

高エネルギー宇宙物理学 のための ROOT 入門

– 第 5 回 –

奥村 瞳

名古屋大学 宇宙地球環境研究所

2016 年 6 月 8 日

git pull して最新版にしてください

```
$ cd RHEA
$ git pull

$ zsh
$ for i in {009..049}; do curl -O https://raw.githubusercontent.com/akira-
okumura/RHEA-Slides/master/photons/lat_photon_weekly_w${i}
_p302_v001_extracted.root; done

$ bash
$ for i in {10..49}; do curl -O https://raw.githubusercontent.com/akira-okumura/
RHEA-Slides/master/photons/lat_photon_weekly_w0${i}_p302_v001_extracted.root;
done
```

TTree

TTree とは

- TH1D や TGraph と違い同じ概念を持つものが他のソフトウェアには（多分）存在しない
- Event の概念を持つ実験データには欠かせない
- 非常に大雑把に説明すると表計算ソフト (Excel など) のシートや FITS の table のようなもの

Event No.	Energy(MeV)	Gal. Longitude (°)	Gal. Latitude (°)	Time (ns)
0	44.5018	123.252	11.7771	2.40E+08
1				
2				
3				

Fermi/LAT のガンマ線イベントデータの例

- ただし、演算機能、データの可視化、ROOT クラスの保存など、TTree でしか実現できない機能が多くある
- どうして ROOT を使うのか、という問い合わせへのひとつ答える

まずは遊んでみる (Fermi/LAT のデータ例)

```
$ root misc/lat_photon/lat_photon_weekly_w009_p302_v001.root
root [0]                                         ① TTree の含まれる ROOT ファイルを引数にする
Attaching file misc/lat_photon/lat_photon_weekly_w009_p302_v001.root as _file0...
(TFile *) 0x7fc0a2f05aa0
root [1] .ls
TFile**      misc/lat_photon/lat_photon_weekly_w009_p302_v001.root
TFile*       misc/lat_photon/lat_photon_weekly_w009_p302_v001.root
  KEY: TTree   photons;1 LAT PASS8 Photons ③ “photons” という名前の TTree がいる
root [2] photons->Print()
*****
*Tree    :photons  : LAT PASS8 Photons
*Entries : 177778 : Total =      27471504 bytes  File  Size =  27453414 *
*          :           Tree compression factor =  1.00
*****
*Br     0 :ENERGY  : ENERGY[1]/F
*Entries : 177778 : Total  Size=      713624 bytes  File Size =  712860 *
*Baskets : 23 : Basket Size=      32000 bytes  Compression=  1.00
*.....
*Br     1 :RA      : RA[1]/F
*Entries : 177778 : Total  Size=      713516 bytes  File Size =  712768 *
*Baskets : 23 : Basket Size=      32000 bytes  Compression=  1.00
*.....
```

- 元ファイルは FITS 形式で、わざわざ ROOT に変換して解析する必要はないファイルですが、演習目的です
- 実際のデータで遊べるチュートリアルはそこらへんに落ちてない

続き

```
root [3] photons->GetEntries()  
(Long64_t) 177778  
root [4] photons->Show(0)  
=====> EVENT:0  
ENERGY = 44.5018  
RA = 14.0396  
DEC = 74.6459  
L = 123.252  
B = 11.7771  
THETA = 43.9611  
PHI = 166.852  
ZENITH_ANGLE = 70.4655  
EARTH_AZIMUTH_ANGLE = 343.811  
TIME = 2.39557e+08  
EVENT_ID = 52785  
RUN_ID = 239557414  
RECON_VERSION = 0
```

