

Requirements for the Module statistics_lib.ordered_functions

Conventions

Requirements listed in this document are constructed according to the following structure:

Requirement ID: REQ-UVW-XYZ

Title: Title / name of the requirement

Description: Description / definition of the requirement

Verification Method: I / A / T / D

The requirement ID starts with the fixed prefix 'REQ'. The prefix is followed by 3 letters abbreviation (in here 'UVW'), which defines the requirement type - e.g. 'FUN' for a functional and capability requirement, 'AWM' for an alarm, warnings and operator messages, etc. The last part of the ID is a 3-digits *hexadecimal* number (0..9|A..F), with the first digit identifying the module, the second digit identifying a class / function, and the last digit - the requirement ordering number for this object. E.g. 'REQ-FUN-112'. Each requirement type has its own counter, thus 'REQ-FUN-112' and 'REQ-AWN-112' requirements are different entities, but they refer to the same object (class or function) within the same module.

The verification method for a requirement is given by a single letter according to the table below:

Term	Definition
Inspection (I)	Control or visual verification
Analysis (A)	Verification based upon analytical evidences
Test (T)	Verification of quantitative characteristics with quantitative measurement
Demonstration (D)	Verification of operational characteristics without quantitative measurement

Functional and capability requirements

Requirement ID: REQ-FUN-200

Title: Functionality implemented by the module (scope)

Description: The module should implement a number of functions to calculate the statistical properties related to the shape of the distribution(s) of 1D data and rank correlation coefficients of 2D data set:

- Minimum value in the sample
- Maximum value in the sample
- Mode(s) of the sample distribution
- Median value of the sample distribution
- First quartile of the sample distribution
- Third quartile of the sample distribution

- An arbitrary N/M quantile ($N < M$) of the sample distribution
- The histogram of the sample distribution with adjustable width of the bins
- Spearman ρ rank correlation coefficient of 2D data set
- Kendall τ rank correlation coefficient of 2D data set

Verification Method: A

Requirement ID: REQ-FUN-201**Title:** Input and output data format

Description: The 1D statistics functions should accept the input data set as any flat sequence (e.g. list or tuple) of real numbers (int or float) or instances of a data type class implementing a 'real life measurement' with the associated uncertainty of the measurement, which is API compatible with the **phyqus_lib.base_classes.MeasuredValue** class, i.e. the 'mean' value of the measurement being accessible via field / property *Value* and the associated uncertainty - via field / property *SE*. Such sequence does not have to be homogeneous, and it may be an arbitrary mixture of such instances, integer and floating point numbers. A 2D statistical data set should be passed as two such sequences of the same length. The return values of the functions must be:

- a real number - integer or floating point - for the functions returning a single scalar value
- list of real numbers - for the function calculating mode(s) of the distribution
- a list of 2-element tuples of a real numbers or a dictionary (real \rightarrow int ≥ 0) for the function calculating a histogram

Verification Method: T

Requirement ID: REQ-FUN-210**Title:** Performance of function to calculate the minimum value of the data sample

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function, it returns the minimum value of all 'mean' values, i.e. the same value as the built-in function *min()* would return on the same sequence with all instances of the measurements with uncertainty class converted into their 'mean' values (integer or floating point).

Verification Method: T

Requirement ID: REQ-FUN-220**Title:** Performance of function to calculate the maximum value of the data sample

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function, it returns the maximum value of all 'mean' values, i.e. the same value as the built-in function *max()* would return on the same sequence with all instances of the measurements with uncertainty class converted into their 'mean' values (integer or floating point).

Verification Method: T

Requirement ID: REQ-FUN-230

Title: Performance of function to calculate the median value of the data sample

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function, it returns the median value of all 'mean' values, i.e. the same value as the Standard Library function *statistics.median()* would return on the same sequence with all instances of the measurements with uncertainty class converted into their 'mean' values (integer or floating point).

Verification Method: T

Requirement ID: REQ-FUN-240

Title: Performance of function to calculate the first quartile value of the data sample

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function (length ≥ 2), it returns the first quartile value of all 'mean' values using the linear interpolation between the values of the adjacent elements in the sorted sample. Using Python v3.8 or later it should return the same value as first element of the list returned by the Standard Library function *statistics.quantiles()* with $n=4$ (default).

Verification Method: T

Requirement ID: REQ-FUN-250

Title: Performance of function to calculate the third quartile value of the data sample

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function (length ≥ 2), it returns the third quartile value of all 'mean' values using the linear interpolation between the values of the adjacent elements in the sorted sample. Using Python v3.8 or later it should return the same value as last element of the list returned by the Standard Library function *statistics.quantiles()* with $n=4$ (default).

