Statistics and Computing (2023) 33:37
https://doi.org/10.1007/s11222-023-10212-8

ORIGINAL PAPER

Probabilistic time series forecasts with autoregressive transformation models

David Rügamer[1,2,3,4] · Philipp F.M. Baumann[5] · Thomas Kneib[6] · Torsten Hothorn[7]

Received: 7 July 2022 / Accepted: 23 November 2022 / Published online: 4 February 2023

Abstract
Probabilistic forecasting of time series is an important matter in many applications and research fields. In order to draw conclusions from a probabilistic forecast, we must ensure that the model class used to approximate the true forecasting distribution is expressive enough. Yet, characteristics of the model itself, such as its uncertainty or its feature-outcome relationship are not of lesser importance. This paper proposes Autoregressive Transformation Models (ATMs), a model class inspired by various research directions to unite expressive distributional forecasts using a semi-parametric distribution assumption with an interpretable model specification. We demonstrate the properties of ATMs both theoretically and through empirical evaluation on several simulated and real-world forecasting datasets.

Keywords Semi-parametric models · Conditional density estimation · Distributional regression · Normalizing flows

1 Introduction

Conditional models describe the conditional distribution $F_{Y|x}(y \mid \boldsymbol{x})$ of an outcome Y conditional on observed features $\boldsymbol{x}$ (see, e.g., Jordan et al. 20). Instead of modeling the complete distribution of $Y \mid \boldsymbol{x}$, many approaches focus on modeling a single characteristic of this conditional distribution. Predictive models, for example, often focus on predicting the average outcome value, i.e., the expectation of the conditional distribution. Quantile regression [25], which is used to model specific quantiles of $Y \mid \boldsymbol{x}$, is more flexible in explaining the conditional distribution by allowing (at least theoretically) for arbitrary distribution quantiles. Various other approaches allow for an even richer explanation by, e.g., directly modeling the distribution's density $f_{Y|\boldsymbol{x}}$ and thus the whole distribution $F_{Y|x}(y \mid \boldsymbol{x})$. Examples include mixture density networks [5] in machine learning, or, in general, probabilistic modeling approaches such as Gaussian processes or graphical models [33]. In statistics and econometrics, similar approaches exist, which can be broadly characterized as distributional regression (DR) approaches [6,11,39,48]. Many of these approaches can also be regarded as conditional density estimation (CDE) models.

Modeling $F_{Y|x}(y \mid \boldsymbol{x})$ is a challenging task that requires balancing the representational capacity of the model (the expressiveness of the modeled distribution) and its risk for overfitting. While the inductive bias introduced by parametric methods can help to reduce the risk of overfitting and is a basic foundation of many autoregressive models, their expressiveness is potentially limited by this distribution assumption (cf. Fig. 1).

Our contributions

In this work, we propose a new and general class of semi-parametric autoregressive models for time series analysis

✉ David Rügamer
david@stat.uni-muenchen.de

Philipp F.M. Baumann
baumann@kof.ethz.ch

Thomas Kneib
tkneib@uni-goettingen.de

Torsten Hothorn
torsten.hothorn@uzh.ch

1 Department of Statistics, TU Dortmund, Dortmund, Germany

2 Institute of Statistics, RWTH Aachen, Aachen, Germany

3 Department of Statistics, LMU Munich, Munich, Germany

4 Munich Center for Machine Learning (MCML), LMU Munich, Munich, Germany

5 KOF Swiss Economic Institute, ETH Zurich, Zurich, Switzerland

6 Chair of Statistics, University of Goettingen, Goettingen, Germany

7 Epidemiology, Biostatistics and Prevention Institute, University of Zurich, Zurich, Switzerland

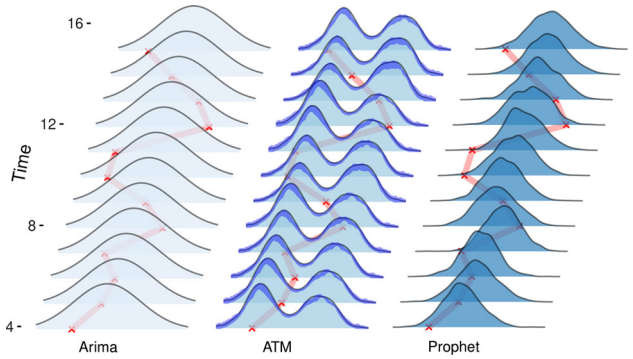

Fig. 1 Exemplary comparison of probabilistic forecasting approaches with the proposed method (ATM; with its uncertainty depicted by the darker shaded area) for a given time series (red line). While other methods are not expressive enough and tailored toward a simple unimodal distribution, our approach allows for complex probabilistic forecasts (here the data-generating process is a bimodal distribution where the inducing mixture variable is unknown to all methods). (Color figure online)

called *autoregressive transformation models* (ATMs; Sect. 3) that learn expressive distributions based on interpretable parametric transformations. ATMs can be seen as a generalization of autoregressive models. We study the autoregressive transformation of order p (AT(p)) in Sect. 4 as the closest neighbor to a parametric autoregressive model, and derive asymptotic results for estimated parameters in Sect. 4.2. Finally, we provide evidence for the efficacy of our proposal both with numerical experiments based on simulated data and by comparing ATMs against other existing time series methods.

2 Background and related work

Approaches that model the conditional density can be distinguished by their underlying distribution assumption. Approaches can be parametric, such as mixture density networks [5] for conditional density estimation and then learn the parameters of a pre-specified parametric distribution or non-parametric such as Bayesian non-parametrics [9]. A third line of research that we describe as semi-parametric, are approaches that start with a simple parametric distribution assumption F_Z and end up with a far more flexible distribution $F_{Y|\boldsymbol{x}}$ by transforming F_Z (multiple times). Such approaches have sparked great interest in recent years, triggered by research ideas such as density estimation using non-linear independent components estimation or real-valued non-volume preserving transformations [8]. A general notion of such transformations is known as normalizing flow (NF; Papamakarios et al. 36), where realizations $\boldsymbol{z} \sim F_Z$ of an error distribution F_z are transformed to observations $\boldsymbol{y}$ via

$$\boldsymbol{y} = h_k \circ h_{k-1} \circ \cdots \circ h_1(\boldsymbol{z}) \tag{1}$$

using k transformation functions. Many different approaches exist to define expressive flows. These are often defined as a chain of several transformations or an expressive neural network and allow for universal representation of $F_{Y|\boldsymbol{x}}$ [36]. Autoregressive models (e.g., Bengio and Bengio 3, Uria et al. 45) for distribution estimation of continuous variables are a special case of NFs, more precisely autoregressive flows (AFs; Kingma et al. 23, Papamakarios et al. 35), with a single transformation.

Transformation models

Transformation models (TMs; Hothorn et al. 17), a similar concept to NFs, only consist of a single transformation and thereby better allow theoretically studying model properties. The transformation in TMs is chosen to be expressive enough on its own and comes with desirable approximation guarantees. Instead of a transformation from $\boldsymbol{z}$ to $\boldsymbol{y}$, TMs define an inverse flow $h(\boldsymbol{y}) = \boldsymbol{z}$. The key idea of TMs is that many well-known statistical regression models can be represented by a base distribution F_Z and some transformation function h. Prominent examples include linear regression or the Cox proportional hazards model [7], which can both be seen as a special case of TMs [17]. Various authors have noted the connection between autoregressive models and NFs (e.g., Papamakarios et al. 36) and between TMs and NFs (e.g., Sick et al. 43). Advantages of TMs and conditional TMs (CTMs) are their parsimony in terms of parameters, interpretability of the input-output relationship, and existing theoretical results [18]. While mostly discussed in the statistical literature, various recent TM advancements have been also proposed in the field of machine learning (see, e.g., Van Belle et al. 46) and deep learning (see, e.g., Baumann et al. 2, Kook et al. 26,27).

