Summer 23: Global Fitting Update

Emrys Peets

07/11/2023

U.S. DEPARTMENT OF ENERGY
Office of Science

SLAC NATIONAL ACCELERATOR LABORATORY

Important vocab to keep in mind

- Window Range: Range by which a function is fit, generally of the form [WinMin, WinMax]
- WinMax: Maximum value for a given window range
- WinMin: minimum value for a given window range

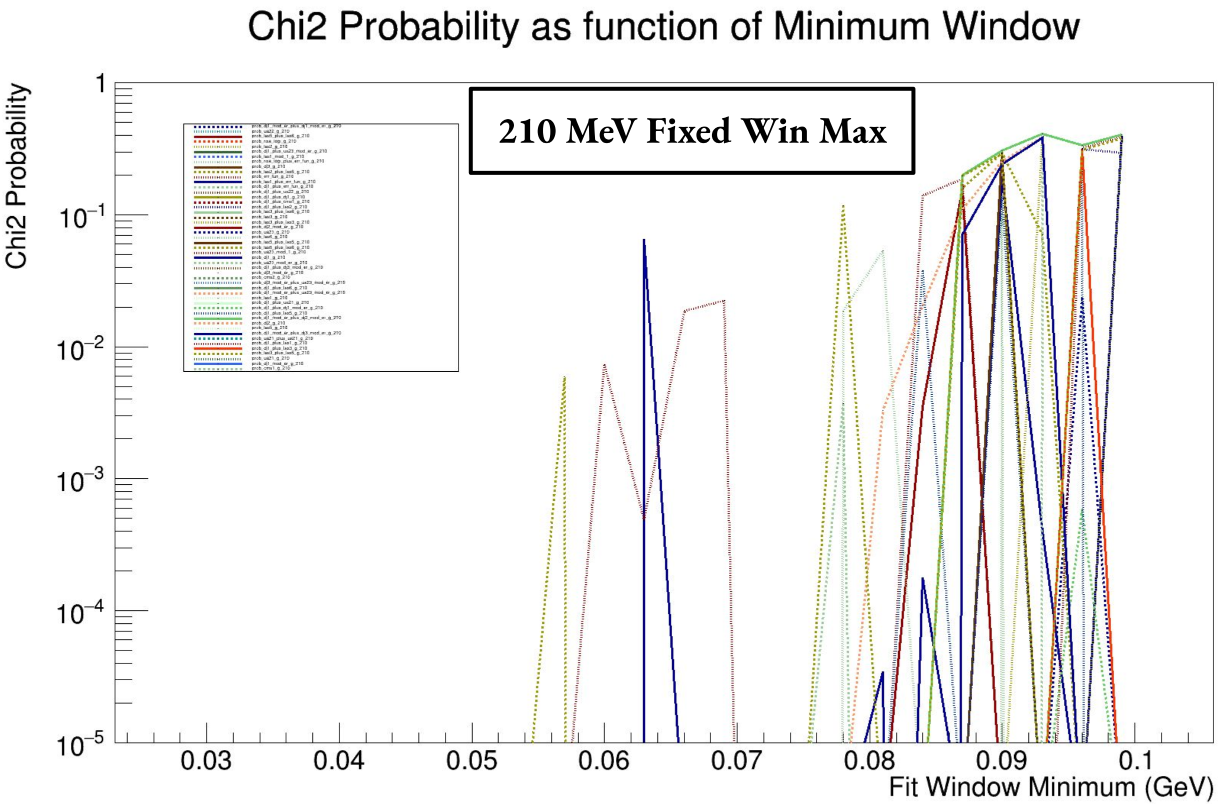
Recent Updates (since last update)

- Fixed major bug responsible for fit failures in all previous functions of the past several studies
    - likelihood fit option no longer failing every time
- Began applying “improve fit” root option
    - fits again with determined best fit parameters
- Started New Study (250-500 iterations)
- 64 Functions → 125 Functions
- Created function filter to tidy plots and expedite function selection
- noticed that many higher parameter functions are being cancelled early in SDF
    - increased memory allotted as potential solution, yet to verify
    - many of the higher parameter fits in following study are at low stats (<50 iterations)

Unintelligible Progress

From last update:

- 1000 iterations
- 64 functions
- much fit failing
- WinMin: [30 MeV - 99 MeV] 3 MeV Steps
- WinMax: [180 MeV - 210 MeV] 3 MeV Steps



