), can be equivalently formulated in terms of a discrete, linear Gaussian transition model

$$p(\mathrm{Z}(\tau + \Delta\tau) \mid \mathrm{Z}(\tau)) = \mathcal{N}(\mathrm{Z}(\tau + \Delta\tau); A(\Delta\tau)\mathrm{Z}(\tau), \Sigma(\Delta\tau)), \tag{B.10}$$

for some time increment $\Delta\tau$. We refer to, e.g., Särkkä et al. [181, Section 6.1] for more details on how to derive the *transition* and *process-noise covariance* functions A and Σ from the drift F and dispersion L of the diffusion process (Equation (4.2)). The forward diffusion process can be simulated over extended time ranges in a single step via sampling from the Gaussian transition density

$$\mathrm{Z}(\Delta\tau) \mid \mathrm{Z}(0) = A(\Delta\tau)\mathrm{Z}(0) + \Sigma^{\frac{1}{2}}(\Delta\tau)\epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \tag{B.11}$$

which is an equivalent formulation of Equation (B.10) for the case $\tau = 0$, i.e., starting from the noise-free data point [94, 199, 196]. The same is not possible in the reverse direction, which would make the generation process trivial. Unfortunately, the backwards transition model associated with Equation (4.3) is Gaussian only for infinitesimally small time decrements $\Delta\tau \to 0$ [65, 194], which leads to sampling quality increasing with the number of steps used for simulating the generative process. However, it is possible to estimate the posterior mean $\hat{\mathrm{Z}}(0)$ based on the current score estimate $s_\theta(\mathrm{Z}(\tau), \tau)$. By re-arranging Equation (B.11) we note that

$$\begin{aligned} \mathrm{Z}(0) &= A^{-1}(\tau)\Big(\mathrm{Z}(\tau) - \Sigma^{\frac{1}{2}}(\tau)\epsilon\Big) && \text{(B.12)} \\ &= A^{-1}(\tau)\big(\mathrm{Z}(\tau) + \nabla_{\mathrm{Z}(\tau)}[p_\tau(\mathrm{Z}(\tau))]\Sigma(\tau)\big) && \text{(B.13)} \\ &\approx A^{-1}(\tau)(\mathrm{Z}(\tau) + s_\theta(\mathrm{Z}(\tau), \tau)\Sigma(\tau)) && \text{(B.14)} \\ &= \hat{\mathrm{Z}}(0). && \text{(B.15)} \end{aligned}$$

The first equality uses the direct correspondence between ϵ and the de-noising score function of Equation (B.10) [94, 108]. Since the true score, like ϵ, is not known while generating a new data point, it is in the next step approximated with our parametric score model s_θ. A more detailed derivation can be found for the scalar case ($A, \Sigma \in \mathbb{R}$) in Chung et al. [39].

With that, we established the necessary background regarding the conditional model, in order to describe the approximation to the conditioning mechanism that is used in this work. Plugging into Equations (B.4) and (B.5) the approximation from Equations (B.6) to (B.9) reveals that we have to compute the gradient

$$\nabla_{\mathrm{Z}(\tau)}\Big[\log p(\mathrm{Y} \mid \hat{\mathrm{Z}}(0))\Big]. \tag{B.16}$$

For our purposes, we assume a Gaussian observation model (Equation (4.4)), which simplifies this gradient to

$$\begin{aligned} \nabla_{\mathrm{Z}(\tau)}\Big[\log p(\mathrm{Y} \mid \hat{\mathrm{Z}}(0))\Big] &= \nabla_{\mathrm{Z}(\tau)}\Big[\log \mathcal{N}\Big(\mathrm{Y}; h(\hat{\mathrm{Z}}(0)), R\Big)\Big] && \text{(B.17)} \\ &= \nabla_{\mathrm{Z}(\tau)}\Big[\|\mathrm{Y} - h(\hat{\mathrm{Z}}(0))\|_R\Big], && \text{(B.18)} \end{aligned}$$

where $\|\cdot\|_R$ is a Mahalanobis norm in the multivariate case and a Euclidean norm in the scalar case. Let us denote the residual between conditioning information and the observed predicted signal as $r(\mathrm{Y}, \hat{\mathrm{Z}}(0)) := \mathrm{Y} - h(\hat{\mathrm{Z}}(0))$. Then, using the chain rule of differentiation, we obtain

$$\nabla_{\mathrm{Z}(\tau)}\Big[\|r(\mathrm{Y}, \hat{\mathrm{Z}}(0))\|_R\Big] = \nabla_{r(\mathrm{Y},\hat{\mathrm{Z}}(0))}\Big[\|r(\mathrm{Y}, \hat{\mathrm{Z}}(0))\|_R\Big] \cdot \nabla_{\hat{\mathrm{Z}}(0)}\Big[h\Big(\hat{\mathrm{Z}}(0)\Big)\Big] \cdot \nabla_{\mathrm{Z}(\tau)}\Big[\hat{\mathrm{Z}}(0)\Big]. \tag{B.19}$$

We plug the equality from Equations (B.14) and (B.15) into the third component of the gradient to obtain

$$\begin{aligned} \nabla_{\mathrm{Z}(\tau)}\Big[\hat{\mathrm{Z}}(0)\Big] &= \nabla_{\mathrm{Z}(\tau)}\big[A^{-1}(\tau)(\mathrm{Z}(\tau) + s_\theta(\mathrm{Z}(\tau), \tau)\Sigma(\tau))\big] && \text{(B.20)} \\ &= A^{-1}(1 + \nabla_{\mathrm{Z}(\tau)}[s_\theta(\mathrm{Z}(\tau), \tau)\Sigma(\tau)] && \text{(B.21)} \\ &\approx A^{-1}. && \text{(B.22)} \end{aligned}$$