Time series forecasting

In time series forecasting, many approaches rely on autoregressive models, with one of the most commonly known linear models being autoregressive (integrated) moving average (AR(I)MA) models (see, e.g., Shumway et al. 42). Extensions include the bilinear model of [14], [38], or the Markov switching autoregressive model by [15]. Related to these autoregressive models are stochastic volatility models [21] building upon the theory of stochastic processes. In probabilistic forecasting, Bayesian model averaging [37] and distributional regression forecasting [41] are two further popular approaches while many other Bayesian and non-Bayesian techniques exist (see, e.g., Gneiting and Katzfuss 12, for an overview).

2.1 Transformation models

Parametrized transformation models as proposed by [17], [18] are likelihood-based approaches to estimate the CDF F_Y of Y. The main ingredient of TMs is a monotonic transformation function h to convert a simple base distribution F_Z to a more complex and appropriate CDF F_Y. Conditional TMs (CTMs) work analogously for the conditional distribu-

tion of Y given features $\boldsymbol{x} \in \chi$ from feature space χ:

$$F_{Y|\boldsymbol{x}}(y) = \mathbb{P}(Y \leq y \mid \boldsymbol{x}) = F_Z(h(y \mid \boldsymbol{x})). \tag{2}$$

CTMs learn $h(y \mid \boldsymbol{x})$ from the data, i.e., estimate a model for the (conditional) aleatoric uncertainty. A convenient parameterization of h for continuous Y are Bernstein polynomials (BSPs; Farouki 10) with order M (usually $M \ll 50$). BSPs are motived by the Bernstein approximation [4] with uniform convergence guarantees for $M \to \infty$, while also being computationally attractive with only $M + 1$ parameters. BSPs further have easy and analytically accessible derivatives, which makes them a particularly interesting choice for the change of random variables. We denote the BSP basis by $\boldsymbol{a}_M : \Xi \mapsto \mathbb{R}^{M+1}$ with sample space Ξ. The transformation h is then defined as $h(y \mid \boldsymbol{x}) = \boldsymbol{a}_M(y)^\top \boldsymbol{\vartheta}(\boldsymbol{x})$ with feature-dependent basis coefficients $\boldsymbol{\vartheta}$. This can be seen as an evaluation of y based on a mixture of Beta densities $f_{Be(\kappa,\mu)}$ with different distribution parameters κ, μ and weights $\boldsymbol{\vartheta}(\boldsymbol{x})$:

$$\boldsymbol{a}_M(y)^\top \boldsymbol{\vartheta}(\boldsymbol{x}) = \frac{\sum_{m=0}^{M} \vartheta_m(\boldsymbol{x}) f_{Be(m+1, M-m+1)}(\tilde{y})}{M+1}, \tag{3}$$

where $\tilde{y}$ is a rescaled version of y to ensure $\tilde{y} \in [0, 1]$. Restricting $\vartheta_m > \vartheta_{m-1}$ for $m = 1, \ldots, M + 1$ guarantees monotonicity of h and thus of the estimated CDF. Roughly speaking, using BSPs of order M, allows to model the polynomials of degree M of y.

2.2 Model definition

The transformation function h can include different data dependencies. One common choice [2,16] is to split the transformation function into two parts

$$h(y \mid \boldsymbol{x}) = h_1(y, \boldsymbol{x}) + h_2(\boldsymbol{x}) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}(\boldsymbol{x}) + \beta(\boldsymbol{x}), \tag{4}$$

where $\boldsymbol{a}(y)$ is a pre-defined basis function such as the BSP basis (omitting M for readability in the following), $\boldsymbol{\vartheta} : \chi_{\vartheta} \mapsto \mathbb{R}^{M+1}$ a conditional parameter function defined on $\chi_{\vartheta} \subseteq \chi$ and $\beta(\boldsymbol{x})$ models a feature-induced shift in the transformation function. The flexibility and interpretability of TMs stems from the parameterization

$$\boldsymbol{\vartheta}(\boldsymbol{x}) = \sum_{j=1}^{J} \boldsymbol{\Gamma}_j . \boldsymbol{b}_j(\boldsymbol{x}), \tag{5}$$

where the matrix $\boldsymbol{\Gamma}_j \in \mathbb{R}^{(M+1)\times O_j}$, $O_j \geq 1$, subsumes all trainable parameters and represents the effect of the interaction between the basis functions in $\boldsymbol{a}$ and the chosen predictor terms $\boldsymbol{b}_j : \chi_{b_j} \mapsto \mathbb{R}^{O_j}$, $\chi_{b_j} \subseteq \chi$. The predictor terms $\boldsymbol{b}_j$ have a role similar to base learners in boosting and represent simple learnable functions. For example, a predictor term can be the jth feature, $\boldsymbol{b}_j(\boldsymbol{x}) = x_j$, and $\boldsymbol{\Gamma}_j \in \mathbb{R}^{(M+1)\times 1}$ describes the linear effect of this feature on the $M + 1$ basis coefficients, i.e., how the feature x_j relates to the density transformation from Z to $Y \mid \boldsymbol{x}$. Other structured non-linear terms such as splines allow for interpretable lower-dimensional non-linear relationships. Various authors also proposed neural network (unstructured) predictors to allow potentially multidimensional feature effects or to incorporate unstructured data sources [2,26,43]. In a similar fashion, $\beta(\boldsymbol{x})$ can be defined using various structured and unstructured predictors.

Interpretability

Relating features and their effect in an additive fashion allows to directly assess the impact of each feature on the transformation and also whether changes in the feature just shift the distribution in its location or if the relationship also transforms other distribution characteristics such as variability or skewness (see, e.g., Baumann et al. 2, for more details).

Relationship with autoregressive flows

In the notation of AFs, $h^{-1}(\cdot)$ is known as *transformer*, a parameterized and bijective function. By the definition of (4), the transformer in the case of TMs is represented by the basis function $\boldsymbol{a}(\cdot)$ and parameters $\boldsymbol{\vartheta}$. In AFs, these transformer parameters are learned by a *conditioner*, which in the case of TMs are the functions $\boldsymbol{b}_j$. In line with the assumptions made for AFs, these conditioners in TMs do not need to be bijective functions themselves.

3 Autoregressive transformations

Inspired by TMs and AFs, we propose autoregressive transformation models (ATMs). Our work is the first to adapt TMs for time series data and thereby lays the foundation for future extensions of TMs for time series forecasting. The basic idea is to use a parameter-free base distribution F_Z and transform this distribution in an interpretable fashion to obtain $F_{Y|\boldsymbol{x}}$. One of the assumptions of TMs is the stochastic independence of observations, i.e., $Y_i \mid \boldsymbol{x}_i \perp Y_j \mid \boldsymbol{x}_j, i \neq j$. When Y is a time series, this assumption does clearly not hold. In contrast, this assumption is not required for AFs.

