Supplementary Material: Probabilistic Time Series Forecasts with Autoregressive Transformation Models

David Rügamer[1,2*], Philipp F.M. Baumann[3], Thomas Kneib[4] and Torsten Hothorn[5]

[1*]Department of Statistics, LMU Munich, Munich, Germany.
[2]Institute of Statistics, RWTH Aachen, Aachen, Germany.
[3]KOF Swiss Economic Institute, ETH Zurich, Zurich, Switzerland.
[4]Chair of Statistics, University of Goettingen, Goettingen, Germany.
[5]Epidemiology, Biostatistics and Prevention Institute, University of Zurich, Zurich, Switzerland.

*Corresponding author(s). E-mail(s): david@stat.uni-muenchen.de;
Contributing authors: baumann@kof.ethz.ch; tkneib@uni-goettingen.de;
torsten.hothorn@uzh.ch;

Appendix A Further Details

A.1 Definitions

The following definition of the error distribution follows Hothorn et al (2018).

Definition 1 Error Distributions Let $Z : \Omega \to \mathbb{R}$ be a $\mathfrak{U} - \mathfrak{B}$ measurable function from $(\Omega, \mathfrak{U})$ to the Euclidian space with Borel σ-algebra $\mathfrak{B}$ with absolutely continuous distribution $\mathbb{P}_Z = f_Z \odot \mu_L$ on the probability space $(\mathbb{R}, \mathfrak{B}, \mathbb{P}_Z)$ and μ_L the Lebesque measure. We define F_Z and F_Z^{-1} as the corresponding distributions and assume $F_Z(-\infty) = 0$, $F_Z(\infty) = 1$. $0 < f_Z(z) < \infty \; \forall z \in \mathbb{R}$ with log-concave, twice-differentiable density f_Z with bounded first and second derivatives.

A.2 Propositions

Proposition 1 (**Interpretation of (11)**) *The ATM as defined in (8) and further specified in (11) can be seen as an additive regression model with outcome $h_{1t}(y_t)$, predictor $h_{2t}((h_{1t} \odot \mathcal{Y}_t \mid \mathcal{F}_{t-1}, \boldsymbol{x}) \mid \boldsymbol{x})$ and error term $\varepsilon \sim F_Z$.*

Proof We first define an additive regression model with outcome $\lambda_1 := h_{1t}(y_t)$, predictor $\tilde{\lambda}_2 := -h_{2t}((h_{1t} \odot \mathcal{Y}_t \mid \mathcal{F}_{t-1}, \boldsymbol{x}) \mid \boldsymbol{x})$ and error term $\varepsilon \sim F_Z$, i.e.,

$$\lambda_1 = \tilde{\lambda}_2 + \varepsilon, \; \varepsilon \sim F_Z,$$

where we use $\tilde{\lambda}_2 = -\lambda_2$ instead of λ_2 for convenience without loss of generality. This implies that $\lambda_1 - \tilde{\lambda}_2 = \lambda_1 + \lambda_2 = \varepsilon$ or equally $\lambda_1 + \lambda_2 \sim F_Z$. Optimizing this model is equal to fitting an ATM as defined in (8) with structural assumption as defined in (11).

Proposition 2 (**Equivalence of AR(p) and AT(p) models**) *An autoregressive model of order p (AR(p)) with independent white noise following the distribution F_Z in the location-scale family is equivalent to an AT(p) model for $M = 1$, $\boldsymbol{\vartheta}(\boldsymbol{x}) \equiv \boldsymbol{\vartheta}$, $r(\boldsymbol{x}) \equiv 0$ and error distribution F_Z.*

Proof The transformation function of an AT(p) model with BSPs of order M defined on an interval $[\iota_l, \iota_u]$, $\boldsymbol{\vartheta}(\boldsymbol{x}) \equiv \boldsymbol{\vartheta}$ and $r(\boldsymbol{x}) \equiv 0$ is given by

$$h_{1t} + h_{2t} = \boldsymbol{a}(y_t)^\top \boldsymbol{\vartheta} + \sum_{j=1}^{p} \phi_j \boldsymbol{a}(y_{t-j})^\top \boldsymbol{\vartheta}.$$

