STATY

Documentation

v0.1

1 About

STATY is an educational project designed and developed by Danijela Markovic and Oskar Kärcher with the aim of improving data literacy among students of natural and social sciences.

STATY lets you focus on the analysis and interpretation of the results from many different statistical and machine learning methods using top-notch libraries without any need to write a single line of code:

› Prepare your data for analysis with data cleaning, data imputation and variable transformation

› Examine different visualizations of your variables, even of geospatial data, and find patterns in your data

› Explore web scraping, text data analysis and further data analysis tools like Distribution fitting, Correlation analysis or ANOVA to dive deeper into the characteristics of your included variables

› Use the modelling method that suits your data best:
Time series models like MA, ARMA or ARIMA, panel data models with fixed and random effects, or further regression models including multiple linear regression, logistic regression, generalized additive models, random forest, boosted regression trees and artificial neural networks

Start by simply uploading your data and letting **STATY** do the work for you!

2 Access

You can access **STATY** right here!

https://quant-works.de/staty/

STATY will open in a tab of your browser.

3 Features

STATY has many different methods for specific types of data. You can select the respective feature for analysing your data without much effort! In every feature available for data analysis all output tables (as xlsx-file) and figures are available for download.

3.1 General

Before diving into the different methodological applications, here are some basic information.

3.1.1 Notation

Throughout the documentation the following notation will be used:

$n = sample\ size$

$k = number\ of\ (explanatory)\ variables$

$X = matrix\ of\ (explanatory)\ variables$

$x_k = (explanatory)\ variable\ k$

$x_i = i^{th}\ observation\ of\ the\ matrix\ X$

$x_{ik} = i^{th}\ observation\ of\ (explanatory)\ variable\ k$

$Y = vector\ for\ the\ response\ variable$

$y_i = i^{th}\ observation\ of\ the\ response\ variable\ Y$

$\beta = coefficient\ vector$

$\beta_k = coefficient\ of\ explanatory\ variable\ k$

$\hat{y}_i = i^{th}\ model\ prediction\ for\ the\ i^{th}\ observation\ of\ the\ response\ variable$
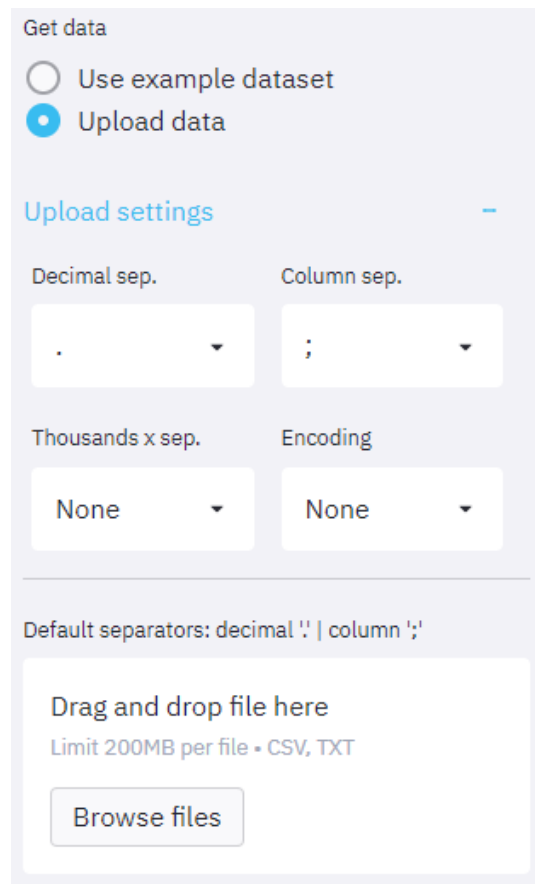
$\varepsilon = error$

$\varepsilon_i = i^{th}\ model\ error\ of\ the\ i^{th}\ observation\ of\ the\ response\ variable$

$N = number\ of\ groups\ in\ grouped\ data$

$T = number\ of\ time\ periods\ in\ panel\ data$

3.1.2 Data upload

You upload you own data in the sidebar. Here you have different upload settings, so that **STATY** identifies your data correctly (Figure 1). Please check your data prior to uploading, so that you can select the decimal separator, column separator, thousands separator and the encoding accordingly.

Figure 1 Data upload settings.

3.1.3 Data summary

After uploading your data, **STATY** offers you many different parameters for summarizing your data before further data processing and analyses. Besides information about missing values, duplicates, number of unique values and the data types, you have the option to inspect main summary statistics:

$$Mean_k = \frac{1}{n} \sum_i x_{ik} = \bar{x}_k$$

$$Mode_k = value\ of\ variable\ k\ that\ occurs\ most\ often$$

$$Median_k = 50\%\ quantile$$

$$Standard\ deviation_k = \sqrt{\frac{1}{n-1} \sum_i (x_{ik} - \bar{x}_k)^2} = s_k$$

$$Variance_k = \frac{1}{n-1} \sum_i (x_{ik} - \bar{x}_k)^2 = s_k^2$$

$$Skewness_k = see\ \underline{here}$$

$$Kurtosis_k = see\ \underline{here}$$

$$Min_k = \min_i x_{ik}$$

$$p\% - quantiles_k: \frac{number\ of\ values\ of\ x_k \leq p-quantile}{n} \geq p\ \&\ \frac{number\ of\ values\ of\ x_k \geq p-quantile}{n} \geq 1 - p$$

$$Max_k = \max_i x_{ik}$$

3.1.4 Data processing

Prepare your data for the final data analysis with several different data processing options.

Data cleaning

Duplicate rows, rows with missing values, as well as manually selected rows and columns can be deleted. Rows and specific columns that should be kept can also be selected.

Data filtering

You can filter data by the categories of a variable.

Data imputation

Missing values for numeric variables can be replaced by either the mean, mode, median or some random value of the respective variable. For other variable types, missing values can only be replaced by the mode or a random value. The imputation can be also conducted for grouped data in some cases.

Data transformation

Variables can be transformed with log, sqrt, by squaring, centering, standardizing, and normalizing.

The log and sqrt transformation are only calculated for positive values. If the minimum of the selected variable is <0, then $\min(x_i) + 1$ is added to every realization of the selected variable.

If a variable should be centred, the mean of the variable is subtracted.

Please note that standardization and normalization are only carried out, if the standard deviation is ≠0 and $(\max - \min) \neq 0$, respectively.

Additionally, variables (without NAs) can be transformed into integer-type (numeric) categories, which are either based on the sorted values of the selected variable or on manually assignable categories for up to maximal 5 categories, multiplied and also divided.

Variables selected for transformation will be not replaced in the data set but added as new columns. Transformation options will be shown based on the uploaded data.

3.2 Uni- and bivariate data

For data with one or two variables, or for analyses of maximum two variables, specific tools can be used to explore the relationships between variables, the distribution type of included variables and differences between groups of a discrete variable. You can use frequency analysis, ANOVA, distribution fitting, correlation analysis, linear regression techniques and contingency tables to find out more about your data.

3.2.1 Univariate frequency analysis

With the univariate frequency analysis, you can determine the absolute, relative, and cumulative frequencies of each group or class of a discrete or continuous variable. The respective frequencies can be displayed as bar chart or table. Depending on the variable type, you have additional settings for continuous and discrete variables. For discrete variables, you can additionally specify the order of the x-labels (Figure 2).

Figure 2 Additional settings for discrete variables.

Figure 3 Additional settings for continuous variables.

For continuous variables, you can select the value range used for the frequency analysis and the number of classes (Figure 3). Furthermore, it can be selected whether the data used for the analysis should be included in the output file available for download or not for both cases.

The relative and cumulative frequency for each group x_j of variable x is determined as follows:

$$h_j \ (relative \ frequency) = \frac{n_j}{n},$$

$where \ n_j \ is \ the \ absolute \ frequency \ of \ group \ or \ class \ x_j$

$$F(x_j) = \sum_{i \leq j} h_i \, ,$$

$where \ F \ is \ the \ cumulative \ distribution \ function \ (cdf).$

3.2.2 ANOVA

With Analysis Of Variance (ANOVA) you can compare the means of several groups to test whether the observations of the groups were drawn from populations with the same mean (null hypothesis) or not (alternative hypothesis). The ANOVA table consists of the sum of squares between, sum of squares within and sum of squares total, which are used to calculate the final F-statistic:

$$SS_{between} = \sum_{i=1}^{N} n_i(x_i - \bar{x})^2, \qquad df = N - 1$$

$$SS_{within} = \sum_{i=1}^{N} (n_i - 1)s_i^2, \qquad df = n - N$$

$$SS_{total} = SS_{between} + SS_{total}$$

$$MS_{between} = \frac{SS_{between}}{N-1}$$

$$MS_{within} = \frac{SS_{within}}{n-N}$$

$$F\ statistic = \frac{MS_{between}}{MS_{within}}$$

For the ANOVA test result to be reliable, the following assumptions must be fulfilled:

$(1)\ Independent\ samples\ of\ the\ response\ variable$

$(2)\ Homogeneity\ of\ variance\ across\ groups$

$(3)\ Normal\ distribution\ across\ groups$

In order to perform one-way ANOVA, you have to select the target variable that you want to analyse as well as the variable that should be used as a classifier variable (Figure 4).

Figure 4 ANOVA settings.

Furthermore, it can be selected whether the data used for the analysis should be included in the output file available for download or not.

Assumptions can be checked by using the provided boxplots and residual distribution plots based on the regression problem $y \sim x_{categories}$. Here, the standardized residuals are calculated as $\frac{residuals - residuals_{mean}}{residuals_{standard\ deviation}}$.

3.2.3 Hypothesis testing

With hypothesis testing by using the z-test or one sample location t-test you can test whether the deviation of the mean value based on your sample from a specified value or population mean is statistically significant or not. By using the two sample location t-test you can compare two mean values by grouping your data and test for a statistically significant deviation. Thus, you can identify whether the unknown population means for these two independent groups are equal or not.

Based on the test that you want to conduct, you must provide different information. For the z-test you must enter the population mean value and the population variance (Figure 5).

Figure 5 Settings for the z-test.

$$z\ statistic = \frac{\bar{x} - \mu_0}{\sqrt{\frac{\sigma^2}{n}}}$$

$where$

$n = sample\ size$

$\mu_0 = population\ mean$

$\sigma^2 = population\ variance$

$p - value = according\ to\ z\ statistic$

The test statistic follows the normal distribution.

For the one sample location t-test only the population mean must be provided (Figure 6).

Select the test

One sample location t-test

Select the test variable

UrbanPopulation

Enter sollwert/population mean

0,00000000   −   +

Figure 6 Settings for one sample location t-test.

$$t\ statistic = \frac{\bar{x} - \mu_0}{\sqrt{\frac{s^2}{n}}}$$

$where$

$n = sample\ size$

$\mu_0 = population\ mean$

$s^2 = sample\ variance$

$DOF\ (degrees\ of\ freedom) = n - 1$

$p - value = according\ to\ t\ statistic$

The test statistic follows a t-distribution.

For the two sample location t-test you have to provide information about the variance assumption (either unequal or equal variance for the two samples) and the variable that should be used to split the data into two samples. If unequal variance was selected, Welch-test is performed. In case of more than two realisations of the variable used for grouping, you can reclassify your data, whereas reclassification options depend on the variable type (Figure 7).

Select the test

Two sample location t-test

Select the test variable

UrbanPopulation

Select variance assumption

unequal variance

The variable LegalOrigin has more than two realisations! Choose another variable or reclassify LegalOrigin!

Select the sample info variable

LegalOrigin

☑ Reclassify LegalOrigin

First group is where values are ...

english

Figure 7 Settings for two sample location t-test.

<u>Equal variance</u>

$$t\ statistic = \sqrt{\frac{n_1 * n_2}{n_1 + n_2}}\ \frac{\bar{x}_1 - \bar{x}_2}{s}$$

$where$

$n_1 = sample\ size\ of\ first\ sample$

$n_2 = sample\ size\ of\ second\ sample$

$\bar{x}_1 = mean\ value\ of\ first\ sample$

$\bar{x}_2 = mean\ value\ of\ second\ sample$

$$s = \sqrt{\frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}}$$

$with$

$s_1^2 = variance\ of\ first\ sample$

$s_2^2 = variance\ of\ second\ sample$

$DOF\ (degrees\ of\ freedom) = n_1 + n_2 - 2$

$p - value = according\ to\ t\ statistic\ and\ test\ problem$

<u>Unequal variance (Welch test)</u>

$$t\ statistic = \frac{\bar{x}_1 - \bar{x}_2}{s}$$

$where$

$n_1 = sample\ size\ of\ first\ sample$

$n_2 = sample\ size\ of\ second\ sample$

$\bar{x}_1 = mean\ value\ of\ first\ sample$

$\bar{x}_2 = mean\ value\ of\ second\ sample$

$$s = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$$

$with$

$s_1^2 = variance\ of\ first\ sample$

$s_2^2 = variance\ of\ second\ sample$

$$DOF\ (degrees\ of\ freedom) = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{n_1 - 1} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{n_2 - 1}}$$

$p - value = according\ to\ t\ statistic\ and\ test\ problem$

The test statistic follows a t-distribution.

3.2.4 Distribution fitting

For the variables in your data set you can determine the distribution that fits your data best. The best fit is based on the Chi-Square Goodness of Fit Test and the selected distributions (Figure 8). You can find a list of the continuous distributions included here. You can additionally specify the range and the number of bins for the histogram, which will be used to determine the best fit (Figure 8).

Select a variabe for dist. fitting

logGDPI

Please choose if you would like to fit all distributions or a selection of distributions:
- ( ) all (Note, this may take a while!)
- (•) selection

Select theoretical distributions for distribution fitting

[Normal X] [Lognormal X] [Weibull minimum X]

[✓] Additional settings

Sample value range for the variable "logGDPI":
min=5.092489
max=9.619843

Start fitting from the "logGDPI" value?

0,00   −   +

End fitting at the "logGDPI" value?

0,00   −   +

Number of classes for your histogram?

10   −   +

[ ] Include data from distribution fitting in the output file

Figure 8 Settings for distribution fitting.

The output of the distribution fitting is given in tabular form and includes the test results and the relative and expected relative frequencies for each class of the best fit distribution.

$$SSD \; (sum \; of \; squared \; differences) = \sum_{i=1}^{\# \, classes} \left(n_i - n_{expected,i} \right)^2$$

where n_i and $n_{expected,i}$ are the absolute frequency and the expected absolute frequency of the respective distribution of group i.

$$Chi-squared\ (test\ statistic) = \sum_{i=1}^{\#\ classes} \frac{\left(n_i - n_{expected,i}\right)^2}{n_{expected,i}}$$

$$DOF\ (degrees\ of\ freedom) = m_1 - m_2 - 1$$

where m_1 corresponds to the number of classes with non-zero observations, m_2 is the number of parameters needed for the respective distribution.

$$p-value = corresponding\ to\ the\ test\ statistic$$

The test statistic follows a χ^2 distribution.