① この TTree には 177,778 イベントが含まれる
② ひとつひとつのイベントを見たいとき

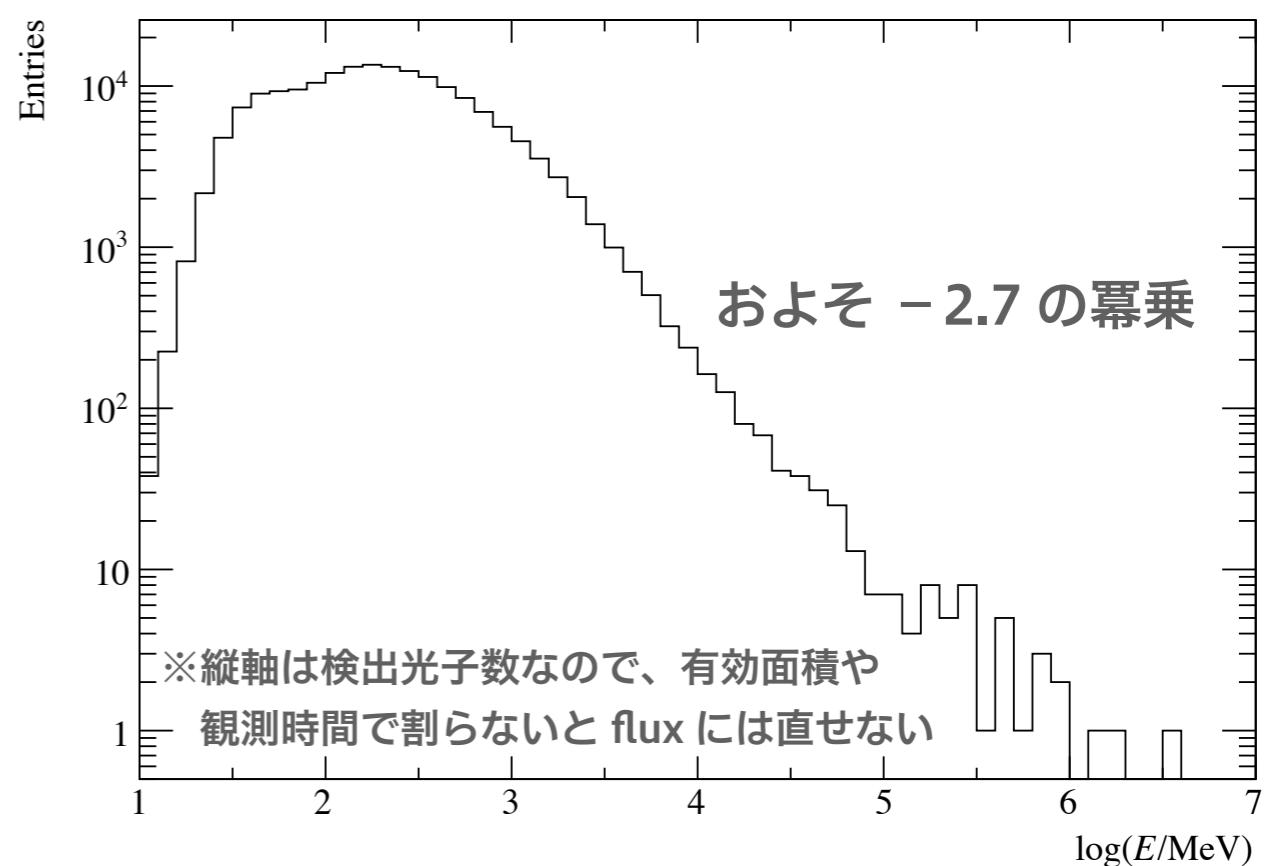
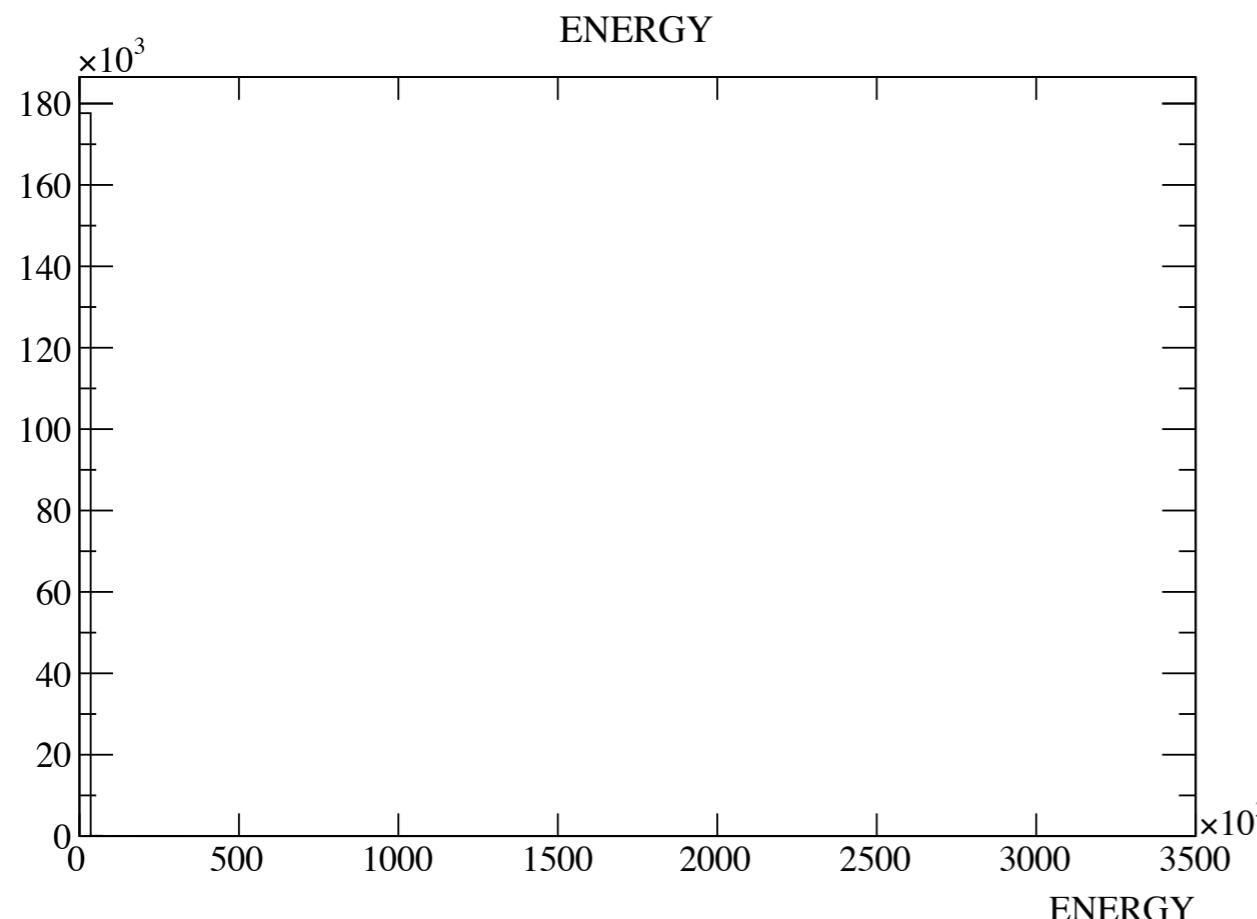
- 「イベント」ごとに、色々な情報が入っている
- 数百から数億イベントになってくると、TTree を使う事のありがたみが分かってくる

