MiSim User Guide

Version 1.0

Author: Santhosh Kumar Kasa &

Sumit Bafna

Contents

[About MiSim: 2](#_Toc407045419)

[Under the Hood: 2](#_Toc407045420)

[Components: 2](#_Toc407045421)

[Single Input Simulation: 4](#_Toc407045422)

[Features: 6](#_Toc407045423)

[Multi-Input Simulation: 9](#_Toc407045424)

# About MiSim:

MiSim is a python based tool that pairs with Microsoft Excel as an Add-In and provides function in order to facilitate stochastic simulations.

It allows the user to model the simulation problem in excel using the custom-made MiSim function. Once the problem is modelled, the user can select the number of runs. The tool then simulates the problem specified number of times and presents the results on a GUI.

# Under the Hood:

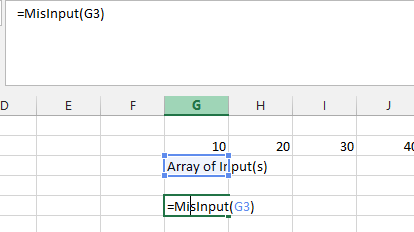
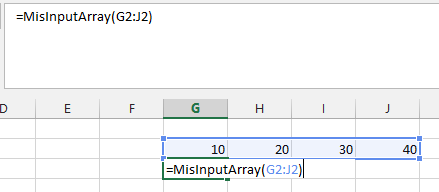
The custom functions are a wrapper around the standard distribution functions in Python. The user uses these custom functions to model the problem at hand. Once the problem is modelled, the python keeps on changing the values in those input (stochastic distribution functions) and collects the output.

In other words, it is similar to having a spreadsheet and refreshing the spreadsheet a certain number of times (number of simulations). Once the output is collected, it is displayed as a frequency distribution on a GUI window.

In order to find out the exact technical details, please contact Prof. [Roehrig](mailto:roehrig@andrew.cmu.edu) or [Santhosh Kasa](mailto:kasa@cmu.edu) for the technical documentation.

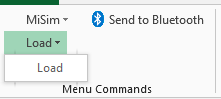
# Components:

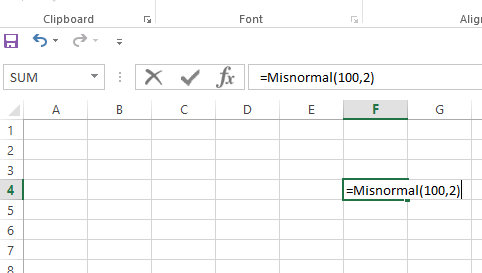
These functions are separated in three categories:

1. **Frequency Distribution Functions:** This class of functions have various frequency distributions like Normal, Poisson, Binomial distribution etc. When simulations are run in MiSim, these functions supply different values to the cell that they are applied to. Values of one or more of these function holding cells can be applied to the input functions (the next class) in order to perform stochastic simulation.
2. **Input Functions:** These are two functions in this class. First is MISInputArray. This function is useful if a comparative analysis is to be done for a discrete set of input values. For example, if there is a need to compare simulation results of for different set of variables, then these variables should be passed to the MISInputArray function. The second function in this class is MISInput Function. This is the function to which we can the MISInputArray cell.   
     
   Now, model the problem with the MisInput cell. The tool will take care of simulating the problem for all the inputs specified in the input array.
3. **Output Function:** the MISOutput function is the cell on which the simulation takes place. Set this function with the cell whose output you want to collect

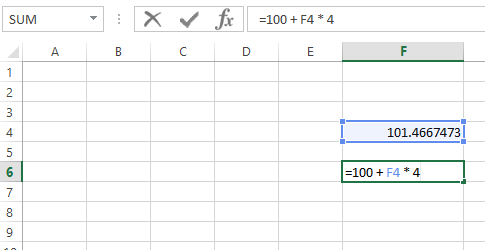
# Single Input Simulation:

Steps to create a simulation for a single input cell:

