Probabilistic Forecasting of Electricity Prices

Nikolaus Umlauf, Rudolf Hakenschmidt

https://nikum.org/

Challenges in Forecasting Electricity Prices

Day-ahead electricity prices from Germany, 2015/01/01 - 2020/12/31.

Auction takes place at noon, all hours of the following day are traded in an uniform price auction.

- High volatility and seasonality.
- Influence of external factors.
- Complex dependency structures.
- Choice of probabilistic model.
- Variable selection.
- Evaluation and interpretation of forecasts.

Dataset (Preliminary)

P: Prices (EUR/MWh)

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-130.09	25.92	34.02	34.57	43.59	200.04

L: Load DA forecast

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
28824	47044	54776	54852	62982	75912

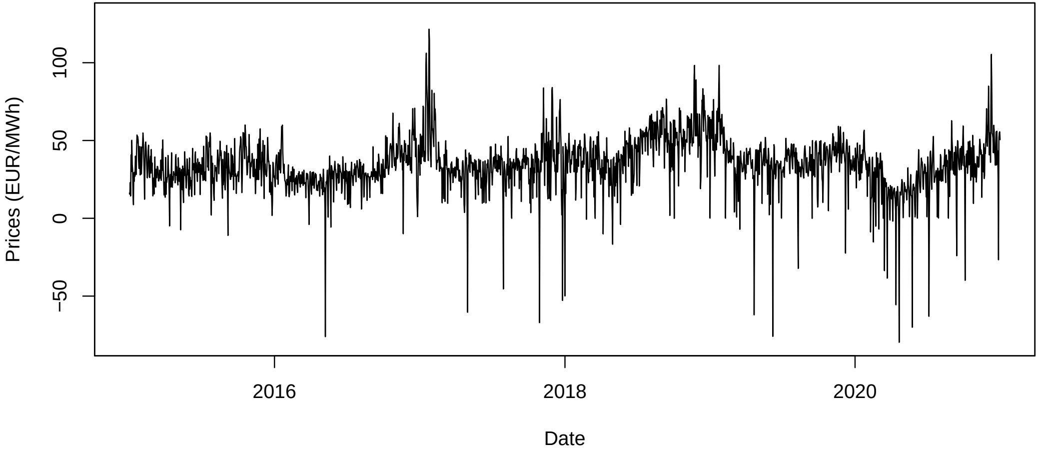
R: Renewables DA forecast:

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
574.2	7907.8	14437.8	16309.9	23086.1	62490.1

E: EUA; C: API2 coal; G: TTF gas; O: Brent oil.

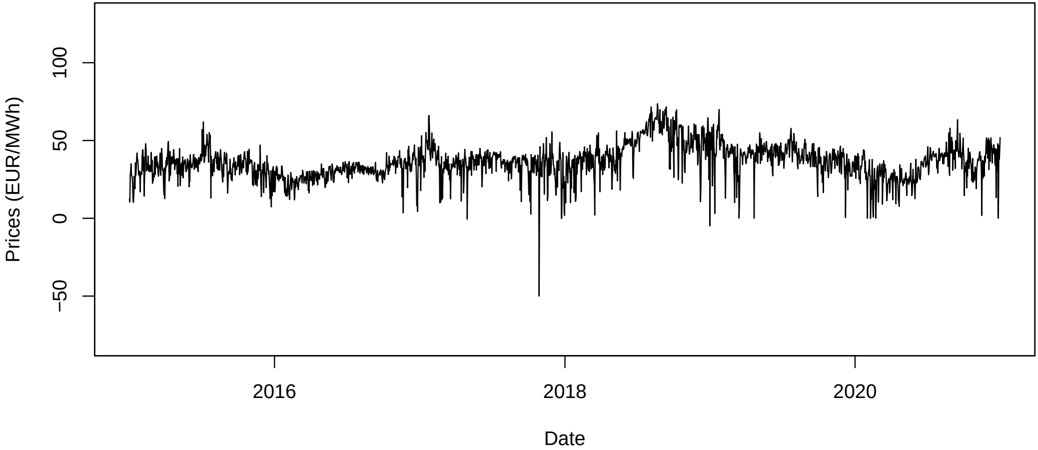
Hourly Prices

Hour 12, raw data:



Hourly Prices

Hour 22, raw data:



Regression

A typical regression model for electricity price forecasting

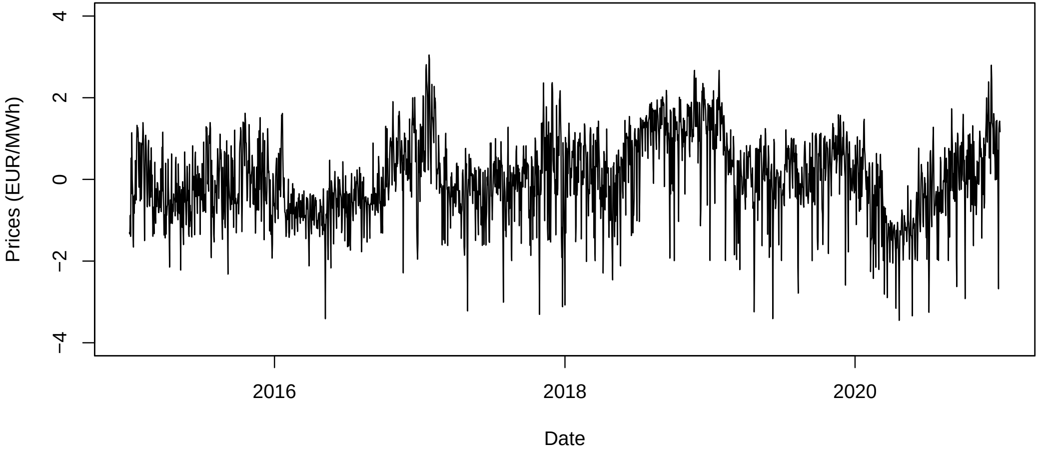
$$\mathtt{P}_{d,h} = \beta_0 + \beta_1 \mathtt{P}_{d-1,h} + \beta_2 \mathtt{L}_{d,h} + \beta_3, \mathtt{R}_{d,h} + \ldots + \varepsilon_{d,h}, \quad \varepsilon_{d,h} \sim N(0, \sigma^2)$$

where:

- d, h: The day and hour.
- $\mathtt{P}_{d,h}$: Electricity price (usually transformed).
- $\mathtt{P}_{d-1,h}$: Electricity price of previous day (lagged price).
- $\mathtt{L}_{d,h}$, $\mathtt{R}_{d,h}$: Load and renewable forecast.
- $\ldots$: Seasonal factors (e.g., time of day, day of week), other lagged variables $\mathtt{E}$, $\mathtt{C}$, $\mathtt{G}$, $\mathtt{O}$.
- $\beta_0, \beta_1, \beta_2, \beta_3, \ldots$: Coefficients to be estimated.