We can further simplify the model by making $\boldsymbol{a}(y_t)$ more explicit:

$$\boldsymbol{a}(y_t) = (M+1)^{-1} \begin{pmatrix} f_{BE(1,M+1)}(\tilde{y}_t) \\ \vdots \\ f_{BE(m,M-m+1)}(\tilde{y}_t) \\ \vdots \\ f_{BE(M+1,1)}(\tilde{y}_t) \end{pmatrix} \in \mathbb{R}^{M+1}$$

with $\tilde{y}_t = (y - \iota_l)/(\iota_u - \iota_l)$ and Beta distribution density $f_{BE(\kappa,\mu)}$ with parameters κ, μ. For simplicity and w.l.o.g. assume that $y_t \equiv \tilde{y}_t$. Setting M to 1, we get

$$\begin{aligned} h_{1t} &= (\vartheta_0 f_{BE(1,2)} + \vartheta_1 f_{BE(2,1)})/2 \\ &= \vartheta_0(1 - y_t) + \vartheta_1 y_t \\ &= \vartheta_0 + (\vartheta_1 - \vartheta_0) y_t \\ &= \vartheta_0 + \tilde{\vartheta}_1 y_t. \end{aligned}$$

The transformation of the AT(p) model is thus given by

$$\begin{aligned} h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x}) &= \vartheta_0 + \tilde{\vartheta}_1 y_t + \sum_{j=1}^{p} \phi_j(\vartheta_0 + \vartheta_1 y_{t-j}) \\ &= \frac{y_t + \tilde{\vartheta}_0 + \sum_{j=1}^{p} \tilde{\phi}_j y_{t-j}}{\tilde{\vartheta}_1^{-1}} \end{aligned} \tag{A1}$$

with $\tilde{\vartheta}_0 = (\vartheta_0(1 + \sum_j \phi_j))/\tilde{\vartheta}_1$ and $\tilde{\phi}_j = \phi_j \vartheta_1/\tilde{\vartheta}_1$. From (8) we know

$$\mathbb{P}(Y_t \leq y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x}) = F_Z(h_t(y_t \mid \mathcal{F}_{t-1}, \boldsymbol{x})). \tag{A2}$$

The AR(p) model with coefficients $\varphi_0, \ldots, \varphi_p$ is given by

$$\begin{aligned} & y_t = \varphi_0 + \sum_{j=1}^{p} \varphi_j y_{t-j} + \sigma\varepsilon_t,\ \varepsilon_t \sim F_Z \\ \Leftrightarrow \quad & Z = \frac{y_t - \varphi_0 - \sum_{j=1}^{p} \varphi_j y_{t-j}}{\sigma} \sim F_Z. \end{aligned} \tag{A3}$$

The equivalence of (A2) in combination (A1) with (A3) is then given when setting $\tilde{\vartheta}_0 = -\varphi_0$, $\tilde{\phi}_j = -\varphi_j \forall j \in \{1, \ldots, p\}$ and $\sigma = \tilde{\vartheta}_1^{-1}$. Since both models find their parameters using Maximum Likelihood and it holds $\tilde{\vartheta}_1 > 0$ (as required for σ) by the monotonicity restriction on the BSPs coefficient, the models are identical up to different parameterization.

A.3 Proof of Theorems

The provided theorems 1-3 can be proven by observing that AT(p)s' model structure and all made assumptions follow the general asymptotic theory for time series models as given in Ling and McAleer (2010). It is left to show that our setup and assumptions are equivalent to this general theory.

