1. Link nguồn ML:

<http://gallery.cortanaintelligence.com/Experiment/Experiment-created-on-12-8-2016-1>

1. Summarize Data

## [Module Overview](javascript:void(0))

You can use the [Summarize Data](https://msdn.microsoft.com/en-us/library/azure/dn905933.aspx) module to create a set of standard statistical measures that describe each column in the input table. The module does not return the original dataset. Instead, it generates a row for each column, beginning with the column name and followed by relevant statistics for that column, based on its data type.

Such reports are useful when you want to understand the characteristics of the complete dataset. For example, you might need to know:

* How many missing values are there in each column?
* How many unique categorical values are there in a feature column?
* What is the mean and standard deviation of the column?

You can get a partial list of statistics by using the **Visualize** option in any module that outputs a dataset, but the visualization includes only some top number of rows. By outputting the statistics in a tabular dataset, you can use the data in BI reporting tools or provide the values as input to another custom operation in the experiment.

## [How to Use Summarize Data](javascript:void(0))

You can use [Summarize Data](https://msdn.microsoft.com/en-us/library/azure/dn905933.aspx) to calculate the following statistics on dense or sparse columns:

|  |  |
| --- | --- |
| **Type** |  |
| **Counts** | * Count of all rows * Unique value count * Missing value count |
| **Range** | * Minimum value * Maximum value * Range |
| **Statistical summaries** | * Mean * Mean deviation * Median * Mode * Sample variance * Sample standard deviation * Sample skewness * Sample kurtosis |
| **Percentiles** | * First quartile * Third quartile * 0.5% percentile * 1% percentile * 5% percentile * 95% percentile * 99% percentile * 99.5% percentile |

## [Examples](javascript:void(0))

For examples of how to use [Summarize Data](https://msdn.microsoft.com/en-us/library/azure/dn905933.aspx) in an experiment, see these sample experiments in the [Model Gallery](http://gallery.azureml.net/):

* The [Download dataset from UCI](http://go.microsoft.com/fwlink/?LinkId=525938) sample reads a dataset in CSV format by using its URL in the UCI Machine Learning Repository, and it generates some basic statistics about the dataset.
* The [Dataset Processing and Analysis](http://go.microsoft.com/fwlink/?LinkId=525733) sample loads the dataset into the workspace, changes column names, and adds metadata.
* The [Prediction of student performance](http://go.microsoft.com/fwlink/?LinkId=525727) sample reads data stored in TSV format from Azure Blob storage.

## [Technical Notes](javascript:void(0))

* For numeric and Boolean columns, you can output the mean, median, mode, and standard deviation.

For non-numeric columns, only the values for **Count**, **Unique value count**, and **Missing value count** are computed. For other statistics, a null value is returned.

* Columns that contain Boolean values are processed as follows:
  + When calculating **Min**, a logical AND is applied.
  + When calculating **Max**, a logical OR is applied.
  + When computing **Range**, the module first checks whether the number of unique values in the column equals 2.
  + When computing any statistic that requires floating-point calculations, values of True are treated as 1.0, and values of False are treated as 0.0.
* The module calculates these statistics on the assumption that the statistical instances belong to a representative sample of a population. if you need statistics calculated for the population, use the options in [Compute Elementary Statistics](https://msdn.microsoft.com/en-us/library/azure/dn905936.aspx) , which can compute either sample or population statistics.

1. Compute Linear Correlation

## [Module Overview](javascript:void(0))

You can use the [Compute Linear Correlation](https://msdn.microsoft.com/en-us/library/azure/dn905819.aspx) module to compute a set of Pearson correlation coefficients for each possible pair of variables in the input dataset.

The Pearson correlation coefficient, sometimes called Pearson’s R test, is a statistical value that measures the linear relationship between two variables. By examining the coefficient values, you can infer something about the strength of the relationship between the two variables, and whether they are positively correlated or negatively correlated.

## [How to Use Linear Correlation](javascript:void(0))

There are no parameters to set for [Compute Linear Correlation](https://msdn.microsoft.com/en-us/library/azure/dn905819.aspx) . However, some other restrictions apply:

* The [Compute Linear Correlation](https://msdn.microsoft.com/en-us/library/azure/dn905819.aspx) module can process only numeric values. All other types of values, including missing values, non-numeric values, and categorical values, are treated as NaNs.

To avoid creating a lot of unnecessary columns of NaNs, we recommend that you use [Select Columns in Dataset](https://msdn.microsoft.com/en-us/library/azure/dn905883.aspx) to pass in only the columns for which you want to compute coefficients.

* Pearson’s correlation is calculated for all numeric columns in the dataset that are passed as input.
* [Compute Linear Correlation](https://msdn.microsoft.com/en-us/library/azure/dn905819.aspx) is intended to be used with data that has no missing values. Before running this module, use [Clean Missing Data](https://msdn.microsoft.com/en-us/library/azure/dn906028.aspx)or other methods to impute missing values.
* Be sure to replace placeholders with other appropriate values before using this module.

For example, if NaNs were inserted for missing values when the dataset was loaded from the source, it could cause an error. If a placeholder (such as 999 or -1) was used, it could cause bad results.

* The correlation coefficient should not be calculated if the relationship is not linear. You can visually assess the linearity of two variables by using a scatter plot, or you can calculate a regression equation for the two variables.

### [**Results**](javascript:void(0))

Given two feature columns, the [Compute Linear Correlation](https://msdn.microsoft.com/en-us/library/azure/dn905819.aspx) module returns the scalar Pearson product moment (sample) correlation coefficient.

