Design of Experiments

Module Overview

In this module, you learn how to create design matrices, screen factors, and perform regression analysis and Monte Carlo simulation using Mathcad.

Objectives

After successfully completing this module, you will be able to:

- Understand the basics of experimental design.
- Create design matrices.
- Screen factors.
- Perform a regression analysis.
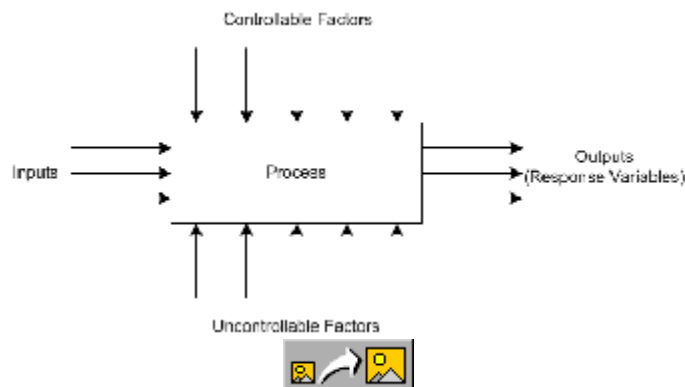- Perform a Monte Carlo simulation.

Introduction to Experimental Design

Design of Experiments (DOE), or Experimental Design, is the design of an experiment, or series or experiments, in which changes are made to the input variables of a process or system and the output response is observed.

The resulting model is used to determine:

- Which input variables have the most effect on the response variables.
- Where to set the controllable factors.

Model of a Process or System

Introduction to Experimental Design

Design of Experiments (DOE), or *Experimental Design*, is the design of an experiment, or series or experiments, in which changes are made to the input variables of a process or system and the output response is observed. DOE provides a model of system performance. This can then be used

to improve performance, reduce costs, and shorten product development and production time.

Experimental Design can be used when trying to understand the relationship between input factors and response variables, given controllable and uncontrollable process factors. A mathematical model is developed relating input and response variables. This model is then used to determine:

- Which input variables have the most effect on the response variables.
- Where to set the controllable factors so:
  - The output response is always close to the desired value.
  - The variability is small.
  - The effects of the uncontrollable factors are minimized.
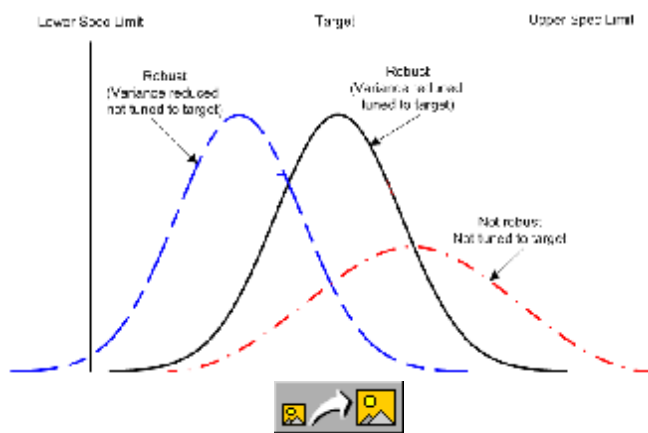
Design of Experiments verses Robust Design

Design of Experiments and Robust Design are techniques that are used to understand the relationship between the inputs and outputs of a process or system.

Design of Experiments

- Plan the experiment.
- Perform the experiment and collect the data.
- Analyze the data using statistical methods.

Robust Design

- System Design
- Parameter Design
- Tolerance Design

Robust Design Methodology

Design of Experiments verses Robust Design

Design of Experiments and Robust Design are techniques used to understand the relationship between the inputs and outputs of a process or system.

Design of Experiments (DOE) is a process for generating data through a series of experiments. This data is then used to evaluate the effect of different parameters on a response variable. In general, the DOE process includes three main steps:

- Plan the experiment.
  - Assess the budget.
  - Identify input and output (response) variables.
  - Assign values to input variables.
- Perform the experiment and collect the data.
- Analyze the data using statistical methods.

