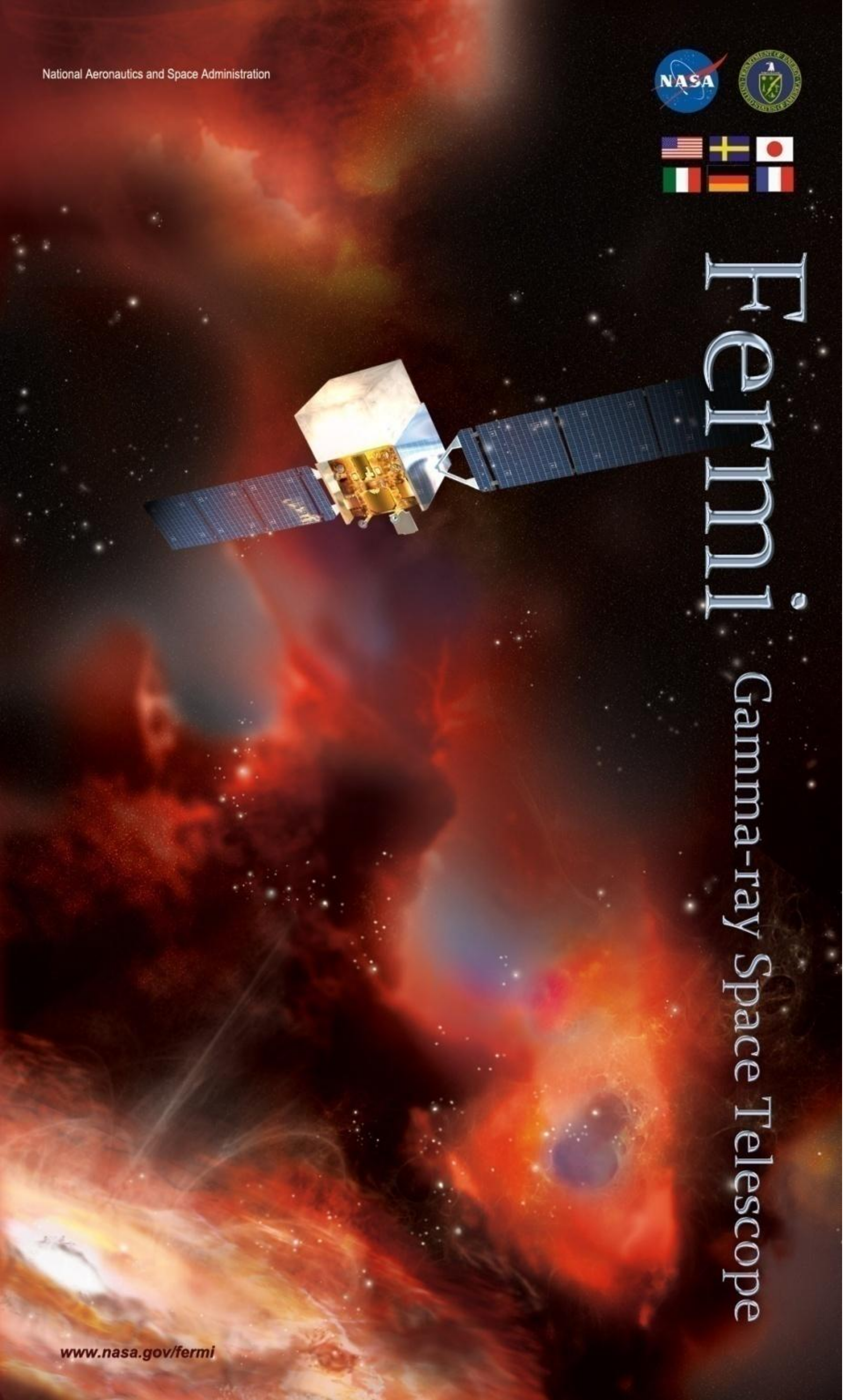


Line Analysis Tutorial

Andrea Albert
(SLAC)

Fermi Summer School
May 29, 2015

Code Overview

- ROOT-based code
  - ROOT is an analysis software used predominantly by particle physicists
  - we are going to use python version pyROOT as well as standard interface

