Ponderosa Computing

Statistical Distributions Excel Add-ins

Paul J. McClellan, Ph.D.

7 July 2020

Table of Contents

1 Ponderosa Computing Statistical Distributions Excel Add-ins

The purpose of the Ponderosa Computing Statistical Distributions Excel Add-ins are to provide alternative statistical (probability) distribution functions to those provided as built-in Excel functions. These distribution functions:

1. Provide explicit probability mass or density functions and lower and upper tail probability and quantile functions.
2. Treat distribution function parameters as exact, rather than truncating them to integer values.
3. Support full random variate domains without implicit truncation of values.
4. Use peer-reviewed and publicly documented algorithms with error estimates.

The Ponderosa Computing Statistical Distributions Excel Add-ins support Microsoft Windows 7, 8, and 10. There are two Add-ins:

- PcStaDistXLL_32.xll supports 32-bit Excel 2007 and later.
- PcStaDistXLL_64.xll supports 64-bit Excel 2010 and later.

Each has a separate installer package available for download from Ponderosa Computing.

This document describes version 1.4 of the Ponderosa Computing Statistical Distributions Excel Add-ins.

2 Statistical Distribution Functions

The Ponderosa Computing Statistical Distributions Excel Add-ins provide the following statistical distribution functions:

Distribution	PMF/PDF	LTP	UTP	LTQ	UTQ
Beta	SD.BETA.PDF	SD.BETA.LTP	SD.BETA.UTP	SD.BETA.LTQ	SD.BETA.UTQ
Standard Beta	SD.SBETA.PDF	SD.SBETA.LTP	SD.SBETA.UTP	SD.SBETA.LTQ	SD.SBETA.UTQ
Binomial	SD.BINOM.PMF	SD.BINOM.LTP	SD.BINOM.UTP	SD.BINOM.LTQ	SD.BINOM.UTQ
Cauchy	SD.CAUCHY.PDF	SD. CAUCHY.LTP	SD. CAUCHY.UTP	SD. CAUCHY.LTQ	SD. CAUCHY.UTQ
Chi-Squared	SD.CHISQ.PDF	SD.CHISQ.LTP	SD.CHISQ.UTP	SD.CHISQ.LTQ	SD.CHISQ.UTQ
Exponential	SD.EXPON.PDF	SD.EXPON.LTP	SD.EXPON.UTP	SD.EXPON.LTQ	SD.EXPON.UTQ
Maximum Extreme Value	SD.EVMAX.PDF	SD. EVMAX.LTP	SD. EVMAX.UTP	SD. EVMAX.LTQ	SD. EVMAX.UTQ
Mininim Extreme Value	SD.EVMIN.PDF	SD. EVMIN.LTP	SD. EVMIN.UTP	SD. EVMIN.LTQ	SD. EVMIN.UTQ
Gamma	SD.GAMMA.PDF	SD.GAMMA.LTP	SD.GAMMA.UTP	SD.GAMMA.LTQ	SD.GAMMA.UTQ
Geometric	SD.GEOM.PMF	SD.GEOM.LTP	SD.GEOM.UTP	SD.GEOM.LTQ	SD.GEOM.UTQ
Hypergeometric	SD.HGEOM.PMF	SD.HGEOM.LTP	SD.HGEOM.UTP	SD.HGEOM.LTQ	SD.HGEOM.UTQ
Logistic	SD.LOGISTIC.PDF	SD.LOGISTIC.LTP	SD.LOGISTIC.UTP	SD.LOGISTIC.LTQ	SD.LOGISTIC.UTQ
Lognormal	SD.LNORM.PDF	SD.LNORM.LTP	SD.LNORM.UTP	SD.LNORM.LTQ	SD.LNORM.UTQ
Negative Binomial	SD.NBINOM.PMF	SD.NBINOM.LTP	SD.NBINOM.UTP	SD.NBINOM.LTQ	SD.NBINOM.UTQ
Normal	SD.NORM.PDF	SD.NORM.LTP	SD.NORM.UTP	SD.NORM.LTQ	SD.NORM.UTQ
Standard Normal	SD.SNORM.PDF	SD.SNORM.LTP	SD.SNORM.UTP	SD.SNORM.LTQ	SD.SNORM.UTQ
Poisson	SD.POISSON.PMF	SD.POISSON.LTP	SD.POISSON.UTP	SD.POISSON.LTQ	SD.POISSON.UTQ
Snedecor F	SD.F.PDF	SD.F.LTP	SD.F.UTP	SD.F.LTQ	SD.F.UTQ
Student's t	SD.T.PDF	SD.T.LTP	SD.T.UTP	SD.T.LTQ	SD.T.UTQ
Weibull	SD.WEIBULL.PDF	SD.WEIBULL.LTP	SD.WEIBULL.UTP	SD.WEIBULL.LTQ	SD.WEIBULL.UTQ

For each statistical distribution these Excel Add-ins provide either a probability mass function (PMF) or a probability density function (PDF), depending on whether the distribution is discrete or continuous, respectively. For each statistical distribution these add-ins also provide explicit lower tail probability (LTP), upper tail probability (UTP), lower tail quantile (LTQ), and upper tail quantile (UTQ) worksheet functions.

2.1 Probability Mass or Density Functions

If a random variable X can take on a continuous range of values, it is called a **continuous random variable** and has a **continuous distribution**. The probability of observing a value in a range of values is defined by the **probability density function pdf(x, A)**, where x represents a specific random value and the parameter A represents one or more fixed distribution parameters. This density function is integrated over a range of possible values to determine the probability of observing the random variable within that range.

If a random variable X can only take on one of a countable set of different values, it is called a **discrete random variable** and has a **discrete distribution**. The probability of observing any specific random value is defined by a **probability mass function pmf(x, A)**, where x is the specific random value and the parameter A represents one or more fixed distribution parameters. In the case of all discrete distributions supported by this add-in, the discrete random variable is restricted to integer values.

For any random variable X the functions pdf(x, A) and pmf(x, A) are defined for values x in the domain of **real numbers** $\mathbb{R} = (-\infty, +\infty)$. To describe limiting behavior of distribution functions it is useful to define these functions for values x in the **affinely extended real numbers** $[-\infty, +\infty]$, which adds the elements $+\infty$ (positive infinity) and $-\infty$ (negative infinity) to the real numbers.

2.2 Lower and Upper Tail Probability Functions

The distribution of a discrete or continuous random variable X can equivalently be defined by the **cumulative distribution function cdf(x, A)**, defined as the probability of observing a value of X such that $X \leq x$. Since statistical tests are typically based on the probability of observing a random variable in the lower or upper extremes, or tails, of possible values, this function is called the **lower tail probability function ltp(x, A)** in that context.

The complement of the cumulative distribution function ccd(x, A) is defined as 1 – cdf(x, A), or the probability of observing a value of X such that $X > x$. This function is also called the survival function or the reliability function, depending on context. This function is called the **upper tail probability function utp(x, A)** in the context of statistical tests.

For any random variable X the functions ltp(x, A) and utp(x, A) are defined for values x in the domain of **real numbers** $\mathbb{R} = (-\infty, +\infty)$. To describe limiting behavior of distribution functions it is useful to define these functions for values x in the **affinely extended real numbers** $[-\infty, +\infty]$, which adds the elements $+\infty$ (positive infinity) and $-\infty$ (negative infinity) to the real numbers.

2.3 Quantile Functions

To define the critical values for statistical tests or to compute confidence bounds and intervals for parameter estimates we compute the **quantile function** of a probability distribution.

From Wikipedia[10]:

*In probability and statistics, the **quantile function**, associated with a probability distribution of a random variable, specifies the value of the random variable such that the probability of the variable being less than or equal to that value equals the given probability. It is also called the **percent-point function** or **inverse cumulative distribution function**.*

The quantile function is one way of prescribing a probability distribution, and it is an alternative to the probability density function (pdf) or probability mass function, the cumulative distribution function (cdf) and the characteristic function. The quantile function, Q, of a probability distribution is the inverse of its cumulative distribution function F.

The **quantile function** of a random variable X having distribution parameters A is represented here as **Q(p, A)**, where the probability value is defined over [0, 1]. The value of the quantile function q = Q(p, A) is the value q such that the probability of observing the random variable X having a value of **q or less** is the specified probability p:

$$\text{Probability}(X \leq q, A) = \text{cdf}(q, A) = p$$

Actually, the definition of the quantile function above is appropriate only for a random variable X having a continuous and strictly increasing cumulative distribution function cdf(x, A). For such a random variable (defined over [-∞, +∞]) and a probability level p (defined over [0, 1]), the quantile function Q(p, A) for this random variable returns the **unique** value q such that cdf(q, A) = p:

$$q = Q(p, A) = \{x \text{ in } [-\infty, +\infty]: \text{cdf}(x, A) = p\}.$$

Hence in this case the quantile function is the inverse cumulative distribution function.

For a discrete random variable X having support on the integer values {m, …, n} the cumulative distribution function cdf(x, A) is discontinuous with flat sections separated by nonzero jumps. Hence there may not exist a value q such that for a specific probability p we have cdf(q, A) = p, or if such a value does exist it may not be unique. We will modify the quantile definition above according to the context in which it is used below.

2.3.1 Lower Tail Quantile Function

The quantile function is directly used to compute the critical value for a lower tail test of a probability distribution, or to compute the lower confidence bound of a parameter estimate. The quantile function probability level p is set to a suitable small probability level representing statistical significance (alpha) and the critical values is computed as q = Q(alpha, A).

To emphasize the direct use of the quantile function Q(p, A) in the context of lower tail tests or lower confidence bounds the quantile function is called here the **lower tail quantile function ltq(p, A)**. Similarly, the cumulative distribution function is also called the **lower tail probability function ltp(x, A)**.

