Matthew Larson
Project Final Report
Computational Physics
December 13th 2018

ReDFPG Final Report

In high energy particle physics data is commonly stored and processed using a set of programs known as the root data framework. The root data framework is a set of tools written in C++ that can be used to process and handle very large data sets. Though very fast to execute, root macros are very difficult to write and can present some headache. This difficulty stems from the C++ language, which requires data type specifiers, and some more complex formatting which make it in some ways more difficult to read in a syntactically sensible way. Within the root data framework the data which is to be processed is held inside of files known as .root files. These .root files contained categorized information about specific data items. This data is organized into rows and columns respectively corresponding to data items and categories describing those items. The goal of this program is to create a diving board for doing cursory investigations of .root data files and starting the initial process of formatting and fitting root data.

The program created to fulfill this goal shall be referred to henceforth as the Root Data Frame Processing GUI (or ReDFPG for short). RedFPG is a python based program which utilizes a graphical user interface (GUI) to allow a user execute some basic processes typically done within the root frame work. This includes importing, plotting, fitting, and applying cuts to data from root files. This program can be broken down into three main parts. The first of which loads and selects the data to be processed, the next takes the data an performs cuts on it then plots the data, and the final part applies fittings to the data.

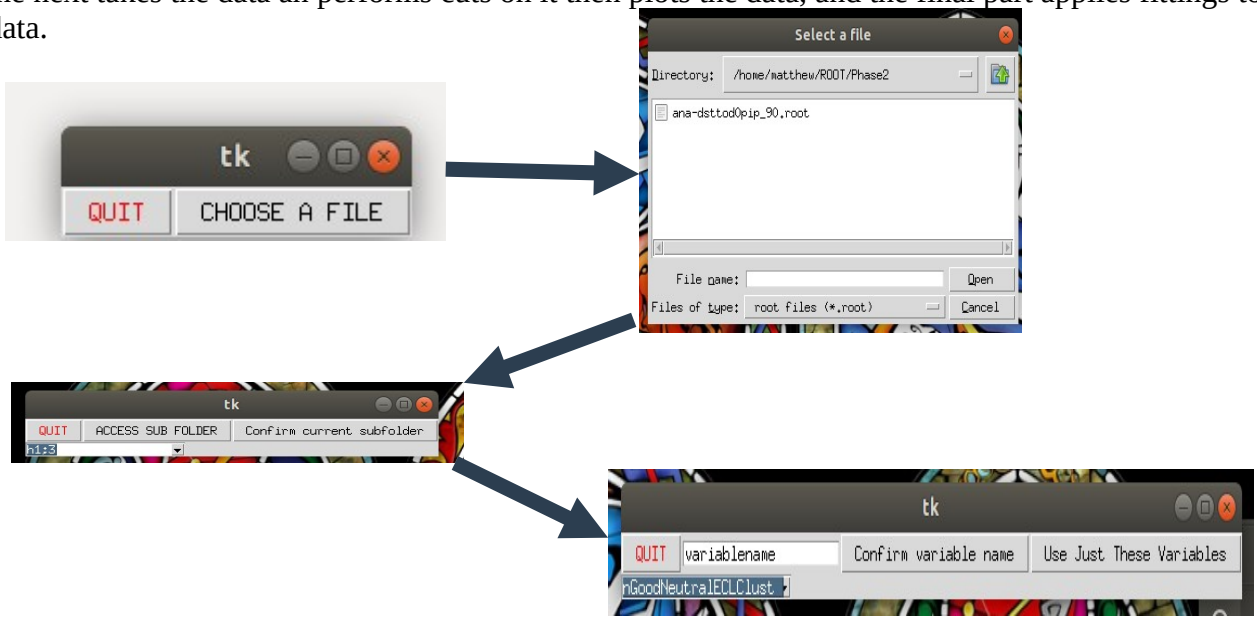

Figure 1 (a-d): This figure shows the steps in the start up process of the ReDFPG program. a) prompts the user to to open a file or quit the program. b) opens an interface showing available root files c) allows the user to select a sub folder from the root file d) allows the user to select a category that will be processed and plotted

The first part of the program that the user interacts with is the interface for loading and selecting the data. In this part of the program the user is prompted through the actions of first opening the loading screen shown in part a of figure 1. This screan presents the user with the option to quit the program or select data. If select data is chosen the user is prompted to select a .root file as shown in

part b of figure 1. If the user selects open after selecting an appropriate .root file the user is brought to the pane in part c of figure 1. From this pane the user is presented with a drop down menu containing the subheads within the root file and two options, Access sub-folder and confirm sub folder. If the items in the drop down menu are the columns of the data set which the user wants to access the user should select the confirm option. If not the user should select the correct sub-folder and hit the access button until the correct columns are presented. Once the Confirm button is selected a new window is opened with different buttons as shown in figure 1 d. In a similar manner a data set is selected but this time it is assigned a call symbol which will be used later in the selection of the data. When a call sign has been assigned and the confirm variable name button has been pressed for all of the data which is desired the "use just these variables" button may be pressed carrying us into the next section of the program.

