

Test Report on the Module statistics_lib.distribution_classes

Conventions

Each test is defined following the same format. Each test receives a unique test identifier and a reference to the ID(s) of the requirements it covers (if applicable). The goal of the test is described to clarify what is to be tested. The test steps are described in brief but clear instructions. For each test it is defined what the expected results are for the test to pass. Finally, the test result is given, this can be only pass or fail.

The test format is as follows:

Test Identifier: TEST-[I/A/D/T]-XYZ

Requirement ID(s): REQ-uvw-xyz

Verification method: I/A/D/T

Test goal: Description of what is to be tested

Expected result: What test result is expected for the test to pass

Test steps: Step by step instructions on how to perform the test

Test result: PASS/FAIL

The test ID starts with the fixed prefix 'TEST'. The prefix is followed by a single letter, which defines the test type / verification method. 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 test ordering number for this object. E.g. 'TEST-T-112'. Each test type has its own counter, thus 'TEST-T-112' and 'TEST-A-112' tests 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

Tests definition (Analysis)

Test Identifier: TEST-A-700

Requirement ID(s): REQ-FUN-700

Verification method: A

Test goal: All required functionality is implemented and performs correctly.

Expected result: All required classes and functions implementing the statistical significance testing are present and function as expected, i.e. all TEST-T-7xy and TEST-D-700 tests defined in this document are passed.

Test steps: Analyze the source code of the module [stat_tests](#) as well as of the unit-test module [/Tests/UT007_stat_tests](#). Execute the mentioned unit-test module. Also run the demonstration test defined as TEST-D-700, see module [/Tests/DT007_stat_tests](#).

Test result: PASS

Tests definition (Test)

Test Identifier: TEST-T-700

Requirement ID(s): REQ-FUN-700

Verification method: T

Test goal: A sub-class of **TypeError** exceptions is raised in response to an improper type argument of a test function

Expected result: The mentioned exception is raised when:

- Not an instance of **statistics_lib.data_classes.Statistics1D** for expected data sample parameter
- Not an instance of **int** or **float** for the expected mandatory or optional test parameter
- Not an enumeration value of the expected test type parameter
- Not a **float** value for the expected confidence level parameter

Test steps: This step is performed for each of the testing functions separately. Try to call the function in question with only one argument at the time violating the restrictions, repeat with the different unacceptable types for the same argument. Each argument **MUST** be checked, including the optional keyword ones. However, more than one 'wrong' argument at the same type check is optional. Check, that the expected exception type is raised each time.

Test result: PASS

Test Identifier: TEST-T-701

Requirement ID(s): REQ-FUN-701

Verification method: T

Test goal: A sub-class of **ValueError** exceptions is raised in response to an improper value of an argument of a test function, if the data type of the argument is as expected.

Expected result: The mentioned exception is raised when:

- The confidence level is a floating number but not in the interval (0, 1) - all tests

- Population standard deviation is an integer or floating point but not positive - Z-test and chi-squared test
- Delta parameter of the generic F-test is an integer or floating point but not positive
- Length of the data is less than 2 elements

Test steps: This step is performed for each of the testing functions separately. Try to call the function in question with only one argument at the time violating the restrictions, repeat with the different unacceptable values of a proper data type for the same argument. Each concerned argument **MUST** be checked. Check, that the expected exception type is raised each time.

Test result: PASS

Test Identifier: TEST-T-702

Requirement ID(s): REQ-SIO-700, REQ-SIO-701, REQ-SIO-702

Verification method: T

Test goal: Check that all functions conform the design signature

Expected result: A function does not raise any exception and returns the expected value (object, an instance of the test report class) with the expected state as long as the arguments are of the proper types and values. Specifically, the following conventions are followed:

- The data sample(s) is passed as the first mandatory positional argument(s) - as either one or two instances of **statistics_lib.data_classes.Statistics1D** class
- The mandatory test parameters (for Z-test, Chi-squared test and single sample Student's t-test) are passed as mandatory positional arguments following the data sample(s) arguments - as native Python data types **int** or **float** values
- The test type (enumeration value) is passed as the last mandatory positional argument, except for the ANOVA, Levene and Brown-Forsythe tests, where it is not needed, since they are by definition single-sided. An enumeration type is used for this argument.
- The confidence level is passed as keyword-only argument of **float** type in the range (0, 1) (with the default value or 0.95)
- The optional parameters (as delta in generic F-test and expected difference in paired t-test) is passed as keyword-only arguments of **int** or **float** types
- The test functions should return an instance of a report class - see REQ-FUN-7B0 and REQ-FUN-7B1 covered by TEST-T-7B0 and TEST-D-700

Test steps: This step is performed for each of the testing functions separately. Try to call the function in question with a set of proper arguments and check the returned value. The actual content / state of the returned object is checked separately for each function, as a part of the specific TEST-T-710 to TEST-T-7A0 tests.