Let $t \in \mathcal{T} \subseteq \mathbb{N}_0$ be a time index for the time series $(Y_t)_{t\in\mathcal{T}}$. Assume

$$Y_t \mid \mathcal{F}_{t-1} \sim G(Y_{t-1}, \ldots, Y_{t-p}; \boldsymbol{\theta}) \tag{6}$$

for some $p \in \{1, \ldots, t\}$, distribution G, parameter $\boldsymbol{\theta} \in \Theta$ with compact parameter space $\Theta \subset \mathbb{R}^v$ and filtration $\mathcal{F}_s$, $s \in \mathcal{T}$, $s < t$, on the underlying probability space. Assume that the joint distribution of $Y_t, Y_{t-1}, \ldots, Y_1$ possesses the Markov property with order p, i.e., the joint distribution, expressed through its absolutely continuous density f, can

be rewritten as product of its conditionals with p lags:

$$f(y_t, \dots, y_1 \mid \boldsymbol{x}) = \prod_{s=p+1}^{t} f(y_s \mid y_{s-1}, \dots, y_{s-p}, \boldsymbol{x}). \quad (7)$$

We use $\boldsymbol{x}$ to denote (potentially time-varying) features that are additional (exogenous) features. Their time-dependency is omitted for better readability here and in the following. Given this autoregressive structure, we propose a time-dependent transformation h_t that extends (C)TMs to account for filtration and time-varying feature information. By modeling the conditional distribution of all time points in a flexible manner, ATMs provide an expressive way to account for aleatoric uncertainty in the data.

Definition 1 Autoregressive transformation models Let $h_t, t \in \mathcal{T}$, be a time-dependent monotonic transformation function and F_Z the parameter-free base distribution as in Definition 1 in the Supplementary Material. We define autoregressive transformation models as follows:

$$\begin{aligned} \mathbb{P}(Y_t \leq y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x}) &= F_{Y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x}}(y_t) \\ &= F_Z(h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})). \end{aligned} \quad (8)$$

This can be seen as the natural extension of (2) for time series data with autoregressive property and time-varying transformation function h_t. In other words, (8) says that after transforming y_t with h_t, its conditional distribution follows the base distribution F_Z, or vice versa, a random variable $Z \sim F_Z$ can be transformed to follow the distribution $Y_t \mid \boldsymbol{x}$ using h_t^{-1}.

Relationship with autoregressive models and autoregressive flows

Autoregressive models (AMs; Bengio and Bengio 3) and AFs both rely on the factorization of the joint distribution into conditionals as in (7). Using the CDF of each conditional in (7) as transformer in an AF, we obtain the class of AMs [36]. AMs and ATMs are thus both (inverse) flows using a single transformation, but with different transformers and, as we will outline in Sect. 3.2, also with different conditioners.

Stationarity

For TMs as defined in (2), strict stationarity is given in the transformed probability space, that is for $Z_t := h(Y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})$ it holds $Z_t \overset{iid}{\sim} F_Z \, \forall t \in \mathcal{T}$. In the general setup of ATMs, strict stationarity is, however, not required for $(Y_t)_{t \in \mathcal{T}}$. Instead, the time-varying transformation h_t is assumed to be expressive enough to map the possibly non-stationary process $(Y_t)_{t \in \mathcal{T}}$ to a time-independent distribution (and hence stationary) process $(Z_t)_{t \in \mathcal{T}}$.

3.1 Likelihood-based estimation

Based on (7), (8) and the change of variable theorem, the likelihood contribution of the tth observation y_t in ATMs is given by

$$\begin{aligned} &f_{Y_t \mid \boldsymbol{x}}(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x}) \\ &= f_Z(h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})) \cdot \left| \frac{\partial h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})}{\partial y_t} \right| \end{aligned}$$

and the full likelihood for T observations from random variable $\boldsymbol{Y}$ thus by

$$\begin{aligned} &f_{\boldsymbol{Y} \mid \boldsymbol{x}}(Y_T, \dots, Y_1 \mid \mathcal{Y}_0, \boldsymbol{x}) \\ &= \prod_{t=1}^{T} \left\{ f_Z(h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})) \cdot \left| \frac{\partial h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})}{\partial y_t} \right| \right\}, \end{aligned} \quad (9)$$

where $\mathcal{Y}_0 = (y_0, \dots, y_{-p+1})$ are known finite starting values and $\mathcal{F}_0$ only contains these values. Based on (9), we define the loss of all model parameters $\boldsymbol{\theta}$ as negative log-likelihood $-\ell(\boldsymbol{\theta}) := -\log f_{Y \mid x}(Y_T, \dots, Y_1 \mid \mathcal{Y}_0, \boldsymbol{x})$ given by

$$\begin{aligned} &-\sum_{t=1}^{T} \Big\{ \log f_Z(h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})) \\ &\quad + log \left| \frac{\partial h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})}{\partial y_t} \right| \Big\}, \end{aligned} \quad (10)$$

and use (10) to train the model.

As for AFs, many special cases can be defined from the above definition and more concrete structural assumptions for h_t make ATMs an interesting alternative to other methods in practice. We will elaborate on meaningful structural assumptions in the following.

3.2 Structural assumptions

In CTMs, the transformation function h is usually decomposed as $h(y \mid \boldsymbol{x}) = h_1(y \mid \boldsymbol{x}) + h_2(\boldsymbol{x})$, where h_1 is a function depending on y and h_2 is a transformation-shift function depending only on $\boldsymbol{x}$. For time-varying transformations h_t our fundamental idea is that the outcome y_t shares the same transformation with its filtration $\mathcal{F}_{t-1}$, i.e., the lags $\mathcal{Y}_t = (y_{t-1}, \dots, y_{t-p})$. In other words, a transformation applied to the outcome must be equally applied to its predecessor in time to make sense of the autoregressive structural assumption. An appropriate transformation structure can thus be described by

$$\begin{aligned} &h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x}) \\ &\quad = h_{1t}(y_t \mid \boldsymbol{x}) + h_{2t}((h_{1t} \odot \mathcal{Y}_t \mid \mathcal{F}_{t-1}, \boldsymbol{x}) \mid \boldsymbol{x}) \\ &\quad =: \lambda_{1t} + \lambda_{2t}, \end{aligned} \quad (11)$$

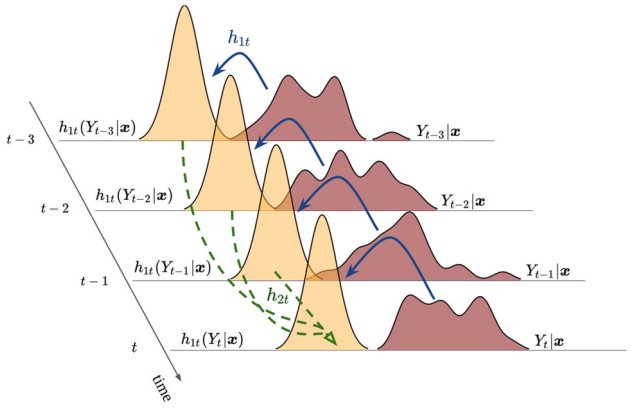

Fig. 2 Illustration of a transformation process induced by the structural assumption of Sect. 3.2. The original data history $\mathcal{F}_{t-1}$ (red) is transformed into a base distribution (orange) using the transformation h_{1t} (solid blue arrow) and then further transformed using h_{2t} (dashed green arrow) to match the transformed distribution of the current time point t. (Color figure online)

for $t \in \mathcal{T}$, where $\odot$ indicates the element-wise application of h_{1t} to all lags in $\mathcal{Y}_t$. In other words, ATMs first apply the same transformation h_{1t} to y_t and individually to $y_{t-1}, y_{t-2}, \ldots$, and then further consider a transformation function h_{2t} to shift the distribution (and thereby potentially other distribution characteristics) based on the transformed filtration. While the additivity assumption of λ_{1t} and λ_{2t} seems restrictive at first glance, the imposed relationship between y_t and $\mathcal{Y}_t$ only needs to hold in the transformed probability space. For example, h_{1t} can compensate for a multiplicative autoregressive effect between the filtration and y_t by implicitly learning a log-transformation (cf. Sect. 5.1). At the same time, the additivity assumption offers a nice interpretation of the model, also depicted in Fig. 2: After transforming y_t and $\mathcal{Y}_t$, (11) implies that training an ATM is equal to a regression model of the form $\lambda_{1t} = \lambda_{2t} + \varepsilon$, with additive error term $\varepsilon \sim F_Z$ (cf. Proposition 1 in Supplementary Material A.2). This also helps explaining why only λ_{2t} depends on $\mathcal{F}_{t-1}$: if λ_{1t} also involves $\mathcal{F}_{t-1}$, ATMs would effectively model the joint distribution of the current time point and the whole filtration, which in turn contradicts the Markov assumption (7).