Proof. Our setup described in Section 4 together with Assumption 1(i) corresponds to the setup described in Ling and McAleer (2010), Section 2. Our Assumption 1(ii-iv) corresponds to their Assumption 2.1. In contrast, we do not consider the case of infinite $\mathcal{Y}_0$, but the extension is straightforward, by replacing initial values by some constant. Since AT(p)s and nonlinear extensions are fully-parameterized time series models (Equation 11) with parameter estimator $\hat{\boldsymbol{\theta}}_T$ found by MLE, all necessary assumptions are met to apply Theorem 2.1 in Ling and McAleer (2010) including the subsequent remark, which yields the proof of our theorems 1-3. □

Appendix B Interpretability Example

Next to the theoretical properties of ATMs described in Section 3.2, we will give an illustrative example in this section to make the different interpretability aspects of ATMs more tangible.

Example 1 Assume that the true generating process is additive on a log-scale and influenced by the two previous time points $t-1$ and $t-2$. For example, t can be thought of as days in a year and the process Y_t is an interest rate. Assume that the interest rate is multiplicatively influenced by the year $x_t \in E$ and further differs in its mean depending on a cyclic effect of the month η_t. An example for a corresponding data generating process would be

$$\begin{aligned} \log(y_t) =& 0.5 \log(y_{t-1}) \left(\sum_{e \in E} \theta_e I(x_t = e) \right) + \\ & 0.2 \log(y_{t-2}) \left(\sum_{e \in E} \theta_e I(x_t = e) \right) + \\ & \sin(\eta_t) + \varepsilon_t, \quad \varepsilon_t \sim F_Z. \end{aligned}$$

In this case, the transformation function h_{1t} can be defined as $h_{1t}(y_t) = \log(y_t)(\sum_{e \in E} \theta_e I(x_t = e))$ and approximated by $\boldsymbol{a}(y_t)^\top \boldsymbol{\vartheta}(x_t)$, where $\boldsymbol{a}$ is the BSP evaluation of y_t and $\boldsymbol{\vartheta}$ a vector of coefficients depending on the year x_t. Further $\phi_1 = 0.5, \phi_2 = 0.2$, and the exogenous shift $r = \sin(\eta_t)$, which in practice would be approximated using a basis function representation. The interpretability properties listed in Section 3.2 can be explained as follows:

1. The additivity assumption in $\boldsymbol{\vartheta}$ allows to interpret the individual effects of the year x_t on the transformation function h_1 individually (ceteris paribus) as $\log(y_t)(\sum_{e \in E} \theta_e I(x_t = e)) =$

$\sum_{e \in E} \log(y_t)\theta_e I(x_t = e)$. Here, this would allow statements how a certain year e influences the interest rate's density.

2. The use of the BSP basis for $\boldsymbol{a}$ in combination with 1. allows to visualize a forecasted density analytically for every additive term in $\boldsymbol{\vartheta}$. For example, to interpret year e, we evaluate $\boldsymbol{\vartheta}(x_t = e)$ and visualize $h_{1t}(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}(x_t = e)$ as a function of y on a given domain of interest.
3. The structural assumptions of ATMs, i.e., their separation into two transformation functions h_1 and h_2, allows to interpret both transformation functions h_1, h_2 individually (ceteris paribus). In this example, the effect of the year can be interpreted using 1. and 2. while keeping the month fixed, and vice versa, the effect of the month can be interpreted by fixing the year. The applied transformation h_1 for AT(p) models further allows to to individually interpret the influence of different lags (here these are the multiplicative effects $phi_1 = 0.5$ and $\phi_2 = 0.2$).

Appendix C Parametric Bootstrap

To assess the parameter uncertainty included in the estimated density, we propose to use a parametric Bootstrap (similar to the one suggested in Hothorn et al, 2018) that is based on the following steps:

1. Generate $\hat{\boldsymbol{\theta}}^{(\nu)}, \nu = 1, \ldots, N$ from the limiting distribution (Theorem 2 and 3);
2. Draw samples $Z_{\tilde{t}} \sim F_Z, \tilde{t} \in \mathcal{T}$ and calculate $Y_{\tilde{t},\nu} = \inf\{y \in \Xi \mid h_{\tilde{t}}(y, \hat{\boldsymbol{\theta}}^{(\nu)}) \geq Z_{\tilde{t}}\}$;
3. Refit the model for each data set $\{Y_{\tilde{t},\nu}\}_{\tilde{t} \in \mathcal{T}}, \nu = 1, \ldots, N$;
4. Calculate the N model densities.

