3.6 DSSD 数据分析

目的：

- 根据相邻条关联和正背面关联，确认多重触发事件的入射位置、能量，并重新组织事件
- 根据位置-能量关联，对事件进行分类
- 得到多重粒子入射事件中，每个事件的pid，能量，位置等信息。

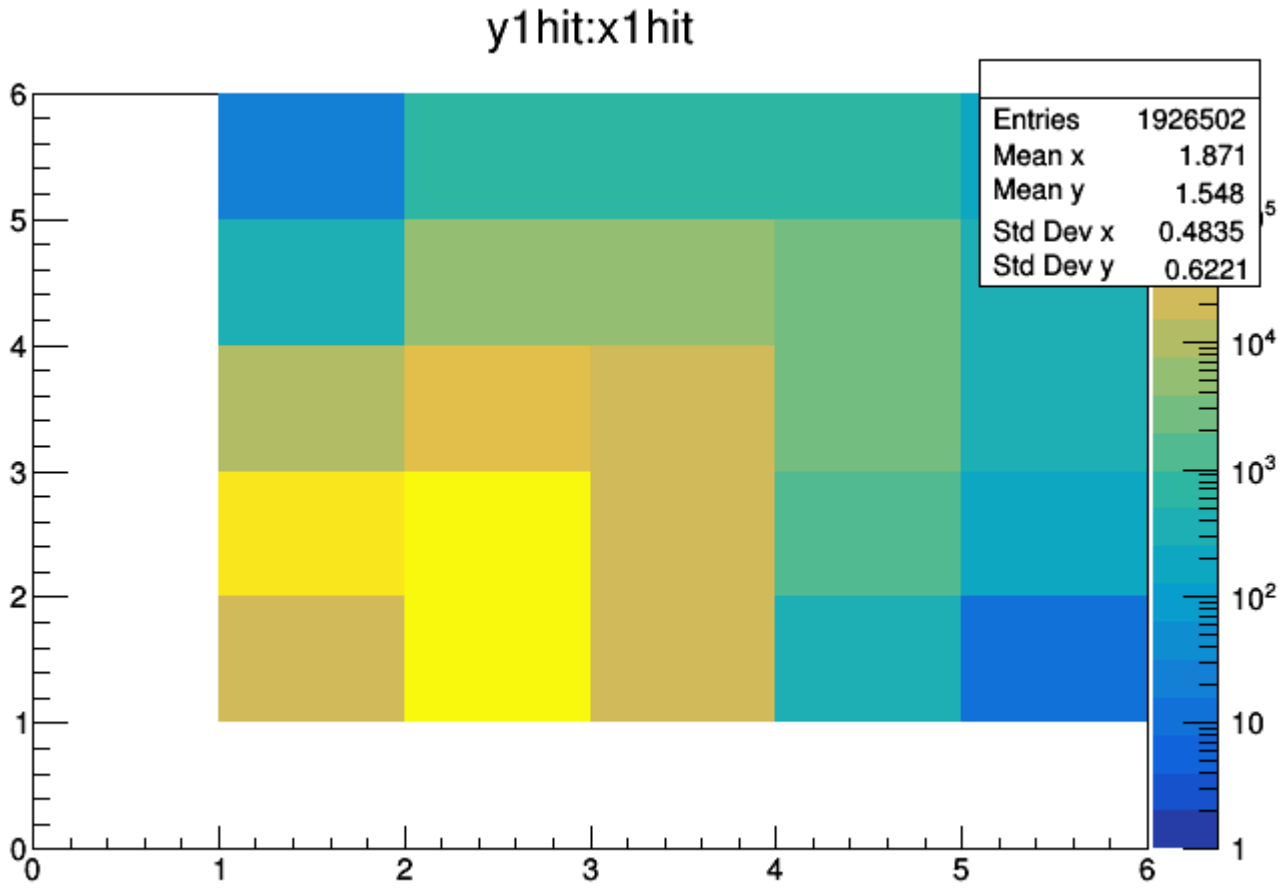
数据：

- d1-d3探测器的幅度已归一化。
- hit结构中能量按照由大到小排序。

In [1]:
```
//%jsroot on
TFile *ipf = new TFile("cal_16C.root");
TTree *tree = (TTree*)ipf->Get("tree");
TCanvas *c1=new TCanvas("c1","c1");
```

观察事件多重性

In [2]:
```
tree->Draw("y1hit:x1hit>>(6,0,6,6,0,6)","","colz");
gPad->SetLogz();
c1->Draw();
```



用Scan观察事件特征，找到分析思路

1. 事件的整体特征

In [3]:
```
tree->Scan("x1s:int(x1e):y1s:int(y1e)","","",10,1);
```

```
*************************************************************************
*    Row   * Instance *       x1s *  int(x1e) *       y1s *  int(y1e) *
*************************************************************************
*        1 *        0 *        17 *      4924 *        18 *      5887 *
*        1 *        1 *        18 *      1007 *           *           *
*        2 *        0 *        23 *      3146 *        11 *      3365 *
*        2 *        1 *        24 *       339 *        10 *       115 *
*        3 *        0 *        23 *      5465 *        15 *      4568 *
*        3 *        1 *           *           *        16 *       911 *
*        4 *        0 *        21 *      5993 *        15 *      5743 *
*        4 *        1 *           *           *        14 *       270 *
*        5 *        0 *        11 *      4195 *        12 *      5320 *
*        5 *        1 *        12 *      1159 *           *           *
*        6 *        0 *        14 *      3889 *        16 *      4245 *
*        6 *        1 *        13 *       388 *           *           *
*        7 *        0 *        12 *      4612 *         9 *      4878 *
*        7 *        1 *        11 *       306 *           *           *
*        8 *        0 *        22 *      4127 *         9 *      3813 *
*        8 *        1 *           *           *        10 *       335 *
*        9 *        0 *        13 *      7003 *        12 *      6646 *
*        9 *        1 *        10 *       143 *        13 *       425 *
*        9 *        2 *        14 *       107 *           *           *
*       10 *        0 *        20 *      3986 *         8 *      3044 *
*       10 *        1 *           *           *         9 *       961 *
*************************************************************************
```

- 事件2，一个粒子入射
  - x:相邻两重，y:相邻两重；x总能量(3485)，y总能量很接近(3480)
- 事件3，4，5，6，8，10，一个粒子入射
  - x，y:一重和相邻两重的组合；x能量，y能量很接近
- 事件1，7

- x，y:一重和相邻两重的组合；x能量与y能量相差40左右
- 事件9，
  - x:三重，其中有两条相邻(13,14)，y:相邻两重(12,13)；x,y面相邻条总能量相近，x面另一条(10)?
  - 两侧总能量不一致。

2. 非相邻两重事件

In [4]:
```
tree->Scan("x1s:int(x1e):y1s:int(y1e)","x1hit==2 && abs(x1s[0]-x1s[1])>1 && y1hit==2","",200,1);
```

```
************************************************************************
*    Row   * Instance *       x1s *  int(x1e) *       y1s *  int(y1e) *
************************************************************************
*       23 *        0 *         4 *       634 *         7 *       630 *
*       23 *        1 *        10 *       486 *        23 *       486 *
*       41 *        0 *         3 *      1188 *        18 *      1187 *
*       41 *        1 *        11 *       401 *        13 *       434 *
*       48 *        0 *        26 *      1349 *        23 *      1350 *
*       48 *        1 *        29 *       445 *        21 *       448 *
*       80 *        0 *        25 *       484 *        27 *       484 *
*       80 *        1 *        27 *       429 *        20 *       428 *
*      105 *        0 *        13 *      2291 *         9 *      2291 *
*      105 *        1 *        10 *      1935 *        16 *      1934 *
*      126 *        0 *        26 *       946 *        16 *       944 *
*      126 *        1 *        28 *       513 *         9 *       514 *
*      166 *        0 *        18 *      1507 *        28 *      1506 *
*      166 *        1 *        25 *       680 *         5 *       680 *
************************************************************************
==> 14 selected entries
```

- 显示的非相邻两重事件，的x-y都有很好的关联,可以明确指定位置组合。

3. 相邻两重事件

In [5]:
```
tree->Scan("x1s:int(x1e):y1s:int(y1e)","x1hit==2 && abs(x1s[0]-x1s[1])==1 && abs(x1e-y1e)<20 && y1hit==2","",1000,1);
```

```
************************************************************************
*    Row   * Instance *       x1s *  int(x1e) *       y1s *  int(y1e) *
************************************************************************
*       15 *        0 *        10 *      4785 *        17 *      4772 *
*       15 *        1 *         9 *       267 *        18 *       261 *
*       47 *        1 *        18 *      1062 *        14 *      1045 *
*      116 *        1 *        11 *       291 *        11 *       306 *
*      388 *        0 *        17 *      4138 *        11 *      4135 *
*      388 *        1 *        16 *       460 *        27 *       449 *
*      399 *        0 *        13 *      4651 *        16 *      4645 *
*      399 *        1 *        14 *       315 *        15 *       303 *
*      665 *        0 *        22 *       712 *        11 *       695 *
*      665 *        1 *        23 *       555 *        14 *       549 *
*      698 *        0 *        30 *       731 *        18 *       726 *
*      698 *        1 *        29 *       636 *        12 *       632 *
*      700 *        0 *        10 *      4099 *         9 *      4083 *
*      700 *        1 *        11 *      1241 *         8 *      1246 *
*      761 *        0 *        16 *      5752 *        12 *      5742 *
*      761 *        1 *        17 *       154 *        11 *       145 *
*      821 *        0 *        13 *      4582 *        10 *      4575 *
*      821 *        1 *        14 *       292 *        11 *       277 *
*      851 *        0 *        16 *      2728 *        12 *      2713 *
*      851 *        1 *        17 *       338 *        11 *       348 *
************************************************************************
==> 20 selected entries
```