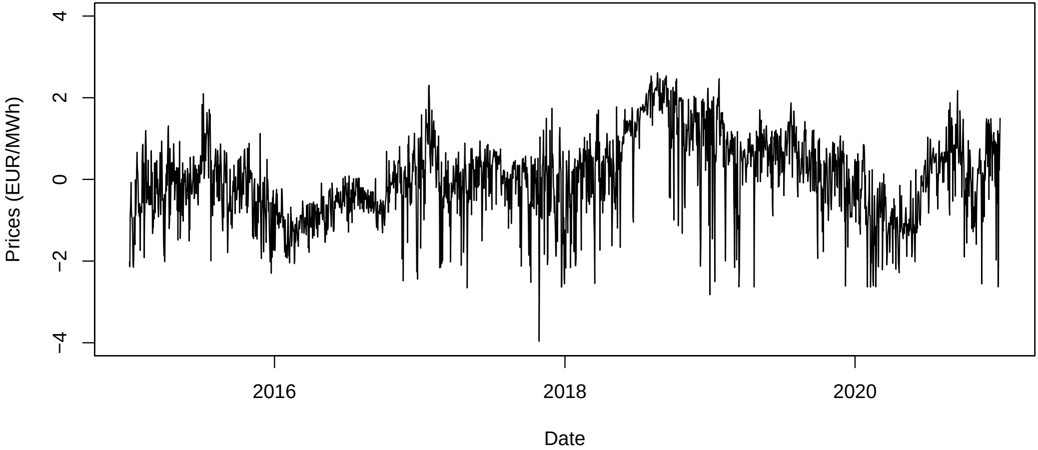
Hourly Prices

Hour 12, transformed data:



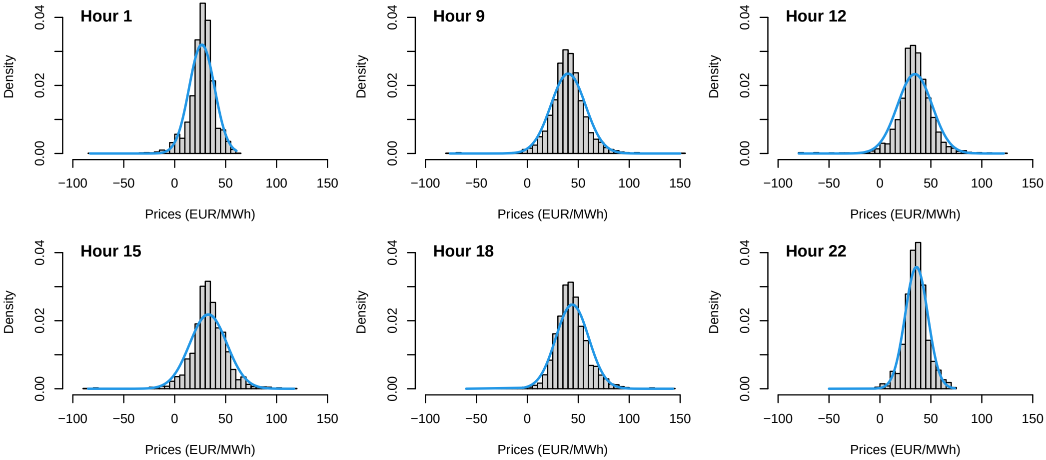
Hourly Prices

Hour 22, transformed data:



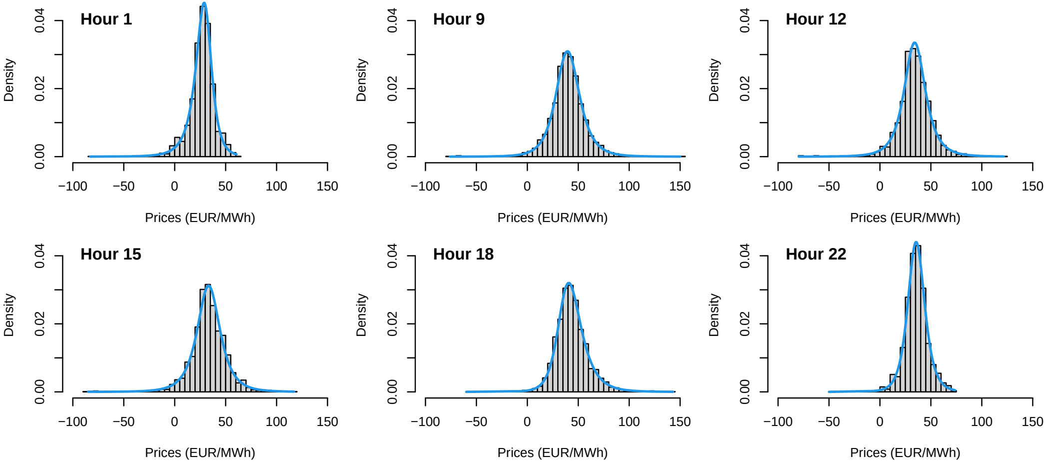
Hourly Distributional Models

Normal distribution:



Hourly Distributional Models

Johnson's SU- distribution:

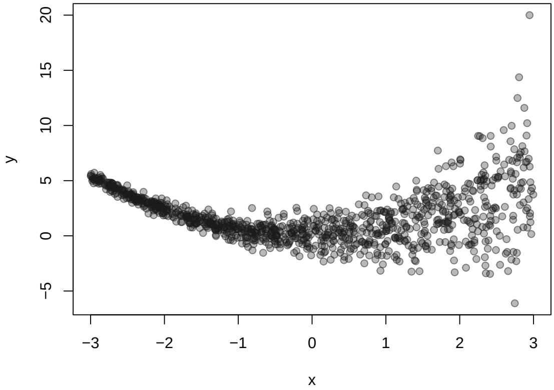


Regression Beyond the Mean

- Classical regression has focused on relating the conditional mean of a response y_i to covariate information x_i for observations $(x_1, y_1), \ldots, (x_n, y_n)$.
- Linear model:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$$

with ε_i i.i.d. $N(0, \sigma^2)$



Regression Beyond the Mean

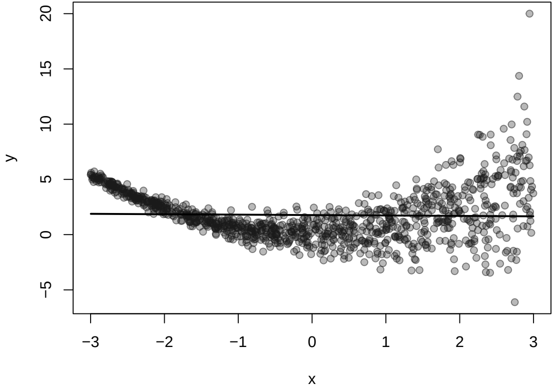
- Classical regression has focused on relating the conditional mean of a response y_i to covariate information x_i for observations $(x_1, y_1), \ldots, (x_n, y_n)$.

- Linear model:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$$

with ε_i i.i.d. $N(0, \sigma^2)$

$$\Rightarrow \ \mathrm{E}(y_i|x_i) = \mu_i(x_i) = \beta_0 + \beta_1 x_i$$



Regression Beyond the Mean

- Classical regression has focused on relating the conditional mean of a response y_i to covariate information x_i for observations $(x_1, y_1), \ldots, (x_n, y_n)$.

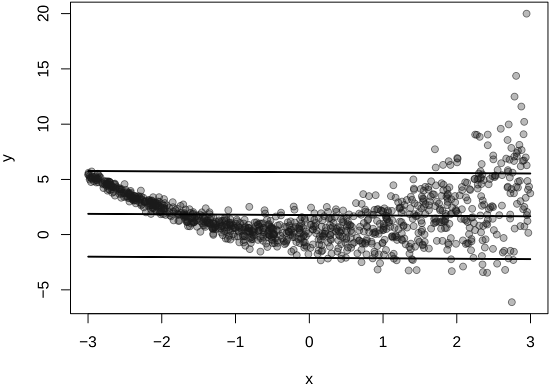
- Linear model:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$$

with ε_i i.i.d. $N(0, \sigma^2)$

$\Rightarrow \; \mathrm{E}(y_i | x_i) = \mu_i(x_i) = \beta_0 + \beta_1 x_i$

$\Rightarrow \; \mathrm{Var}(y_i | x_i) = \sigma^2$



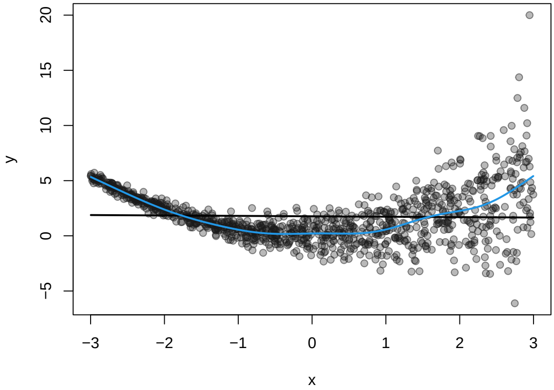
Regression Beyond the Mean

- Classical regression has focused on relating the conditional mean of a response y_i to covariate information x_i for observations $(x_1, y_1), \dots, (x_n, y_n)$.

- Nonparametric model

$$E(y_i|x_i) = \mu_i(x_i) = \beta_0 + f(x_i)$$

with $y_i|x_i \sim N\left(\mu_i(x_i), \sigma^2\right)$



Regression Beyond the Mean

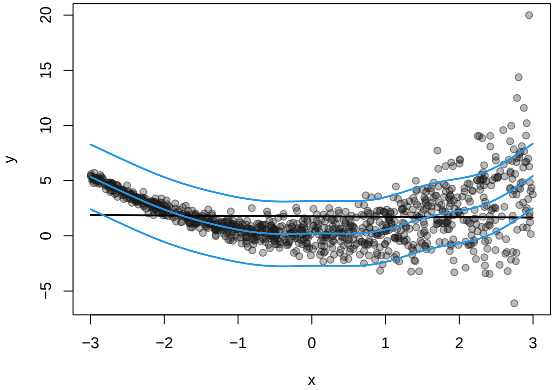
- Classical regression has focused on relating the conditional mean of a response y_i to covariate information x_i for observations $(x_1, y_1), \ldots, (x_n, y_n)$.

- Nonparametric model

$$E(y_i|x_i) = \mu_i(x_i) = \beta_0 + f(x_i)$$

with $y_i|x_i \sim N\left(\mu_i(x_i), \sigma^2\right)$

σ^2 fixed.



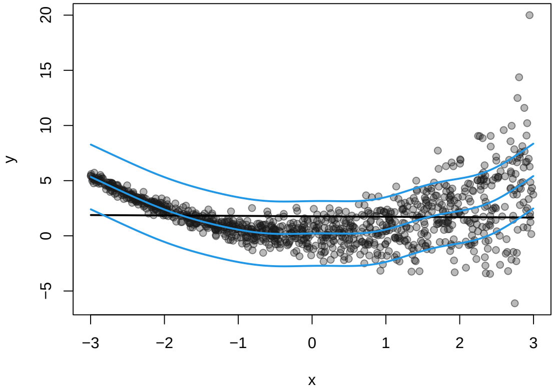
Regression Beyond the Mean

- Nonparametric model for loaction and scale.

- $E(y_i|x_i) = \mu_i(x_i) = \beta_0^{\mu} + f^{\mu}(x_i)$

  $$\text{Var}(y_i|x_i) = \sigma_i^2(x_i) = \exp\left(\beta_0^{\sigma^2} + f^{\sigma^2}(x_i)\right)$$

  with $y_i|x_i \sim N\left(\mu_i(x_i), \sigma_i^2(x_i)\right)$.