Robust Design is also referred to as the *Taguchi Method*. It is a process for making a product, or process, insensitive to the effects of variability or noise. Taguchi's method investigates how different parameters affect the mean and variance of a process performance characteristic that indicates how well the process is functioning. Robust Design uses orthogonal arrays to organize the parameters at various levels. In general, Robust Design includes the following three phases:

- System Design
  - A concept is selected for achieving the desired function.
- Parameter Design
  - This phase determines which combination of parameters will result in the best output using low cost components and tolerances.

ı Tolerance Design
¡ This phase is used to selectively upgrade components and tighten tolerances if the desired output is not in the accepted range.

The orthogonal arrays available in Mathcad are typically used in the Parameter and Tolerance Design phases.
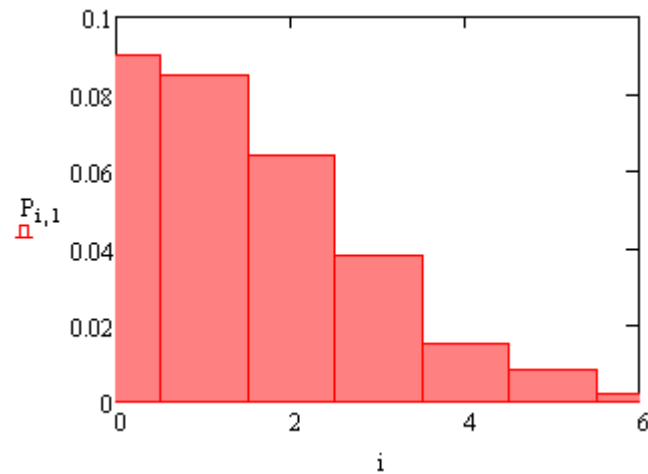
Design of Experiments Functions in Mathcad

Design of Experiments (DOE) functions and plots are used for analyzing the data that results from the experiments performed in a DOE or Robust Design process.

These functions can be used for:

ι Constructing experimental design matrices.
ι Analyzing and screening factors, and plotting results.
ι Developing statistical models.
ι Performing Monte Carlo simulations.

$$X := \text{fractfact}(4,1) = \begin{pmatrix} \text{"Run"} & \text{"Block"} & \text{"A"} & \text{"B"} & \text{"C"} & \text{"D=ABC"} \\ 1 & 1 & -1 & -1 & -1 & -1 \\ 2 & 1 & -1 & -1 & 1 & 1 \\ 3 & 1 & -1 & 1 & -1 & 1 \\ 4 & 1 & -1 & 1 & 1 & -1 \\ 5 & 1 & 1 & -1 & -1 & 1 \\ 6 & 1 & 1 & -1 & 1 & -1 \\ 7 & 1 & 1 & 1 & -1 & -1 \\ 8 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Pareto Plot   Design Matrix

Design of Experiments Functions in Mathcad

05.08.2010

Design of Experiments (DOE) functions and plots are used for analyzing the data that results from the experiments performed in a DOE or Robust Design process. They can be used for:

- Constructing experimental design matrices.
  - Design matrices can be created, analyzed, or modified using Mathcad's built-in functions.
- Analyzing and screening factors, and plotting results.
  - Factors can be screened by calculating their effects, level effects, and interactions. The experimental results can then be plotted using the plot functionality.
- Developing statistical models.
  - Mathcad DOE functions offer different types of regression analysis.
- Performing Monte Carlo simulations.
  - You can generate random numbers for Monte Carlo simulations as well as create a Monte Carlo sample for a given function.

 DOE functions do not accept units.

Design Matrices

A design matrix contains the levels of each factor for all of the run combinations in an experiment.