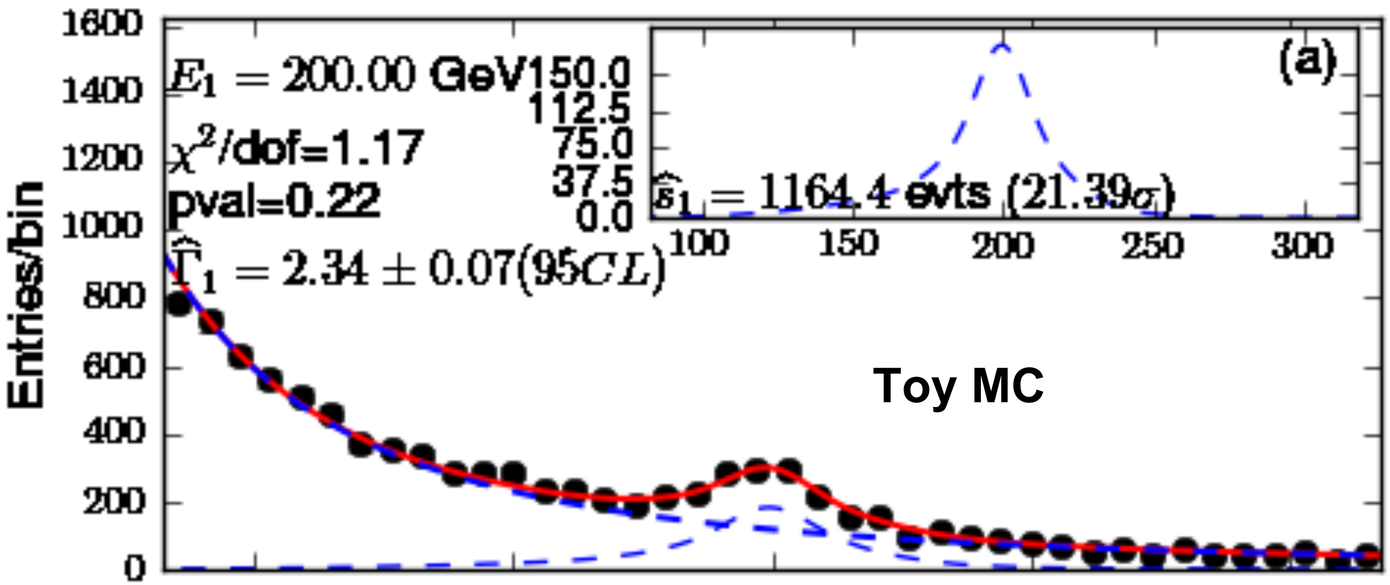
- Uses “RooFit” package
  - for likelihood analyses
  - easy to create model PDFs and package does lots behind the scenes for you (like making sure the normalizations are correct)
  - let’s you do complicated fits (fancy models) and modular (easy to add new pieces)
    - for simple fits don’t use RooFit...use e.g. SciPy

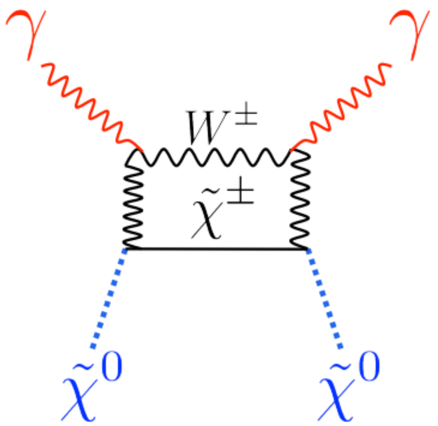
Investigate Merit Data

- **Merit data (ROOT files) are ROOT Trees**
  - **you can find some tutorials on ROOT website**
    - **https://root.cern.ch/drupal/content/howtos#trees**
    - **https://root.cern.ch/root/htmldoc/tutorials/tree/index.html**