- 很多相邻两重事件都是由两个入射粒子引起的。

4. 三重事件

In [6]:
```
tree->Scan("x1s:int(x1e):y1s:int(y1e)","x1hit==3 && abs(x1s[0]-x1s[1])>1","",500,1);
```

```
*************************************************************************
*    Row   * Instance *       x1s *  int(x1e) *       y1s *  int(y1e) *
*************************************************************************
*        9 *        0 *        13 *      7003 *        12 *      6646 *
*        9 *        1 *        10 *       143 *        13 *       425 *
*        9 *        2 *        14 *       107 *           *           *
*      118 *        0 *        18 *       979 *         3 *       981 *
*      118 *        1 *        12 *       519 *        28 *       517 *
*      118 *        2 *        10 *       414 *        11 *       408 *
*      192 *        0 *         2 *       682 *        14 *       682 *
*      192 *        1 *        12 *       591 *         6 *       590 *
*      192 *        2 *        11 *       566 *         5 *       476 *
*      192 *        3 *           *           *         4 *       106 *
*      193 *        0 *        16 *      2604 *        18 *      2607 *
*      193 *        1 *        13 *       536 *        23 *       535 *
*      193 *        2 *         1 *       444 *         2 *       442 *
*      272 *        0 *         4 *      2099 *        17 *      2098 *
*      272 *        1 *        28 *       783 *        23 *      1319 *
*      272 *        2 *        27 *       563 *           *           *
*      282 *        0 *        13 *      1289 *         4 *      1296 *
*      282 *        1 *         9 *       587 *        24 *       606 *
*      282 *        2 *         6 *       575 *        23 *       152 *
*      282 *        3 *           *           *        21 *       148 *
*      289 *        0 *         8 *      1747 *        13 *      1751 *
*      289 *        1 *        14 *       487 *        25 *       490 *
*      289 *        2 *        13 *       157 *           *           *
*      386 *        0 *         8 *       821 *        21 *       821 *
*      386 *        1 *        13 *       763 *        13 *       760 *
*      386 *        2 *        15 *       463 *        23 *       461 *
*      398 *        0 *        12 *      1865 *        18 *      1858 *
*      398 *        1 *         5 *       532 *        24 *       531 *
*      398 *        2 *        13 *       187 *        17 *       189 *
*      429 *        0 *        10 *      4138 *        14 *      4137 *
*      429 *        1 *        12 *       404 *         4 *       401 *
*      429 *        2 *         6 *       130 *         5 *       132 *
*************************************************************************
==> 32 selected entries

Type <CR> to continue or q to quit ==>
```

也可以清晰地看见有三个粒子入射的情况:

- 事件118:粒子1(18,3),粒子2(12,28),粒子3(10,11)

- 事件193：

其他：

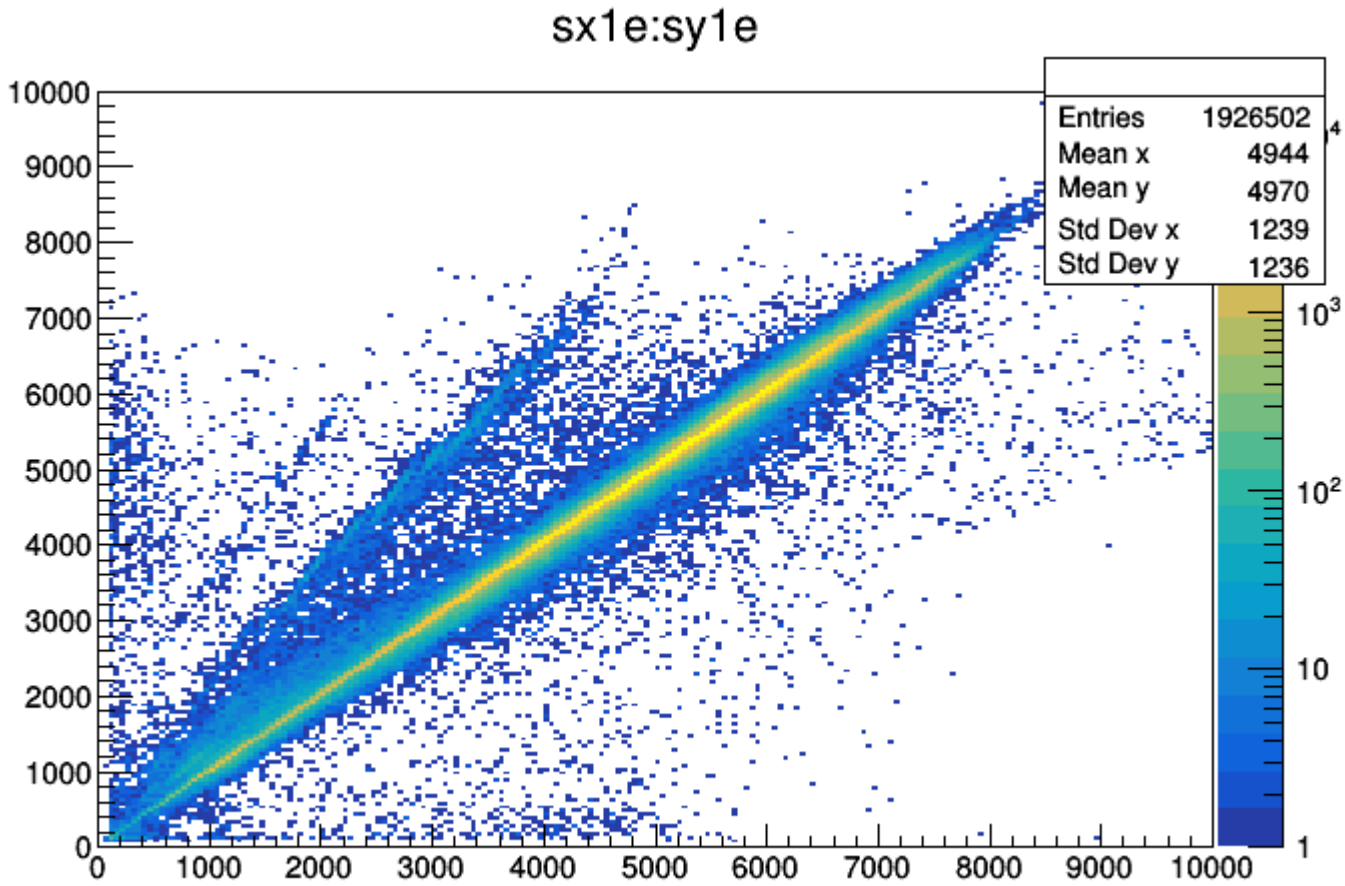
- 事件192:粒子1(2,14),粒子2((12,11),(6,5,7))
- 事件398:x和y方向都有相邻条，但每条之间x，y能量都有很好的关联，因此很大概率是三个入射粒子。
- 事件9,289等两侧总能量不一致。

总结

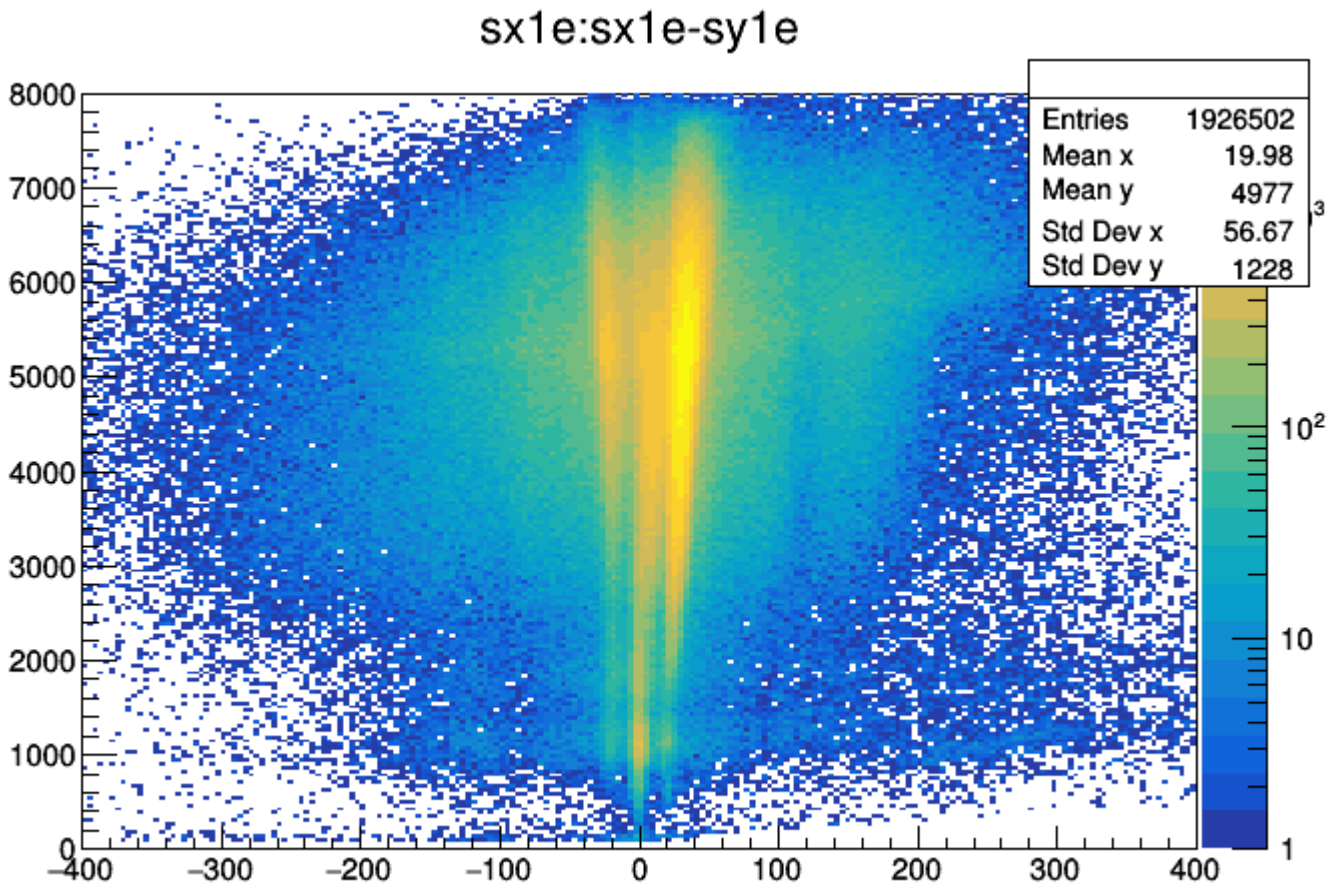
1. 正背面关联关系可以对事件类型(是否为多重，相邻条crosstalk)做很好的区分。
2. 当两侧总能量不一致时，可以认为事件为而非正常事件。
    - 部分能量没有被记录，或有偶然符合
    - 非正常事件，做标记后待后续处理确认。
3. 新参数：
    - 每个探测器x方向总能量：sx1e,sx2e,sx3e;
    - y方向总能量：sy1e,sy2e,sy3e;