Design matrices in Mathcad contain:

- Columns specifying:
  - Runs
  - Blocks
  - Factor levels: Coded or Real values
- A header containing the following strings:
  - "Run"
  - "Block"
  - "A," "B," and so on.

$$
\text{Coded} = \begin{pmatrix}
\text{"Run"} & \text{"Block"} & \text{"A"} & \text{"B"} \\
1 & 1 & -1 & -1 \\
2 & 1 & -1 & 1 \\
3 & 1 & 1 & -1 \\
4 & 1 & 1 & 1
\end{pmatrix}
\qquad
\text{Real} := \begin{pmatrix}
\text{"Run"} & \text{"Block"} & \text{"A"} & \text{"B"} \\
1 & 1 & 10 & \text{"high"} \\
2 & 1 & 20 & \text{"low"} \\
3 & 2 & 10 & \text{"low"} \\
4 & 2 & 20 & \text{"high"}
\end{pmatrix}
$$

Design Matrix With Coded Values   Design Matrix With Real Values

Design Matrices

A design matrix contains the levels of each factor for all of the run combinations in an experiment. The basic consideration when selecting a design matrix is the number of factors that will be used. When selecting a design matrix, there are two main concerns:

- Whether to use a screening or modeling strategy.
- Whether to use a linear (two levels for each factor) or a non-linear (three levels for each factor) test design.

Design matrices in Mathcad contain:

- Columns specifying:
    - Runs — Identifies the run number for the experiment.
    - Blocks — Identifies the block number for a set of experiments. A block is a set of relatively homogeneous experimental conditions.
    - Factor levels: Coded or Real values — Identifies the level of each factor.
- A header containing the following strings:
    - "Run"
    - "Block"
    - "A," "B," and so on.

Mathcad's built-in functions may be used to create design matrices or they can be created manually. If they are created manually, they must have the correct format.

Design Matrix Functions

Mathcad functions are available to create, analyze, and modify design matrices.

Mathcad design matrix functions:

- Return factorial design matrices.
- Return Box-Behnken or Box-Wilson (Central Composite) design matrices.
- Return Taguchi design matrices.
- Analyze the properties of fractional factorial design matrices.
- Modify design matrices.

$$\text{FullFact} = \begin{pmatrix} \text{"Run"} & \text{"Block"} & \text{"A"} & \text{"B"} \\ 1 & 1 & -1 & -1 \\ 2 & 1 & -1 & 1 \\ 3 & 1 & 1 & -1 \\ 4 & 1 & 1 & 1 \end{pmatrix}$$

Full Factorial Design Matrix

05.08.2010

$$BoxW = \begin{pmatrix} \text{"Run"} & \text{"Block"} & \text{"A"} & \text{"B"} \\ 1 & 1 & -1 & -1 \\ 2 & 1 & -1 & 1 \\ 3 & 1 & 1 & -1 \\ 4 & 1 & 1 & 1 \\ 5 & 1 & 0 & 0 \\ 6 & 1 & 0 & 0 \\ 7 & 1 & 0 & 0 \\ 8 & 1 & 0 & 0 \\ 9 & 1 & 0 & 0 \\ 10 & 1 & 1.414 & 0 \\ 11 & 1 & -1.414 & 0 \\ 12 & 1 & 0 & 1.414 \\ 13 & 1 & 0 & -1.414 \end{pmatrix}$$

$$RD = \begin{pmatrix} \text{"Run"} & \text{"Block"} & \text{"A"} & \text{"B"} & \text{"C"} \\ 1 & 1 & -1 & -1 & -1 \\ 2 & 1 & -1 & 1 & 1 \\ 3 & 1 & 1 & -1 & 1 \\ 4 & 1 & 1 & 1 & -1 \end{pmatrix}$$

Box-Wilson Design Matrix   Taguchi Design Matrix

Design Matrix Functions

Mathcad functions are available to create, analyze, and modify design matrices. Mathcad design matrix functions:

- Return factorial design matrices. The factorial design matrices include:
    - Full Factorial — A design used to represent all possible effects. Mathcad uses two levels

per factor unless otherwise specified. If a number of levels is specified, that number is used for all levels.