Based on these N model densities, uncertainty in the originally estimated density can be analyzed, e.g., visually by plotting all densities together as done in Figure 1 and 3.

Appendix D Experimental Setup

All codes are available at https://github.com/davidruegamer/ATMs_experiments.

Table D1 Average MSE in percent (with standard deviation in brackets) of estimated coefficients by the AR(p) and AT(p) model (rows) for different simulation settings (columns) over 100 replications.

		$p = 1$	$p = 2$	$p = 5$
$T = 200$	AR(p)	0.54 (0.73)	0.49 (0.49)	0.55 (0.4)
	AT(p)	0.73 (1)	0.68 (0.6)	0.69 (0.42)
$T = 1000$	AR(p)	0.12 (0.16)	0.12 (0.13)	0.12 (0.09)
	AT(p)	0.17 (0.25)	0.15 (0.16)	0.17 (0.11)
$T = 5000$	AR(p)	0.019 (0.03)	0.02 (0.02)	0.02 (0.02)
	AT(p)	0.06 (0.09)	0.05 (0.05)	0.05 (0.03)

D.1 Simulations

In this subsection, we describe the details of the data generating process used in Figure 1 and provide results on experiments for the *equivalence and consistency* paragraph of Section 5.1 in Section D.1.2.

D.1.1 Data Generating Process Toy Example

For Figure 1, we simulate $T = 1000$ time points $y_1, \ldots, y_T$ that exhibit two modes as follows:

1. Set $y_0 = 0$;
2. Define a shift $\varrho = 2$ and sample $x_1, \ldots, x_T$ from $\{-\varrho, \varrho\}$ with equal probability;
3. Define a autoregressive coefficient $\phi_1 = 0.1$
4. For $t = 1, \ldots, T$, sample $y_t \sim \mathcal{N}(\phi_1 y_{t-1} + x_t, 1)$

When providing the model with the marginal distribution of y_t and defining x_t as latent, unobserved variable, y_t will exhibit two modes centered around $\pm\varrho$.

D.1.2 AR(p) comparison

The data generating process for the simulation of Section 5.1 is an AR model with the p first coefficients 0.4, 0.2, 0.1, 0.05, 0.025. A standard implementation for the AR model was used. For the AT model we use the implementation provided in Rügamer et al (2022) using 2500 epochs, batch size of 50, and early stopping based on 10% of the training data.

Table D2 Mean and standard deviation (brackets) of the mean squared error ($\times 10^2$ 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).

		$p = 1$	$p = 2$	$p = 4$
$T = 200$	Oracle	0.65 (0.84)	0.45 (0.46)	0.46 (0.32)
	AT(p)	0.49 (0.62)	0.57 (0.76)	0.65 (0.45)
$T = 400$	Oracle	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)
$T = 800$	Oracle	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)

D.1.3 Influence of M on the distribution's shape

Based on a Gamma-distributed time series with first-order lag influence, we run 20 simulation repetitions while M to investigate influence of the order of BSP on the the distribution's shape. Figure D1 depicts the true (red) and estimated densities (black) for different M.

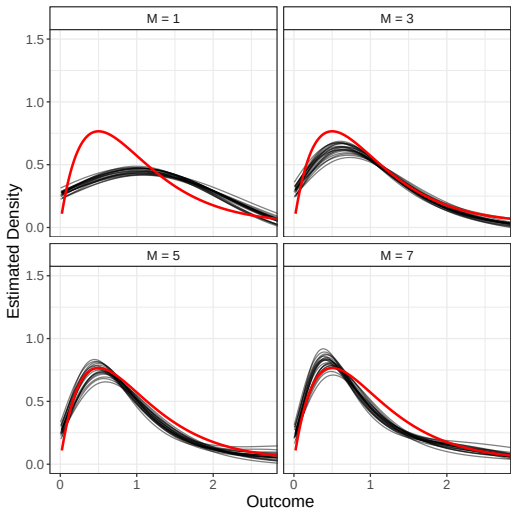

Figure D1 Demonstration of the influence of M (different facets) on the shape of the fitted distribution (black lines; each line corresponds to one simulation repetition) when fitting an AT(p) model to a Gamma-distributed time series (true density in red).