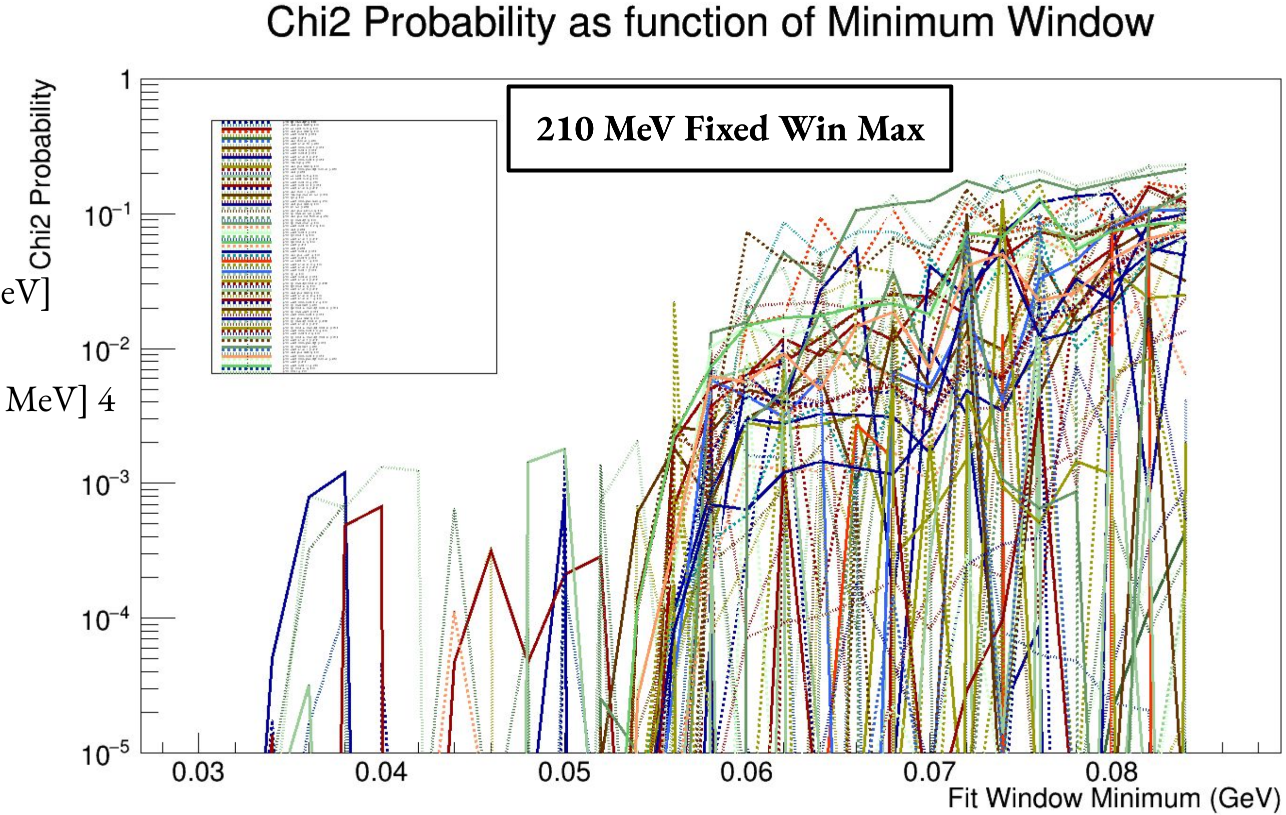
Chi2 Probability as function of Minimum Window

210 MeV Fixed Win Max

Unintelligible Progress

Current Study:

- 25-500 iterations
- 125 Functions
- much less fit failing
- messy
- WinMin: [32 MeV - 86 MeV] 2 MeV Steps
- WinMax: [178 MeV - 210 MeV] 4 MeV Steps