- ¡ Fractional Factorial — A design requiring fewer runs that enables the estimation of all linear effects and desired interactions. Mathcad uses two levels per factor and is limited to less than or equal to 26 factors. The number of runs is constrained to a power of two.
- ¡ Plackett-Burman — A design that reduces a large number of factors to a smaller set of important factors for screening. The number of runs is a multiple of four. In Mathcad the number must be between 8 and 48. By definition, the number of factors is equal to the number of runs minus one.

ı Return Box-Behnken or Box-Wilson (Central Composite) design matrices.

- ¡ Box-Behnken — A design matrix used for modeling quantitative factors with three levels. The number of factors that can be used in Mathcad is between 3 and 7.
- ¡ Box-Wilson (Central Composite) — A design matrix used for quantitative factors. Designed to estimate all linear effects, as well as quadratics and two-way interactions. The number of factors that can be used in Mathcad is between 2 and 9.

ı Return Taguchi design matrices.

- ¡ There are two functions available within Mathcad that return Taguchi design matrices. The function taguchi(n,l) returns a design matrix with *n* factors at *l* levels. The function mtaguchi(n1,l1,n2,l2) returns a mixed matrix with *n1* factors at *l1* levels, and *n2* factors at *l2* levels. There are limited types of designs for each matrix. Details of the types of designs allowed can be found in the Mathcad help files under Taguchi Designs.

ı Analyze the properties of fractional factorial design matrices.

- ¡ The functions available in Mathcad calculate the number of runs, the resolution, and the aliased interactions for each factor and interaction specified. In general, the resolution of a design is one more than the smallest order interaction with which some main effect is confounded (aliased).

ı Modify design matrices.

- ¡ The functions available in Mathcad enable you to randomize the runs, replace labeling and coding with actual labels and values, separate the matrix into blocks, and create a foldover design matrix from an existing design matrix.

Factor Screening

Once data has been collected, factor screening is used to reduce a large number of factors to a smaller subset of important factors.
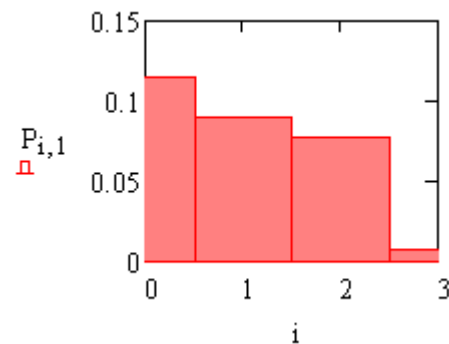
Factor screening functions in Mathcad:

ı Calculate factor effects.
ı Perform analysis of variance on factorial designs.
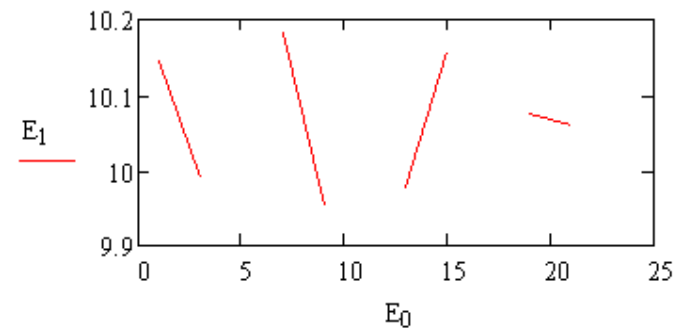ı Plot results.

$$Q := \text{quickscreen}(X, \text{Results})$$

$$Q = \begin{pmatrix} \text{"Factor"} & \text{"(-) Avg"} & \text{"(+) Avg"} & \text{"(+) - (-)"} \\ \text{"A"} & 10.146 & 9.99 & -0.156 \\ \text{"B"} & 10.182 & 9.953 & -0.229 \\ \text{"C"} & 9.978 & 10.157 & 0.179 \\ \text{"D=ABC"} & 10.075 & 10.061 & -0.014 \end{pmatrix}$$

Factor Effects

Pareto Plot

Effects Plot

05.08.2010

Factor Screening

Once data has been collected, factor screening is used to reduce a large number of factors to a smaller subset of important factors. Factor screening functions in Mathcad:

- Calculate factor effects.
  - The quickscreen function returns the mean response for each main factor and interaction specified. This function applies to only two level designs.
  - The effects function returns the level effects associated with each main factor and blocking.
- Perform analysis of variance on factorial designs.