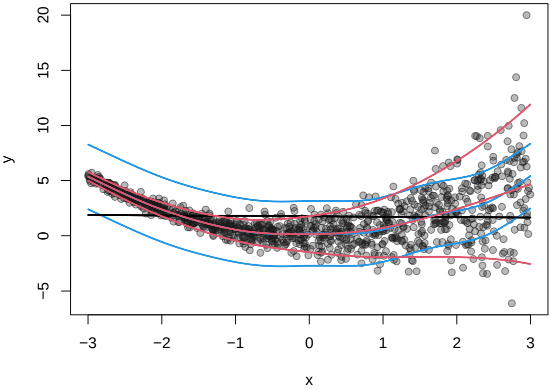
Regression Beyond the Mean

- Nonparametric model for loaction and scale.

- $\mathrm{E}(y_i|x_i) = \mu_i(x_i) = \beta_0^{\mu} + f^{\mu}(x_i)$

$$\mathrm{Var}(y_i|x_i) = \sigma_i^2(x_i)$$
$$= \exp\left(\beta_0^{\sigma^2} + f^{\sigma^2}(x_i)\right)$$

with $y_i|x_i \sim N\left(\mu_i(x_i), \sigma_i^2(x_i)\right)$.



Model Specification

Any parameter of a population distribution $\mathcal{D}$ may be modeled by explanatory variables

$$y \sim \mathcal{D}\left(\theta_1(\mathbf{x}; \boldsymbol{\beta}_1), \ \ldots, \ \theta_K(\mathbf{x}; \boldsymbol{\beta}_K)\right),$$

with $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^\top, \ldots, \boldsymbol{\beta}_K^\top)^\top$.

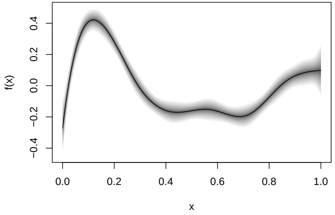
Each parameter is linked to a structured additive predictor

$$h_k(\theta_k(\mathbf{x}; \boldsymbol{\beta}_k)) = f_{1k}(\mathbf{x}; \boldsymbol{\beta}_{1k}) + \ldots + f_{J_k k}(\mathbf{x}; \boldsymbol{\beta}_{J_k k}),$$

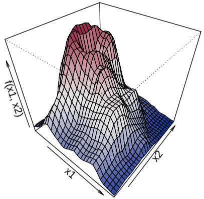
- $j = 1, \ldots, J_k$ and $k = 1, \ldots, K$.
- $h_k(\cdot)$: Link functions for each distribution parameter.
- $f_{jk}(\cdot)$: Model terms of one or more variables.

Model Terms $f_{jk}(\cdot)$

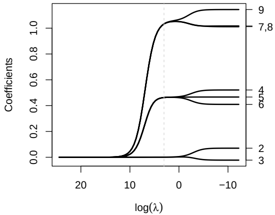
Nonlinear Effects



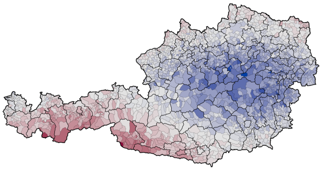
Two−Dimensional Surfaces



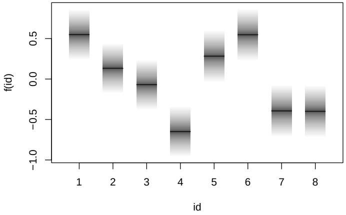
LASSO & Factor Clustering



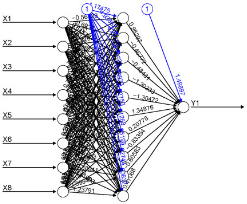
Spatially Correlated Effects f(x) = f(s)



Random Intercepts f(x) = f(id)



Neural Networks



Estimation

The main building block of regression model algorithms is the probability density function $d_y(\mathbf{y}|\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K)$.

Estimation typically requires to evaluate the log-likelihood

$$\ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X}) = \sum_{i=1}^{n} \log d_y(y_i; \theta_1(\mathbf{x}_i; \boldsymbol{\beta}_1), \dots, \theta_K(\mathbf{x}_i; \boldsymbol{\beta}_K)),$$

with $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_K)$.

The log-posterior (frequentist penalized log-likelihood)

$$\log \pi(\boldsymbol{\beta}, \boldsymbol{\tau}; \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha}) \propto \ell(\boldsymbol{\beta}; \mathbf{y}, \mathbf{X}) + \sum_{k=1}^{K} \sum_{j=1}^{J_k} \left[\log p_{jk}(\boldsymbol{\beta}_{jk}; \boldsymbol{\tau}_{jk}, \boldsymbol{\alpha}_{jk})\right],$$

where $p_{jk}(\cdot)$ are priors, $\boldsymbol{\tau}_{jk}$ (smoothing) variances and $\boldsymbol{\alpha}_{jk}$ fixed hyper parameters.

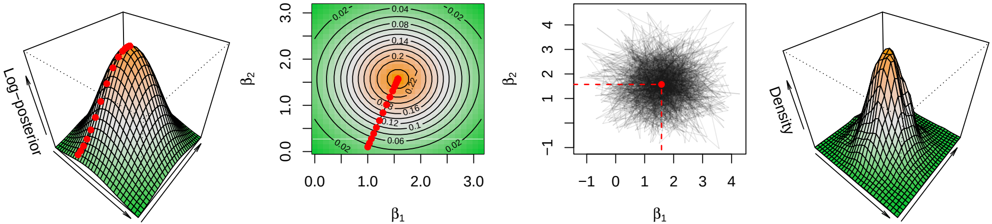
Estimation

Bayesian point estimates of parameters are obtained by:

1. Maximization of the log-posterior for posterior mode estimation.
2. Solving high dimensional integrals, e.g., for posterior mean or median estimation.

Problems 1 and 2 are commonly solved by computer intensive iterative algorithms of the following type:

$$(\boldsymbol{\beta}^{[t+1]}, \boldsymbol{\tau}^{[t+1]}) = \mathbb{U}(\boldsymbol{\beta}^{[t]}, \boldsymbol{\tau}^{[t]}; \mathbf{y}, \mathbf{X}, \boldsymbol{\alpha}).$$



L1-Type Penalization

Idea: depending on the type of covariate effects, subtract a combination of (parts of) the following penalty terms $\tau^{-1}J(\boldsymbol{\beta})$ from the log-likelihood.