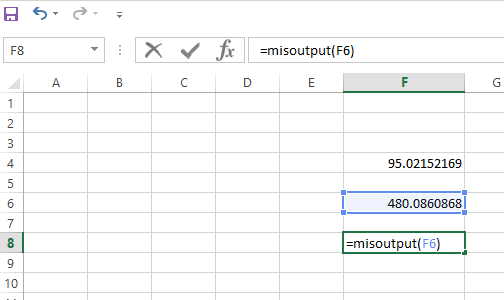
1. Click on Load from the Menu Add-ins Menu.   
   Failure to do this, will result in creating a refresh of all formulas every time an enter is hit on the spreadsheet.  
   
2. Enter a frequency distribution function (eg. Misnormal) with appropriate parameters in any one of the cells.



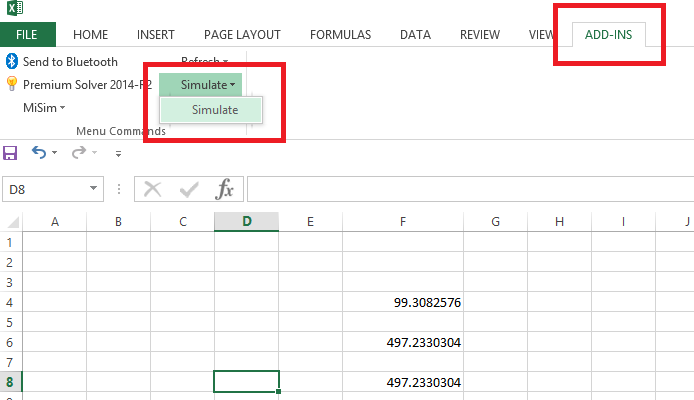
1. Use the cell containing frequency distribution function in arithmetic calculation in any other cell. This is the cell whose value we would like to simulate.



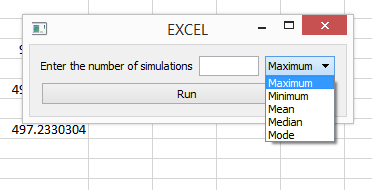
1. In any cell apart from the ones already been used, use the misoutput function and supply the cell that needs to be simulated as a parameter.



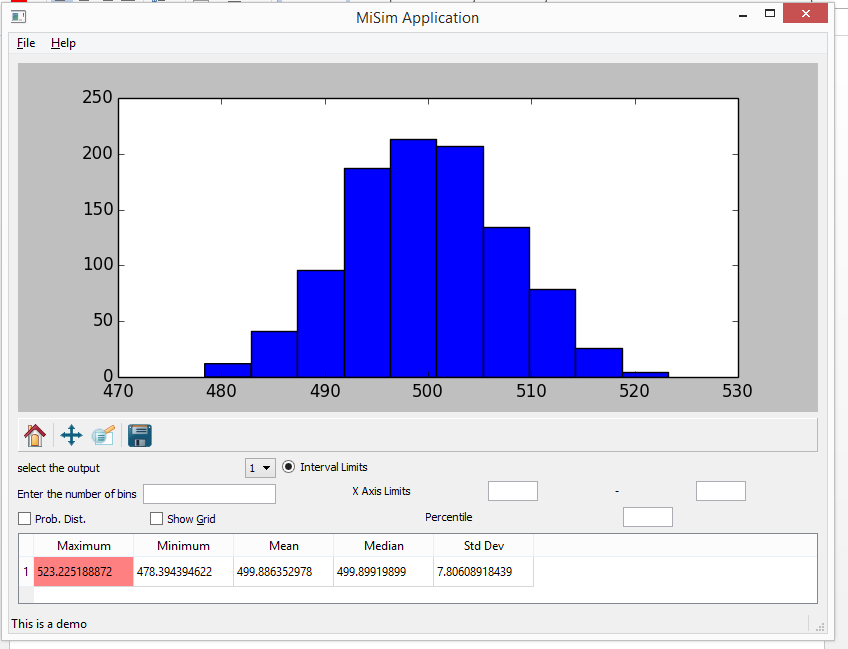
1. On Add-Ins Excel Menu, click on Simulate🡪Simulate



1. Enter number of times you want to simulate the code. Greater this number, more the accuracy. Also choose which simulation parameter you want to highlight in your simulation result.