Moreover, it can be selected whether the data used for the analysis should be included in the output file available for download or not (Figure 8).

3.2.5 Correlation analysis

The correlation coefficient for two variables can be determined by using the method of Pearson, Kendall, or Spearman.

Pearson:

$$r = \frac{Cov(x,y)}{std(x)*std(y)}$$

Kendall:

$$r = \frac{n_c - n_d}{\binom{n}{2}}$$

where n_c is the number of concordant pairs and n_d the number of discordant pairs.

Spearman:

$$r = \frac{Cov(rank(x),\ rank(y))}{std(rank(x))*std(rank(y))}$$

While Pearson is a parametric method, Kendall and Spearman are non-parametric methods based on the order and ranks of observation pairs.

To perform the correlation analysis, you can select the variables for which the coefficients should be calculated (Figure 9). Please note non-numerical variables will be automatically excluded.

Figure 9 Correlation analysis settings.

3.2.6 Regression techniques

Having two numeric variables, you can use linear regression models to further explore the relationship between the two selected variables. You can choose from different types of linear regressions: Simple Linear Regression, Linear-Log Regression, Log-Linear Regression, Log-Log Regression and Polynomial Regression.

All regression techniques predict the selected response variable by establishing a linear relationship between the explanatory and the response variable based on the chosen technique (Figure 10). The optional detailed output includes ANOVA/ F-test and coefficient estimates. It can also be selected whether the data used for the analysis should be included in the output file available for download or not

Figure 10 Settings for Regression techniques.

Simple Linear Regression:

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \text{ for } i = 1 \dots n$$

Linear-Log Regression:

$$y_i = \beta_0 + \beta_1 \log(x_i) + \varepsilon_i \text{ for } i = 1 \dots n$$

Log-Linear Regression:

$$\log(y_i) = \beta_0 + \beta_1 x_i + \varepsilon_i \text{ for } i = 1 \dots n$$

Log-Log Regression:

$$\log(y_i) = \beta_0 + \beta_1 \log(x_i) + \varepsilon_i \text{ for } i = 1 \dots n$$

Polynomial Regression:

$$y_i = \beta_0 + \beta_1 x_i + \sum_{j=2}^{m} x_i^j + \varepsilon_i \text{ for } i = 1 \dots n$$

where m determines the maximum power of the transformed explanatory variable (Figure 11);

20

Figure 11 Settings for the polynomial regression.

and where

$n = $ *number of observations*

$y = $ *response variable*

$x = $ *explanatory variables*

$\beta_1 = $ *coefficient of explanatory variable* x

$\beta_0 = $ *intercept*

$\varepsilon = $ *error*.

The coefficients are estimated with the Ordinary Least Squares (OLS) estimator. For the OLS estimator to be the Best Linear Unbiased Estimator (BLUE), the following assumptions must be fulfilled:

(1) $E(\varepsilon_i) = 0 \; \forall i \; (mean \; error \; of \; 0)$

(2) $Var(\varepsilon_i) = \sigma^2 \; \forall i \; (homoskedasticity)$

(3) $Cov(\varepsilon_i, \varepsilon_j) = 0 \; \forall i \neq j \; (uncorrelated \; errors)$.

When these assumptions are met, the OLS estimator is BLUE according to the Gauss-Markov theorem.

For the definition and description of the output, please see 3.3.1 Multiple Linear Regression where the same estimator is used.

3.2.7 Contingency tables and association measures

In case of a bivariate or a two-dimensional frequency distribution of two discrete variables, you can perform a contingency analysis to identify the strength of the association between the two variables. In case you want to include a continuous variable, you have the option to either reclassify your data or use your data as is (Figure 12).

Figure 12 Options for continuous variables in contingency tables and association measures.

The reclassification allows you to specify the range of the data that should be used and the number of created classes (Figure 13).

Figure 13 Classification options for continuous variables.

The strength of association is measured with the corrected Pearson contingency coefficient, which is based on the chi-squared value. You can also optionally select to display the marginal frequencies for the contingency analysis, show the used data for the analysis and whether the data should be included in the output file.

$$n_{expected,i} = \frac{marginal\ frequency\ of\ x\ for\ combination\ i * marginal\ frequency\ of\ y\ for\ combination\ i}{n}$$

$$\chi^2 = \sum_{i=1}^{\# \ combinations} \frac{\left(n_i - n_{expected,i}\right)^2}{n_{expected,i}}$$

$$p - value = \ corresponding \ to \ \chi^2 - statistic$$

$$K = \sqrt{\frac{\chi^2}{\chi^2 + n}}$$

$$K_{cor} = \frac{K}{K_{max}}$$

$$K_{max} = \sqrt{\frac{\min(k,l) - 1}{\min(k,l)}}, \quad where \ k = \#rows \ and \ l = \#columns \ of \ the \ contingency \ table$$

3.3 Multivariate data

Having multivariate data, you can use the Regression section with machine learning models to predict a continuous or binary response variable or Data decomposition for dimensionality reduction. With the help of the models, you are also able to identify variable importance and influence directions of the explanatory variables.

To adjust the models to your specific data set, you can select several settings for the included modelling techniques. Moreover, you have the option to tune and validate your models by running hyperparameter-tuning and a model validation, respectively.

Regression

3.3.1 Multiple Linear Regression

Definition

Multiple Linear Regression (MLR) is a regression technique that uses several explanatory variables to predict a response variable by establishing linear relationships between the explanatory variables and the response variable:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \varepsilon_i \text{ for } i = 1 \dots n,$$ or in matrix form $Y = X\beta + \varepsilon,$

where

$n = number\ of\ observations$

$k = number\ of\ explanatory\ variables$

$y = response\ variable$

$x_j = explanatory\ variables, j = 1 \dots k$

$\beta_j = coefficient\ of\ explanatory\ variable\ x_j$

$\beta_0 = intercept$

$\varepsilon = error.$

The coefficients are estimated with the Ordinary Least Squares (OLS) estimator. For the OLS estimator of the MLR to be the Best Linear Unbiased Estimator (BLUE), the following assumptions must be fulfilled:

$(1)\ E(\varepsilon_i) = 0\ \forall i\ (mean\ error\ of\ 0)$

$(2)\ Var(\varepsilon_i) = \sigma^2\ \forall i\ (homoskedasticity)$

$(3)\ Cov(\varepsilon_i, \varepsilon_j) = 0\ \forall i \neq j\ (uncorrelated\ errors).$

When these assumptions are met, the OLS estimator is BLUE according to the Gauss-Markov theorem.

Settings

For MLR, it can be selected whether an intercept should be included in the model or not and which covariance type should be used (Figure 14). If the homoskedasticity assumption is violated (see Output for heteroskedasticity tests), estimations for the standard error of coefficients and consequently for the statistical tests are unreliable. Besides a non-robust covariance matrix Σ, you can select a robust covariance matrix Σ (HC = heteroskedasticity-consistent) to derive more reliable conclusions from statistical tests.

Figure 14 Settings for Multiple Linear Regression including intercept and covariance estimator choices.

<u>Non-robust</u>

$$\Sigma = \sigma^2 (X^T X)^{-1}$$

<u>HC0</u>

$$\Sigma = (X^T X)^{-1} X^T \ diag((y_i - \hat{y}_i)^2) \ X (X^T X)^{-1}$$

<u>HC1</u>

$$\Sigma = \frac{n}{n-k-1} (X^T X)^{-1} X^T \ diag((y_i - \hat{y}_i)^2) \ X (X^T X)^{-1} \ (degree \ of \ freedom \ adjustment)$$

<u>HC2</u>

$$\Sigma = (X^T X)^{-1} X^T \ diag\left(\frac{(y_i - \hat{y}_i)^2}{1 - h_{ii}}\right) X (X^T X)^{-1} \ (leverage \ adjustment)$$

$$\Sigma = (X^TX)^{-1}X^T \, diag\left(\frac{(y_i-\hat{y}_i)^2}{(1-h_{ii})^2}\right) X(X^TX)^{-1} \; (squared \; leverage \; adjustment)$$

where

$\hat{y}_i = \; prediction \; of \; the \; model \; for \; y_i$

$diag((y_i - \hat{y}_i)^2) = \; diagonal \; matrix \; with \; squared \; residuals \; on \; the \; diagonal$

$h_{ii} = x_i(X^TX)^{-1}x_i^T$, which are the diagonal values (leverage values) of the hat matrix $X(X^TX)^{-1}X^T$.

Output

Regression statistics

$$R^2 = 1 - \frac{\sum_i(y_i - \hat{y}_i)^2}{\sum_i(y_i - \bar{y})^2} = \frac{\sum_i(\hat{y}_i - \bar{y})^2}{\sum_i(y_i - \bar{y})^2}$$

$$Adj. \; R^2 = 1 - \frac{(1-R^2)(n-1)}{n-k-1}$$

$$Multiple \; correlation \; coefficient = \sqrt{R^2}$$

$$Residual \; standard \; error = \sqrt{\frac{\sum_i(y_i - \hat{y}_i)^2}{n-k-1}}$$

$$Log \; likelihood = \max_\beta \log\left(\prod_i \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(\frac{-(y_i-x_i\beta)^2}{2\sigma^2}\right)\right)$$

$$AIC \; (Akaike \; Information \; Criterion) = 2(k+1) - 2 \; Log \; likelihood$$

$$BIC \; (Bayesian \; Information \; Criterion) = (k+1)\ln(n) - 2 \; Log \; likelihood$$

ANOVA

$$Explained \; Sum \; of \; Squares \; (ESS) = \sum_i (\hat{y}_i - \bar{y})^2$$

$$Residual \; Sum \; of \; Squares \; (SSR) = \sum_i (y_i - \hat{y}_i)^2$$

$$Total \; Sum \; of \; Squares \; (SST) = \sum_i (y_i - \bar{y})^2$$

$$Mean \; Sum \; of \; Squares \; (MS) = \frac{Sum \; of \; Squares}{DF}$$

$$F \; statistic \; (non-robust) = \frac{MS_{explained}}{MS_{residual}}$$

$$F \; statistic \; (robust) = \; Wald \; statistic$$

The test statistic follows a F-distribution.

Coefficients

$$Coefficients = \beta \; estimates \; of \; the \; OLS \; estimator$$

$$Standard \; error = diagonal \; of \; the \; covariance \; matrix \; \Sigma$$

$$t - statistic = \frac{coefficient}{standard\ error}$$

$$p - value = \ corresponding\ to\ t - statistic$$

$$lower\ 95\%\ confidence\ \approx coefficient - 1.96 * standard\ error$$

$$upper\ 95\%\ confidence\ \approx coefficien + 1.96 * standard\ error$$

Heteroskedasticity test

Breusch-Pagan test: No interactions are included. Estimation of γ for (e.g., with two explanatory variables)

$$(y - \hat{y})^2 = \ \gamma_0 + \ \gamma_1 x_1 + \ \gamma_2 x_2 + \varepsilon$$

White test (without interactions): No interactions, but squared explanatory variables are added. Estimation of γ for (e.g., with two explanatory variables)

$$(y - \hat{y})^2 = \ \gamma_0 + \ \gamma_1 x_1 + \ \gamma_2 x_2 + \ \gamma_3 x_1^2 + \ \gamma_4 x_2^2 + \varepsilon$$

White test (with interactions): Squared explanatory variables and interactions of the explanatory variables are added. Estimation of γ for (e.g., with two explanatory variables)

$$(y - \hat{y})^2 = \ \gamma_0 + \ \gamma_1 x_1 + \ \gamma_2 x_2 + \ \gamma_3 x_1^2 + \ \gamma_4 x_2^2 + \ \gamma_5 x_1 x_2 + \varepsilon$$

For the tests, the following test statistic is used:

$$Test\ statistic = \ nR^2.$$

The test statistic follows a χ^2 distribution.

Variable importance (via permutation)

For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. R² is used as scoring metric to determine the importance:

$$importance = score_{full\ model} - \frac{1}{10} \sum_{i=1}^{10} score_i.$$

Please note that the importance might be misleading for strongly correlated explanatory variables.

Analytical plots
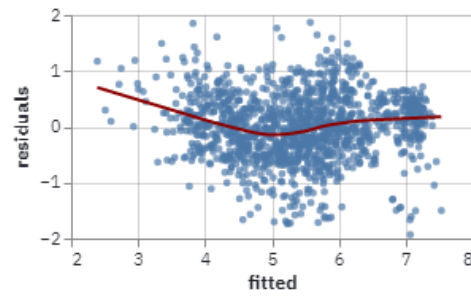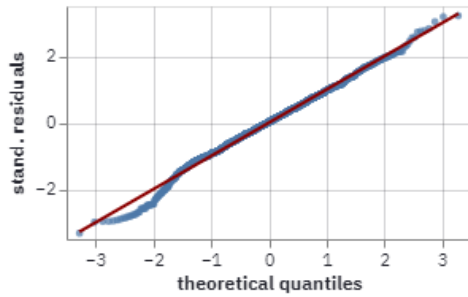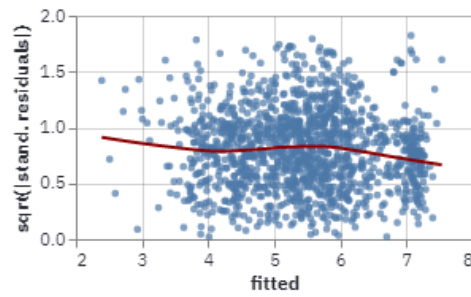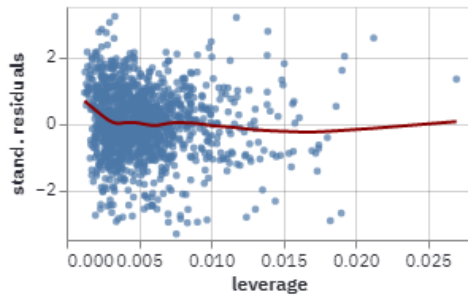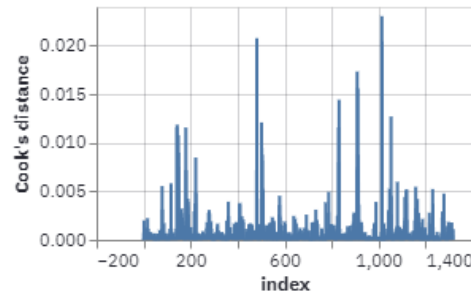
Figure 15 Graphical output of Multiple Linear Regression.