画图观察，并确定x-y关联的能量差范围

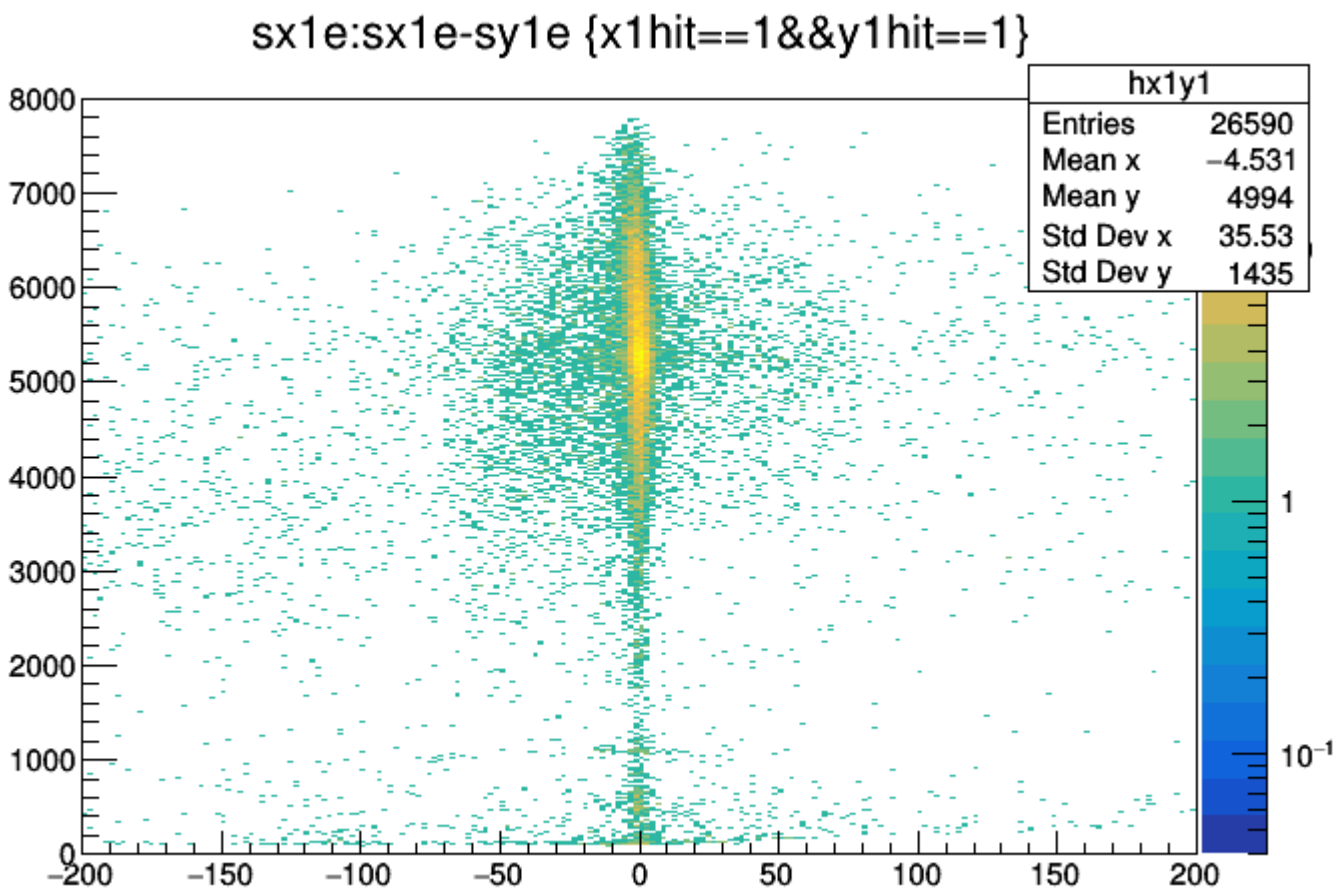
In [7]:
```
tree->Draw("sx1e:sy1e>>(200,0,10000,200,0,10000)","","colz");
gPad->SetLogz();
c1->Draw();
```



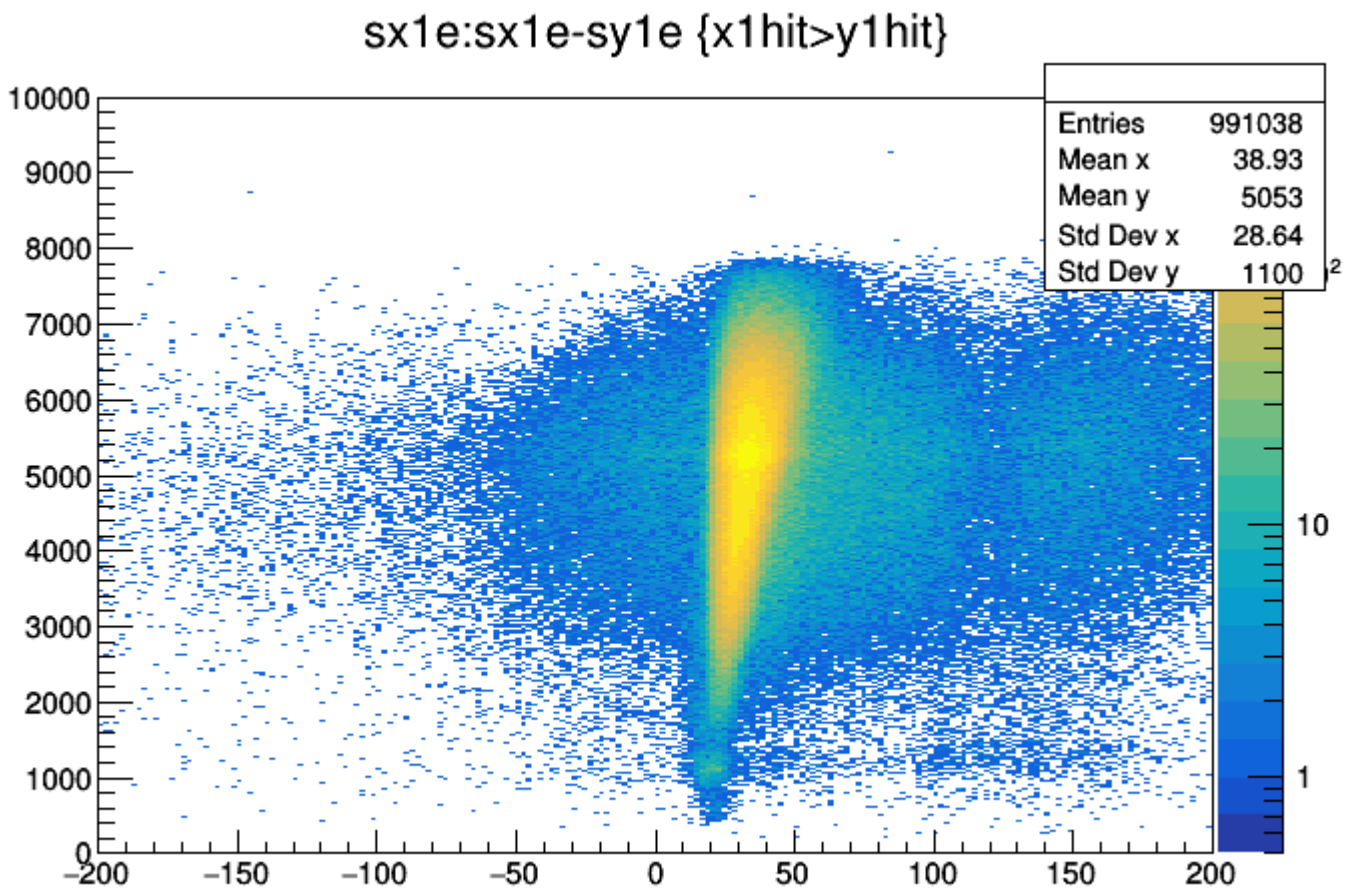
In [8]:
```
tree->Draw("sx1e:sx1e-sy1e>>(200,-400,400,200,0,8000)","","colz");
c1->Draw();
```