Classical LASSO (Tibshirani, 1996): For a metric covariate x_{jk} use

$$J_m(\beta_{jk}) = |\beta_{jk}| \,.$$

Group LASSO (Meier et al., 2008): For a (dummy-encoded) categorical covariate $\mathbf{x}_{jk}$ use

$$J_g(\boldsymbol{\beta}_{jk}) = ||\boldsymbol{\beta}_{jk}||_2 \,,$$

with vector $\boldsymbol{\beta}_{jk}$ collecting all corresponding coefficients.

L1-Type Penalization

Alternatively, for categorical covariates often *clustering* of categories with implicit *factor selection* is desirable.

Fused LASSO (Gertheiss and Tutz, 2010): Depending on the *nominal* (left) or *ordinal* scale level (right) of the covariate, use

$$J_f(\boldsymbol{\beta}_{jk}) = \sum_{l>m} w_{lm}^{(jk)} \left|\beta_{jkl} - \beta_{jkm}\right| \quad \text{or} \quad J_f(\boldsymbol{\beta}_{jk}) = \sum_{l=1}^{c_{jk}} w_{l}^{(jk)} \left|\beta_{jkl} - \beta_{jk,l-1}\right|$$

where c_{jk} is the number of levels of categorical predictor $\mathbf{x}_{jk}$ and $w_{lm}^{(jk)}, w_{l}^{(jk)}$ denote suitable weights. Choosing $l = 0$ as the reference, $\beta_{jk0} = 0$ is fixed.

New: Fused LASSO for metric covariates incl. splines for nonlinear effects.

L1-Type Penalization

Quadratic approximations of the penalties (compare Oelker & Tutz, 2017)

$$J_{jk}(\boldsymbol{\beta}_{jk}) \approx J_{jk}(\boldsymbol{\beta}_{jk}^{(t)}) + \frac{1}{2}\left(\boldsymbol{\beta}_{jk}^{\top}\mathbf{P}_{jk}(\boldsymbol{\beta}_{jk})\boldsymbol{\beta}_{jk} + (\boldsymbol{\beta}_{jk}^{(t)})^{\top}\mathbf{P}_{jk}(\boldsymbol{\beta}_{jk}^{(t)})\boldsymbol{\beta}_{jk}^{(t)}\right),$$

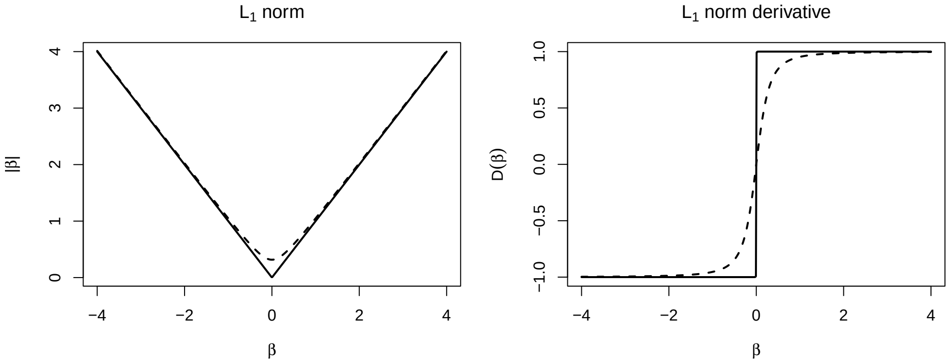
with

$$\mathbf{P}_{jk}(\boldsymbol{\beta}_{jk}^{(t)}) = q'_{jk}\left(\left\|\mathbf{a}_{jk}^{\top}\boldsymbol{\beta}_{jk}^{(t)}\right\|_{N_{jk}}\right) \cdot \frac{D_{jk}(\mathbf{a}_{jk}^{\top}\boldsymbol{\beta}_{jk}^{(t)})}{\mathbf{a}_{jk}^{\top}\boldsymbol{\beta}_{jk}^{(t)}} \cdot \mathbf{a}_{jk}\mathbf{a}_{jk}^{\top}.$$

E.g., $\|\beta\|_1 = |\beta|$ is approximated by $\sqrt{\beta^2 + c}$, hence, IWLS based updating functions $\mathtt{U}_{\mathtt{jk}}(\cdot)$ are relatively easy to implement.

L1-Type Penalization

Example of the approximation of the L_1 norm.



Usually setting the constant to $c \approx 10^{-5}$ works well.

Application

Install and load.

```
R> install.packages("gamlss2",
+    repos = c("https://gamlss-dev.R-universe.dev", "https://cloud.R-project.org"))
R> library("gamlss2")
```

Set up a model formula.

```
R> f <- ~ la(~L+R+P1+L1+R1+P2+E2+C2+G2+O2+P3+P7+L7, type=3) +
+      s(month, bs="cc") + s(wday, bs="cc", k=5) + s(hour, k=20, bs="cc")
R> f <- rep(list(f), 4)
R> f[[1]] <- update(f[[1]], P ~ .)
```

Estimate model.

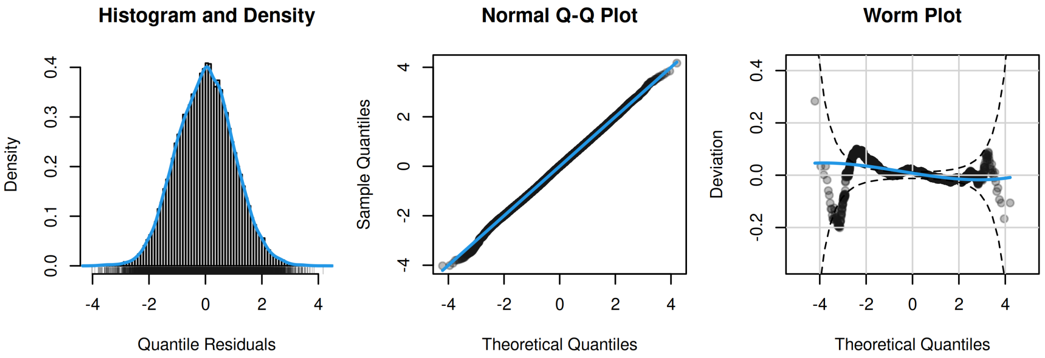
```
R> b <- gamlss2(f, data = df$train, family = JSU,
+    add_ridge = TRUE, criterion = "BIC")
```