The leverage in Figure 15 corresponds to the diagonal values of the hat matrix and Cook's distance is based on a re-calculated regression where observation i was removed:

$$Cook's\ distance_i = \frac{\Sigma_{j=1, j \neq i}^{n}\left(\hat{y}_j - \hat{y}_{j, without\ obs.i}\right)^2}{(k+1)\sigma^2}.$$

3.3.2 Logistic Regression

Definition

Logistic Regression (LR) is a regression technique that models the probability of occurrence for a response variable with two realizations. Modelling the probability is approached by at first formulating the regression problem using logit:

$$logit(y_{0,1}) = log\left(\frac{P(y=1)}{1-P(y=1)}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k = X\beta$$

where

$k = number\ of\ explanatory\ variables$

$y = binary\ response\ variable$

$x_j = explanatory\ variables, j = 1 \ldots k$

$\beta_j = coefficient\ of\ explanatory\ variable\ x_j$

$\beta_0 = intercept.$

After transforming the above equation, the probability for $P(y = 1)$ can be estimated

$$P(y = 1) = \frac{1}{1+\exp\left(-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k)\right)} = \frac{1}{1+\exp(-X\beta)}.$$

The coefficients are determined with the Maximum Likelihood Estimator (MLE).

Settings

For LR, it can be selected whether an intercept should be included in the model or not and which covariance type should be used (Figure 16). Currently, only "HC0" is available.

Figure 16 Settings for Logistic Regression including intercept and covariance estimator choices.

Output

<u>Regression statistics</u>

$AUC\ ROC = $ area under the receiver operating characteristic curve

$Pseudo\ R^2 = 1 - \frac{LLF}{LL-Null}$

$Log\ likelihood\ (LLF) = \max_{\beta} \sum_i y_i \log\left(\frac{1}{1+\exp(-(x_i\beta))}\right) + (1-y_i)\log\left(1 - \frac{1}{1+\exp(-(x_i\beta))}\right)$

$LL - Null = Log\ likelihood\ of\ the\ constant - only\ model$

$Residual\ deviance = -2\ LLF$

$Null\ deviance = -2\ LL - Null$

$LLR\ (Log\ likelihood\ ratio) = -2\ (LL - Null - LLF)$

$LLR\ p - value = \chi^2\ probability\ corresponding\ to\ LLR$

$AIC\ (Akaike\ Information\ Criterion) = 2(k+1) - 2\ Log\ likelihood$

$BIC\ (Bayesian\ Information\ Criterion) = (k+1)\ln(n) - 2\ Log\ likelihood$

<u>Coefficients</u>

$Coefficients = \beta\ estimates\ of\ the\ MLE$

$Standard\ error = diagonal\ of\ the\ covariance\ matrix\ \Sigma$

$t - statistic = \frac{coefficient}{standard\ error}$

$p - value = corresponding\ to\ t - statistic$

$lower\ 95\%\ confidence \approx coefficient - 1.96 * standard\ error$

$upper\ 95\%\ confidence \approx coefficien + 1.96 * standard\ error$

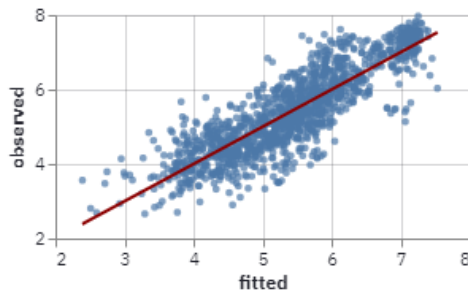<u>Variable importance (via permutation)</u>

For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. AUC ROC is used as scoring metric to determine the importance:

$importance = score_{full\ model} - \frac{1}{10}\sum_{i=1}^{10} score_i.$

Please note that the importance might be misleading for strongly correlated explanatory variables.

Analytical plots
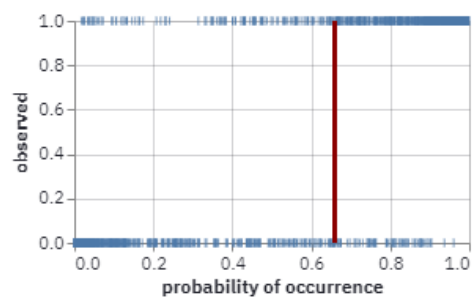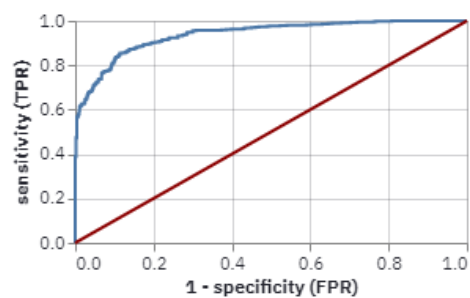
*Figure 17 Graphical output of Logistic Regression.*

In Figure 17 the threshold corresponds to the Youden Index:

$$threshold = \max_{t \in thresholds} (sensitivity_t + specificity_t - 1).$$

Partial probability plots

Partial probability plots are based on univariate LR modelling with each selected explanatory variable.

3.3.3 Generalized Additive Models

Definition

Generalized Additive Models (GAMs) are flexible and additive modelling techniques for identifying non-linear relationships between the response and explanatory variables by using smoothing functions, e.g., based on splines. GAMs have the following general form:

$$g\big(E(Y|X)\big) = \beta_0 + f_1(X_1) + f_2(X_2) + \cdots + f_k(X_k)$$

Where

$k = number\ of\ explanatory\ variables$

$Y = response\ variable$

$X_j = explanatory\ variables, j = 1 \ldots k$

$f_j = smooth\ feature\ function\ for\ X_j$

$g = link\ function$

$\beta_0 = intercept$

$E(Y|X) = expected\ value\ of\ Y.$

Y follows an exponential family distribution, here either normal or binomial for continuous and binomial response variables, respectively. The link function is defined accordingly, such that for normal distributions the identity function and for binomial distributions the logit function is used:

$$E(Y|X) = \beta_0 + f_1(X_1) + f_2(X_2) + \cdots + f_k(X_k)$$

or

$$logit\left(E\big(Y_{0,1}|X\big)\right) = log\left(\frac{P(Y=1|X)}{P(Y=0|X)}\right) = \beta_0 + f_1(X_1) + f_2(X_2) + \cdots + f_k(X_k)\,.$$

The smooth feature functions f_j are based on penalized B-splines, for which several parameters can be set. The smoothing parameter estimation problem is solved by using the generalized cross-validation (GCV) and un-biased risk estimator (UBRE) score for normal and binomial distributions, respectively.

Settings

For each feature function, the spline settings can be specified, and it can be decided whether an intercept should be included or not (Figure 18). The number of splines or basis functions used for the estimation of the feature function, the spline order, which determines the degree of the piecewise polynomial functions (degree = spline order - 1), and the lambda value controlling for the penalty and thus overfitting can be set manually. For the splines, a second

derivative smoothing is used. Please note that lambda controls for the bias-variance trade-off and higher values of lambda will lead to smoother functions.

Figure 18 Settings for Generalized Additive Models including intercept and spline parameter choices.

Additionally, a search for the best lambda value can be conducted, which either aims at minimizing GCV or UBRE depending on the response variable.

Figure 19 Lambda search settings.

For the lambda search, the interval boundaries for possible lambda values and the number of values taken from this interval for each selected explanatory variable can be specified (Figure 19). The values taken from the interval are evenly spaced. Besides the settings for the lambda search, further spline parameters can still be specified for each explanatory variable.

Output (continuous response variable)

<u>Regression statistics</u>

$Log\ likelihood$

$AIC\ (Akaike\ Information\ Criterion) = 2(Effective\ DF + 1) - 2\ Log\ likelihood$

$AICc\ (second\ order\ AIC) = AIC + \frac{2\ (Effective\ DF+1)^2 + 2\ (Effective\ DF+1)}{n-(Effective\ DF+1)-1}$

$GCV\ (Generalized\ Cross-Validation\ score) = minimized\ GCV\ score\ of\ fitted\ GAM$

$Scale = residual\ standard\ error\ squared$

$Pseudo\ R^2 = deviance\ explained$

<u>Feature significance</u>

$Coefficients = shown\ if\ intercept\ included$

$Lambda = final\ lambda\ value$

$rank = number\ of\ splines$

$edof = effective\ degrees\ of\ freedom\ after\ regularization$

$p-value = significance\ of\ corresponding\ feature$

<u>Variable importance (via permutation)</u>

For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. R² is used as scoring metric to determine the importance:

$importance = score_{full\ model} - \frac{1}{10}\sum_{i=1}^{10} score_i.$

Please note that the importance might be misleading for strongly correlated explanatory variables.

<u>Partial dependence plots (one-way)</u>

The partial dependence plots display the contribution of each feature to the overall prediction while keeping the other predictors constant and the 95% confidence intervals.

Output (binary response variable)

<u>Regression statistics</u>

$UBRE\ (Un-Biased\ Risk\ Estimator) = minimized\ UBRE\ score\ of\ fitted\ GAM$

$Scale = 1$

$AUC\ ROC = area\ under\ the\ receiver\ operating\ characteristic\ curve$

34

Variable importance (via permutation)

For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. AUC ROC is used as scoring metric to determine the importance:

$$importance = score_{full\ model} - \frac{1}{10} \sum_{i=1}^{10} score_i.$$

Please note that the importance might be misleading for strongly correlated explanatory variables.

Analytical plots

See Output for Logistic Regression.

Partial dependence plots (one-way)

See Output (continuous response variable) for Generalized Additive Models.

3.3.4 Random Forest

Definition

Random Forest (RF) is a tree-based approach that combines several decision trees. The combination of the single trees creates the forest, where each single tree is built based on a bootstrap sample, i.e., a sample consisting of randomly drawn observations with replacement. The single decision trees either use a (randomly selected) subset of the included explanatory variables or all of them for splitting at each node. Thus, two sources of randomness can be included into this modelling technique, which aims at decreasing the variance of the model. The quality of the split is measured by the mean squared error and Gini impurity for continuous and binary response variables, respectively. The prediction of the random forest model is determined by taking the average of the predictions of all trees. Consequently, for binary response variables the probability predictions are averaged.

Hyperparameters settings

For RF models, parameters controlling the size of the forest and the characteristic of each single tree can be specified. Here, the number of trees, maximum tree depth, maximum number of features and sample rate can be tuned (Figure 20).

Random Forest settings

☑ Adjust settings for Random Forest

Range for number of trees
50   500
50   1000

Specify maximum tree depth

No ▾

Range for maximum number of features
1   6
1   6

Range for sample rate
0.80   0.99
0.50   0.99
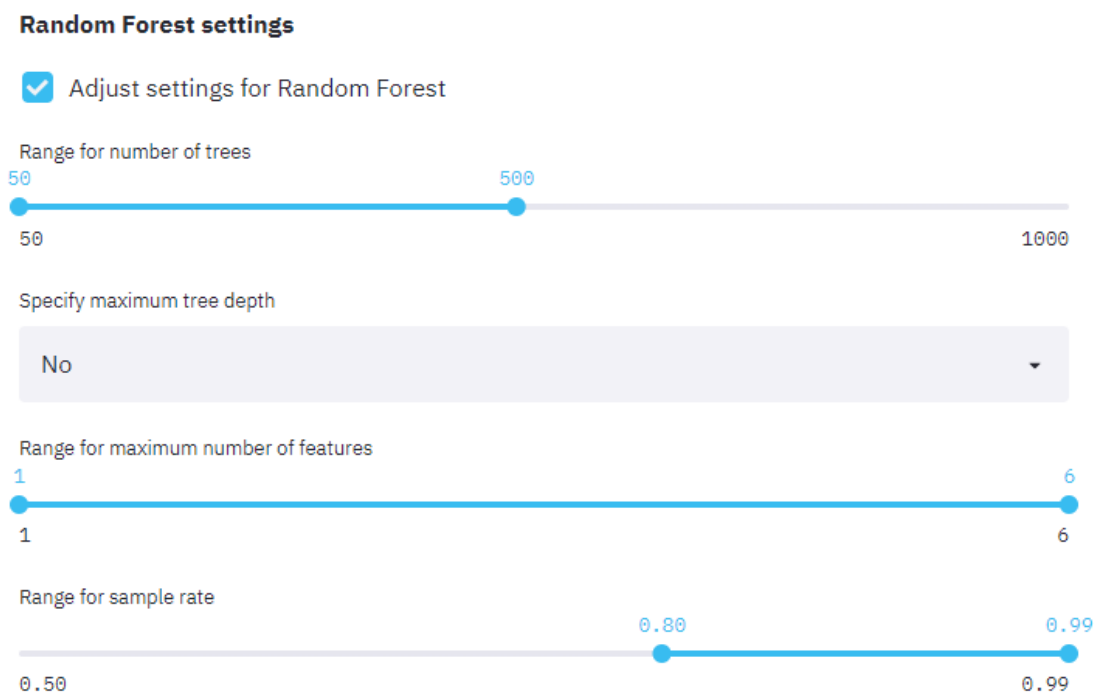
Figure 20 Range options for hyperparameters of Random Forest.

If the maximum tree depth is not specified, no restriction on the maximum tree depth is used and if only one explanatory variable is included, the range for maximum number of features, i.e. the number of features to consider at a split, cannot be specified.

For the search methods grid-search and random grid-search, a grid based on the ranges for each hyperparameter must be created. The grid is formed by using the minimum and maximum of the specified range (Figure 20) and a pre-defined step size (Table 1).

Hyperparameter	Default value	Range	Default range	Step size
Number of trees	100	50 – 1000	50 – 500	5
Maximum tree depth	None	1 – 50	None/ 2 – 10	1
Maximum number of features	#features	1 – #featues	1 – #featues	1
Sample rate	0.99	0.5 – 0.99	0.8 – 0.99	0.05

Table 1 Hyperparameter specifications for Random Forest.

Please note that for the search methods Bayes optimization and sequential model-based optimization the definition of step sizes is not necessary as the next hyperparameter combination is automatically proposed within the specified ranges (see 3.3.7 Hyperparameter-tuning).

Output (continuous response variable)

Regression information

$$OOB \ (out \ of \ bag) \ score = estimation \ of \ the \ generalization \ score \ (R^2)$$

Regression statistics

$$Mean \ squared \ error \ (MSE) = \frac{1}{n} \sum_i (y_i - \hat{y}_i)^2$$

$$Root \ mean \ squared \ error \ (RMSE) = \sqrt{\frac{1}{n} \sum_i (y_i - \hat{y}_i)^2}$$

$$Mean \ absolute \ error \ (MAE) = \frac{1}{n} \sum_i |y_i - \hat{y}_i|$$

$$Residual \ standard \ error = \sqrt{\frac{\sum_i (y_i - \hat{y}_i)^2}{n-k-1}}$$

Variable importance (via permutation)

For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. R² is used as scoring metric to determine the importance:

$$importance = score_{full \ model} - \frac{1}{10} \sum_{i=1}^{10} score_i.$$

Please note that the importance might be misleading for strongly correlated explanatory variables.