- **root -l Vela_22d_100z_base_p7r_clean-select.root**
- **MeritTuple.GetEntries()**
  - **total number of events in your dataset**
- **MeritTuple.Show(0)**
  - **each event in dataset is a different entry…each entry has values defined as “branches”**
- **MeritTuple.GetEntries(“CTBBestEnergy>1000”)**
  - **find number of events above 1 GeV**
- **MeritTuple.Draw(“log10(CTBBestEnergy)”)**
- **MeritTuple.Draw(“FT1B:FT1L”,””,”COLZ”)**

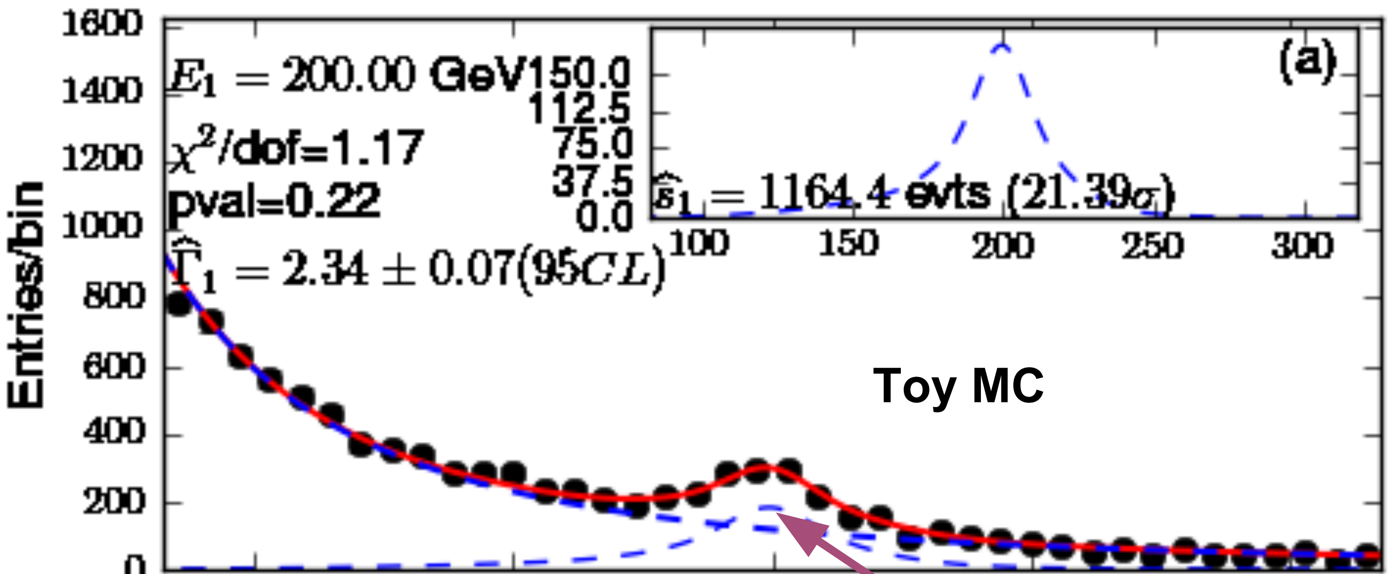
Search for Gamma-ray Spectral Lines

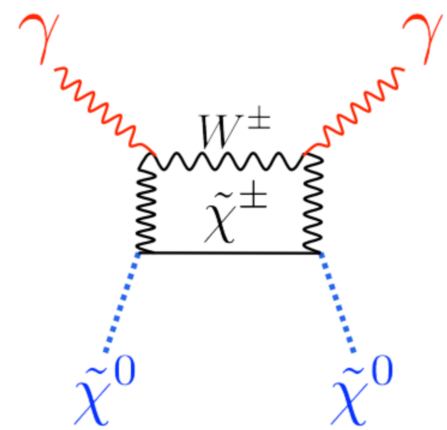




- Annihilation/decay directly into Xγ (X = γ, Z^0, H^0)
  - Location of line in energy spectrum related to m_χ
- Advantage: sharp, distinct feature
  - Looking for a "bump" in the energy spectrum
- Disadvantage: low predicted counts