続き



① ROOT ファイルを見るためのブラウザが立ち上がる

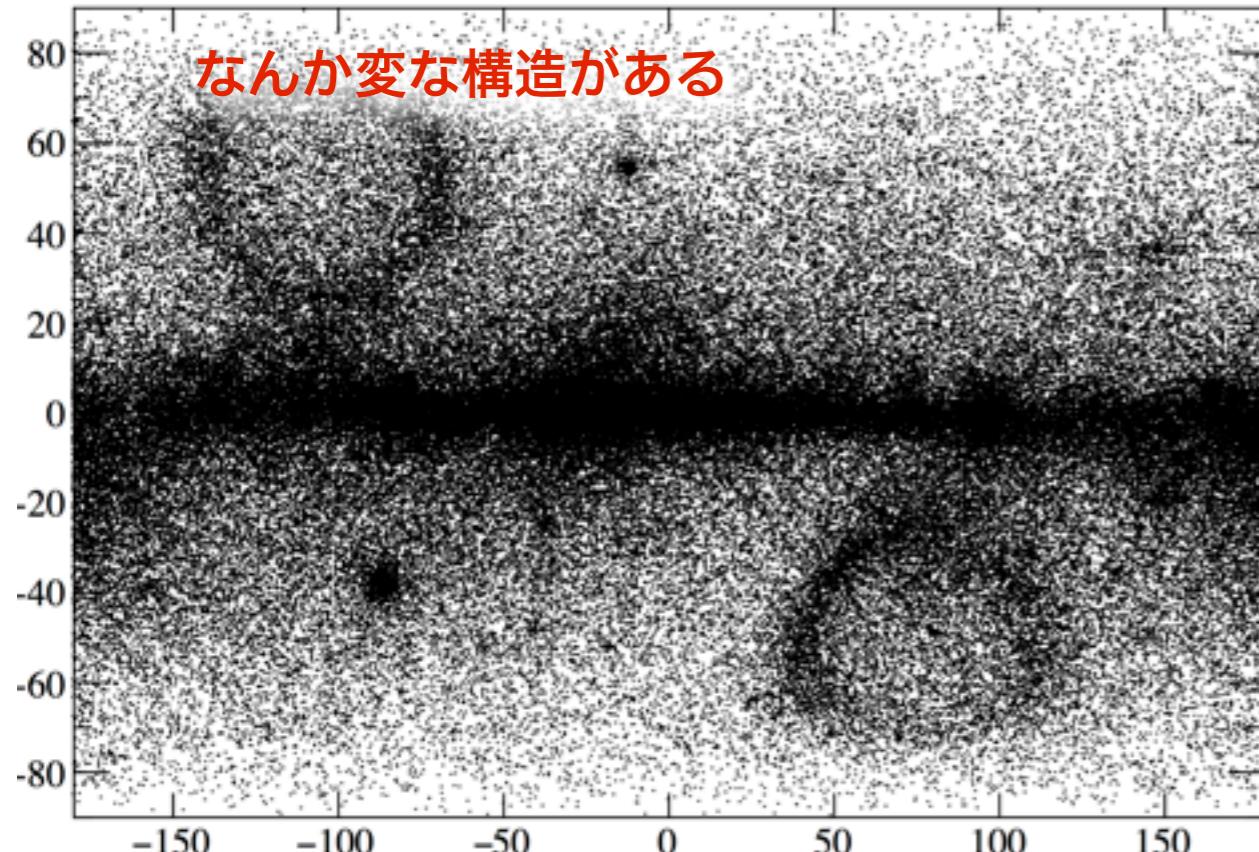
まずはガンマ線のエネルギー分布を見てみる



```
root [3] photons->Draw("ENERGY") ①好きなbranchでヒストグラムを書くことができる  
root [4] TH1D* hEnergy = new TH1D("hEnergy", ";log(#it{E}/MeV);Entries", 60, 1, 7) ③あらかじめ好きなヒストグラムを作つておくと…  
root [5] photons->Draw("log10(ENERGY)>>hEnergy")  
root [6] gPad->SetLogy(1) ④そこにTTree::Drawの結果を詰めることができる
```

