Oracle® Crystal Ball, Fusion Edition

Statistical Guide

RELEASE 11.1.2

ORACLE®

ENTERPRISE PERFORMANCE
MANAGEMENT SYSTEM

Crystal Ball Statistical Guide, 11.1.2

Contents

1

Welcome

Introduction

Oracle Crystal Ball, Fusion Edition is a user-friendly, graphically oriented forecasting and risk analysis program that takes the uncertainty out of decision-making.

Crystal Ball runs on several versions of Microsoft Windows and Microsoft Excel. For a complete list of required hardware and software, see the current *Oracle Crystal Ball Installation and Licensing Guide.*

About This Guide

The *Oracle Crystal Ball Statistical Guide* contains distribution defaults and formulas and other statistical information. It includes the following additional chapters:

● Chapter 2, "Statistical Definitions"

This chapter describes basic statistical concepts and explains how they are used in Crystal Ball.

● Chapter 3, "Equations and Methods"

This chapter lists the mathematical formulas used in Crystal Ball to calculate distributions and descriptive statistics and describes the type of random number generator used in Crystal Ball. This appendix is designed for users with sophisticated knowledge of statistics.

● Chapter 4, "Default Names and Distribution Parameters"

This chapter summarizes the default values of Crystal Ball.

● Chapter 5, "Predictor Formulas and Statistics"

This chapter provides formulas and techniques used in Predictor.

Note: Because of round-off differences in various system configurations, you might obtain calculated results that are slightly different from those in the examples.

Technical Support and More

Oracle offers technical support, training, and other services to help you use Crystal Ball. See:

`http://www.oracle.com/crystalball`

2 Statistical Definitions

Introduction

This chapter provides formulas for the following types of statistics:

- "Measures of Central Tendency" on page 9
- "Measures of Variability" on page 11
- "Other Measures for a Data Set" on page 12
- "Other Statistics" on page 15

It also describes methodology and statistics for:

- "Simulation Sampling Methods" on page 18
- "Confidence Intervals" on page 19
- "Random Number Generation" on page 21
- "Process Capability Metrics" on page 21

Statistics

This section discusses basic statistics used in Crystal Ball.

Measures of Central Tendency

The measures of central tendency for a data set are mean, median, and mode.

Mean

The mean of a set of values is found by adding the values and dividing their sum by the number of values. The term "average" usually refers to the mean. For example, 5.2 is the mean, or average, of 1, 3, 6, 7, and 9.

Formula:

$$\frac{1}{n}\sum_{i=1}^{n} x_i \qquad\qquad (\bar{x})$$

Median

The median is the middle value in a set of sorted values. For example, 6 is the median of 1, 3, 6, 7, and 9 (recall that the mean is 5.2).

If an odd number of values exists, you find the median by ordering the values from smallest to largest and then selecting the middle value.

If an even number of values exists, then the median is the mean of the two middle values.

Mode

The mode is the value that occurs most frequently in a set of values. The greatest degree of clustering occurs at the mode.

The modal wage, for example, is the one received by the greatest number of workers. The modal color for a new product is the one preferred by the greatest number of consumers.

In a perfectly symmetrical distribution, such as the normal distribution (the distribution on the left, below), the mean, median, and mode converge at one point.

In an asymmetrical, or skewed, distribution, such as the lognormal distribution, the mean, median, and mode tend to spread out, as shown in the second distribution (on the right) in the following example (Figure 1).

Figure 1   Symmetrical and Asymmetrical Distributions
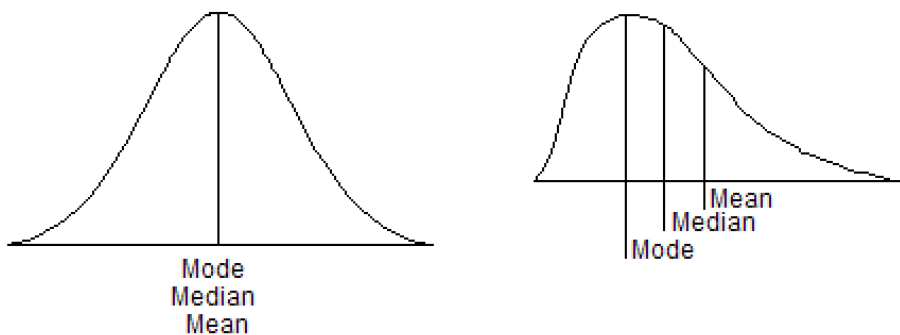
Note: When running simulations, forecast data likely will be continuous and no value will occur more than once. In such a case, Crystal Ball sets the mode to '---' in the Statistics view to indicate that the mode is undefined.

Measures of Variability

The measures of variability for a data set are variance, standard deviation, and range (or range width).

Variance

Variance is a measure of the dispersion, or spread, of a set of values about the mean. When values are close to the mean, the variance is small. When values are widely scattered about the mean, the variance is larger.

Formula:

$$\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2 \qquad (s^2)$$

➤ To calculate the variance of a set of values:

1 Find the mean or average.

2 For each value, calculate the difference between the value and the mean.

3 Square the differences.

4 Divide by *n* - 1, where *n* is the number of differences.

For example, suppose your values are 1, 3, 6, 7, and 9. The mean is 5.2. The variance, denoted by *s2* , is calculated as follows:

$$s^2 = \frac{(1-5.2)^2 + (3-5.2)^2 + (6-5.2)^2 + (7-5.2)^2 + (9-5.2)^2}{5-1}$$

$$= \frac{40.8}{4} = 10.2$$

Note: The calculation uses *n* - 1 instead of *n* to correct for the fact that the mean was calculated from the data sample, thus removing one degree of freedom. This correction makes the sample variances slightly larger than the variance of the entire population.

Standard Deviation

The standard deviation is the square root of the variance for a distribution. Like the variance, it is a measure of dispersion about the mean and is useful for describing the "average" deviation. See the description for the variance in the next section.

For example, you can calculate the standard deviation of the values 1, 3, 6, 7, and 9 by finding the square root of the variance that is calculated in the variance example that follows.

Formula:

$$\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2} \qquad (s)$$

The standard deviation, denoted as *s*, is calculated from the variance as follows:

$$s = \sqrt{10.2} = 3.19$$

Coefficient of Variability

The coefficient of variability provides you with a measurement of how much your forecast values vary relative to the mean value. Because this statistic is independent of the forecast units, you can use it to compare the variability of two or more forecasts, even when the forecast scales differ.

For example, if you are comparing the forecast for a penny stock with the forecast for a stock on the New York Stock Exchange, you would expect the average variation (standard deviation) of the penny stock price to appear smaller than the variation of the NYSE stock. However, if you compare the coefficient of variability statistic for the two forecasts, you will notice that the penny stock shows significantly more variation on an absolute scale.

The coefficient of variability typically ranges from a value greater than 0 to 1. It might exceed 1 in a few cases in which the standard deviation of the forecast is unusually high. This statistic is computed by dividing the standard deviation by the mean.

The coefficient of variability is calculated by dividing the standard deviation by the mean, as follows.

$$\text{coefficient of variability} = \frac{s}{\bar{x}}$$

To present this number as a percentage, multiply the result of the coefficient of variability calculation by 100.

Range (Also Range Width)

The range minimum is the smallest number in a set of values; the range maximum is the largest number.

The range is the difference between the range minimum and the range maximum.

For example, if the range minimum is 10, and the range maximum is 70, then the range is 60.

Other Measures for a Data Set

These statistics also describe the behavior of a data set: skewness, kurtosis, and mean standard error.

Skewness

A distribution of values (a frequency distribution) is said to be "skewed" if it is not symmetrical.

For example, suppose the curves in the example below represent the distribution of wages within a large company (Figure 2).

Figure 2   Positive and Negative Skewness

Curve A illustrates positive skewness (skewed "to the right"), where most of the wages are near the minimum rate, although some are much higher. Curve B illustrates negative skewness (skewed "to the left"), where most of the wages are near the maximum, although some are much lower.

If you describe the curves statistically, curve A is positively skewed and might have a skewness coefficient of 0.5, and curve B is negatively skewed and might have a -0.5 skewness coefficient.

A skewness value greater than 1 or less than -1 indicates a highly skewed distribution. A value between 0.5 and 1 or -0.5 and -1 is moderately skewed. A value between -0.5 and 0.5 indicates that the distribution is fairly symmetrical.

Method:

Skewness is computed by finding the third moment about the mean and dividing by the cube of the standard deviation.

Formula:

$$\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3}{s^3}$$

Kurtosis

Kurtosis refers to the peakedness of a distribution. For example, a distribution of values might be perfectly symmetrical but look either very "peaked" or very "flat," as illustrated below (Figure 3).

Figure 3   Peaked and Flat Kurtosis
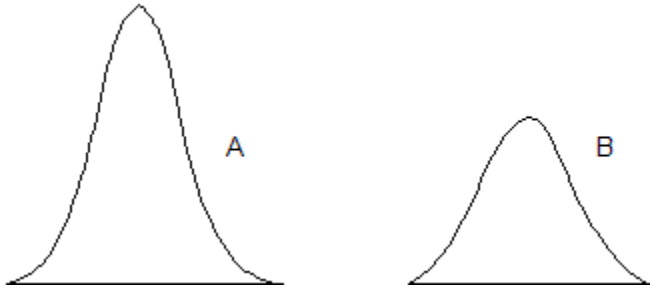
Suppose the curves in Figure 3 represent the distribution of wages in a large company. Curve A is fairly peaked, because most of the employees receive about the same wage, with few receiving very high or low wages. Curve B is flat-topped, indicating that the wages cover a wider spread.

Describing the curves statistically, curve A is fairly peaked, with a kurtosis of about 4. Curve B, which is fairly flat, might have a kurtosis of 2.

A normal distribution usually is used as the standard of reference and has a kurtosis of 3. Distributions with kurtosis values of less than 3 are described as platykurtic (meaning flat), and distributions with kurtosis values of greater than 3 are leptokurtic (meaning peaked).

Method:

Kurtosis, or peakedness, is calculated by finding the fourth moment about the mean and dividing by the quadruple of the standard deviation.

Formula:

$$\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^4}{s^4}$$

Mean Standard Error

The mean standard error statistic enables you to determine the accuracy of your simulation results and how many trials are necessary to ensure an acceptable level of error. This statistic tells you the probability of the estimated mean deviating from the true mean by more than a specified amount. The probability that the true mean of the forecast is the estimated mean (plus or minus the mean standard error) is approximately 68 percent.

Note: The mean standard error statistic provides information only on the accuracy of the mean and can be used as a general guide to the accuracy of the simulation. The standard error for other statistics, such as mode and median, probably will differ from the mean standard error.

Formula:

$$\frac{s}{\sqrt{n}}$$

where *s* = standard deviation and *n* = number of trials.

The error estimate might be inverted to show that the number of trials needed to yield a desired error:

$$n = \frac{s^2}{\varepsilon^2}$$

Other Statistics

These statistics describe relationships between data sets (correlation coefficient, rank correlation) or other data measurements (certainty, percentile, confidence intervals).