Restating from above, for a random variable X having a continuous and strictly increasing cumulative distribution function cdf(x, A) the lower tail quantile function ltq(p, A) returns the **unique** value q such that ltp(q, A) = p:

$$q = \text{ltq}(p, A) = \{x \text{ in } [-\infty, +\infty]: \text{ltp}(x, A) = p\}$$

so the lower tail quantile function is the inverse lower tail probability function.

The value of the lower tail quantile function q = ltq(p, A) is the unique value q such that the probability of observing the random variable X having a value of **q or less** is the specified probability p:

$$\text{Probability}(X \leq q, A) = \text{ltp}(q, A) = p.$$

For a discrete random variable X having support on the integer values {m, …, n} the lower tail probability function ltp(x, A) is discontinuous with flat sections separated by nonzero jumps. Hence there may not exist a value q such that for a specific probability p we have ltp(q, A) = p, or if such a value does exist it may not be unique. So we need to more generally define the lower tail quantile function for discrete random variables.

In constructing statistical tests at probability level p = alpha, or in computing lower confidence bounds, we wish to be conservative and require any computed critical value q to satisfy

$$\text{Probability}(X \leq q, A) = \text{ltp}(q, A) \leq p$$

Within this constraint, we wish to maximize the sensitivity (or power) of statistical tests and tighten the lower confidence bound by maximizing the critical value q.

For a discrete random variable X having support on the integer values {m, …, n} we more generally define the lower tail quantile function ltq(p, A) as the largest value q in {m-1, …, n} for which the probability of observing the random variable X having a value of **q or less** is **at most p**:

$$q = \text{ltq}(p, A) = \max\{x \text{ in } \{m-1, \ldots, n\}: \text{ltp}(x, A) \leq p\}.$$

This definition provides the limiting cases ltq(0, A) = m-1 and ltq(1, A) = n.

The lower tail quantile function illustrates the utility of defining the domain of the lower tail probability function over the affinely extended real numbers. If the specified probability p is less than the probability of the smallest possible value m of a discrete random variable X, ltq(p, A) returns the value q = m-1, and the relationship $0 = \text{ltp}(q, A) \leq p$ is preserved.

2.3.2 Upper Tail Quantile Function

The complement of the cumulative distribution function ccd(x, A) of a random variable X having distribution parameters A is the probability of observing a value of X such that X >x. This function is called the **survival function** or the **reliability function**, depending on context, and is called the **upper tail probability function utp(x, A)** in the context of statistical tests.

For a random variable X (defined over [-∞, +∞]) having a continuous and strictly increasing cumulative distribution function cdf(x, A), the **upper tail quantile function utq(p, A)** returns the unique value q such that utp(q, A) = p:

$$q = \text{utq}(p, A) = \{x \text{ in } [-\infty, +\infty]: \text{utp}(x, A) = p\}$$

so the upper tail quantile function is the inverse upper tail probability function (inverse survival function).

The value of the upper tail quantile function q = utq(p, A) is the value q such that the probability of observing the random variable X having a value of **exceeding q** is the specified probability p:

$$\text{Probability}(X > q, A) = \text{utp}(q, A) = p$$

The upper tail quantile function is directly used to compute the critical value for an upper tail test of a probability distribution, or to compute the upper confidence bound of a parameter estimate. The upper tail quantile function probability level p is set to a suitable small probability level representing statistical significance (alpha) and the critical values is computed as q = utq(alpha, A).

For a discrete random variable X having support on the integer values {m, …, n} the upper tail probability function utp(x, A) is discontinuous with flat sections separated by nonzero jumps. Hence there may not exist a value q such that for a specific probability p we have utp(q, A) = p, or if such a value does exist it may not be unique. So, as in the case of the lower tail quantile function, we need to more generally define the upper tail quantile function for discrete random variables.

In constructing statistical tests at probability level p = alpha, or in computing upper confidence bounds, we wish to be conservative and require any computed critical value q to satisfy

$$\text{Probability}(X > q, A) = \text{utp}(q, A) \leq p$$

Within this constraint, we wish to maximize the sensitivity (or power) of statistical tests and tighten the upper confidence bound by minimizing the critical value q.

For a discrete random variable X having support on the integer values{m, …, n} we more generally define the upper tail quantile function utq(p, A) as the smallest value q in {m-1, …, n} for which the probability of observing the random variable X having a value **exceeding q** is **at most p**:

$$q = \text{utq}(p, A) = \min\{x \text{ in } \{m-1, \ldots, n\}: \text{utp}(x, A) \leq p\}.$$

This definition provides the limiting cases utq(0, A) = n and utq(1, A) = m-1.

The upper tail quantile function illustrates the utility of defining the domain of the upper tail probability function over the affinely extended real numbers. If the specified probability p is less than the probability of the largest possible value n of a discrete random variable X, utq(p, A) returns the value $q = n$, and the relationship $0 = \text{utp}(q, A) \leq p$ is preserved.

3 Computational Methods and Numerical Precision

Microsoft improved the accuracy of Excel's statistical distribution functions in Excel 2010 over previous Excel versions in response to various academic papers over the years detailing accuracy issues in Excel [1][2][3]. In doing so, Microsoft implemented completely new algorithms for these functions, partnering with Frontline Systems, the Numerical Algorithms Group, and ScienceOps to select, implement, and validate these algorithms [3]. Microsoft also changed function names in Excel 2010 to improve the consistency of the function library [4].

However, these new algorithms and their error estimates have not been fully and publicly disclosed, and consequently users of Excel 2010 and subsequent Excel versions do not know the accuracy limitations of the build-in Excel statistical distribution functions.

3.1 Uses the Boost C++ Math Toolkit Implementation

The Ponderosa Computing Statistical Distributions Excel Add-ins use the **Boost C++ Math Toolkit version 2.9.0** [8], part of the Boost C++ Libraries version 1.70.0 provided by Boost.org [7], as the computational engine of its statistical distributions functions. The Boost C++ Math Toolkit is a peer-reviewed, open source library under continuing development and testing.

The Ponderosa Computing Statistical Distributions Excel Add-ins modified the Boost C++ Math Toolkit file boost_1_70_0\boost\math\tools\user.hpp to change the default policies:

```
// Pole errors:
//
// #define BOOST_MATH_POLE_ERROR_POLICY throw_on_error
#define BOOST_MATH_POLE_ERROR_POLICY ignore_error
//
// Overflow Errors:
//
// #define BOOST_MATH_OVERFLOW_ERROR_POLICY throw_on_error
#define BOOST_MATH_OVERFLOW_ERROR_POLICY ignore_error
//
// Internal Evaluation Errors:
//
// #define BOOST_MATH_EVALUATION_ERROR_POLICY throw_on_error
#define BOOST_MATH_EVALUATION_ERROR_POLICY ignore_error
```

This change allows propagation of nonfinite values through the statistical distribution computations.

3.2 Distribution Parameters are Assumed Exact

Each of the statistical distributions is indexed by one or more parameters A describing the likelihood of observing a value or range of values of a random variable X defined by that distribution. Since the distributions are defined by these parameters the Ponderosa Computing

Statistical Distributions Excel Add-in worksheet functions assume these distribution parameters are provided exactly, and the functions do not truncate or round distribution parameters.

If a particular distribution function is not defined for some parameter value A, the Ponderosa Computing Statistical Distributions Excel Add-in worksheet function returns the #NUM! error value. This is in contrast to some built-in Excel statistical functions [5], such as BINOM.DIST(), which truncate integer-defined distribution parameter values to integers. These Excel Add-ins include some front-end checks to implement the expanded argument ranges for its distribution functions.

3.3 Computations use IEEE 754 Double Precision Format

The Ponderosa Computing Statistical Distributions Excel Add-ins implement their worksheet functions using the double precision format defined by the IEEE Standard 754 for Binary Floating-Point Arithmetic [9]. This includes supporting signed zeros and normalized, denormalized, and nonfinite (infinite and indeterminate) numbers. Excel, however, does not fully support this format [6]. This limits the range of argument and return values for add-in worksheet functions.

3.3.1 Worksheet Function Support of Nonfinite Numbers

The Ponderosa Computing Statistical Distributions Excel Add-in worksheet functions PMF, PDF, LTP and UTP support nonfinite argument values. However, Excel does not support nonfinite arguments to functions, so in practice these worksheet functions will not encounter nonfinite values of x.

For some values of p the quantile functions ltq(p, A) and utq(p, A) may logically return nonfinite values. Ponderosa Computing Statistical Distributions Excel Add-in worksheet functions LTQ and UTQ support such return values. However, Excel does not support nonfinite function return values, so the Excel Add-in worksheet functions replace nonfinite results with the #NUM! error value. For example, while the logical return value of the standard normal ltq(1.0) is $+\infty$, the add-in worksheet function SD.SNORM.LTQ(1.0) returns the #NUM! error value.

3.3.2 Worksheet Function Finite Return Values May be Modified by Excel

Excel does not support IEEE 754 denormalized numbers. For example, Excel truncates the return value of SD.SNORM.PDF(37.7) to 0.0, even though this add-in worksheet function returns a tiny nonzero value to Excel as its return value.

The Ponderosa Computing Statistical Distributions Excel Add-ins and Excel internally support normalized IEEE 754 double precision numbers with 53 binary digits of precision (equivalent to nearly 16 decimal digits of precision). However, Excel displays at most 15 decimal digits of precision in numeric values [6].

4 Worksheet Functions

For any random variable X the probability mass function pmf(x, A), the probability density function pdf(x, A), and the lower and upper tail probability functions ltp(x, A) and utp(x, A) are defined for values x in the domain of affinely extended real numbers [-∞, +∞]. Also, for some values of p the lower and upper tail quantile functions ltq(p, A) and utq(p, A) may logically return nonfinite values.

The Ponderosa Computing Statistical Distributions Excel Add-in PMF, PDF, LTP, and UTP functions are defined for values x in [-∞, +∞] and the LTQ and UTQ functions may return nonfinite values to Excel. However, Excel does not support nonfinite values, either as arguments to functions or function return values and will convert nonfinite return values to the #NUM! error value.