D.2 Details on the benchmark study

D.2.1 Datasets

Table D3 summarizes the characteristics of the data sets used. For `elec` and `traffic` we use the 24 hours forecasting horizon and a pre-defined subset of one week of data. For `m4` and `tour` the test sets are already pre-defined with 48 hours and 24 months forecast windows, respectively.

Electricity

The dataset is available at https://archive.ics.uci.edu/ml/datasets/ElectricityLoadDiagrams20112014. According to Chen et al (2020), Appendix A.3, the dataset describes the series of the electricity consumption (kWh) of 370 customers. The electricity usage values are recorded per 15 minutes from 2011 to 2014. We select the data of the last three years. By aggregating the records of the same hour, we use the hourly consumption data of size $370 \cdot 26304$, where 26304 is the length of the time series (Yu et al, 2016). The data used for modelling ranges from '2014-06-07 23:00:00' to '2014-06-09 23:00:00' including 1 day of validation and test data.

Exchange

The dataset is available from Lai et al (2018) and contains 8 bilateral exchange rate series for business days between Jan 1991 and May 2013. The split between training (60%), validation (20%) and test (20%) is done based on the chronological order.

Traffic

The traffic dataset is available at https://archive.ics.uci.edu/ml/datasets/PEMS-SF. It describes the occupancy rates (between 0 and 1) of 963 car lanes of San Francisco bay area freeways. The measurements are carried out over the period from 2008-01-01 to 2009-03-30 and are sampled every 10 minutes. The original dataset is split into training and test. Hourly aggregation is applied to obtain hourly traffic data (Yu et al, 2016). The final time series are of length 10560 (the occupancy rates). The data used for modelling ranges from '2008-05-01 00:00:00' to '2008-05-09 23:00:00' including 1 day of validation and test data.

Tourism

The dataset is available at https://robjhyndman.com/publications/the-tourism-forecasting-competition/. Data is available on a monthly, quarterly and yearly level. We used the 366 monthly series which measure tourism demand. The data is split into test and train. 67 month are the minimum that is available for training and forecasting horizon is defined to be 24 months. The starting date for each monthly series is different. See Section 4 of Athanasopoulos et al (2011) for details.

m4

The dataset is taken from Makridakis et al (2018). It contains 414 time series which are summarized in the m4 hourly data set. The split between training and test is already provided. Details on further background can be found on Wikipedia: https://en.wikipedia.org/wiki/Makridakis_Competitions. The starting point of each series is different. The minimum training length is 700 hours. The forecasting horizon is 48 hours.

Software

For ATMs we extended the software `deepregression` (Rügamer et al, 2022) by including an additional additive component for lags and used optimization

Table D3 Characteristics of the benchmark datasets.

	`electricity`	`exchange`	`traffic`	`tourism`	`m4`
# time series	370	8	963	366	414
frequency	hourly	daily	hourly	monthly	hourly
forecast horizon	24/72	1219	24/72	24	48
# training samples	71040	39048	184896	10980	269514

techniques considered in Rügamer et al (2020); Baumann et al (2021). For ARIMA, we use the `forecast` R package (Hyndman et al, 2021).

D.2.2 Further Results

In the benchmark of the main paper, we only include an indicator variable for every measurement unit as covariate for the shift term. Figure D2 exemplary depicts how the shift term of the AT(p) model is influenced if we would include the hour of the day as another variable in the exogenous.