Given a matrix, the [Compute Linear Correlation](https://msdn.microsoft.com/en-us/library/azure/dn905819.aspx) module returns a set of Pearson product moment correlations between each pair of feature columns.

The Pearson correlation coefficient (often denoted as r) ranges in value from +1 to -1, where +1 indicates a strong linear relationship, and -1indicates no linear relationship between the two variables.

The interpretation of the coefficients depends very much on the problem you are modeling and the variables you are studying.

* For example, if you are certain the variables are unrelated and yet the Pearson’s correlation coefficient is positive, you should investigate further.
* Conversely, if you use linear correlation on two variables that you know to be perfectly correlated, and the coefficient values are not what you expect, it might indicate a problem in the data.

Thus it is important to understand the context of the data when reporting and interpreting Pearson's correlation coefficient.

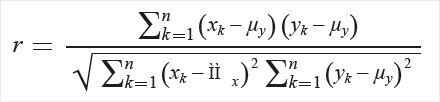
## [Examples](javascript:void(0))

To see how this module is used in machine learning experiments, see these sample experiments in the [Model Gallery](http://gallery.azureml.net/):

* In the [Data Processing and Analysis](http://go.microsoft.com/fwlink/?LinkId=525733) sample experiment, [Compute Linear Correlation](https://msdn.microsoft.com/en-us/library/azure/dn905819.aspx) is used to identify potential feature columns.

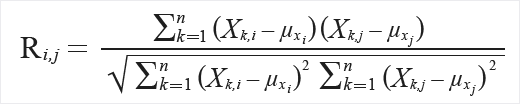
## [Technical Notes](javascript:void(0))

If the column that is passed as input contains scalars, the input arrays (x and y) are treated as vectors and the Pearson product moment correlation is computed as follows:



In this formula, each array contains n elements and the means of the x and y samples are µx and µy respectively.

For a matrix, a matrix of data (X) is input, in which each column represents a vector of values. The data matrix should be n-by-m. The output is the m-by-m matrix, R as defined by



In this formula, μx represents the mean value of the column xi. The elements at I,j always equal 1, as they represent the correlation of a vector with itself.

1. Evaluate Probability Function

## [Module Overview](javascript:void(0))

You can use the [Evaluate Probability Function](https://msdn.microsoft.com/en-us/library/azure/dn905813.aspx) module to calculate a variety of useful statistical measures that describe a column’s distribution, such as the Bernoulli, Pareto, or Poisson distributions.

When you evaluate your data against a probability distribution, you are mapping column values against a set of values with known properties. By knowing how well your data corresponds to one of these well-known distributions, you might be able infer other properties of your data. In general, you can get better predictions from a model when you can identify the distribution that fits the data best.

The question of which probability distribution function to use depends on the data and the variables that are being measured. For example, some distributions are designed to describe probabilities of discrete values; others are intended for use only with continuous numerical variables.

For all probability distributions, you can choose from these methods of representing the distribution:

* Cumulative distribution function (**cdf**)
* Inverse cumulative distribution function (**InverseCdf**)
* Probability density function (**pdf**), or mass function

The **Probability Function Evaluation** module requires a dataset that contains at least one column of numerical values as input. It returns a data table that contains the values computed by the specified probability function. Optionally, you can also return the original values from the analyzed columns.

## [How to Use Evaluate Probability Function](javascript:void(0))

This module has many options, which change depending on the type of probability distribution you want to compute.

1. Identify the columns for which to calculate the probability distribution. The columns you select must have a numerical data type.
2. The range of data in the column must also be valid, given the selected probability function. Otherwise, an error or NaN result may occur.
3. Choose the kind of probability distribution that you want to calculate.
4. Set parameters as required by the distribution.
5. Choose one of three methods to use in evaluating each probability function:

**Cdf** computes the cumulative distribution function.

**InverseCdf** computes the inverse cumulative distribution function.

**Pdf** computes the probability density function, or mass function.

Decide how you want to output the results. You can replace column values with the probability distribution values, append the new values to the dataset, or return only the probability distribution values.

|  |
| --- |
| **System_CAPS_noteNote** |
| For sparse columns, any values that correspond to background zeros will not be processed. |

### [**Supported Probability Distributions**](javascript:void(0))

By using the [Evaluate Probability Function](https://msdn.microsoft.com/en-us/library/azure/dn905813.aspx) module, you can calculate the following distribution types:

#### [Bernoulli](javascript:void(0))

The Bernoulli distribution is a distribution over bits – in other words, a discrete distribution with only two possible values. The parameter pspecifies the probability that a 1 is generated.

To calculate the **Bernoulli** distribution, you must set the following options:

***Probability of success***

Type a number (**float**) between 0.0 and 1.0 that specifies the probability of success. The default is .5.

#### [Beta](javascript:void(0))

The Beta distribution is a continuous univariate distribution.

To calculate the **Beta** distribution, you must set the following options:

***Shape***

Type a value to change the shape of the distribution.

A shape parameter is any parameter of a probability distribution that does not define its location or scale. Therefore, when you enter a value for shape, the parameter changes the shape of the distribution rather than moving, stretching, or shrinking it.

The value must be a number (**double**). The default is 1.0.

***Scale***

Type a number to use for scaling the distribution.

By applying a scale value to the distribution, you can shrink or stretch it.

The default value is 1.0. Values must be positive numbers.