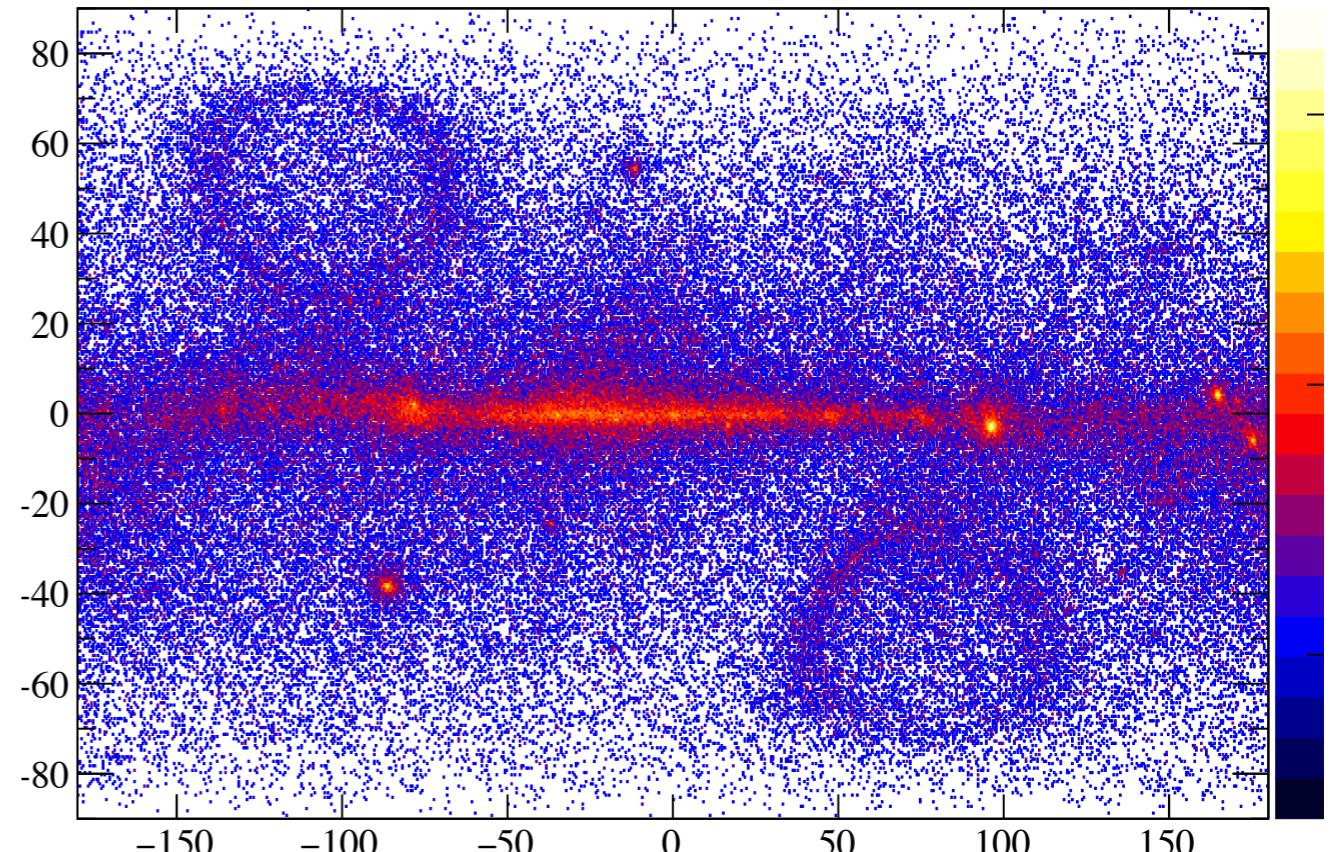
銀河座標でガンマ線の分布を見てみる

B:-(L > 180 ? L - 360 : L)



※ ROOT は TGaxis を使わないと、正から負に向かう軸を書けません

B:-(L > 180 ? L - 360 : L)