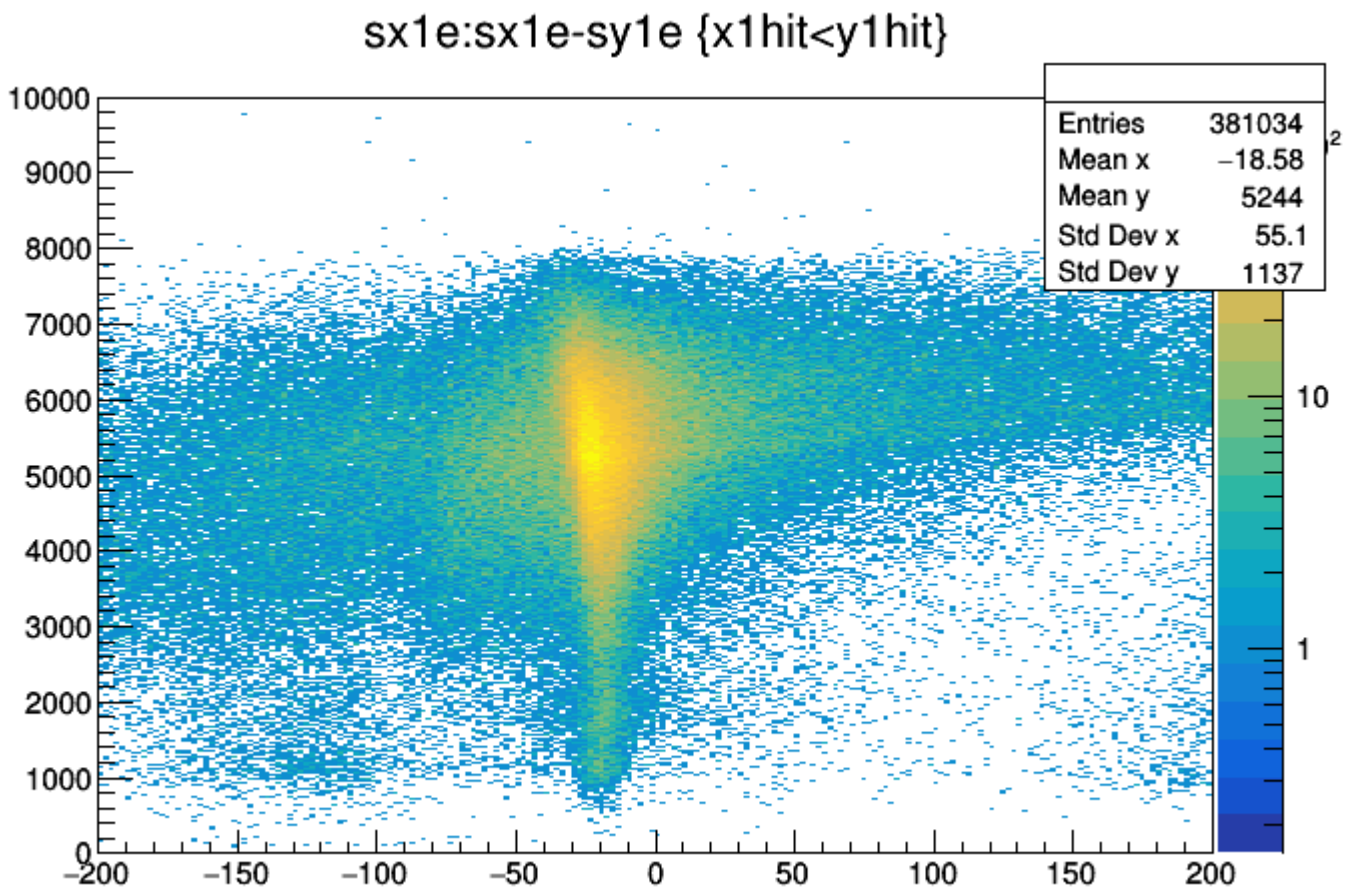
In [9]:
```
tree->Draw("sx1e:sx1e-sy1e>>hx1y1(200,-200,200,1000,0,8000)","x1hit==1&&y1hit==1","colz");
c1->Draw();//-10~+10
```



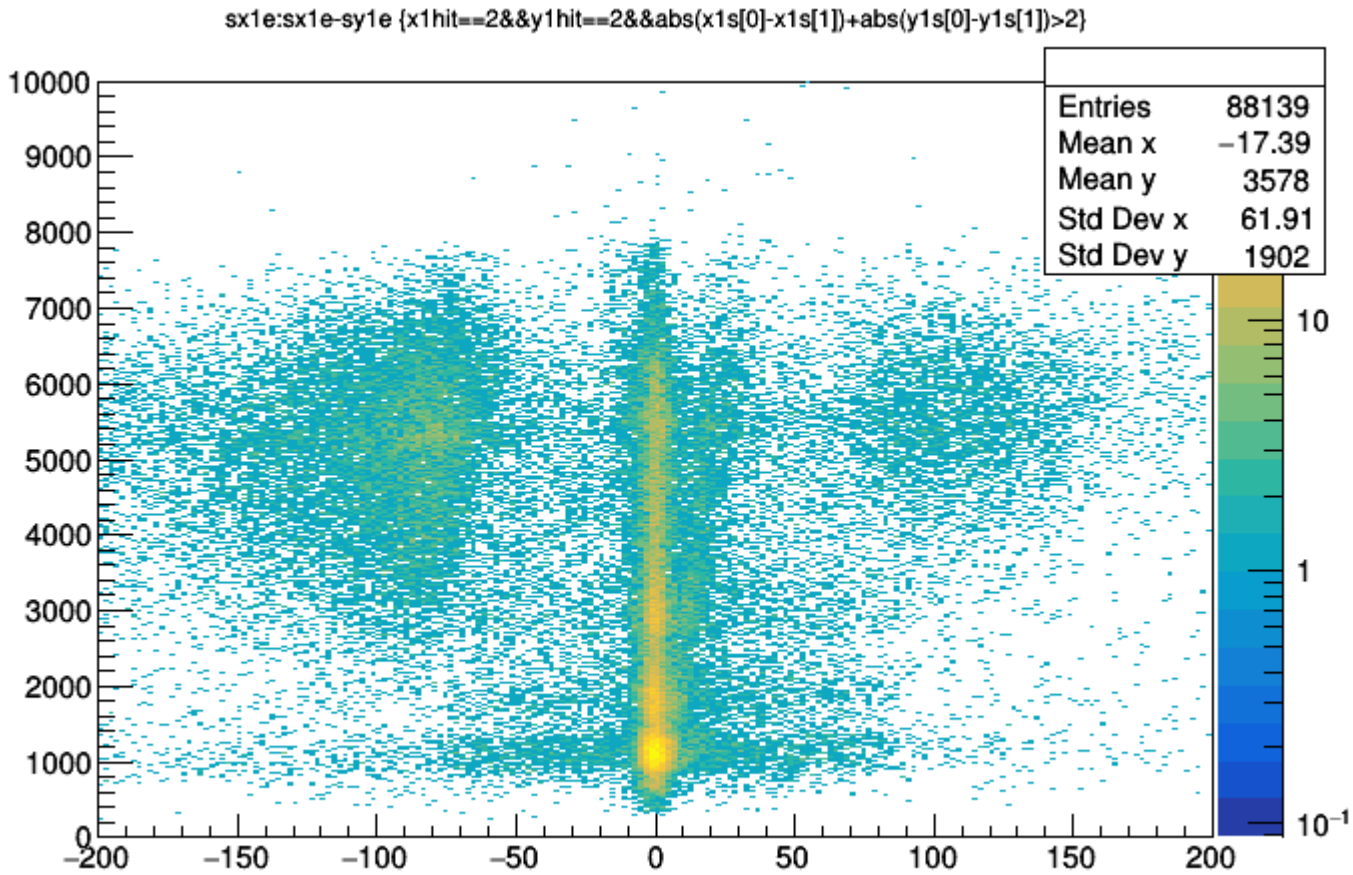
In [10]:
```
tree->Draw("sx1e:sx1e-sy1e>>(200,-200,200,1000,0,10000)","x1hit>y1hit","colz");
c1->Draw(); //0~+60 x1hit>y1hit;
```