Model summary.

```
R> summary(b)
```

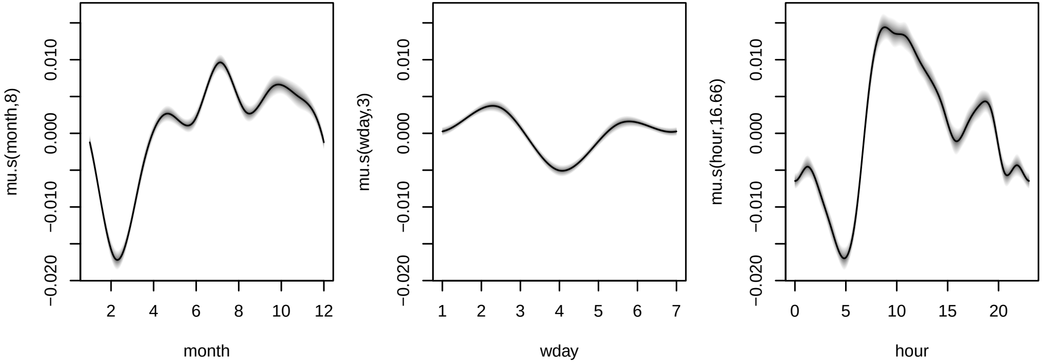
Residuals

```
R> plot(b, which = "resid")
```



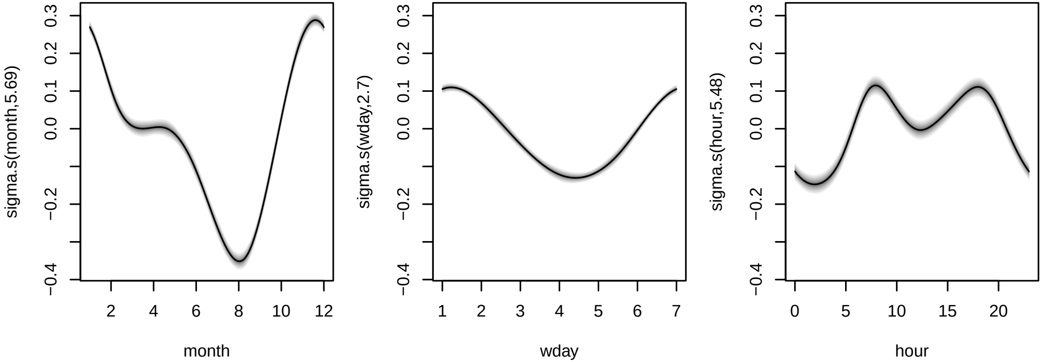
Effects

```
R> plot(b, model = "mu")
```



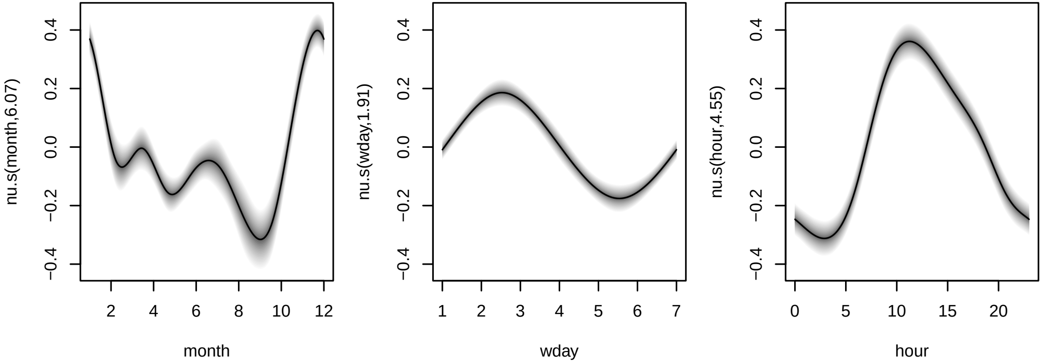
Effects

```
R> plot(b, model = "sigma")
```



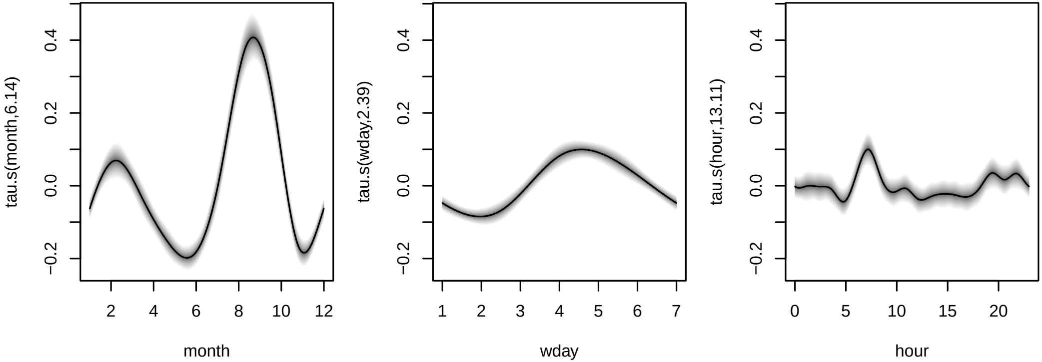
Effects

```
R> plot(b, model = "nu")
```



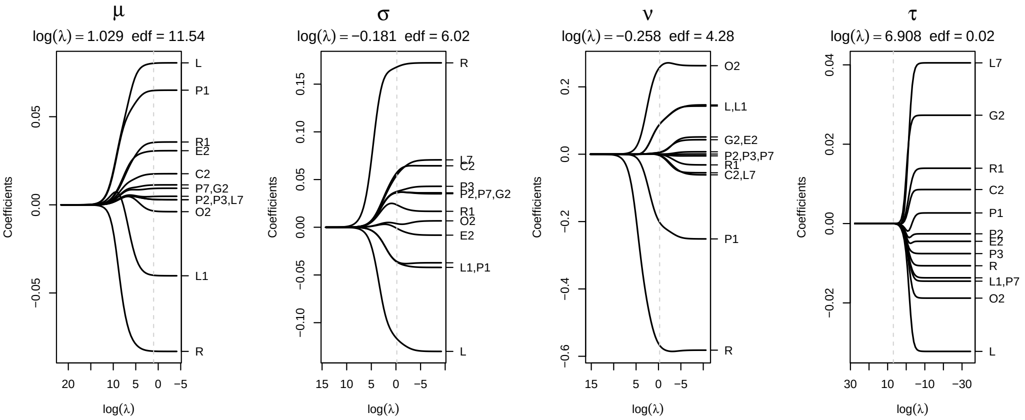
Effects

```
R> plot(b, model = "tau")
```



Coefficients

```
R> plot_lasso(b, which = "coefficients")
```



Predictions

Predict parameters.