These results can then be plotted using box plots, effects plots, or pareto plots. The important factors are identified using these plots.

Analysis of Variance

Analysis of variance (ANOVA) compares the variability of values within groups to the variability between groups.

Analysis of Variance enables you to test two complementary hypotheses:

- A null hypotheses.
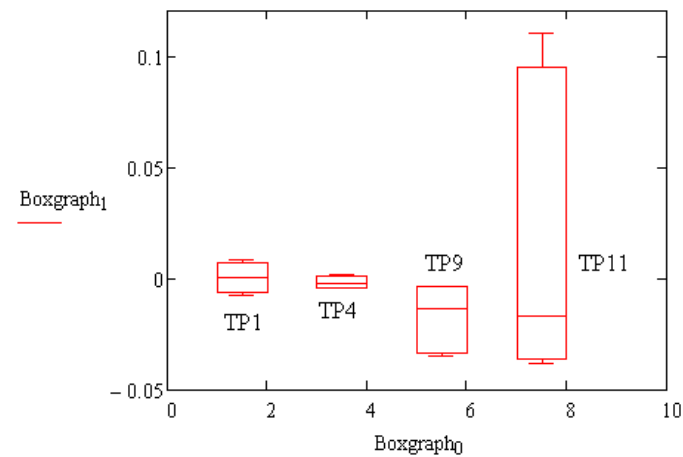- An alternative hypotheses.

A plot can also be created to display the observed data.

$$Y := \begin{pmatrix} 0.0005 & -0.0075 & 0.0006 & 0.0082 \\ -0.0017 & -0.0032 & -0.0043 & 0.0017 \\ -0.0040 & -0.0038 & -0.0350 & -0.0230 \\ -0.0200 & -0.0380 & -0.0140 & 0.1100 \end{pmatrix}$$

Observed Data

$$AN = \begin{pmatrix} \text{"Source"} & \text{"SSE"} & \text{"DF"} & \text{"MSE"} & \text{"F"} & \text{"P"} \\ \text{"A"} & 1.388 \times 10^{-3} & 3 & 4.627 \times 10^{-4} & 0.38 & 0.687 \\ \text{"Error"} & 0.015 & 12 & 1.219 \times 10^{-3} & \text{NaN} & \text{NaN} \\ \text{"Total"} & 0.016 & 15 & \text{NaN} & \text{NaN} & \text{NaN} \end{pmatrix}$$

Analysis of Variance

Box Plot of Observed Data

if you prefer to read Lecture Notes, Click here

05.08.2010

Analysis of Variance

Analysis of variance (ANOVA) compares the variability of values within groups to the variability between groups. ANOVA is used to determine if the variability between groups is large enough to be confident that the factor, or factors, being tested is significant in predicting the output response and not a random sampling variation. That is, it should be included in the prediction equation for the system.

Analysis of Variance enables you to test two complementary hypotheses:

- A null hypotheses indicating that the mean response is equal at each factor setting and the factor is not significant.
- An alternative hypotheses indicating that the mean response is different at each factor setting and the factor is significant.