These Excel Add-ins use the **Boost C++ Math Toolkit version 2.9.0** [8], part of the Boost C++ Libraries version 1.70.0 provided by Boost.org [7], as the computational engine of its statistical distributions functions. We provide links to the Boost C++ Math Toolkit documentation for distribution function implementation and accuracy details.

These Excel Add-ins augment the Boost C++ Math Toolkit implementation with special case handling where needed to achieve the extended worksheet function domains and return values.

4.1 Continuous Distributions

However, Excel does not support nonfinite values, either as arguments to functions or function return values. For example, while the logical return value of SD.SNORM.LTQ(1.0) is +∞, Excel converts this function's return value to the #NUM! error value.

4.1.1 Beta Distribution

The **beta distribution** is a continuous distribution with probability density function:

$$\mathrm{pdf}(x;\alpha,\beta,\mathrm{a},\mathrm{b}) = \frac{(x-a)^{(\alpha-1)}(b-x)^{(\beta-1)}}{B(\alpha,\ \beta)(b-a)^{(\alpha+\beta-1)}}\,,\qquad a \le x \le b;\ \alpha,\beta > 0$$

$$\mathrm{pdf}(x;\alpha,\beta,\mathrm{a},\mathrm{b}) = 0\,,\qquad x < a\ or\ x > b;\ \alpha,\beta > 0$$

Here $B(\alpha,\beta)$ is the beta function:

$$B(\alpha,\beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}dt$$

These Excel Add-ins use the Boost C++ Math Toolkit implementation of the standard beta distribution, followed by location and scaling adjustments.

The beta distribution shape parameters alpha (α) and beta (β) and domain parameters a and b match the *alpha*, *beta*, *a*, and *b* parameters used by these Excel Add-ins and the Microsoft Excel BETA.DIST() and BETA.INV() built-in functions with domain parameters *A* = *a* and *B* = *b*.

The shape parameters *alpha* and *beta* must be positive and the domain parameters must satisfy *a* < *b* or the Excel Add-in beta distribution worksheet functions will return the #NUM! error value.

4.1.1.1 SD.BETA.PDF()

The worksheet function SD.BETA.PDF(*x*, *alpha*, *beta*, *a*, *b*) returns the value of the beta distribution probability density function for values *x* in the domain of affinely extended real numbers. This function returns zero for values *x* < *a* or *x* > *b*.

SD.BETA.PDF(*x*, *alpha*, *beta*, *a*, *b*) is an alternative to the Excel BETA.DIST(*x*, *alpha*, *beta*, *FALSE*, *a*, *b*) built-in function.

4.1.1.2 SD.BETA.LTP(), SD.BETA.UTP()

The worksheet functions SD.BETA.LTP(*x*, *alpha*, *beta*, *a*, *b*) and SD.BETA.UTP(*x*, *alpha*, *beta*, *a*, *b*) return the beta distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.BETA.LTP(*x*, *alpha*, *beta*, *a*, *b*) is an alternative to the Excel BETA.DIST(*x*, *alpha*, *beta*, *TRUE*, *a*, *b*) built-in function.

4.1.1.3 SD.BETA.LTQ(), SD.BETA.UTQ()

The worksheet functions SD.BETA.LTQ(*probability*, *alpha*, *beta*, *a*, *b*) and SD.BETA.UTQ(*probability*, *alpha*, *beta*, *a*, *b*) return the beta distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.BETA.LTQ(*probability*, *alpha*, *beta*, *a*, *b*) is an alternative to the Excel BETA.INV(*probability*, *alpha*, *beta*, *a*, *b*) built-in function.

4.1.2 Standard Beta Distribution

The **standard beta distribution** is the special case of the beta distribution with domain parameters a = 0 and b = 1 and probability density function:

$$\text{pdf}(x; \alpha, \beta) = \frac{x^{(\alpha-1)}(1-x)^{(\beta-1)}}{B(\alpha,\ \beta)}\ , \qquad 0 \le x \le 1;\ \alpha, \beta > 0$$

$$\text{pdf}(x; \alpha, \beta) = 0\ , \qquad x < 0 \ or\ x > 1;\ \alpha, \beta > 0$$

Here $B(\alpha, \beta)$ is the beta function:

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}dt$$

The Boost C++ Math Toolkit implementation and accuracy of the standard beta distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/beta_dist.html

The beta distribution shape parameters alpha (α) and beta (β) match the *alpha* and *beta* parameters used by these Excel Add-ins and the Microsoft Excel BETA.DIST() and BETA.INV() built-in functions with optional domain parameters *A* = 0 and *B* = 1.

The shape parameters *alpha* and *beta* must be positive or the Excel Add-in standard beta distribution worksheet functions will return the #NUM! error value.

4.1.2.1 SD.SBETA.PDF()

The worksheet function SD.SBETA.PDF(*x*, *alpha*, *beta*) returns the value of the standard beta distribution probability density function for values x in the domain of affinely extended real numbers. This function returns zero for values *x* < 0 or *x* > 1.

SD.SBETA.PDF(*x*, *alpha*, *beta*) is an alternative to the Excel BETA.DIST(*x*, *alpha*, *beta*, *FALSE*) built-in function.

4.1.2.2 SD.SBETA.LTP(), SD.SBETA.UTP()

The worksheet functions SD.SBETA.LTP(*x*, *alpha*, *beta*) and SD.SBETA.UTP(*x*, *alpha*, *beta*) return the standard beta distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.SBETA.LTP(*x*, *alpha*, *beta*) is an alternative to the Excel BETA.DIST(*x*, *alpha*, *beta*, *TRUE*) built-in function.

4.1.2.3 SD.SBETA.LTQ(), SD.SBETA.UTQ()

The worksheet functions SD.SBETA.LTQ(*probability*, *alpha*, *beta*) and SD.SBETA.UTQ(*probability*, *alpha*, *beta*) return the standard beta distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.SBETA.LTQ(*probability*, *alpha*, *beta*) is an alternative to the Excel BETA.INV(*probability*, *alpha*, *beta*) built-in function.

4.1.3 Cauchy Distribution

The **Cauchy (or Cauchy-Lorentz) distribution** is a continuous distribution with probability density function:

$$pdf(x;\ t, s) = \frac{1}{s\,\pi\,(1 + ((x - t)/s)^2)}, \qquad s > 0$$

Here t is the location parameter and s is the scale parameter.

The mean and standard deviation of the Cauchy distribution are undefined. The Cauchy probability distribution appears similar to the normal probability distribution function, but has “heavier” tails. Hypothesis tests based on normal distribution assumptions can be tested for robustness against heavy-tail departures from normality using Cauchy distribution random variables.

The Boost C++ Math Toolkit implementation and accuracy of the Cauchy-Lorentz distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/cauchy_dist.html

The Cauchy distribution location (*t*) and scale (*s*) parameters match the *location* and *scale* parameters used by these Excel Add-ins. There are no Microsoft Excel built-in Cauchy distribution functions.

The *scale* parameter must be greater than zero or the Excel Add-in Cauchy distribution worksheet functions will return the #NUM! error value.

4.1.3.1 SD.CAUCHY.PDF()

The worksheet function SD.CAUCHY.PDF(*x, location, scale*) returns the value of the Cauchy distribution probability density function for values *x* in the domain of affinely extended real numbers.

4.1.3.2 SD.CAUCHY.LTP(), SD.CAUCHY.UTP()

The worksheet functions SD.CAUCHY.LTP(*x, location, scale*) and SD.CAUCHY.UTP(*x, location, scale*) return the Cauchy distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

4.1.3.3 SD.CAUCHY.LTQ(),SD.CAUCHY.UTQ()

The worksheet functions SD.CAUCHY.LTQ(*probability, location, scale*) and SD.CAUCHY.UTQ(*probability, location, scale*) return the Cauchy distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

4.1.4 Chi-Squared Distribution

The **Chi-Squared distribution** is a special case of the gamma distribution with μ = 0, θ = 1 and κ = ν/2. Its probability density function is then:

$$pdf(x;\nu) = \frac{x^{(\frac{\nu}{2}-1)} e^{-\frac{x}{2}}}{2^{\frac{\nu}{2}}\,\Gamma(\frac{\nu}{2})}\,, \qquad x > 0;\ \nu > 0$$

$$pdf(x;\nu) = 0\,, \qquad x \le 0;\ \nu > 0$$

Here ν is the shape parameter and specifies the **number of degrees of freedom** of the Chi-Squared distribution. In the context of statistical tests ν is a positive integer.

Here $\Gamma(\kappa)$ is the gamma function:

$$\Gamma(\kappa) = \int_0^{\infty} t^{\kappa-1} e^{-t} dt$$

The Boost C++ Math Toolkit implementation and accuracy of the Chi-Squared distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/chi_squared_dist.html

The degrees of freedom parameter (ν) matches the *deg_freedom* parameter used by these Excel Add-ins and the Microsoft Excel CHISQ.DIST(), CHISQ.DIST.RT(), CHISQ.INV(), and CHISQ.INV.RT() built-in functions, except that in these Excel Add-ins *deg_freedom* can be any positive number and is not truncated.

The *deg_freedom* parameter must be greater than zero or the Excel Add-in Chi-Squared distribution worksheet functions will return the #NUM! error value.

4.1.4.1 SD.CHISQ.PDF()

The worksheet function SD.CHISQ.PDF(*x*, *deg_freedom*) returns the value of the Chi-Squared distribution probability density function for values *x* in the domain of affinely extended real numbers. This function returns zero for values *x* < 0.

SD.CHISQ.PDF(*x*, *deg_freedom*) is an alternative to the Excel CHISQ.DIST(*x*, *deg_freedom*, *FALSE*) built-in function.