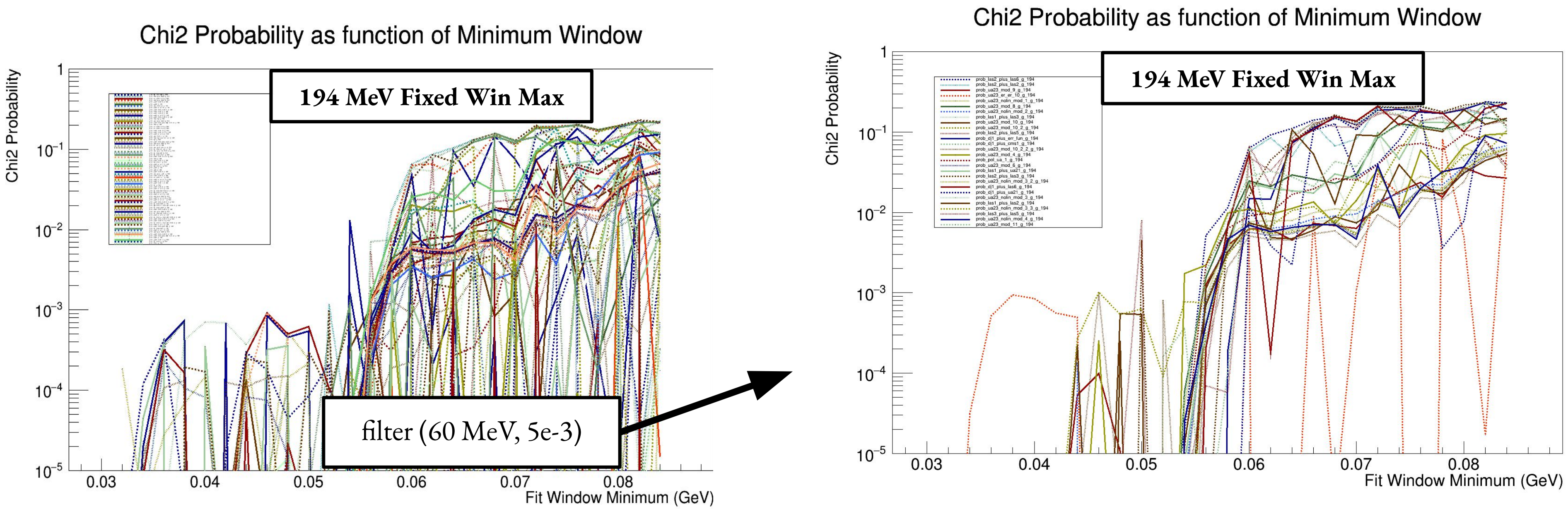


210 MeV Fixed Win Max

Filter Use

Use command line options to filter based on a lower bound p-value threshold for a given window minimum

-F (specified window min) (specified pvalue threshold)



194 MeV Fixed Win Max

filter (60 MeV, 5e-3)

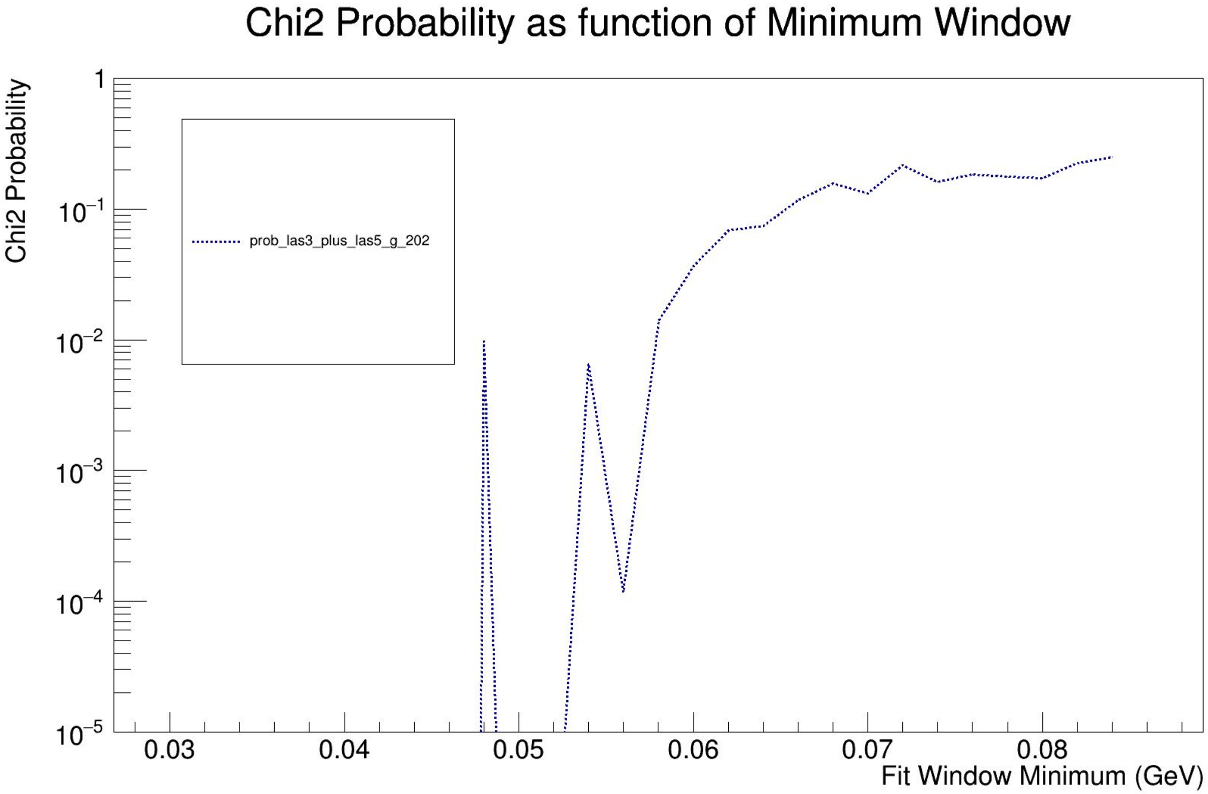
194 MeV Fixed Win Max

Promising low window fits (filter at [48, 5e-3])