The final step approximation avoids the computational demanding differentiation of the score function with respect to the perturbed state, which cuts away the majority of the computational cost in the conditioning. Altogether, the approximate conditioning term (c.f. Equation (B.5)), used throughout this work, is

$$\begin{aligned}\nabla_{\mathrm{Z}(\tau)}[\log p(\mathrm{Y} \mid \mathrm{Z}(\tau)] &\approx \nabla_{\mathrm{Z}(\tau)}\Big[\log p(\mathrm{Y} \mid \hat{\mathrm{Z}}(0))\Big] \\ &\approx \nabla_{r(\mathrm{Y},\hat{\mathrm{Z}}(0))}\Big[\|r(\mathrm{Y},\hat{\mathrm{Z}}(0))\|_R\Big] \cdot \nabla_{\hat{\mathrm{Z}}(0)}\Big[h\Big(\hat{\mathrm{Z}}(0)\Big)\Big] \cdot A^{-1},\end{aligned} \tag{B.23}$$

which is very cheap to compute in each de-noising step.

B.7 Anomalies

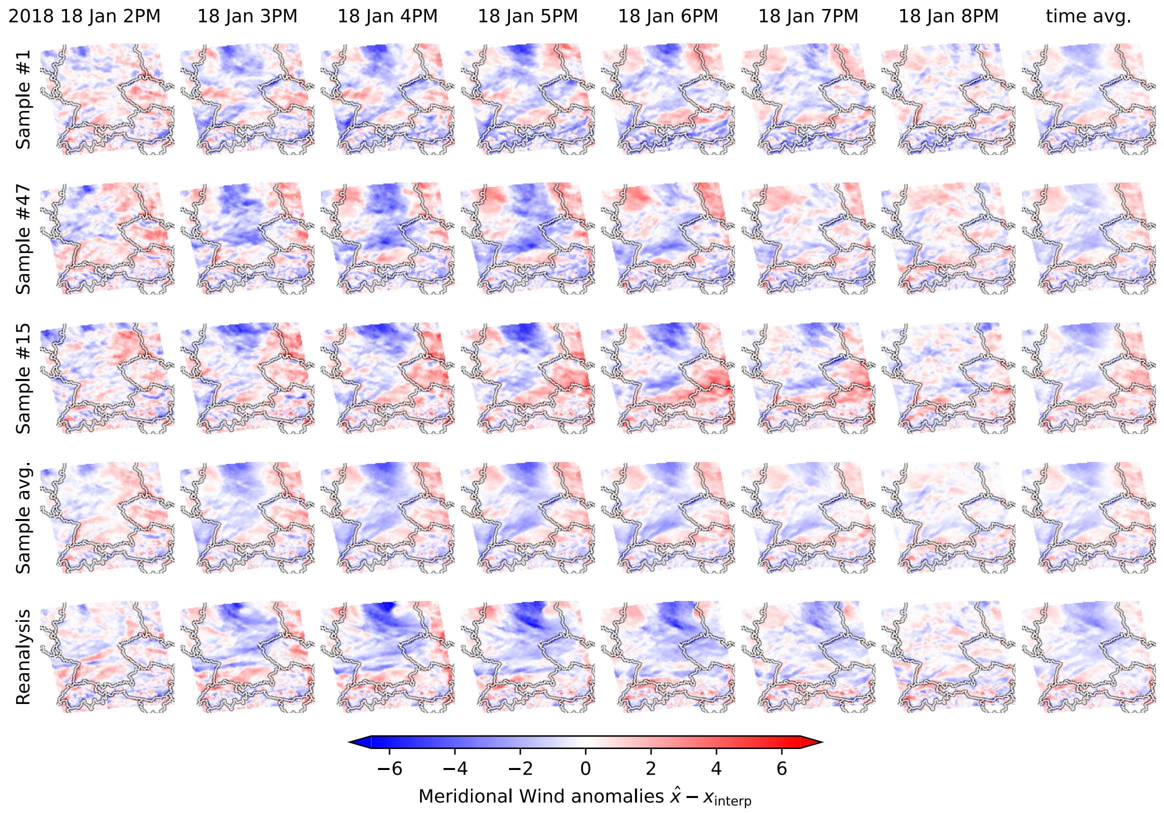


Figure B.9: **Anomalies between downscaled and interpolated versions of coarse input.** This plot supplements Figure 4.4. It visualizes the differences between a spatiotemporal interpolation of the coarse input x_{interp} (Figure 4.4, fourth row) and different fine-scale time series $\hat{x}$: 1) three samples from the proposed downscaling model (*top three rows*), 2) the average of the three samples (*fourth row*), and 3) the ground-truth reanalysis data (*bottom row*). This visualization exposes the local spatial and temporal patterns on the fine grid that are not contained in the coarse data. The rightmost column plots the corresponding temporal averages of the spatial anomalies. As in Figure 4.4, the downscaling model is only conditioned at 2PM (first column) and 8PM (penultimate column). Especially at those conditioning points, the local spatial patterns predicted in each sample (rows 1 through 3) are structurally similar to those in the reanalysis data (bottom row). Between the conditioning points (3PM through 7PM), the ground-truth anomalies (bottom row) expose that the temporal evolution of the cyclone is not predicted by the smooth temporal interpolation. Our model predicts spatiotemporal structure on the fine grid, adding information that cannot be trivially inferred from the coarse input.