4.1.4.2 SD.CHISQ.LTP(), SD.CHISQ.UTP()

The worksheet functions SD.CHISQ.LTP(*x*, *deg_freedom*) and SD.CHISQ.UTP(*x*, *deg_freedom*) return the Chi-Squared distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.CHISQ.LTP(*x*, *deg_freedom*) is an alternative to the Excel CHISQ.DIST(*x*, *deg_freedom*, *TRUE*) built-in function.

SD.CHISQ.UTP(x, deg_freedom) is an alternative to the Excel CHISQ.DIST.RT(*x*, *deg_freedom*) built-in function.

4.1.4.3 SD.CHISQ.LTQ(), SD.CHISQ.UTQ()

The worksheet functions SD.CHISQ.LTQ(*probability*, *deg_freedom*) and SD.CHISQ.UTQ(*probability*, *deg_freedom*) return the Chi-Squared distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.CHISQ.LTQ(*probability*, *deg_freedom*) is an alternative to the Excel CHISQ.INV(*probability*, *deg_freedom*) built-in function.

SD.CHISQ.UTQ(*probability*, *deg_freedom*) is an alternative to the Excel CHISQ.INV.RT(*probability*, *deg_freedom*) built-in function.

4.1.5 Exponential Distribution

The **exponential distribution** is a continuous distribution with probability density function:

$$pdf(x; \mu, \lambda) = \lambda e^{-(x-\mu)\lambda}, \qquad x \geq \mu;\ \lambda > 0$$

$$pdf(x; \mu, \lambda) = 0, \qquad x < \mu;\ \lambda > 0$$

Here μ is the location parameter and λ is the scale parameter. These Excel Add-ins assume μ = 0:

$$pdf(x; \lambda) = \lambda e^{-x\lambda}, \qquad x \geq 0;\ \lambda > 0$$

$$pdf(x; \lambda) = 0, \qquad x < 0;\ \lambda > 0$$

The Boost C++ Math Toolkit implementation and accuracy of the exponential distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/exp_dist.html

The scale parameter lambda (λ) is the inverse of the mean of the distribution and matches the *lambda* parameter used by these Excel Add-ins and the Microsoft Excel EXPON.DIST() built-in function. This built-in function also assumes μ = 0.

The *lambda* parameter must be greater than zero or the Excel Add-in exponential distribution worksheet functions will return the #NUM! error value.

4.1.5.1 SD.EXPON.PDF()

The worksheet function SD.EXPON.PDF(*x*, *lambda*) returns the value of the exponential distribution probability density function for values *x* in the domain of affinely extended real numbers. This function returns zero for values $x < 0$.

SD.EXPON.PDF(*x*, *lambda*) is an alternative to the Excel EXPON.DIST(*x*, *lambda, FALSE*) built-in function.

4.1.5.2 SD.EXPON.LTP(), SD.EXPON.UTP()

The worksheet functions SD.EXPON.LTP(*x, lambda*) and SD.EXPON.UTP(*x, lambda*) return the exponential distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.EXPON.LTP(*x, lambda*) is an alternative to the Excel EXPON.DIST(*x, lambda, TRUE*) built-in function.

4.1.5.3 SD.EXPON.LTQ(), SD.EXPON.UTQ()

The worksheet functions SD.EXPON.LTQ(*probability*, *lambda*) and SD.EXPON.UTQ(*probability*, *lambda*) return the exponential distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

4.1.6 Maximum Extreme Value Distribution

The **Extreme Value type I (Gumbel) distribution** has two forms, one based on the largest extreme value of a sample and the other based on the smallest extreme value of a sample. This distribution is used to model the maxima and minima of samples of random variables following normal or exponential types of distributions.

The Maximum Extreme Value distribution is a continuous distribution with probability density function:

$$pdf(x;\ \mu, s) = \frac{1}{\beta}\ e^{-\left(\frac{x-\mu}{\beta}\right)} e^{-e^{-\left(\frac{x-\mu}{\beta}\right)}}, \quad \beta > 0$$

Here μ is the location parameter and β is the scale parameter.

The Boost C++ Math Toolkit implementation and accuracy of the Maximum Extreme Value distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/extreme_dist.html

The Maximum Extreme Value distribution location (μ) and scale (β) parameters match the *location* and *scale* parameters used by these Excel Add-ins. There are no Microsoft Excel built-in Maximum Extreme Value distribution functions.

The *scale* parameter must be greater than zero or the Excel Add-in Maximum Extreme Value distribution worksheet functions will return the #NUM! error value.

4.1.6.1 SD.EVMAX.PDF()

The worksheet function SD.EVMAX.PDF(*x, location, scale*) returns the value of the Maximum Extreme Value distribution probability density function for values *x* in the domain of affinely extended real numbers.

4.1.6.2 SD.EVMAX.LTP(),SD.EVMAX.UTP()

The worksheet functions SD.EVMAX.LTP(*x, location, scale*) and SD.EVMAX.UTP(*x, location, scale*) return the Maximum Extreme Value distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

4.1.6.3 SD.EVMAX.LTQ(),SD.EVMAX.UTQ()

The worksheet functions SD.EVMAX.LTQ(*probability, location, scale*) and SD.EVMAX.UTQ(*probability, location, scale*) return the Maximum Extreme Value distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

4.1.7 Minimum Extreme Value Distribution

The **Extreme Value type I (Gumbel) distribution** has two forms, one based on the largest extreme value of a sample and the other based on the smallest extreme value of a sample. This distribution is used to model the maxima and minima of samples of random variables following normal or exponential types of distributions.

The Minimum Extreme Value distribution is a continuous distribution with probability density function:

$$pdf(x;\ \mu, s) = \frac{1}{\beta}\, e^{\left(\frac{x-\mu}{\beta}\right)} e^{-e^{\frac{x-\mu}{\beta}}}, \quad \beta > 0$$

Here μ is the location parameter and β is the scale parameter.

These Excel Add-ins compute the Minimum Extreme Value distribution functions using the Boost C++ Math Toolkit implementation of the Maximum Extreme Value distribution functions. The implementation and accuracy of the Boost Maximum Extreme Value distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/extreme_dist.html

The Minimum Extreme Value distribution location (μ) and scale (β) parameters match the *location* and *scale* parameters used by these Excel Add-ins. There are no Microsoft Excel built-in Minimum Extreme Value distribution functions.

The *scale* parameter must be greater than zero or the Excel Add-in Minimum Extreme Value distribution worksheet functions will return the #NUM! error value.

4.1.7.1 SD.EVMIN.PDF()

The worksheet function SD.EVMAX.PDF(*x*, *location*, *scale*) returns the value of the Minimum Extreme Value distribution probability density function for values *x* in the domain of affinely extended real numbers.

4.1.7.2 SD.EVMIN.LTP(),SD.EVMIN.UTP()

The worksheet functions SD.EVMIN.LTP(*x*, *location*, *scale*) and SD.EVMIN.UTP(*x*, *location*, *scale*) return the Minimum Extreme Value distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

4.1.7.3 SD.EVMIN.LTQ(),SD.EVMIN.UTQ()

The worksheet functions SD.EVMIN.LTQ(*probability*, *location*, *scale*) and SD.EVMIN.UTQ(*probability*, *location*, *scale*) return the Minimum Extreme Value distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

4.1.8 Gamma Distribution

The **gamma distribution** is a continuous distribution with probability density function:

$$pdf(x; \mu, \kappa, \lambda) = \frac{(x-\mu)^{(\kappa-1)} e^{-\frac{(x-\mu)}{\theta}}}{\theta^{\kappa} \Gamma(\kappa)}, \qquad x > \mu;\ \kappa, \theta > 0$$

$$pdf(x; \mu, \kappa, \lambda) = 0, \qquad x \le \mu;\ \kappa, \theta > 0$$

Here μ is the location parameter, κ is the shape parameter, θ is the scale parameter, and $\Gamma(\kappa)$ is the gamma function:

$$\Gamma(\kappa) = \int_0^\infty t^{\kappa-1} e^{-t} dt$$

These Excel Add-ins assume μ = 0:

$$pdf(x; \kappa, \lambda) = \frac{x^{(\kappa-1)} e^{-\frac{x}{\theta}}}{\theta^\kappa \Gamma(\kappa)}, \qquad x > 0;\ \kappa, \theta > 0$$

$$pdf(x; \kappa, \lambda) = 0, \qquad x \leq 0;\ \kappa, \theta > 0$$

The Boost C++ Math Toolkit implementation and accuracy of the gamma distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/gamma_dist.html

The gamma distribution shape (κ) and scale (θ) parameters match the *shape* and *scale* parameters used by these Excel Add-ins. They also match the Microsoft Excel GAMMA.DIST() and GAMMA.INV() built-in functions with parameters *alpha* and *beta*, respectively.

The *shape* and *scale* parameters must be positive or the Excel Add-in gamma distribution worksheet functions will return the #NUM! error value.

Since the **Erlang distribution** functions can be implemented using the gamma distribution functions with a shape parameter k (κ) having an integer value > 0, the Erlang distribution worksheet functions are not provided by these Excel Add-ins.

4.1.8.1 SD.GAMMA.PDF()

The worksheet function SD.GAMMA.PDF(*x*, *shape*, *scale*) returns the value of the gamma distribution probability density function for values *x* in the domain of affinely extended real numbers. This function returns zero for values *x* < 0.

SD.GAMMA.PDF(*x*, *shape*, *scale*) is an alternative to the Excel GAMMA.DIST(*x*, *alpha*, *beta*, *FALSE*) built-in function.

4.1.8.2 SD.GAMMA.LTP(), SD.GAMMA.UTP()

The worksheet functions SD.GAMMA.LTP(*x*, *shape*, *scale*) and SD.GAMMA.UTP(*x*, *shape*, *scale*) return the gamma distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.GAMMA.LTP(*x*, *shape*, *scale*) is an alternative to the Excel GAMMA.DIST(*x*, *alpha*, *beta*, *TRUE*) built-in function.

