MATLAB STATFILE_ANALYSIS WALKTHROUGH

By: Connor Smith

A MANUAL EXPLAINING HOW TO USE THE STATFILE_ANALYSIS SCRIPT FOR ALL STATISTICAL NEEDS

Manual Contents

MAIN MENU

The main menu contains 11 options for you to follow. To perform one of the actions, simply type the number of the menu number which you want to do.

Any other input will simply bring up the main menu page again.

SET USER NAME AND OUTPUT FILE NAME

The default User Name is ‘user’ and the default Output file name is ‘empty’. Both must be changed before printing the output file.

PLOT HISTOGRAM, HISTOGRAM FIT, & PROBABILITY PLOTS

To plot a histogram, the program ensures that data is loaded. Otherwise, no histogram can be plotted. Once that is ensured, the program asks for how many bins are preferred for the histogram. If no amount is entered, the default Matlab-determined amount will be used. Finally, the histogram will be displayed. Again, data must be loaded to plot the probability plot. If there is data, a norm plot is generated from the data.

DO NOT CLOSE ANY PLOTS*

BASIC STATISTICAL DATA

Before outputting any statistics, the program ensures that the user has entered a user name, output file name, and determined if the data seems to be normal. If any of these 3 conditions are not true, the user must enter this information before continuing. The reasoning for this, is to ensure a fully accurate data file can be created.

After this information is secured, all general statistical information and other related information is displayed in the console. Additionally, a .txt file with the given output name is created and

*Visit General Information/Warnings for more information

saved containing all of this information. More information about the outputted information can be found in General Information/Warnings*

FIND PROBABILITY GIVEN X OR Z AND FIND X OR Z GIVEN PROBABILITY

Unless the user has already determined whether the data is normal, the program will ask the user to determine whether the data seems to be normal or not. The user can determine this from the norm plot, and histogram plots. Unfortunately, once this section is opened, the user must give an answer. Changing a given answer is explained in General Information/Warnings*. Luckily, the answer does not change any of the results. Simply if a no is answered, a warning is displayed with each result, reminding the user of the unreliability of these calculations.

Finding probability from x/z

After the normality of the data is determined, the program asks for a 'x' or a 'z'. Only one can be input. From there, it uses the x/z value (Dependent on which you input) and finds the probability using the normcdf function on Matlab.

Finding x/z from probability

After the normality of the data is determined, the program asks if the user wants to solve for a 'x', 'z', or 'both'. Next, the program asks for a probability. The x and z values are calculated from norminv on Matlab and the formula: z = ((x – mean)/standard deviation) From there, depending on which options were chosen, the results will be shown.

GENERAL INFORMATION/WARNINGS

In regards to images, each plot is displayed on the same image box. In turn, if this image box is closed, the new plots will not appear. Then, either the image box will need to be opened

*Visit General Information/Warnings for more information

manually through matlab, or the program must be restarted.As for changing whether the data is normal or not, the user must clear data and reload it. Either overwriting the file, or clearing then loading will work. Then the new data will be evaluated, and the user can change the answer as to whether the data is normal or not.

Following are the outputted statistics and information from the Basic Statistical Data section:

[Username, Output File Name, Normality of Data (According to User), Standard Deviation Used, Mean, Median, Mode, Var, StDev, Min, Max, Count]

TESTING

Some of this has been covered in previous sections, but my testing process was fairly algorithmic. For the data, I inputted the "CIData.txt" from a class activity, to ensure that it could handle long inputs. Additionally, this data was normal, so I used it to ensure that all of my plots were accurate. After testing the entire program with this data set, I created a short 2x4 array of random numbers to ensure this program could take 2 column arrays. This data was not large enough to be normal, and was less than 30 numbers. So this tested the population standard deviation, and that the histograms displayed this data reasonably. For data possibilities, these 2 files covered every possibility.

The only thing left to test after data was inputs. For each input, I handled it as a string first (to ensure it took every input) then did error checking to evaluate if the input was what was wanted. Knowing this, I tested each input for a string, a number, symbols, whitespace, and just clicking enter. Afterwards, I ensured that the input only accepted values that were needed for the input.

These were the only 2 areas that needed to be error proof, every other area has not interaction with the user.

*Visit General Information/Warnings for more information