Correlation Coefficient

Note: Crystal Ball uses rank correlation to determine the correlation coefficient of variables. For more information on rank correlation, see "Rank Correlation" on page 16.

When the values of two variables depend upon one another in whole or in part, the variables are considered correlated. For example, an "energy cost" variable likely will show a positive correlation with an "inflation" variable. When the "inflation" variable is high, the "energy cost" variable is also high; when the "inflation" variable is low, the "energy cost" variable is low.

In contrast, "product price" and "unit sale" variables might show a negative correlation. For example, when prices are low, high sales are expected; when prices are high, low sales are expected.

By correlating pairs of variables that have such a positive or negative relationship, you can increase the accuracy of your simulation forecast results.

The correlation coefficient is a number that describes the relationship between two dependent variables. Coefficient values range between -1 and 0 for a negative correlation and 0 and +1 for a positive correlation. The closer the absolute value of the correlation coefficient is to either +1 or -1, the more strongly the variables are related.

When an increase in one variable is associated with an increase in another, the correlation is called positive (or direct) and is indicated by a coefficient between 0 and 1. When an increase in one variable is associated with a decrease in another variable, the correlation is called negative (or inverse) and is indicated by a coefficient between 0 and -1. A value of 0 indicates that the variables are unrelated to one another. The example below shows three correlation coefficients (Figure 4).

Figure 4   Types of Correlation
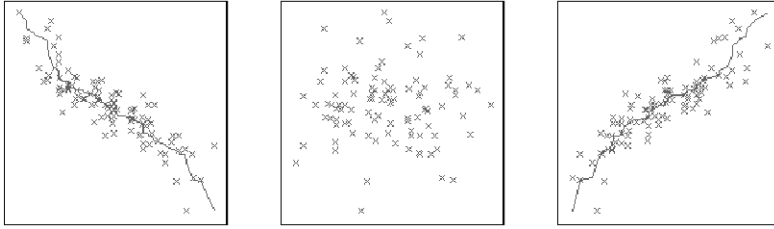
Negative correlation   Zero correlation   Positive correlation

For example, assume that total hotel food sales might be correlated with hotel room rates. Total food sales likely will be higher, for example, at hotels with higher room rates. If food sales and room rates correspond closely for various hotels, the correlation coefficient is close to 1. However, the correlation might not be perfect (correlation coefficient is less than 1). Some people might eat meals outside of the hotel, and others might skip some meals.

When you select a correlation coefficient to describe the relationship between two variables in your simulation, you must consider how closely they are related. You should never need to use an actual correlation coefficient of 1 or -1. Generally, you should represent these types of relationships as formulas on your spreadsheet.

Formula:

$$\frac{n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{\sqrt{n \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2} \cdot \sqrt{n \sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i\right)^2}}$$

Note: Crystal Ball uses rank correlation to correlate assumption values. This means that assumption values are replaced by their rankings from lowest to highest value by the integers 1 to *n*, before computing the correlation coefficient. This method allows distribution types to be ignored when correlating assumptions.

Rank Correlation

A correlation coefficient measures the strength of the linear relationship between two variables. However, if the two variables do not have the same probability distributions, they are not likely related linearly. Under such circumstances, the correlation coefficient calculated on their raw values has little meaning.

If you calculate the correlation coefficient using rank values instead of actual values, the correlation coefficient is meaningful even for variables with different distributions.

You determine rank values by arranging the actual values in ascending order and replacing the values with their rankings. For example, the lowest actual value will have a rank of 1; the next-lowest actual value will have a rank of 2; and so on.

Crystal Ball uses rank correlation to correlate assumptions. The slight loss of information that occurs using rank correlation is offset by two advantages:

- First, the correlated assumptions need not have similar distribution types. In effect, the correlation function in Crystal Ball is *distribution-independent*. The rank correlation method works even when a distribution has been truncated at one or both ends of its range.

- Second, the values generated for each assumption are not changed; they are merely *rearranged* to produce the desired correlation. In this way, the original distributions of the assumptions are preserved.

Certainty

The forecast chart shows not only the range of results for each forecast, but also the probability, or certainty, of achieving results within a range. Certainty is the percent chance that a forecast value will fall within a specified range.

By default, the certainty range is from negative infinity to positive infinity. The certainty for this range is always 100 percent. However, you might want to estimate the chance of a forecast result falling in a specific range, say from zero to infinity (which you might want to calculate to ensure that you make a profit).

For example, consider the forecast chart in Figure 5. If your objective is to make a minimum return of $2,000,000, you might choose a range of $2,000,000 to +Infinity. In this case, the certainty is almost 75 percent.

Figure 5   Certainty of a $2 Million Net Profit
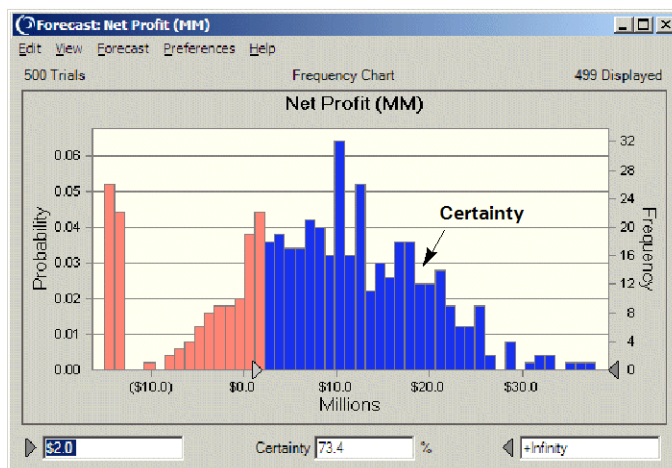
Percentiles

A percentile is a number on a scale of 0–100 that indicates the percent of a distribution that is equal to or less than a value (by default). Standardized tests usually report results in percentiles. If you are in the 95th percentile, then 95 percent of test takers had either the same score or a lower score. This number does not mean that you answered 95 percent of the questions correctly. You might have answered only 20 percent correctly, but your score was better than, or as good as, 95 percent of the other test takers' scores.

Crystal Ball calculates percentiles of forecast values using an interpolation algorithm. This algorithm is used for both discrete and continuous data, resulting in the possibility of having real numbers as percentiles for even discrete data sets. If an exact forecast value corresponds to a calculated percentile, Crystal Ball accepts that as the percentile. Otherwise, Crystal Ball proportionally interpolates between the two nearest values to calculate the percentile.

Note: When calculating medians, Crystal Ball does not use the proportional interpolation algorithm; it uses the classical definition of median, described in "Median" on page 10.

Percentiles for a normal distribution look like the following figure (Figure 6).

Figure 6   Normal Distribution with Percentiles
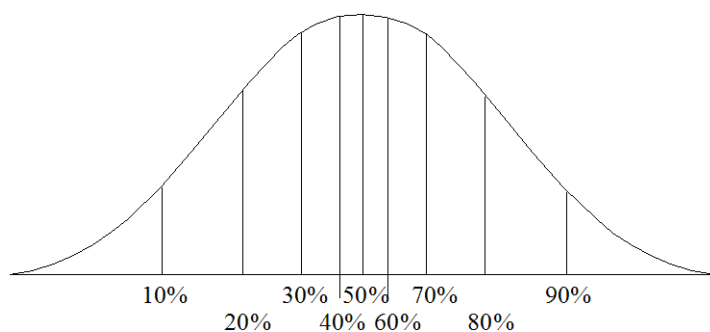
Simulation Sampling Methods

During each trial of a simulation, Crystal Ball selects a random value for each assumption in your model. Crystal Ball selects these values based on the Sampling dialog box (displayed when you select Run, then Run Preferences). The sampling methods:

- Monte Carlo: Randomly selects any value from the defined distribution of each assumption.
- Latin Hypercube: Randomly selects values and spreads them evenly over the defined distribution of each assumption.

Monte Carlo Sampling

Monte Carlo simulation randomly and repeatedly generates values for uncertain variables to simulate a model. The values for each assumption's probability distribution are random and totally independent. In other words, the random value selected for one trial have no effect on the next random value generated.

Monte Carlo simulation was named for Monte Carlo, Monaco, whose casinos feature games of chance such as roulette, dice, and slot machines, all of which exhibit random behavior.

Such random behavior is similar to how Monte Carlo simulation selects variable values at random to simulate a model. When you roll a die, you know that a 1, 2, 3, 4, 5, or 6 will come up, but you do not know which for any particular trial. It is the same with the variables that have

a known range of values and an uncertain value for any particular time or event (for example, interest rates, staffing needs, stock prices, inventory, phone calls per minute).

Using Monte Carlo sampling to approximate the true shape of the distribution requires more trials than Latin Hypercube.

Use Monte Carlo sampling to simulate "real world" what-if scenarios for your spreadsheet model.

Latin Hypercube Sampling

In Latin Hypercube sampling, Crystal Ball divides each assumption's probability distribution into nonoverlapping segments, each having equal probability, as illustrated below (Figure 7).

Figure 7   Normal Distribution with Latin Hypercube Sampling Segments
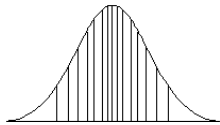
While a simulation runs, Crystal Ball selects a random assumption value for each segment according to the segment's probability distribution. This collection of values forms the Latin Hypercube sample. After Crystal Ball has sampled each segment exactly once, the process repeats until the simulation stops.

The Sample Size option (displayed when you select Run Preferences, then Sample), controls the number of segments in the sample.

Latin Hypercube sampling is generally more precise when calculating simulation statistics than is conventional Monte Carlo sampling, because the entire range of the distribution is sampled more evenly and consistently. Latin Hypercube sampling requires fewer trials to achieve the same level of statistical accuracy as Monte Carlo sampling. The added expense of this method is the extra memory required to track which segments have been sampled while the simulation runs. (Compared to most simulation results, this extra overhead is minor.)

Use Latin Hypercube sampling when you are concerned primarily with the accuracy of the simulation statistics.

Confidence Intervals

Because Monte Carlo simulation uses random sampling to estimate model results, statistics computed on these results, such as mean, standard deviation and percentiles, always contain some kind of error. A confidence interval (CI) is a bound calculated around a statistic that attempts to measure this error with a given level of probability. For example, a 95 percent confidence interval around the mean statistic is defined as a 95 percent chance that the mean will be contained within the specified interval. Conversely, a 5 percent chance exists that the mean will lie outside the interval. Shown graphically, a confidence interval around the mean looks like Figure 8.

Figure 8   Confidence Interval
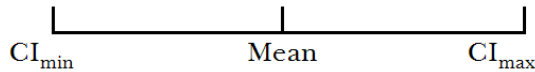
For most statistics, the confidence interval is symmetrical around the statistic, so that $x = (\text{CImax} - \text{Mean}) = (\text{Mean} - \text{CImin})$. This accuracy lets you make statements of confidence such as "the mean will lie within the estimated mean plus or minus x with 95 percent probability."

Confidence intervals are important for determining the accuracy of statistics, hence, the accuracy of the simulation. Generally speaking, as more trials are calculated, the confidence interval narrows and the statistics become more accurate. The precision control feature of Crystal Ball lets you stop the simulation when the specified precision of the chosen statistics is reached. Crystal Ball periodically checks whether the confidence interval is less than the specified precision.