Search for Gamma-ray Spectral Lines

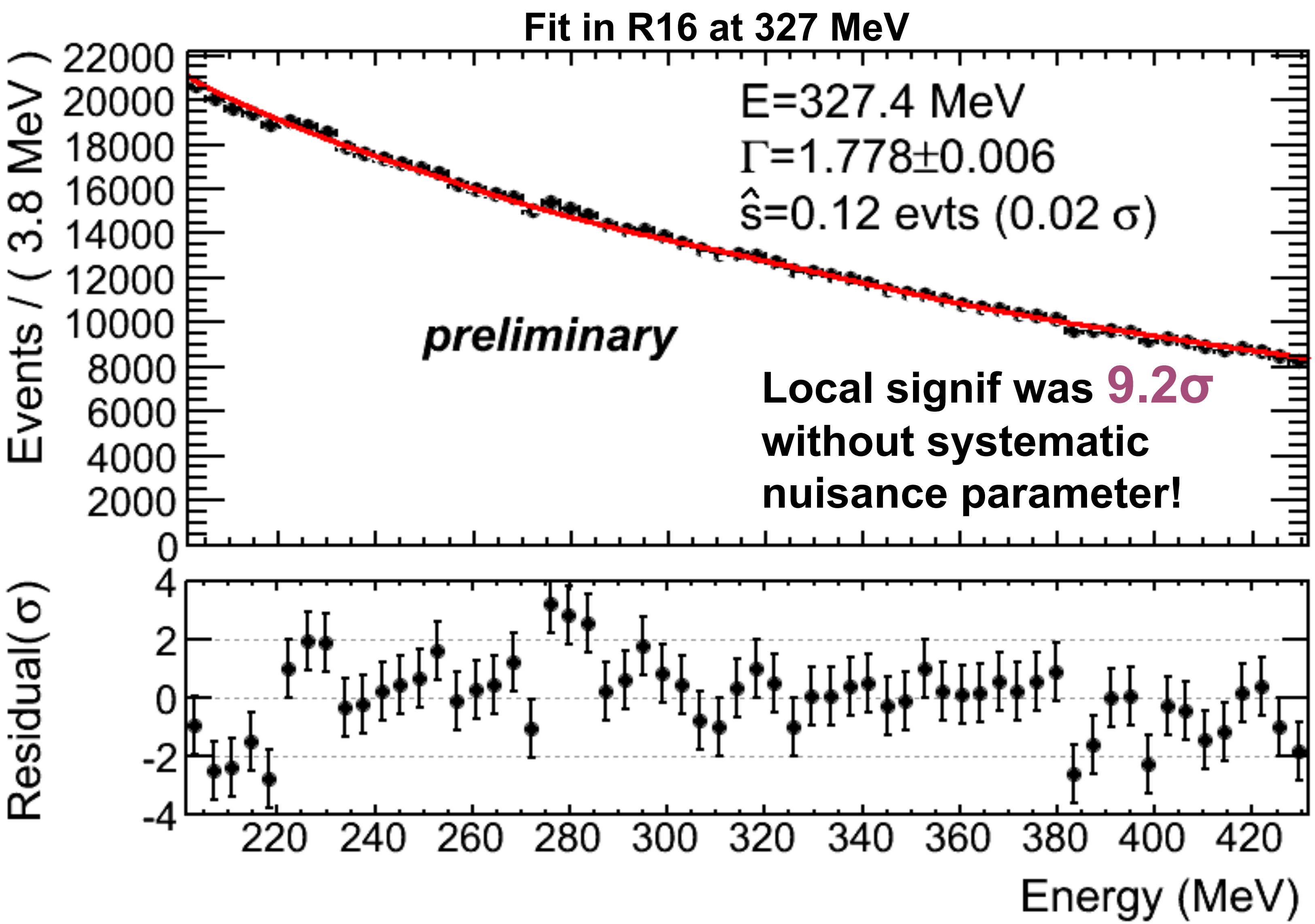




This line is very obvious…but we don't have a bright gamma-ray line in the data so we need to be careful

- Annihilat
  - Locat
- Advantag
  - Looking for a "bump" in the energy spectrum
- Disadvantage: low predicted counts

Cautionary Tale -- Low Energy Line Search Example Fit – Including n_{sys} is essential!