B.8 Generative de-noising process

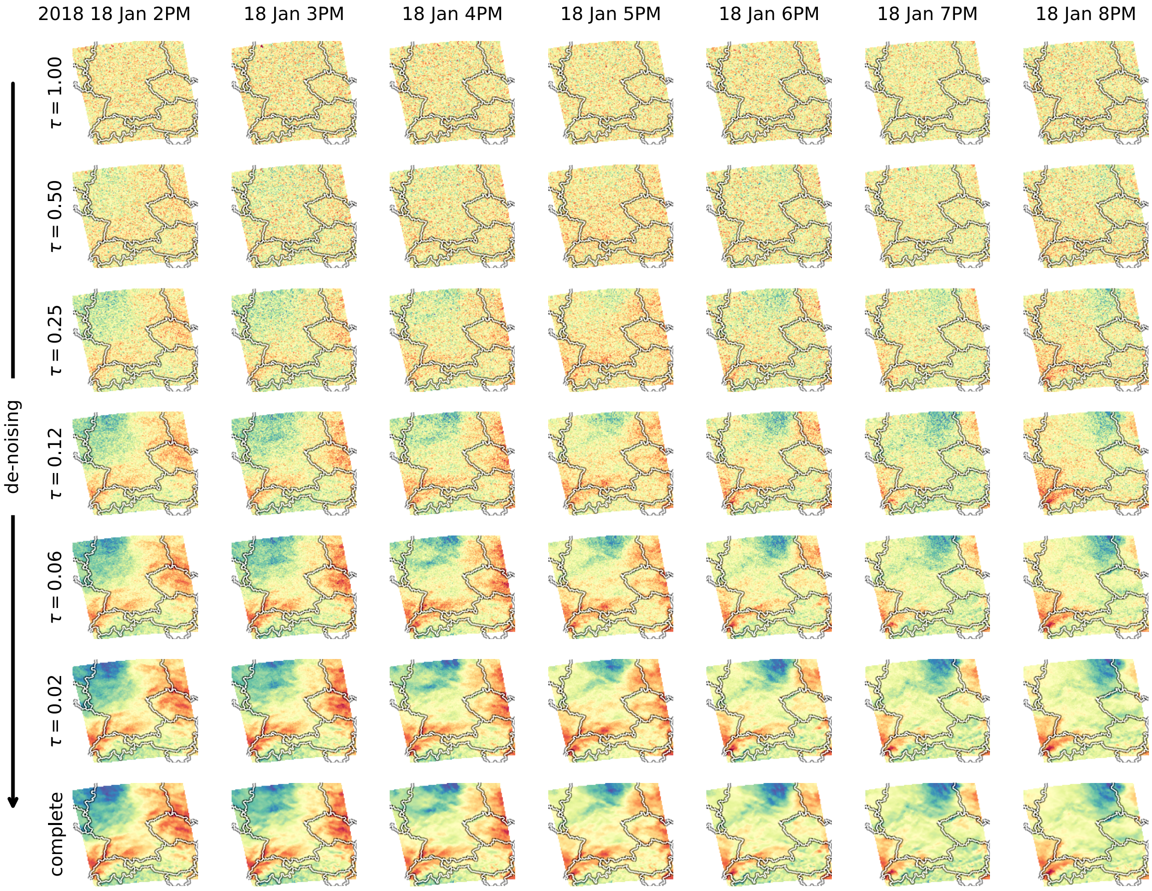


Figure B.10: **The generative process of the diffusion model.** Diffusion models learn a mapping from a tractable noise distribution ($\tau = 1$; often Gaussian noise) to the training-data distribution. Using a statistical model for the score function, which separates noise from signal, an initial random-noise sample is iteratively de-noised into a data point that lies in a region of high data density. This figure shows seven (of 256) steps of this generative process, beginning at $\tau = 1$ (*top row*; Gaussian noise) and ending at $\tau = 0$ (*bottom row*). Fine-scale features are generated by the model towards the end of the generative process. Notably, the spatial and temporal structure emerge jointly, since the score function is estimated for the entire time series.

B.9 Relationship between downscaled variables

This experiment demonstrates that the prediction for one variable is affected by conditioning information about the other remaining variables through inter-variable relationships that the generative model learned through training. To show this, we isolate one variable "v" of interest (here: meridional wind speeds) and denote the remaining variables (here: mean sea-level pressure, surface temperature, zonal wind speeds) as "¬v". We predict four different downscaled sequences for v:

1. First, we draw from the prior. Sampling from the unconditioned generative downscaling model, yields an uninformed sequence of weather patterns.
2. Second, we draw from the generative model that is conditioned only on ¬v.
3. Third, we draw from the generative model that is conditioned only on v.
4. Finally, we draw from the fully conditioned generative downscaling model, providing the model with the information about all considered variables, v and ¬v, as is the default case in the other experiments (e.g., Figure 4.4).