Upper bound

Type a number (**double**) that represents the upper bound of the distribution. The default is 1.0.

***Lower bound***

Type a number (**double**) that represents the lower bound of the distribution. The default is 0.0.

#### [Binomial](javascript:void(0))

This is a discrete univariate distribution.

The binomial distribution is used to model the number of successes in a sample. Replacement is used when sampling. For sampling without replacement, use the [Hypergeometric distribution](https://msdn.microsoft.com/en-us/library/azure/dn905813.aspx#bkmk_Hypergeometric).

To calculate the **Binomial** distribution, you must set the following options:

***Probability of success***

Type a number (**float**) between 0.0 and 1.0 that indicates the probability of success. The default is .5.

***Number of trials***

Specify the number of trials.

Use an **integer**, with a minimum value of 1. The default is 3.

#### [Cauchy](javascript:void(0))

The Cauchy distribution is a symmetric continuous probability distribution.

To calculate the **Cauchy** distribution, you must set the following options:

***Location***

Type a number (**double**) that represents the location of the 0th element.

By specifying a value for the **Location** parameter, you can shift the probability distribution up or down a numeric scale.

The default is 0.0.

#### [ChiSquare](javascript:void(0))

This distribution is a sum of the squares of k independent, standard, normal, random variables.

To calculate the **ChiSquare** distribution, you must set the following options:

***Number of degrees of freedom***

Type a number (**double**) to specify the degrees of freedom. The default is 1.0.

#### [ChiSquareRightTailed](javascript:void(0))

To calculate the **ChiSquareRightTailed** distribution, you must set the following options:

***Number of degrees of freedom***

Type a number (**double**) to specify the degrees of freedom. The default is 1.0.

#### [Exponential](javascript:void(0))

The exponential distribution is a distribution over the real numbers parameterized by one non-negative parameter.

To calculate the **Exponential** distribution, you must set the following options:

***Lambda***

Type a number (**double**) to use as the lambda parameter. The default is 1.0.

#### [FFisher](javascript:void(0))

The **FFisher** option lets you create the Fisher distribution, also known as the Fisher F-distribution, Snedecor distribution, or Fisher-Snedecor distribution. This distribution is two-tailed.

To calculate this distribution, you must set the following options:

***Numerator degrees of freedom***

Type a number (**double**) to specify the degrees of freedom that is used in the numerator. The default is 3.0.

***Denominator degrees of freedom***

Type a number (**double**) to specify the degrees of freedom that is used in the denominator. The default is 6.0.

#### [FFisherRightTailed](javascript:void(0))

With the **FfisherRightTailed** option, you can create a right-tailed Fisher distribution. The Fisher distribution is also known as the Fisher F-distribution, Snedecor distribution, or Fisher-Snedecor distribution. This particular form of the distribution is right-tailed.

To calculate the **FFisherRightTailed** distribution, you must set the following options:

***Numerator degrees of freedom***

Type a number (**double**) to specify the degrees of freedom that is used in the numerator. The default is 3.0.

***Denominator degrees of freedom***

Type a number (**double**) to specify the degrees of freedom that is used in the denominator. The default is 6.0.

#### [Gamma](javascript:void(0))

The gamma distribution is a family of continuous probability distributions with two parameters. For example, chi-squared is a special case of the gamma distribution.

To calculate the **Gamma** distribution, you must set the following options:

***Scale***

Type a value to use for scaling the distribution.

By applying a scale value to the distribution, you can shrink or stretch it.

The default value is 1.0. Values must be positive numbers.

***Location***

Type a number (**double**) that represents the location of the 0th element.

By specifying a value for the **Location** parameter, you can shift the probability distribution up or down a numeric scale.

The default is 0.0.

#### [GeneralizedExtremeValues](javascript:void(0))

The **GeneralizedExtremeValues** option lets you create a distribution developed to handle extreme values. The generalized extreme value (GEV) distribution is actually a group of continuous probability distributions that combines the Gumbel, Fréchet, and Weibull distributions (also known as type I, II, and III extreme value distributions).

For more information about extreme value theory, see this article in Wikipedia: [Fisher-Tippet-Gnedenko theorem](http://en.wikipedia.org/wiki/Fisher%e2%80%93Tippett%e2%80%93Gnedenko_theorem).

To calculate the **GeneralizedExtremeValues** distribution, you must set the following options:

***Shape***

Type a value to change the shape of the distribution.

A shape parameter is any parameter of a probability distribution that does not define its location or scale. Therefore, when you enter a value for shape, the parameter changes the shape of the distribution rather than moving, stretching, or shrinking it.

The value must be a number (**double**). The default is 1.0.

***Scale***

Type a value to use for scaling the distribution.

By applying a scale value to the distribution, you can shrink or stretch it.

The default value is 1.0. Values must be positive numbers.

***Location***

Type a number (**double**) that represents the location of the 0th element.

By typing a value for the **Location** parameter, you can shift the probability distribution up or down a numeric scale.

The default is 0.0.

#### [Geometric](javascript:void(0))

The Geometric distribution is a distribution over positive integers parameterized by one positive real number. This implementation of the Geometric distribution will never generate zeros.

To calculate the **Geometric** distribution, you must set the following options:

***Probability of success***

Type a number (**float**) between 0.0 and 1.0 that indicates the probability of success. The default is .5.

#### [GumbelMax](javascript:void(0))

The Gumbel distribution is one of several extreme value distributions. The **GumbelMax** option implements the Maximum Extreme Value Type 1 distribution.