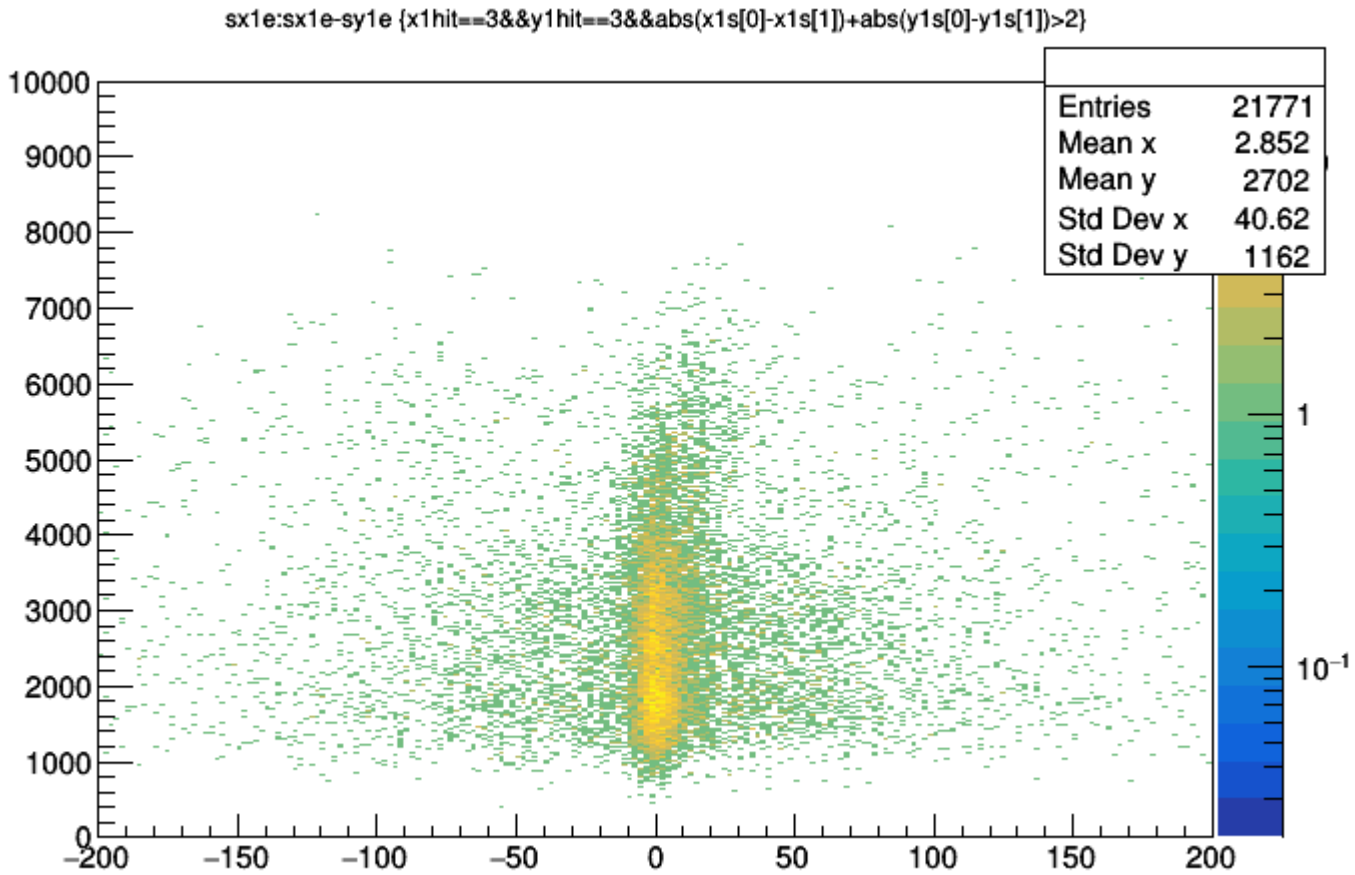
In [11]:
```
tree->Draw("sx1e:sx1e-sy1e>>(200,-200,200,1000,0,10000)","x1hit<y1hit","colz");
c1->Draw(); //-60~+10;
```



In [12]:
```
tree->Draw("sx1e:sx1e-sy1e>>(200,-200,200,1000,0,10000)","x1hit==2&&y1hit==2&&abs(x1s[0]-x1s[1])+abs(y1s[0]-y1s[1])>2","
c1->Draw();//-10~+10 ; x1hit=y1hit==2
```



In [13]:
```
tree->Draw("sx1e:sx1e-sy1e>>(200,-200,200,1000,0,10000)","x1hit==3&&y1hit==3&&abs(x1s[0]-x1s[1])+abs(y1s[0]-y1s[1])>2","
c1->Draw();//-10~+10 ; x1hit=y1hit==3
```



总结

- 当两侧多重性不一致时：sx1e-sy1e 中心值不为零，向正(x1hit>y1hit),负(x1hit<y1hit)的一侧偏移
  - 意味着在能量沉积大时，有些条的感应信号因为阈值(硬件pedal的阈值)没有被记录
  - 但这个问题实际并不严重：在能量接近8000的时候最大偏移在50左右，对能量的相对影响为$\Delta E/E = 50/8000 \approx 0.006$, 小于1%。

- 对于多粒子入射引起的x,y多重性事件：abs(sx1e-sy1e)<10
  - 当相邻条满足上述条件，也认为是两个粒子入射导致的。

新文件

- 重新对事件进行分类，确定每个事件的实际多重性(多重性=入射粒子数目)

部分代码示例

```
Int_t x1h,y1h,x2h,y2h,x3h,y3h;//final multiplicity
Double_t x1ea[100],x2ea[100],x3ea[100];//x-energy
Double_t y1ea[100],y2ea[100],y3ea[100];//y-energy
Int_t x1sa[100],x2sa[100],x3sa[100];//x-strip
Int_t y1sa[100],y2sa[100],y3sa[100];//y-strip
Int_t flag1,flag2,flag3;//event flag
```

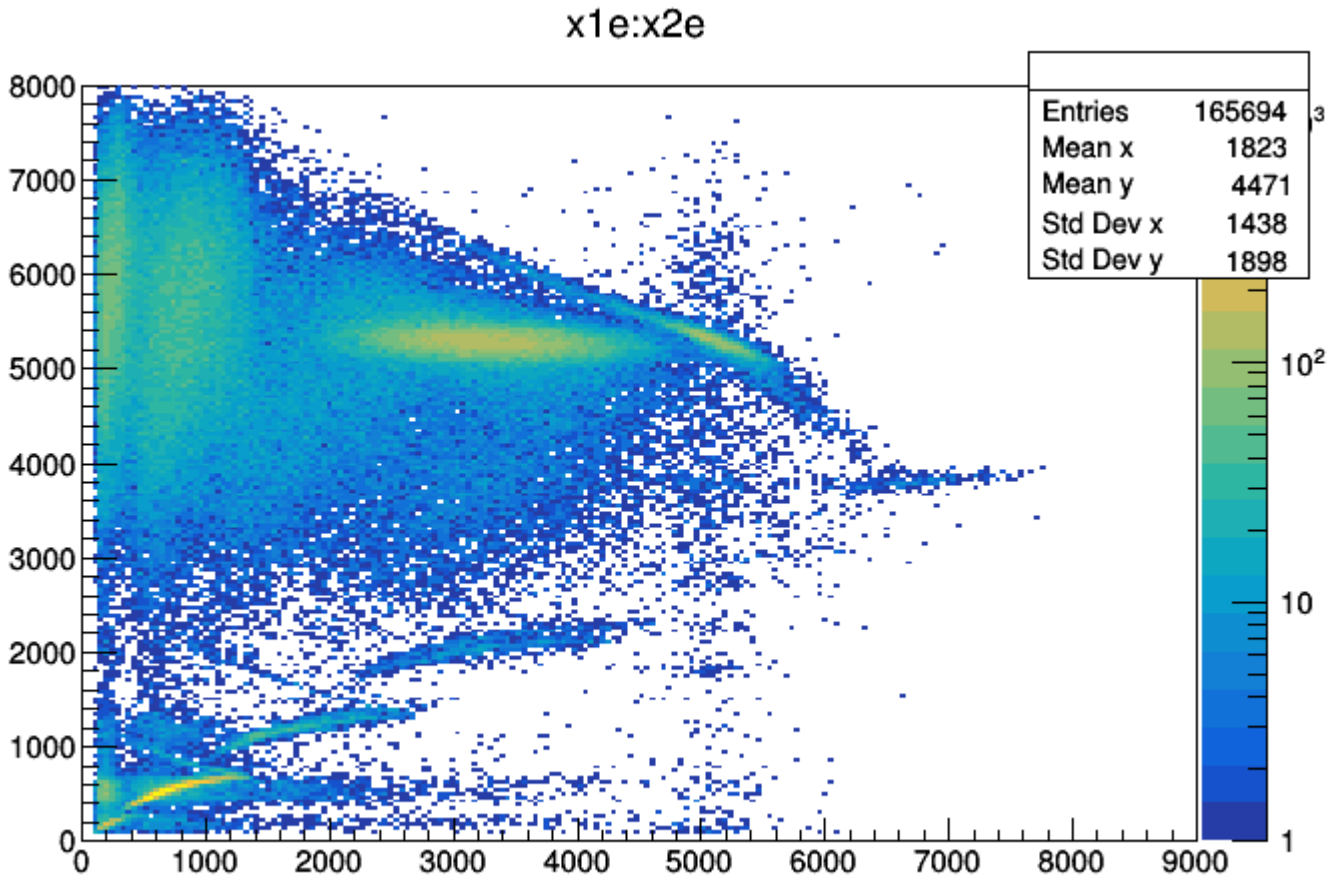
```
////////////////////////////////////
      //one-one DSSD1
   if(x1hit==1 && y1hit==1 && abs(sx1e-sy1e)<10) {
        x1h=1;y1h=1;
        x1ea[0]=x1e[0]; x1sa[0]=x1s[0];
        y1ea[0]=y1e[0]; y1sa[0]=y1s[0];
        flag1=111;
     }
////////////////////////////////////

////////////////////////////////////
      //two-two DSSD1
   if(x1hit==y1hit && x1hit==2) {
        if(abs(x1e[0]-y1e[0])<10 && abs(x1e[1]-y1e[1])<10) {
            x1h=2;y1h=2;
            for(int i=0;i<2;i++) {
                x1ea[i]=x1e[i]; x1sa[i]=x1s[i];
                y1ea[i]=y1e[i]; y1sa[i]=y1s[i];
            }
        flag1=222;
        }
        else if(sx1e-sy1e>-10 && sx1e-sy1e<60) {
            if(abs(x1s[0]-x1s[1])==1 && abs(y1s[0]-y1s[1])==1) {
                x1h=1;y1h=1;
                x1ea[0]=x1e[0]+x1e[1];
                x1sa[0]=x1s[0];
                y1ea[0]=y1e[0]+y1e[1];
                y1sa[0]=y1s[0];
                flag1=221;
              }
            }
    }
////////////////////////////////////
//three-two DSSD1
  if(x1hit==2 && y1hit==3) {
    if(sx1e-sy1e>-35 && sx1e-sy1e<10) {
      int ix=-1,iy=-1;
      for(int i=0;i<2;i++)
        for(int j=0;j<3;j++) {
```

```
            if(abs(x1e[i]-y1e[j])<10) {//search x-y pair
              ix=i;
              iy=j;
             }
           }
        if(ix>-1) {
          x1sa[0]=-1; x1sa[1]=-1;
          y1sa[0]=-1; y1sa[1]=-1;

          x1ea[0]=x1e[ix]; x1sa[0]=x1s[ix];
          y1ea[0]=y1e[iy]; y1sa[0]=y1s[iy];

          if(ix==0) {
            x1ea[1]=x1e[1]; x1sa[1]=x1s[1];
          }
          else {
            x1ea[1]=x1e[0]; x1sa[1]=x1s[0];
          }
          if(iy==0 &&abs(y1s[1]-y1s[2])==1) {
            y1ea[1]=y1e[1]+y1e[2]; y1sa[1]=y1s[1];
          }
          if(iy==1 &&abs(y1s[0]-y1s[2])==1) {
            y1ea[1]=y1e[0]+y1e[2]; y1sa[1]=y1s[0];
          }
          if(iy==2 &&abs(y1s[0]-y1s[1])==1) {
            y1ea[1]=y1e[0]+y1e[1]; y1sa[1]=y1s[0];
          }
          if(x1sa[0]>-1 && x1sa[1]>-1 &&y1sa[0]>-1 && y1sa[1]>-1) {
            x1h=2;y1h=2;
            flag1=232;
          }
        }
      }
    }

  if(x1hit==3 && y1hit==2) {
     if(sx1e-sy1e>5 && sx1e-sy1e<65) {
       int ix=-1,iy=-1;
       for(int i=0;i<3;i++)
```