Specifying h_{1t} very flexible clearly results in overfitting. As for CTMs, we use a feature-driven basis function representation $h_{1t}(y_t \mid \boldsymbol{x}) = \boldsymbol{a}(y_t)^\top \boldsymbol{\vartheta}(\boldsymbol{x})$ with BSPs $\boldsymbol{a}$ and specify their weights as in (5). The additional transformation h_{2t} ensures enough flexibility for the relationship between the transformed response and the transformed filtration, e.g., by using a non-linear model or neural network. An interesting special case arises for linear transformations in h_{2t}, which we elaborate in Sect. 4 in more detail.

Interpretability

The three main properties that make ATMs interpretable are 1) their additive predictor structure as outlined in (5); 2) the clear relationship between features and the outcome through the BSP basis, and 3) ATM's structural assumption as given in (11). As for (generalized) linear models, the additivity assumption in the predictor allows interpreting feature influences through their partial effect ceteris paribus. On the other hand, choices of M and F_Z will influence the relationship between features and outcome by inducing different types of models. A normal distribution assumption for F_Z and $M = 1$ will turn ATMs into an additive regression model with Gaussian error distribution (see also Sect. 4). For $M > 1$, features in h_1 will also influence higher moments of $Y \mid \boldsymbol{x}$ and allow more flexibility in modeling $F_{Y|\boldsymbol{x}}$. For example, a (smooth) monotonously increasing feature effect will induce rising moments of $Y \mid \boldsymbol{x}$ with increasing feature values. Other choices for F_Z such as the logistic distribution also allow for easy interpretation of feature effects (e.g., on the log-odds ratio scale; see Kook et al. 26). Finally, the structural assumption of ATMs enforces that the two previous interpretability aspects are consistent over time. We will provide an additional illustrative example in Sect. 5.2, further explanations in Supplementary Material B, and refer to [17] for more details on interpretability of CTMs.

Implementation

In order to allow for a flexible choice of transformation functions and predictors $\boldsymbol{b}_j$, we propose to implement ATMs in a neural network and use stochastic gradient descent for optimization. While this allows for complex model definitions, there are also several computational advantages. In a network, weight sharing for h_{1t} across time points is straightforward to implement and common optimization routines such as Adam [22] prove to work well for ATMs despite the monotonicity constraints required for the BSP basis. Furthermore, as basis evaluations for a large number of outcome lags in $\mathcal{F}_{t-1}$ can be computationally expensive for large p (with space complexity $\mathcal{O}(t \cdot M \cdot p)$) and add M additional columns per lag to the feature matrix, an additional advantage is the dynamic nature of mini-batch training. In this specific case, it allows for evaluating the bases only during training and separately in each mini-batch. It is therefore never required to set up and store the respective matrices.

Relationship to implicit copulas

An anonymous reviewer pointed out the potential relationship of ATMs and implicit copulas [44]. We start with a general formulation of multivariate transformation models similar to [24], first without the time series context, by defining a componentwise bijective, strictly monotonically increasing multivariate transformation function $\mathfrak{h} : \mathbb{R}^T \to \mathbb{R}^T$ mapping $\boldsymbol{Y} = (Y_1, \ldots, Y_T)^\top$ to a vector $\boldsymbol{Z} \in \mathbb{R}^T$ with absolutely continuous random variable entries $Z_1 \sim F_{Z_t}, \ldots, Z_T \sim F_{Z_T}$, marginal distributions $F_{Z_t}, t = 1, \ldots, T$, joint distribution $F_{\boldsymbol{Z}}$, and define the distribution of $\boldsymbol{Y}$ by $F_{\boldsymbol{Y}}(\boldsymbol{y}) = F_{\boldsymbol{Z}}(z) := F_{\boldsymbol{Z}}(\mathfrak{h}(\boldsymbol{y}))$. We assume that $\partial \mathfrak{h}(\boldsymbol{y}) / \partial \boldsymbol{y}$ is lower-diagonal. This is a com-

mon assumption used in multivariate TMs, AFs, and also ATMs. Then, following the change of variable theorem, $f_{\boldsymbol{Y}}(\boldsymbol{y}) = f_{\boldsymbol{Z}}(z) \cdot \prod_{t=1}^{T} |\partial \mathfrak{h}_t(\boldsymbol{y})/\partial y_t|$. Now let the entries of $\mathfrak{h}$ be defined by $\mathfrak{h}_t(\boldsymbol{y}) = F_{Z_t}^{-1} F_{Y_t}(y_t)$, where F_{Y_t} is an (arbitrary) marginal distribution of Y_t and $F_{Z_t}^{-1}$ the quantile function of F_{Z_t}. It follows that $F_{\boldsymbol{Y}}$ is now exactly defined as in the case of implicit copulas, where the copula function $C(\boldsymbol{u}) = F_{\boldsymbol{Z}}(F_{Z_1}^{-1}(u_1), \ldots, F_{Z_T}^{-1}(u_T))$ is defined via the transformation model's error distribution $F_{\boldsymbol{Z}}$ together with the outer quantile function transformation $F_{Z_t}^{-1}$ in $\mathfrak{h}_t$, and $\boldsymbol{u} = (u_1, \ldots, u_T)^\top$ in this case corresponds to the inner transformation $F_{Y_t}(y_t)$ in $\mathfrak{h}_t$. Further, as $\partial \mathfrak{h}_t(y_t)/\partial y_t = f_{Y_t}(y_t)/f_{Z_t}(z_t)$ in the transformation model, the equivalence of the copula density c of this implicitly defined copula C can also be directly confirmed. ATMs, however, assume $Z_t \overset{iid}{\sim} F_Z$ as the transformation $\mathfrak{h}$ already accounts for all dependencies in the time series. This is in contrast to implicit copulas, where $\mathfrak{h}$ is defined rather simple and $F_{\boldsymbol{Z}}$ is more complex (does not factorize into the product of marginals).

The previously discussed connections will allow to better understand ATMs, and also to elaborate on the question of stationarity of ATMs in the future, which is extensively discussed in copula literature (see, e.g., Nagler et al. 34, Smith 44).

4 AT(p) Model

A particular interesting special case of ATMs is the AT(p) model. This model class is a direct extension of the well-known autoregressive model of order p (short AR(p) model; Shumway et al. 42) to transformation models.

Definition 2 AT(p) model We define the AT(p) model, a special class of ATMs, by setting $h_{1t}(y_t \mid \boldsymbol{x}) = h_1(y_t \mid \boldsymbol{x}) = \boldsymbol{a}(y_t)^\top \boldsymbol{\vartheta}(\boldsymbol{x})$, and $h_{2t}(\mathcal{F}_{t-1}, \boldsymbol{x}) = h_2(\mathcal{F}_{t-1}, \boldsymbol{x}) = \sum_{j=1}^{p} \phi_j h_1(y_{t-j}) + r(\boldsymbol{x})$, i.e., an autoregressive shift term with optional exogenous remainder term $r(\boldsymbol{x})$.