Effective Background

F_{sig} and F_{bkg} are the signal and bkg PDFs

$$I_{00} = \sum_i \frac{F_{sig}^2(\vec{x}_i)}{F_{bkg}(\vec{\alpha} \mid \vec{x}_i)}$$

I_{00} can be interpreted as how much you can distinguish between the signal and bkg PDFs.

$$b_{eff} \sim \frac{1}{I_{00} - 1}$$

Larger I_{00} -> smaller beff

$$f = \frac{n_{sig}}{b_{eff}}$$

$$signif \approx \frac{n_{sig}}{\sqrt{b_{eff}}}$$

P8 Line Search
Accounting for f_{sys} in Likelihood

- **Include nuisance parameter (n_{sys}) for systematically-induced line-like features**
  - **Only detect a significant line if larger than the line-like features we see in the control regions**
  - **Introduced method in low-energy line paper (A. Albert et al. JCAP10(2014)023)**
  - **Similar technique used to incorporate J-factor uncertainties dSph analysis**
    - **Can be applied whenever accounting for systematic uncertainties is important**

$$C(E,\alpha) = \left((n_{sig} + n_{sys})S(E,E_{\gamma}) + n_{bkg}B(E,\Gamma_{bkg})\right) * G_{sys}$$

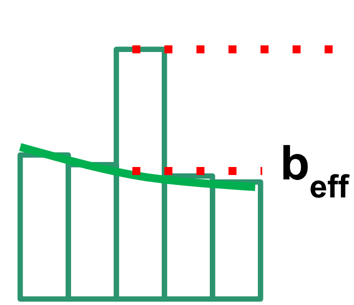
$$\sigma_{sys} = \delta f_{sys} * b_{eff}$$

n_{sys} is constrained using δf_{sys} estimated with control regions

$$G_{sys} = \frac{1}{\sigma_{sys}\sqrt{2\pi}} e^{-n_{sys}^2/2\sigma_{sys}^2}$$

Gaussian constraint on n_{sys}

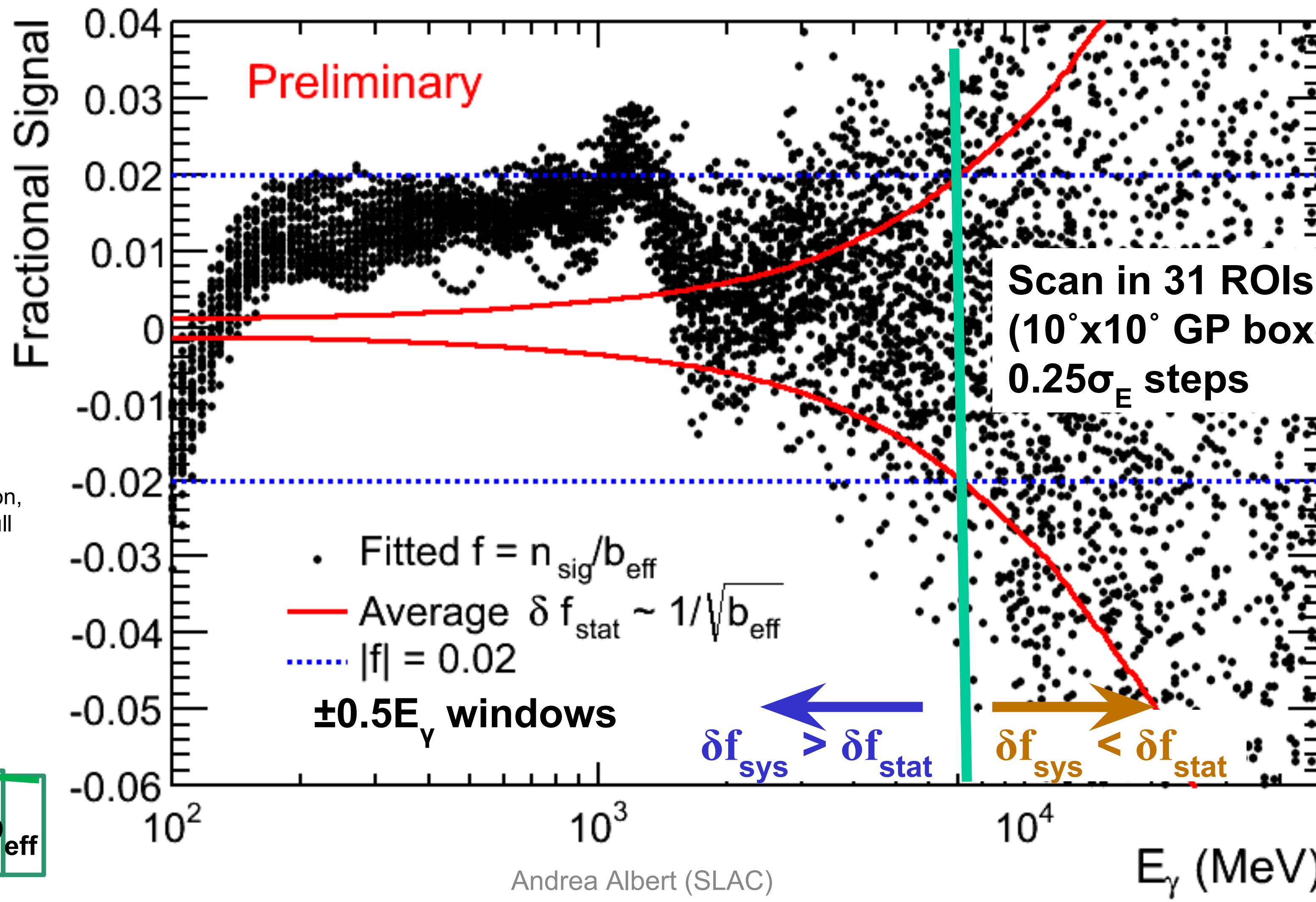
$$f = \frac{n_{sig}}{b_{eff}} \approx \frac{TS}{n_{sig}}$$