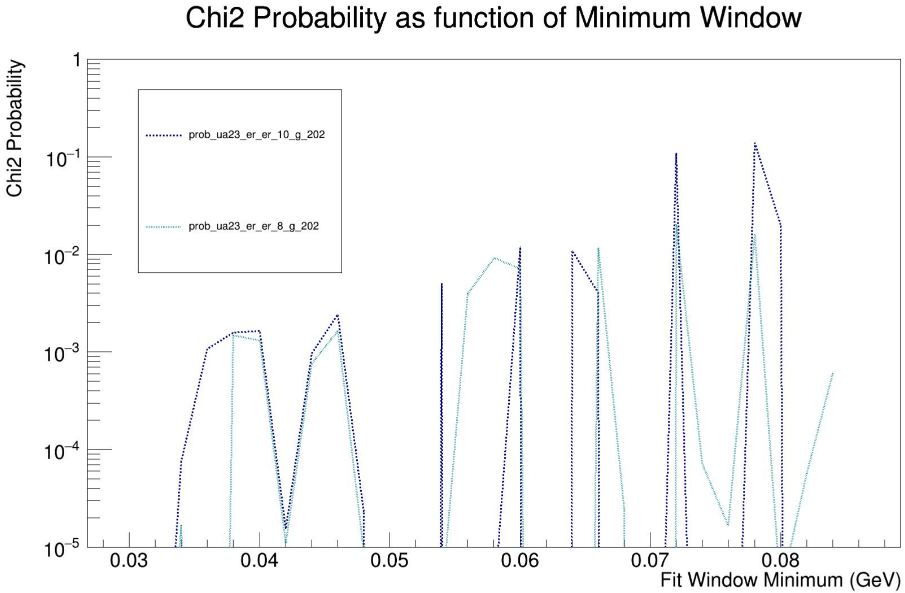
- Filter applied at 48 MeV requires pvalue > 5e-3

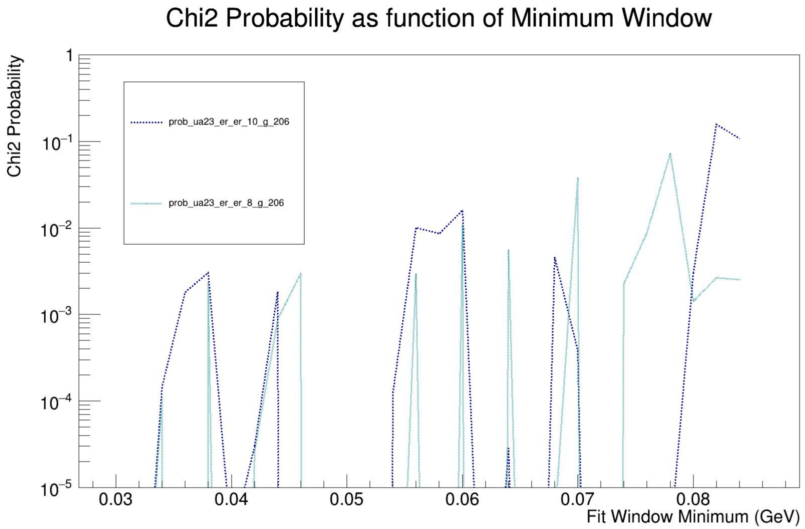
- function fits well (>1e-2) over the range 48 MeV - 202 MeV

- slight good fit oscillation with varied minimum as this function doesnt fit 50, 52 MeV
  - likely to be corrected with higher stats



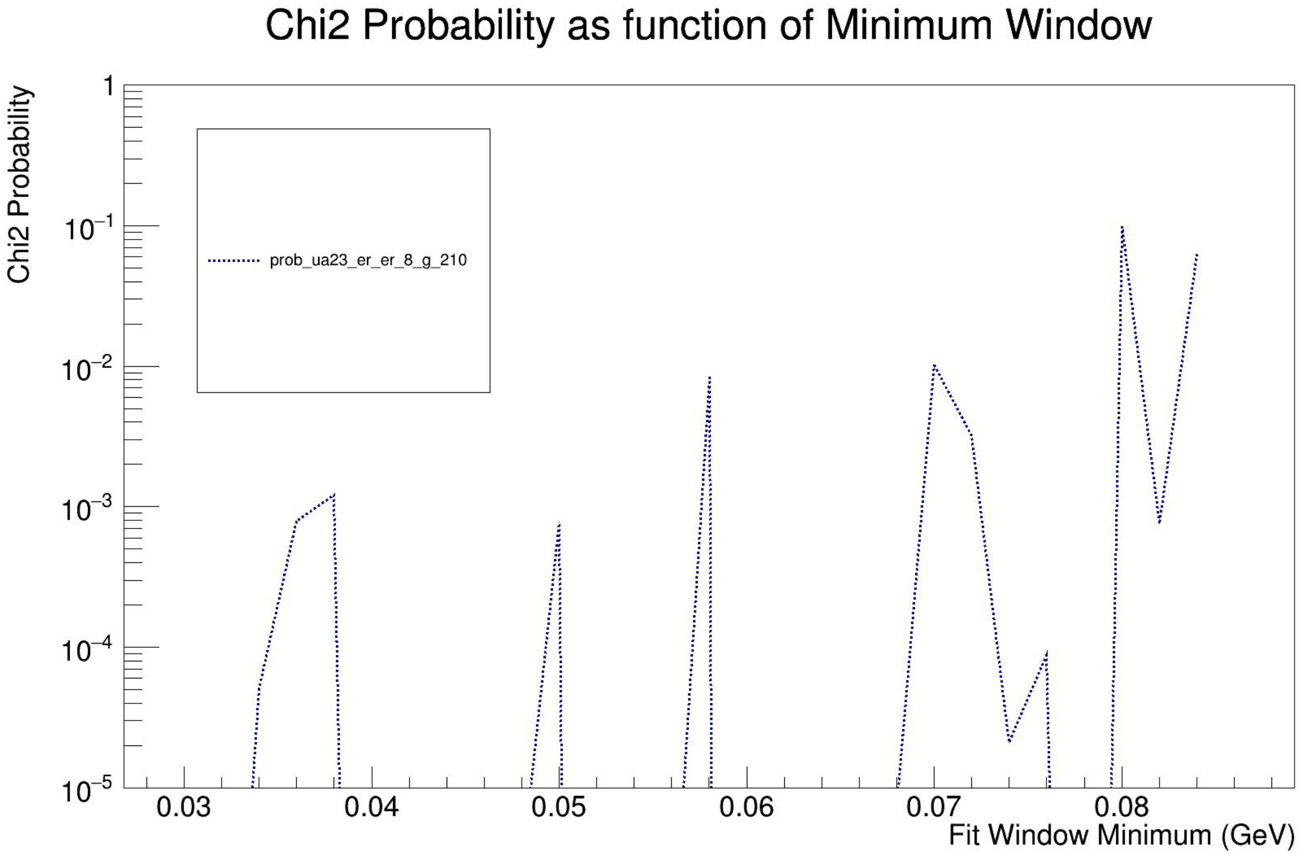
Promising ideal range fits (filter at [38,1e-3])