Notice that the Bootstrap tool in Crystal Ball enables you to calculate the confidence intervals for any set of statistics using empirically-based methods.

The following sections describe how Crystal Ball calculates the confidence interval for each statistic.

Mean Confidence Interval

Formula:

$$z \cdot \frac{s}{\sqrt{n}}$$

where s is the standard deviation of the forecast, n is the number of trials, and z is the z value based on the specified confidence level (to set the confidence level, from Run Preferences, select Trials).

Standard Deviation Confidence Interval

Formula:

$$z \cdot s \cdot \sqrt{\frac{k-1}{4 \cdot (n-1)}}$$

where s is the standard deviation of the forecast, k is the kurtosis, n is the number of trials, and z is the z value based on the specified confidence level (from Run Preferences, select Trials).

Percentiles Confidence Interval

To calculate the confidence interval for the percentiles, instead of a mathematical formula, Crystal Ball uses an analytical bootstrapping method.

Random Number Generation

Crystal Ball uses the random number generator described in the following iteration formula as the basis for all nonuniform generators. For no starting seed value, Crystal Ball takes the value of the number of milliseconds elapsed since Windows started.

Method: Multiplicative Congruential Generator

This routine uses the iteration formula:

$$r \leftarrow (62089911 \bullet r) \bmod (2^{31} - 1)$$

Comment:

The generator has a period of length of 2^{31} - 2, or 2,147,483,646. This means that the cycle of random numbers repeats after several billion trials. This formula is discussed in detail in the *Simulation Modeling & Analysis* and *Art of Computer Programming*, Vol. II, references in the *Crystal Ball User's Guide* bibliography.

Process Capability Metrics

The Crystal Ball process capability metrics are provided to support quality improvement methodologies such as Six Sigma, Design for Six Sigma (DFSS), and Lean Principles. They appear in forecast charts when a forecast definition includes a lower specification limit (LSL), upper specification limit (USL), or both. Optionally, a target value can be included in the definition.

The following sections describe capability metrics calculated by Crystal Ball. In general, capability indices beginning with C (such as Cpk) are for short-term data, and long-term equivalents begin with P (such as Ppk).

Cp

Short-term capability index indicating what quality level the forecast output potentially is capable of producing. It is defined as the ratio of the specification width to the forecast width. If a Cp is equal to or greater than 1, then a short-term 3-sigma quality level is possible.

Formula:

$$C_p = \frac{USL - LSL}{6\sigma}$$

Pp

Long-term capability index indicating what quality level the forecast output is potentially capable of producing. It is defined as the ratio of the specification width to the forecast width. If a Pp is equal to or greater than 1, then a short-term 3-sigma quality level is possible.

Formula:

$$P_p = \frac{USL - LSL}{6\sigma}$$

Cpk-lower

One-sided short-term capability index; for normally distributed forecasts, the ratio of the difference between the forecast mean and lower specification limit over three times the forecast short-term standard deviation; often used to calculate process capability indices with only a lower specification limit.

Formula:

$$C_{pk-LOWER} = \frac{\mu - LSL}{3\sigma}$$

Ppk-lower

One-sided long-term capability index; for normally distributed forecasts, the ratio of the difference between the forecast mean and lower specification limit over three times the forecast long-term standard deviation; often used to calculate process capability indices with only a lower specification limit.

Formula:

$$P_{pk-LOWER} = \frac{\mu - LSL}{3\sigma}$$

Cpk-upper

One-sided short-term capability index; for normally distributed forecasts, the ratio of the difference between the forecast mean and upper specification limit over three times the forecast short-term standard deviation; often used to calculate process capability indices with only an upper specification limit.

Formula:

$$C_{pk-UPPER} = \frac{USL - \mu}{3\sigma}$$

Ppk-upper

One-sided long-term capability index; for normally distributed forecasts, the ratio of the difference between the forecast mean and upper specification limit over three times the forecast long-term standard deviation; often used to calculate process capability indices with only an upper specification limit.

Formula:

$$P_{pk-UPPER} = \frac{USL - \mu}{3\sigma}$$

Cpk

Short-term capability index (minimum of calculated Cpk-lower and Cpk-upper) that takes into account the centering of the forecast with respect to the midpoint of the specified limits; a Cpk equal to or greater than 1 indicates a quality level of 3 sigmas or better.

Formula:

$$C_{pk} = \min(C_{pk-UPPER}, C_{pk-LOWER}) = C_p(1-k)$$

where:

$$k = \frac{\left|\left(\frac{USL + LSL}{2}\right) - \mu\right|}{(USL - LSL)/2}$$

Ppk

Long-term capability index (minimum of calculated Cpk-lower and Cpk-upper) that takes into account the centering of the forecast with respect to the midpoint of the specified limits; a Ppk equal to or greater than 1 indicates a quality level of 3 sigmas or better.

Formula:

$$P_{pk} = \min(P_{pk-UPPER}, P_{pk-LOWER}) = P_p(1-k)$$

where:

$$k = \frac{\left|\left(\frac{USL + LSL}{2}\right) - \mu\right|}{(USL - LSL)/2}$$

Cpm

Short-term Taguchi capability index; similar to Cpk but considers a target value, which may not necessarily be centered between the upper and lower specification limits.

Formula:

$$C_{pm} = \frac{USL - LSL}{6\sqrt{(\mu - T)^2 + \sigma^2}}$$

where T is Target value; default is:

$$\frac{USL + LSL}{2}$$

Ppm

Long-term Taguchi capability index; similar to Ppk but considers a target value, which may not necessarily be centered between the upper and lower specification limits.

Formula:

$$P_{pm} = \frac{USL - LSL}{6\sqrt{(\mu - T)^2 + \sigma^2}}$$

where T is Target value; default is:

$$\frac{USL + LSL}{2}$$

Z-LSL

The number of standard deviations between the forecast mean and the lower specification limit.

Note: Z scores typically are reported only for normal data.

Formula:

$$Z_{LSL} = \frac{\mu - LSL}{\sigma}$$

Z-USL

The number of standard deviations between the forecast mean and the upper specification limit.

Note: Z scores typically are reported only for normal data.

Formula:

$$Z_{USL} = \frac{USL - \mu}{\sigma}$$

Zst

For short-term data, $Z_{ST} = Z_{TOTAL}$, expressed as Zst-total,

where

$$Z_{TOTAL} = \Phi^{-1}(p(N/C)_{TOTAL})$$

and

$\Phi^{-1}(x)$

is the inverse normal cumulative distribution function, which assumes a right-sided tail.

In Microsoft Excel:

$\Phi^{-1}(x) = -\text{NORMSINV}(x)$

When displaying short-term metrics, Z_{ST} appears as Zst-total. This metric is equal to Z-LSL if there is only a lower specification limit, or Z-USL if there is only an upper specification limit.

For long-term data, $Z_{ST} = Z_{LT} + \text{ZScoreShift}$. When displaying long-term metrics, Z_{ST} appears in the capability metrics table as Zst.

Note: Z scores typically are reported only for normal data. The maximum value for Z scores calculated by Crystal Ball from forecast data is 21.18.

Zst-total

For short-term metrics when both specification limits are defined, the number of standard deviations between the short-term forecast mean and the lower boundary of combining all defects onto the upper tail of the normal curve. Also equal to Zlt-total plus the Z-score shift value if long-term metrics are calculated.

When short-term metrics are calculated, Zst-total is equivalent to Z_{ST}, described in the previous section.

Note: Z scores typically are reported only for normal data.

Zlt

For long-term data, $Z_{LT} = Z_{TOTAL}$, expressed as Zlt-total,

where

$Z_{TOTAL} =$

$\Phi^{-1}(p(N/C)_{TOTAL})$

and

$\Phi^{-1}(x)$

is the inverse normal cumulative distribution function, which assumes a right-sided tail.

In Microsoft Excel:

$\Phi^{-1}(x) = -\text{NORMSINV}(x)$

When displaying long-term metrics, Z_{LT} appears as Zlt-total. This metric is equal to Z-LSL if there is only a lower specification limit or Z-USL if there is only an upper specification limit.

For short-term data, $Z_{LT} = Z_{ST}$ - ZScoreShift. When displaying short-term metrics, Z_{LT} appears in the capability metrics table as Zlt.

Note: Z scores typically are reported only for normal data. The maximum value for Z scores calculated by Crystal Ball from forecast data is 21.18.

Zlt-total

For long-term metrics when both specification limits are defined, the number of standard deviations between the long-term forecast mean and the lower boundary of combining all defects onto the upper tail of the normal curve. Also equal to Zst-total minus the Z-score shift value if short-term metrics are calculated.

When long-term metrics are calculated, Zlt-total is equivalent to Z_{LT}, described in the previous section.

Note: Z scores typically are reported only for normal data.

p(N/C)-below

Probability of a defect below the lower specification limit; DPU_{BELOW}.

Formula:

$$p(N/C)_{BELOW} = \Phi(Z_{LSL})$$

where F is the area beneath the normal curve below the LSL, otherwise known as unity minus the normal cumulative distribution function for the LSL (assumes a right-sided tail).

In Microsoft Excel:

$$\Phi(Z) = 1 - NORMSDIST(Z)$$

p(N/C)-above

Probability of a defect above the upper specification limit; DPU_{ABOVE}.

Formula:

$$p(N/C)_{ABOVE} = \Phi(Z_{USL})$$

where F is the area beneath the normal curve above the USL, otherwise known as unity minus the normal cumulative distribution function for the USL (assumes a right-sided tail).

In Microsoft Excel:

$$\Phi(Z) = 1 - NORMSDIST(Z)$$

p(N/C)-total

Probability of a defect outside the lower and upper specification limits; DPU_{TOTAL}.

Formula:

$$p(N/C)_{TOTAL} = p(N/C)_{ABOVE} + p(N/C)_{BELOW}$$

PPM-below

Defects below the lower specification limit, per million units.

Formula:

$$PPM_{BELOW} = p(N/C)_{BELOW} \bullet 10^6$$

PPM-above

Defects above the upper specification limit, per million units.

Formula:

$$PPM_{ABOVE} = p(N/C)_{ABOVE} \bullet 10^6$$

PPM-total

Defects outside both specification limits, per million units.

Formula:

$$PPM_{TOTAL} = PPM_{ABOVE} + PPM_{BELOW}$$

LSL

Lower specification limit; the lowest acceptable value of a forecast involved in process capability, or quality, analysis. User-defined by direct entry or reference when defining a forecast.

USL

Upper specification limit; the highest acceptable value of a forecast involved in process capability, or quality, analysis. User-defined by direct entry or reference when defining a forecast.

Target

The ideal target value of a forecast involved in process capability analysis. User-defined by direct entry or reference when defining a forecast.

Z-score Shift

An optional shift value to use when calculating long-term capability metrics. The default, set in the Capability Options dialog box, is 1.5.

Equations and Methods

3

Introduction

This chapter provides formulas for the probability distributions.

Formulas for other statistical terms are included in Chapter 2.

Formulas for Probability Distributions

This section contains the formulas used to calculate probability distributions.