Test result: PASS

Test Identifier: TEST-T-710

Requirement ID(s): REQ-FUN-710

Verification method: T

Test goal: Check that the Z-test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate a sequence of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma. Instantiate **Statistics1D** class with this sequence. Obtain the sample's mean and standard deviation using the respective properties of the instance. Obtain lower and upper critical values for 95% and 90% confidence from an instance of **Z_Distribution** class. Calculate the offset to the sample's mean that should be used instead of real underlying distribution's mean in combination with the population's standard deviation in order obtain a test value larger / smaller than the critical values, so the respective test will fail to reject the null hypothesis.

For each of the 1-sided / 2-sided test variant check the following situations:

- Failed test due to too large difference between the sample's mean and the expected population mean (corrected, not actual)
- Test passes (null hypothesis is rejected) with:
 - Lower confidence level, OR
 - Larger value of the population's standard , OR
 - Passed population's mean value closer to the actual sample's mean

In each of the cases check the relation of the calculated test value to the critical values; of the p-value with respect to the selected confidence level; and the overall test result.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-720

Requirement ID(s): REQ-FUN-720

Verification method: T

Test goal: Check that the Student's t-test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate a sequence of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma. Instantiate **Statistics1D** class with this sequence. Obtain the sample's mean and standard deviation using the respective properties of the instance. Obtain lower and upper critical values for 95% and 90% confidence from an instance of **Student** class. Calculate the offset to the sample's mean that should be used instead of real underlying distribution's mean in combination with sample's standard deviation in order obtain a test value larger / smaller than the critical values, so the respective test will fail to reject the null hypothesis.

For each of the 1-sided / 2-sided test variant check the following situations:

- Failed test due to too large difference between the sample's mean and the expected population mean (corrected, not actual)
- Test passes (null hypothesis is rejected) with:
 - Lower confidence level, OR
 - Passed population's mean value closer to the actual sample's mean

In each of the cases check the relation of the calculated test value to the critical values; of the p-value with respect to the selected confidence level; and the overall test result.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-730

Requirement ID(s): REQ-FUN-730

Verification method: T

Test goal: Check that the chi-squared test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate a sequence of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma. Instantiate **Statistics1D** class with this sequence. Obtain the sample's mean and standard deviation using the respective properties of the instance. Obtain lower and upper critical values for 95% and 90% confidence from an instance of **ChiSquared** class. Calculate the population standard deviation required to obtain a test value larger / smaller than the critical values, so the respective test will fail to reject the null hypothesis, which can be passed instead of the actual standard deviation of the underlying population.

For each of the 1-sided / 2-sided test variant check the following situations:

- Failed test due to too large difference between the sample's mean and the expected population mean (corrected, not actual)
- Test passes (null hypothesis is rejected) with:
 - Lower confidence level, OR
 - Larger value of the population's standard deviation

In each of the cases check the relation of the calculated test value to the critical values; of the p-value with respect to the selected confidence level; and the overall test result.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-740

Requirement ID(s): REQ-FUN-740

Verification method: T

Test goal: Check that the unpaired Student's t-test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate two sequences of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma - use the same sigma for the both sequences, but make sure that the mean values are different. The lengths of the sequences do not have to be the same. Create two instances of the **Statistics1D** class with these sequences. Obtain the sample's mean and standard deviation using the respective properties of the instances. Calculate the actual test value, calculate the CDF value using an instance of **Student** class (use the proper number of degrees of freedom) and select the confidence levels to expect failure and success outcomes.