The observed data can be plotted using a box plot to get a graphical description of the variability. A box plot displays the minimum, the maximum, the lower and upper quartiles (the 25th and the 75 percentile, respectively), and the median (the 50th percentile) on a rectangular box. The box extends from the lower quartile to the upper quartile, and a line is drawn through the box at the median. Lines (or, *whiskers*) extend from the ends of the box to (typically) the minimum and maximum values.
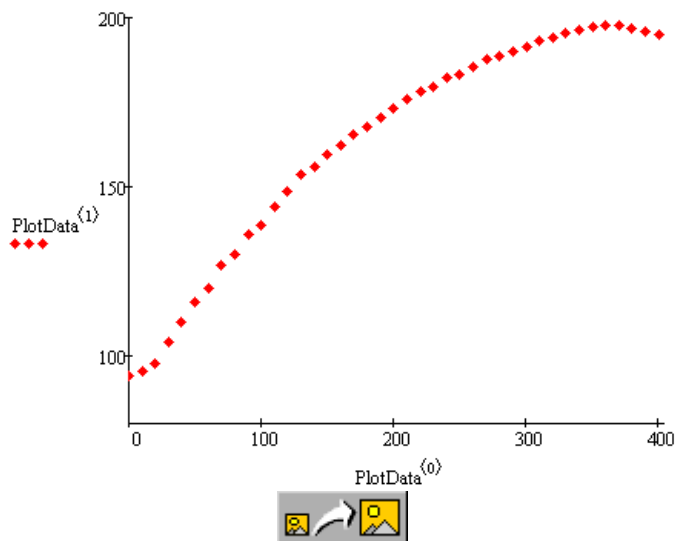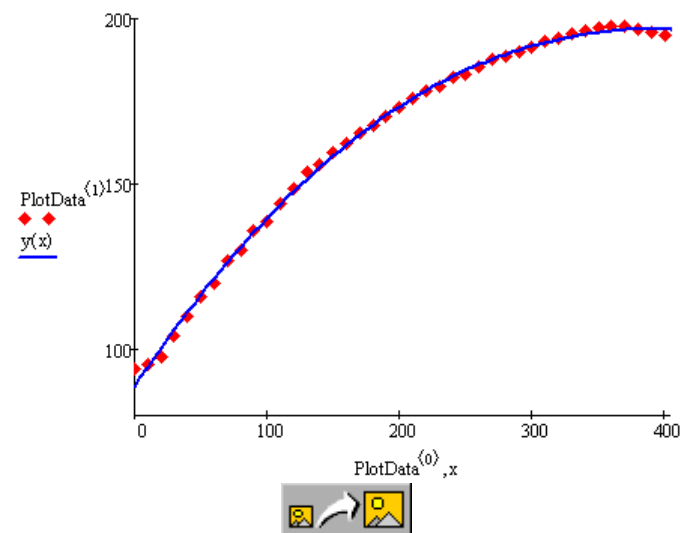
Regression Analysis

Regression analysis determines the mathematical relationship between a measured quantity and the variables it depends on, and enables predictions to be made from it.

After the experiment is designed and the data is collected and inspected, a regression analysis includes:

ı Determining a functional relationship between the inputs and outputs.
ı Testing the accuracy of the functional relationship statistically.
ı Validating the functional relationship determined on the physical system.

Observed Data Plot

Regressed and Observed Data Plot

if you prefer to read Lecture Notes, Click here

05.08.2010

Regression Analysis

Regression analysis determines the mathematical relationship between a measured quantity and the variables it depends on, and enables predictions to be made from it. Regression is used to determine empirical models for a system.

After the experiment is designed and the data is collected and inspected, a regression analysis will include:

- Determining a functional relationship between the inputs and outputs. This is based on a knowledge of physics, an appropriate ideal function, and the form of the data when plotted.
- Testing the accuracy of the functional relationship statistically. Analysis of variance techniques can be used here to determine if the relationship selected is significant.
- Validating the functional relationship determined on the physical system.

The output of the regression analysis is a function that analytically relates the inputs to the outputs.
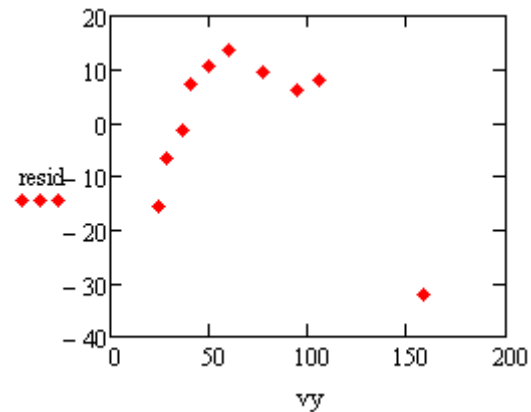
Regression Analysis Functions

Mathcad provides functions for performing a regression analysis.