NOTE: low stats < 50 iterations

Promising ideal range fits (filter at [38,1e-3])



Functions found by multiplying two error functions together for rise, then increasing complexity of exponential function for tail.

Fit info:
pvalue > 1e-3 for
(38-202) MeV
(38-206) MeV
(38 -210) MeV

Best fit so far:
pvalue ~3e-3 at 38-202 MeV

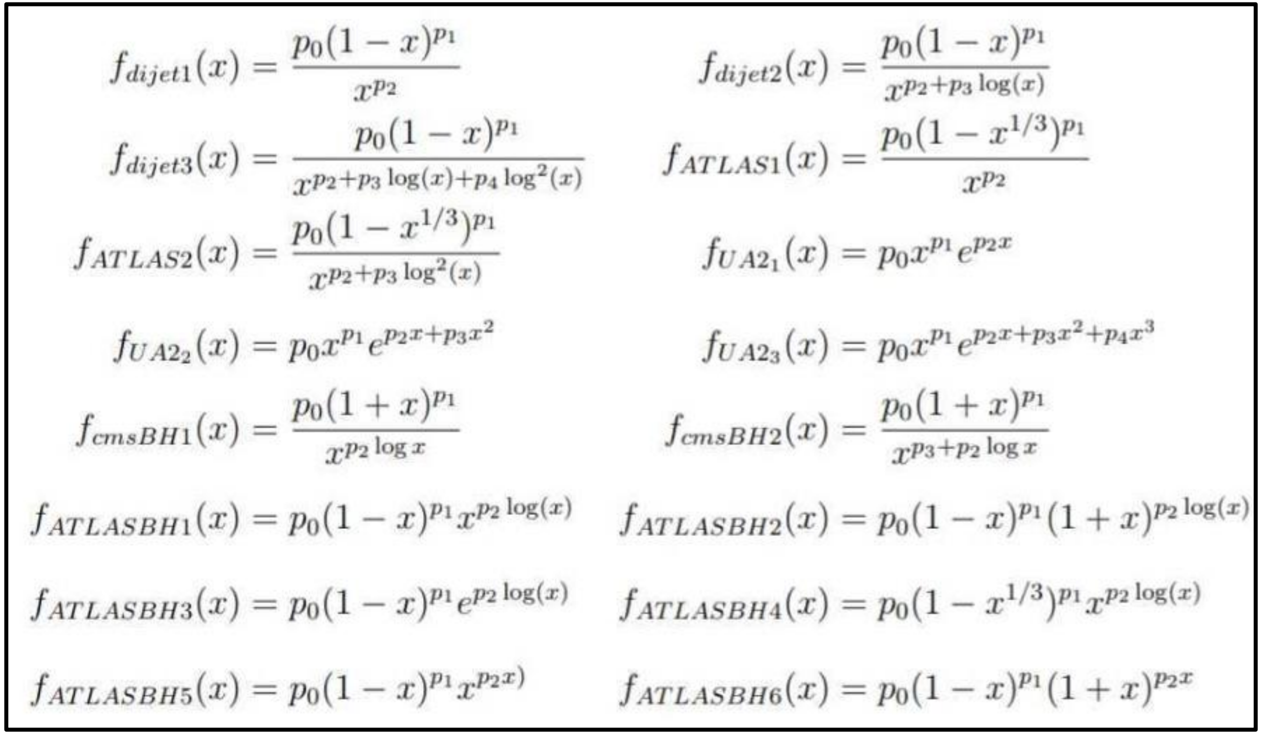
Goals moving forward

- start a new higher stats study to make sure all higher parameter functions complete
- continue to search for function capable of good fit at <40 MeV
- begin piecing together resonance search infrastructure to determine what is necessary to claim reach and next steps once an optimal function is found
- reimplement summary plots from fitting toolkit previously developed
    - ideally only produce plots for functions that pass a specified filter

addl slides (from last update)