A screen similar to image below should appear:

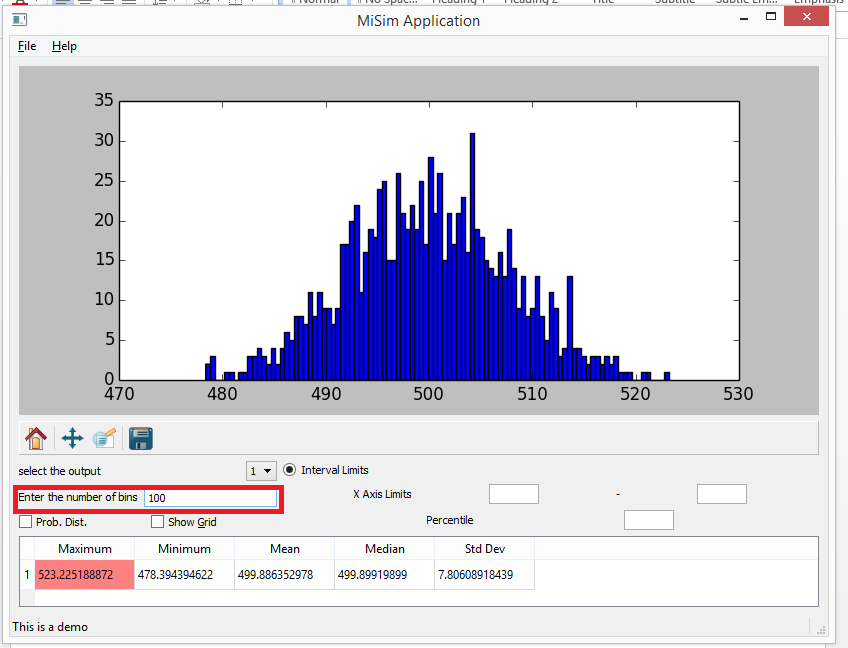


This complete our basic simulation.

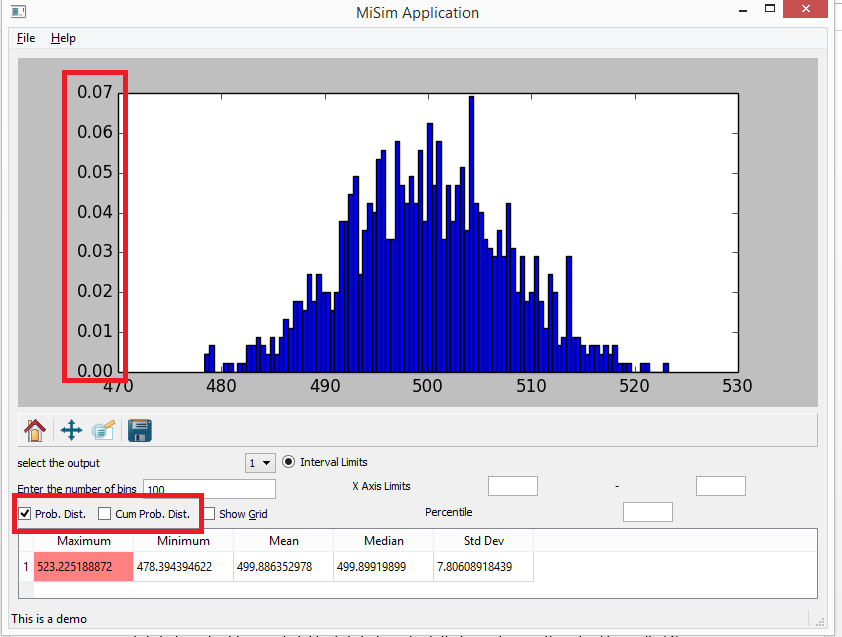
# Features:

Next, we will try to explore some features of the simulation UI.