Beta Distribution

Parameters: Minimum value (*Min*), Maximum value (*Max*), Alpha (α), Beta (β)

Formula:

$$f(x) = \left\{ \frac{z^{(\alpha-1)}(1-z)^{(\beta-1)}}{\beta(\alpha,\beta)} \right.$$

for $Min \leq x \leq Max$

where:

$$z = \frac{x - Min}{Max - Min}$$

where:

$$\beta(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$$

and where Γ is the Gamma function.

Method 1: Gamma Density Combination

Comment: The Beta variate is obtained from:

$$Min + \left(\frac{u}{u+v}\right) \times (Max - Min)$$

where $u =$ Gamma $(\alpha, 1)$ and $v =$ Gamma $(\beta, 1)$.

Method 2: Rational Fraction Approximation method with a Newton Polish step

Comment: This method is used instead of Method 1 when Latin Hypercube sampling is in effect.

BetaPERT Distribution

Parameters: Minimum value (*Min*), Most likely value (*Likely*), Maximum value (*Max*)

Formula:

$$f(x) = \frac{(x - Min)^{\alpha-1}(Max - x)^{\beta-1}}{B(\alpha, \beta)(Max - Min)^{\alpha+\beta-1}}$$

for $Min \leq x \leq Max$

where:

$$\alpha = 6\left(\frac{\mu - Min}{Max - Min}\right)$$

$$\beta = 6\left(\frac{Max - \mu}{Max - Min}\right)$$

$$\mu = \frac{Min + 4 \times Likely + Max}{6}$$

and $B(\alpha, \beta)$ is the beta integral.

Binomial Distribution

Parameters: Probability of success in each trial (p), Number of total trials (n)

Formula:

$$P\{x = i\} = \binom{n}{i}p^i(1-p)^{(n-i)}$$

for $i = 0,1,2,...n$; p is greater than 0; 0 is less than n is less than 1,000

where:

$$\binom{n}{i} = \frac{n!}{i!(n-i)!}$$

and $x =$ number of successful trials

Method: Direct Simulation

Comment: Computation time increases linearly with number of trials.

Discrete Uniform Distribution

Parameters: Minimum value (*Min*), Maximum value (*Max*)

Formula:

$$f(x) = \begin{cases} \dfrac{1}{(Max - Min + 1)} & \text{if } Min < x < Max \\ 0 & \text{otherwise} \end{cases}$$

Comment: This is the discrete equivalent of the uniform distribution, described in " Uniform Distribution" on page 38.

Exponential Distribution

Parameters: Success rate (λ)

Formula:

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & \text{if } x \geq 0 \text{ and } \lambda > 0 \\ 0 & \text{if } x < 0 \end{cases}$$

Method: Inverse Transformation

Gamma Distribution

This distribution includes the Erlang and Chi-Square distributions as special cases.

Parameters: Location (*L*), Scale (*s*), Shape (β)

Formula:

$$f(x) = \begin{cases} \dfrac{\left(\dfrac{x-L}{s}\right)^{\beta-1} e^{-\frac{x-L}{s}}}{\Gamma(\beta)s} & \text{if } x > L, \ 0 < \beta < \infty, 0 < s < \infty \\ 0 & \text{if } x \leq L \end{cases}$$

where Γ is the gamma function.

Note: Some textbook Gamma formulas use:

$$s = \frac{1}{\lambda}$$

Method 1:

When β is less than 1, Vaduva's rejection from a Weibull density.

When β is greater than 1, Best's rejection from a t density with 2 degrees of freedom.

When $\beta = 1$, inverse transformation.

Method 2: Rational Fraction Approximation method with a Newton Polish step

Comment: This method is used instead of Method 1 when Latin Hypercube sampling is in effect.

Geometric Distribution

Parameters: Probability of success in each trial (p)

Formula:

$$P\{x = i\} = p(1-p)^i$$

for:

$$i = 0, 1, 2, \ldots, n$$
$$p > 0$$

where x = number of failures before the first success

Method: Inverse Transformation

Hypergeometric Distribution

Parameters:

Number of successful items in the population (N_x), sampled trials (n), population size (N)

Formula:

$$P\{x = i\} = \frac{\binom{N_x}{i}\binom{N - N_x}{n - i}}{\binom{N}{n}}$$

where:

$$\binom{n}{i} = \frac{n!}{i!(n-i)!}$$

for:

$$i = \text{Max}(n - (N - N_x), 0) \ldots \quad \text{Min}(n, N_x)$$

and

$N \leq 1000$

and x = number of successful trials,

so N_x = number of successful items in the population.

Method: Direct Simulation

Comment: Computation time increases linearly with population size.

Logistic Distribution

Parameters:

Mean (μ), Scale (s)

Formula:

$$f(x) = \frac{z}{s(1+z)^2}$$

for:

$-\infty < x < \infty,$
$-0 < s < \infty,$
$-\infty < \mu < \infty$

where:

$$z = e^{-\left(\frac{x - \mu}{s}\right)}$$

Method: Inverse Transformation

Lognormal Distribution

Parameters: Location (L), Mean (μ_{ar}), Standard Deviation (σ_{ar})

Mean =

$$\mu_{ar} = L + e^{\mu_{log} + \frac{\sigma_{log}^2}{2}}$$

Median =

$$L + e^{\mu_{log}}$$

Mode =

$$L + e^{\mu_{log} - \sigma_{log}^2}$$

Translation from arithmetic to log parameters:

L = L; L is always in arithmetic space.

Log mean =

$$\mu_{log} = \ln\left(\frac{\mu_{ar} - L}{e^{\frac{(\sigma_{log})^2}{2}}}\right)$$

Log standard deviation =

$$\sigma_{log} = \sqrt{\ln\left(e^{2 \cdot \ln\frac{\sigma_{ar}}{\mu_{ar} - L}} + 1\right)}$$

where ln = natural logarithm.

Formula:

$$f(x, L, \mu_{log}, \sigma_{log}^2) = \frac{1}{\sigma\sqrt{2\pi}\,(x - L)}\, e^{-[\ln(x - L) - \mu_{log}]^2/2\sigma_{log}^2}$$

for:

$$L < x < \infty, \sigma_{log}^2 > 0$$

Method: Inverse transformation

Translation from log to geometric parameters: L = L

Geometric mean =

$$e^{\mu_{log}}$$

Geometric std. dev. =

$$e^{\sigma_{log}}$$

Translation from log to arithmetic parameters:

L = L

Arithmetic mean =

$$\mu_{ar} = L + e^{\left(\frac{\sigma_{log}^2}{2}\right) + \mu_{log}}$$

Arithmetic variance =

$$\sigma_{ar}^2 = e^{2\mu_{log} + \sigma_{log}^2}\left(e^{\sigma_{log}^2} - 1\right)$$

Maximum Extreme Distribution

The maximum extreme distribution is the positively skewed form of the extreme value distribution.

Parameters: Likeliest (m), Scale (s)

Formula:

$$f(x) = \frac{1}{s} \cdot z \cdot e^{-z}$$

for:

$-\infty < x < \infty,$

$-\infty < m < \infty,$ and

$s > 0$

where:

$$z = e^{\left(\frac{-(x-m)}{s}\right)}$$

Method: Inverse Transformation

Minimum Extreme Distribution

The minimum extreme distribution is the negatively skewed form of the extreme value distribution.

Parameters: Likeliest (m), Scale (s)

Formula:

$$f(x) = \frac{1}{s} \cdot z \cdot e^{-z}$$

for:

$-\infty < x < \infty,$

$-\infty < m < \infty,$ and

$s > 0$

where:

$$z = e^{\left(\frac{x-m}{s}\right)}$$

Method: Inverse Transformation

Negative Binomial Distribution

Parameters: Probability of success in each trial (p), Shape (β)

Formula:

$$P\{x = i\} = \begin{cases} \binom{i-1}{\beta-1} p^{\beta}(1-p)^{i-\beta} & \text{for } p > 0 \text{ and} \\ & i = \beta, \beta+1, \\ & \quad \beta+2, \ldots \\ 0 \end{cases}$$

where:

$$\binom{i-1}{\beta-1} = \frac{(i-1)!}{(\beta-1)!(i-\beta)!}$$

and x = total number of trials required to achieve *Shape* number of successes.

Method: Direct Simulation through summation of Geometric variates

Comment: Computation time increases linearly with Shape.

Normal Distribution

This distribution is also known as the Gaussian distribution.

Parameters:

Mean (μ), Standard Deviation (σ)

Formula:

$$f(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-(x-\mu)^2/2\sigma^2}$$

for:

$-\infty < x < \infty$
$-\infty < \mu < \infty$
$\sigma > 0$

Method 1: Polar Marsaglia

Comment: This method is somewhat slower than other methods, but its accuracy is essentially perfect.

Method 2: Rational Fraction Approximation

Comment: This method is used instead of the Polar Marsaglia method when Latin Hypercube sampling is in effect.

This method has a 7–8 digit accuracy over the central range of the distribution and a 5–6 digit accuracy in the tails.

Pareto Distribution

Parameters:

Location (L), Shape (β)

Formula:

$$f(x) = \begin{cases} \dfrac{\beta \cdot L^{\beta}}{x^{(\beta + 1)}} & \text{if } x \geq L \\[2ex] 0 & \text{if } x < L \end{cases}$$

for $\beta > 0$

Method: Inverse Transformation

Poisson Distribution

Parameters: Rate (λ)

Formula:

$$P\{x = i\} = e^{-\lambda} \frac{\lambda^{i}}{i!}$$

for x and $\lambda \geq 0$

Method: Direct Simulation through Summation of Exponential Variates

Comment: Computation time increases linearly with Rate.

Student's *t*-Distribution

Parameters: Midpoint (m), Scale (s), Degrees of Freedom (d)

Formula:

$$f(z, d) = \frac{\Gamma\left(\dfrac{d + 1}{2}\right)}{\sqrt{d\pi}\,\Gamma\left(\dfrac{d}{2}\right)\left(1 + \dfrac{z^2}{d}\right)^{\frac{(d + 1)}{2}}}$$

where:

$-\infty < x < \infty$, integer
$0 < d \leq 30$
$s > 0$

and where:

$$z = \frac{x - m}{s}$$

and where:

Γ = the gamma function

Triangular Distribution

Parameters: Minimum value (*Min*), Most likely value (*Likeliest*), Maximum value (*Max*)

Formula:

$$f(x) = \begin{cases} \dfrac{h(x - Min)}{Likeliest - Min} & \text{if } Min \leq x \leq Likeliest, \ Min < Max \\[2mm] \dfrac{h(Max - x)}{Max - Likeliest} & \text{if } Likeliest \leq x \leq Max, \ Min < Max \\[2mm] 0 & \text{otherwise} \end{cases}$$

where:

$$h = \frac{2}{Max - Min}$$

Method: Inverse Transformation

Uniform Distribution

Parameters: Minimum value (*Min*), Maximum value (*Max*)

Formula:

$$f(x) = \begin{cases} \dfrac{1}{(Max - Min)} & \text{if } Min \leq x \leq Max, \ Min < Max \\[2mm] 0 & \text{otherwise} \end{cases}$$

Method: Multiplicative Congruential Generator

This routine uses the iteration formula:

$$r \leftarrow (62089911 \bullet r) \bmod (2^{31} - 1)$$

Comment:

The generator has a period of length 2^{31} - 2, or 2,147,483,646. This means that the cycle of random numbers repeats after several billion trials. This formula is discussed in detail in the Simulation Modeling & Analysis and Art of Computer Programming, Vol. II, references in the *Crystal Ball User's Guide* bibliography.