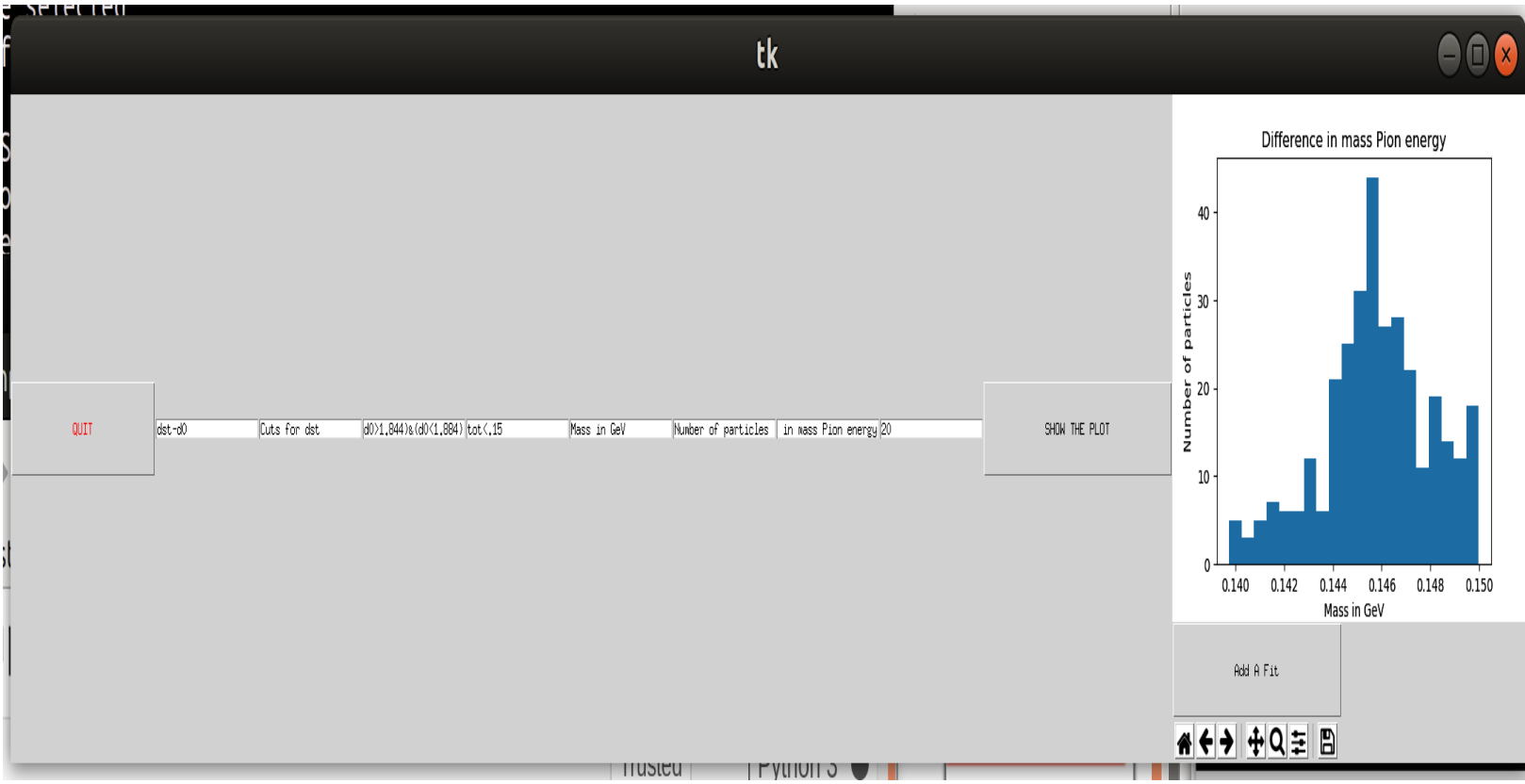

Figure 2: This figure shows the selections available in the second section of the ReDFPG program. In this section the user selects the data to be plotted, applies cuts to that data, sets plot titles and chooses the number of bins to use. The data plotted is based on data collected by the Belle Experiment in 2010.[1]

In the second portion of the ReDFPG program a set of data fields are made available to the user as shown in figure 2 above. It is in this part of the program that the user utilizes the previously assigned variable names from part one. In the first cell the user selects the data that will be plotted. In the next N (N being the number of variables) the cuts that will be applied to each variable are selected. The following cell is the cuts that are made to the total data set. The next selections are then used to select the x y and main axis titles. Finally the last cell selects the number of bins to to use in making the final histogram. The last three objects on the screen tell the program to plot, allow the user to edit and save the plot and add a fitting to the plot. The final button is what brings us to the next section of the code, the fitting.