Figure B.11 visualizes these differently-informed samples of v as spatiotemporal sequences (cf. Figure 4.4). A comparison between the unconditioned sample (1.; first row in Figure B.11), the different partly-conditioned predictions (2. and 3.; second and third rows in Figure B.11), and the fully-conditioned predictions (4.; fourth row in Figure B.11), demonstrates that the multivariate downscaling model has learned relationships between the variables, which it uses for generating downscaled predictions. Note that this experiment is purely diagnostic and serves to validate that the model learns and does not neglect relationships between the considered variables.

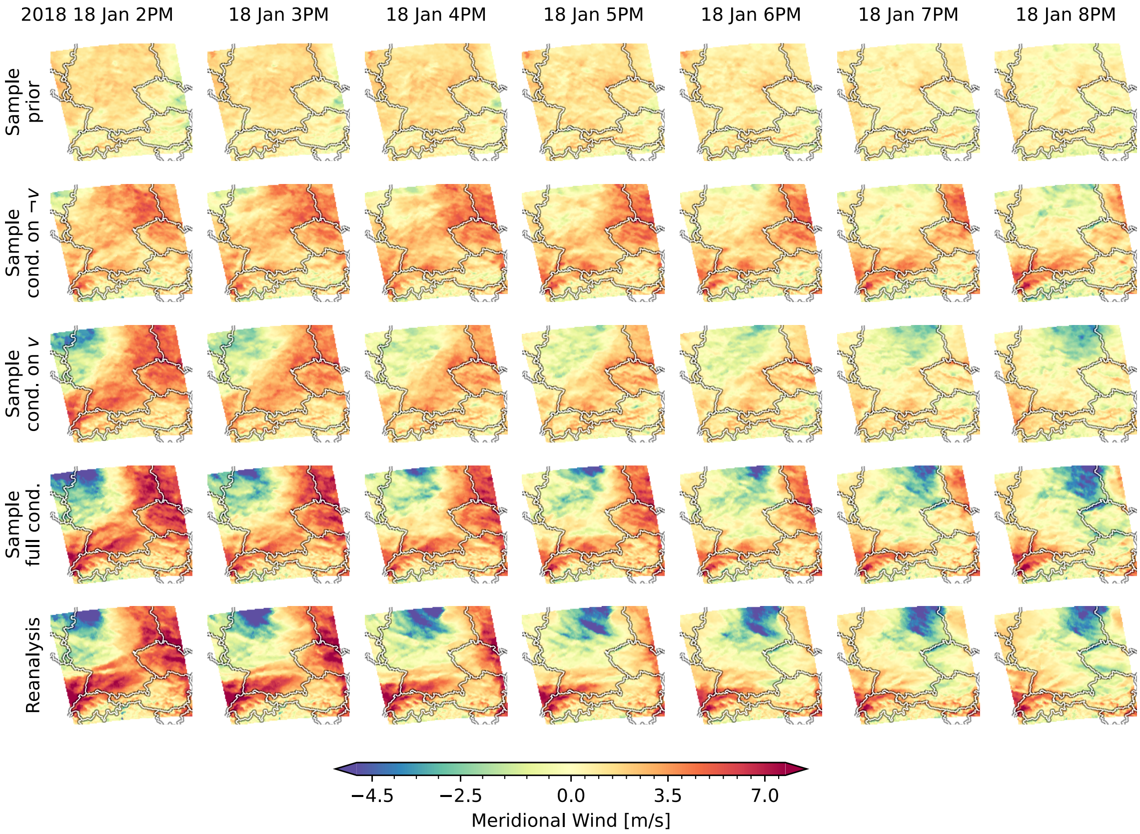


Figure B.11: **The model learns and uses relationships between variables.** The first four rows of this plot visualize four different predictions for downscaled meridional wind during a cyclone ("Friederike, January 2018). The bottom row shows the reanalysis data for comparison. For the conditioned predictions (rows 2 through 4), no information is provided to the model between the first (2:00 PM) and the last (8:00 PM) visualized time point, exactly as in Figure 4.4. We denote the visualized variable of interest (here: meridional wind speeds) as "v", and the other variables (here: mean sea-level pressure, surface temperature, and zonal wind speeds) as "¬v". As in Figure 4.4, the sign of the wind speed value defines its direction and time progresses from left to right hourly, starting 2018 January 18 at 02:00 PM and ending the same day at 08:00 PM. The *first row* shows a sample from the unconditioned generative model, which is entirely uninformed by any coarse input. The *second row* shows the downscaled v, predicted by the generative model that is only conditioned on ¬v. The *third row* shows the downscaled v, predicted by the generative model that is only conditioned on v. The *fourth row* shows the downscaled v, predicted by the generative model conditioned on all variables, v and ¬v. The plot serves to demonstrate the effect of the multivariate nature of the downscaling model. Comparing the model outputs when conditioning on different sets of variables demonstrates that the prediction of a variable v is affected by incorporating information about the other variables ¬v. This allows the conclusion that the downscaling model has learned inter-variable relationships, which it uses for prediction.

C Appendix for Chapter 5

This Supplementary Information (SI) provides additional details and supporting analyses for Chapter 5. We begin by evaluating the performance of the downscaling framework through a comparison of variable distributions between ground truth and downscaled predictions (Section C.1, Figure C.1), followed by a visual example illustrating the reconstruction of fine-scale spatial temperature patterns for a representative day (Figure C.2). Next, we describe the methodology for computing national average capacity factors (Section C.2), highlighting the spatial weighting of grid cells within Germany (Figure C.3), and present the spatial distribution of wind and solar capacity factors across the country (Figure C.4). We then compare time series of combined wind and solar capacity factors at different spatial and temporal resolutions, demonstrating the importance of high-resolution data and bias correction (Figure C.5). For future projections, we assess in Section C.3 the local risk of prolonged low-capacity-factor events under different emission scenarios, showing both the spatial distribution of event frequency and the inter-model variability across the ensemble (Figure C.6). Finally, we provide more details on the time-conditioning of the generative downscaling model in Section C.4.