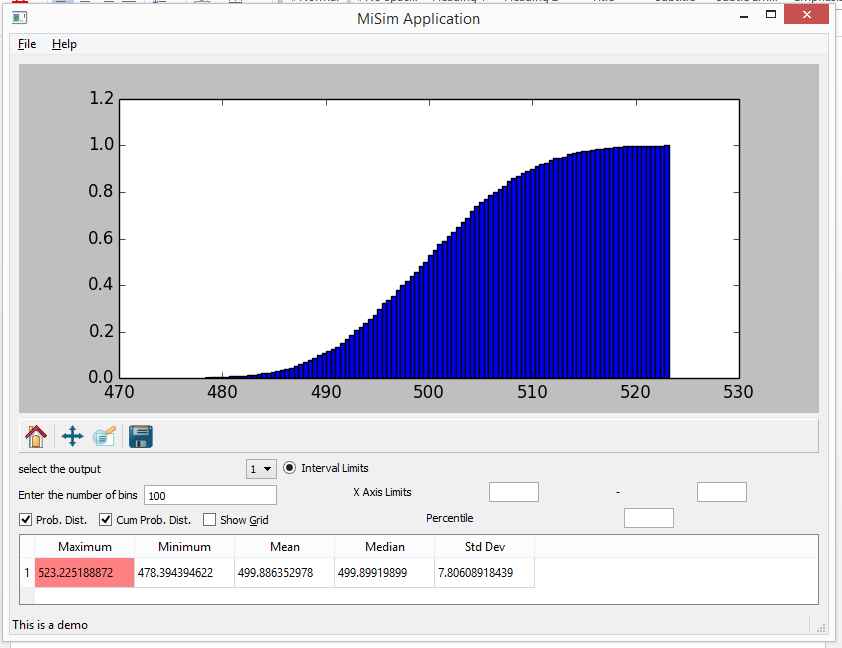
1. Notice a textbox called Enter the number of bins. This decides precision on the generated graph. Enter value 100 in this textbox and hit enter and notice increased precision on the graph.



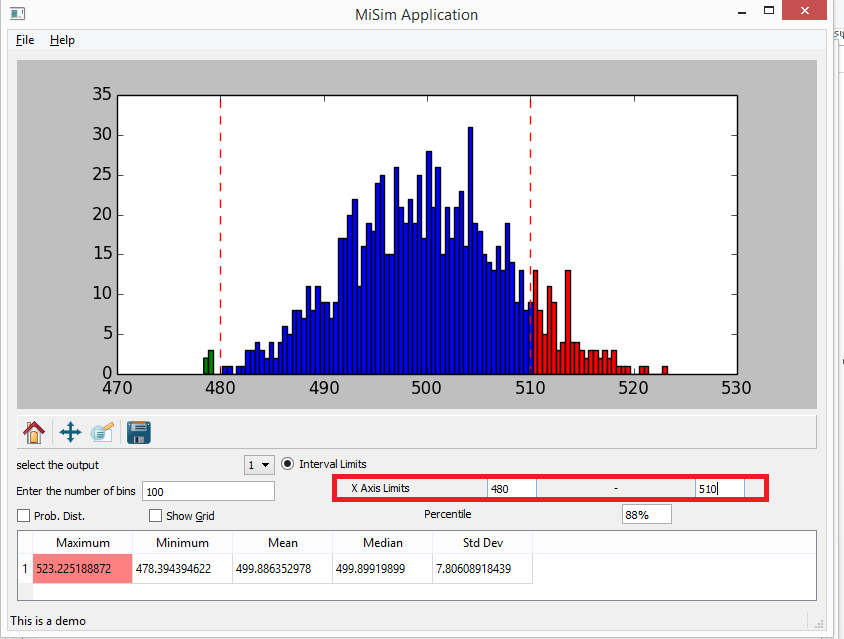
2. Notice check box Prob. Dist. Clicking on this checkbox changes your graph from a frequency distribution chart to a probability distribution chart. It also makes another checkbox called Cum. Prob. Dist. visible. Also notice the that y-axis changes to represent probability distribution.