4.1.8.3 SD.GAMMA.LTQ(), SD.GAMMA.UTQ()

The worksheet functions SD.GAMMA.LTQ(*probability*, *shape*, *scale*) and SD.GAMMA.UTQ(*probability*, *shape*, *scale*) return the gamma distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.GAMMA.LTQ(*probability*, *shape*, *scale*) is an alternative to the Excel GAMMA.INV(*probability*, *alpha*, *beta*) built-in function.

4.1.9 Logistic Distribution

The logistic distribution is a continuous distribution with probability density function:

$$pdf(x;\ \mu, s) = \frac{e^{-(x-\mu)/s}}{s\,(1 + e^{-(x-\mu)/s})^2}, \qquad s > 0$$

$$pdf(x;\ \mu, s) = 0, \qquad x\ \le \theta;\ s > 0$$

Here μ is the location parameter and s is the scale parameter.

The Boost C++ Math Toolkit implementation and accuracy of the logistic distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/logistic_dist.html

The logistic distribution location (μ) and scale (s) parameters match the *mean* and *scale* parameters used by these Excel Add-ins. There are no Microsoft Excel built-in logistic distribution functions.

The *scale* parameter must be greater than zero or the Excel Add-in logistic distribution worksheet functions will return the #NUM! error value.

4.1.9.1 SD.LOGISTIC.PDF()

The worksheet function SD.LOGISTIC.PDF(*x*, *mean*, *scale*) returns the value of the logistic distribution probability density function for values *x* in the domain of affinely extended real numbers.

4.1.9.2 SD.LOGISTIC.LTP(), SD.LOGISTIC.UTP()

The worksheet functions SD.LOGISTIC.LTP(*x*, *mean*, *scale*) and SD.LOGISTIC.UTP(*x*, *mean*, *scale*) return the logistic distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

4.1.9.3 SD.LOGISTIC.LTQ(), SD.LOGISTIC.UTQ()

The worksheet functions SD.LOGISTIC.LTQ(*probability*, *mean*, *scale*) and SD.LOGISTIC.UTQ(*probability*, *mean*, *scale*) return the logistic distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

4.1.10 Lognormal Distribution

The l**ognormal distribution** is a continuous distribution with probability density function of most general form:

$$pdf(x;\ \theta, m, \sigma) = \frac{1}{(x-\theta)\sigma\sqrt{2\pi}}\, e^{-\frac{(\ln((x-\theta)/m))^2}{2\sigma^2}}, \qquad x > \theta;\ m, \sigma > 0$$

$$pdf(x;\ \theta, m, \sigma) = 0, \qquad x \le \theta;\ m, \sigma > 0$$

Here σ is the shape parameter (and standard deviation of the log of the distribution), θ is the location parameter, and m is the scale parameter (and the median of the distribution).

The case where θ = 0 is called the **2-parameter lognormal distribution**. If the 2-parameter lognormal distribution is parameterized with μ = ln(m), the probability density function is:

$$pdf(x;\ \mu, \sigma) = \frac{1}{x\sigma\sqrt{2\pi}}\, e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}}, \qquad x > 0;\ \sigma > 0$$

$$pdf(x;\ \mu, \sigma) = 0, \qquad x \le 0;\ \sigma > 0$$

In this parameterization if a random variable X is lognormally distributed then Y = ln(X) is normally distributed, μ is the mean of Y, and σ is the standard deviation of Y. This is the parameterization used by these Excel Add-ins and the Microsoft Excel LOGNORM.DIST() and LOGNORM.INV() built-in functions.

The case where θ = 0 and m = 1 is called the **standard lognormal distribution**.

The Boost C++ Math Toolkit implementation and accuracy of the 2-parameter lognormal distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/lognormal_dist.html

The 2-parameter lognormal distribution mean (μ) and standard_dev (σ) parameters match the *mean* and *standard_dev* parameters used by these Excel Add-ins and the Microsoft Excel LOGNORM.DIST() and LOGNORM.INV() built-in functions.

The *standard_dev* parameter must be greater than zero or the Excel Add-in lognormal distribution worksheet functions will return the #NUM! error value.

4.1.10.1 SD.LNORM.PDF()

The worksheet function SD.LNORM.PDF(*x*, *mean, standard_dev*) returns the value of the lognormal distribution probability density function for values *x* in the domain of affinely extended real numbers.

SD.LNORM.PDF(*x*, *mean, standard_dev*) is an alternative to the Excel LOGNORM.DIST(*x*, *mean, standard_dev, FALSE*) built-in function.

4.1.10.2 SD.LNORM.LTP(), SD.LNORM.UTP()

The worksheet functions SD.LNORM.LTP(*x*, *mean*, *standard_dev*) and SD.LNORM.UTP(*x*, *mean*, *standard_dev*) return the lognormal distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.LNORM.LTP(*x*, *mean*, *standard_dev*) is an alternative to the Excel LOGNORM.DIST(*x*, *mean*, *standard_dev*, *TRUE*) built-in function.

4.1.10.3 SD.LNORM.LTQ(), SD.LNORM.UTQ()

The worksheet functions SD.LNORM.LTQ(*probability*, *mean*, *standard_dev*) and SD.LNORM.UTQ(*probability*, *mean*, *standard_dev*) return the lognormal distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.LNORM.LTQ(*probability*, *mean*, *standard_dev*) is an alternative to the Excel LOGNORM.INV(*probability*, *mean*, *standard_dev*) built-in function.

4.1.11 Normal Distribution

The **normal (Gaussian) distribution** is a continuous distribution with probability density function:

$$pdf(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \qquad \sigma > 0$$

Here μ is the location parameter (mean) and σ is the scale parameter (standard deviation).

The Boost C++ Math Toolkit implementation and accuracy of the normal distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/normal_dist.html

The mean (μ) and standard deviation (σ) parameters match the *mean* and *standard_dev* parameters used by these Excel Add-ins and the Microsoft Excel NORM.DIST() and NORM.INV() built-in functions.

The *standard_dev* parameter must be greater than zero or the Excel Add-in normal distribution worksheet functions will return the #NUM! error value.

4.1.11.1 SD.NORM.PDF()

The worksheet function SD.NORM.PDF(*x*, *mean*, *standard_dev*) returns the value of the normal distribution probability density function for values *x* in the domain of affinely extended real numbers.

SD.NORM.PDF(*x, mean*, *standard_dev*) is an alternative to the Excel NORM.DIST(*x, mean*, *standard_dev*, *FALSE*) built-in function.

4.1.11.2 SD.NORM.LTP(), SD.NORM.UTP()

The worksheet functions SD.NORM.LTP(*x, mean*, *standard_dev*) and SD.NORM.UTP(*x*, *mean*, *standard_dev*) return the normal distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.NORM.LTP(*x, mean*, *standard_dev*) is an alternative to the Excel NORM.DIST(*x, mean*, *standard_dev*, *TRUE*) built-in function.

4.1.11.3 SD.NORM.LTQ(), SD.NORM.UTQ()

The worksheet functions SD.NORM.LTQ(*probability*, *mean*, *standard_dev*) and SD.NORM.UTQ(*probability, mean*, *standard_dev*) return the normal distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.NORM.LTQ(*probability*, *mean, standard_dev*) is an alternative to the Excel NORM.INV(*probability*, *mean*, *standard_dev*) built-in function.

4.1.12 Standard Normal Distribution

The **standard normal distribution** is a special case of the normal distribution with μ = 0 and σ = 1 and probability density function:

$$pdf(x) = \frac{1}{\sqrt{2\pi}}\, e^{-\frac{x^2}{2}}$$

The Boost C++ Math Toolkit implementation and accuracy of the standard normal distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/normal_dist.html

The standard normal distribution is supported by these Excel Add-ins and the Microsoft Excel NORM.DIST() and NORM.INV() built-in functions.

4.1.12.1 SD.SNORM.PDF()

The worksheet function SD.SNORM.PDF(*x*) returns the value of the standard normal distribution probability density function for values *x* in the domain of affinely extended real numbers.

SD.SNORM.PDF(*x*) is an alternative to the Excel NORM.S.DIST(*x*, *FALSE*) built-in function.

4.1.12.2 SD.SNORM.LTP(), SD.SNORM.UTP()

The worksheet functions SD.SNORM.LTP(*x*) and SD.SNORM.UTP(*x*) return the standard normal distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.SNORM.LTP(*x*) is an alternative to the Excel NORM.S.DIST(*x*, *TRUE*) built-in function.

4.1.12.3 SD.SNORM.LTQ(), SD.SNORM.UTQ()

The worksheet functions SD.SNORM.LTQ(*probability*) and SD.SNORM.UTQ(*probability*) return the standard normal distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.SNORM.LTQ(*probability*) is an alternative to the Excel NORM.S.INV(*probability*) built-in function.

4.1.13 Snedecor F Distribution

The **Snedecor F distribution** has probability density function:

$$\text{pdf}(x; n, m) = \frac{m^{\frac{m}{2}} n^{\frac{n}{2}} x^{\left(\frac{n}{2}-1\right)}}{(m + nx)^{\frac{(n+m)}{2}} B\left(\frac{n}{2}, \frac{m}{2}\right)}, \qquad x \geq 0;\ n, m > 0$$

$$\text{pdf}(x; n, m) = 0, \quad x < 0;\ n, m > 0$$

Here n and m are shape parameters and specify the **numerator** and **denominator degrees of freedom**, respectively, of the Snedecor F distribution. In the context of statistical tests n and m are positive integers.

Here $B(\alpha, \beta)$ is the beta function:

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}dt$$

The Boost C++ Math Toolkit implementation and accuracy of the Snedecor F distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/f_dist.html

The Snedecor F distribution degrees of freedom parameters n and m match the *deg_freedom1* and *deg_freedom2* parameters used by these Excel Add-ins and the Microsoft Excel F.DIST(), F.DIST.RT(), F.INV(), and F.INV.RT() built-in functions, except that in these Excel Add-ins the *deg_freedom1* and *deg_freedom2* parameters can be any positive number and are not truncated.