As for classical time series approaches, ϕ_j are the regression coefficients relating the different lags to the outcome and r is a structured model component (e.g., linear effects) of exogenous features that do not vary over time.

4.1 Model details

The AT(p) model is a very powerful and interesting model class for itself, as it allows to recover the classical time series AR(p) model when setting $M = 1$, $\boldsymbol{\vartheta}(\boldsymbol{x}) \equiv \boldsymbol{\vartheta}$ and $r(\boldsymbol{x}) \equiv 0$ (see Proposition 2 in Supplementary Material A for a proof of equivalence). But it can also be extended to more flexible autoregressive models in various directions. We can increase M to get a more flexible density, allowing us to deviate from the base distribution assumption F_Z, e.g., to relax the normal distribution assumption of AR models. Alternatively, incorporating exogenous effects into h_{1t} allows to estimate the density data-driven or to introduce exogenous shifts in time series using features $\boldsymbol{x}$ in $r(\boldsymbol{x})$. ATMs can also recover well-known transformed autoregressive models such as the multiplicative autoregressive model [47] as demonstrated in Sect. 5.1. When specifying M large enough, an AT(p) model will, e.g., learn the log-transformation function required to transform a multiplicative autoregressive time series to an additive autoregressive time series on the log-scale. In general, this allows the user to learn autoregressive models without the need to find an appropriate transformation before applying the time series model. This means that the uncertainty about preprocessing steps (e.g., a Box-Cox transformation; Sakia 40) is incorporated into the model estimation, making parts of the pre-processing obsolete for the modeler and its uncertainty automatically available.

Non-linear extensions of AT(p) models can be constructed by modeling $\mathcal{Y}_t$ in h_{2t} non-linearly, allowing ATMs to resemble model classes such as non-linear AR models with exogenous terms (e.g., Lin et al. 29). In practice, values for p can, e.g., be found using a (forward) hyperparameter search by comparing the different model likelihoods.

4.2 Asymptotic theory

An important yet often neglected aspect of probabilistic forecasts is the epistemic uncertainty, i.e., the uncertainty in model parameters. Based on general asymptotic theory for time series models [30], we derive theoretical properties for AT(p)s in this section.

Let $\boldsymbol{\theta}^*$ be the true value of $\boldsymbol{\theta}$ and interior point of Θ. We define the following quantities involved in standard asymptotic MLE theory: Let $\hat{\boldsymbol{\theta}}_T = \arg\min_\Theta -\ell(\boldsymbol{\theta})$ be the parameter estimator based on Maximum-Likelihood estimation (MLE), $\nabla_T(\boldsymbol{\theta}) = \partial \ell_T(\boldsymbol{\theta})/\partial \boldsymbol{\theta}$, $\mathcal{J}_T(\boldsymbol{\theta}) = -\partial^2 \ell_T(\boldsymbol{\theta})/(\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}')$, $\mathcal{I} = \mathbb{E}_G(\mathcal{J}_T(\boldsymbol{\theta}^*))$ and $\mathfrak{J} = \mathbb{E}_G(\nabla_T(\boldsymbol{\theta}^*)\nabla_T^\top(\boldsymbol{\theta}^*))$. We further state necessary assumptions to apply the theory of [30] for a time series $(Y_t)_{t \in \mathcal{T}}$ with known initial values $\mathcal{Y}_0$ as defined in Sect. 3.

Assumption 1 Assume

(i) $(Y_t)_{t \in \mathcal{T}}$ is strictly stationary and ergodic;
(ii) $\mathbb{E}_G\{\sup_{\boldsymbol{\theta} \in \Theta}[\ell_T(\boldsymbol{\theta})]\} < \infty$ and $\boldsymbol{\theta}^*$ is unique;
(iii) $\nabla_T(\boldsymbol{\theta}^*)$ is a martingale difference w.r.t. $\mathcal{F}_{T-1}$ with $0 < \mathfrak{J} < \infty$;
(iv) $\mathcal{I}$ is positive-definite and for some $\xi > 0$ $\mathbb{E}_G\{\sup_{\boldsymbol{\theta}: ||\boldsymbol{\theta} - \boldsymbol{\theta}^*|| < \xi} ||\mathcal{J}_T(\boldsymbol{\theta})||\} < \infty$.

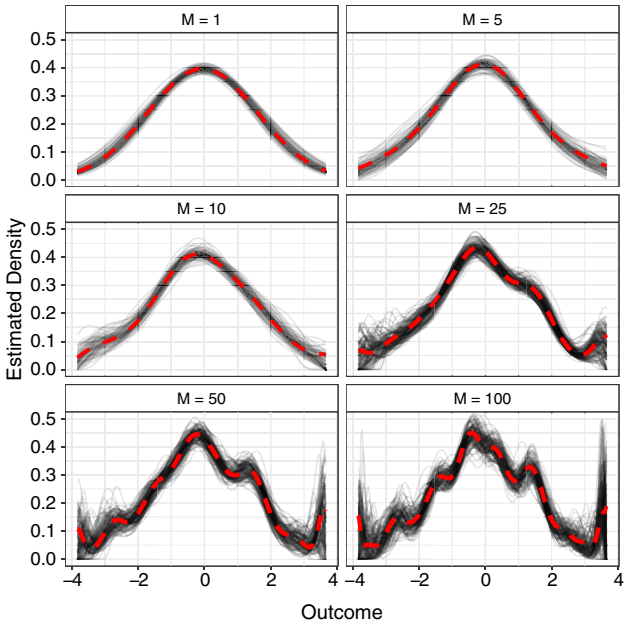

Fig. 3 Aleatoric vs. epistemic uncertainty: Different plots correspond to different orders of the BSP basis M, inducing different amounts of expressiveness and aleatoric uncertainty (and here also multimodality). In each plot, the fitted density is shown in red, and model uncertainties of this density based on the epistemic uncertainty in black. Epistemic uncertainty is generated according to results in Theorem 2 and 3

Assumptions 1 are common assumptions required for many time series models. We require only these and no other assumptions since AT(p)s and non-linear extensions are fully-parameterized time series models. This allows us to derive general statistical inference theory for AT(p) models.

Theorem 1 (Consistency) *If elements in $\mathcal{Y}_0$ are finite and Assumption 1(i) holds, then $\hat{\boldsymbol{\theta}}_T \xrightarrow{a.s.} \boldsymbol{\theta}^*$ for $T \to \infty$.*

As stated in [18], Assumption 1 (ii) holds if $\boldsymbol{a}$ is not arbitrarily ill-posed. In practice, both a finite $\mathcal{Y}_0$ and Assumption 1 (i) are realistic assumptions. Making two additional and also rather weak assumptions (1(iii)–(iv)) allows to derive the asymptotic normal distribution for $\hat{\boldsymbol{\theta}}$.

Theorem 2 (Asymptotic Normality) *If y_0 is finite and Assumptions 1 hold, then for $T \to \infty$,*

$$\hat{\boldsymbol{\theta}}_T = \boldsymbol{\theta}^* + O(\sqrt{(\log\log T)/T})$$

and

$$\sqrt{T}(\hat{\boldsymbol{\theta}}_T - \boldsymbol{\theta}^*) \xrightarrow{D} \mathcal{N}(0, \mathcal{I}^{-1}\mathfrak{J}\mathcal{I}^{-1}).$$

Based on the same assumptions, a consistent estimator for the covariance can be derived.