C.1 Downscaling, distributions and bias correction

The downscaling framework is evaluated by comparing the value distributions of ERA5 reanalysis data (0.25°) to the downscaled predictions from artificially coarsened ERA5 (1°). The results are shown in Figure C.1. Overall, the downscaled values match the distribution of the ground truth (red shading). We find that the downscaled values for the u-component of wind (Figure C.1 **a.**), and v-component of wind (Figure C.1 **b.**) slightly overestimate the distribution around zero. This artifact is a known problem that arises from temporal averaging of wind data [53], as daily averages ignore the variability throughout the day-night cycle (compare the light versus dark blue lines to the ground truth red line in Figure C.1 **a.**, **b.**). We therefore apply a further post-processing bias correction step to the downscaled values, which leads to a better match with the ground truth (compare light blue lines to the red line in Figure C.1 **a.**, **b.**).

For surface air temperature (Figure C.1 **c.**), the downscaled samples (light and dark blue lines) closely match the ground truth. For the downward solar radiation, the coarse input data has a very different distribution (compare blue to green lines) due to the daily averaging of the input data. The downscaled distribution still matches that of the ground truth closely.

An illustrative example for the spatiotemporal downscaling is presented for a representative day, October 10, 2023, in Figure C.2. The visual comparison confirms that the downscaled

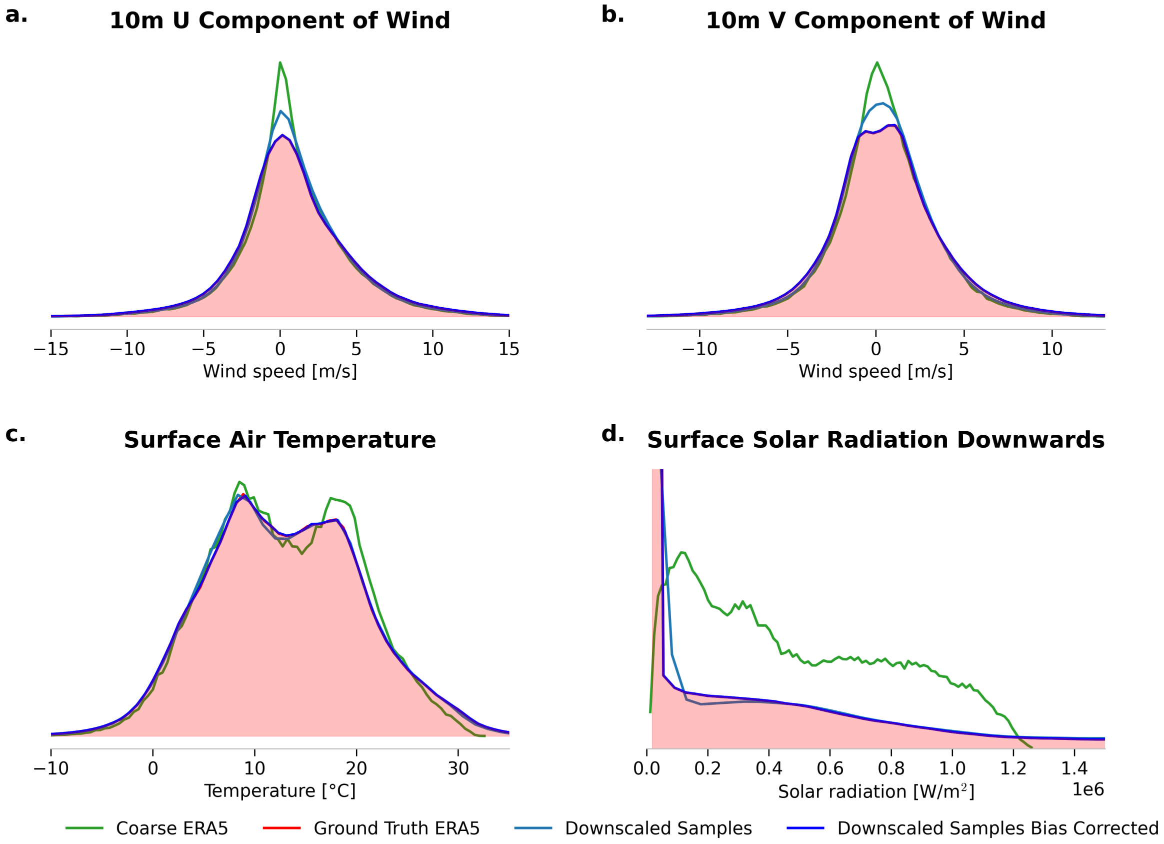


Figure C.1: **Comparison of ground truth and downscaled values for key variables.** The figure shows the performance of the downscaling framework for four variables: (**a.**) u-component of wind, (**b.**) v-component of wind, (**c.**) surface air temperature, and (**d.**) downward solar radiation at the surface. For each variable, the distribution of ground truth (ERA5 reanalysis, red shading) values is compared to the downscaled output (light blue: samples, dark blue: post-processed bias corrected samples). The coarse input data (ERA5, artificially coarsened to 1°, daily resolution, green) is also shown for reference. The evaluation period ranges from January 01, 2023 to December 31, 2024.

fields show a strong agreement with the ground truth, indicating that the framework effectively captures temporal variability, including the diurnal cycle.