Variable importance (impurity-based)

The impurity-based variable importance is a regression tree specific feature. The importance

corresponds to the (normalized) total reduction of the used splitting criterion. The sum of the importance of each single explanatory variable is one. Please note that impurity-based variable importance can be misleading for high cardinality features, i.e., features that have many unique values. As alternative the permutation-based variable importance can be considered.

Partial dependence plots (one-way)
The partial dependence plots can be interpreted as the expected change in the response variable along the gradient of the considered explanatory variable. Here, the 'brute' method over the whole range of the respective gradient with a grid resolution of 100 was used.

Output (binary response variable)
Regression information
$OOB\ (out\ of\ bag)\ score = estimation\ of\ the\ generalization\ score\ (mean\ accuracy)$

Regression statistics
$AUC\ ROC = \ area\ under\ the\ receiver\ operating\ characteristic\ curve$

$AP = \sum_{m} (R_m - R_{m-1})P_m\ ,\ \ R_m = recall\ and\ P_m = precision\ at\ m^{th}\ threshold$

$AUC\ PRC = \ area\ under\ the\ precision - recall\ curve$

$LOG - LOSS = \frac{Log\ likelihood}{n}$

Variable importance (via permutation)
For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. AUC ROC is used as scoring metric to determine the importance:

$importance = score_{full\ model} - \frac{1}{10} \sum_{i=1}^{10} score_i.$

Please note that the importance might be misleading for strongly correlated explanatory variables.

Variable importance (impurity-based)
See Output (continuous response variable) for Random Forest.

Analytical plots
See Output for Logistic Regression.

Partial dependence plots (one-way)
See Output (continuous response variable) for Random Forest.

3.3.5 Boosted Regression Trees

Definition

Boosted Regression Trees (BRT) is a tree-based machine learning technique. For evaluating the learning of the BRT model, a loss function is defined. Here, the least squares and deviance are implemented as loss functions for continuous and binary response variables, respectively. During learning, a certain number of trees are sequentially fitted on the negative gradient of the given loss function, with each tree being typically built on a bootstrap data set (Figure 21).

$$X_1 < z_1$$

$$X_2 < z_2 \qquad X_1 < z_3$$

$$X_i - variable$$
$$z_i - threshold$$
$$R_i - result$$

$$X_3 < z_4$$

$$R_1 \qquad R_4 \qquad R_5$$

$$R_2 \qquad R_3$$
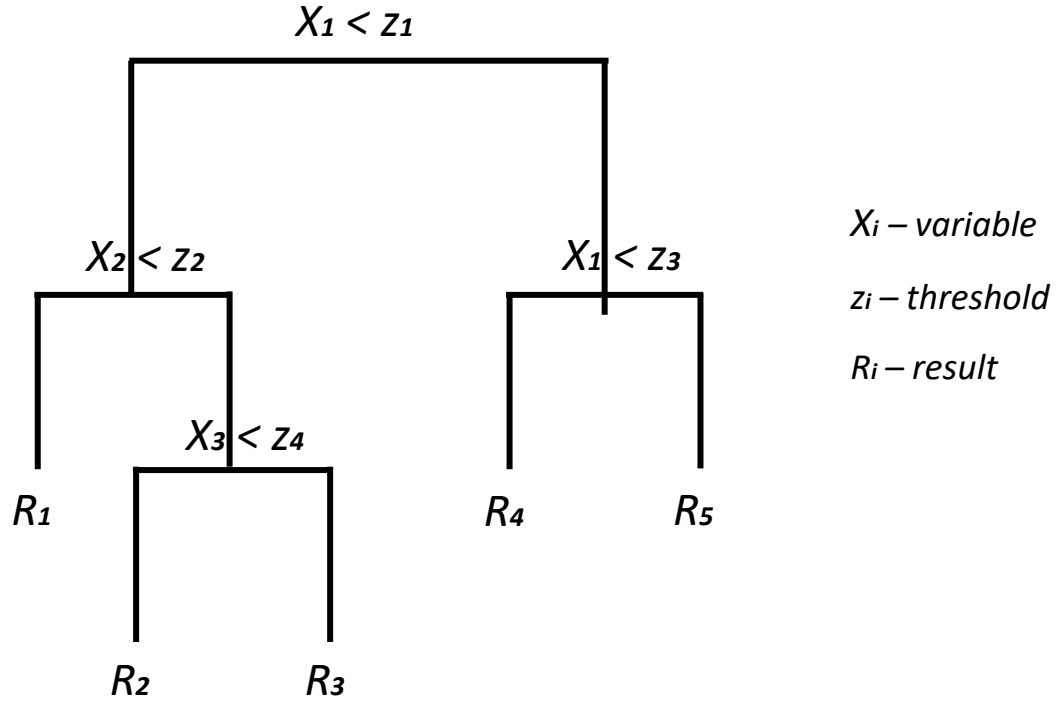
Figure 21 Example of a decision tree $f^t(x)$ based on a bootstrap data set.

The structure of the trees can be defined by the number of nodes or the maximum tree depth and the influence of each tree in the final model can be controlled through a learning rate:

$$f(x) = \sum_{t=1}^{N_t} \lambda f^t(x), \qquad \lambda = learning\ rate, \qquad N_t = number\ of\ trees\ .$$

Hyperparameters settings

In BRT models, the parameters number of trees, learning rate, maximum tree depth and the sample rate for building the single trees can be tuned (Figure 22). Please note that a small learning rate leads to a slow learner, which tends to perform well, and that larger values for the maximum tree depth can imply variable interactions.
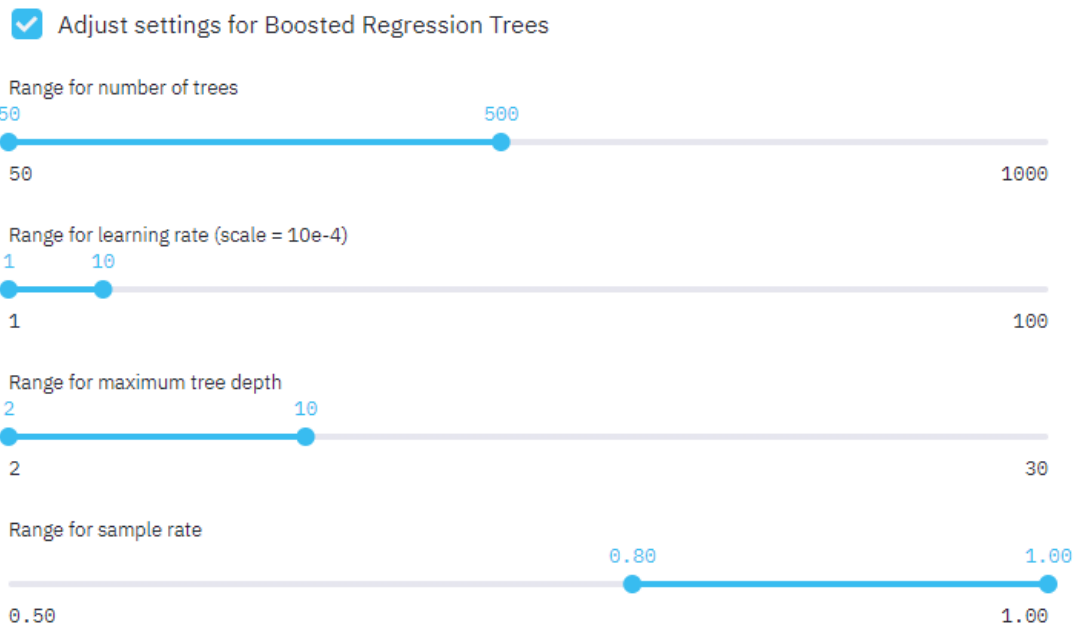
Figure 22 Range options for hyperparameters of Boosted Regression Trees.

For the search methods grid-search and random grid-search, a grid based on the ranges for each hyperparameter must be created. The grid is formed by using the minimum and maximum of the specified range (Figure 22) and a pre-defined step size (Table 2).

Hyperparameter	Default value	Range	Default range	Step size
Number of trees	100	50 – 1000	50 – 500	5
Learning rate	0.1	0.001 – 0.1	0.001 – 0.02	0.001
Maximum tree depth	3	1 – 30	2 – 10	1
Sample rate	1	0.5 – 1.0	0.8 – 1.0	0.05

Table 2 Hyperparameter specifications for Boosted Regression Trees.

Please note that for the search methods Bayes optimization and sequential model-based optimization the definition of step sizes is not necessary as the next hyperparameter combination is automatically proposed within the specified ranges (see 3.3.7 Hyperparameter-tuning).

Output (continuous response variable)

<u>Regression statistics</u>

$$Mean\ squared\ error\ (MSE) = \frac{1}{n}\sum_i (y_i - \hat{y}_i)^2$$

$$Root\ mean\ squared\ error\ (RMSE) = \sqrt{\frac{1}{n}\sum_i (y_i - \hat{y}_i)^2}$$

$$Mean\ absolute\ error\ (MAE) = \frac{1}{n}\sum_{i}|y_i - \hat{y}_i|$$

$$Residual\ standard\ error = \sqrt{\frac{\sum_i(y_i - \hat{y}_i)^2}{n-k-1}}$$

Variable importance (via permutation)

For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. R² is used as scoring metric to determine the importance:

$$importance = score_{full\ model} - \frac{1}{10}\sum_{i=1}^{10}score_i.$$

Please note that the importance might be misleading for strongly correlated explanatory variables.

Variable importance (impurity-based)

The impurity-based variable importance is a regression tree specific feature. The importance corresponds to the (normalized) total reduction of the used splitting criterion. The sum of the importance of each single explanatory variable is one. Please note that impurity-based variable importance can be misleading for high cardinality features, i.e., features that have many unique values. As alternative the permutation-based variable importance can be considered.

Partial dependence plots (one-way)

The partial dependence plots can be interpreted as the expected change in the response variable along the gradient of the considered explanatory variable. Here, the 'brute' method over the whole range of the respective gradient with a grid resolution of 100 was used.

Output (binary response variable)

Regression statistics

$$AUC\ ROC = area\ under\ the\ receiver\ operating\ characteristic\ curve$$

$$AP = \sum_{m}(R_m - R_{m-1})P_m\ ,\ \ R_m = recall\ and\ P_m = precision\ at\ m^{th}\ threshold$$

$$AUC\ PRC = area\ under\ the\ precision - recall\ curve$$

$$LOG - LOSS = \frac{Log\ likelihood}{n}$$

Training score

$$Deviance = 2\ LOG - LOSS$$

Variable importance (via permutation)

For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. AUC ROC is used as scoring metric to determine the

importance:

$$importance = score_{full\ model} - \frac{1}{10} \sum_{i=1}^{10} score_i.$$

Please note that the importance might be misleading for strongly correlated explanatory variables.

Variable importance (impurity-based)

See Output (continuous response variable) for Boosted Regression Trees.

Analytical plots

See Output for Logistic Regression.

Partial dependence plots (one-way)

See Output (continuous response variable) for Boosted Regression Trees.

3.3.6 Artificial Neural Networks

Definition

Artificial Neural Networks (ANNs) is a machine learning technique that resembles the biological neural system, i.e., ANNs include neurons with a specified number of layers that are linked by so-called activation functions (Figure 23).
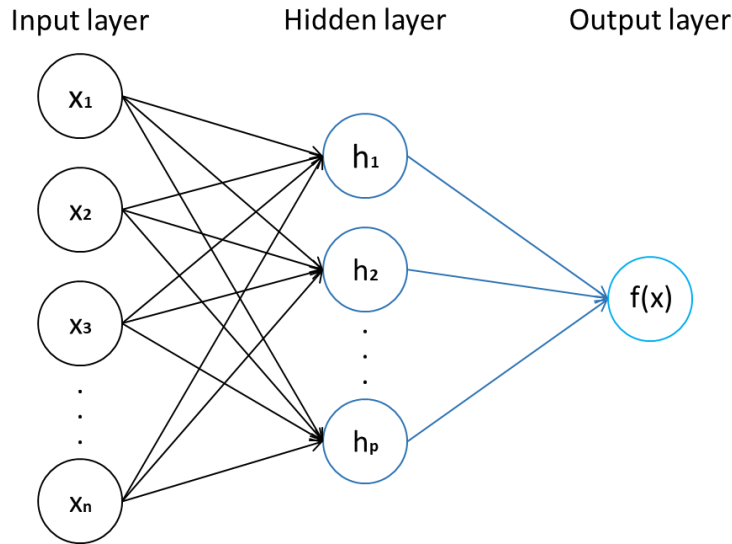
Figure 23 Artificial Neural Network with p neurons in one hidden layer.

The single observations x_i are linearly weighted, adjusted with a bias value and afterwards transformed by the activation function $f_{activation}$. For example, for the first neuron in the hidden layer we get

$$h_1 = f_{activation}(w_1 x_1 + w_2 x_2 + \cdots + w_n x_n + bias),$$

so that h_1 is now the input for the next layer. When all neurons of the last hidden layer have been calculated, the values of these neurons are transformed one last time by the output function f_{output} to get the wanted output/ prediction. For continuous and binary response variables the identity and logistic function are used as output functions, respectively.

The learning of the ANN consists of updating the weights, i.e., the weight optimization, by using a specific solver algorithm, e.g., gradient-based optimization, and several different hyperparameters. For evaluating the learning of the ANN model, the least squares and log-loss are implemented as loss functions for continuous and binary response variables, respectively.

Please note that Artificial Neural Networks are sensitive to different scales of the data. Therefore, the data is re-scaled by standardization, such that the mean is 0 and the variance is 1. In case of testing during hyperparameter-tuning or model validation, the same scaling is applied to the test data set.

Hyperparameters settings

In ANN models, the parameters weight optimization solver, maximum number of iterations, activation function, hidden layer sizes, learning rate and L² regularization can be tuned (Figure 24).

Figure 24 Range options for hyperparameters of Artificial Neural Networks.

Please note that, for the solver 'adam', the maximum number of iterations determines the number of epochs, i.e., it defines how often the data will be seen by the model. Also, the learning rate, which is used to control the weight updates throughout weight optimization, is only used for the solver 'adam'. L² regularization is used to control for the weight size and thus penalizing complex models and controlling overfitting.

For the search methods grid-search and random grid-search, a grid based on the ranges for each hyperparameter must be created. The grid is formed by using the minimum and maximum

of the specified range (Figure 24) and a pre-defined step size (Table 3). For the number of hidden layers and the corresponding nodes all combinations are considered.