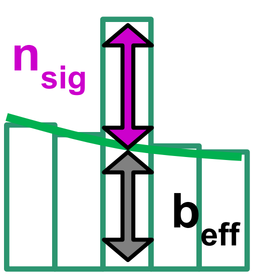
Warning: cartoon, see paper for full b_{eff} definition
A. Albert et al. JCAP10(2014)023

Pass 8 Line Search
f_{sys} from Galactic Plane Scans

- There are some common features likely from the effective area (A_{eff})
- Displacement from 0 is mostly from A_{eff}, while spread is from bkg. modeling
- Larger systematic effect with wider windows (since power-law approx. gets worse)



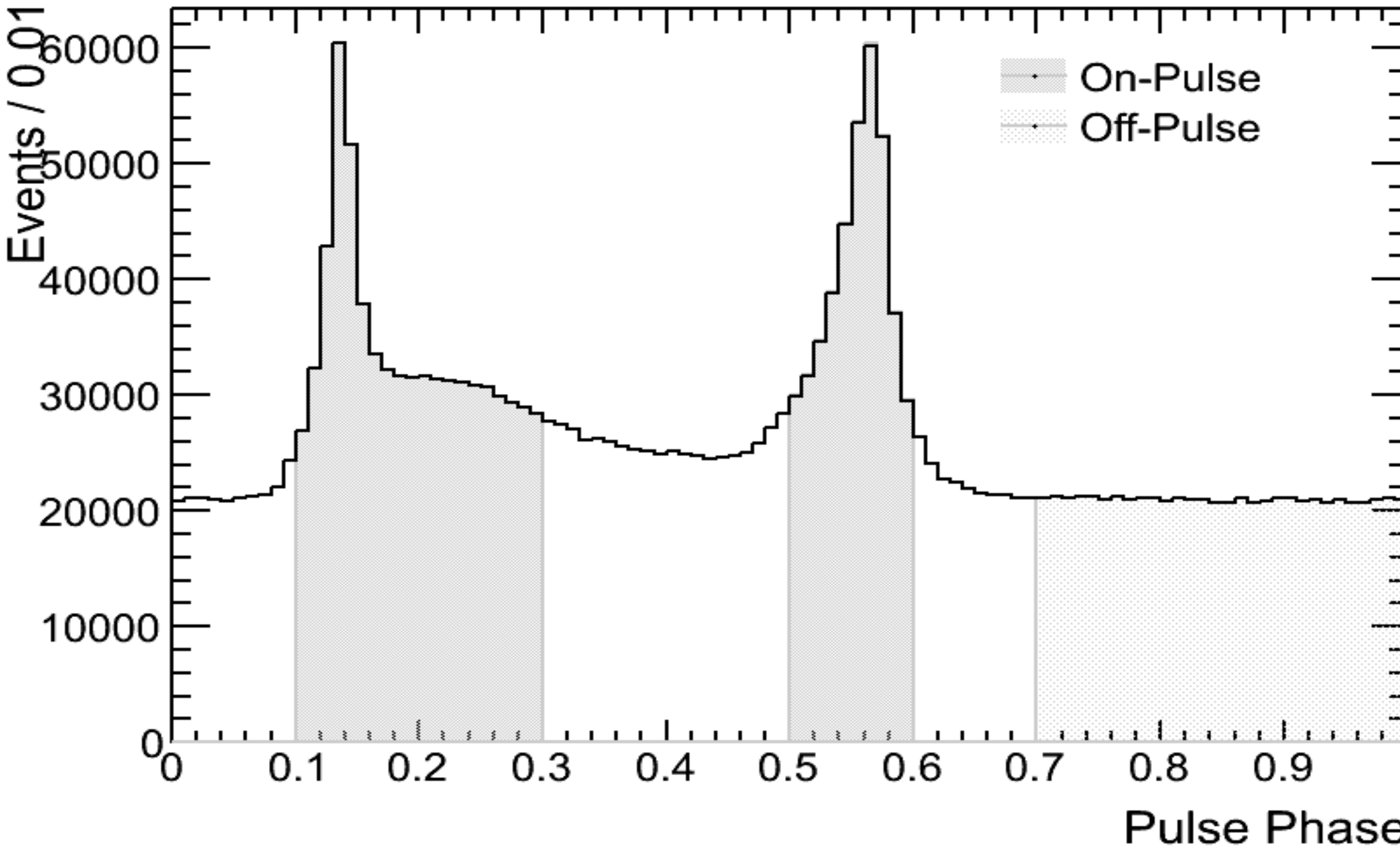
Warning: cartoon, see paper for full b_{eff} definition
A. Albert et al. JCAP10(2014) 023



Vela as a Control Sample

- Can use pulse phase to select pure gamma-ray sample
- Vela is a common control sample for LAT systematic investigations
- Don't expect spectral lines from Vela
  - we see how big the "line" in Vela's spectrum are
- Our dataset:
  - ROI: 22° radius circle centered on Vela
  - Energy Range: 100 MeV – 100 GeV
  - Standard data quality cuts
  - $\theta_r < 52°$