To calculate the **GumbelMax** distribution, you must set the following options:

***Scale***

Type a value to use for scaling the distribution.

By applying a scale value to the distribution, you can shrink or stretch it.

The default value is 1.0. Values must be positive numbers.

***Location***

Type a number (**double**) that represents the location of the 0th element.

By typing a value for the **Location** parameter, you can shift the probability distribution up or down a numeric scale.

The default is 0.0.

#### [GumbelMin](javascript:void(0))

The Gumbel distribution is one of several extreme value distributions. The Gumbel distribution is also referred to as the Smallest Extreme Value (SEV) distribution or the Smallest Extreme Value (Type I) distribution. The **GumbelMin** option implements the Minimum Extreme Value Type 1 distribution.

To calculate the **GumbelMin** distribution, you must set the following options:

***Scale***

Type a value to use for scaling the distribution.

By applying a scale value to the distribution, you can shrink or stretch it.

The default value is 1.0. Values must be positive numbers.

***Location***

Type a number (**double**) that represents the location of the 0th element.

By typing a value for the **Location** parameter, you can shift the probability distribution up or down a numeric scale.

The default is 0.0.

#### [Hypergeometric](javascript:void(0))

This distribution is a discrete probability distribution that describes the number of successes in a sequence of n draws from a finite population without replacement, just as the binomial distribution describes the number of successes for draws with replacement.

To calculate the **Hypergeometric** distribution, you must set the following options:

***Number of samples***

Type an integer that indicates the number of samples to use. The default is 9.

***Number of success***

Type an integer that defines the value for success. The default is 24.

***Population size***

Specify the population size to use when estimating the hypergeometric distribution.

#### [Laplace](javascript:void(0))

The Laplace distribution is a distribution over the real numbers parameterized by a mean and scale parameter.

To calculate the **Laplace** distribution, you must set the following options:

***Scale***

Type a value to use for scaling the distribution.

By applying a scale value to the distribution, you can shrink or stretch it.

The default value is 1.0. Values must be positive numbers.

***Location***

Type a number (**double**) that represents the location of the 0th element.

By typing a value for the **Location** parameter, you can shift the probability distribution up or down a numeric scale.

The default is 0.0.

#### [Logistic](javascript:void(0))

The logistic distribution is similar to the normal distribution, but it has no limit on the left side of the distribution. The logistic distribution is used in logistic regression and neural network models and for modeling life sciences data.

To calculate the **Logistic** distribution, you must set the following options:

***Scale***

Type a value to use for scaling the distribution.

By applying a scale value to the distribution, you can shrink or stretch it.

The default value is 1.0. Values must be positive numbers.

***Mean***

Type a number (**double**)that indicates the estimated mean value of the distribution. The default is 0.0.

#### [Lognormal](javascript:void(0))

The lognormal distribution is a continuous univariate distribution.

To calculate the **Lognormal** distribution, you must set the following options:

***Mean***

Type a number (**double**) that indicates the estimated mean value of the distribution. The default is 0.0.

***Standard deviation***

Type a positive number (**double**) that indicates the estimated standard deviation of the distribution. The default is 1.0.

#### [NegativeBinomial](javascript:void(0))

The negative binomial distribution is a distribution over the natural numbers with two parameters (r, p). In the special case that r is an integer, you can interpret the distribution as the number of tails before the rth head when the probability of the head is p.

To calculate the **NegativeBinomial** distribution, you must set the following options:

***Probability of success***

Type a number (**float**) between 0.0 and 1.0 that indicates the probability of success. The default is .5.

***Number of success***

Type an integer that specifies the value for success. The default is 24.

#### [Normal](javascript:void(0))

Also known as the Gaussian distribution.

To calculate the **Normal** distribution, you must set the following options:

***Mean***

Type a number (**double**) that indicates the estimated mean value of the distribution. The default is 0.0.

***Standard deviation***

Type a positive number (**double**) that indicates the estimated standard deviation of the distribution. The default is 1.0.

#### [Pareto](javascript:void(0))

The Pareto distribution is a power-law probability distribution that coincides with social, scientific, geophysical, actuarial, and many other types of observable phenomena.

To calculate the **Pareto** distribution, you must set the following options:

***Shape***

Type a value (optional) to change the shape of the distribution.

A shape parameter is any parameter of a probability distribution that does not define its location or scale. Therefore, when you enter a value for shape, the parameter changes the shape of the distribution rather than moving, stretching, or shrinking it.

The value must be a number (**double**). The default is 1.0.

***Scale***

Type a value (optional) to change the shape of the distribution.

A shape parameter is any parameter of a probability distribution that does not define its location or scale. Therefore, when you enter a value for shape, the parameter changes the shape of the distribution rather than moving, stretching, or shrinking it.

The value must be a number (**double**). The default is 1.0.

#### [Poisson](javascript:void(0))

For more information about the Poisson distribution, see [Poisson Regression](https://msdn.microsoft.com/en-us/library/azure/dn905988.aspx). In this implementation, Knuth's method is used to generate Poisson distributed random variables.

To calculate the **Poisson** distribution, you must set the following options:

***Mean***

Type a number (**double**) that indicates the estimated mean value of the distribution. The default is 0.0.

#### [Rayleigh](javascript:void(0))

The Rayleigh distribution is a continuous probability distribution. As an example of how it arises, the wind speed will have a Rayleigh distribution if the components of the two-dimensional wind velocity vector are uncorrelated and normally distributed with equal variance.