Hyperparameter	Default value	Range	Default range	Step size
Solver	adam	adam	adam	-
Maximum iterations	200	10 – 1000	100 – 200	5
Activation function	relu	relu, identity, logistic, tanh	relu	-
Hidden layers	1	1 – 3	1	-
Nodes per hidden layer	100	5 – 500	50 – 100	-
Learning rate	0.001	0.0001 – 0.01	0.0001 – 0.002	0.0005
L² regularization	0.0001	0 – 0.001	0.00001 – 0.0002	0.00005

Table 3 Hyperparameter specifications for Artificial Neural Networks.

Please note that for the search methods Bayes optimization and sequential model-based optimization the definition of step sizes is not necessary as the next hyperparameter combination is automatically proposed within the specified ranges (see 3.3.7 Hyperparameter-tuning).

Output (continuous response variable)

Regression statistics

$$Mean\ squared\ error\ (MSE) = \frac{1}{n} \sum_i (y_i - \hat{y}_i)^2$$

$$Root\ mean\ squared\ error\ (RMSE) = \sqrt{\frac{1}{n} \sum_i (y_i - \hat{y}_i)^2}$$

$$Mean\ absolute\ error\ (MAE) = \frac{1}{n} \sum_i |y_i - \hat{y}_i|$$

$$Residual\ standard\ error = \sqrt{\frac{\sum_i (y_i - \hat{y}_i)^2}{n-k-1}}$$

$$Best\ loss = loss\ from\ loss\ cruve$$

Loss curve

Loss value for each iteration.

Variable importance (via permutation)

For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. R² is used as scoring metric to determine the importance:

$$importance = score_{full\,model} - \frac{1}{10} \sum_{i=1}^{10} score_i.$$

Please note that the importance might be misleading for strongly correlated explanatory variables.

<u>Partial dependence plots (one-way)</u>

The partial dependence plots can be interpreted as the expected change in the response variable along the gradient of the considered explanatory variable. Here, the 'brute' method over the whole range of the respective gradient with a grid resolution of 100 was used.

Output (binary response variable)

<u>Regression statistics</u>

$AUC\,ROC = \;area\,under\,the\,receiver\,operating\,characteristic\,curve$

$$AP = \sum_{m} (R_m - R_{m-1})P_m\;,\;\;R_m = recall\,and\,P_m = precision\,at\,m^{th}\,threshold$$

$AUC\,PRC = \;area\,under\,the\,precision - recall\,curve$

$LOG - LOSS = \frac{\text{Log\,likelihood}}{n}$

$Best\,loss = loss\,from\,loss\,cruve$

<u>Loss curve</u>

Loss value for each iteration.

<u>Variable importance (via permutation)</u>

For the determination of the permutation based variable importance the data set is shuffled 10 times for each explanatory variable. AUC ROC is used as scoring metric to determine the importance:

$$importance = score_{full\,model} - \frac{1}{10} \sum_{i=1}^{10} score_i.$$

Please note that the importance might be misleading for strongly correlated explanatory variables.

<u>Analytical plots</u>

See Output for Logistic Regression.

<u>Partial dependence plots (one-way)</u>

See Output (continuous response variable) for Artificial Neural Networks.

3.3.7 Hyperparameter-tuning

Hyperparameter-tuning is an optional setting. It can be conducted for Random Forest, Boosted Regression Trees and Artificial Neural Networks.

Definition

The parameters that influence the learning of a machine learning method are called hyperparameters. For these hyperparameters, different values are worth considering to get the best model possible. However, finding the specific values is a great challenge. Here, hyperparameter-tuning tries to give a solution to this problem.

The aim of hyperparameter-tuning is to find the best possible set of hyperparameters for a modelling technique across a broad range of possible combinations of values called the parameter space/ grid. The best parameter set from the space/ grid is the one that yields the highest R^2 or AUC for continuous or binary variables, respectively, or the best value for the corresponding objective function. Under hyperparameter-tuning settings you can adjust the ranges for single hyperparameters and thus construct the parameter space/ grid from which the best parameter set will be selected.

Search methods

There are several strategies for finding the best hyperparameter combination across the parameter space/ grid. You can select from the classical grid-search, random grid-search, Bayesian optimization and sequential model-based optimization using decision trees.

Grid-search

Full search across all hyperparameter combinations defined by the parameter grid.

Random grid-search

Search across randomly selected hyperparameter combinations. The selected number of iterations determines how many combinations are considered.

Bayes optimization

Bayesian hyperparameter search using Gaussian process regression to approximate the objective function in order to propose the next hyperparameter combination. The selected number of iterations determines how many combinations are considered.

Sequential model-based optimization

Sequential hyperparameter search using decision trees to approximate the objective function in order to propose the next hyperparameter combination. The selected number of iterations determines how many combinations are considered.

Tuning process

For the hyperparameter-tuning the underlying data set is randomly split into training and test data. The training and test data consist of 80% and 20% of the full data set, respectively. Only the training data is used for tuning the model and finding the best possible hyperparameter combination. For each proposed hyperparameter combination, cross-validation is performed

to determine the performance. The number of folds for the cross-validation can be set under hyperparameter-tuning settings (Figure 25). If the response variable is binary, a stratified cross-validation is conducted. By selecting the number of iterations, you can limit the number of hyperparameter combinations that are considered during the tuning for random grid-search, Bayesian optimization and sequential model-based optimization (Figure 25). Please note that high numbers for iterations will result in searches that take a lot of time. After determining the best model either by using mean % VE (percentage of explained variance, continuous) or AUC (binary) from the cross-validation runs, the model is tested on the unseen test data.

Hyperparameter-tuning settings

Use hyperparameter-tuning

> Yes ▾

> WARNING: Hyperparameter-tuning can take a lot of time!

Hyperparameter-search method

> random grid-search ▾

Select number for n-fold cross-validation

5

2   10

Select number of iterations for search

20

20   1000

Figure 25 Hyperparameter-tuning settings.

Output

The results of the hyperparameter-tuning are shown in tabular form and consist of the final hyperparameter set for each modelling technique and the tuning performance, including the utilized scoring metric (R^2 or AUC), the number of iterations/ considered models, the mean performance across the cross-validation runs and the corresponding standard deviation, and the performance on the test data. Please note that the performance score on the test data is usually smaller than the mean performance from the cross-validation, since the model makes predictions for unseen data.

3.3.8 Model comparison

Different performance metrics for model comparisons are provided for continuous and binary response variables.

Continuous response variable

$$\% \, VE \, (proportion \; of \; variance \; explained) = \; 1 - \frac{\Sigma_i(y_i - \hat{y}_i)^2}{\Sigma_i(y_i - \bar{y})^2}$$

$$Mean \; squared \; error \; (MSE) = \frac{1}{n} \sum_i (y_i - \hat{y}_i)^2$$

$$Root \; mean \; squared \; error \; (RMSE) = \sqrt{\frac{1}{n} \sum_i (y_i - \hat{y}_i)^2}$$

$$Mean \; absolute \; error \; (MAE) = \; \frac{1}{n} \sum_i |y_i - \hat{y}_i|$$

$$MaxErr \; (maximum \; residual \; error) = \max_i(|y_i - \hat{y}_i|)$$

$$EVRS \; (explained \; variance \; regression \; score) = \; 1 - \frac{Var\{y - \hat{y}\}}{Var\{y\}}$$

$$SSR \; (sum \; of \; squared \; residuals) = \sum_i (y_i - \hat{y}_i)^2$$

Binary response variable

Threshold-independent metrics

$$AUC \; ROC = \; area \; under \; the \; receiver \; operating \; characteristic \; curve$$

$$AP = \sum_m (R_m - R_{m-1})P_m \, , \;\; R_m = recall \; and \; P_m = precision \; at \; m^{th} \; threshold$$

$$AUC \; PRC = \; area \; under \; the \; precision - recall \; curve$$

$$LOG - LOSS = \frac{Log \; likelihood}{n}$$

Threshold

$$threshold \; (Yoden \; Index) = \max_{t \in thresholds} \; (sensitivity_t + specificity_t - 1) \, .$$

Threshold-dependent metrics

$$TPR \; (true \; positive \; rate, sensitivity, recall) = \frac{TP}{TP + FN}$$

$$FNR \; (false \; negative \; rate) = \frac{FN}{TP + FN}$$

$$TNR \; (true \; negative \; rate, specificity) = \frac{TN}{TN + FP}$$

$$FPR \; (false \; positive \; rate) = \frac{FP}{TN + FP}$$

$$TSS \; (true \; skill \; statistic) = sensitivity + specificity - 1$$

$$PREC \; (precision) = \frac{TP}{TP + FP}$$

$$F1 \ (F1 \ score) = \frac{2 * precision * recall}{precision + recall}$$

$$KAPPA \ (Cohen's \ Kappa) = \frac{\frac{TP+TN}{n} - \left(\frac{TP+FP}{n} * \frac{TP+FN}{n} + \frac{FN+TN}{n} * \frac{FP+TN}{n} \right)}{1 - \left(\frac{TP+FP}{n} * \frac{TP+FN}{n} + \frac{FN+TN}{n} * \frac{FP+TN}{n} \right)}$$

$$ACC \ (accuracy) = \frac{TP+FN}{n}$$

$$BAL \ ACC \ (balanced \ accuracy) = \frac{TPR+FNR}{2}$$

3.3.9 Model predictions

Model predictions are always provided for the original data (default or uploaded in the sidebar menu) and optionally also for newly uploaded data (Figure 26), i.e., if you want to make predictions based on the calibrated model using the original data. In case of variable transformations in 3.1.4 Data processing, **STATY** automatically checks whether the transformed variables are included in the new data set and conducts the transformation accordingly. Rows with missing values in your new data are also automatically deleted.

Figure 26 Model prediction settings for new data.

3.3.10 Model validation

Model validation is an optional setting. If model validation is used, it will be performed for all selected modelling techniques.

Definition

The aim of model validation is to evaluate a trained model with a test data set and thus test the generalization ability. Prior to the evaluation the model is calibrated with a separate training data set and with final hyperparameters, if hyperparameter-tuning was conducted, or default hyperparameters. Afterwards, the prediction performance of the model for the test data set is determined.

Validation process

For randomly splitting the data into a training and test data set, you can select the fraction for the training data size (Figure 27). The respective modelling technique will be trained with the training data set and afterwards evaluated on the test data set. This process will be repeated according to the number for validation runs (Figure 27). To assure comparability among the selected modelling techniques, the training and test data will be the same throughout the single validation runs.

Validation settings

Use model validation

Yes ▼

Select training data size

0.80

0.50   0.95

Select number for validation runs

10

5   100

Figure 27 Validation settings.

Output

The results of the model validation for the selected modelling techniques are shown in tabular and graphical form containing the main performance measures (mean and standard deviation), boxplots of the residual distribution and the % of the explained variance, variable importance (mean and standard deviation) and main quantiles of the residuals across all validation runs. Please note that values of the performance measures are usually below those obtained for the full model.

Data decomposition

3.3.11 Principal Component Analysis

Definition

In Principal Component Analysis (PCA) the so-called principal components are computed. The principal components are the eigenvectors $e_1, \dots, e_k$ of the $k \times k$ covariance matrix Σ obtained from the standardized data X_s or the correlation matrix of X using the selected variables $x_1, \dots, x_k$ for decomposition. Therefore, prior to PCA, the data X is standardized to X_s to account for different scales. Based on the eigenvectors $e_1, \dots, e_k$, where the entries e_{ij} for $i, j = 1, \dots, k$ are also called loadings, one can identify which variables have the highest loading in a component and thus form groups and explore the data. Each principal component e_i for $i = 1, \dots, k$ accounts for a specific amount of variation of the data, which can be measured by the corresponding eigenvalues λ_i for $i = 1, \dots, k$ of the covariance matrix Σ. The eigenvalues λ_i are sorted from largest to smallest value $\lambda_1 > \cdots > \lambda_k$. By using for example only the first components that account for the largest amount of variation, i.e., the eigenvectors that correspond to the largest eigenvalues, the original data is projected to a lower dimensional subspace via an orthogonal transformation, i.e., the data X_s are multiplied with the eigenvector matrix e. This can be helpful for building predictive machine learning models using data with a lower number of explanatory variables, i.e., a lower dimensionality.

Output

Eigenvalues and explained variance:

$eigenvalue_i = \lambda_i \; of \; the \; covariance \; matrix \; \Sigma$

$explained \; variance \; ratio_i = \dfrac{\lambda_i}{\sum_i \lambda_i}$

$cumulative \; explained \; variance_i = \sum_{j \leq i} explained \; variance \; ratio_j$

Eigenvectors:

$eigenvectors \; e_i \; of \; the \; covariance \; matrix \; \Sigma \; with \; loadings \; per \; variable$

Transformed data:

$transformed \; data = X_s * e$

Transformation is conducted by using all principal components. Please note that columns that are not needed can be simply excluded in further analysis.

3.3.12 Factor Analysis

Definition

Factor Analysis (FA) is a method to reduce dimensionality of multivariate data by describing the covariance relations of the selected variables $x_1, \dots, x_k$ for data decomposition with unobservable or also called latent variables or factors $F_1, \dots, F_m$. In other words, the included variables are explained as linear combinations of these factors plus some error:

$$x_1 - \mu_1 = l_{11}F_1 + \cdots + l_{1m}F_m + \varepsilon_1$$
$$x_2 - \mu_2 = l_{21}F_1 + \cdots + l_{1m}F_m + \varepsilon_2$$
$$\vdots \qquad \vdots \qquad \vdots \qquad \vdots$$
$$x_k - \mu_k = l_{k1}F_1 + \cdots + l_{km}F_m + \varepsilon_k$$

or

$$X - \mu = LF + \varepsilon$$

where

$X = \begin{pmatrix} x_1 \\ \vdots \\ x_k \end{pmatrix}$ is the vector of the observed variables

$\mu = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_k \end{pmatrix}$ is the vector of the population means

$L = \begin{pmatrix} l_{11} & \cdots & l_{1m} \\ \vdots & \ddots & \vdots \\ l_{k1} & \cdots & l_{km} \end{pmatrix}$ is the loading matrix

$F = \begin{pmatrix} F_1 \\ \vdots \\ F_m \end{pmatrix}$ is the vector of the common factors

$\varepsilon = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_m \end{pmatrix}$ is the vector of errors or the specific factors

$m = number\ of\ factors\ (can\ be\ specified, m \ll k)$
$k = number\ of\ variables.$

For FA, the following assumptions must hold:

$$E(F_i) = 0 \ and \ Var(F_i) = 1$$
$$E(\varepsilon_i) = 0 \ and \ Var(\varepsilon_i) = \psi_i, where \ \psi_i \ is \ called \ the \ uniqueness$$
$$Cov(F_i, F_j) = 0 \ for \ i \neq j$$
$$Cov(\varepsilon_i, \varepsilon_j) = 0 \ for \ i \neq j$$
$$Cov(\varepsilon_i, F_j) = 0.$$

Given this model, we obtain the following relationship:

$$\Sigma = \mathrm{LL^T} + \varepsilon, where\ \Sigma\ is\ the\ covariance\ matrix,$$

which means that the variance of an observed variable can be represented by the sum of the respective squared loadings (also called communality) and the uniqueness

$$Var(x_i) = \sum_{j=1}^{m} l_{ij}^2 + \psi_i.$$

Please note that a large communality for a variable implies a better model performance for this specific variable, whereas a large uniqueness implies that the factors do not perform well in accounting for the variance and that consequently the variable might be of lower importance in the factor model. To account for different scales, X is standardized, so that $Var(x_i) = 1$. Moreover, based on the loadings one can label factors similar to PCA by examining variables with high loadings.