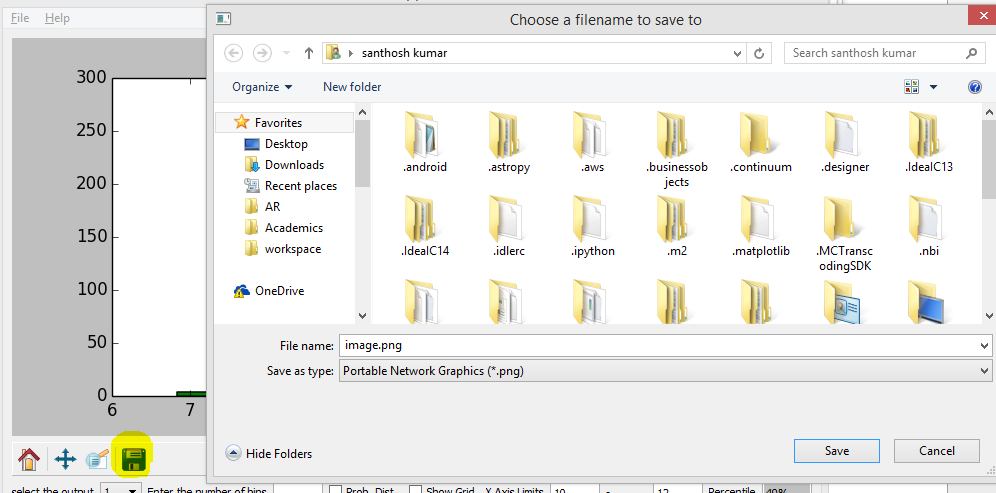


3. Click on Cum. Prob. Dist. and a cumulative probability distribution chart will be visible to represent same simulation result.

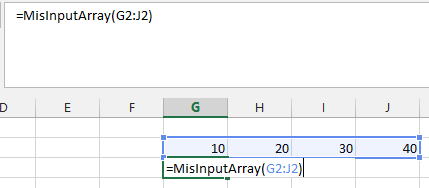
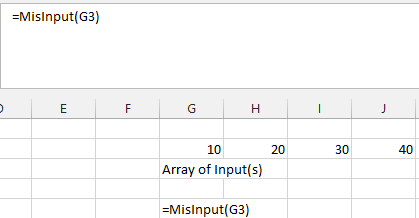
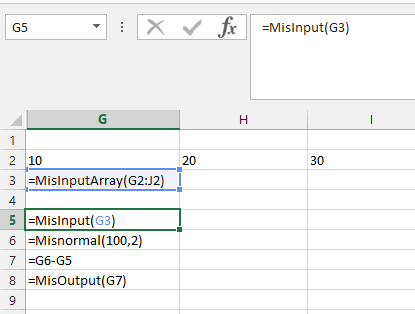


4. Next notice X Axis Limits textboxes. Enter some values in those boxes. I entered range 480 to 510 here and noticed that 88% of results of the simulation actually lie in this range which is visible in Percentile textbox.



5. You can save the figure by clicking on the Save button.  


# Multi-Input Simulation:

1. Specify the set of inputs in different contiguous cells.
2. Use MisInputArray to specify the array of cells  
   
3. Use MisInput and pass the array as a input parameter.  
   
4. Now, model the rest of the problem. In this case, I am using a normal distribution and will subtract the input from the normal distribution.  
   
5. Once I run the simulation with 100 runs, the output looks as below. You can see that the output can be selected from 1-4 and the entire statistics of the output is presented in a table.  
   