The *deg_freedom1* and *deg_freedom2* parameters must be greater than zero or the Excel Add-in Snedecor F distribution worksheet functions will return the #NUM! error value.

4.1.13.1 SD.F.PDF()

The worksheet function SD.F.PDF(*x*, *deg_freedom1*, *deg_freedom2*) returns the value of the Snedecor F distribution probability density function for values *x* in the domain of affinely extended real numbers. This function returns zero for values $x < 0$.

SD.F.PDF(*x*, *deg_freedom1*, *deg_freedom2*) is an alternative to the Excel F.DIST(*x*, *deg_freedom1*, *deg_freedom2*, *FALSE*) built-in function.

4.1.13.2 SD.F.LTP(), SD.F.UTP()

The worksheet functions SD.F.LTP(*x*, *deg_freedom1*, *deg_freedom2*) and SD.F.UTP(*x*, *deg_freedom1*, *deg_freedom2*) return the Snedecor F distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.F.LTP(*x*, *deg_freedom1*, *deg_freedom2*) is an alternative to the Excel F.DIST(*x*, *deg_freedom1*, *deg_freedom2*, *TRUE*) built-in function.

SD.F.UTP(*x*, *deg_freedom1*, *deg_freedom2*) is an alternative to the Excel F.DIST.RT(*x*, *deg_freedom1*, *deg_freedom2*) built-in function.

4.1.13.3 SD.F.LTQ(), SD.F.UTQ()

The worksheet functions SD.F.LTQ(*probability*, *deg_freedom1*, *deg_freedom2*) and SD.F.UTQ(*probability*, *deg_freedom1*, *deg_freedom2*) return the Snedecor F distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.F.LTQ(*probability*, *deg_freedom1*, *deg_freedom2*) is an alternative to the Excel F.INV(*probability*, *deg_freedom1*, *deg_freedom2*) built-in function.

SD.F.UTQ(*probability*, *deg_freedom1*, *deg_freedom2*) is an alternative to the Excel F.INV.RT(*probability*, *deg_freedom1*, *deg_freedom2*) built-in function.

4.1.14 Student's t Distribution

The **Student's t distribution** has probability density function:

$$pdf(x;\nu) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\,\Gamma(\frac{\nu}{2})(1+\frac{x^2}{\nu})^{\frac{\nu+1}{2}}}, \qquad \nu > 0$$

Here ν is the shape parameter and specifies the **number of degrees of freedom** of the Student's t distribution. In the context of statistical tests ν is a positive integer.

Here $\Gamma(\kappa)$ is the gamma function:

$$\Gamma(\kappa) = \int_0^\infty t^{\kappa-1} e^{-t} dt$$

The Boost C++ Math Toolkit implementation and accuracy of the Student's t distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/students_t_dist.html

The degrees of freedom parameter (ν) matches the *deg_freedom* parameter used by these Excel Add-ins and the Microsoft Excel T.DIST(), T.DIST.RT(), T.INV(), and T.INV.RT() built-in functions, except that in these Excel Add-ins the *deg_freedom* parameter can be any positive number and is not truncated.

The *deg_freedom* parameter must be greater than zero or the Excel Add-in Student's t distribution worksheet functions will return the #NUM! error value.

4.1.14.1 SD.T.PDF()

The worksheet function SD.T.PDF(*x*, *deg_freedom*) returns the value of the Student's t distribution probability density function for values *x* in the domain of affinely extended real numbers.

SD.T.PDF(*x*, *deg_freedom*) is an alternative to the Excel T.DIST(*x*, *deg_freedom*, *FALSE*) built-in function.

4.1.14.2 SD.T.LTP(), SD.T.UTP()

The worksheet functions SD.T.LTP(*x*, *deg_freedom*) and SD.T.UTP(*x*, *deg_freedom*) return the Student's t distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.T.LTP(*x*, *deg_freedom*) is an alternative to the Excel T.DIST(*x*, *deg_freedom*, *TRUE*) built-in function.

SD.T.UTP(*x*, *deg_freedom*) is an alternative to the Excel T.DIST.RT(*x*, *deg_freedom*) built-in function.

4.1.14.3 SD.T.LTQ(), SD.T.UTQ()

The worksheet functions SD.T.LTQ(*probability*, *deg_freedom*) and SD.T.UTQ(*probability*, *deg_freedom*) return the Student's t distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.T.LTQ(*probability*, *deg_freedom*) is an alternative to the Excel T.INV(*probability*, *deg_freedom*) built-in function.

4.1.15 Weibull Distribution

The **Weibull distribution** is a continuous distribution with probability density function:

$$pdf(x; \mu, \alpha, \beta) = \frac{\alpha}{\beta^{\alpha}}(x-\mu)^{(\alpha-1)} e^{-((x-\mu)/\beta)^{\alpha}}, \qquad x \geq \mu;\ \alpha, \beta > 0$$

$$pdf(x; \mu, \alpha, \beta) = 0, \qquad x < \mu;\ \alpha, \beta > 0$$

Here μ is the location parameter, α is the shape parameter, and β is the scale parameter.

The case where μ = 0 is called the **2-parameter Weibull distribution** with probability density function:

$$pdf(x; \alpha, \beta) = \frac{\alpha}{\beta^{\alpha}} x^{(\alpha-1)} e^{-(x/\beta)^{\alpha}}, \qquad x \geq 0;\ \alpha, \beta > 0$$

$$pdf(x; \alpha, \beta) = 0, \qquad x < 0;\ \alpha, \beta > 0$$

The 2-parameter Weibull distribution is the parameterization used by these Excel Add-ins and the Microsoft Excel WEIBULL.DIST() built-in function.

The case where μ = 0 and β = 1 is called the **standard Weibull distribution**.

The case where μ = 0 and α = 1 is the exponential distribution with λ = 1/β:

$$pdf(x; \beta) = \frac{1}{\beta} e^{-(x/\beta)}, \qquad x \geq 0;\ \beta > 0$$

$$pdf(x; \beta) = 0, \qquad x < 0;\ \beta > 0$$

The Boost C++ Math Toolkit implementation and accuracy of the 2-parameter Weibull distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/weibull_dist.html

The 2-parameter Weibull distribution alpha (α) and beta (β) parameters match the *alpha* and *beta* parameters used by these Excel Add-ins and the Microsoft Excel WEIBULL.DIST() built-in function with parameters.

The *alpha* and *beta* parameters must be greater than zero or the Excel Add-in Weibull distribution worksheet functions will return the #NUM! error value.

4.1.15.1 SD.WEIBULL.PDF()

The worksheet function SD.WEIBULL.PDF(*x*, *alpha*, *beta*) returns the value of the Weibull distribution probability density function for values *x* in the domain of affinely extended real numbers.

SD.WEIBULL.PDF(*x*, *alpha*, *beta*) is an alternative to the Excel WEIBULL.DIST(*x*, *alpha*, *beta*, *FALSE*) built-in function.

4.1.15.2 SD.WEIBULL.LTP(), SD.WEIBULL.UTP()

The worksheet functions SD.WEIBULL.LTP(*x*, *alpha*, *beta*) and SD.WEIBULL.UTP(*x*, *alpha*, *beta*) return the Weibull distribution lower and upper tail probability functions for values *x* in the domain of affinely extended real numbers.

SD.WEIBULL.LTP(*x*, *alpha*, *beta*) is an alternative to the Excel WEIBULL.DIST(*x*, *alpha*, *beta*, *TRUE*) built-in function.

4.1.15.3 SD.WEIBULL.LTQ(), SD.WEIBULL.UTQ()

The worksheet functions SD.WEIBULL.LTQ(*probability*, *alpha*, *beta*) and SD.WEIBULL.UTQ(*probability*, *alpha*, *beta*) return the Weibull distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

4.2 Discrete Distributions

4.2.1 Binomial Distribution

The **binomial distribution** is a discrete distribution used when there are exactly two mutually exclusive outcomes of a Bernoulli trial. We label these outcomes "success" and "failure". The binomial distribution probability mass function returns the probability of observing X = x successes in n trials, with the probability of success on a single trial denoted by p. The binomial distribution assumes that p is fixed for all trials:

$$pmf(x; p, n) = \binom{n}{x} p^x (1-p)^{(n-x)}, \qquad x = 0, 1, 2, \ldots, n;\ 0 \le p \le 1$$

$$pmf(x; p, n) = 0, \qquad x\ elsewhere;\ 0 \le p \le 1$$

Here

$$\binom{n}{k} = \frac{n!}{x!\,(n-x)!}$$

If the number of trails n is zero, then a binomial random variable X is zero with probability 1.

The random variable X for the **binomial distribution** is the number of successes for a fixed number of trials whereas for the **negative binomial distribution** the random variable is the number of failures observed for a fixed number of successes.

The Boost C++ Math Toolkit implementation and accuracy of the binomial distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/binomial_dist.html

The number of successes (x), number of trails (n), and the probability of success on each trial (p) used by this distribution match the *number_s*, *number_t*, and *probability_s* arguments used by these Excel Add-ins and the *number_s*, *trials*, and *probability_s* arguments used by the Microsoft Excel BINOM.DIST() built-in function.

The number of trials (*number_t*) parameter must be a nonnegative integer and the probability of success (*probability_s*) parameter must be in [0, 1] or the Excel Add-in binomial distribution worksheet functions will return the #NUM! error value.

4.2.1.1 SD.BINOM.PMF()

The worksheet function SD.BINOM.PMF(*number_s, number_t, probability_s*) returns the value of the binomial distribution probability mass function for values of *number_s* in the domain of affinely extended real numbers. This function returns zero for values of *number_s* outside the set { 0, 1, 2, …, *number_t*}.

SD.BINOM.PMF(*number_s, number_t, probability_s*) is an alternative to the Excel BINOM.DIST(*number_s, trials, probability_s, FALSE*) built-in function.