To calculate the **Rayleigh** distribution, you must set the following options:

***Lower bound***

Type a number (**double**) that represents the lower bound of the distribution. The default is 0.0.

#### [StandardNormal](javascript:void(0))

To calculate the **StandardNormal** distribution, all you need to do is select the columns.

#### [TStudent](javascript:void(0))

The **TStudent** option implements the univariate Student’s t-distribution.

To calculate the **TStudent** distribution, you must set the following options:

***Number of degrees of freedom***

Type a number (**double**) to specify the degrees of freedom. The default is 1.0.

#### [TStudentRightTailed](javascript:void(0))

Implements the univariate Student’s t-distribution by using one right tail.

To calculate the **TStudentRightTailed** distribution, you must set the following options:

***Number of degrees of freedom***

Type a number (**double**) to specify the degrees of freedom. The default is 1.0.

#### [TStudentTwoTailed](javascript:void(0))

Implements a two-tailed Student’s t-distribution.

To calculate the **TStudentTwoTailed** distribution, you must set the following options:

***Number of degrees of freedom***

Type a number (**double**) to specify the degrees of freedom. The default is 1.0.

#### [Uniform](javascript:void(0))

The uniform distribution is also known as the rectangular distribution.

To calculate the **Uniform** distribution, you must set the following options:

***Lower bound***

Type a number (**double**) that represents the lower limit of the distribution. The default is 0.0.

***Upper bound***

Type a number (**double**) that represents the upper limit of the distribution. The default is 1.0.

#### [Weibull](javascript:void(0))

The Weibull distribution is widely used in reliability engineering. It can use the **Shape** parameter to model many other distributions.

To calculate the **Weibull** distribution, you must set the following options:

***Shape***

Type a value (optional) to change the shape of the distribution.

A shape parameter is any parameter of a probability distribution that does not define its location or scale. Therefore, when you enter a value for shape, the parameter changes the shape of the distribution rather than moving, stretching, or shrinking it.

The value must be a number (**double**). The default is 1.0.

***Scale***

Type a value (optional) to change the shape of the distribution.

A shape parameter is any parameter of a probability distribution that does not define its location or scale. Therefore, when you enter a value for shape, the parameter changes the shape of the distribution rather than moving, stretching, or shrinking it.

The value must be a number (**double**). The default is 1.0.

## [Technical Notes](javascript:void(0))

This module supports all distributions that are provided in the open source MATH.NET Numerics library. For more information, see the documentation for the [Math.Net.Numerics.Distribution](http://numerics.mathdotnet.com/api/MathNet.Numerics.Distributions/index.htm) library.

Right-tailed and two-tailed distributions appear as separate distributions, not as parameterized versions of base distributions. The current behavior is to preserve compatibility with Excel.

1. Test Hypothesis using t-Test

## [Module Overview](javascript:void(0))

You can use the [Test Hypothesis Using t-Test](https://msdn.microsoft.com/en-us/library/azure/dn905917.aspx) module to generate scores for three types of t-tests:

* Single sample t-test
* Paired t-test
* Unpaired t-test

In general, a t-test helps you compare whether two groups have different means. For example, suppose you are evaluating trial data for patients who received Drug A vs. patients who received Drug B, and you need to compare a recovery rate metric for both groups. The null hypothesis would assume that the recovery rate is the same in both groups, and furthermore, that the values for the recovery rate have a normal distribution in both two groups.

By using [Test Hypothesis Using t-Test](https://msdn.microsoft.com/en-us/library/azure/dn905917.aspx) and providing the columns that contain the recovery rates as input, you can get scores that indicate whether the difference is meaningful, which would signify that the null hypothesis should be rejected. The test takes into account factors such as how big the difference is between the values, the size of the sample (larger is better), and how big the standard deviation is (lower is better).

By reviewing the results of the [Test Hypothesis Using t-Test](https://msdn.microsoft.com/en-us/library/azure/dn905917.aspx) module, you can determine whether the null hypothesis is TRUE or FALSE, and review the confidence (P) scores from the t-test.

## [How to Configure Test Hypothesis Using t-Test](javascript:void(0))

1. Decide which kind of t-test to perform.

**Single sample t-test**

Use this option when:

* + You have a single sample of scores.
  + All scores are independent from each other.
  + The sampling distribution of xˉ is normal.

In general, the single sample t-test is used to compare an average value to a known number.

**Paired t-test**

Use this option when:

* + You have a matched pairs of scores. For example, you might have two different measures per person, or matched pairs of individuals (such as a husband and wife).
  + Each pair of scores is independent of every other pair.
  + The sampling distribution of dˉ is normal.

A paired t-test is useful when comparing related cases. By averaging the differences between the scores of the paired cases, you can determine whether the total difference is statistically significant.

**Unpaired t-test**

Use this option when:

* + You have two independent samples of scores. That is, there is no basis for pairing scores in sample 1 with those in sample 2.
  + All scores within a sample are independent of all other scores within that sample.
  + The sampling distribution of x1 - x2 is normal.
  + Optionally, satisfy the requirement that the variance among the groups be roughly equal.

1. Set parameters dictated by the type of data you have and the type of t-test you chose. See the [Options](https://msdn.microsoft.com/en-us/library/azure/dn905917.aspx#bkmk_Options) section for more information.
2. Run the experiment.
3. The output of the module is a dataset containing the t-test scores, and a transformation that you can optionally save to re-apply to this or another dataset using [Apply Transformation](https://msdn.microsoft.com/en-us/library/azure/dn913055.aspx).