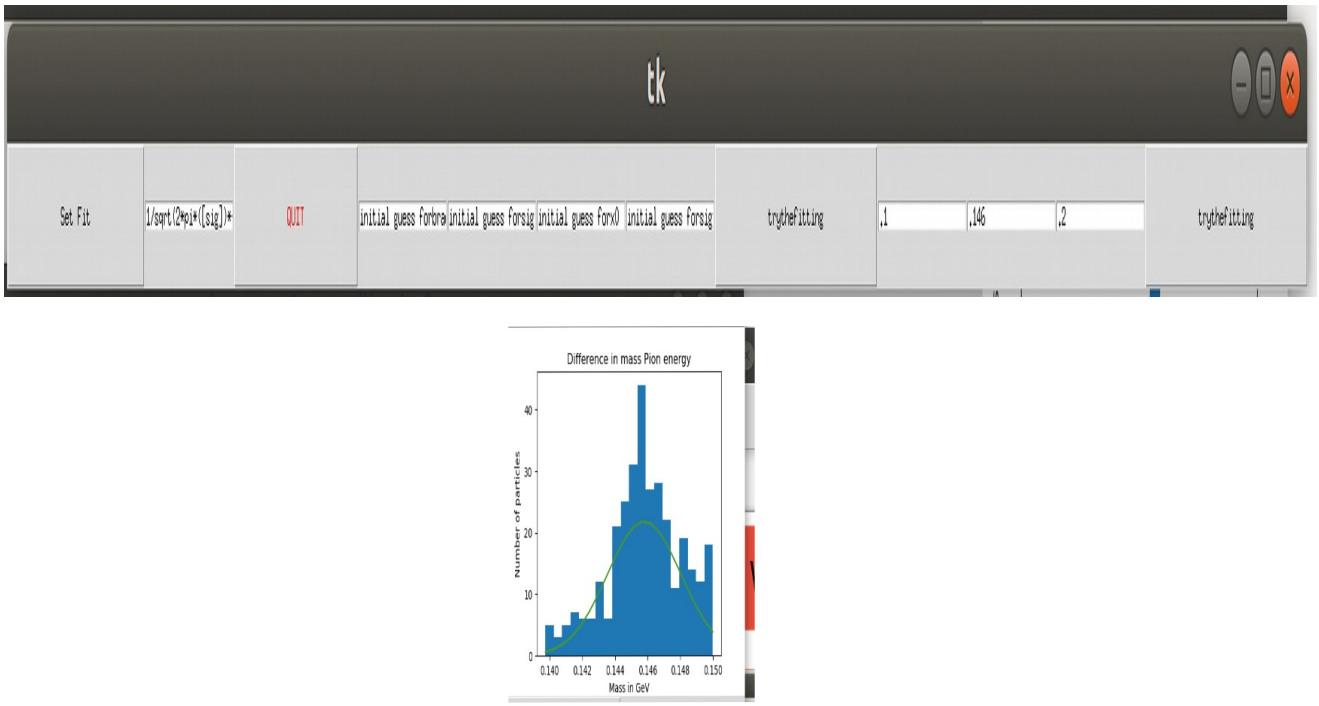

Figure 3a-b: These figures show the interface and output of the third section of the ReDFPG software. Part a shows the selection widget for the fitting with sections for variable guess. Part b shows the final plot of the fitting against an example data set. The data plotted is based on data collected by the Belle Experiment in 2010.[1]

In the third portion of the ReDFPG software the data is fitted based on a user specified function. The first step in this portion is shown in figure 3 part a and includes cells for entering the fit, selecting initial values and finally asking the program to attempt a fitting on the data. When plot the fit is select the program runs a scypy minimize function using the initial variables and a set of intervals determined by the values entered. If the fitting works then the fit is simply added to the plot mentioned in section two and from there the user can adjust the plot as needed and add further fittings.

In the future the plan is to expand upon this program by making some adjustments the first of which would be editing the font on all of the cells so that is slightly more legible on high definition displays. The next plan is to tweak the fitting program so that it is slightly more consistent in selecting the variables and does not end up double dipping on the variables when they are used twice within the same function. Other than this some additional features could be added to the function to append data to .root files and create root macros based on the process developed in this program.

Through the use of this software an efficient method of loading and processing .root data files has been established. This software is capable of sorting through a .root file and selecting a specific data column from within the sub folders of that data set. It can then plot a histogram based on this data and apply cuts to the data. Finally it can use a fitting to investigate the histogram and produce values which can then be used as a way of evaluating the data sets. Some improvements will be made in the future but currently it is a functioning program.

References:

Kinoshita, Kay. “Belle Phase II Data.” KEK , 2010.