Settings

For FA, the number of factors, the fitting method to estimate the parameters of the factor model (Maximum Likelihood, MINRES, Principal Factor) and the rotation method (None, varimax, promax, oblimin, oblimax, quartimin, quartimax, equamax), which is commonly used to improve interpretability of the model, can be selected (Figure 28).

| Number of factors | | | Fitting method |
| 8 | − | + | Maximum Likelihood ▾ |

| Rotation |
| None ▾ |

Figure 28 Default settings for Factor Analysis.

Output

Bartlett's Sphericity test:
$$statistic = -1 * \left(n - 1 - \frac{2k+5}{6}\right) * \ln(\det(\Sigma))$$
$$dof\ (degrees\ of\ freedom) = k\ * \frac{(k-1)}{2}$$
$$p - value = \ corresponding\ to\ statistic$$

The statistic follows a χ^2 distribution; testing whether the correlation matrix is equal to the identity matrix (null hypothesis), i.e., testing whether FA is appropriate

Kaiser-Meyer-Olkin criterion:
$$KMO = overall\ Kaiser - Meyer - Olkin\ criterion$$

55

The Kaiser-Meyer-Olkin criterion lies between 0 and 1, where generally a value < 0.6 implies that a factor analysis for the data is inadequate.

Eigenvalues:

$eigenvalue_i = \ eigenvalue\ of\ the\ correlation\ matrix$

$explained\ variance\ ratio_i = \frac{eigenvalue_i}{\sum_i eigenvalue_i}$

$cumulative\ explained\ variance_i = \sum_{j \leq i} explained\ variance\ ratio_j$

$common\ factor\ eigenvalue_i$
$= eigenvalue\ of\ the\ correlation\ matrix\ with\ communalities\ on\ the\ diagonal$

Explained variance:

$SS\ loadings_i (sum\ of\ squared\ loadings) = \sum_{j=1}^{k} l_{ji}^2$

$explained\ variance\ ratio_i = \frac{SS\ loadings_i}{\sum_i eigenvalue_i}$

$cumulative\ explained\ variance_i = \sum_{j \leq i} explained\ variance\ ratio_j$

Communalities and uniqueness:

$communality_i = \sum_{j=1}^{m} l_{ij}^2$

$uniqueness_i = 1 - communality_i$

Loadings:

Elements of the loading matrix.

Transformed data:

$transformed\ data = \ factor\ matrix$

3.4 Panel data

STATY provides different linear models tailored to panel data like the entity fixed effects, time fixed effects, two-ways fixed effects, random entity effects and pooled model.

You have the option to adjust the covariance matrix for each model and to carry out a model validation. Before starting, you have to specify the corresponding variables for entity and time.

3.4.1 Entity Fixed Effects

Definition

Entity Fixed Effects (EFE) is a linear regression technique that uses several explanatory variables and entity-specific effects to predict a continuous response variable:

$$y_{it} = \beta_1 x_{it1} + \cdots + \beta_k x_{itk} + \alpha_i + \varepsilon_{it}$$

where

$i = entity\ index$

$t = time\ index$

$k = number\ of\ explanatory\ variables$

$y = response\ variable$

$x_j = explanatory\ variables, j = 1 \dots k$

$\beta_j = coefficient\ of\ explanatory\ variable\ x_j$

$\alpha_i = entity\ effect\ capturing\ unobserved\ heterogeneity$

$\varepsilon = idiosyncratic\ error.$

The coefficients are estimated with the within estimator. The estimator applies OLS to the following transformed regression problem:

$$y_{it} - \overline{y_i} = (x_{it} - \overline{x_i})^T \beta + (\varepsilon_{it} - \overline{\varepsilon_i}).$$

For the estimator of EFE to be consistent and efficient, given the observed data and entity effects the following assumptions must be fulfilled:

$(1)\ E(\varepsilon_{it}|x_{it}, \alpha_i) = 0\ (strict\ exogeneity)$

$(2)\ E(\varepsilon_{it}^2|x_{it}) = \sigma_\varepsilon^2 \quad \rightarrow (independent\ and\ identically\ distributed$

$(3)\ E(\varepsilon_{it}\varepsilon_{js}|x_{it}) = 0. \rightarrow \quad error\ over\ i\ and\ t,\ \varepsilon_{it} \sim iid(0, \sigma_\varepsilon^2))$

Moreover, for the EFE model with the within estimator it is allowed that the entity effects are correlated with the observations: $E(\alpha_i|x_{it}) \neq 0$, as the within estimator eliminates α_i. Another possibility to obtain entity effects is by using Least Squares Dummy Variable (LSDV) with dummy variables for the entities.

Settings

The within estimator includes the OLS estimator, for which certain assumptions must hold for getting reliable standard errors and statistical test results. Therefore, you can select either a non-robust (homoscedastic) covariance matrix Σ or robust (heteroskedastic, clustered) covariance matrix Σ (Figure 29).

Select covariance type

homoskedastic	▾

homoskedastic

heteroskedastic
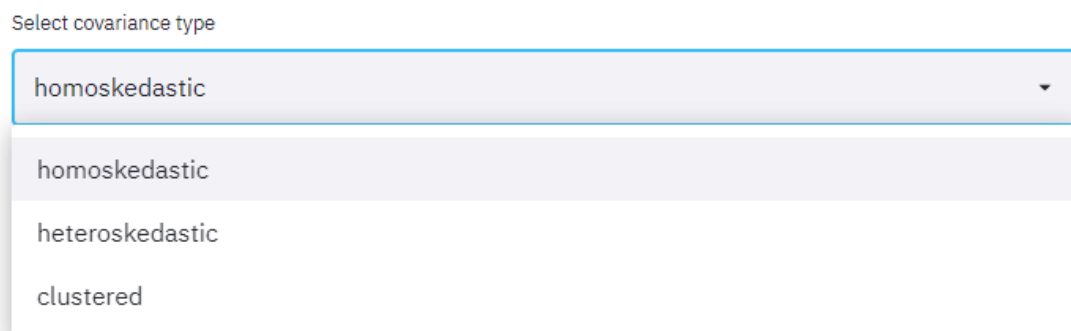
clustered

Figure 29 Covariance types for panel data models.

If 'clustered' is selected, either one- or two-way clustering is possible (Figure 30). If the same variable is used for clustering as in the effects model, the degrees of freedom used in estimating the effects are not counted.

Select cluster type

entity	▾

entity

time

both
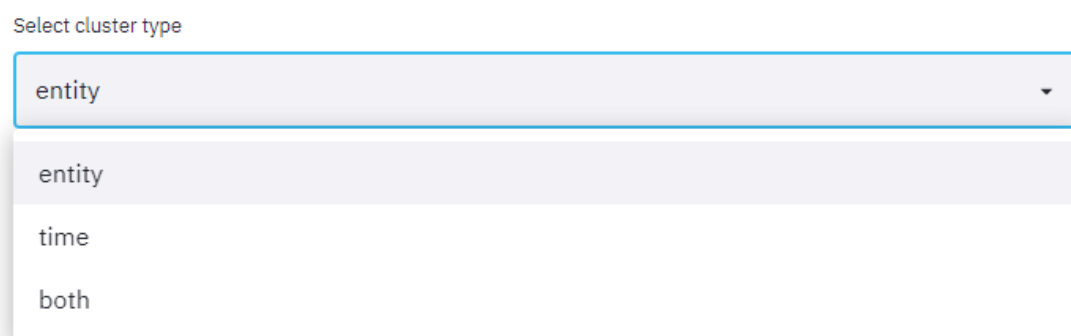
Figure 30 Options for the clustered covariance estimator.

Homoskedastic

$$\Sigma = s^2 \Sigma_{XX}^{-1},$$

$$where \ \Sigma_{XX} = \sum_{i=1}^{N} \sum_{t=1}^{T_i} x_{it}^T x_{it}$$

$$and \ N = number \ of \ entities$$

$$and \ s^2 = (n-k) \sum_{i=1}^{N} \sum_{t=1}^{T_i} \hat{\varepsilon}_{it}^2$$

Heteroskedastic

$$\Sigma = \frac{n}{(n-k)} \Sigma_{XX}^{-1} \hat{S} \Sigma_{XX}^{-1},$$

where $\hat{S} = \sum_{i=1}^{N} \sum_{t=1}^{T_i} \hat{\varepsilon}_{it}^2 \, x_{it}^T x_{it}$

<u>Clustered (one-way)</u>

$\Sigma = \frac{n}{(n-k)} \Sigma_{XX}^{-1} \hat{S}_G \Sigma_{XX}^{-1},$

where $\hat{S}_G = \frac{G}{G-1} * \frac{n-1}{n} \sum_{g=1}^{G} \xi_g^T \xi_g, \quad (G = number \ of \ groups)$

where $\xi_g = \sum_{it \, \epsilon \, G_g} \hat{\varepsilon}_{it}^2 \, x_{it}^T x_{it}$

where $it \, \epsilon \, G_g$ indicates that the observation belongs to group g

<u>Clustered (two-way)</u>

$\hat{S}_G = \hat{S}_{G_1} + \hat{S}_{G_2} - \hat{S}_{G_1 \cap G_2}$

Output

<u>Regression statistics</u>

$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2} = 1 - \frac{SSR}{SST}$

$R^2 \ (between) = Corr^2(\hat{y}_i, \bar{y}_i), \qquad \hat{y}_i = \bar{x}_i^T \hat{\beta}$

$R^2 \ (within) = Corr^2(\hat{y}_{it} - \hat{y}_i, y_{it} - \bar{y}_i), \qquad \hat{y}_{it} = \bar{x}_{it}^T \hat{\beta}$

$R^2 \ (overall) = Corr^2(\hat{y}_{it}, y_{it})$

Log likelihood

$SST = \sum_{i=1}^{N} \sum_{t=1}^{T} (y_{it} - \bar{y}_i)^2$

$SST \ (overall) = \sum_{i=1}^{N} \sum_{t=1}^{T} (y_{it} - \bar{y})^2$

<u>Overall performance (with effects)</u>

see 3.3.8 Model comparison

<u>Coefficients</u>

$Coefficients = \beta \ estimates \ of \ the \ within \ estimator$

$Standard \ error = diagonal \ of \ the \ covariance \ matrix \ \Sigma$

$t - statistic = \frac{coefficient}{standard \ error}$

$p - value = \ corresponding \ to \ t - statistic$

$lower \ 95\% \ confidence \ \approx coefficient - 1.96 * standard \ error$

$upper \ 95\% \ confidence \ \approx coefficien + 1.96 * standard \ error$

<u>Entity effects/ Time effects/ Combined effects</u>

intercept for each entity, time period and their combinations

<u>F-tests and Hausman-test</u>

F-test (non-robust):

$F = \frac{\frac{SSR_R - SSR_u}{k}}{\frac{SSR_u}{df_u}}$ where SSR_R is the restricted sum of squares from the model where the

coefficients on all exogenous variables is zero, excluding a constant if one was included; SSR_u is the unrestricted residual sum of squares; k is the number of non-constant regressors in the model; df_u is the residual degree of freedom in the unrestricted model

F-test (robust):

$W = \hat{\beta}^T \Sigma^{-1} \hat{\beta}$ where W is the Wald statistic using the estimated parameter covariance and thus inheriting the robustness from the selected covariance estimator; $\hat{\beta}$ as coefficient vector does not include the model constant; Σ is the estimated covariance matrix; W is additionally divided by the number of restrictions and consequently following the F-distribution

F-test (poolability):

$F = \frac{\frac{SSR_{pool} - SSR_{effect}}{df_{pool} - df_{effect}}}{\frac{SSR_{effect}}{df_{effect}}}$ where SSR_{pool} is the residual sum of squares from a no-effect (pooled)

model; SSR_{effect} is the residual sum of squares from a model with effects; df_{pool} is the residual degree of freedom in the pooled regression; df_{effect} is the residual degree of freedom from the model with effects; the test statistic follows an F_{k,df_u} distribution where $k = df_{pool} - df_{effect}$; testing whether a pooled model is more appropriate (null hypothesis); please note that the underlying test is only correct under the assumption of homoskedasticity

Hausman-test:

$Wu - Hausman\ statistic = (\hat{\beta}_{FE} - \hat{\beta}_{RE})^T (Var(\hat{\beta}_{FE}) - Var(\hat{\beta}_{RE}))^{-1} (\hat{\beta}_{FE} - \hat{\beta}_{RE})$ where $\hat{\beta}_{FE}$ are the coefficients of the fixed effects model; $\hat{\beta}_{RE}$ are the coefficients of the random effects model; testing whether both models are consistent (null hypothesis, RE model is preferred because of higher efficiency); the test statistic follows a $\chi^2(k)$ distribution; please note that the Hausman-test is only conducted for EFE and a homoskedastic covariance type.

3.4.2 Time Fixed Effects

Definition

Time Fixed Effects (TFE) is a linear regression technique that uses several explanatory variables and time-specific effects to predict a continuous response variable:

$$y_{it} = \beta_1 x_{it1} + \cdots + \beta_k x_{itk} + \gamma_t + \varepsilon_{it}$$

where

$i = entity\ index$

$t = time\ index$

$k = number\ of\ explanatory\ variables$

$y = response\ variable$

$x_j = explanatory\ variables, j = 1 \ldots k$

$\beta_j = coefficient\ of\ explanatory\ variable\ x_j$

$\gamma_t = time\ effect$

$\varepsilon = idiosyncratic\ error.$

TFE uses the same assumptions and estimator as EFE. Please see 3.4.1 Entity Fixed Effects for details.

Settings

See 3.4.1 Entity Fixed Effects for details.

Output

Please see 3.4.1 Entity Fixed Effects for details.