Global Fit to the Invariant Mass Distribution

Initial functions →



$$f_{dijet1}(x) = \frac{p_0(1-x)^{p_1}}{x^{p_2}} \qquad f_{dijet2}(x) = \frac{p_0(1-x)^{p_1}}{x^{p_2+p_3\log(x)}}$$

$$f_{dijet3}(x) = \frac{p_0(1-x)^{p_1}}{x^{p_2+p_3\log(x)+p_4\log^2(x)}} \qquad f_{ATLAS1}(x) = \frac{p_0(1-x^{1/3})^{p_1}}{x^{p_2}}$$

$$f_{ATLAS2}(x) = \frac{p_0(1-x^{1/3})^{p_1}}{x^{p_2+p_3\log^2(x)}} \qquad f_{UA2_1}(x) = p_0 x^{p_1} e^{p_2 x}$$

$$f_{UA2_2}(x) = p_0 x^{p_1} e^{p_2 x + p_3 x^2} \qquad f_{UA2_3}(x) = p_0 x^{p_1} e^{p_2 x + p_3 x^2 + p_4 x^3}$$

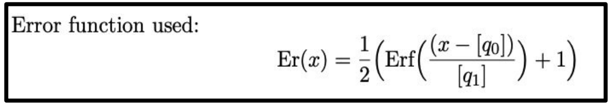
$$f_{cmsBH1}(x) = \frac{p_0(1+x)^{p_1}}{x^{p_2\log x}} \qquad f_{cmsBH2}(x) = \frac{p_0(1+x)^{p_1}}{x^{p_3+p_2\log x}}$$

$$f_{ATLASBH1}(x) = p_0(1-x)^{p_1} x^{p_2\log(x)} \qquad f_{ATLASBH2}(x) = p_0(1-x)^{p_1}(1+x)^{p_2\log(x)}$$

$$f_{ATLASBH3}(x) = p_0(1-x)^{p_1} e^{p_2\log(x)} \qquad f_{ATLASBH4}(x) = p_0(1-x^{1/3})^{p_1} x^{p_2\log(x)}$$

$$f_{ATLASBH5}(x) = p_0(1-x)^{p_1} x^{p_2 x)} \qquad f_{ATLASBH6}(x) = p_0(1-x)^{p_1}(1+x)^{p_2 x}$$

We multiply each function by an error function to fit the rise:

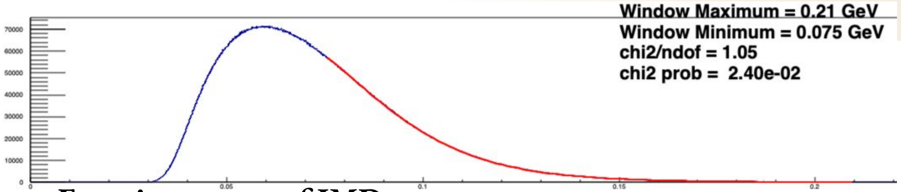


Error function used:

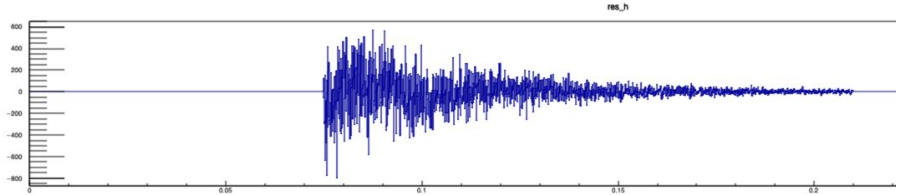
$$\mathrm{Er}(x) = \frac{1}{2}\left(\mathrm{Erf}\left(\frac{(x-[q_0])}{[q_1]}\right)+1\right)$$

C. Bravo. *Thesis linked here*

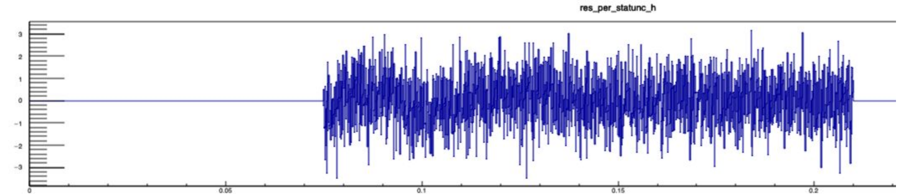
Representative “Good” Fit Using Global Fitting Tool