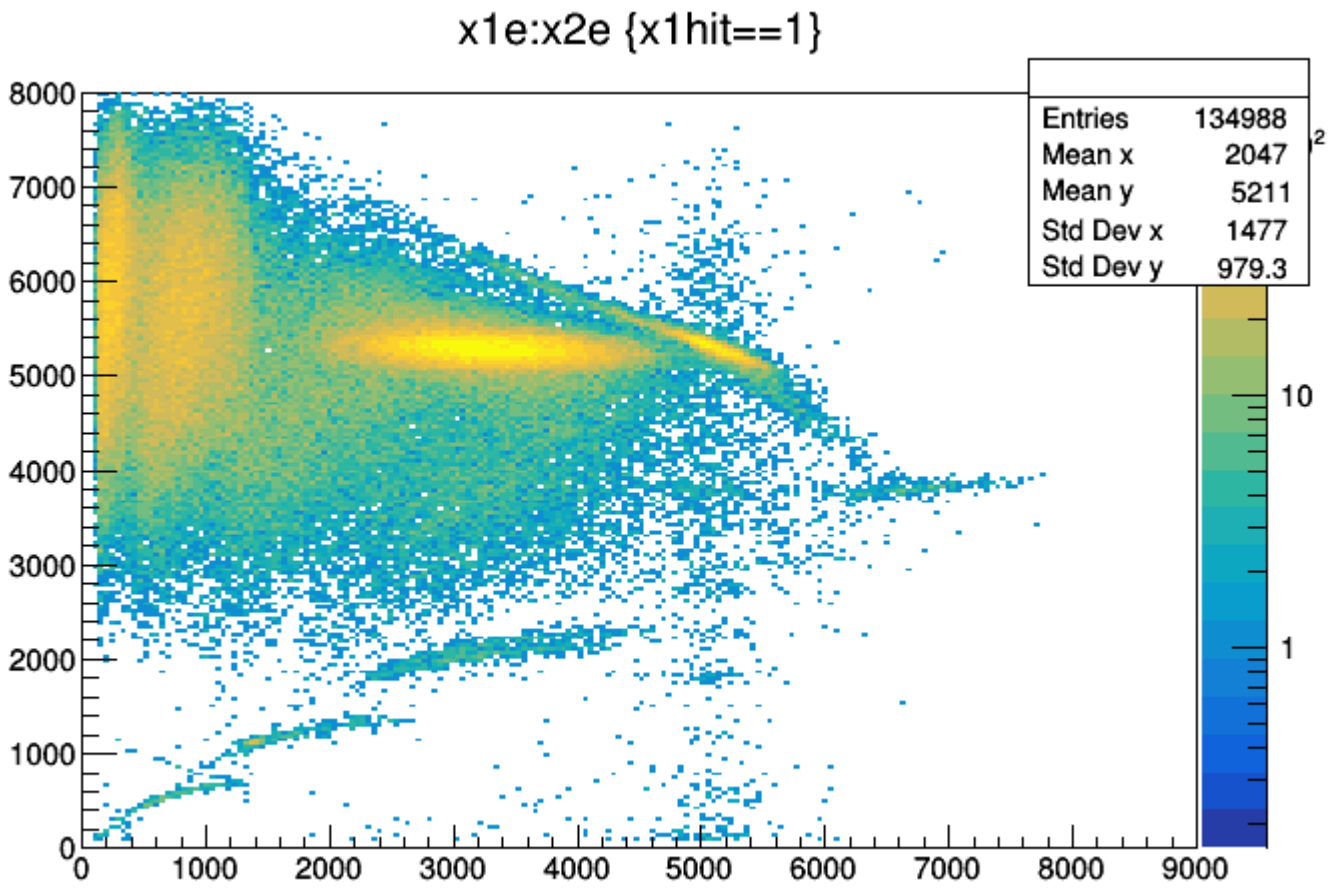
```
          for(int j=0;j<2;j++) {
            if(abs(x1e[i]-y1e[j])<10) {//search x-y pair
              ix=i;
              iy=j;
            }
          }
        if(ix>-1) {
          x1sa[0]=-1; x1sa[1]=-1;
          y1sa[0]=-1; y1sa[1]=-1;
          x1ea[0]=x1e[ix]; x1sa[0]=x1s[ix];
          y1ea[0]=y1e[iy]; y1sa[0]=y1s[iy];

          if(iy==0) {
            y1ea[1]=y1e[1]; y1sa[1]=y1s[1];
          }
          else {
            y1ea[1]=y1e[0]; y1sa[1]=y1s[0];
          }
          if(ix==0 &&abs(x1s[1]-x1s[2])==1) {
            x1ea[1]=x1e[1]+x1e[2]; x1sa[1]=x1s[1];
          }
          if(ix==1 &&abs(x1s[0]-x1s[2])==1) {
            x1ea[1]=x1e[0]+x1e[2]; x1sa[1]=x1s[0];
          }
          if(ix==2 &&abs(x1s[0]-x1s[1])==1) {
            x1ea[1]=x1e[0]+x1e[1]; x1sa[1]=x1s[0];
          }
          if(x1sa[0]>-1 && x1sa[1]>-1 &&y1sa[0]>-1 && y1sa[1]>-1) {
            x1h=2;y1h=2;
            flag1=232;
          }
        }
      }
    }

... ... ...
 if((x1h>0 && x1h==x2h && x3h==0) || (x1h>0 && x1h==x2h && x2h==x3h)) tree->Fill();
```

In [14]:
```
TFile *fsort=new TFile("sort_16C.root");
TTree *tr=(TTree*)fsort->Get("tree");
//hit:    x1hit,x2hit,x3hit,y1hit,y2hit,y3hit;
//energy: x1e,x2e,x3e,y1e,y2e,y3e;
//strip:  x1s,x2s,x3s,y1s,y2s,y2s;
```