Regression statistics

$$SST = \sum_{i=1}^{N} \sum_{t=1}^{T} (y_{it} - \bar{y}_t)^2$$

3.4.3 Two-ways Fixed Effects

Definition

Two-ways Fixed Effects (TWFE) is a linear regression technique that uses several explanatory variables and entity- as well as time-specific effects to predict a continuous response variable:

$$y_{it} = \beta_1 x_{it1} + \cdots + \beta_k x_{itk} + \alpha_i + \gamma_t + \varepsilon_{it}$$

where

$i = entity\ index$

$t = time\ index$

$k = number\ of\ explanatory\ variables$

$y = response\ variable$

$x_j = explanatory\ variables, j = 1 \dots k$

$\beta_j = coefficient\ of\ explanatory\ variable\ x_j$

$\alpha_i = entity\ effect$

$\gamma_t = time\ effect$

$\varepsilon = idiosyncratic\ error.$

TWFE uses the same assumptions and estimator as EFE and TFE. Please see 3.4.1 Entity Fixed Effects for details.

Settings

See 3.4.1 Entity Fixed Effects for details.

Output

Please see 3.4.1 Entity Fixed Effects for details.

<u>Regression statistics</u>

$$SST = \sum_{i=1}^{N} \sum_{t=1}^{T} (y_{it} - \bar{y}_i - \bar{y}_t)^2$$

3.4.4 Random Effects

Definition

Random Effects (RE) is a linear regression technique that uses several explanatory variables and random entity-specific effects to predict a continuous response variable:

$$y_{it} = \beta_1 x_{it1} + \cdots + \beta_k x_{itk} + \alpha_i + \varepsilon_{it}$$

where

$i = entity\ index$

$t = time\ index$

$k = number\ of\ explanatory\ variables$

$y = response\ variable$

$x_j = explanatory\ variables, j = 1 \ldots k$

$\beta_j = coefficient\ of\ explanatory\ variable\ x_j$

$\alpha_i = entity\ effect$

$\varepsilon = idiosyncratic\ error.$

The coefficients are estimated with the quasi-demeaned estimator. The estimator applies OLS to the following equation:

$$y_{it} - \theta_i \overline{y_i} = (x_{it} - \theta_i \overline{x_i})^T \beta + (1 - \theta_i)\alpha_i + (\varepsilon_{it} - \theta_i \overline{\varepsilon_i})$$

where θ_i is the parameter for the demeaning, which is estimated by running a specific prior regression (using the between estimator). Note, that $\theta_i = 0$ leads to the pooled model and $\theta_i = 1$ to the within estimation.

Similar assumptions as for the Entity Fixed Effects (EFE) model must be fulfilled:

(1) $E(\varepsilon_{it}|x_{it}, \alpha_i) = 0\ (strict\ exogeneity)$
(2) $E(\varepsilon_{it}^2|x_{it}) = \sigma_\varepsilon^2 \quad \rightarrow (independent\ and\ identically\ distributed$
(3) $E(\varepsilon_{it}\varepsilon_{js}|x_{it}) = 0. \rightarrow \quad error\ over\ i\ and\ t, \varepsilon_{it} \sim iid(0, \sigma_\varepsilon^2))$

In contrast to the EFE model with the within estimator where it is allowed that the entity effects are correlated with the observations: $E(\alpha_i|x_{it}) \neq 0$, this correlation is not allowed in the RE estimator: $E(\alpha_i|x_{it}) = 0$. The RE model assumes an independent and identically distributed effect: $\alpha_i \sim iid(0, \sigma_\alpha^2)$.

Settings

Please see 3.4.1 Entity Fixed Effects for details.

Output

Please also see 3.4.1 Entity Fixed Effects for details.

Regression statistics

$$SST = \sum_{i=1}^{N} \sum_{t=1}^{T} (\tilde{y}_{it} - \bar{\tilde{y}})^2, \qquad \tilde{y}_{it} = (y_{it} - \theta_i \bar{y}_i)$$

Variance decomposition

$$\hat{\sigma}_\varepsilon^2 (idiosyncratic\ error) = \frac{\sum_{i=1}^{N} \sum_{t=1}^{T} \hat{\varepsilon}_{it}^2}{\sum_{i=1}^{N} T_i - N - k + 1}$$

$\hat{\sigma}_\alpha^2 (individual\ error) = \max\{0, \frac{SSR_b}{N-k} - \frac{\hat{\sigma}_\varepsilon^2}{\bar{T}}\}$ where SSR_b is the residual sum of squares of the between estimator and $\bar{T} = \frac{n}{\sum_{i=1}^{n} T_i^{-1}}$

$$\theta_i = 1 - \sqrt{\frac{\hat{\sigma}_\varepsilon^2}{T_i \hat{\sigma}_\alpha^2 + \hat{\sigma}_\varepsilon^2}}$$

3.4.5 Pooled

See 3.3.1 Multiple Linear Regression.

3.4.6 Model predictions

See 3.3.9 Model predictions.

Please note that when models with effects are used and predictions should be made for new data, effects are only included in the predictions if the effect of the corresponding entity or time period could be identified by the calibrated model with the original data. Consequently, if no effects are available, the prediction is solely based on the selected explanatory variables.

3.4.7 Model validation

See 3.3.10 Model validation.

3.5 Time series data

Whenever the ordering of the observed data in time is important to understand the underlying processes, the use of time series methods is appropriate. The time series itself represents thus a collection of data observed at multiple (consecutive) points in time. Analysing the nature of temporal variations and correlations is also useful for forecasting future outcomes. Characterising and forecasting time series are usually done by specifying mathematical models for the observed variations in time-, frequency- or the time-frequency domain. Time-domain methods such as the autocorrelation function operate in the same temporal space as the observed data. Frequency-domain analyses are focussed at representing observed data in terms of contributions of different drivers acting across different time scales or frequencies.

The time series data module of **STATY** splits the modelling process into three phases: 1) diagnosis plots and tests (i.e., autocorrelation and partial autocorrelation plots, fixed- and moving window statistics checks etc.), 2) time series detrending and seasonal adjustment (i.e., basic time series decomposition), and 3) model specification (i.e., specification of the mathematical model and the corresponding model parameters).

3.5.1 Diagnosis plots and tests

Time series pattern

One of the key ideas behind time series forecasting is that the past and future values will be statistically similar. The latter implies an assumption that the probability distribution of the studied time series does not change over time (*stationarity*). Stationarity can be defined in precise mathematical terms, but in practice one usually looks if at least the mean, variance and the autocorrelation function do not change through time (weak stationarity). Furthermore, the importance of stationarity lies in the fact that most time series analysis and modelling methods assume stationarity of the data. However, stationarity as a time series property is rather an exception, prior to any kind one modelling one needs to inspect the temporal variability of the key time series statistics.
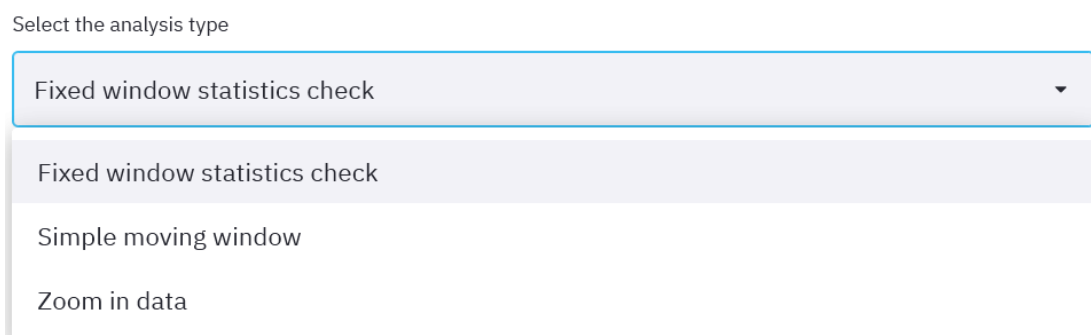

Figure 31 Basic time series screening.

The basic time series screening of **STATY** includes *fixed window* and *moving window* based calculation of the key time-series statistics (mean, variance and standard deviation). Specifically, the option "Fixed window statistics check" enables calculation of the key statistics over a fixed time interval, "Simple moving window" over a moving window with a user specified width, while the option "Zoom in data" enables the user to zoom in and out of time series figure towards discovering potentially interesting features (see Figure 31). Recall that for a time series $y_t, t = 1, ..., n,$ the general expression for a time series moving average is $m_t = (y_t + y_{t-1} + \cdots + y_{t-w+1})/w$, where w is the window width. Therefore, plotting the data paired with the window-based estimates of time series statistics provide the basic "diagnostics" of the stationarity issue and of the presence of trends, cycles, random-walking, and other non-stationary behaviour. We note that the overall time series mean, that is the average of all past observations, is only then a useful forecasting estimate when there are no trends in time series.

Autocorrelation and partial autocorrelation plots

As outlined above, in the process of time series diagnostics, one also looks at the autocorrelation function. Namely, in time series data the values are typically correlated due to effects of external driving mechanisms that are acting and interacting at different temporal scales. The autocorrelation function (ACF) summarizes the correlation of a time series $y_t, t = 1, ..., n,$ with its own lagged values. As such, the lag k ACF is defined as

$$r_k = \frac{cov(y_t, y_{t-k})}{\sqrt{var(y_t)var(y_{t-k})}}$$

The lag k partial autocorrelation function (PACF) describes the conditional correlation between y_t and y_{t-k}, conditional on $y_{t-k+1} ... y_{t-1}$ - the set of observations between the time points t and $t - k$.

The ACF plot is useful for as well non-stationarity detection, as identification of the appropriate Box-Jenkins model. For example, the ACF with very slow decay often suggests non-stationarity. Furthermore, the simplest Box-Jenkins model, the first-order autoregression $AR(1)$ model, usually has a sample ACF characterized with an exponential decay. If the ACF has one or more peaks while the rest are approximately zero autocorrelations, then the moving average Box-Jenkins model (MA) of the order identified by looking where the ACF becomes zero is a good starting point. A mixed autoregressive and moving average ($ARMA$) model is appropriate in cases when a decay of the ACF after a few lags is observed. If the ACF suggest approximately zero autocorrelations at all lags, then the data are most likely random. In contrast, when the ACF does not decay to zero then one should look for options on how to deal with non-stationarity in time series.

The PACF is mainly useful in the context of Box-Jenkins autoregression model diagnostics as the p-order autoregression $AR(p)$ process has a PACF that is zero after the lag p.

Augmented Dickey Fuller Test

While the Dickey-Fuller test focusses on the $AR(1)$ model and test the hypothesis that the series is non-stationary and contains a (stochastic) trend, the augmented Dickey-Fuller (ADF) is a one sided test that focusses on $AR(p)$ Box-Jenkins models. Under the null hypothesis the time series has a stochastic trend, while under the alternative hypothesis, the series is stationary:

$$H_0: \delta = 0 \ \ vs. \ \ H_1: \delta < 0$$

$$\Delta y_t = \beta_0 + \delta y_{t-1} + \gamma_1 \Delta y_{t-1} + \cdots + \gamma_p \Delta y_{t-p} + e_t.$$

3.5.2 Differencing, detrending and seasonal adjustment

The main features of many time series are trends and seasonal variations. As discussed in the previous section, if present, such forms of non-stationarity should have been detected within the "Diagnosis plots and tests" of the time series module. Consequently, the focus of the section "Differencing, detrending and seasonal adjustment" is on handling non-stationarity through time series differencing and on time series decomposition into trend and/or seasonal component.
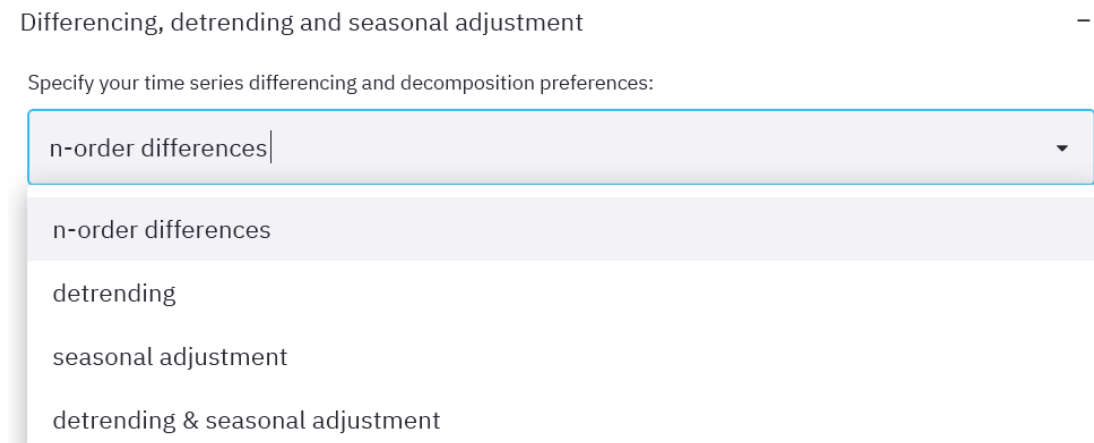

Figure 32 Differencing, detrending and seasonal adjustment menu.

Based on the user's method selection (Figure 32), the corresponding time series, ACF, PACF and the ADF test results are calculated (see the pervious section for more details on the methods). Specifically, if the selected option is "n-order differences", then for all time series differences of the order $\leq n$ the corresponding series, ACF, PACF and the ADF test results are directly provided. Note that the first-order difference of a time series $y_t, t = 1, \ldots, n$ is simply the series of changes from one time period to the next one as $y_t - y_{t-1}$. The corresponding ADF test result provides an orientation regarding the question on whether or not higher order differencing is needed towards meeting the stationarity condition.

As for the time-series decomposition methods (detrending and seasonal adjustment), the output includes the corresponding time series components and the ADF test results, while the ACF and PACF can be optionally added by selecting the corresponding check box.

Derivation of trend and seasonal component is based on the naive additive decomposition as implemented in the statsmodels function *tsa.seasonal.seasonal_decompose*:

$$y_t = T_t + S_t + e_t$$

where T_t and S_t are the trend and the seasonal component, respectively, while e_t denotes the residual term, a series of random variables with zero mean.

Model Specification

As outlined in the above sections, the process of time series <u>model</u> identification is a multi-step journey that starts with stationarity diagnostics (e.g., by calculating moving average of key statistics, ACF plot, ADF test etc.), proceeds to detecting trends and seasonality and calls for differencing or transforming to achieve stationarity. After the questions of stationarity and seasonality have been addressed, one can proceed to identifying the model order using e.g., the ACF and the PACF properties of the pre-processed series.

3.5.3 Moving Average (MA)

A moving average process of the order q, $MA(q)$, is a linear combination of white noise terms and is defined by:

$$y_t = \mu + \varepsilon_t + \beta_1 \varepsilon_{t-1} + \cdots + \beta_q \varepsilon_{t-q}$$

where μ is the mean of the series, $\beta_1, \ldots, \beta_q$ are the parameters of the model and $\varepsilon_t, \varepsilon_{t-1}, \ldots, \varepsilon_{t-q}$ are white noise terms. As such, the MA model is a linear regression of the series as a dependent variable and the random shocks of one or more prior values of the series as repressors. The random shocks are commonly assumed to come from the normal distribution.