For each of the 1-sided / 2-sided test variant check the following situations:

- Failed test due to too high confidence level
- Test passes (null hypothesis is rejected) with the lower confidence level

In each of the cases check the relation of the calculated test value to the critical values; of the p-value with respect to the selected confidence level; and the overall test result.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-750

Requirement ID(s): REQ-FUN-750

Verification method: T

Test goal: Check that the paired Student's t-test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate two sequences of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma - use the same sigma for the both sequences, but make sure that the mean values are different. The lengths of the sequences **MUST** be the same. Create two instances of the **Statistics1D** class with these sequences. Obtain the sample's mean and standard deviation using the respective properties of the instances. Calculate the actual test value, calculate the CDF value using an instance of **Student** class (use the proper number of degrees of freedom) and select the confidence levels to expect failure and success outcomes.

For each of the 1-sided / 2-sided test variant check the following situations:

- Failed test due to too high confidence level
- Test passes (null hypothesis is rejected) with the lower confidence level, OR
- Test passes due to the expected bias passed with the value close to the actual samples means difference

In each of the cases check the relation of the calculated test value to the critical values; of the p-value with respect to the selected confidence level; and the overall test result.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-760

Requirement ID(s): REQ-FUN-760

Verification method: T

Test goal: Check that the Welch t-test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate two sequences of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma - use the slightly different sigma values for the both sequences, but make sure that the mean values are different. The lengths of the sequences can be the same or different. Create two instances of the **Statistics1D** class with these sequences. Obtain the sample's mean and standard deviation using the respective properties of the instances. Calculate the actual test value and the degrees of freedom, calculate the CDF value using an instance of **Student** class (use the proper number of degrees of freedom) and select the confidence levels to expect failure and success outcomes.

For each of the 1-sided / 2-sided test variant check the following situations:

- Failed test due to too high confidence level
- Test passes (null hypothesis is rejected) with the lower confidence level

In each of the cases check the relation of the calculated test value to the critical values; of the p-value with respect to the selected confidence level; and the overall test result.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-770

Requirement ID(s): REQ-FUN-770

Verification method: T

Test goal: Check that the Welch t-test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate two sequences of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma - use different sigma values for the both sequences. The lengths of the sequences do not have to be the same. Create two instances of the **Statistics1D** class with these sequences. Obtain the samples' standard deviation using the respective properties of the instances. Calculate the actual test value, calculate the CDF value using an instance of **F_Distribution** class (use the proper numbers of degrees of freedom) and select the confidence levels to expect failure and success outcomes.

For each of the 1-sided / 2-sided test variant check the following situations:

- Failed test due to too high confidence level
- Test passes (null hypothesis is rejected) with the lower confidence level, OR
- Test passes with the ratio of the samples' standard deviations ratio compensated using the proper value of the delta parameter

In each of the cases check the relation of the calculated test value to the critical values; of the p-value with respect to the selected confidence level; and the overall test result.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-780

Requirement ID(s): REQ-FUN-780

Verification method: T

Test goal: Check that the ANOVA F-test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate two sequences of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma - use different sigma values and mean values for the both sequences. The lengths of the sequences do not have to be the same. Create two instances of the **Statistics1D** class with these sequences. Run the test (call the respective function) at the default confidence level (95%) and check the outcome - the rejection of the null hypothesis and p-value.

- If the test passes (null hypothesis is rejected) - choose higher confidence level based on the p-value for the test to fail, and re-run the test.
- If the test fails (null hypothesis is not rejected) - choose lower confidence level based on the p-value for the test to pass, and re-run the test.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-790

Requirement ID(s): REQ-FUN-790

Verification method: T

Test goal: Check that the Levene test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate two sequences of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma - use different sigma values and mean values for the both sequences. The lengths of the sequences do not have to be the same. Create two instances of the **Statistics1D** class with these sequences. Run the test (call the respective function) at the default confidence level (95%) and check the outcome - the rejection of the null hypothesis and p-value.

- If the test passes (null hypothesis is rejected) - choose higher confidence level based on the p-value for the test to fail, and re-run the test.
- If the test fails (null hypothesis is not rejected) - choose lower confidence level based on the p-value for the test to pass, and re-run the test.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-7A0

Requirement ID(s): REQ-FUN-7A0

Verification method: T

Test goal: Check that the Brown-Forsythe test function functions as expected

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Generate two sequences of random numbers ($N \sim 20$) using Gaussian distribution class with randomly selected mean and sigma - use different sigma values and mean values for the both sequences. The lengths of the sequences do not have to be the same. Create two instances of the **Statistics1D** class with these sequences. Run the test (call the respective function) at the default confidence level (95%) and check the outcome - the rejection of the null hypothesis and p-value.