C.2 Capacity factors

Capacity factors (CF) are computed per grid cell and per technology (onshore wind, offshore wind, photovoltaics) and then aggregated spatially, yielding a scalar time series. For a single cell i and technology k at time t, we define

$$\mathrm{CF}_{i,k}(t) = \frac{P_{i,k}(t)}{P_{i,k}^{\max}}, \tag{C.1}$$

where $P_{i,k}(t)$ is the instantaneous power output estimated from the meteorological fields (wind components, surface temperature, downward shortwave radiation) using the conversion routines

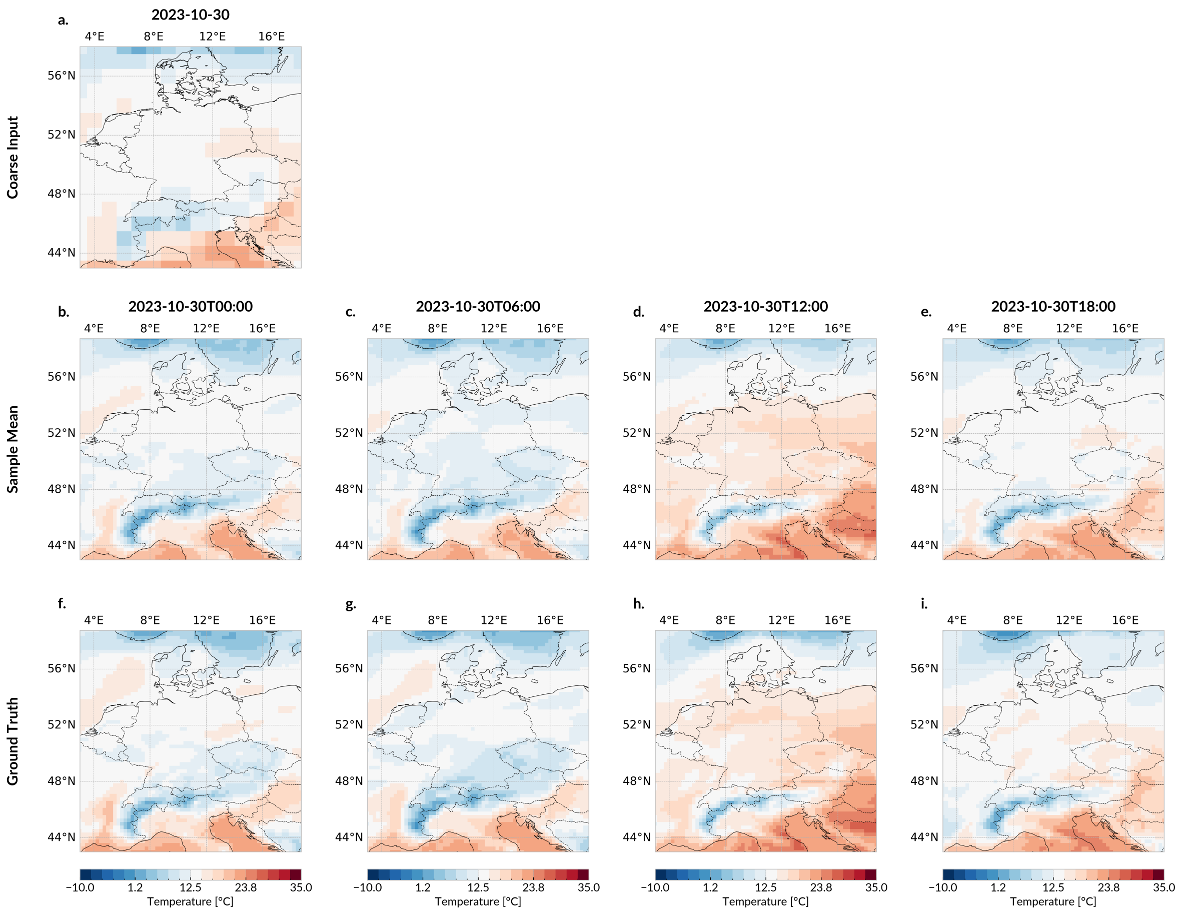


Figure C.2: **Visual example of the downscaling process for surface air temperature on October 30, 2023.** The figure compares the coarse-resolution input (**a.**), the downscaled output (**b.**–**e.**), and the ground truth (ERA5 reanalysis, **f.**–**i.**) for surface air temperature over Germany. The downscaling framework successfully reconstructs fine-scale spatial patterns that are absent in the coarse input, closely matching the ground truth.

from the `atlite` package (hub-height extrapolation and technology power curves). $P_{i,k}^{\max}$ is the installed capacity assigned to that cell and technology, respectively.

Installed capacity is spatially allocated within Germany, proportional to the long-term average local resource for each technology. The nominal capacity assigned to cell i and technology k is

$$C_{i,k} = C_k^{\text{tot}} \frac{\overline{\text{CF}}_{i,k}}{\sum_{j \in \mathcal{I}_{\text{ger}}} \overline{\text{CF}}_{j,k}}, \tag{C.2}$$

where C_k^{tot} is the total installed capacity of technology k (national share) and $\overline{\text{CF}}_{i,k}$ is the long-term mean CF for that cell. The index set $\mathcal{I}_{\text{ger}}$ contains all cells within Germany. Cells intersecting the national border are included with an area fraction weight $a_i \in (0, 1)$.