Theorem 3 (Consistent Covariance Estimator) *For finite y_0 and under Assumptions 1,*

$$\hat{\mathcal{I}}_T = \frac{1}{T}\sum_{t=1}^{T} \mathcal{J}_T(\hat{\boldsymbol{\theta}}_T) \text{ and } \hat{\mathfrak{J}}_T = \frac{1}{T}\sum_{t=1}^{T} \nabla_T(\hat{\boldsymbol{\theta}}_T)\nabla_T^\top(\hat{\boldsymbol{\theta}}_T)$$

are consistent estimators for $\mathcal{I}$ and $\mathfrak{J}$, respectively.

The previous theorems can be proven by observing that the AT(p) model structure and all made assumptions follow the general asymptotic theory for time series models as given in [30]. See Supplementary Material A for details.

Using the above results, we can derive statistically valid UQ. An example is depicted in Fig. 3. Since h is parameterized through $\boldsymbol{\theta}$, it is also possible to derive the so-called structural uncertainty of ATMs, i.e., the uncertainty induced by the discrepancy between the model's CDF $F_{Y|x}(y \mid \boldsymbol{x}; \boldsymbol{\theta})$ and the true CDF $F^*_{Y|x}(y \mid \boldsymbol{x})$ [31]. More specifically, h can be represented using a linear transformation of $\boldsymbol{\theta}$, $h = \boldsymbol{\Upsilon}\boldsymbol{\theta}$, implying the (co-)variance $\boldsymbol{\Upsilon}\mathcal{I}^{-1}\mathfrak{J}(\boldsymbol{\theta}^*)\mathcal{I}^{-1}\boldsymbol{\Upsilon}^\top$ for $\hat{h}$.

Practical application

ATM define the distribution $F_{Y_t|\mathcal{F}_{t-1},x}$ via $F_{Y|\mathcal{F}_{t-1},x} = F_Z \circ h_t$, where h_t is parameterized by $\boldsymbol{\theta}$. In order to assess parameter uncertainty in the estimated density as, e.g. visualized in Fig. 1 and 3, we propose to use a parametric Bootstrap described in detail in Supplementary Material C.

Seasonality

Many forecasting problems exhibit seasonality with a significant impact on the nature of the time series. ATMs allow the modeler to account for this in two different ways. If seasonality manifests itself through a (non-)linear shift in the values of the time series, the function h_{2t} can be used to incorporate seasonal effects by either adding (exogenous) covariates such as dummy variables or seasonal lags for the particular season, or by including a non-linear, seasonal effect of time, e.g., using cyclic penalized splines. If the seasonality also changes the distribution of y_t itself, our approach can be used to its full potential by incorporating the derived time variables in $\boldsymbol{\vartheta}(\boldsymbol{x})$ and thereby allowing a seasonally-varying distribution.

Stationarity

As for the general class of ATMs, the $AT(p)$ model assumes Z_t to be independently and identically distributed according to F_Z. In contrast to ATMs – if we want to make inference statements about $\boldsymbol{\theta}$ as given in 1–3 – we additionally require strict stationarity of the untransformed $(Y_t)_{t\in\mathcal{T}}$. An interesting direction for future research is to relax this assumption.

Model diagnostics

To assess the goodness-of-fit of an ATM/AT(p) model given by $\hat{z}_t := \hat{h}_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})$, the empirical quantiles of $\hat{z}_t$ can be compared with the theoretical quantiles of the error distribution F_Z, e.g., by means of a quantile–quantile plot or a Kolmogorov–Smirnov statistic [17]. If the distribution of

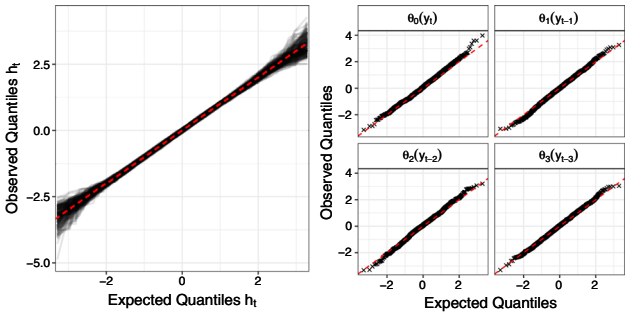

Fig. 4 Empirical evidence for the correctness of our theoretical results on PU: Expected vs. observed quantiles of the transformation function h_t (left; one line per dataset) and model parameters $\boldsymbol{\theta}$ for the different (lagged) transformed outcomes (right; one cross per dataset) based on 1000 simulation replications. The ideal angle bisector is plotted in red

Table 1 Average and standard deviation (brackets) of the MSE (multiplied by 100 for better readability) between estimated and true coefficients in an AR(p) model using our approach on the tampered data (bottom row) and the corresponding oracle based on the true data (Oracle)

	T	$p = 1$	$p = 2$	$p = 4$
Oracle	400	0.33 (0.31)	0.22 (0.19)	0.25 (0.13)
AT(p)		0.52 (0.46)	0.33 (0.3)	0.34 (0.23)
Oracle	800	0.27 (0.34)	0.13 (0.12)	0.13 (0.085)
AT(p)		0.26 (0.36)	0.17 (0.17)	0.18 (0.12)

$\hat{z}_t$ does not match the distribution F_Z, the model is likely to be misspecified. Further, the independence assumption (and therefore stationarity) of $\hat{z}_t$ can be assessed by means of an auto-correlation function plot.

5 Experiments

We will first investigate theoretical properties of ATMs and the validity of statistical inference statements using simulation studies. We then compare our approach against other state-of-the-art methods described in the previous section on probabilistic forecasting tasks in a benchmark study. Additional results can be found in the Supplementary Material D.

5.1 Simulation study

Equivalence and consistency

We first demonstrate Theorem 1 and Proposition 2 in the Supplementary Material, i.e., for growing number of observations AT(p) models can recover AR(p) models when equally specified. We therefore simulate various AR models using lags $p \in \{1, 2, 4\}$, $T \in \{200, 400, 800\}$ and estimate both a classical AR(p) model and an AT(p) model for 20 replications. For the latter, we use the mapping derived in Proposition 2 to obtain the estimated AR coefficients from the AT(p) model. In Table D1 in the Supplementary Material D we compare both models based on their estimated coefficients against the ground truth using the mean squared error (MSE). Results show that the AT(p) model can empirically recover the AR(p) model very well.

Flexibility

Next, we demonstrate how the AT(p) model with $M = 30$ can recover a multiplicative autoregressive process. We therefore generate data using an AR model with different lags p and observations n as before. This time, however, we provide the AT(p) model only with the exponentiated data $\check{y}_t = \exp(y_t)$. This means the model needs to learn the inverse transformation back to y_t itself. Despite having to estimate the log-transformation in addition, the AT(p) model recovers the true model well and, for larger n, is even competitive to the ground truth model (Oracle) that has access to the original non-exponentiated data (cf. Table D2 for an excerpt of the results).

Epistemic uncertainty

In this experiment we validate our theoretical results proposed in Sect. 4.2. As in the previous experiment, we try to learn the log-transformed AR model using an AT($p = 3$) model with coefficients $(0.3, 0.2, 0.1)$. After estimation, we check the empirical distribution of $\hat{\boldsymbol{\theta}}$ and $\hat{h}$ against their respective theoretical one in 1000 simulation replications. Fig. 4 depicts a quantile-quantile plot of the empirical and theoretical distribution for both h and all 4 parameters (intercept and three lag coefficients). The empirical distributions are well aligned with their theoretical distribution as derived in Sect. 4.2, confirming our theoretical results.