4.2.1.2 SD.BINOM.LTP(), SD.BINOM.UTP()

The worksheet functions SD.BINOM.LTP(*number_s, number_t, probability_s*) and SD.BINOM.UTP(*number_s, number_t, probability_s*) return the binomial distribution lower and upper tail probability functions for values of *number_s* in the domain of affinely extended real numbers.

SD.BINOM.LTP(*number_s, number_t, probability_s*) is an alternative to the Excel BINOM.DIST(*number_s, trials, probability_s, TRUE*) built-in function.

4.2.1.3 SD.BINOM.LTQ(), SD.BINOM.UTQ()

The worksheet functions SD.BINOM.LTQ(*probability, number_t, probability_s*) and SD.BINOM.UTQ(*probability, number_t, probability_s*) return the binomial distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

SD.BINOM.LTQ(*probability, number_t, probability_s*) is an alternative to the Excel BINOM.INV(*trials, probability_s, alpha*) built-in function.

4.2.2 Geometric Distribution

The **geometric distribution** is a discrete distribution used when there are exactly two mutually exclusive outcomes of a Bernoulli trial. We label these outcomes "success" and "failure". For a sequence of independent Bernoulli trials with fixed success fraction p the geometric distribution probability mass function returns the probability of observing X = x failures before the first success. The geometric distribution is therefore a special case of the negative binomial distribution where the number of successes r = 1:

$$pmf(x; r, p) = p(1-p)^x , \qquad x = 0, 1, 2, \ldots ;\ 0 \le p \le 1$$

$$pmf(x;r,p) = 0\,, \qquad x\ elsewhere;\ 0 \le p \le 1$$

The Boost C++ Math Toolkit implementation and accuracy of the geometric distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/geometric_dist.html

The number of failures (x) and the probability of success on each trial (p) used by this distribution match the *number_f*, and *probability_s* arguments used by these Excel Add-ins.

The probability of success (*probability_s*) parameter must be in [0, 1] or the Excel Add-in geometric distribution worksheet functions will return the #NUM! error value.

4.2.2.1 SD.GEOM.PMF()

The worksheet function SD.GEOM.PMF(*number_f*, *probability_s*) returns the value of the geometric distribution probability mass function for values of *number_f* in the domain of affinely extended real numbers.

4.2.2.2 SD.GEOM.LTP(), SD.GEOM.UTP()

The worksheet functions SD.GEOM.LTP(*number_f*, *probability_s*) and SD.GEOM.UTP(*number_f*, *probability_s*) return the geometric distribution lower and upper tail probability functions for values of *number_f* in the domain of affinely extended real numbers.

4.2.2.3 SD.GEOM.LTQ(), SD.GEOM.UTQ()

The worksheet functions SD.GEOM.LTQ(*probability*, *probability_s*) and SD.GEOM.UTQ(*probability*, *probability_s*) return the geometric distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

4.2.3 Hypergeometric Distribution

The **hypergeometric distribution** is a discrete distribution used when drawing elements without replacement from a finite population containing elements of two distinct types, some labeled "success" and the others labeled "failure".

The hypergeometric distribution probability mass function returns the probability of observing X = x successes in n draws from a finite population of size N containing exactly K successes:

$$pmf(x;K,n,N) = \frac{\binom{K}{x}\binom{N-K}{n-x}}{\binom{N}{n}}, \qquad \max(0, n+K-N) \le\ x \le \min(K,n)$$

$$pmf(x; K, n, N) = 0\,, \qquad x\ elsewhere$$

We require n, K ≤ N. Here:

$$\binom{n}{k} = \frac{n!}{x!\,(n-x)!}$$

The random variable X for the **hypergeometric distribution** is the number of successes in n draws **without replacement** from a finite population of size N containing exactly K successes whereas for the **binomial distribution** the random variable is the number of successes in n draws **with replacement**.

The Boost C++ Math Toolkit implementation and accuracy of the hypergeometric distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/hypergeometric_dist.html

The Ponderosa Computing Statistical Distributions Excel Add-ins use a modification of the Boost C++ Math Toolkit version 2.1.0 implementation for lower and upper tail quantiles. The Boost implementation uses a "fudge factor" computed on line 115 in the file hypergeometric_quantile.hpp:

```
T fudge_factor = 1 + tools::epsilon<T>() * ((N <=
boost::math::prime(boost::math::max_prime - 1)) ? 50 : 2 * N);
```

These Excel Add-ins replace that computation by setting fudge_factor to 1.

```
fudge_factor = 1.0;
```

The number of successes in the sample (x), the sample size (n), the number of successes in the population (K), and the population size (N) used by this distribution match the *sample_s*, *sample_n*, *population_s*, and *population_n* arguments used by these Excel Add-ins and the *sample_s*, *number_sample*, *population_s*, and *number_pop* arguments used by the Microsoft Excel HYPGEOM.DIST() built-in function.

If the *sample_n* or *population_s* parameter exceeds the *population_n* parameter the Excel Add-in hypergeometric distribution worksheet functions will return the #NUM! error value.

The Boost C++ Math Toolkit documentation states that its implementation of the hypergeometric distribution functions loose significant precision for large population sizes. For a population size N one should expect to lose $\log_{10} N$ decimal digits of precision, with the results becoming meaningless for $N \geq 10^{15}$. Due to the computational complexity and loss of precision for large population sizes in the hypergeometric distribution function implementation, the sample size (*sample_n*), the number of successes in the population (*population_s*), and the population size (*population_n*) parameters must be nonnegative integers in [0, 4294967295] or the Excel Add-in hypergeometric distribution worksheet functions will return the #NUM! error value.

4.2.3.1 SD.HGEOM.PMF()

The worksheet function SD.HGEOM.PMF(*sample_s, sample_n, population_s, population_n*) returns the value of the hypergeometric distribution probability mass function for values of *sample_s* in the domain of affinely extended real numbers.

SD.HGEOM.PMF(*sample_s, sample_n, population_s, population_n*) is an alternative to the Excel HYPGEOM.DIST(*sample_s, number_sample, population_s, number_pop, FALSE*) built-in function.

4.2.3.2 SD.HGEOM.LTP(), SD.HGEOM.UTP()

The worksheet functions SD.HGEOM.LTP(*sample_s, sample_n, population_s, population_n*) and SD.HGEOM.UTP(*sample_s, sample_n, population_s, population_n*) return the hypergeometric distribution lower and upper tail probability functions for values of *sample_s* in the domain of affinely extended real numbers.

SD.HGEOM.LTP(*sample_s, sample_n, population_s, population_n*) is an alternative to the Excel HYPGEOM.DIST(*sample_s, number_sample, population_s, number_pop, TRUE*) built-in function.

4.2.3.3 SD.HGEOM.LTQ(), SD.HGEOM.UTQ()

The worksheet functions SD.HGEOM.LTQ(*probability, sample_n, population_s, population_n*) and SD.HGEOM.UTQ(*probability, sample_n, population_s, population_n*) return the hypergeometric distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value. Note that these Excel Add-ins have modified the Boost implementation of lower and upper tail quantiles as noted above.

4.2.4 Negative Binomial Distribution

The **negative binomial distribution** is a discrete distribution used when there are exactly two mutually exclusive outcomes of a Bernoulli trial. We label these outcomes "success" and "failure". For a sequence of independent Bernoulli trials with fixed success fraction p the negative binomial distribution probability mass function returns the probability of observing X = x failures and r successes with success on the last trial:

$$pmf(x; r, p) = \binom{x + r - 1}{x} p^r (1 - p)^x \,, \qquad x = 0, 1, 2, \ldots; \; 0 \leq p \leq 1$$

$$pmf(x; r, p) = 0 \,, \qquad x \; elsewhere; \; 0 \leq p \leq 1$$

Here

$$\binom{n}{k} = \frac{n!}{x!\,(n-x)!}$$

The random variable X for the **negative binomial distribution** is the number of failures observed for a fixed number of successes whereas for the **binomial distribution** the random variable X is the number of successes for a fixed number of trials.

The Boost C++ Math Toolkit implementation and accuracy of the negative binomial distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/negative_binomial_dist.html

The number of failures (x), number of successes (r), and the probability of success on each trial (p) used by this distribution match the *number_f*, *number_s*, and *probability_s* arguments used by these Excel Add-ins and the Microsoft Excel NEGBINOM.DIST() built-in function.

The number of successes (*number_s*) parameter must be a positive integer and the probability of success (*probability_s*) parameter must be in [0, 1] or the Excel Add-in negative binomial distribution worksheet functions will return the #NUM! error value.

4.2.4.1 SD.NBINOM.PMF()

The worksheet function SD.NBINOM.PMF(*number_f*, *number_s*, *probability_s*) returns the value of the negative binomial distribution probability mass function for values of *number_f* in the domain of affinely extended real numbers. This function returns zero for values of *number_f* outside the set { 0, 1, 2, …}.

SD.NBINOM.PMF(*number_f*, *number_s*, *probability_s*) is an alternative to the Excel NEGBINOM.DIST(*number_f*, *number_s*, *probability_s*, *FALSE*) built-in function.

4.2.4.2 SD.NBINOM.LTP(), SD.NBINOM.UTP()

The worksheet functions SD.NBINOM.LTP(*number_f*, *number_s*, *probability_s*) and SD.NBINOM.UTP(*number_f*, *number_s*, *probability_s*) return the negative binomial distribution lower and upper tail probability functions for values of *number_f* in the domain of affinely extended real numbers.

SD.NBINOM.LTP(*number_f*, *number_s*, *probability_s*) is an alternative to the Excel NEGBINOM.DIST(*number_f*, *number_s*, *probability_s*, *TRUE*) built-in function.