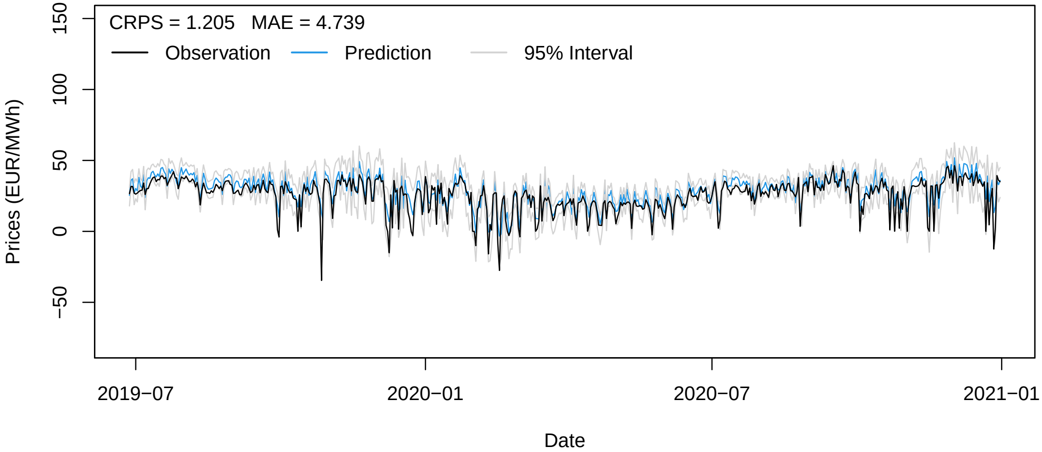
```
R> par <- predict(b, newdata = df$test)
```

Predict quantiles.

```
R> qu <- c(0.01, 0.1, 0.5, 0.9, 0.99)
R> p <- NULL
R> for(i in qu) {
+    p <- cbind(p, family(b)$q(i, par) * 100)
+  }
R> colnames(p) <- paste0(qu * 100, "%"); rownames(p) <- df$test$time
R> print(head(p))
                          1%      10%      50%      90%      99%
2019-06-27 00:00:00 17.18461 23.70882 27.80250 30.93435 34.71296
2019-06-27 01:00:00 15.12706 21.57546 25.62057 28.71319 32.44268
2019-06-27 02:00:00 14.75880 21.21844 25.24996 28.28070 31.88446
2019-06-27 03:00:00 13.73722 20.18522 24.22360 27.29841 30.99556
2019-06-27 04:00:00 13.94446 20.42561 24.43978 27.55404 31.44244
2019-06-27 05:00:00 15.49339 21.95250 25.99306 29.34135 33.79201
```

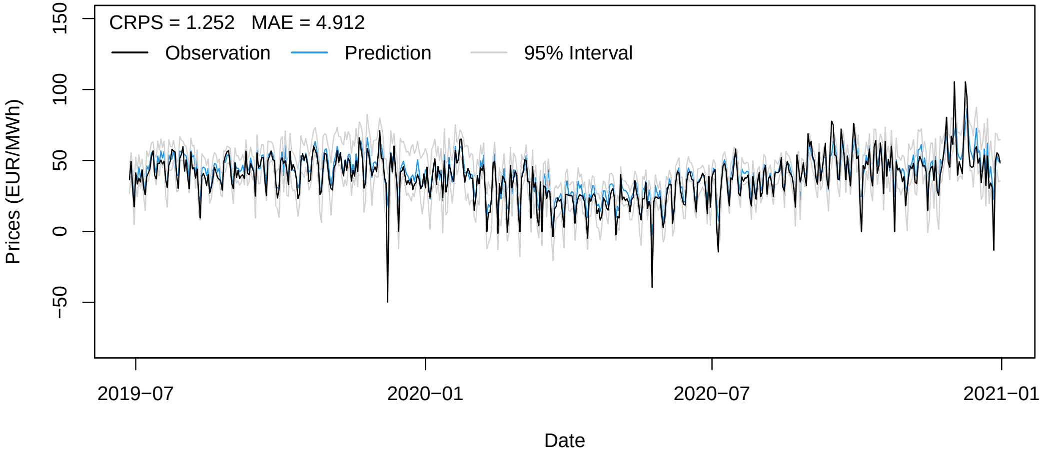
Hourly Prices

Hour 1, forecast:



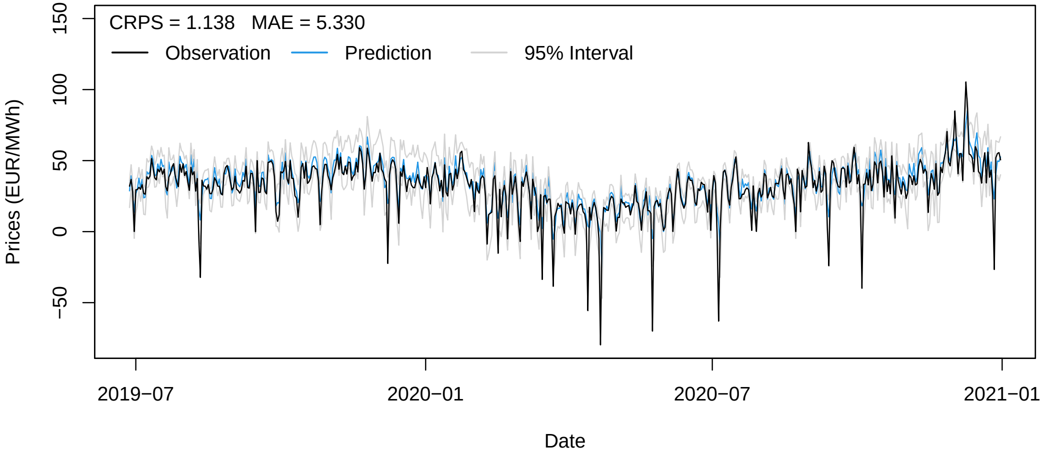
Hourly Prices

Hour 9, forecast:



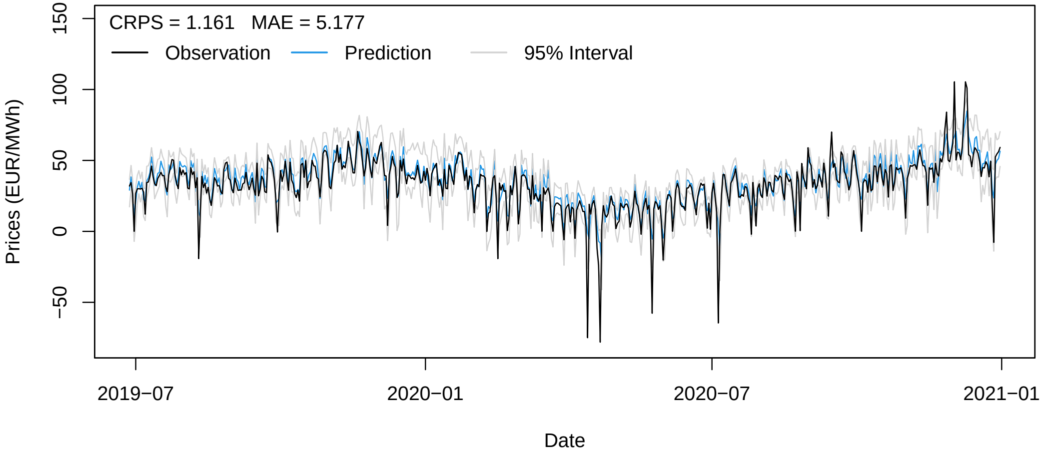
Hourly Prices

Hour 12, forecast:



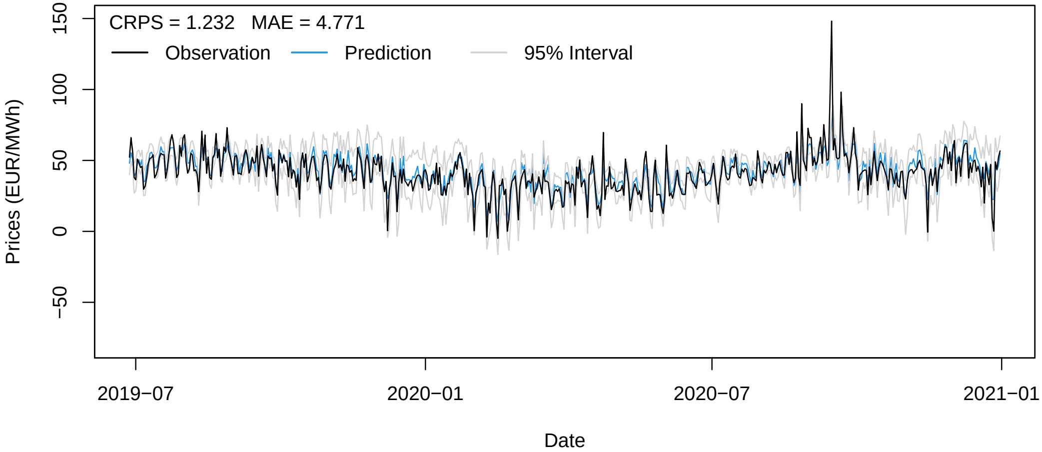
Hourly Prices

Hour 16, forecast:



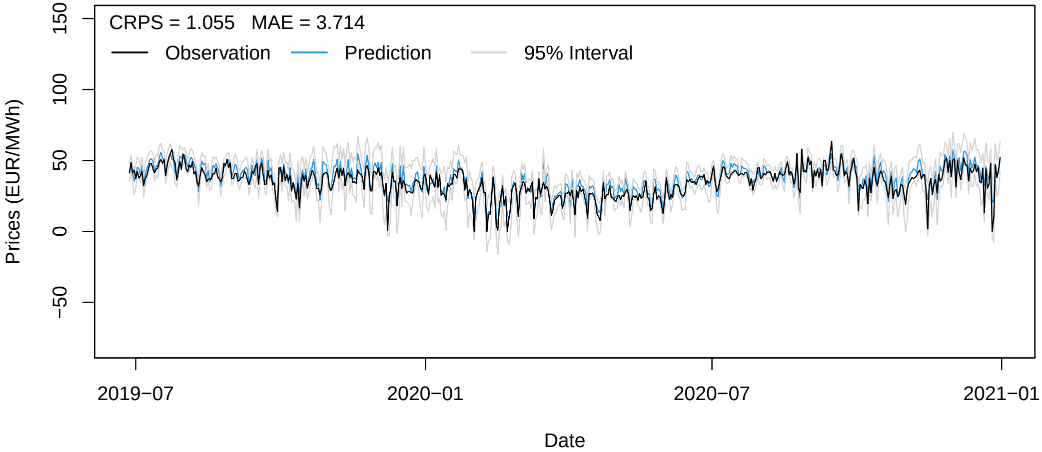
Hourly Prices

Hour 20, forecast:



Hourly Prices

Hour 22, forecast:



References

- Tibshirani (1996). *Regression Shrinkage and Selection via the Lasso*. Journal of the Royal Statistical Society B 58, `doi:10.1111/j.2517-6161.1996.tb02080.x`

- Rigby, and Stasinopoulos (2005). *Generalized Additive Models for Location, Scale and Shape*. Journal of the Royal Statistical Society: Series C (Applied Statistics), `doi:10.1111/j.1467-9876.2005.00510.x`

- Oelker, and Tutz (2017). *A Uniform Framework for the Combination of Penalties in Generalized Structured Models*. Advances in Data Analysis and Classification, `doi:10.1007/s11634-015-0205-y`

- Umlauf, Klein, and Zeileis (2018). *BAMLSS: Bayesian Additive Models for Location, Scale and Shape (and Beyond)*. Journal of Computational and Graphical Statistics, `doi:10.1080/10618600.2017.1407325`

- Groll, Hambuckers, Kneib, and Umlauf (2019). *LASSO-Type Penalization in the Framework of Generalized Additive Models for Location, Scale and Shape*. Computational Statistics & Data Analysis, `doi:10.1016/j.csda.2019.06.005`

- Stasinopoulos, Rigby, Umlauf, Stauffer, and Zeileis (2024). *gamlss2: GAMLSS Infrastructure for Flexible Distributional Regression*, R package version 0.1-0, `https://gamlss-dev.github.io/gamlss2/`