The following functions are available in Mathcad:

- polyfit
- polyfitc
- polyfitstat
- multidfit

Residual Plot

$x := polyfitstat(vx, vy, 1)$

$$
x_{8,1} = \begin{pmatrix}
\text{"Source"} & \text{"SSE"} & \text{"DF"} & \text{"MSE"} & \text{"F"} & \text{"P"} \\
\text{"Regression"} & 1.404 \times 10^4 & 2 & 7.02 \times 10^3 & 26.426 & 6.398 \times 10^{-5} \\
\text{"A"} & -5.821 \times 10^{-11} & 1 & -5.821 \times 10^{-11} & -2.191 \times 10^{-13} & 0 \\
\text{"B"} & -2.183 \times 10^{-11} & 1 & -2.183 \times 10^{-11} & -8.217 \times 10^{-14} & 0 \\
\text{"Residual} \quad \text{"} & 1.859 \times 10^3 & 7 & 265.638 & \text{NaN} & \text{NaN} \\
\text{"} \quad \text{Lack of Fit"} & 1.859 \times 10^3 & 7 & 265.638 & \text{NaN} & \text{NaN} \\
\text{"} \quad \text{Pure Error "} & 0 & 0 & 0 & \text{NaN} & \text{NaN} \\
\text{"Total} \quad \text{"} & 1.59 \times 10^4 & 9 & \text{NaN} & \text{NaN} & \text{NaN}
\end{pmatrix}
$$

Polyfitstat Function Results — ANOVA Matrix

Regression Analysis Functions

Mathcad provides functions for performing a regression analysis. The following functions are available:

- polyfit
  - The polyfit function defines a function that describes a multivariate polynomial regression surface fitting the results recorded in a matrix of experimental data. The polynomial can be defined by the order n, or by its terms that can be specified in a matrix or a string.
- polyfitc
  - The polyfitc function returns the regression coefficients for a multivariate polynomial regression surface fitting the results recorded in a matrix of experimental data. The polynomial can be defined by the order n, or by its terms that can be specified in a matrix or a string. A confidence interval may also be specified. If it is not specified, a default of 95% is used.
- polyfitstat
  - The polyfitstat function returns statistical data on a poynomial regression model, including an ANOVA matrix for the regression model.
- multidfit
  - The multidfit function returns the multivariate fit parameters for an arbitrary function F, fitting the results recorded in a matrix of experimental data.
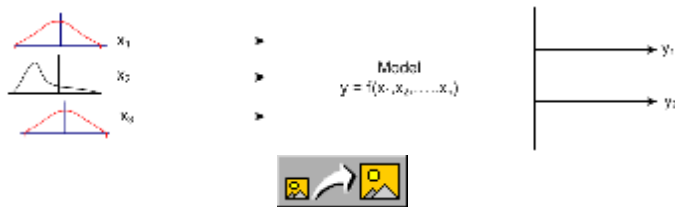
Monte Carlo Simulation

Monte Carlo Simulation is a computer-experimental method that uses random numerical inputs to estimate distributions of simulated outputs.

To perform a Monte Carlo Simulation, the following is required:

- A transfer function.
- Descriptive statistics of the input factors.

Model of Monte Carlo Simulation

Monte Carlo Simulation

Monte Carlo Simulation is a computer-experimental method that uses random numerical inputs to estimate distributions of simulated outputs.

To perform a Monte Carlo Simulation, the following is required:

- A transfer function.
    - A transfer function is a mathematical model of a system. The arguments of the function are the input factors. The output of the function is a desired response.
- Descriptive statistics of the input factors.

ᵢ The descriptive statistics used can include the mean, the variance, and the distribution of the input factors. The descriptive statistics of the inputs are then used to estimate the descriptive statistics of the outputs.