No matter which type of t-test you choose, the module outputs two scores:

* + A probability score that indicates the confidence of the null hypothesis
  + A value that indicates whether the Null hypothesis should be rejected

The goal is only to determine if we can reject the null hypothesis. Therefore, a score of 0 doesn’t mean we should accept it. It means that we don’t have enough data, and further investigation is needed.

### [**Options**](javascript:void(0))

The options for configuring the [Test Hypothesis Using t-Test](https://msdn.microsoft.com/en-us/library/azure/dn905917.aspx) module depend on which type of t-test you apply.

#### [Single Sample Set](javascript:void(0))

***Null hypothesized µ***

Type the value to use as the null-hypothesized mean for the sample.

In this case, specify the expected mean value against which the sample mean will be tested.

***Target column***

Use the Column Selector to choose a single numeric column for testing.

***Hypothesis type***

Choose the type of hypothesis to use when testing. The default is a two-tailed test.

***Two Tail*** (two-tailed test):

This is the most common type of test, in which the expected distribution is symmetric around zero.

***One Tail GT*** (one-tailed greater than test):

A one-tailed test provides more power to detect an effect in one direction, by not testing the effect in the other direction.

***One Tail LT*** (one-tailed less than test):

A one-tailed test provides more power to detect an effect in one direction, by not testing the effect in the other direction.

***α***

Specify a confidence factor.

If the value of P (the first output of the module) is lower than the confidence factor, the null hypothesis is rejected.

#### [Paired Sample](javascript:void(0))

Null hypothesized µ

Type a value that represents the sample difference between the pair of samples.

***Target column***

Use the Column Selector to choose a pair of numeric columns for testing.

***Hypothesis type***

Choose the type of hypothesis to use when testing. The default is a two-tailed test.

***Two Tail***.   Use a two-tailed test.

This is the most common type of test, in which the expected distribution is symmetric around zero.

***One Tail GT***.    Use a one-tailed greater than test.

A one-tailed test provides more power to detect an effect in one direction, by not testing the effect in the other direction.

***One Tail LT***.   Use a one-tailed less than test.

A one-tailed test provides more power to detect an effect in one direction, by not testing the effect in the other direction.

***α***

Specify a confidence factor.

If the value of P (the first output of the module) is lower than the confidence factor, the null hypothesis is rejected.

#### [Unpaired T-Test](javascript:void(0))

***Assume equal variances***

Deselect this option if you want the test to

***Null hypothesized µ1***

Type the value to use as the null-hypothesized mean for the first sample.

In other words, provide the expected mean value against which the sample mean will be tested.

***Null hypothesized µ2***

Type the value to use as the null-hypothesized mean for the second sample.

In other words, provide the expected mean value against which the sample mean will be tested.

***Target columns***

Use the Column Selector to choose numeric columns for testing.

***Hypothesis type***

Choose the type of hypothesis to use when testing. The default is a two-tailed test.

***Two Tail***.   Use a two-tailed test.

This is the most common type of test, in which the expected distribution is symmetric around zero.

***One Tail GT***.    Use a one-tailed greater than test.

A one-tailed test provides more power to detect an effect in one direction, by not testing the effect in the other direction.

***One Tail LT***.   Use a one-tailed less than test.

A one-tailed test provides more power to detect an effect in one direction, by not testing the effect in the other direction.

***α***

Specify a confidence factor.

If the value of P (the first output of the module) is lower than the confidence factor, the null hypothesis is rejected.

## [Technical Notes](javascript:void(0))

The module automatically names the output columns according to the following conventions, depending on which type of t-test was selected, and whether the result was to reject or accept the null hypothesis.

Given input columns with names {0} and {1}, the module creates the following names:

|  |  |  |  |
| --- | --- | --- | --- |
| **Columns** | **SingleSampleSet** | **PairedSamples** | **UnpairedSamples** |
| Output column P | P\_ss({0}) | P\_ps({0}, {1}) | P\_us({0}, {1}) |
| Output column **RejectH0** | RejectH0\_ss({0})" | RejectH0\_ps({0}, {1}) | RejectH0\_us({0}, {1}) |
|  |  |  |  |

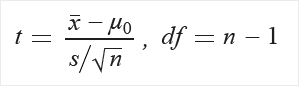
### [**How Scores are Computed**](javascript:void(0))

This module computes and uses the sample standard deviation; therefore, the equation used contains (n-1) in the denominator.

#### [Single Sample Set](javascript:void(0))

Given a single sample of scores, all independent of each other, and a normal distribution, the score is calculated as follows:

1. Take the following input:
   * A single column of values from the dataset
   * The null hypothesis (H0) parameter μ0
   * The confidence score specified by **α**
2. Extract the number of samples (n).
3. Calculate the mean of the sample data.
4. Calculate the standard deviation (s) of the sample data.
5. Calculate t and degrees of freedom (df):



1. Extract probability P from distribution table T by using t and df.

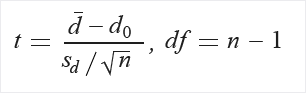
#### [Paired T-Test](javascript:void(0))

Given a matched set of scores, with each pair independent of the other, and a normal distribution in each set, the score is calculated as follows:

1. Take the following input:
   * Two columns of values from the dataset
   * The null hypothesis (H0) parameter d0
   * The confidence score specified by **α**
2. Extract some number of sample pairs (n).
3. Calculate the mean of differences for the sample data:

formula for mean of differences

1. Calculate the standard deviation of differences (sd).
2. Calculate t and the degrees of freedom (df):



1. Extract probability (P) from the distribution table (T) by using t and df.

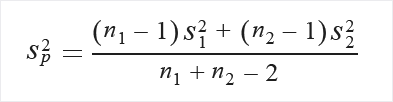
#### [Unpaired T-Test](javascript:void(0))

Given two independent samples of scores, with a normal distribution of values in each sample, the score is calculated as follows:

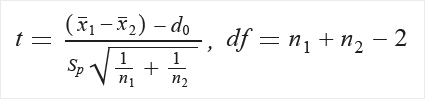
1. Take the following input:
   * A dataset that contains two columns of **doubles**
   * The null hypothesis (H0) parameter (d0)
   * The confidence score specified by **α**
2. Extract a number of samples in each group, n1 and n2.
3. Calculate the means for each of the sample sets.
4. Calculate the standard deviation for each group as s1 and s2.
5. Calculate t and degrees of freedom (df):

Optionally, satisfy the requirement that the variance among the groups be roughly equal, as follows:

* + Calculate the pooled standard deviation first:



* + If there is no assumption about variance equality, calculate as follows:



1. Extract P from the distribution table (T) by using t and df.

#### [Computing Null Hypothesis](javascript:void(0))

The probability of the null hypothesis, designated as P, is calculated as follows:

* If P < α, set the Reject flag to True.

If P ≥ α, set the Reject flag to False.

1. Compute Elementary Statistic

## [Module Overview](javascript:void(0))

You can use the [Compute Elementary Statistics](https://msdn.microsoft.com/en-us/library/azure/dn905936.aspx) module to generate a summary report for your dataset that lists key statistics such as mean, standard deviation, and the range of values for each of the selected columns.

This report is useful for analyzing the central tendency, dispersion, and shape of data.

## [How to Use the Compute Elementary Statistics module](javascript:void(0))

1. Click the **Method** dropdown list, and choose the type of value that you want to calculate for each column.

See the [Options](https://msdn.microsoft.com/en-us/library/azure/dn905936.aspx#bkmk_Options) section for a full list of available statistics and what they mean.

1. Optionally, launch the column selector to pick the numeric columns for which you want a report. All columns that you choose must be numeric.

By default, the value you selected in **Method** dropdown list will be calculated for all columns in the dataset that have a numeric data type. If any column has values that prevent the value from being calculated, an error will be raised and the report will not be created.

1. Run the experiment.

The generated report provides the name of each column and the statistic that was calculated.

For example, the following table shows statistics generated for the **mpg** column.

Each time you run [Compute Elementary Statistics](https://msdn.microsoft.com/en-us/library/azure/dn905936.aspx) it generates only a single value for each of the selected columns, but you can use the[Summarize Data](https://msdn.microsoft.com/en-us/library/azure/dn905933.aspx) module to merge the results into a single table as shown here.

|  |  |  |
| --- | --- | --- |
| **DeviationSquared(mpg)** | **Max(mpg)** | **Min(mpg)** |
| 9674.312 | 25.21951 | 13 |

### [**Options**](javascript:void(0))

This module supports the following standard descriptive statistics.

**Deviation squared**

Calculates the squared deviation of the column values. Also known as the sum of squares.

Squared deviation is a measure of how far values are dispersed from the mean.

**Geometric mean**

Calculates the geometric mean of the column values.

The geometric mean can be used to measure the central tendency of a set of numbers. Compared to the arithmetic mean, it is less affected by a small number of extreme values. It can also be used to compare measurements on different scales, since it effectively normalizes the scales of the numbers being compared. The geometric means is sometimes used to estimate compound annual growth rates.

The equivalent function in Excel is GEOMEAN.

**Harmonic mean**

Calculates the harmonic mean of the column values.

To compute the harmonic mean, all values are converted to their reciprocals, and then the mean is taken of those values. The harmonic mean is the reciprocal of that mean. If the column values are positive, larger numbers are weighted less than smaller numbers.

The harmonic mean is always less than the geometric mean, which is always less than the arithmetic mean. The harmonic mean is useful for averaging variables that represent rates, such as speed (distance over time) or sales per quarter.

The equivalent function in Excel is HARMEAN.

**Interquartile distance**

Calculates the interquartile difference for the first and the last quartiles of the column values. Also called the quartile range. When the quartile falls between two numbers, the quartile value is the average of the two values on either side of the cut.

The quartile value divides the column of values into four groups with an equal number of values. Thus, one quarter of the values are less than or equal to the 25th percentile. Three quarters of the values are less than or equal to the 75th percentile. By reviewing the quartile range you can get an idea of how widely spread the data values are.

**K-th central moment**

Calculates K-th central moment for the column values.

When calculating K-th central moment, you must also specify the **Order**, meaning the value of k. The value of k can range from 0 to any allowed integer value, though higher order values are generally not meaningful.

Generally, in descriptive statistics, a moment is a measure that describes the shape of a set of points. Central moments are moments about the mean, which are usually used because they provide better information about the distribution's shape. An order of 2 usually represents the variance; an order of 4 is used for kurtosis. The first order moment is the mean. Thus the collection of all moments uniquely describes the distribution of values in the column.

**Max**

Finds the maximum value in the column.

**Mean**

Calculates the arithmetic mean of the column values.

The equivalent function in Excel is AVERAGE.

**Mean deviation**

Calculates the mean absolute deviation for the column values.