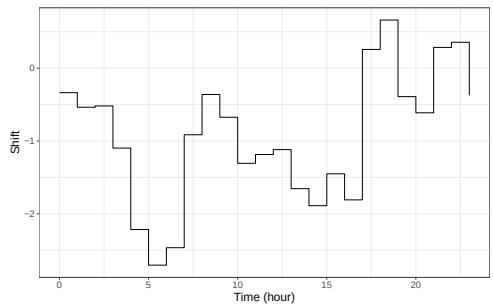

Figure D2 Estimated shift of the distribution for different times of the day (hour) when including the variable hour in $r(\boldsymbol{x})$ in h_{2t} for the `traffic` dataset.

D.2.3 Computational Setup

All models were run on a server with 90GB RAM, 20 vCPUs from type Intel Xeon Processor (Skylake, IBRS), and a server with 64GB RAM, 32 vCPUs from type Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz.

Appendix E Run-time Complexity

In addition to forecasting performance comparisons, we also conduct a run-time benchmark to compare the run-time complexity of ATMs with other approaches. We use two different implementations for ATMs and measure their run-time. We contrast these run-times with the ARIMA model as implemented in the `forecast` R package (Hyndman et al, 2021) and additionally include Prophet from the `prophet` R package (Taylor and Letham, 2021) as another fast alternative method for Bayesian forecasting.

The timing benchmark results (averaged over 10 replications) for different numbers of observations T are given in Table E4. Results suggest that - as expected

Table E4 Comparison of run-times for different methods (in columns) on different numbers of observations (#Obs.) T (in rows).

#Obs.	ATM (plain)	ARIMA	Prophet	ATM (neural)
10^2	0.199	0.005	0.372	22.20
10^3	0.513	0.024	0.097	31.30
10^4	3.920	0.118	0.342	28.80
10^5	94.62	1.121	33.99	32.30

– ATMs in a neural network are very slow compared to ARIMA, Prophet and also a plain ATM implementation in `R`. However, all methods show an exponential increase in time consumption while the time consumption of the neural network implementation of ATMs (ATM (neural)) with mini-batch training and early stopping does only slightly increase in runtime for an exponential increase in number of observations. Moreover, for 10^5 observations, ATM (plain) and Prophet already yield longer runtimes.

References

Athanasopoulos G, Hyndman RJ, Song H, et al (2011) The tourism forecasting competition. International Journal of Forecasting 27(3):822–844

Baumann PFM, Hothorn T, Rügamer D (2021) Deep Conditional Transformation Models. In: Machine Learning and Knowledge Discovery in Databases. Research Track. Springer International Publishing, Cham, pp 3–18

Chen J, Vaughan J, Nair VN, et al (2020) Adaptive Explainable Neural Networks (AxNNs). arXiv preprint arXiv:200402353 https://arxiv.org/abs/arXiv:2004.02353

Hothorn T, Möst L, Bühlmann P (2018) Most likely transformations. Scandinavian Journal of Statistics 45(1):110–134

Hyndman R, Athanasopoulos G, Bergmeir C, et al (2021) forecast: Forecasting functions for time series and linear models. R package version 8.15

Lai G, Chang WC, Yang Y, et al (2018) 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

Ling S, McAleer M (2010) A general asymptotic theory for time-series models. Statistica Neerlandica 64(1):97–111

Makridakis S, Spiliotis E, Assimakopoulos V (2018) The m4 competition: Results, findings, conclusion and way forward. International Journal of Forecasting 34(4):802–808

Rügamer D, Pfisterer F, Bischl B (2020) Neural mixture distributional regression. arXiv preprint arXiv:201006889 https://arxiv.org/abs/arXiv:2010.06889

Rügamer D, Kolb C, Fritz C, et al (2022) deepregression: a flexible neural network framework for semi-structured deep distributional regression. Journal of Statistical Software Accepted, https://arxiv.org/abs/arXiv:2104.02705

Taylor S, Letham B (2021) prophet: Automatic Forecasting Procedure. R package version 1.0

Yu HF, Rao N, Dhillon IS (2016) Temporal regularized matrix factorization for high-dimensional time series prediction. In: NIPS, pp 847–855