- If the test passes (null hypothesis is rejected) - choose higher confidence level based on the p-value for the test to fail, and re-run the test.

- If the test fails (null hypothesis is not rejected) - chose lower confidence level based on the p-value for the test to pass, and re-run the test.

Also perform the demonstration test TEST-D-700.

Test result: PASS

Test Identifier: TEST-T-7B0

Requirement ID(s): REQ-FUN-7B0, REQ-FUN-7B1

Verification method: T

Test goal: Check that the test report class functions properly

Expected result: The function being tested generates and returns an object (test report class instance) with its state properly reflecting the result of the test based on the parameters of the test and the statistical properties of the used random data sample.

Test steps: Intantiate the respective class with arbitrary values for the string (descriptors) arguments, an arbitrary CDF value in the range (0.05, 0.95) and some arbitrary chosen critical values

- (None, some value) - should be treated as 1-sided right-tailed, 'greater'
- (some value, None) - should be treated as 1-sided left-tailed, 'less'
- (some value, some other greater value) - should be treated as 2-sided, 'not equal'
- (some value, the same value) - should be treated as 1-sided right-tailed, 'not equal'

whereas the test value MUST be chosen with respect to the critical values to ensure the failure and pass outcomes of the test - in the separate calls.

Considering the 2-sided test, make sure that $CDF > 0.5$ for the test value closer to the upper bound, and $CDF < 0.5$ in the opposite case.

Check that the boolean test result and the floating point p-value are calculated correctly. Also perform the demonstration test TEST-D-700.

Test result: PASS

Tests definition (Demonstration)

Test Identifier: TEST-D-700

Requirement ID(s): REQ-FUN-7B0, REQ-FUN-7B1

Verification method: D

Test goal: Check functionality of the test report class

Expected result: An instance of the concerned class properly derives the results of a performed statistical test from the parameters of the instantiation method. The human-readable text report provides all the required data, and it is correct, including:

- Name and type of the test (1- or 2-sided, etc.) with the (optional) parameters and the confidence level
- Identifiers of the used data set(s)
- Used model distribution
- Null and alternative hypothesis
- Critical value(s) and the test value
- If the null hypothesis is rejected
- The calculated p-value

Test steps: Instantiate the class in question with some arbitrary but proper values, emulating results of the testing functions. Specifically, check the following cases of the critical values tuple:

- (None, some value) - should be treated as 1-sided right-tailed, 'greater'
- (some value, None) - should be treated as 1-sided left-tailed, 'less'
- (some value, some other greater value) - should be treated as 2-sided, 'not equal'
- (some value, the same value) - should be treated as 1-sided right-tailed, 'not equal'

Use the actual test value such that some tests are considered to fail to reject the null hypothesis, whilst other - reject the null hypothesis.

Generate two random samples using the same Gaussian distribution with arbitrary parameters and use them to demonstrate the outcome of all of the implemented testing functions.

Test result: PASS

Traceability

For traceability the relation between tests and requirements is summarized in the table below:

Requirement ID	Covered in test(s)	Verified [YES/NO]
REQ-FUN-700	TEST-A-700	YES
REQ-FUN-710	TEST-T-710	YES
REQ-FUN-720	TEST-T-720	YES
REQ-FUN-730	TEST-T-730	YES
REQ-FUN-740	TEST-T-740	YES
REQ-FUN-750	TEST-T-750	YES
REQ-FUN-760	TEST-T-760	YES
REQ-FUN-770	TEST-T-770	YES
REQ-FUN-780	TEST-T-780	YES
REQ-FUN-790	TEST-T-790	YES
REQ-FUN-7A0	TEST-T-7A0	YES
REQ-FUN-7B0	TEST-T-7B0, TEST-D-700	YES

Requirement ID	Covered in test(s)	Verified [YES/NO])
REQ-FUN-7B1	TEST-T-7B0, TEST-D-700	YES
REQ-SIO-700	TEST-T-702	YES
REQ-SIO-701	TEST-T-702	YES
REQ-SIO-702	TEST-T-702	YES
REQ-AWM-700	TEST-T-700	YES
REQ-AWM-701	TEST-T-701	YES
Software ready for production [YES/NO]		Rationale
YES		All tests passed