Weibull Distribution

A Weibull distribution with Shape = 2 is also known as the Rayleigh distribution.

Parameters:

Location (*L*), Scale (*s*), Shape (β)

Formula:

$$f(x) = \begin{cases} \left(\dfrac{\beta}{s}\right)\left(\dfrac{x-L}{s}\right)^{\beta-1} e^{-\left(\frac{x-L}{s}\right)^{\beta}} & \text{if } x \geq L, s > 0, \beta > 0 \\ 0 & \text{if } x < L \end{cases}$$

where:

Γ is the Gamma function.

Method: Inverse Transformation

Yes-No Distribution

This distribution is equivalent to the binomial distribution with Trials = 1. For details, see

Custom Distribution

Formula:

The formula consists of a lookup table of single data points, continuous ranges, and discrete ranges. Each item in the table has a distinct probability relative to the other items. In addition, ranges might be positively or negatively sloped, giving values on one side or the other a higher probability of occurring.

Method: Sequential search of relative probabilities table.

Comments:

A Uniform variate is generated in the range (0, total relative probability). A sequential search of the relative probabilities table is then performed. The Inverse Transformation method is used whenever the uniform variate falls within a continuous or discrete range that is sloped in one direction or the other.

Additional Comments

All of the nonuniform generators use the same uniform generator as the basis for their algorithms.

The Inverse Transformation method is based on the property that the cumulative distribution function for any probability distribution increases monotonically from zero to one. Thus, the inverse of this function can be computed using a random uniform variate in the range (0, 1) as input. The resulting values then have the desired distribution.

The Direct Simulation method actually performs a series of experiments on the selected distribution. For example, if a binomial variate is being generated with Prob = .5 and Trials = 20, then 20 uniform variates in the range (0, 1) are generated and compared with Prob. The number of uniform variates found to be less than Prob then becomes the value of the binomial variate.

Distribution Fitting Methods

During distribution fitting, Crystal Ball computes Maximum Likelihood Estimators (MLEs) to fit most of the probability distributions to a data set. In effect, this method chooses values for the parameters of the distributions that maximize the probability of producing the actual data set. Sometimes, however, the MLEs do not exist for some distributions (for example, gamma, beta). In these cases, Crystal Ball resorts to other natural parameter estimation techniques.

When the MLEs do exist, they exhibit desirable properties:

- They are minimum-variance estimators of the parameters.

- As the data set grows, the biases in the MLEs tend to zero.

For several of the distributions (for example, uniform, exponential), it is possible to remove the biases after computing the MLEs to yield minimum-variance unbiased estimators (MVUEs) of the distribution parameters. These MVUEs are the best possible estimators.

4

Default Names and Distribution Parameters

Introduction

This first section of this chapter describes the process Crystal Ball uses to name assumptions, decision variables, and forecasts. The second section shows the values it assigns to each of the distribution types.

Naming Defaults

When defining an assumption, decision variable, or forecast, Crystal Ball uses the following sequence to generate a default name for the data cell:

1. Checks for a range name and, if found, uses it as the cell name.

2. Checks the cell immediately to the left of the selected cell. If it is a text cell, Crystal Ball uses that text as the cell name.

3. Checks the cell immediately above the selected cell. If it is a text cell, Crystal Ball uses that text as the cell name.

4. If there is no applicable text or range name, Crystal Ball uses the cell coordinates for the name (for example, B3 or C7).

Distribution Parameter Defaults

This section lists the initial values Crystal Ball provides for the primary parameters in the Define Assumption dialog:

- "Beta" on page 42

- "BetaPERT" on page 43

- "Binomial" on page 43

- "Custom" on page 43

If an alternate parameter set is selected as the default mode, the primary parameters are still calculated as described below before conversion to the alternate parameters.

Note: Extreme values on the order of 1e±9 or ±1e16 may yield results somewhat different from those listed here.

Beta

If the cell value is 0:

Minimum is -10.00

Maximum is 10.00

Alpha is 2

Beta is 3

Otherwise:

Minimum is cell value - (absolute cell value divided by 10)

Maximum is cell value + (absolute cell value divided by 10)

Alpha is 2

Beta is 3

For out-of-range values, such as ±1e300:

Minimum is 0

Maximum is 1

Alpha is 2

Beta is 3

BetaPERT

If the cell value is 0:

Likeliest is 0

Minimum is -10.00

Maximum is 10.00

Otherwise:

Likeliest is cell value

Minimum is cell value - (absolute cell value divided by 10)

Maximum is cell value + (absolute cell value divided by 10)

Binomial

If the cell value is between 0 and 1:

Probability is the cell value

Trials is 50

If the cell value is between 1 and 1,000 (the maximum number of binomial trials):

Probability (Prob) is 0.5

Trials is cell value

Otherwise:

Probability (Prob) is 0.5

Trials is 50

Custom

Initially empty.

Discrete Uniform

If the cell value is 0 or -1e9:

Minimum is 0

Maximum is 10

Otherwise:

Minimum is cell value - INT (absolute cell value divided by 10)

Maximum is cell value + INT (absolute cell value divided by 10)

Exponential

If the cell value is 0, rate is 1.0.

Otherwise, rate is 1 divided by the absolute cell value.

Gamma

If the cell value is 0:

Location is 0.00

Scale is 1.00

Shape is 2

Otherwise:

Location is cell value

Scale is absolute cell value divided by 10

Shape is 2

Geometric

If the cell value is greater than 0 and less than 1, probability is cell value.

Otherwise, probability is 0.2.

Hypergeometric

If the cell value is greater than 0 and less than 1:

Success is 100 times cell value

Trials is 50

Population size is 100

If the cell value is between 2 and the maximum number of Hypergeometric trials (1,000):

Success is cell value divided by 2 (rounded downward)

Trials is cell value divided by 2 (rounded downward)

Population size is cell value

Otherwise:

Sucess is 50

Trials is 50

Population size is 100

Logistic

If the cell value is 0:

Mean is 0

Scale is 1.0.

Otherwise:

Mean is cell value

Scale is absolute cell value divided by 10

Lognormal

If the cell value is greater than 0:

Mean is cell value

Standard deviation is absolute cell value divided by 10

Otherwise:

Mean is e

Standard deviation is 1.0

Maximum Extreme Value

If the cell value is 0:

Likeliest is 0

Scale is 1

Otherwise:

Likeliest is cell value

Scale is absolute cell value divided by 10

Minimum Extreme Value

If the cell value is 0:

Likeliest is 0

Scale is 1

Otherwise:

Likeliest is cell value

Scale is absolute cell value divided by 10

Negative Binomial

If the cell value is less than or equal to 0:

Probability is 0.2

Shape is 10

If the cell value is greater than 0 and less than 1:

Probability is cell value

Shape is 10

Otherwise, unless the cell value is greater than 100:

Probability is 0.2

Shape is cell value

If the cell value is greater than 100, the shape is 10.

Normal

If the cell value is 0:

Mean is 0

Standard deviation is 1.00

Otherwise, unless the cell value is more than 100:

Mean is cell value

Standard deviation is absolute cell value divided by 10.0

Pareto

If the cell value is between 1.0 and 1,000:

Location is cell value

Shape is 2

Otherwise:

Location is 1.00

Shape is 2

Poisson

If the cell value is less than or equal to 0, the rate is 10.00.

If the cell value is greater than 0 and less than or equal to the maximum rate (1,000), the rate is the cell value.

Otherwise, the rate is 10.00

Student's *t*

If the cell value is 0:

Midpoint is 0

Scale is 1.00

Degrees is 5

Otherwise:

Midpoint is cell value

Scale is absolute cell value divided by 10

Degrees is 5

Triangular

If the cell value is 0:

Likeliest is 0

Minimum is -10.00

Maximum is 10.00

Otherwise:

Likeliest is cell value

Minimum is cell value minus absolute cell value divided by 10

Maximum is cell value plus absolute cell value divided by 10

Uniform

If the cell value is 0:

Minimum is -10.00

Maximum is 10.00

Otherwise:

Minimum is cell value minus absolute cell value divided by 10.0

Maximum is cell value plus absolute cell value divided by 10.0

Weibull

If the cell value is 0:

Location is 0

Scale is 1.00

Shape is 2

Otherwise:

Location is cell value

Scale is absolute cell value divided by 10

Shape is 2

Yes-No

If the cell value is greater than 0 and less than 1, the probability of Yes(1) equals the cell value.

Otherwise, the probability of Yes(1) equals 0.5.

5

Predictor Formulas and Statistics

Introduction

This chapter provides formulas and techniques used in Predictor. It contains these main topics:

- "Time-Series Forecasting Techniques" on page 49
- "Classic Time-series Forecasting Method Formulas" on page 52
- "Error Measure and Statistic Formulas" on page 56
- "Regression Methods" on page 63
- "Regression Statistic Formulas" on page 65

Time-Series Forecasting Techniques

This section discusses statistics related to time-series forecasting techniques available in Predictor:

- "Standard Forecasting" on page 50
- "Holdout Forecasting" on page 50
- "Simple Lead Forecasting" on page 51
- "Weighted Lead Forecasting" on page 51

Related terms:

- **Time series**—The original data, expressed as Y_t

- **Fit array**—A retrofit of the time series, consisting of one-period-ahead forecasts performed from the data of previous periods; expressed as F_t

- **Residual array**—A set of positive or negative residuals, expressed as r_t, and defined as $r_t = Y_t - F_t$

- **RMSE**—Root mean square error for forecasting, calculated as described in "RMSE" on page 57, where n is the number of periods for which a fit is available. RMSE depends on the specific forecasting method and technique.

- **Forecasts**—Value projections calculated using the formula for the specific method; they are 1 to k periods ahead, where k is the number of forecasts required

- **Standard error of forecasts**—Used to calculate confidence intervals; see "Confidence Intervals" on page 58

Standard Forecasting

In standard forecasting, if the method parameters are already provided by the user, the following are calculated: RMSE and other error measures, forecasts, and standard error. If the parameters are not provided by the user, then the parameters are optimized to minimize the error measure between the fit values and the historical data for the same period.

Holdout Forecasting

In holdout forecasting:

- The last few data points are removed from the data series. The remaining historical data series is called in-sample data, and the holdout data is called out-of-sample data. Suppose p periods have been removed as holdout from a total of N periods.

- Parameters are optimized by minimizing the fit error measure for in-sample data. If method parameters are provided by the user, those are used in the final forecasting.

- After the parameters are optimized, the forecasts for the holdout periods (p periods) are calculated.

- The error statistics (RMSE, MAD, MAPE) are out-of-sample statistics, based on only the numbers in the hold-out period. The RMSE for holdout forecasting is often called holdout RMSE. The holdout error measures are the ones reported to the user and are used to sort the forecasting methods.