Monte Carlo Simulation involves the following steps:

1. Create a transfer function, $y = f(x_1, x_2, \ldots x_n)$.

2. Generate a set of random inputs, $x_{i1}, x_{i2}, \ldots x_{in}$.

3. Evaluate the transfer function and store the result, y_i.

4. Repeat steps 2 and 3. The number of simulations is typically limited by how long the simulation takes and the computer memory.

5. Analyze the results. The mean, variance, and distribution can now be calculated from the simulated results, y_i.
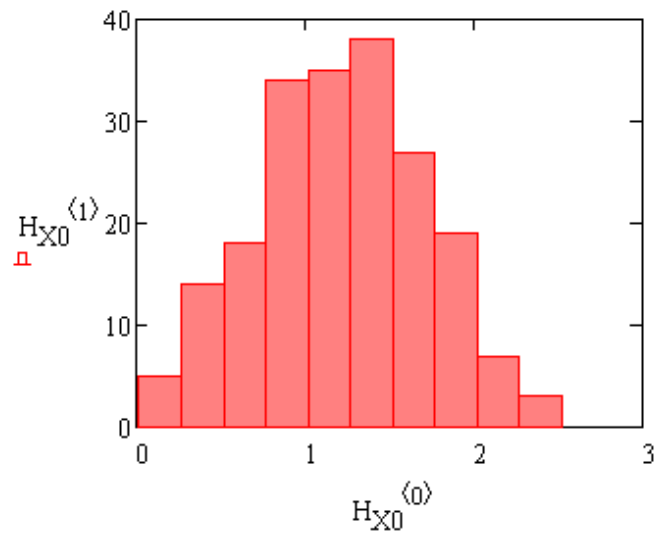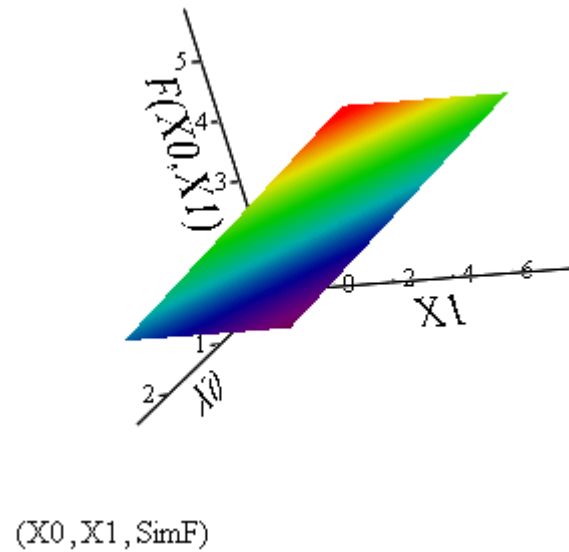
Monte Carlo Simulation Functions

Mathcad provides functions that enable you to create simulated data sets using Monte Carlo Simulation.

The functions Mathcad provides enable you to:

ι Create a Monte Carlo simulation for a given function.
ι Generate random numbers for Monte Carlo simulations.

Generated Input Data Histogram

Monte Carlo Simulated Results

Monte Carlo Simulation Functions

Mathcad provides functions that enable you to create simulated data sets using Monte Carlo Simulation.

The functions Mathcad provides enable you to:

- Create a Monte Carlo simulation for a given function.
    - Mathcad provides the montecarlo function of the form montecarlo(F,n,Rvals). This function returns n Monte Carlo simulations for the given function F. The variables used in the function F are randomly generated using the mean and standard deviation listed in the matrix Rvals. If no additional information is given, the variables are generated using a normal distribution. Limits and a type of distribution can also be specified if a different distribution is needed.
- Generate random numbers for Monte Carlo simulations.
    - Random numbers can be generated that can then be used in the montecarlo function. The distributions that Mathcad can generate are:
        - Standard log normal
        - Normal
        - Uniform
        - Wiebull