4.2.4.3 SD.NBINOM.LTQ(), SD.NBINOM.UTQ()

The worksheet functions SD.NBINOM.LTQ(*probability*, *number_s*, *probability_s*) and SD.NBINOM.UTQ(*probability*, *number_s*, *probability_s*) return the negative binomial distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

4.2.5 Poisson Distribution

The **Poisson distribution** is a discrete distribution used to model the number of events X occurring within a fixed time interval, provided these events occur with a known mean rate (λ) of events within the time interval and the events are independent of the time since the last event.

The Poisson distribution probability mass function returns the probability of X = x events occurring in the fixed time interval:

$$pmf(x;\lambda) = \frac{e^{-\lambda}\lambda^{x}}{x!}\;, \qquad x = 0, 1, 2, \ldots;\ \lambda > 0$$

$$pmf(x;\lambda) = 0\,, \qquad x\ not\ postive\ integer;\ \lambda > 0$$

Here λ is the shape parameter which indicates the average number of events in the fixed time interval.

The Boost C++ Math Toolkit implementation and accuracy of the Poisson distribution functions are described here:

https://www.boost.org/doc/libs/1_70_0/libs/math/doc/html/math_toolkit/dist_ref/dists/poisson_dist.html

The number of events (x) and shape parameter (λ) match the *x* deviate and *mean* parameter used by these Excel Add-ins and the Microsoft Excel POISSON.DIST() built-in function.

The *mean* parameter must be nonnegative or the Excel Add-in Poisson distribution worksheet functions will return the #NUM! error value.

4.2.5.1 SD.POISSON.PMF()

The worksheet function SD.POISSON.PMF(*x*, *mean*) returns the value of the Poisson distribution probability mass function for values of *x* in the domain of affinely extended real numbers. This function returns zero for values of *x* outside the set of nonnegative integers.

SD.POISSON.PMF(*x*, *mean*) is an alternative to the Excel POISSON.DIST(*x*, *mean*, *FALSE*) built-in function.

4.2.5.2 SD.POISSON.LTP(), SD.POISSON.UTP()

The worksheet functions SD.POISSON.LTP(*x*, *mean*) and SD.POISSON.UTP(*x*, *mean*) return the Poisson distribution lower and upper tail probability functions for values of *x* in the domain of affinely extended real numbers.

SD.POISSON.LTP(*x*, *mean*) is an alternative to the Excel POISSON.DIST(*x*, *mean*, *TRUE*) built-in function.

4.2.5.3 SD.POISSON.LTQ(), SD.POISSON.UTQ()

The worksheet functions SD.POISSON.LTQ(*probability*, *mean*) and SD.POISSON.UTQ(*probability*, *mean*) return the Poisson distribution lower and upper tail quantiles for values of *probability* in [0, 1]. If *probability* is outside this range these quantile functions will return the #NUM! error value.

5 Installing and Activating these Excel Add-ins

There are three steps to enabling a Ponderosa Computing Statistical Distributions Excel Add-in to be used in Excel spreadsheets. These are (1) installing the add-in on your computer, (2) registering the add-in with Excel, and (3) activating the add-in in Excel. Here we describe these steps for the 32-bit Excel Add-in PcStaDistXLL_32.xll. The steps for the 64-bit Excel Add-in PcStaDistXLL_64.xll are similiar.

Installing the 32-bit Excel Add-In on your Computer

1. Go to the Ponderosa Computing downloads page (http://www.ponderosacomputing.com/downloads/) and identify the installation package for the Ponderosa Computing Statistical Distributions Excel Add-in.
2. Click on the link to the Excel Add-in installation package. In the **Opening** dialog box that appears click **Save File**, and save the installation package at a convenient location on your computer.
3. On your computer navigate to the location of the saved installation package and double-click on the file.
4. In the **Open File – Security Warning** dialog you can click on the Publisher link to verify the Digital Signature. Then click **Run**.
5. The installation program will guide you through the rest of the process of installing the Excel Add-in on your computer.
    1. A per-machine installation will install the 32-bit Excel Add-in PcStaDistXLL_32.xll by default in C:\Program Files (x86)\Ponderosa Computing\PcStaDistXLL_32.
    2. A per-user installation will install the 32-bit Excel Add-in PcStaDistXLL_32.xll by default in C:\Users\USER\AppData\Local\Programs\Ponderosa Computing\PcStaDistXLL_32.
6. You may delete the downloaded installation package.

Registering and Activating the Excel Add-in in Excel

1. Open a 32-bit version of Microsoft Excel.
2. Click the **File** tab, click **Options**, and then click the **Add-Ins** category.
3. In the Manage box, select **Excel Add-ins**, and then click **Go**.
4. The **Add-Ins** dialog box appears. Click **Browse**, locate and select the Excel Add-in XLL C:\Program Files (x86)\Ponderosa Computing\PcStaDistXLL\PcStaDistXLL.xll, and click **OK**.
5. The Excel Add-in has now been registered with Excel. The **Add-Ins** dialog box reappears and now contains a checked entry for the add-in. If you uncheck the entry for the add-in and click **OK**, or if you click **Cancel** the add-in will remain registered, but not activated. If you leave the entry checked and click **OK** the add-in will also be activated and a confirmation message will appear.

Activating a Registered Excel Add-in in Excel

If the Excel Add-in has been registered with, but not activated in Excel, then follow these steps to activate the add-in so it is accessible from Excel worksheets:

1. Open Excel (if not already open).
2. Click the **File** tab, click **Options**, and then click the **Add-Ins** category.
3. In the **Manage** box, click **Excel Add-ins**, and then click **Go**.
4. The **Add-Ins** dialog box appears. Select the check box next to the add-in you want to activate, and click **OK**.
5. A confirmation message will appear. Dismiss it.

The Excel Add-in is now loaded into Excel.

The activated Excel Add-in will be loaded into Excel whenever Excel is started until it is inactivated. Also, the Excel Add-in command menu will appear in the Menu Commands section of the Add-Ins tab on the Ribbon.

6 Inactivating and Uninstalling this Excel Add-ins

Should you decide to inactivate, unregister, uninstall, or remove a Ponderosa Computing Statistical Distributions Excel Add-in from Excel or your computer, you can do so by following these instructions.

Inactivating the Excel Add-In

If an activated Excel Add-in is inactivated, it will remain registered with Excel, but not loaded when Excel starts. Follow these steps to inactivate the Excel Add-in:

1. Open Excel (if not already open).
2. Click the **File** tab, click **Options**, and then click the **Add-Ins** category.
3. In the **Manage** box, click **Excel Add-ins**, and then click **Go**.
4. In the **Add-Ins** dialog box clear the check box next to the add-in that you want to inactivate and then click **OK**.
5. A confirmation message will appear. Dismiss it.

The command menu for an inactivated Excel Add-in will not appear in the Menu Commands section of the Add-Ins tab on the Ribbon. You can reactivate a registered Excel Add-in using the instructions above.

Unregistering the Excel Add-in from Excel

Inactivating the Excel Add-in does not unregister it from Excel. To unregister the add-in from Excel you must first uninstall the add-in from your computer and then remove it from Excel. These steps are described next.

Uninstalling the Excel Add-In from your Computer

Inactivating the Excel Add-in does not remove it from your computer. To remove the add-in from your computer, you must uninstall it.

1. Exit Excel (click the **File** tab, and then click **Exit**).
2. Open the **Control Panel**.
3. In the Control Panel, click **Programs** and **Uninstall a program**.
4. Click the name of the Excel Add-in in the list of installed programs, and then click the **Uninstall** button.
5. The **Windows Installer** dialog will appear. Click **Yes**.
6. The installation program will guide you through the rest of the process of uninstalling the Excel Add-in from your computer.

Removing an Uninstalled Excel Add-in from Excel

1. Open Excel.
2. Click the **File** tab, click **Options**, and then click the **Add-Ins** category.

3. In the **Manage** box, click **Excel Add-ins**, and then click **Go**.
4. The **Add-Ins** dialog box appears. Select the check box next to the add-in you want to unregister.
5. Excel will advise you that it cannot find the add-in and ask you whether to delete it from the list. Click **Yes**.
6. The Excel Add-in will be removed from the list of available add-ins.

7 References

[1] McCullough, Bruce D.; Heiser, David A. (2008). "On the accuracy of statistical procedures in Microsoft Excel 2007," Computational Statistics & Data Analysis, vol. **52** (10), pp 4570-4578, June 2008. https://www.sciencedirect.com/science/article/pii/S0167947308001606

[2] Yalta, A. Talha (2008). "The Accuracy of Statistical Distributions in Microsoft Excel 2007." Computational Statistics & Data Analysis, **52** (10), pp 4579-4586, June 2008. https://www.sciencedirect.com/science/article/pii/S0167947308001618

[3] Diego M. Oppenheimer (2009), Function Improvements in Excel 2010 (Microsoft 365 Blog, 2009-09-10). https://www.microsoft.com/en-us/microsoft-365/blog/2009/09/10/function-improvements-in-excel-2010/

[4] Diego M. Oppenheimer (2009), Function Consistency Improvements in Excel 2010 (Microsoft 365 Blog, 2009-09-14). https://www.microsoft.com/en-us/microsoft-365/blog/2009/09/14/function-consistency-improvements-in-excel-2010/

[5] Microsoft. "Excel Statistical Functions Reference." https://support.office.com/en-us/article/statistical-functions-reference-624dac86-a375-4435-bc25-76d659719ffd

[6] Microsoft. "Floating-point arithmetic may give inaccurate results in Excel." https://support.microsoft.com/en-us/help/78113/floating-point-arithmetic-may-give-inaccurate-results-in-excel

[7] Boost.Org

[8] Boost Math Toolkit 2.9.0

[9] Wikipedia: IEEE Standard for Floating-Point Arithmetic (IEEE 754). https://en.wikipedia.org/wiki/IEEE_754

[10] Wikipedia: Quantile function. https://en.wikipedia.org/wiki/Quantile_function

[11] *NIST/SEMATECH e-Handbook of Statistical Methods*, http://www.itl.nist.gov/div898/handbook/ , 10/30/2013.