- Other statistics such as Theil's U, Durbin-Watson, and Ljung-Box are in-sample statistics, based on the non-holdout period.

- Final forecasting is performed on both the in-sample and out-of-sample periods (all N periods) using the standard technique.

- The standard error for the forecasts is also calculated using all N periods.

To improve the optimized parameter values obtained for the method, holdout forecasting should be used only when there are at least 100 data points for non-seasonal methods and 5 seasons for

seasonal methods. For best results, use no more than 5 percent of the data points as holdout, no matter how large the number of total data points.

Simple Lead Forecasting

Simple lead forecasting optimizes the forecasting parameters to minimize the error measure between the historical data and the fit values, both offset by a specified number of periods (lead). Use this forecasting technique when a forecast for some future time period has more importance than forecasts for previous or later periods.

For example, suppose a company must order extremely expensive manufacturing components two months in advance, making the forecast for two months out the most important. In this case, the company could use simple lead forecasting with a lead of 2 periods.

In simple lead forecasting:

- The fit for period t is calculated as the (lead)-period-ahead forecast from period $t = 0$. The fit for $t = 1$ calculated with simple lead forecasting is the same as the fit for the standard forecast, which is a 1-period-ahead forecast from period $t = 0$.

- The residual at period t is calculated as the difference between the historical value at period t and the lead-period-fit obtained for period (t).

- The lead RMSE is calculated as the root mean square of the residuals as calculated previously.

- The forecasts for future periods and the standard errors for those forecasts are calculated as for standard forecasts.

If the method parameters are already provided by the user, simple lead forecasting is performed as described previously. If the parameters are not provided, then the parameters are optimized to minimize the lead error measure (for example, lead RMSE). After the parameters are optimized, the fit and the forecast are then calculated as for standard forecasting method.

Weighted Lead Forecasting

Weighted lead forecasting optimizes the forecasting parameters to minimize the **average** error measure between the historical data and the fit values, offset by 1, 2, and so on, up to the specified number of periods (lead value). It uses the simple lead technique for several lead periods, averages the error measure over the periods, and then optimizes this average value to obtain the method parameters.

Use weighted lead forecasting when the future forecast for several periods is most important. For example, suppose your company must order extremely expensive manufacturing components one and two months in advance, making forecasts for all the time periods up to two months out the most important.

In weighted lead forecasting:

- Simple lead error measures are calculated for lead values from 1 to the specified lead value.

- The weighted lead RMSE is calculated as the average of the simple lead RMSEs starting from lead value = 1 to the specified lead value period. For a lead value of 3, simple lead error measures for 1, 2, and 3 are obtained and then averaged to get the weighted lead RMSE.

- Method parameters are then obtained by minimizing the weighted lead RMSE.

- After the parameters are obtained, forecasts for future periods and the standard errors for those forecasts are calculated as for standard forecasts.

If method parameters are provided by the user, weighted lead forecasting is performed as described previously. If the parameters are not provided, they are optimized to minimize the weighted lead error measure, such as weighted lead RMSE.

After parameters are optimized, the fit and the forecast are then calculated as for the standard forecasting method. For a lead value = 1, weighted lead forecasting is the same as simple lead forecasting and standard forecasting.

Classic Time-series Forecasting Method Formulas

This section provides formulas for the following classic time-series forecasting methods used in Predictor:

- "Classic Nonseasonal Forecasting Method Formulas" on page 52
- "Classic Seasonal Forecasting Method Formulas" on page 54

For ARIMA formulas, see "ARIMA Time-series Forecasting Formulas" on page 61.

Classic Nonseasonal Forecasting Method Formulas

Formulas for the classic nonseasonal time-series forecasting methods:

- "Single Moving Average" on page 52
- "Double Moving Average" on page 53
- "Single Exponential Smoothing" on page 53
- "Double Exponential Smoothing" on page 53

Single Moving Average

Single moving average formulas:

$$F_t = \frac{1}{p} \sum_{k=t}^{t-p+1} Y_k$$

(Fit)

(Forecast for period m) $F_{t+m} = F_t$

where the parameterse are:

p—Order of moving average

Double Moving Average

Predictor uses the following equations for the double moving average method:

(Level) $L_t = 2 * M_t - M_t'$

(Trend) $$T_t = \frac{2}{p-1}(M_t - M_t')$$

(Fit) $F_t = L_{t-1} + T_{t-1}$

(Forecast for period m) $F_{t+m} = L_t + m*T_t$

Where the parameters are:

p—Order of moving average

M_t—First order moving average for period t

M_t'—Second order moving average for period t

Note: First fit is available from period $(2*p-1)$.

Single Exponential Smoothing

Predictor uses the following formulas for single exponential smoothing:

(Initialization) $F_1 = 0$, $F_2 = Y_1$

(Fit) $F_t = \alpha * Y_{t-1} + (1-\alpha) * F_{t-1}$

(Forecast for period m) $F_{t+m} = F_t$

Note: First fit is available from period 2.

Double Exponential Smoothing

Crystal Ball uses Holt's double exponential smoothing formula as follows:

(Initialization) $L_1 = Y_1$, $T_1 = 0$

Level: $L_t = \alpha * Y_t + (1 - \alpha) * (L_{t-1} + T_{t-1})$

Trend: $T_t = \beta * (L_t - L_{t-1}) + (1 - \beta) * T_{t-1}$

Fit: $F_t = \alpha * Y_{t-1} + (1 - \alpha) * F_{t-1}$

Forecast for period m: $F_{t+m} = L_t + m*T_t$

Note: First fit is available from period 2.

Classic Seasonal Forecasting Method Formulas

Formulas for the classic seasonal time-series Predictor forecasting methods:

- "Seasonal Additive Smoothing" on page 54
- "Seasonal Multiplicative Smoothing" on page 54
- "Holt-Winters' Additive Seasonal Smoothing" on page 55
- "Holt-Winters' Multiplicative Seasonal Smoothing" on page 56

Seasonal Additive Smoothing

Crystal Ball uses the following initialization equation for this method:

$$P = \frac{\sum_{t=1}^{s} Y_t}{s}$$

Set $L_t = P$, $S_t = Y_t - P$ for t = 1 to s

Crystal Ball uses the following equations to calculate this method:

(Level) $L_t = \alpha * (Y_t - S_{t-s}) + (1 - \alpha) * L_{t-1}$

(Seasonal) $S_t = \gamma * (Y_t - L_t) + (1 - \gamma) * S_{t-s}$

(Forecast for period m) $F_{t+m} = L_t + S_{t+m-s}$

where the parameters are:

α —Alpha

γ —Gamma

m—Number of periods ahead to forecast

s—Length of seasonality

L_t—Level of the series at time t

S_t—Seasonal component at time t

Note: First fit is available from period $(s + 1)$

Seasonal Multiplicative Smoothing

Crystal Ball uses the following initialization equation for this method:

$$P = \frac{\sum_{t=1}^{s} Y_t}{s}$$

Set $L_t = P$, $S_t = Y_t/P$ for t = 1 to s

Crystal Ball uses the following equations to calculate this method:

(Level) $L_t = \alpha * (Y_t / S_{t-s}) + (1 - \alpha) * L_{t-1}$

(Seasonal) $S_t = \gamma * (Y_t / L_t) + (1 - \gamma) * S_{t-s}$

(Forecast for period m) $F_{t+m} = L_t * S_{t+m-s}$

where the parameters are:

α —Alpha

γ —Gamma

m—Number of periods ahead to forecast

s—Length of seasonality

L_t—Level of the series at time t

S_t—Seasonal component at time t

Note: First fit is available from period $(s + 1)$

Holt-Winters' Additive Seasonal Smoothing

To find the initial values:

Calculate:

$$P = \frac{\sum\limits_{t=1}^{s} Y_t}{s}$$

Set: $L_t = P$, $b_t = 0$, $S_t = Y_t - P$, for $t = 1$ to s

For the remaining periods, use the following formulas:

(Level) $L_t = \alpha * (Y_t - S_{t-s}) + (1 - \alpha) * (L_{t-1} + b_{t-1})$

(Trend) $b_t = \beta * (L_t - L_{t-1}) + (1 - \beta) * b_{t-1}$

(Seasonal) $S_t = \gamma * (Y_t - L_t) + (1 - \gamma) * S_{t-s}$

(Forecast for period m) $F_{t+m} = L_t + m*b_t + S_{t+m-s}$

where the parameters are:

α —Alpha

β —Beta

γ —Gamma

m—Number of periods ahead to forecast

s—Length of the seasonality

L_t—Level of the series at time t

b_t—Trend of the series at time t

S_t—Seasonal component at time t

Note: First fit is available from period $(s + 1)$

Holt-Winters' Multiplicative Seasonal Smoothing

To find the initial values:

Calculate: $P = \dfrac{\sum\limits_{t=1}^{s} Y_t}{s}$

Set: $L_t = P$, $b_t = 0$, $S_t = Y_t/P$, for $t = 1$ to s

For the remaining periods, use the following formulas:

(Level) $L_t = \alpha * (Y_t / S_{t-s}) + (1 - \alpha) * (L_{t-1} + b_{t-1})$

(Trend) $b_t = \beta * (L_t - L_{t-1}) + (1 - \beta) * b_{t-1}$

(Seasonal) $S_t = \gamma * (Y_t / L_t) + (1 - \gamma) * S_{t-s}$

(Forecast for period m) $F_{t+m} = (L_t + m*b_t) * S_{t+m-s}$

where the parameters are:

α −Alpha

β −Beta

γ −Gamma

m—Number of periods ahead to forecast

s—Length of the seasonality

L_t—Level of the series at time t

b_t—Trend of the series at time t

S_t—Seasonal component at time t

Note: First fit is available from period $(s + 1)$

Error Measure and Statistic Formulas

This section provides formulas for the following types of statistics used in Predictor:

Time-Series Forecast Error Measures

Crystal Ball calculates three different error measures for the fit of each time-series forecast. Crystal Ball uses one of these error measures to determine which time-series forecasting method is the best:

RMSE

Root mean squared error is an absolute error measure that squares the deviations to keep the positive and negative deviations from canceling one another out. This measure also tends to exaggerate large errors, which can help when comparing methods.

The formula for calculating RMSE:

$$\sqrt{\frac{\sum_{t=1}^{n} (Y_t - \hat{Y}_t)^2}{n}}$$

where Y_t is the actual value of a point for a given time period t, n is the total number of fitted points, and

$$\hat{Y}_t$$

is the fitted forecast value for the time period t.

MAD

Mean absolute deviation is an error statistic that averages the distance between each pair of actual and fitted data points.

The formula for calculating the MAD:

$$\frac{\sum_{t=1}^{n} |Y_t - \hat{Y}_t|}{n}$$

where Y_t is the actual value of a point for a given time period t, n is the total number of fitted points, and

$$\hat{Y}_t$$

is the forecast value for the time period t.

MAPE

Mean absolute percentage error is a relative error measure that uses absolute values to keep the positive and negative errors from canceling one another out and uses relative errors to enable you to compare forecast accuracy between time-series models.