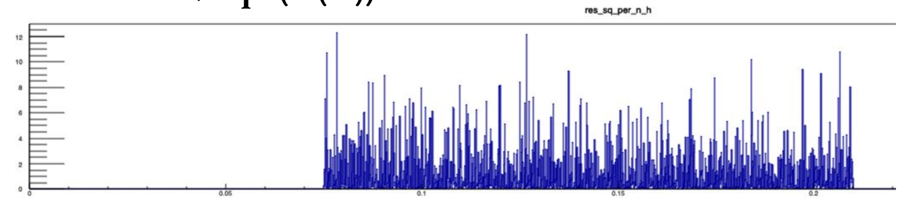
Function on top of IMD



Residual Plot

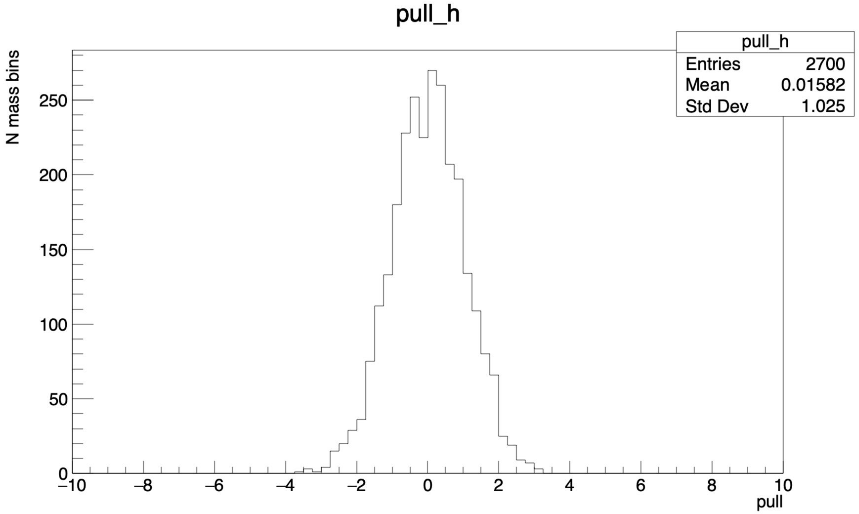


Residual / sqrt(N(m))



Residual^2 / (N(m))

- UA23 Function
- Fit Range: 75 MeV - 210 MeV
- Good ⇒ pvalue >10^{-2}



What’s new? (1/2)

- removed sum function generator from global fitting script
  - now able to create input parameter and function files for the sum of two independent functions before running fitting (allows massive scale up of total functions testable)
- store best fit parameters for each window in txt file (great for viewing parameters)
  - [win_min win_max best_param1 best_param2 best_param3 ... chi2/ndf pvalue]
- changed fitting logic to extend beyond local minimums
  - for each iteration, width of generated gaussian increases by .01*(iteration number)*(initial mean)
- Modified terminal input to utilize additional parameter txt file for every function
- integrated workflow into SSH to generate fitting script for each function to run remotely

Sum Function Generator

Terminal Input:

- python3 sum_fun_gen.py -i ./functions/[function1.txt] -f ./functions/[function2.txt] -d ./functions/ -e ./parameters/

Expected Output:

- generates function1_plus_function2.txt file in /resonance_fitting/functions/
  - with m (= f1+ f2) many parameters of the form [0],[1],..,[m-1]
- generates function1_plus_function2.txt file in /resonance_fitting/parameters/
  - created using starting parameters of summands of the form [p1 p2 ... pm]

Making global fitting scripts for every function

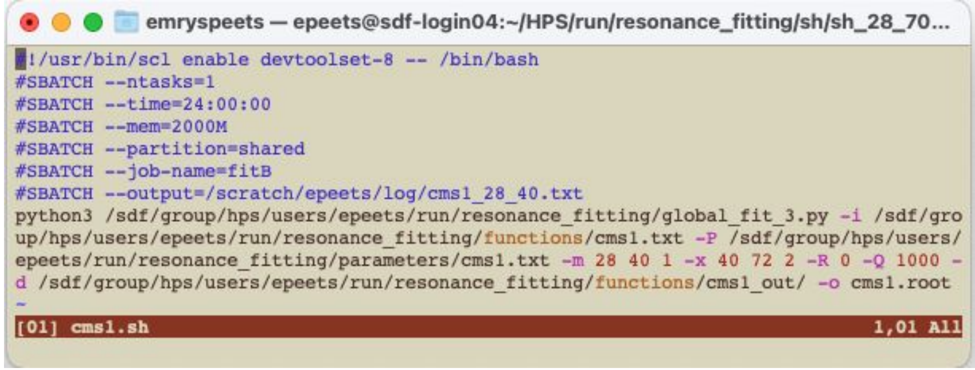
Terminal Input

```
python3 /sdf/group/hps/users/epeets/run/resonance_fitting/makeGlobalFitScripts.py -d
/sdf/group/hps/users/epeets/run/resonance_fitting/sh/ -m 28 40 1 -x 40 72  2 -F
/sdf/group/hps/users/epeets/run/resonance_fitting/functions/
```