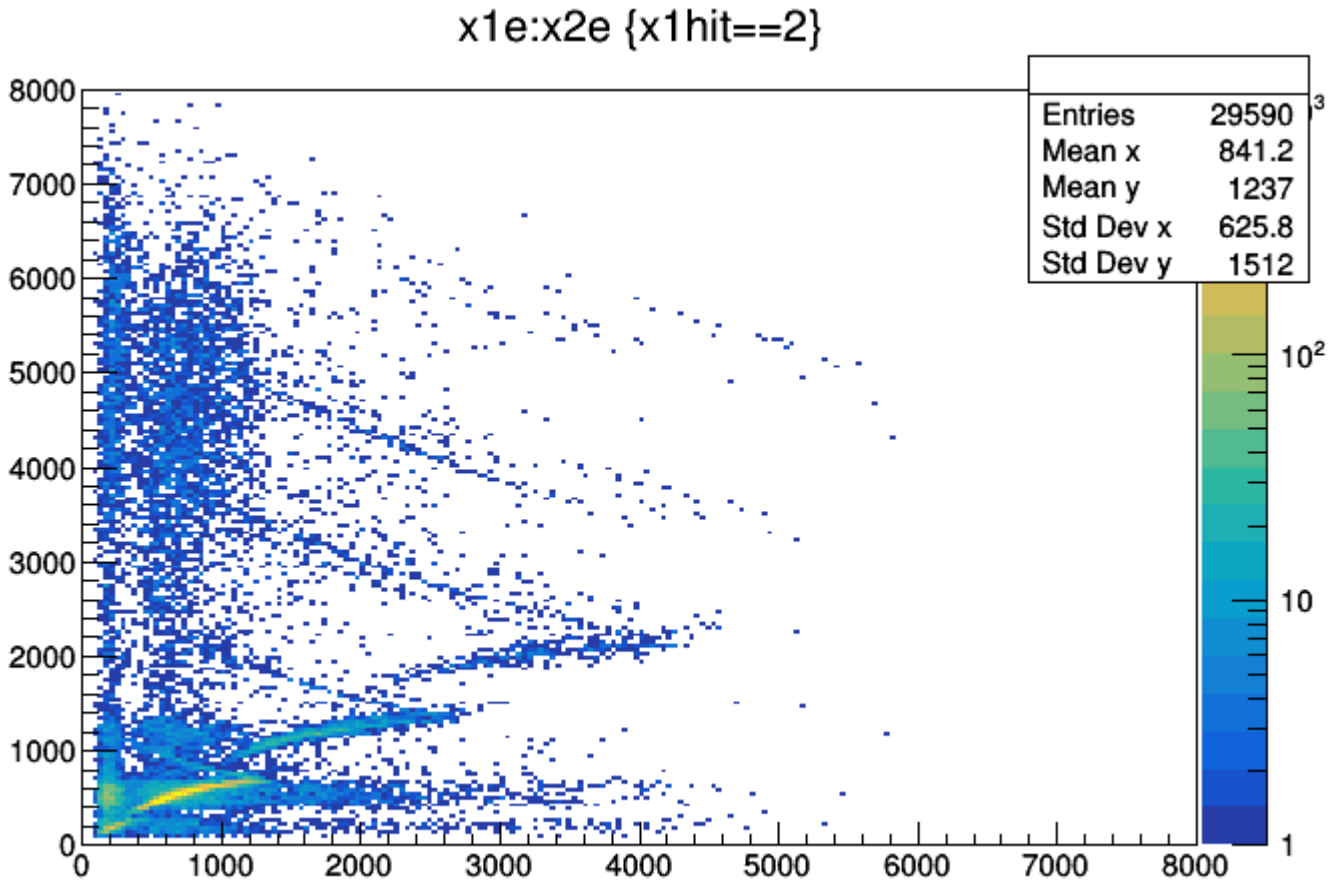
In [15]:
```
tr->Draw("x1e:x2e>>(200,0,9000,200,0,8000)","","colz");
c1->Draw();
```



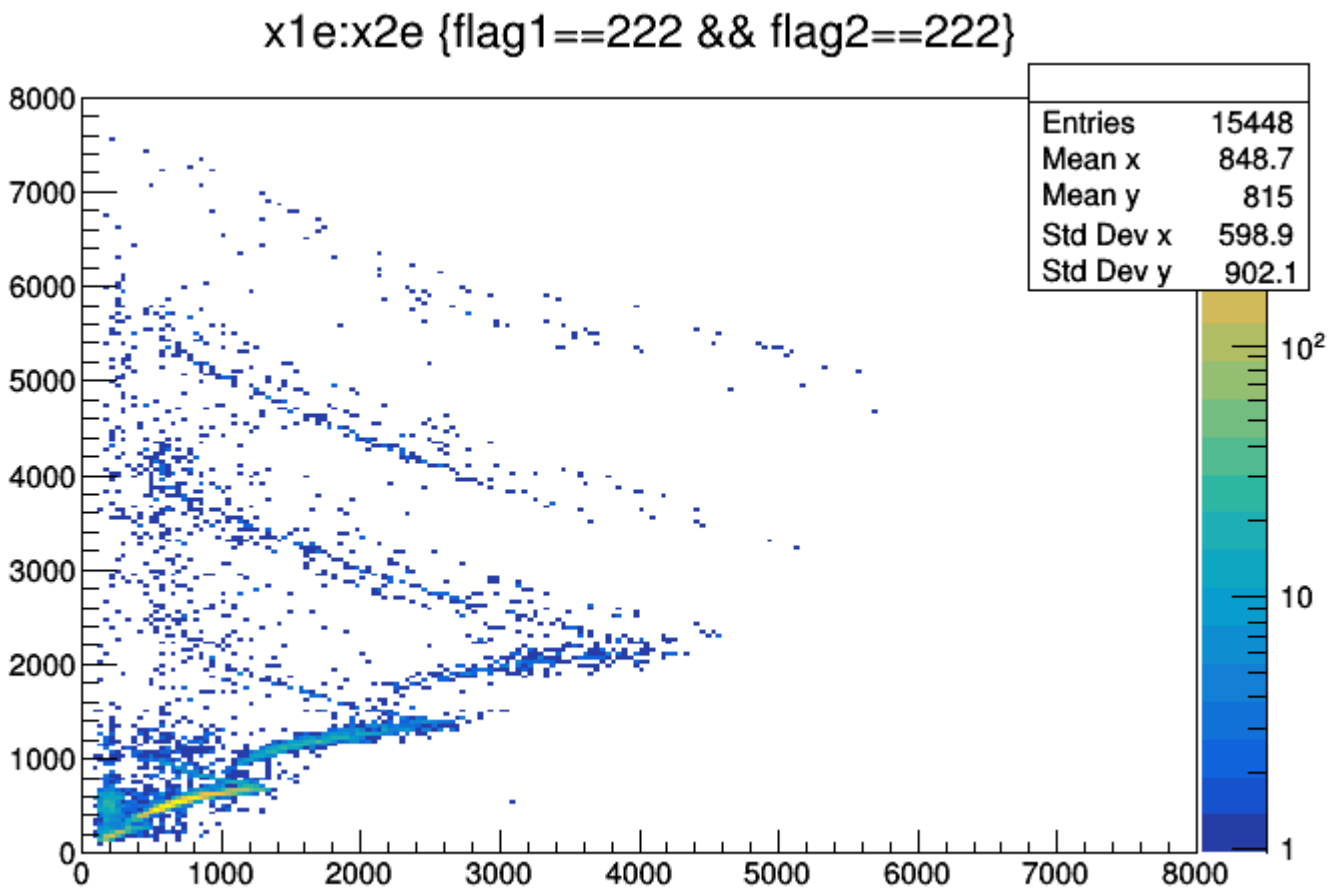
In [16]:
```
tr->Draw("x1e:x2e>>(200,0,9000,200,0,8000)","x1hit==1","colz");
c1->Draw();
```



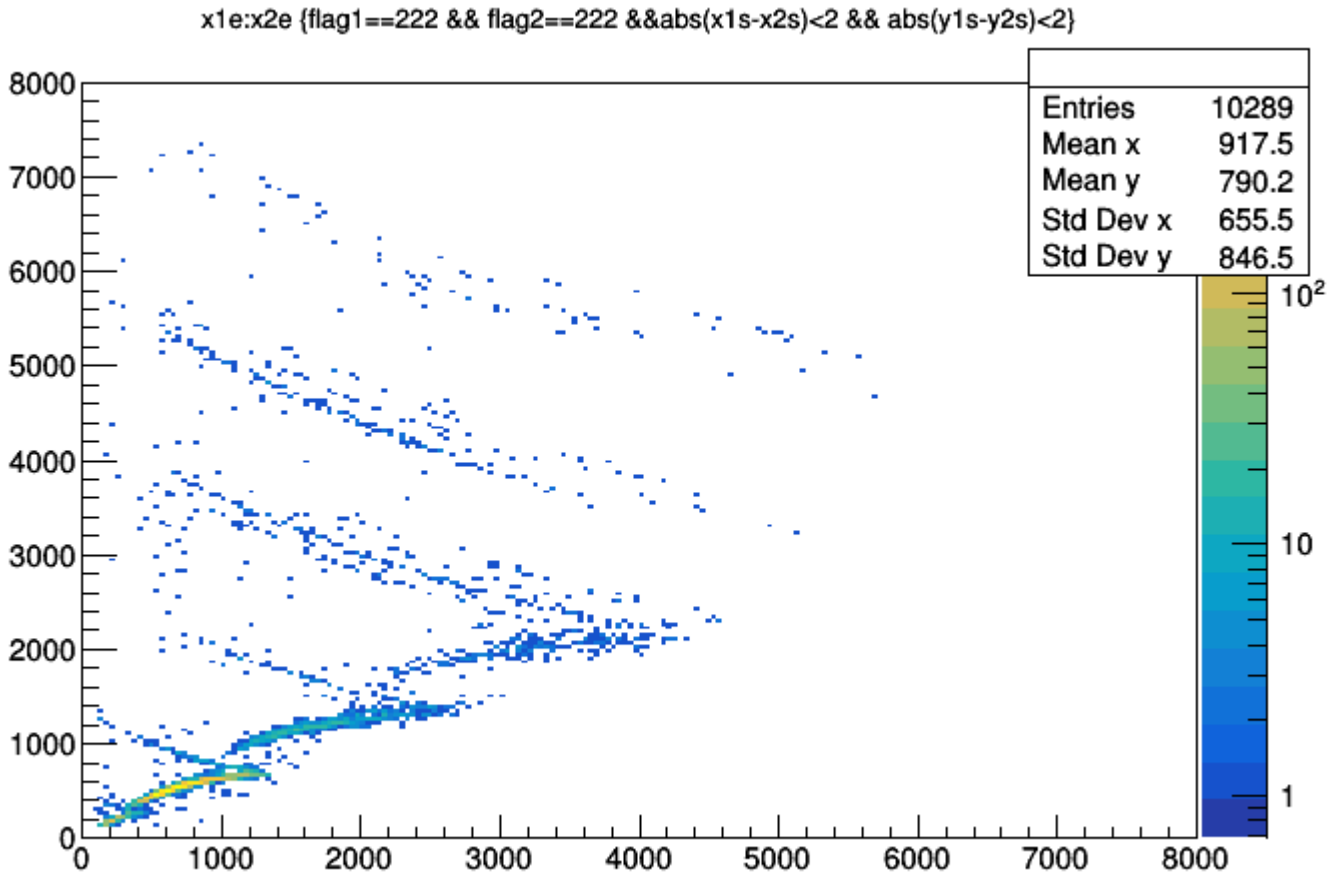
In [17]:
```
tr->Draw("x1e:x2e>>(200,0,8000,200,0,8000)","x1hit==2","colz");
c1->Draw();
```