The German (area- and capacity-weighted) combined CF time series is then

$$\text{CF}_{\text{ger}}(t) \;=\; \sum_k \sum_{i \in \mathcal{I}_{\text{ger}}} w_{i,k}\, CF_{i,k}(t), \qquad w_{i,k} \;=\; \frac{C_{i,k}}{\sum_k \sum_{i \in \mathcal{I}_{\text{ger}}} C_{i,k}}, \tag{C.3}$$

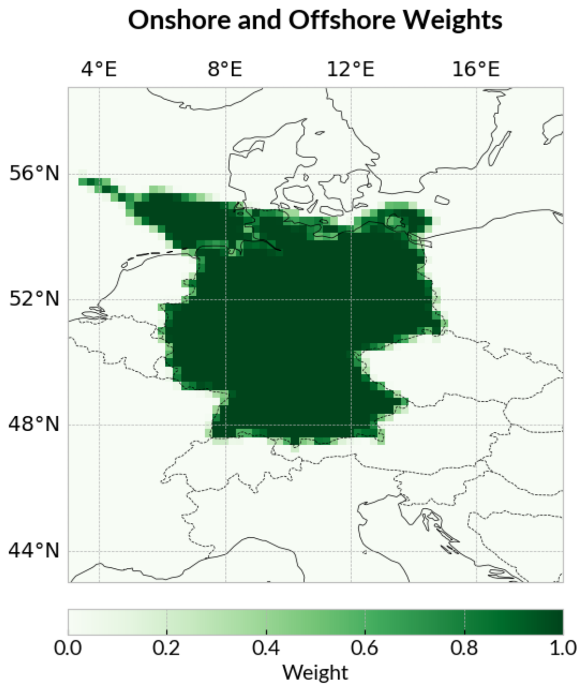


Figure C.3: **Visualization of the contribution of each grid cell to the total CF time series in Germany.** This figure shows the spatial distribution of the weights assigned to each 0.25° grid cell within Germany when computing the national average CF time series. Cells at the border of Germany are only partially included in the average, and their contribution is weighted by the area of the cell that is within Germany.

such that the weights sum to one. The weighting factors are show in Figure C.3. The total installed capacities per technology are taken from the German Federal Network Agency (Bundesnetzagentur) for the year 2024: 60.4 GW onshore wind, 7.7 GW offshore wind, and 66.5 GW PV [28].

The capacity factors for the different technologies are shown in Figure C.4.

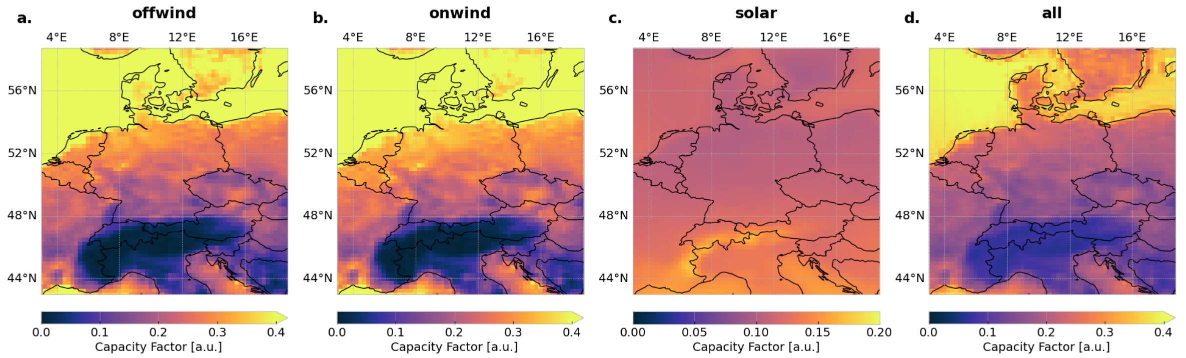


Figure C.4: **Capacity factors for wind and solar energy in Germany.** This figure shows the spatial distribution of the capacity factors for wind and solar energy across each 0.25° grid cell within Germany. Panel **a.** shows the offshore wind capacity factors (only over sea areas), **b.** the onshore wind capacity factors, **c.** the solar capacity factors, and **d.** the weighted mean capacity factor, where the offshore wind is only considered over sea grid cells.

In Figure C.5, we compare time series of combined wind and solar capacity factors at different spatial and temporal resolutions, demonstrating the importance of high-resolution data and bias correction.