  - $\theta_z < 100°$ for ROI applied w/ *gtmktime*

Definition of Vela Phase Ranges



On-Pulse: [0.1-0.3, 0.5-0.6]
Off-Pulse: [0.7-1.0]

Line Code

- **Top level is FitVela.py**
- **python FitVela.py -h**
- **python -i FitVela.py -i Vela_22d_100z_base_p7r_clean-select.root -o "test" -f 500. -m 200. -M 800. -S Models/Line_P7REP_CLEAN_V15_1D_Vela.yaml -B Models/Vela.yaml --negative -s 0.02**

Line Code -- Fit Results

- root -l test_FitResults.root
- FIT_RESULTS.Show(0)
  - What is the significance?
- FIT_RESULTS_SYST.Show(0)
  - What is the fractional size of this line?
  - What is the significance?

Line Code

- Try fitting at 250 MeV, 527 MeV, 1 GeV, 5 GeV, and 9.7 GeV
  - instead of using '-f -m -M' options, call range file
    - -r FitRanges_P7REP_CLEAN_V15_comb_2sig.txt
- Try plotting fractional size vs. Fit Energy
  - FIT_RESULTS.Draw("(Signal_Counts)/(Bkg_Effective):Fit_Energy")
- Try plotting the statistical uncertainty (sqrt(Beff)) vs Fit Energy