(WinMin,WinMax)

Expected Output

- resonance_fitting/sh/subJob_28_70.sh (to sbatch each function)
- resonance_fitting/sh/sh_28_70/[function.sh]


emryspeets — epeets@sdf-login04:~/HPS/run/resonance_fitting/sh/sh_28_70...

```
#!/usr/bin/scl enable devtoolset-8 -- /bin/bash
#SBATCH --ntasks=1
#SBATCH --time=24:00:00
#SBATCH --mem=2000M
#SBATCH --partition=shared
#SBATCH --job-name=fitB
#SBATCH --output=/scratch/epeets/log/cms1_28_40.txt
python3 /sdf/group/hps/users/epeets/run/resonance_fitting/global_fit_3.py -i /sdf/gro
up/hps/users/epeets/run/resonance_fitting/functions/cms1.txt -P /sdf/group/hps/users/
epeets/run/resonance_fitting/parameters/cms1.txt -m 28 40 1 -x 40 72 2 -R 0 -Q 1000 -
d /sdf/group/hps/users/epeets/run/resonance_fitting/functions/cms1_out/ -o cms1.root
~
[01] cms1.sh                                                                 1,01 All
```

Automated fitting terminal input

What’s new? (2/2)

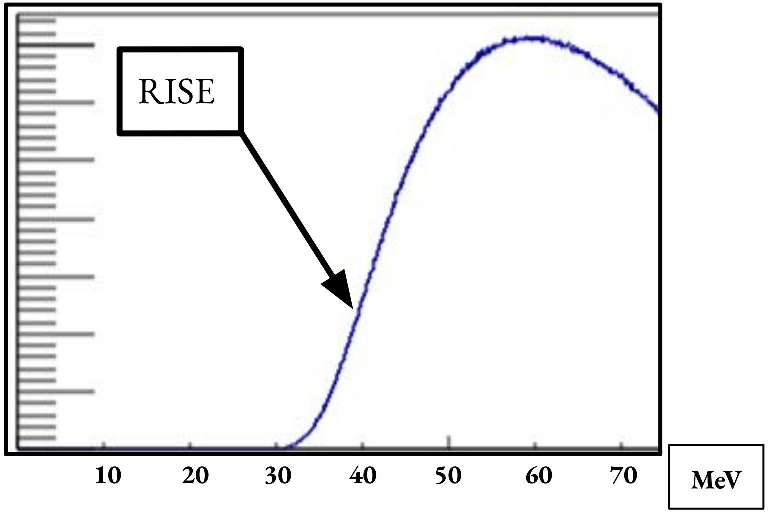
- Discovered bug that caused the failure of all >10 parameter fits (*thanks Cam*)
  - offers solid strategy towards finding *the one true function*
- cleaned code to run more efficiently
- Started process of performing likelihood fits in addition to chi2 fits
- Generously scaled up total functions being used in tests
  - new class of functions without error function
  - mixing and matching functions
  - frankenstein functions
- conducted preliminary study making use of full fitting infrastructure
- began higher statistics study for global range and only rise range

Fitting the Rise of Background Distribution (28-70 MeV)

Necessary contingency if single function unable to fit global distribution.

Purpose of rise study is to determine the component of a piecewise function dedicated to fitting only the rise of distribution.

Finer granularity (step size) useful for rapid rise of data collected.



Tentatively Promising Functions (subject to change)

From the rise only study:

dj1_mod_er_plus_dj1_mod_er (no error function!)

[0.036, 0.062, 2.7275, -12.0245, 3.641, -386976.40, 65.5842, 1.1578, 0.007565]

[0.037, 0.062, 3.511, -8.7734, 3.7145, -20874.21, 47.8108, 1.15455, 0.0097726]

Note similar parameters for each window range

[0.036, 0.07, 426.865, 12.5755, 2.5965, -113100.192, 49.262, 1.19239, 0.000410]

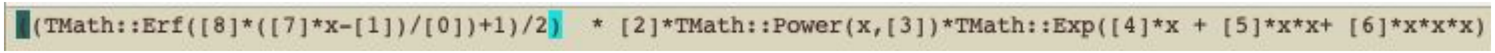
[0.037, 0.07, 435.970, 12.2184, 2.5459, -212454.257, 53.0277, 1.18481, 0.000751]

[0.038, 0.07, 1494.30692, 17.2085, 2.2636, -263171.685, 51.5685, 1.140395, 0.00797777]

[0.039, 0.07, 3175.1608, 19.230, 1.9868, -1098238.979, 58.6093, 1.12298202, 0.0182566]

From Global Fit Study (incomplete study at the moment):

UA23_mod_1 (as illustrated in previous plots)



```
((TMath::Erf([8]*([7]*x-[1])/[0])+1)/2)  * [2]*TMath::Power(x,[3])*TMath::Exp([4]*x + [5]*x*x+ [6]*x*x*x)
```

mod

Depending on window range: this function consistently produces "good fits" from (57 MeV - 210 MeV)