One of the approaches to derive the order q of the MA model is by looking at the ACF: if the ACF has one or more peaks while the rest are approximately zero autocorrelations, then the MA order q corresponds to the *lag* just before the ACF becomes zero.

3.5.4 Autoregressive model (AR)

An autoregressive process of order p, $AR(p)$, is defined by:

$$y_t = \alpha_0 + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_p y_{t-p} + \varepsilon_t$$

where α_0 is a constant, $\alpha_1, \ldots, \alpha_p$ are the model paramaters and ε_t is white noise. As such, an autoregressive model is simply a linear regression of the time series against one or more prior values of the series.

The simplest Box-Jenkins model, the first-order autoregression $AR(1)$ model, usually has a sample ACF characterized with an exponential decay.

3.5.5 Autoregressive Moving Average (ARMA)

The Box-Jenkins $ARMA(p, q)$ model is a combination of the auroregressive $AR(p)$ model and the moving average $MA(q)$ model:

$$y_t = \alpha_0 + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_p y_{t-p} + \varepsilon_t + \beta_1 \varepsilon_{t-1} + \cdots + \beta_q \varepsilon_{t-q}$$

where α_0 is a constant, $\alpha_1, \dots, \alpha_p$ are the AR model paramaters, $\beta_1, \dots, \beta_q$ are the parameters of the MA model, $\varepsilon_t, \varepsilon_{t-1}, \dots, \varepsilon_{t-q}$ are white noise terms, p is the order of the AR model and q is the order of the MA model.

There are a couple of points that should be noted about the $ARMA(p, q)$ models:

a) The $ARMA(p, q)$ model assumes that the time series is stationary.

b) The $AR(p)$ and $MA(q)$ models are the special cases of the $ARMA(p, q)$ model. As such the software packages usually assume even more complex model setups, implying the need for specification of additional parameters, even when one simply wants to fit the $AR(1)$ process. We tried to simplify the fitting procedure as much as possible, but this might be at the cost of lower parameter efficiency for simpler models. For example, the $ARMA(p, q)$ model is usually more parameter efficient (requires less parameters), than the $AR(p)$ and $MA(q)$ models.

Only necessary terms should be included in the model – please go through the diagnostic part before you start fitting the model.

3.5.6 Non-seasonal Autoregressive Integrated Moving Average (non-seasonal ARIMA)

One of the common techniques to transform a non-stationary time series to stationary is to use time series differencing. As outlined above, the first-order difference of a time series $y_t, t = 1, \dots, n$ is simply the series of changes from one time period to another $(y_t - y_{t-1})$.

Obviously, the first-order differences series will contain one observation less than the initial time series. We note that lag-1 differencing is usually applied to remove the trend component from time series. In some cases, first-order differences will not result in stationarity, so one can try higher order differencing. The differencing order that leads to stationarity (see ADF test after n-order differencing) is usually denoted d. As the differenced series needs to be transformed or 'integrated' to recover the original series, the underlying stochastic process is called *autoregressive integrated moving average* $ARIMA(p, d, q)$, where p is the order of the AR process, q is the order of the MA process and d is the differencing order. Moreover, a time series follows an $ARIMA(p, d, q)$ process, if the dth differences follow an $ARMA(p, q)$ process. A quite brief formulation of the $ARIMA(p, d, q)$ model is:

$$\theta_p(B)(1 - B)^d y_t = \phi_q(B)\varepsilon_t$$

Where θ_p and ϕ_q are polynomials of order p and q, respectively, B is the backward shift operator and ε_t is white noise. We note that the series is 'integrated' of order d, if the dth differences is white noise ε_t. Because $\nabla^d \equiv (1 - B)^d$, a series is 'integrated' of order d if $(1 - B)^d y_t = \varepsilon_t$.

3.5.7 Seasonal Autoregressive Integrated Moving Average (seasonal ARIMA)

The $ARIMA(p, d, q)$ process can be extended to include seasonal terms, which is referred to as the seasonal integrated moving average model seasonal $ARIMA(p, d, q)(P, D, Q)_s$. The brief formulation of the model is using the backword shift operator B (see section above) can be found in any good textbook on time series modelling. Obviously, seasonal $ARIMA$ models can have a large number of parameters (see Figure 33) suggesting that the best fitting model should be selected using an appropriate statistical criterion such as the AIC or the BIC.
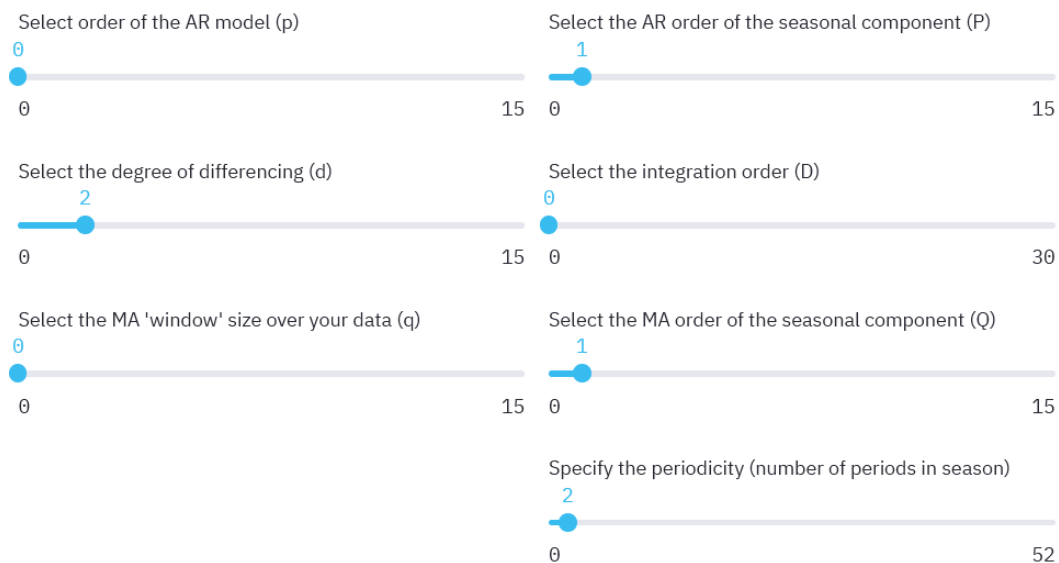
Select order of the AR model (p)
0
0   15

Select the AR order of the seasonal component (P)
1
0   15

Select the degree of differencing (d)
2
0   15

Select the integration order (D)
0
0   30

Select the MA 'window' size over your data (q)
0
0   15

Select the MA order of the seasonal component (Q)
1
0   15

Specify the periodicity (number of periods in season)
2
0   52

Figure 33 Parameter settings for the seasonal ARIMA model.

The parameters of the seasonal $ARIMA(p, d, q)(P, D, Q)_s$ are as follows:

- p is the order of the autoregressive AR model (i.e., the number of lag observations)
- d is the degree of differencing.
- q is he order of the moving average MA model or the "window" size over your data
- P is the order of the seasonal component for the AR model
- D is the integration order of the seasonal process
- Q is the order of the seasonal component of the MA model
- The number of periods in season s is the parameter that reassembles the number of observations per seasonal cycle and must be known before the model run! For example, s is 1 for <u>non-seasonal data</u>, 4 for quarterly, 7 for daily, 12 for monthly, 52 for weekly etc.

Modelling remarks

Within **STATY** manual and the automatic model calibration are possible. We used statsmodels SARIMAX for the manual model calibration while the automatic model calibration is done using the library pmdarima. In order to find the best model, pmdarima optimizes the model parameters using the Akaike Information Criterion ('AIC') as a default value, but this can be overridden by selecting any of the following options: 'BIC', 'HQIC' and 'OOB' (i.e., Bayesian Information Criterion, Hannan-Quinn Information Criterion, or "out of bag"–for validation scoring–respectively).

The option 'use model validation' provide the opportunity to split the data into the calibration and validation sets and test the model performance on the validation data set whose size is specified by the user. The option 'use model for forecast' enables the model use for forecast following the user specification for the number of forecast steps.

3.6 Web scraping and text data

With web scraping and text data you can perform basic NLP text analysis, get automatic summaries of web pages and have access to stock prices (Figure 34).

Figure 34 Analysis options for web scraping and text data.

3.6.1 Text analysis

Based on provided links to web pages or text inputs you can create calculate basic NLP metrics, visualise text statistics such as word frequency, bigram frequency, word similarity and create word clouds with a selectable main colour (Figure 35). Words that you would like to exclude can be selected prior to creating the word cloud (stop words). In addition, sentences and n-grams with a user-specified search words can be extracted from the text.

Figure 35 Options for text analysis.

3.6.2 Web-Page summary

The web page summary provides a small web page preview of the inserted link and an automatic web page summary consisting of five sentences (Figure 36). The summary is created by using natural language processing and a neural network language model.

What what web page should I summarize in five sentences for you?

https://en.wikipedia.org/wiki/Data_mining

Press to start the data processing...

Figure 36 Web-Page summary input area.

3.6.3 Stock data analysis

Stock data analysis uses stock data from yahoo finance. You can insert any stock ticker symbol to get price information for specific time periods and select further stocks for comparison (Figure 37).

Figure 37 Input for stock data analysis.

3.7 Geospatial data

Geospatial data can be graphically displayed with **STATY**. For the correct identification, a country name or country code has to be provided. If a time variable is included, the temporal development of the selected variable for each geographical region can be animated (Figure 38).

Figure 38 Settings for geospatial data visualization.

Additionally, histograms and boxplots for the variable of interest are created.

4 Default data

4.1 Uni- and bivariate data

Data source

The data come from the World Bank's study on financial intermediation and growth.

Citation

Levine, Ross; Loayza, Norman; Beck, Thorsten.1999. Financial intermediation and growth:
Causality and causes (English). Policy, Research working paper; no. WPS 2059 Washington,
D.C.: World Bank Group.

Variables in the dataset

UrbanPopulation	Share of urban population
PrivateSaving	Private saving rate as the ratio of gross private savings and GPDI
logGDPI	Log of real per capita GPDI
GrowtRate	Growth rate of real GPDI
GovermentSaving	Government saving as share of real GDP
LogTermsTrade	Log of terms of trade
OlderThan65	Share of population over 65 in total population
Under15	Share of population under 15 in total population
CommercialCentralBank	Commercial-Central Bank, period average
LiquidLiabilities	Liquid Liabilities, period average
PrivateCredit	Private Credit, period average
BankCredit	Bank credit, period average
LegalOrigin	English, French, German or Scandinavian

4.2 Multivariate data

<u>Data source</u>

The data come from the Gallup World Poll surveys from 2018 to 2020. For more details see the World Happiness Report 2021.

<u>Citation</u>

Helliwell, John F., Richard Layard, Jeffrey Sachs, and Jan-Emmanuel De Neve, eds. 2021. World Happiness Report 2021. New York: Sustainable Development Solutions Network.

<u>Variables in the dataset</u>

Country	country name
Year	year ranging from 2005 to 2020
Ladder	happiness score or subjective well-being with the best possible life being a 10, and the worst possible life being a 0
Log GDP per capita	in purchasing power parity at constant 2017 international dollar prices
Social support	the national average of the binary responses (either 0 or 1) to the question regarding relatives or friends to count on
Healthy life expectancy at birth	based on the data extracted from the World Health Organization's Global Health Observatory data repository
Freedom to make life choices	national average of responses to the corresponding question
Generosity	residual of regressing national average of response to the question regarding money donations in the past month on GDP per capita
Perceptions of corruption	the national average of the survey responses to the corresponding question
Positive affect	the average of three positive affect measures (happiness, laugh and enjoyment)
Negative affect (worry, sadness and anger)	the average of three negative affect measures (worry, sadness and anger)

4.3 Panel data

<u>Data source</u>

This is the original 11-firm data set from Grunfeld's Ph.D. thesis (*Grunfeld, 1958, The Determinants of Corporate Investment, Department of Economics, University of Chicago*). For more details see online complements for the article The Grunfeld Data at 50.

<u>Citation</u>

Kleiber C, Zeileis A (2010). "The Grunfeld Data at 50," German Economic Review, 11(4), 404-417. doi:10.1111/j.1468-0475.2010.00513.x

<u>Variables in the dataset</u>

invest	Gross investment, defined as additions to plant and equipment plus maintenance and repairs in millions of dollars deflated by the implicit price deflator of producers' durable equipment (base 1947).
value	Market value of the firm, defined as the price of common shares at December 31 (or, for WH, IBM and CH, the average price of December 31 and January 31 of the following year) times the number of common shares outstanding plus price of preferred shares at December 31 (or average price of December 31 and January 31 of the following year) times number of preferred shares plus total book value of debt at December 31 in millions of dollars deflated by the implicit GNP price deflator (base 1947).
capital	Stock of plant and equipment, defined as the accumulated sum of net additions to plant and equipment deflated by the implicit price deflator for producers' durable equipment (base 1947) minus depreciation allowance deflated by depreciation expense deflator (10 years moving average of wholesale price index of metals and metal products, base1947).
firm	General Motors (GM), US Steel (US), General Electric (GE), Chrysler (CH), Atlantic Refining (AR), IBM, Union Oil (UO), Westinghouse (WH), Goodyear (GY), Diamond Match (DM), American Steel (AS).
year	Year ranging from 1935 to 1954.

4.4 Time series data

<u>Data source</u>

The data come from Box & Jenkins (1970), but we use the version that is integrated in the R package 'astsa' which is a companion to the book 'Time Series Analysis and Its Applications' by Shumway & Stoffer's (2017) .

<u>Citation</u>

Box, G.E.P. and G.M. Jenkins (1970). Time Series Analysis, Forecasting, and Control. Oakland, CA: Holden-Day

Shumway, R.H, and D.S. Stoffer (2017). Time Series Analysis and Its Applications: With R Examples. New York: Springer

<u>Variables in the dataset</u>

Air passengers	The monthly totals of international airline passengers
Date	Month ranging from January 1949 to December 1960

4.5 Geospatial data

See 4.2 Multivariate data.

5 Contact

Prof. Dr. Danijela Markovic-Bredthauer
Quantitative Methods
Osnabrück University of Applied Sciences
Caprivistr. 30 A, 49076 Osnabrück, Germany

Dr. Oskar Kärcher
Quantitative Methods
Osnabrück University of Applied Sciences

Mail: staty@quant-works.de

6 Disclaimer

The developers assume no responsibility or liability for any errors or omissions in the content of **STATY**. The information contained in **STATY** is provided on an "as is" basis with no guarantees of completeness, accuracy, usefulness, or timeliness.