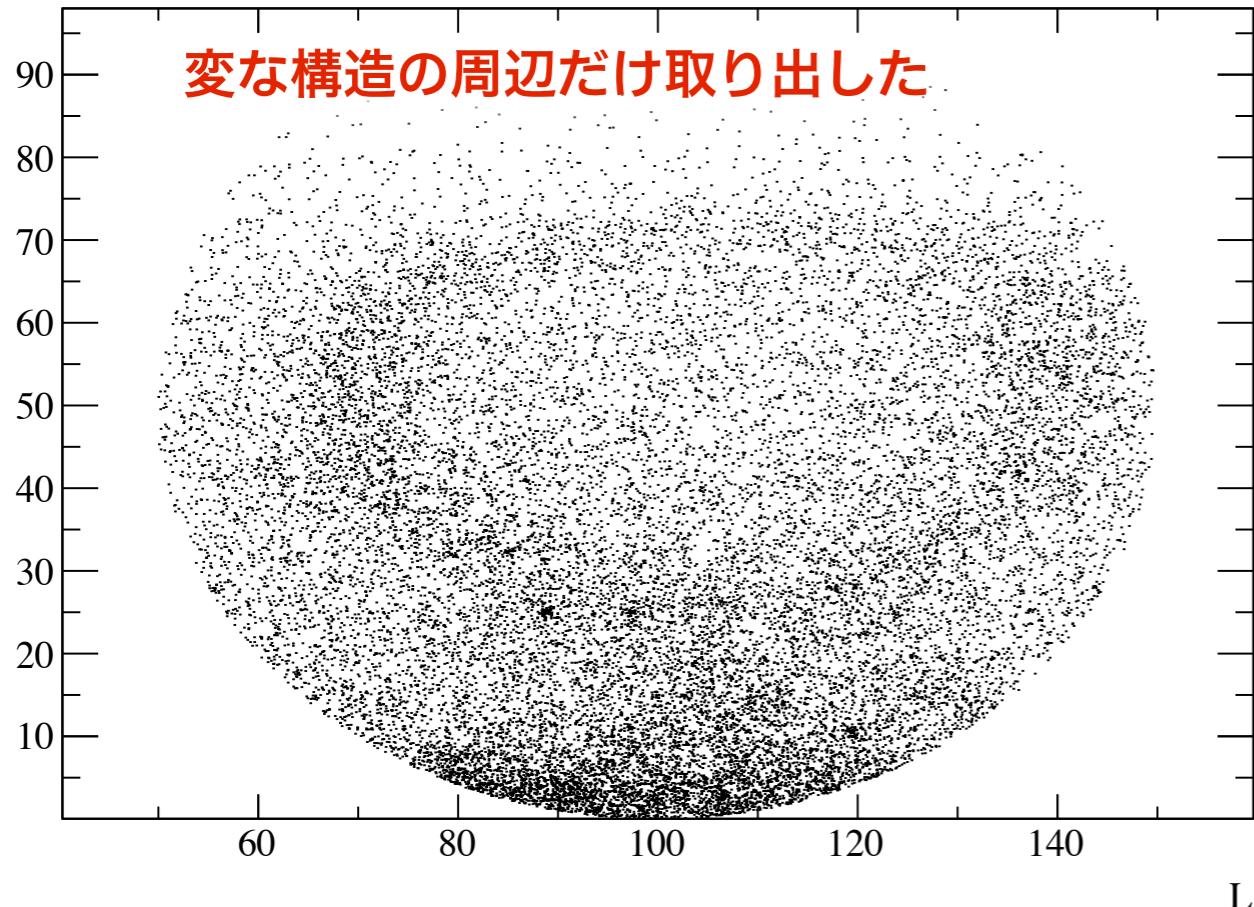
※ ROOT でこのような図を吐くと PDF が非常に重いので注意



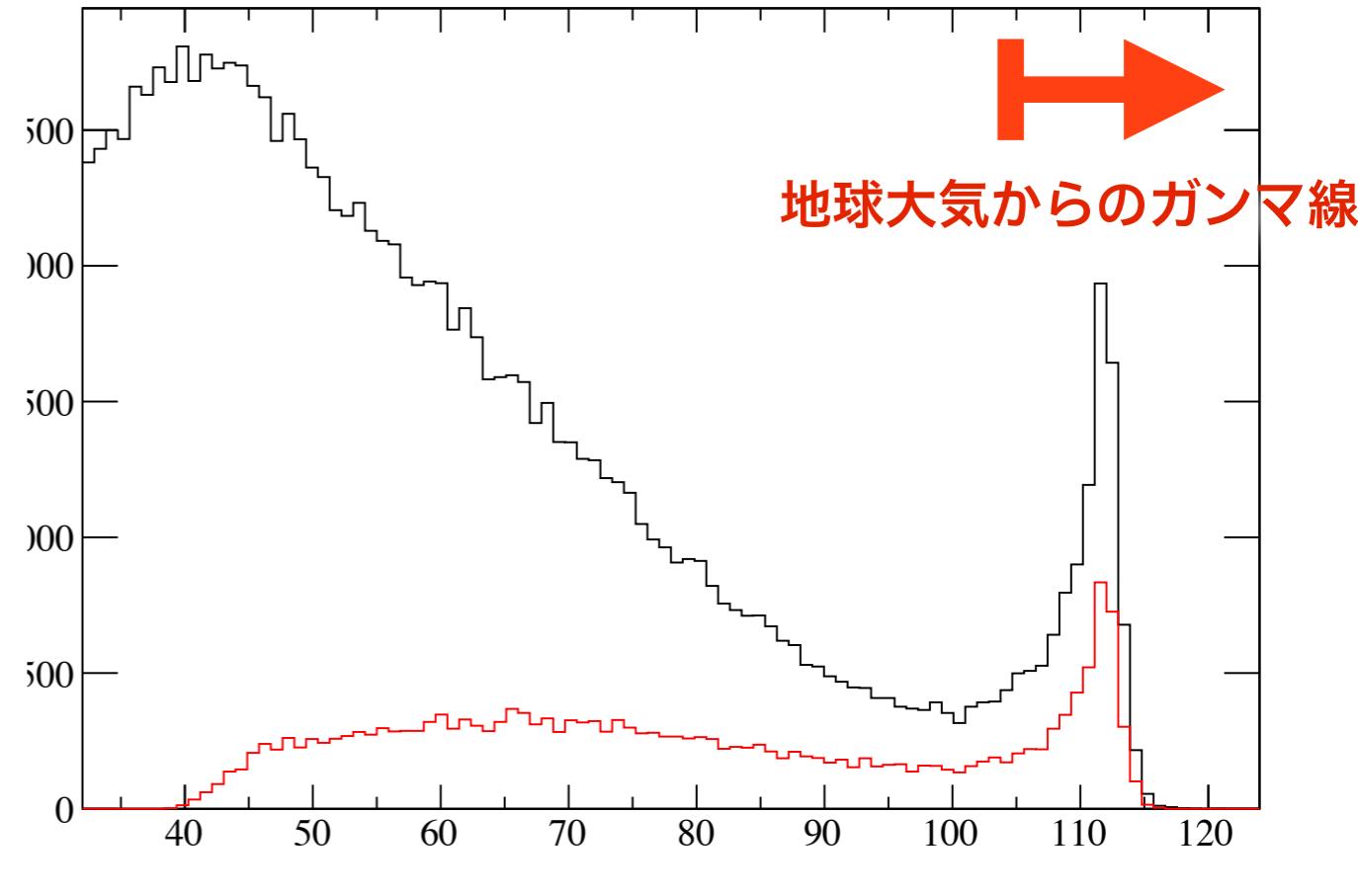
```
root [7] photons->Draw("B:-(L > 180 ? L - 360 : L)>>hGal(720, -180, 180, 360, -90, 90)")  
① 2変数使う ② 銀河中心を図の真ん中に持ってくる  
root [8] hGal  
(TH2F *) 0x7fe63ed034c0 ③ TH2F (float の 2次元) が自動で生成された  
root [9] photons->Draw("B:-(L > 180 ? L - 360 : L)>>hGal(720, -180, 180, 360, -90, 90)", "", "colz")  
④ "colz"をつけて色をつける  
root [10] gPad->SetLogz()
```