Verification Method: T

Requirement ID: REQ-FUN-260

Title: Performance of function to calculate a generic k-th of m-quantile of the data sample

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function (length ≥ 2), it returns the k-th of m-quantile value of all 'mean' values using the linear interpolation between the values of the adjacent elements in the sorted sample. Using Python v3.8 or later it should return the same value as the (k-1)th element (indexing from 0) of the list returned by the Standard Library function *statistics.quantiles()* with

$n=m$, providing $0 < k < m$. The 0-th quantile is the first element of the sorted in ascending order sequence of the 'mean' values, whereas the m -th quantile is the last element of that sequence.

Verification Method: T

Requirement ID: REQ-FUN-270

Title: Performance of function to calculate a histogram of the data sample

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function (length ≥ 2), it returns a histogram of all 'mean' values, i.e. how many data points fall into each of the equidistant bins. The following rules are applied:

- The minimum value observed in the data sample belong to the left-most (min value) bin
- The maximum value observed in the data sample belong to the right-most (max value) bin
- The value of a bin is the mid-point of the range that bin covers
- The bins are equidistant, i.e. each cover the same range, equal to the difference between the values of the adjacent bins
- All bins are present, even if they are empty
- The min/max values of the data sample, number of bins N and the bin size S are related as

$$N - 2 \leq \frac{\max(X) - \min(X)}{S} \leq N$$
- The number of bins or the bin size can be requested specifically via keyword-only arguments:
 - If the number of bins N is passed, the minimum and maximum values in the sample are centered to the left- and right-most bins, and the step is defined as $S = \frac{\max(X) - \min(X)}{N-1}$
 - If the bin size S is passed, the arithmetic mean of the sample is centered to its respective bin, and the total number of bins is defined as

$$N = \lceil \frac{\max(X) - \langle X \rangle}{S} - \frac{1}{2} \rceil + \lceil \frac{\langle X \rangle - \min(X)}{S} - \frac{1}{2} \rceil + 1$$
 , where the first and the second parts of the sum are the number of bins to the right and to the left from the 'central' one
 - If neither N nor S are defined by the user, the $N = 20$ number of bins is used
 - If both N and S are defined by the user, the bin size argument is ignored, and value defined by the number of bins is used instead
- The edge case is when the data sample contain only the same value elements, in which case the histogram should contain only a single key : value pair

Verification Method: T

Requirement ID: REQ-FUN-280

Title: Performance of function to calculate the modes of sample data

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function, it returns a list of all values, which occur most frequently occurring values, i.e. the mode(s) of the sample distribution. Using Python v3.8 or later it should return a list containing the same elements but, possibly, in a different order as the list returned by the Standard Library function *statistics.multimode()*.

Verification Method: T

Requirement ID: REQ-FUN-290

Title: Performance of function to calculate the Spearman rank correlation coefficient of the data sample

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function, it returns the Spearman rank correlation coefficient of the 'means', which is the Pearson's correlation of the *fractional ranks* of the sorted X and Y sequences.

Verification Method: T

Requirement ID: REQ-FUN-2A0

Title: Performance of function to calculate the Kendall rank correlation coefficient of the data sample

Description: With a random sequence of the mix of integer, floating point numbers and instances of the measurements with uncertainty class passed into the function, it returns the Kendall rank correlation coefficient of the 'means' using the τ -b variant of the algorithm, i.e. accounting for the ties.

Verification Method: T

Alarms, warnings and operator messages

Requirement ID: REQ-AWM-200

Title: Unacceptable type of the input data

Description: The **TypeError** or its sub-class should be raised in response to the unacceptable input (argument(s) data type(s)) in the following situations:

- A 1D data set (or one of the sub-sets of the 2D data set) is not a flat sequence of real numbers of 'real life measurements' with the associated uncertainties, which means:
 - The respective argument is not a sequence, OR
 - The respective argument is a sequence, but, at least, one element of it is neither integer, or floating point number, or an instance of a measurement with the associated uncertainty data type class
- The required quantile index or total number of quantiles argument is not an integer number
- The requested number of bins (in histogram) is not an integer number
- The requested bin size (in histogram) is not a real number

Verification Method: T

Requirement ID: REQ-AWM-201

Title: Unacceptable value(s) of the input data

Description: The **ValueError** or its sub-class should be raised in response to the unacceptable input (argument(s) data value(s)) in the following situations:

- A 1D data set (or one of the sub-sets of the 2D data set) is an empty sequence
- A 1D data set is shorter than 2 points in the case of the 1st and the 3rd quartile as well as the quantile functions
- The lengths of the sub-sets of a 2D data set are not equal
- The total number of quantiles is an integer, but not positive
- The required quantile index is negative or larger than the total number of quantiles
- The requested number of bins (in histogram) is an integer but not positive
- The requested bin size (in histogram) is a real number but not positive

Verification Method: T