The formula for calculating the MAPE:

$$\frac{\sum\limits_{t=1}^{n} \left| \frac{(Y_t - \hat{Y}_t)}{Y_t}(100) \right|}{n}$$

where Y_t is the actual value of a point for a given time period t, n is the total number of fitted points, and

$$\hat{Y}_t$$

is the forecast value for the time period t.

Note: If Yt equals zero, Crystal Ball drops the term:

$$\frac{(Y_t - \hat{Y}_t)}{Y_t}$$

Confidence Intervals

The confidence interval defines the range within which a forecasted value has some probability of occurring. Predictor uses an empirical method of calculating confidence intervals, using the standard error of forecasts:

- For an m-period-ahead forecast, the error term $r_t(m)$ is defined as $Y_t - F_t(m)$, where $F_t(m)$ is the m-period-ahead fit for period t.

- The standard error of prediction for an m-period-ahead forecast is then expressed as

$$S(m) = \sqrt{\frac{\sum [r_t(m)]^2}{n}} \qquad \text{where } n \text{ is the number of periods for which } r_t(m) \text{ is defined.}$$

Assuming that forecast errors are normally distributed, the formula for predicting the future value of

$$y_{t+m}$$

at time t within a 95 percent confidence interval is

$$\hat{y}_{t+m}(t) \pm 1.959996 \; S(m)$$

The empirical method is reasonably accurate when historical data amount is sufficiently large.

Time-Series Forecast Statistics

Another statistic calculated for any time-series forecast is Theil's *U*.

Theil's U Statistic

This statistic compares forecasted results with the results of forecasting with minimal historical data.

The formula for calculating Theil's *U* statistic:

$$U = \sqrt{\frac{\sum\limits_{t=1}^{n-1}\left(\frac{\hat{Y}_{t+1} - Y_{t+1}}{Y_t}\right)^2}{\sum\limits_{t=1}^{n-1}\left(\frac{Y_{t+1} - Y_t}{Y_t}\right)^2}}$$

where Y_t is the actual value of a point for a given time period t, n is the number of data points, and

$$\hat{Y}_t$$

is the forecasted value.

Autocorrelation Statistics

Measures of autocorrelation describe the relationship among values of the same data series at different time periods.

The number of autocorrelations calculated is equal to the effective length of the time series divided by 2, where the effective length of a time series is the number of data points in the series without the pre-data gaps. The number of autocorrelations calculated ranges between a minimum of 2 and a maximum of 400.

Autocorrelation formula:

$$r_k = \frac{\sum\limits_{t=k+1}^{n}(y_t - \bar{y})(y_{t-k} - \bar{y})}{\sum\limits_{t=1}^{n}(y_t - \bar{y})^2}$$

where r_k is the autocorrelation for lag k.

Related statistics:

Autocorrelation Probability

Autocorrelation probability is the probability of obtaining a certain autocorrelation for a particular data series by chance alone, if the data were completely random. To calculate autocorrelation probability:

- Calculate the standard error of autocorrelation:

$$SE(r_k) = \frac{1 + 2 \sum_{i=1}^{k-1} r_i^2}{n}$$

  where

  $SE(r_k)$ = standard error of autocorrelation at lag k

  r_i = autocorrelation at lag i

  k = the time lag

  n = number of observations in the time series

  Reference: Hanke et al. *Business Forecasting.* 7th ed. Prentice Hall, 2001. Chapter 3, pg 59–60

- Calculate the t statistic:

$$t = \frac{r_k}{SE(r_k)}$$

- Calculate the p-value from the absolute t statistic; the probability is double the area of $(1 - CDF(t))$

Durbin-Watson Statistic

The Durbin-Watson statistic calculates autocorrelation at lag 1.

The formula for calculating the Durbin-Watson statistic:

$$\frac{\sum_{t=2}^{n} (e_t - e_{t-1})^2}{\sum_{t=1}^{n} e_t^2}$$

where e_t is the difference between the estimated point

$\hat{Y}_i$

and the actual point (Y_i) and n is the number of data points.

Ljung-Box Statistic

This statistic measures whether a set of autocorrelations is significantly different from a set of autocorrelations that are all zero. The formula for calculating the Ljung-Box statistic:

$$Q' = n(n+2) \sum_{k=1}^{h-1} \frac{r_k^2}{(n-k)}$$

where:

Q' is the Ljung-Box statistic; the probability that the set of autocorrelations is the same as a set of autocorrelations which are all 0.

n is the amount of data in the data sample.

h is the size of the set of autocorrelations used to calculate the statistic.

r_k is the autocorrelation with a lag of k.

The size of the set of autocorrelations is equal to one-third the size of the data sample (or 100, if the sample is greater than 300).

ARIMA Time-series Forecasting Formulas

Subtopics

- ARIMA Equations
- Estimation of ARIMA Model Coefficients
- ARIMA Constants
- Stationarity

This topic provides basic formulas for the ARIMA (autoregressive integrated moving average) model implementation used in Predictor. For more information, see the references in the ARIMA section of "Bibliography" included in the *Oracle Crystal Ball Predictor User's Guide*.

ARIMA Equations

- Equation for a p-th order autoregressive (AR) model — that is, AR(p) model:

$$y_t = C + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \varepsilon_t$$

    Where $\{y_t\}$ is the data on which the ARMA model is to be applied. That means, the series is already power-transformed and differenced, in that order. The parameters ϕ_1, ϕ_2 and so on are AR coefficients.

- Equation for a q-th order moving average (MA) model — that is, MA(q) model:

$$y_t = C + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \ldots - \theta_q \varepsilon_{t-q}$$

Where $\{y_t\}$ is as defined previously and θ_1, θ_2 and so on are MA coefficients.

- Equation for an ARMA(p,q) model:

$$y_t = C + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \ldots - \theta_q \varepsilon_{t-q}$$

Where $\{y_t\}$, ϕ_1, $\phi_2 \ldots$, θ_1, $\theta_2 \ldots$ are as defined previously.

- Equation for a SARMA(p,q)(P,Q) model (seasonal):

$$y_t = C + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{i=1}^{P} \Phi_i y_{t-is} + \varepsilon_t - \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \sum_{i=1}^{Q} \Theta_i \varepsilon_{t-is}$$

Where $\{y_t\}$, $\{\phi\}$, and $\{\theta\}$ are as defined previously, and $\{\Phi\}$ and $\{\Theta\}$ are the seasonal counterparts.

Estimation of ARIMA Model Coefficients

For a given ARIMA model, Predictor uses the unconditional least square method to estimate model coefficients. Instead of using matrix algebra, a simpler iterative scheme is used.

ARIMA Constants

The constant term in an ARIMA equation introduces deterministic trend into the model and extends that trend indefinitely into the future. If the model includes a single difference (either non-seasonal or seasonal) and the constant term is present, the trend is linear, where a twice-differenced model has quadratic trend. The **AutoSelect** setting for ARIMA constants in the Predictor ARIMA Options dialog leaves out the constant term in the model if it contains one or more non-seasonal or seasonal differences.

When the constant term is included in the model, the value of the term is calculated by the following equation:

$$C = \mu \times \left(1 - \sum \phi_i\right)\left(1 - \sum \Phi_i\right)$$

Where ϕ_i are non-seasonal AR coefficients, Φ_i are seasonal AR coefficients, and μ is the mean of the series.

Stationarity

ARIMA time-series forecasting assumes that the time series mean, variance, and autocorrelation are stationary over time. This characteristic is called stationarity. If a time series statistic has nonstationarity, it must be adjusted:

- **Nonstationarity in the mean**—In this case, the mean is not constant but drifts slowly. This can be true for both seasonal and non-seasonal series and is removed by differencing the series. The automatic ARIMA implementation of Predictor determines the amount of non-seasonal differencing required to make a series stationary by using repeated KPSS

(Kwiatkowski-Phillips-Schmidt-Shin) tests with appropriate alpha values. For seasonal series, repeated Canova-Hansen tests with appropriate alpha values are used.

- **Nonstationarity in variance**—In this case, the time series is heteroscedastic; the variance of the data around the mean changes over time. This nonstationarity in variance is removed by applying the Box-Cox transformation, a special type of power transformation:

$$z_t = \frac{y_t^{\lambda} - 1}{\lambda}$$, if lambda is not equal to 0

$$= Ln(y_t)$$, if lambda equals 0

Where the original series is $\{x_t\}$, the transformed series is $\{z_t\}$, and the power transformation constant is lambda (λ).

Predictor determines a suitable value of lambda with an algorithm that uses the seasonality information to divide the dataset into groups, and then tries to fina a lambda value that renders the variance stationary across groups.

For users who want more control over the Box-Cox transformation, Predictor provides commonly used power-transformation options, such as log transformation (lambda = 0) or square-root transformation (lambda = 0.5), and even a custom transformation with a user-selected lambda between –5 and +5 (inclusive). However, Predictor prevents the use of custom lambda values that would result in transformed values being too large or too small.

Regression Methods

Predictor supports two types of multiple linear regression, standard and stepwise (forward and iterative). Some rules:

- Only standard forecasting is used for independent variables.

- Lags can be specified for each independent variable. They must be less than the effective length of the series (not including pre-data gaps).

- The number of historical data points must be greater than or equal to the number of independent variables, counting the included constant.

For details:

-
-

Calculating Standard Regression

Standard regression can be calculated with or without a constant:

-

Standard Regression with a Constant

The regression equation with constant is

$$y_i = b_0 + b_1 x_{1,i} + b_2 x_{2,i} + b_3 x_{3,i} + \ldots + b_m x_{m,i} + \varepsilon$$

This can be written in matrix format as $Y = bX + \in$,

where Y and b are column vectors of dimension n by 1 and X is a matrix of the dimension n by $(m+1)$, where n is the number of observations and m is the number of independent variables. The first column of X is 1, to include the regression constant. It is assumed that $n > m$.

Predictor uses singular value decomposition (SVD) to determine the coefficients of a regression equation. The primary difference between the singular value decomposition and the least squares techniques is that the singular value decomposition technique can handle situations where the equations used to determine the coefficients of the regression equation are singular or close to singular, which happens when performing regression on equations that represent parallel lines or surfaces. In these cases, the least squares technique returns no solution for the singular case and extremely large parameters for the close-to-singular case.

Crystal Ball uses the matrix technique for singular value decomposition. Starting with:

$y = bX$

Following SVD, X can be rewritten:

$X = [U][w][V]$

where U, w, and V are the factor matrices. The matrix w, a square matrix of dimension $(m+1)$ by $(m+1)$, is a diagonal matrix with the singular values (or eigenvalues). U and V are other factor matrices..

The coefficients can then be calculated. For example, the **b** matrix is $b = [V][w]^{-1}[U^T][y]$

The fit vector $(\hat{y})$ is then calculated as $\hat{y} = bX$

For related regression statistics, see "Statistics, Standard Regression with Constant" on page 65.

Standard Regression without a Constant

This case is also known as *regression through origin*.