変な構造の原因を探る

B:L $\{(L-100)^{**2} + (B-50)^{**2} < 50^{**2}\}$



ZENITH_ANGLE $\{(L-100)^{**2} + (B-50)^{**2} < 50^{**2}\}$



```
root [11] photons->Draw("B:L", "(L-100)**2 + (B-50)**2 < 50**2")
```

- ① 第二引数に条件 (cut) を指定する
- ② cut に当てはまったイベントの数が返り値

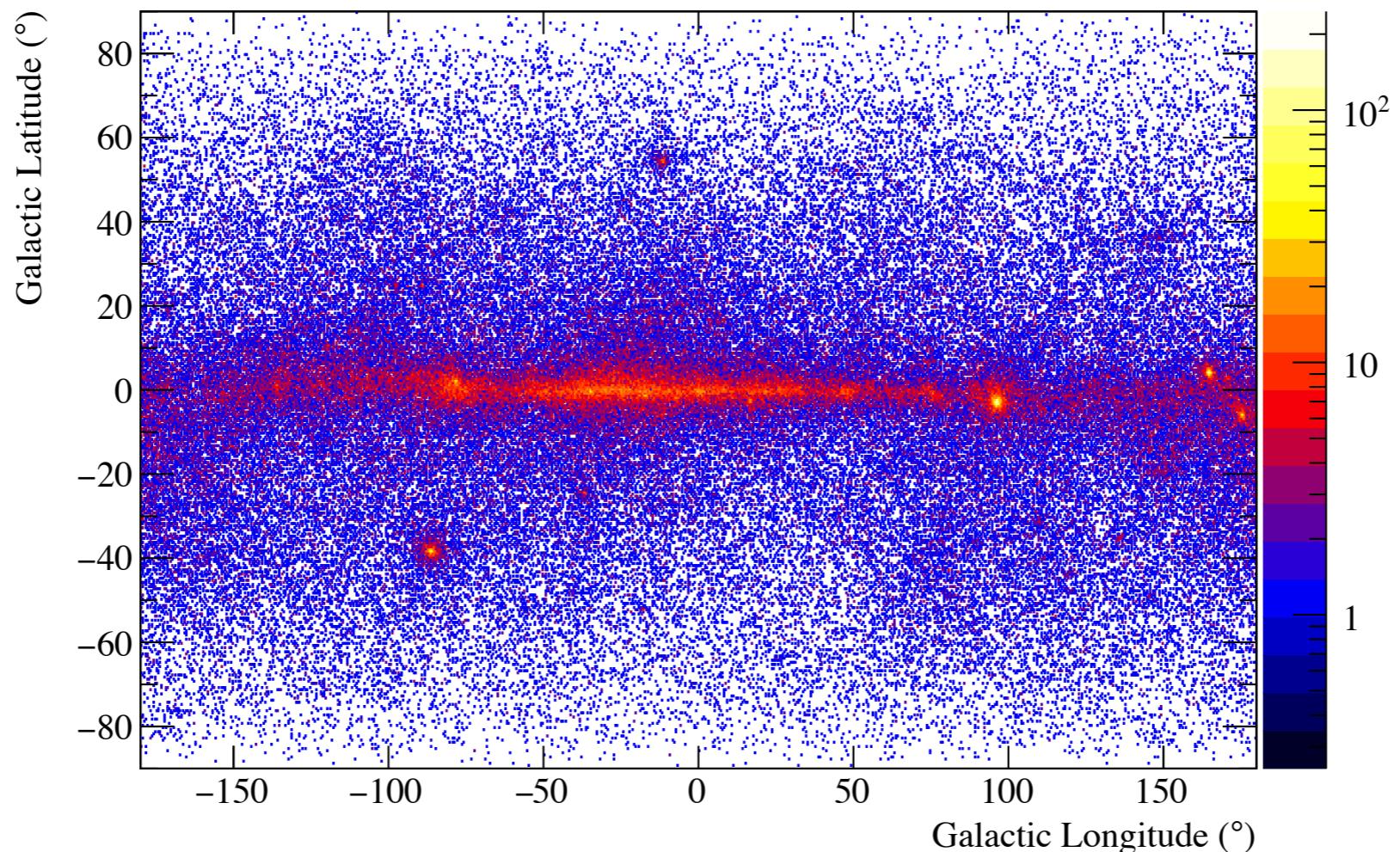
```
(Long64_t) 20676
```

```
root [28] photons->Draw("ZENITH_ANGLE>>z1") ③ 全ガンマ線の天頂角分布
```

```
root [29] photons->Draw("ZENITH_ANGLE>>z2", "(L-100)**2 + (B-50)**2 < 50**2",  
"same") ④ 変な構造周辺のガンマ線のみの天頂角分布
```

```
root [30] z2->SetLineColor(2)
```