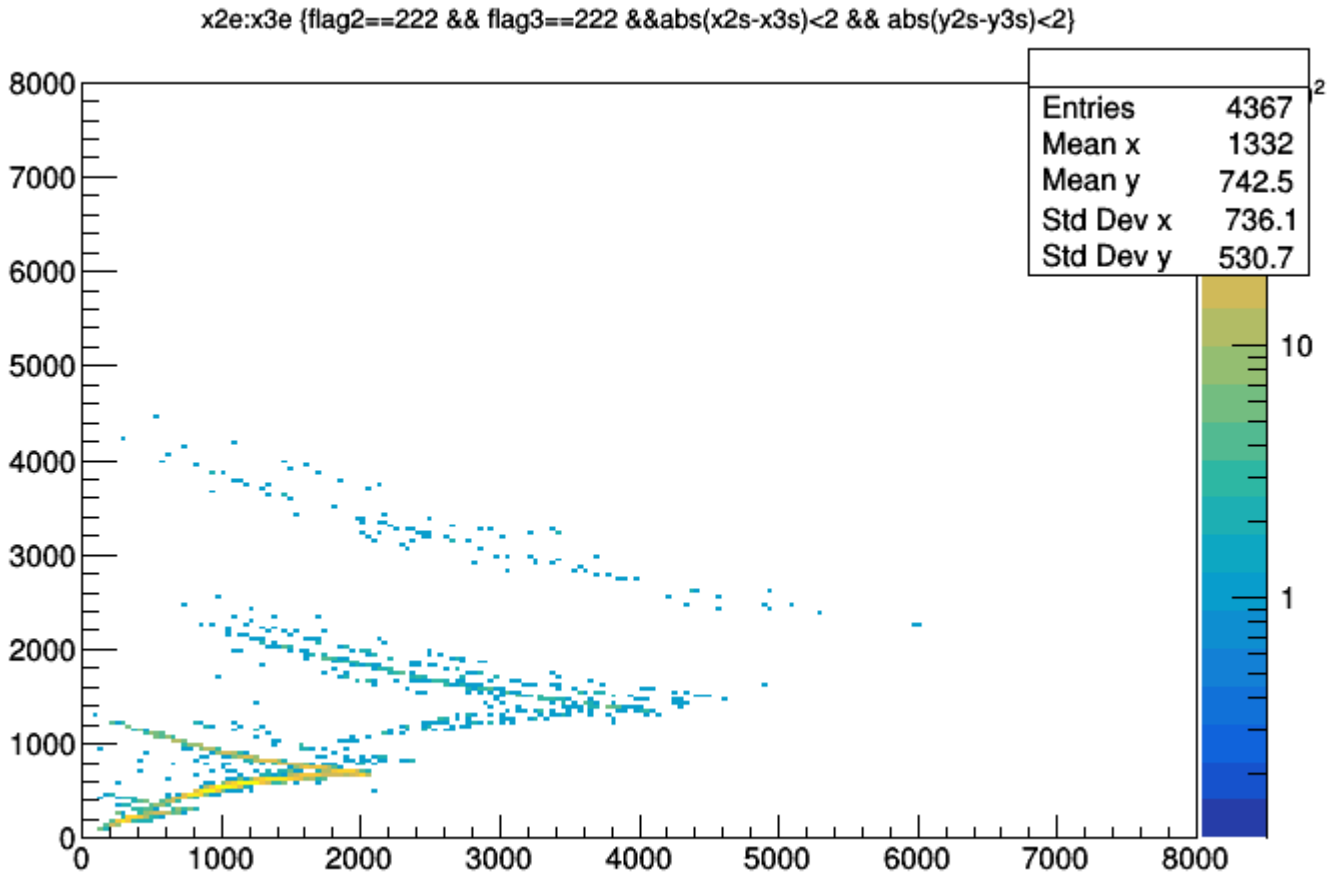
In [18]:
```
tr->Draw("x1e:x2e>>(200,0,8000,200,0,8000)","flag1==222 && flag2==222","colz");
c1->Draw();
```



In [19]:
```
tr->Draw("x1e:x2e>>(200,0,8000,200,0,8000)","flag1==222 && flag2==222 &&abs(x1s-x2s)<2 && abs(y1s-y2s)<2","colz");
c1->Draw();
```



In [20]:
```
tr->Draw("x2e:x3e>>(200,0,8000,200,0,8000)","flag2==222 && flag3==222 &&abs(x2s-x3s)<2 && abs(y2s-y3s)<2","colz");
c1->Draw();
```



遗留问题

按照单个探测器的x-y关联不能完全确定不同探测器之间的事件之间对应关系

In [21]:
```
tr->Scan("x1s:x2s:x3s:y1s:y2s:y3s","flag1==222 &&abs(x1s-x2s)>2","",2000,1);
```

```
************************************************************************************************
*    Row    * Instance *       x1s *       x2s *       x3s *       y1s *       y2s *       y3s *
************************************************************************************************
*       60 *        0 *         4 *        14 *        14 *        16 *        13 *        13 *
*       60 *        1 *        14 *         2 *         1 *        13 *        16 *        16 *
*      102 *        0 *        19 *        22 *        24 *        21 *        16 *        10 *
*      102 *        1 *        15 *        20 *        21 *        17 *        22 *        22 *
*      764 *        0 *         7 *         4 *         6 *        13 *        15 *        12 *
*      958 *        0 *        24 *        16 *        25 *        21 *        12 *        22 *
*      958 *        1 *        16 *        24 *        16 *        13 *        22 *        12 *
*     1082 *        0 *        19 *        22 *        23 *         9 *        10 *         9 *
*     1144 *        0 *        15 *        18 *        16 *        19 *        18 *        23 *
*     1334 *        1 *        11 *         8 *        10 *        21 *         9 *        25 *
*     1897 *        0 *        12 *        15 *        11 *        23 *        22 *        25 *
*     1897 *        1 *        15 *        12 *        15 *        21 *        24 *        23 *
************************************************************************************************
==> 12 selected entries
```

- 事件 60，958
  - 60： x1s-x2s错位
  - 958： x1s-x2s错位 正确匹配顺序应该为

```
//************************************************************************************************
//*    Row    * Instance *       x1s *       x2s *       x3s *       y1s *       y2s *       y3s *
//************************************************************************************************
//*       60 *        0 *        14 *        14 *        14 *        13 *        13 *        13 *
//*       60 *        1 *         4 *         2 *         1 *        16 *        16 *        16 *
//*      958 *        0 *        24 *        24 *        25 *        13 *        22 *        22 *
//*      958 *        1 *        16 *        16 *        16 *        21 *        12 *        12 *
```

- 探测器间事件匹配错位，导致de-e图上的正确关联关系外的散点。

需要进行trak matching：

按照如下条件重新进行多探测器之间事件匹配

- 探测器之间的位置不能相差太多，否则为粒子探测器上的散射或反应。
- 多个探测器位置构建的位置要指向靶的有效区域。

部分事件无法进行trak matching

In [22]:
```
tr->Scan("x1s:x2s:x3s:y1s:y2s:y3s","flag1==222 &&abs(x1s[0]-x1s[1])==1 && abs(x2s[0]-x2s[1])==1","",1000,1);
```

```
*****************************************************************************************
*    Row   * Instance *       x1s *       x2s *       x3s *       y1s *       y2s *       y3s *
*****************************************************************************************
*      277 *        0 *         7 *         7 *         7 *         8 *         7 *         7 *
*      277 *        1 *         6 *         6 *         5 *        12 *        13 *        13 *
*      620 *        0 *         9 *         8 *           *         6 *        11 *           *
*      620 *        1 *         8 *         9 *           *        11 *         5 *           *
*      747 *        0 *        10 *         9 *           *        12 *        12 *           *
*      747 *        1 *         9 *         8 *           *        11 *        10 *           *
*      980 *        0 *        17 *        17 *        17 *        13 *        13 *        13 *
*      980 *        1 *        16 *        16 *        16 *        24 *        26 *        27 *
*****************************************************************************************
==> 8 selected entries
```

- 事件277：两个事件的x1s和x2s几乎一致，无法进行track matching

最终还要通过pid matching确定匹配顺序

- 观察哪种排列方法符合正确的pid关系

In [23]:
```
!jupyter nbconvert 3.6_DSSD_data_analysis.ipynb --to html
```

```
[NbConvertApp] Converting notebook 3.6_DSSD_data_analysis.ipynb to html

[NbConvertApp] Writing 953124 bytes to 3.6_DSSD_data_analysis.html
```

In [ ]:
```

```