That is, the mean is computed for the column, and the deviation computed for each value in the column. The average of the absolute values of the individual deviation values is the mean deviation.

This statistic tells you how spread out from the mean your column of numbers is.

**Median**

Returns the median of the column values.

The median is the number in the middle of a column of numbers. If there is an even number of numbers in the column, the median is the average of the two numbers in the middle.

The median, together with the mean and the mode, is one of three statistics that measures central tendency. If the values are symmetrical around the mean, the three numbers will be about the same. However, the median is more robust to outliers than the mean.

**Median deviation**

Calculates the median deviation for the column.

That is, the median is computed for the column, and the deviation computed for each value in the column. The median value of the absolute values of the individual deviation values is taken.

The median absolute deviation is also known as MAD, and is used to describe the variability of a sample of numbers. MAD tells you how spread out from the mean your column of numbers is.

**Min**

Returns the minimum value of the column values.

**Mode**

Finds all modes for the column.

The mode is the value that appears the most in the column. If several values appear the same number of times, the column can have multiple modes.

As a measure of central tendency, mode is more robust to outliers than the mean, and can be used with nominal data too.

**Population standard deviation**

Calculates the population standard deviation for the column values.

This statistic assumes that the column values represent the entire population. If your data is only a sample of the population, you must compute the standard deviation by using **Sample standard deviation**. However, in large datasets, the two statistics return approximately equal values

The standard deviation is computed as the square root of the column variance. This statistic captures the amount of variability in the column.

**Population variance**

Calculates the population variance for the column values.

Variance measures how much a set of numbers is spread out. If variance is zero, all numbers are the same.

This statistic assumes that the column of values represents the entire population. If your data contains only a sample of the values, you should compute variance by using **Sample variance**.

The equivalent Excel function is VAR.P.

**Product**

Calculates the product of the column's elements.

To get the product, you multiple all the numbers in the column. The result is not in itself useful as a descriptive statistic but the function is useful for a variety of other calculations.

**Range**

Calculates the range of the column values. The range is defined as the maximum value minus the minimum value.

**Sample kurtosis**

Calculates the sample kurtosis for the column values.

Kurtosis describes the shape of the distribution of values-- that is, how peaked or flat the distribution of values is, compared with the normal distribution.

* The normal distribution has a kurtosis of 0.
* High kurtosis values indicate that the probability mass is concentrated either around a peak, or in the tail of the distribution.
* Negative kurtosis values indicate a relatively flat distribution.

**Sample skewness**

Calculates the sample skewness for the column values.

Skew describes whether the bulk of the values are at the center, shifted to the left, or shifted to the right. Two distributions might have the same mean and standard deviation, yet be shaped very differently. You can use skewness and kurtosis to characterize the shape.

* Negative skew values means the distribution is skewed to the left.
* 0 denotes the normal distribution.
* Positive skewness values mean the distribution is skewed to the right.

**Sample standard deviation**

Calculates the sample standard deviation for the column values.

The standard deviation of the sample measures how spread out the values in the column are from the mean. It represents the average distance between the values of the data in the set and the mean.

This statistic assumes that the column values represent a sample of the population. If your data represents the entire population, you must compute the standard deviation using **Population standard deviation**.

The equivalent Excel function is ST.DEV.S.

**Sample variance**

Calculates the sample variance for the column values.

This method assumes that the column values represent a sample of the population. If the column contains the entire population, you should use **Population standard variance**.

The equivalent Excel function is VAR.S.

**Sum**

Calculates the sum of the column values.

## [Examples](javascript:void(0))

There are currently no examples that use this module. However, the following experiments in the [Model Gallery](http://gallery.azureml.net/) demonstrate how you can create a summary report that contains descriptive statistics for an entire dataset.

After reviewing the summary report, you might save it as a dataset and then add more detailed statistics, using the options in [Compute Elementary Statistics](https://msdn.microsoft.com/en-us/library/azure/dn905936.aspx) .

* In the [Download dataset from UCI](http://gallery.azureml.net/Experiment/24c4e869c5c448958ce923c2e2bfbb27) sample, the [Summarize Data](https://msdn.microsoft.com/en-us/library/azure/dn905933.aspx) module is used to generate a summary report on all columns in the dataset.
* In the [Dataset Processing and Analysis](http://gallery.azureml.net/Experiment/943c3d4becb7470e8acac6af699d6ea9) sample, the [Summarize Data](https://msdn.microsoft.com/en-us/library/azure/dn905933.aspx) module is used to generate a summary report on all columns in the dataset.

## [Technical Notes](javascript:void(0))

The following conditions must be satisfied when using the [Compute Elementary Statistics](https://msdn.microsoft.com/en-us/library/azure/dn905936.aspx) module:

* There must be a sufficient number of data points (rows) to compute the selected statistic. For example, to compute **Sample standard deviation** requires at least two data points; otherwise, the result is NaN.
* Input columns must be numeric or Boolean.
* Boolean columns are processed as follows:
  + MIN is computed as logical AND.
  + MAX is computed as logical OR.
  + RANGE checks whether the number of unique values in the column equals 2.
* Missing values are ignored.
* For statistics that require floating-point calculations, True = 1.0 and False = 0.0
* By default, all numeric columns are selected. However, if any numeric columns are marked as categorical, you might get the following error.

Error 0056: Column with name <column name> is not in an allowed category.

To correct the error, add an instance of the [Edit Metadata](https://msdn.microsoft.com/en-us/library/azure/dn905986.aspx) module, select the column with the problem, and use the option **Remove categorical**.