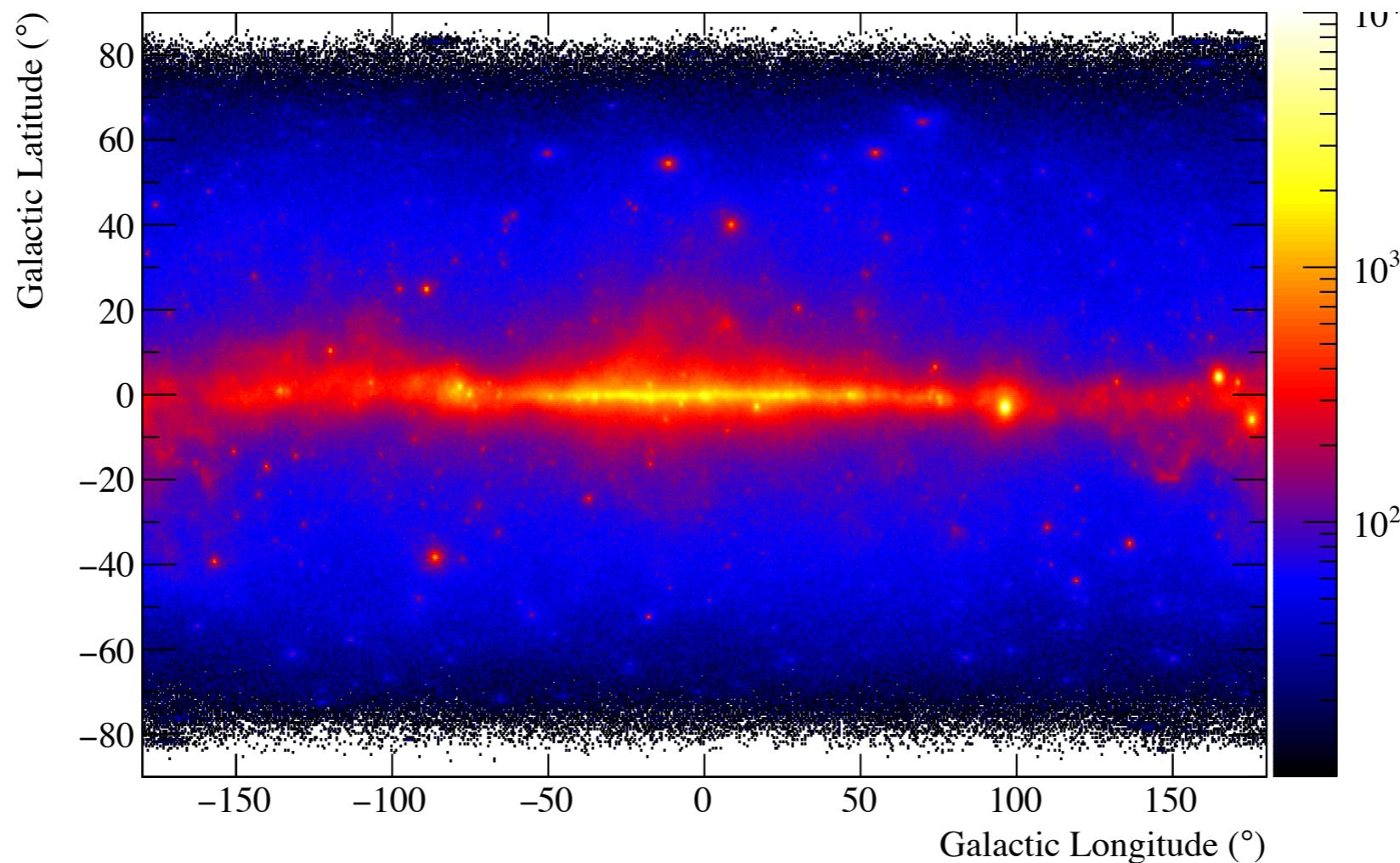
Cut をかけて銀河座標で再度



```
root [6] photons->Draw("B:-(L > 180 ? L - 360 : L)>>hGal", "ZENITH_ANGLE<100",  
"colz")  
root [7] gPad->SetLogz(1)
```

① 天頂角でカットをかけることで必要なデータのみ得られた

複数の ROOT ファイルから TTree を結合する (TChain)



```
root [0] TChain chain("photons")      ① まず同名の TChain を作る
(TChain &) Name: photons Title:
root [1] for(int i = 9; i < 50; i++) chain.Add(Form("lat_photon_weekly_w
%03d_p302_v001_extracted.root", i)); ② 同名の TTree が含まれる ROOT ファイルを追加する
root [2] photons->Draw("B:L>>hGal(720,-180,180,360,-90,90)", "", "colz")
root [3] hGal->SetContour(100)        ③ 後は TTree と同様にできる
root [4] hGal->SetMinimum(10)
root [5] hGal->SetMaximum(1e4)
root [6] hGal->SetTitle(";Galactic Longitude (#circ);Galactic Latitude (#circ)")
root [7] gPad->SetLogz(1)
```