5.2 Benchmarks

Finally, we compare our approach to its closest neighbor in the class of additive models, the ARIMA model [19], against a simple Box-Cox transformation (BoxCox), a neural network for mean-variance estimation (MVN) and a mixture density network (MDN; Bishop 5). While there are many further forecasting techniques, especially in deep learning, we purposely exclude more complex machine and deep learning approaches to compare AT(p)s with approaches of similar complexity. More specifically, the different competitors were chosen to derive the following insights: The comparison of the AT(p) model with the ARIMA model will indicate whether relaxing the parametric assumption using TMs can improve performance while both methods take time series lags into account. The comparison of our method with Box-Cox, on the other hand, will show similar performance if there is no relevant information in the lags of the time series. The MVN can potentially learn time series-specific variances but is not given the lagged information as input. A good performance of the MVN will thus indicate heteroscedasticity in the

Table 2 Mean log-scores (higher is better) across 10 different initializations with standard deviations in brackets for each method (columns) and benchmark dataset (rows). Results for ARIMA are based on only one trial as there is typically no stochasticity in its results. The best performing method per data set is highlighted in bold

	ARIMA	AT(p)	Box Cox	MDN	MVN
`elec`	−5.44	−5.35 (0.01)	−8.37 (0.00)	**−5.20** (0.01)	−9.51 (0.00)
`exchange`	0.37	3.50 (0.05)	−0.69 (0.00)	**4.02** (0.12)	−0.70 (0.00)
`m4`	−573.11	**−6.72** (0.07)	−10.7 (0.00)	−6.75 (1.17)	−12.0 (0.00)
`tourism`	−9.78	**−9.38** (0.01)	−11.5 (0.00)	−77.8 (99.5)	−12.7 (0.00)
`traffic`	0.23	**1.09** (0.33)	0.03 (0.00)	1.06 (0.02)	−0.25 (0.00)

data generating process which can, however, be accounted for using a parametric distributional regression approach. Finally, the MDN is an alternative approach to the AT(p) model that tries to overcome the parametric assumption by modeling a mixture of normal distributions. In this benchmark, all comparisons investigate one-step-ahead forecasts. In Section 6, we will discuss and outline the use of ATMs for multi-step-ahead forecasts.

Hyperparameter setup

We define the AT(p) model by using an unconditional ϑ parameter and use the lag structure as well as a time series identifier as a categorical effect in the additive predictor of β. We further investigate different number of BSPs $M \in \{5, 10, 30\}$ and different number of lags $p \in \{1, 2, 3\}$. Model training for all models but the ARIMA model was done using 1000 epochs with early stopping and a batch size of 128. For the MDN, we define 3 mixtures and use the AT(p)'s β as an additive predictor for the mean of every mixture component. The MVN uses the time series identifier to learn individual means and variances. For ARIMA we used the `auto.arima` implementation [19] and performed a stepwise search via the AICc with different starting values for the order of the AR and the MA term. For the AR term, we consider the length of the corresponding forecasting horizon and halve this value. The search space for the MA term started either with 0 or 3. We chose the ARIMA model with the lowest AICc on the validation set. For the `auto.arima` model on the `m4` data, we restrict the observations to be used for model selection to 242 in order to reduce the computational complexity. A larger number did not give higher logscores.

Datasets

We compare approaches on commonly used benchmark datasets electricity (`elec`; Yu et al. 49), traffic forecasting (`traffic`; Yu et al. 49), monthly `tourism` [1], the hourly `m4` dataset [32] and currency `exchange` [28]. A short summary of these datasets can be found in Table D3 in the Supplementary Material.

Evaluation

For each proposed method and dataset, we report the logscores [13] and average results across time series and time points. The datasets are split into a training, validation, and test set by adhering to their time ordering. Evaluation windows are defined as done in the reference given for every dataset.

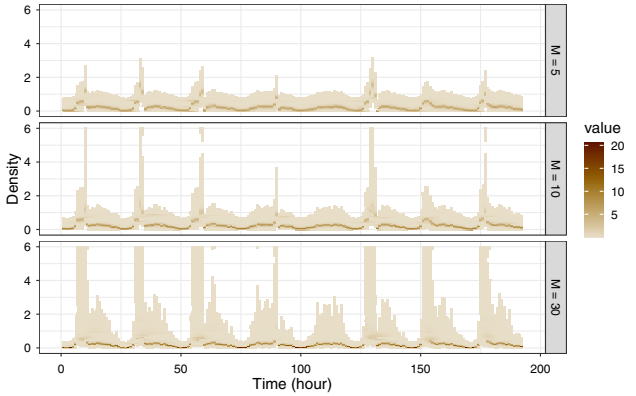

Fig. 5 Exemplary forecasted densities for one time series in the data set `traffic` for different values of M (rows) showcasing the increased expressiveness

Results

Table 2 shows the results of the comparison. Our approach always yields competitive and consistently good results while outperforming other models on most data sets.

Figure 5 further exemplarily depicts how the distribution is influenced when using an AT(p) model when varying the BSP order M. In the Appendix, we also provide an example when including the hour of the day into the exogenous shift term (not done for the give benchmark).

6 Conclusion and outlook

We have proposed ATMs, a flexible and comprehensible model class combining and extending various existing modeling approaches. ATMs allow for expressive probabilistic forecasts using a base distribution and a single transformation modeled by Bernstein polynomials. Additionally, a parametric inference paradigm based on MLE allows for statistical inference statements. ATMs empirically and theoretically recover well-known models, and demonstrate competitive performance on real-world datasets.

ATMs are the first adaption of transformation models to time series applications. Although our approach can be easily extended to incorporate deep neural network architectures, this invalidates statistical inference statements (e.g., because the uniqueness of $\boldsymbol{\theta}^*$ cannot be guaranteed). Future research will investigate this trade-off between larger model complexity and less statistical guarantees for the model.

An intriguing and practically relevant aspect of forecasting is to predict multiple steps into the future. Whereas classical (parametric) methods plug in an estimate (such as the mean) for the missing time series values when extrapolating more than one step into the future, ATMs are motivated by the observation that unimodal probabilistic forecasts do potentially not provide enough flexibility to model complex autoregressive processes. This raises the question of which value to plug in for missing lags in a multi-step-ahead forecast. Figure 1 makes clear that the (distribution's) mean is not necessarily a good choice. In addition, a sensible method would have to account for the additional uncertainty in every step, both to preserve the underlying motivation to model the aleatoric uncertainty in the data-generating process and to account for epistemic uncertainty in the parameter estimation. This constitutes a challenging task for a semi-parametric approach like ours and opens up various interesting questions for future research.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11222-023-10212-8.

Acknowledgements DR has been partially supported by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. TK gratefully acknowledges funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), grant KN 922/9-1. TH was supported by the Swiss National Science Foundation, grant number 200021_184603. We thank the two reviewers and the AE for their thoughtful comments and for pointing out ways to improve the manuscript. We further thank Lucas Kook and Thomas Nagler for a fruitful discussion on various topics.