The regression equation without constant is

$$y_i = b_1 x_{1,i} + b_2 x_{2,i} + b_3 x_{3,i} + \ldots + b_m x_{m,i} + \varepsilon$$

This can be written in matrix format as $Y = bX + \in$, where Y and b are column vectors of dimension n by 1 and X is a matrix of the dimension n by m, where n is the number of observations and m is the number of independent variables. It is assumed that $n > m$.

Here, too, Predictor uses singular value decomposition (SVD) to determine *b*, the coefficients of the regression equation. The only difference between this case and regression with a constant is the dimension of the matrices.

For related regression statistics, see "Statistics, Standard Regression without Constant" on page 68.

Calculating Stepwise Regression

Stepwise regression is described in Appendix C of the *Crystal Ball Predictor User's Guide*. Note that the partial F statistic is only used in calculating stepwise regression. For a discussion, see "Statistics, Stepwise Regression" on page 69.

Regression Statistic Formulas

The statistics used to analyze a regression are different from those used to analyze a time-series forecast. Regression statistics:

- "Statistics, Standard Regression with Constant" on page 65
- "Statistics, Standard Regression without Constant" on page 68
- "Statistics, Stepwise Regression" on page 69

Statistics, Standard Regression with Constant

These statistics describe a standard regression including the constant:

- "ANOVA, Standard Regression with Constant" on page 65
- "R^2, Regression with Constant" on page 66
- "Adjusted R^2, Regression with Constant" on page 66
- "SSE, Regression with Constant" on page 66
- "F, Regression with Constant" on page 67
- "Statistics for Individual Coefficients" on page 67

ANOVA, Standard Regression with Constant

ANOVA (analysis of variance) statistics for standard regression with a constant:

Table 1 ANOVA Statistics, Standard Regression with a Constant

Source	Sum of Squares	Degrees of Freedom	Mean Square	F Ratio
Regression	$SSR = \sum(\hat{y}_i - \bar{y})^2$	m	$MSR = SSR/m$	$F = \dfrac{MSR}{MSE}$

Source	Sum of Squares	Degrees of Freedom	Mean Square	F Ratio
Error	$SSE = \sum (y_i - \hat{y}_i)^2$	$n - m - 1$	$MSE = SSE/(n - m - 1)$	$F = \dfrac{MSR}{MSE}$
Total	$SST = \sum (y_i - \bar{y})^2$	$n - 1$	n/a	n/a

The F statistic follows an F distribution with $(m, n - m - 1)$ degrees of freedom. This information is used to calculate the p-value of the F statistic.

R², Regression with Constant

R^2 is the coefficient of determination. This statistic represents the proportion of error for which the regression accounts.

You can use many methods to calculate R^2. Predictor uses the equation:

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

Adjusted R², Regression with Constant

You can calculate a regression equation by using the same number of data points as you have equation coefficients. However, the regression equation will not be as universal as a regression equation calculated using three times the number of data points as equation coefficients.

To correct the R^2 for such situations, an adjusted R^2 takes into account the degrees of freedom of an equation. When you suspect that an R^2 is higher than it should be, calculate the R^2 and adjusted R^2. If the R^2 and the adjusted R^2 are close, then the R^2 is probably accurate. If R^2 is much higher than the adjusted R^2, you probably do not have enough data points to calculate the regression accurately.

The formula for adjusted R^2:

$$\text{Adjusted } R^2 = \frac{1 - (1 - R^2)\dfrac{n - 1}{n - m - 1}}{}$$

where n is the number of data points and m is the number of independent variables.

SSE, Regression with Constant

SSE (standard error of measurement) is a measure of the amount the actual values differ from the fitted values. The formula for SSE:

$$SSE = \frac{\sum\limits_{i=1}^{n} (y_i - \hat{y}_i)^2}{n - m - 1}$$

where n is the number of data points you have and m is the number of independent variables.

F, Regression with Constant

The F statistic checks the significance of the relationship between the dependent variable and the particular combination of independent variables in the regression equation. The F statistic is based on the scale of the Y values, so analyze this statistic in combination with the p–value (described in the next section). When comparing the F statistics for similar sets of data with the same scale, the higher F statistic is better.

The formula for the F statistic is given in .

Statistics for Individual Coefficients

Following are the statistics for the pth coefficient, including the regression constant:

- "Coefficient" on page 67
- "Standard Error of Coefficient" on page 67
- "t" on page 67
- "p" on page 68

Coefficient

The coefficient of interest is expressed as b_p, the pth component in the b vector.

Standard Error of Coefficient

The standard error of this coefficient is expressed as $se(b_p)$, or

$$S\sqrt{c_{pp}}$$

where S is the standard error of estimate (SSE) and c_{pp} is the diagonal element at (p,p) of the matrix $(X^TX)^{-1}$.

t

If the F statistic in ANOVA and the corresponding p indicate a significant relationship between the dependent and the independent variables as a whole, you then want to see the significance of the relationship of the dependent variable to each independent variable. The t statistic tests for the significance of the specified independent variable in the presence of the other independent variables.

The formula for the t statistic:

$$t = \frac{b_p}{se(b_p)}$$

where bp is the coefficient to check and $se(bp)$ is the standard error of the coefficient.

The *t* statistic ("t" on page 67) follows a *t* distribution with (n – m – 1) degrees of freedom.

Statistics, Standard Regression without Constant

These statistics describe a standard regression including the constant:

ANOVA, No Constant

ANOVA (analysis of variance) statistics for standard regression without a constant:

Table 2   ANOVA Statistics, Standard Regression without a Constant

Source	Sum of Squares	Degrees of Freedom	Mean Square	F Ratio
Regression	$SSR = \sum \hat{y}_i^2$	m	MSR = SSR/m	$F = \dfrac{MSR}{MSE}$
Error	$SSE = \sum (y_i - \hat{y}_i)^2$	$n - m$	MSE = SSE/(n – m – 1)	$F = \dfrac{MSR}{MSE}$
Total	$SST = \sum y_i^2$	n	n/a	n/a

The *F* statistic follows an *F* distribution with $(m, n - m)$ degrees of freedom. This information is used to calculate the *p*-value of the *F* statistic.

R^2, No Constant

R^2 is the coefficient of determination. This statistic represents the proportion of error for which the regression accounts.

You can use many methods to calculate R^2. Predictor uses the equation:

$$R^2 = \frac{SSR}{SST} = \frac{\sum \hat{y}_i^2}{\sum y_i^2}$$

R^2 can be extremely large in cases when the regression constant is omitted, even when the correlation between Y and X is weak. Because it can be meaningless, many applications do not

mention this statistic. Predictor provides this statistic but it is not used for stepwise regression when there is no regression constant.

Adjusted R^2, No Constant

Adjusted R^2 can be calculated for regression without a constant:

Adjusted R^2 =

$$R^2 = \frac{SSR}{SST} = \frac{\sum \hat{y}_i^{\,2}}{\sum y_i^{\,2}}$$

where n is the number of data points and m is the number of independent variables.

Like R^2 for regression without a constant, this is also a very large number without much meaning.

SSE, No Constant

SSE (standard error of measurement) is a measure of the amount the actual values differ from the fitted values. The formula for SSE:

$$SSE = \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n - m}$$

where n is the number of data points you have and m is the number of independent variables.

F, No Constant

The F statistic checks the significance of the relationship between the dependent variable and the particular combination of independent variables in the regression equation. The F statistic is based on the scale of the Y values, so analyze this statistic in combination with the p–value (described in the next section). When comparing the F statistics for similar sets of data with the same scale, the higher F statistic is better.

The formula for the F statistic is given in .

Statistics for Individual Coefficients, No Constant

The statistics for the pth coefficient for regressions without a constant are the same as those for regressions with a constant. See .

Statistics, Stepwise Regression

Stepwise regression is discussed in Appendix C of the *Oracle Crystal Ball, Fusion Edition Predictor User's Guide*. Information about the partial F statistic, not discussed elsewhere, follows.

Partial F Statistic, Stepwise Regression

Predictor uses the *p*-value of the partial F statistic to determine if a stepwise regression needs to be stopped after an iteration. ANOVA (analysis of variance) statistics for standard regression with a constant:

For addition of a variable, the partial F statistic for step *t*, (PF*t*):

$$PF_t = \frac{SSE_{t-1} - SSE_t}{MSE_{t-1}}$$

PF_t follows the F distribution with degrees of freedom equal to (1, Error DF at step *t*). Users provide a maximum *p*-value, below which the variable is added to the regression.

For deletion of a variable, the partial F statistic for step *t*, (PF*t*):

$$PF_t = \frac{SSE_t - SSE_{t-1}}{MSE_{t-1}}$$

PF_t follows the F distribution with degrees of freedom equal to (1, Error DF at step *t*). Users provide a maximum *p*-value, above which the variable is removed from the regression.

Index

error measures
    MAD formula, 57
    MAPE formula, 58
    RMSE formula, 57
exponential distribution
    formula, 31

F

F statistic (regression with constant)
    formula, 67
F statistic (regression without constant)
    formula, 69
forecasting formulas
    Predictor classic , 52
forecasting methods
    classic nonseasonal formulas, 52
    classic seasonal formulas, 54
forecasting techniques
    time-series, 49
formulas
    adjusted R squared (regression with constant), 66
    adjusted R squared (regression without constant), 69
    ARIMA, 61
    autocorrelation, 59
    autocorrelation probability, 60
    beta distribution, 29
    betaPERT distribution, 30
    binomial distribution, 30
    classic nonseasonal Predictor methods, 52
    classic seasonal methods, 54
    classic time-series Predictor methods, 52
    custom distribution, 39
    discrete uniform distribution, 31
    double moving average, 53
    Durbin-Watson, 60
    exponential distribution, 31
    F statistic (regression with constant), 67
    F statistic (regression without constant), 69
    gamma distribution, 31
    geometric distribution, 32
    Holt's double exponential smoothing, 53
    Holt-Winters' additive seasonal smoothing, 55
    Holt-Winters' multiplicative seasonal smoothing, 56
    hypergeometric distribution, 32
    logistic distribution, 33

lognormal distribution, 33
    MAD, 57
    MAPE, 58
    maximum extreme distribution, 35
    minimum extreme distribution, 35
    negative binomial distribution, 35
    normal distribution, 36
    Pareto distribution, 36
    Poisson distribution, 37
    precision control, 19
    R squared (regression with constant), 66
    R squared (regression without constant), 68
    random number, 21
    regression statistic, 65
    regression with constant statistics, 65
    regression without constant statistics, 68
    RMSE, 57
    seasonal additive smoothing, 54
    seasonal multiplicative smoothing, 54
    single exponential smoothing, 53
    single moving average, 52
    SSE (regression with constant), 66
    SSE (regression without constant), 69
    standard regression, 63
    stepwise regression statistics, 69
    Student's t distribution, 37
    t statistic, 67
    Theil's U, 59
    triangular distribution, 38
    uniform distribution, 38
    Weibull distribution, 38

G

gamma distribution
    formula, 31
geometric distribution
    formula, 32

H

Holt's double exponential smoothing
    formulas, 53
Holt-Winters' additive seasonal smoothing
    formulas, 55
Holt-Winters' multiplicative seasonal smoothing
    formulas, 56
how this manual is organized, 7