もう少し cut の練習 (TCut を使う)

```
void lat_resolution(const char* directory) {
    TChain* chain = new TChain("photons");
    for(int i = 9; i < 50; i++) chain->Add(Form("%s/lat_photon_weekly_w%03d_p302_v001_extracted.root",
        directory, i));

    TCut cut1("cut1", "ENERGY > 200");
    TCut cut2("cut2", "ENERGY > 1000");

    TCanvas* can = new TCanvas("can", "can", 800, 800);
    can->Divide(2, 2);

    TH2F* hCrab[3];
    TH1D* prox[3];

    for(int i = 0; i < 3; i++) {
        const double kLongitude = 184.33;
        const double kLatitude = -5.47;
        hCrab[i] = new TH2F(Form("hCrab%d", i),
            ";Galactic Longitude (deg);Galactic Latitude (deg)",
            100, kLongitude - 3, kLongitude + 3,
            100, kLatitude - 3, kLatitude + 3);
        can->cd(i + 1);
        if (i == 0) chain->Draw("B:L>>hCrab0", !cut1, "colz"); ② TCut を第二引数に使う
        else if(i == 1) chain->Draw("B:L>>hCrab1", cut1&&(!cut2), "colz");
        else chain->Draw("B:L>>hCrab2", cut2, "colz"); ③ 論理和や論理積も使える

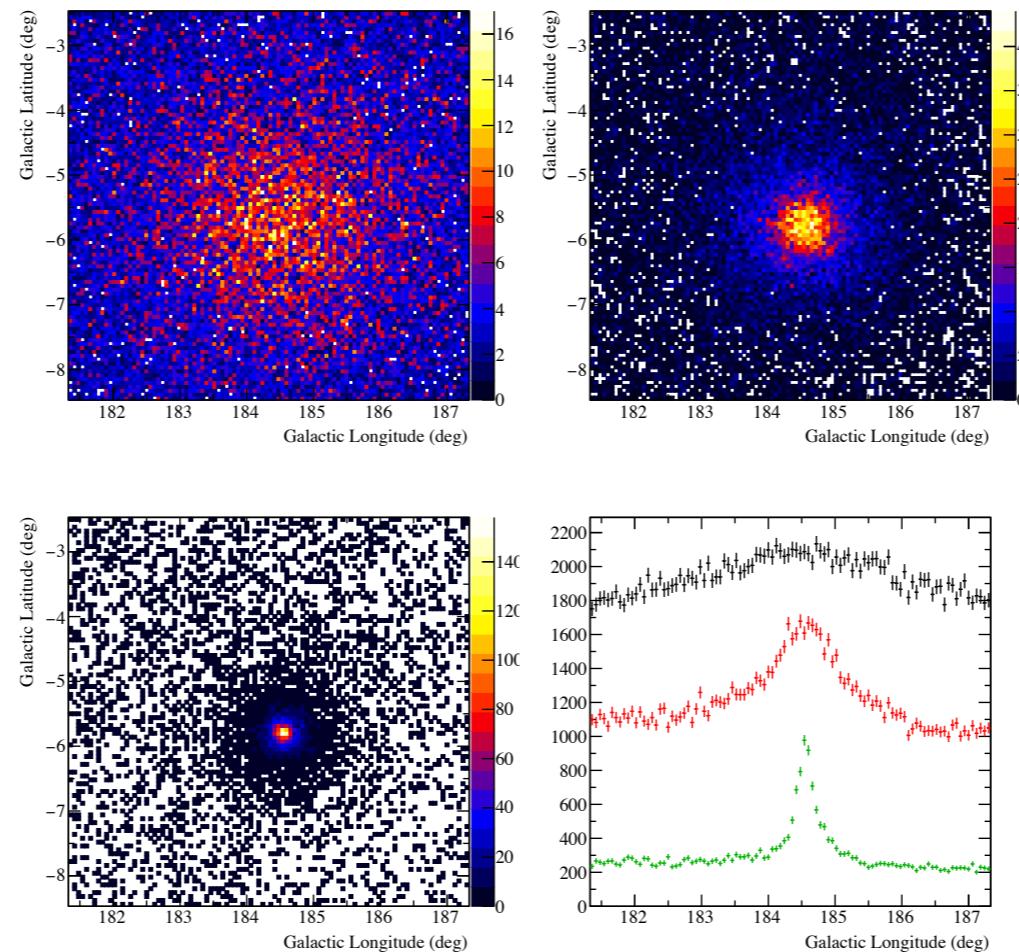
        prox[i] = hCrab[i]->ProjectionX(Form("pro%d", i));
        prox[i]->SetMinimum(0);
        prox[i]->SetMarkerColor(i + 1);
    }
}
```

① TCut を使ってカットを事前に定義する

② TCut を第二引数に使う

③ 論理和や論理積も使える

もう少し cut の練習



```
$ root  
root [0] .x lat_resolution.C("~/git/RHEA-Slides/photons")
```

TTree の作り方

- 来週以降やります…