Funding Open Access funding enabled and organized by Projekt DEAL.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Athanasopoulos, G., Hyndman, R.J., Song, H., et al.: The tourism forecasting competition. Int. J. Forecast. **27**(3), 822–844 (2011)
2. Baumann, P.F.M., Hothorn, T., Rügamer, D.: Deep Conditional Transformation Models. In: Machine Learning and Knowledge Discovery in Databases, pp. 3–18. Research Track. Springer International Publishing, Cham (2021)
3. Bengio, Y., Bengio, S.: Modeling high-dimensional discrete data with multi-layer neural networks. MIT Press, NIPS'99, p 400-406 (1999)
4. Bernstein, S.: Démonstration du théorème de weierstrass fondée sur le calcul des probabilités. Commun. Kharkov Math. Soc. **13**(1), 1–2 (1912)
5. Bishop, C.M.: Mixture density networks (1994)
6. Chernozhukov, V., Fernández-Val, I., Melly, B.: Inference on counterfactual distributions. Econometrica **81**(6), 2205–2268 (2013)
7. Cox, D.R.: Regression models and life-tables. J. Roy. Stat. Soc. Ser. B Methodol. **34**(2), 187–202 (1972)
8. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real nvp. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24 - 26, 2017, Conference Track Proceedings (2017)
9. Dunson, D.B.: Nonparametric Bayes applications to biostatistics. Bayesian Nonparametrics **28**, 223–273 (2010)
10. Farouki, R.T.: The Bernstein polynomial basis: a centennial retrospective. Comput. Aided Geom. Des. **29**(6), 379–419 (2012)
11. Foresi, S., Peracchi, F.: The conditional distribution of excess returns: an empirical analysis. J. Am. Stat. Assoc. **90**(430), 451–466 (1995)
12. Gneiting, T., Katzfuss, M.: Probabilistic forecasting. Annu. Rev. Stat. Appl. **1**, 125–151 (2014)
13. Gneiting, T., Balabdaoui, F., Raftery, A.E.: Probabilistic forecasts, calibration and sharpness. J. Royal Stat. Soc. Ser. B Stat. Methodol. **69**(2), 243–268 (2007)
14. Granger, C.W., Andersen, A.: On the invertibility of time series models. Stoch. Process. Appl. **8**(1), 87–92 (1978)
15. Hamilton JD (2010) Regime switching models. In: Macroeconometrics and time series analysis. Springer, p 202–209
16. Hothorn, T.: Transformation boosting machines. Stat. Comput. **30**(1), 141–152 (2020)
17. Hothorn, T., Kneib, T., Bühlmann, P.: Conditional transformation models. J. Royal Stat. Soc. Ser. B Stat. Methodol. **76**(1), 3–27 (2014)
18. Hothorn, T., Möst, L., Bühlmann, P.: Most likely transformations. Scand. J. Stat. **45**(1), 110–134 (2018)
19. Hyndman, R., Athanasopoulos, G., Bergmeir, C., et al.: Forecast: forecasting functions for time series and linear models. R. Package Vers. **8**, 15 (2021)
20. Jordan, A., et al.: On discriminative vs. generative classifiers: a comparison of logistic regression and naive Bayes. Adv. Neural. Inf. Process. Syst. **14**(2002), 841 (2002)
21. Kastner, G., Frühwirth-Schnatter, S., Lopes, H.F.: Efficient bayesian inference for multivariate factor stochastic volatility models. J. Comput. Graph. Stat. **26**(4), 905–917 (2017)
22. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
23. Kingma, D.P., Salimans, T., Jozefowicz, R., et al.: Improved variational inference with inverse autoregressive flow. In: Lee, D.,

Sugiyama, M., Luxburg, U., et al. (eds.) Advances in Neural Information Processing Systems, vol. 29. Curran Associates Inc (2016)
24. Klein, N., Hothorn, T., Barbanti, L., et al.: Multivariate conditional transformation models. Scand. J. Stat. **49**(1), 116–142 (2022)
25. Koenker, R.: Quantile Regression, vol. Economic. Society monographs, Cambridge University Press (2005)
26. Kook, L., Herzog, L., Hothorn, T., et al.: Deep and interpretable regression models for ordinal outcomes. Pattern Recognit. **122**, 108263 (2021)
27. Kook, L., Götschi, A., Baumann, P.F., et al.: Deep interpretable ensembles. arxiv:2205.12729 (2022)
28. Lai, G., Chang, W.C., Yang, Y., et al.: Modeling long-and short-term temporal patterns with deep neural networks. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pp 95–104 (2018)
29. Lin, T., Horne, B.G., Tino, P., et al.: Learning long-term dependencies in narx recurrent neural networks. IEEE Trans. Neural Netw. **7**(6), 1329–1338 (1996)
30. Ling, S., McAleer, M.: A general asymptotic theory for time-series models. Stat. Neerl. **64**(1), 97–111 (2010)
31. Liu, J., Paisley, J., Kioumourtzoglou, M.A., et al.: Accurate uncertainty estimation and decomposition in ensemble learning. In: Wallach, H., Larochelle, H., Beygelzimer, A., et al. (eds.) Advances in Neural Information Processing Systems, vol. 32. Curran Associates Inc (2019)
32. Makridakis, S., Spiliotis, E., Assimakopoulos, V.: The m4 competition: results, findings, conclusion and way forward. Int. J. Forecast. **34**(4), 802–808 (2018)
33. Murphy, K.P.: Machine learning: a probabilistic perspective. MIT press (2012)
34. Nagler, T., Krüger, D., Min, A.: Stationary vine copula models for multivariate time series. J. Econ. **227**(2), 305–324 (2022)
35. Papamakarios, G., Pavlakou, T., Murray, I.: Masked autoregressive flow for density estimation. In: Guyon I, Luxburg UV, Bengio S, et al (eds) Adv. Neural Inform. Process. Syst. (2017)
36. Papamakarios, G., Nalisnick, E., Rezende, D.J., et al.: Normalizing flows for probabilistic modeling and inference. J. Mach. Learn. Res. **22**(57), 1–64 (2021)
37. Raftery, A.E., Gneiting, T., Balabdaoui, F., et al.: Using bayesian model averaging to calibrate forecast ensembles. Mon. Weather Rev. **133**(5), 1155–1174 (2005)
38. Rao, T.S.: On the theory of bilinear time series models. J. Roy. Stat. Soc. Ser. B Methodol. **43**(2), 244–255 (1981)
39. Rügamer, D., Kolb, C., Klein, N.: Semi-Structured Deep Distributional Regression: A Combination of Additive Models and Deep Learning. arXiv preprint arXiv:2002.05777 (2020)
40. Sakia, R.M.: The box-cox transformation technique: a review. J. Royal Stat. Soc. Ser. D Stat. **41**(2), 169–178 (1992)
41. Schlosser, L., Hothorn, T., Stauffer, R., et al.: Distributional regression forests for probabilistic precipitation forecasting in complex terrain. Ann. Appl. Stat. **3**, 1564–89 (2019)
42. Shumway, R.H., Stoffer, D.S., Stoffer, D.S.: Time series analysis and its applications, vol. 3. Springer (2000)
43. Sick, B., Hothorn, T., Dürr, O.: Deep transformation models: Tackling complex regression problems with neural network based transformation models. In: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, pp. 2476–2481 (2021)
44. Smith, M.S.: Implicit copulas: An overview. Econometrics and Statistics (2021)
45. Uria, B., Côté, M.A., Gregor, K., et al.: Neural autoregressive distribution estimation. J. Mach. Learn. Res. **17**(205), 1–37 (2016)
46. Van Belle, V., Pelckmans, K., Suykens, J.A., et al.: Learning transformation models for ranking and survival analysis. J. Mach. Learn. Res. 12(3) (2011)
47. Wong, C.S., Li, W.K.: On a mixture autoregressive model. J. Royal Stat. Soc. Ser. B Stat. Methodol. **62**(1), 95–115 (2000)
48. Wu, C.O., Tian, X.: Nonparametric estimation of conditional distributions and rank-tracking probabilities with time-varying transformation models in longitudinal studies. J. Am. Stat. Assoc. **108**(503), 971–982 (2013)
49. Yu, H.F., Rao, N., Dhillon, I.S.: Temporal regularized matrix factorization for high-dimensional time series prediction. In: NIPS, pp. 847–855 (2016)