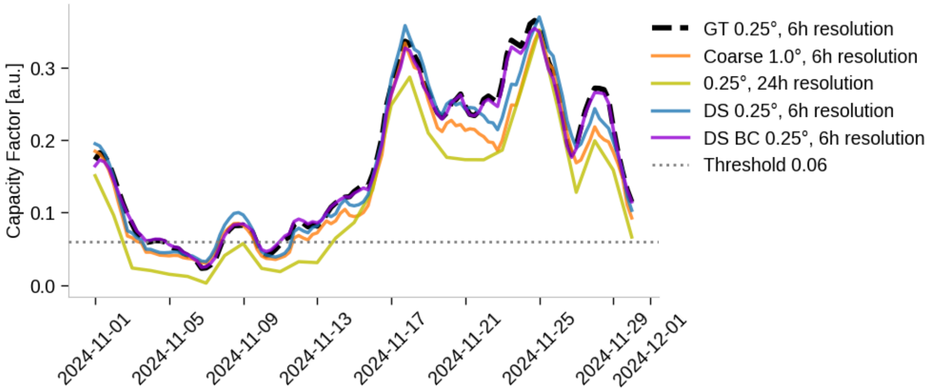


Figure C.5: **Time series of capacity factors for combined wind and solar energy in Germany for different spatial and temporal resolutions.** Different resolutions capture varying degrees of temporal variability, with higher resolutions providing more detailed representations of short-term fluctuations. We compare the time series of capacity factors for combined wind and solar energy in Germany for different spatial and temporal resolutions. The evaluation period spans November 1, 2024, to December 1, 2024, using test data (not included during model training). The dashed black line represents the ground truth (ERA5 reanalysis data at 0.25° spatial and hourly temporal resolution). The orange line shows results for a coarser spatial resolution (1°) but the same temporal resolution (6 hourly). The yellow line shows results for the same spatial resolution (0.25°) but a coarser temporal resolution (daily). The light blue line shows results for the downscaled coarse spatial (1°) and temporal (daily) resolution. The purple line shows the output from the bias corrected downscaling framework, averaged over ten samples drawn from the generative model.

C.3 Future Dunkelflaute events

In Figure C.6, we assess the inter-model variability across the ensemble for the spatial distribution of low capacity factors (<6%) per grid cell for Germany and surrounding regions. We find that the standard deviation across the ensemble is of similar magnitude as the differences between the ensemble mean and the ERA5 reference period, indicating that the uncertainty across models is substantial and indicating that the wind fields are highly variable.

C.4 Time-conditional DM training

The downscaling framework revolves around a conditional diffusion model that receives temporal information *during training* in order to inform the model about periodic effects. Concretely, the vector field that guides the generative process depends additionally on a time stamp including the hour of day (t_{HOD}) and the month (t_{M}):

$$s^{(\theta)}(\mathrm{Z}(\tau), \tau, t_{\mathrm{HOD}}, t_{\mathrm{M}}). \tag{C.4}$$

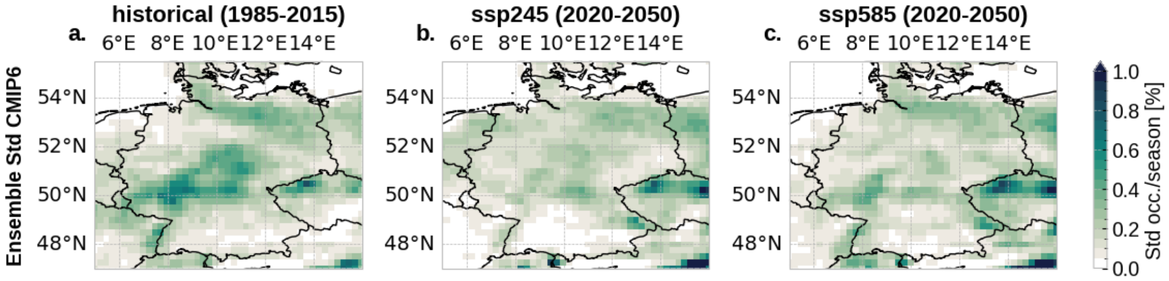


Figure C.6: **Standard Deviation of local risk of prolonged (48 hours) low capacity factors (<6%) for different emission scenarios.** The figure illustrates the inter-model variability across the ensemble for the spatial distribution of low capacity factors (<6%) per grid cell for Germany and surrounding regions. The *left* plot presents the downscaled CMIP6 ensemble mean for 1980-2014. The *middle* and *right* plot display projected Dunkelflaute occurrences standard deviation under the ssp245 (optimistic emission reduction) and ssp585 (high emission) scenarios, respectively.

The generative process that is guided by Equation (C.4) approximately samples from a conditional distribution

$$p(\mathrm{Z}(0) \mid t_{\mathrm{HOD}}, t_{\mathrm{M}}). \tag{C.5}$$

The conditional model is trained following the "classifier-free guidance" mechanism proposed by Ho et al. [95]. This requires training the flow model $v^{(\theta)}$ on pairs of simulations and corresponding `MMHH`-time stamps. The network learns an additional embedding for the timestamps, which it provides as input in addition to the perturbed state $\mathrm{Z}(\tau)$ and the noise level τ.

Conditioning the trained diffusion model on coarse climate simulations Y follows the mechanism used by Schmidt et al. [186, Eq. (5), Supplementary Section 6]. This finally yields an approximation to the posterior score

$$s^{(\theta)}_{t-k:t+k} \approx \nabla_{\mathrm{Z}} \log p(\mathrm{Z} \mid \mathrm{Y}, t_{\mathrm{HOD}}, t_{\mathrm{M}}). \tag{C.6}$$

We found that including additional temporal information ($t_{\mathrm{HOD}}, t_{\mathrm{M}}$) greatly facilitated the learning task. Furthermore, conditioning the model directly on temporal information makes sampling temporally aligned predictions possible, even when the coarse external information Y are long-term aggregates that span a complete diurnal cycle or more (read: climate simulations with $\geq$ daily temporal resolution).