

TP

# Root et analyse de données

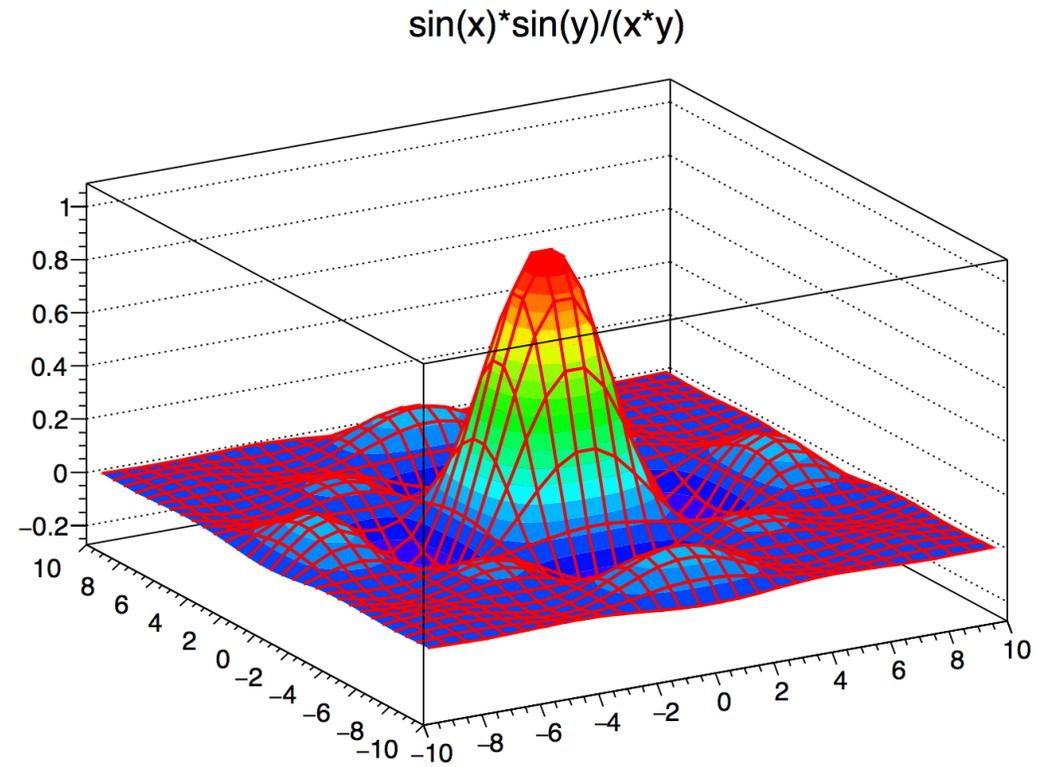
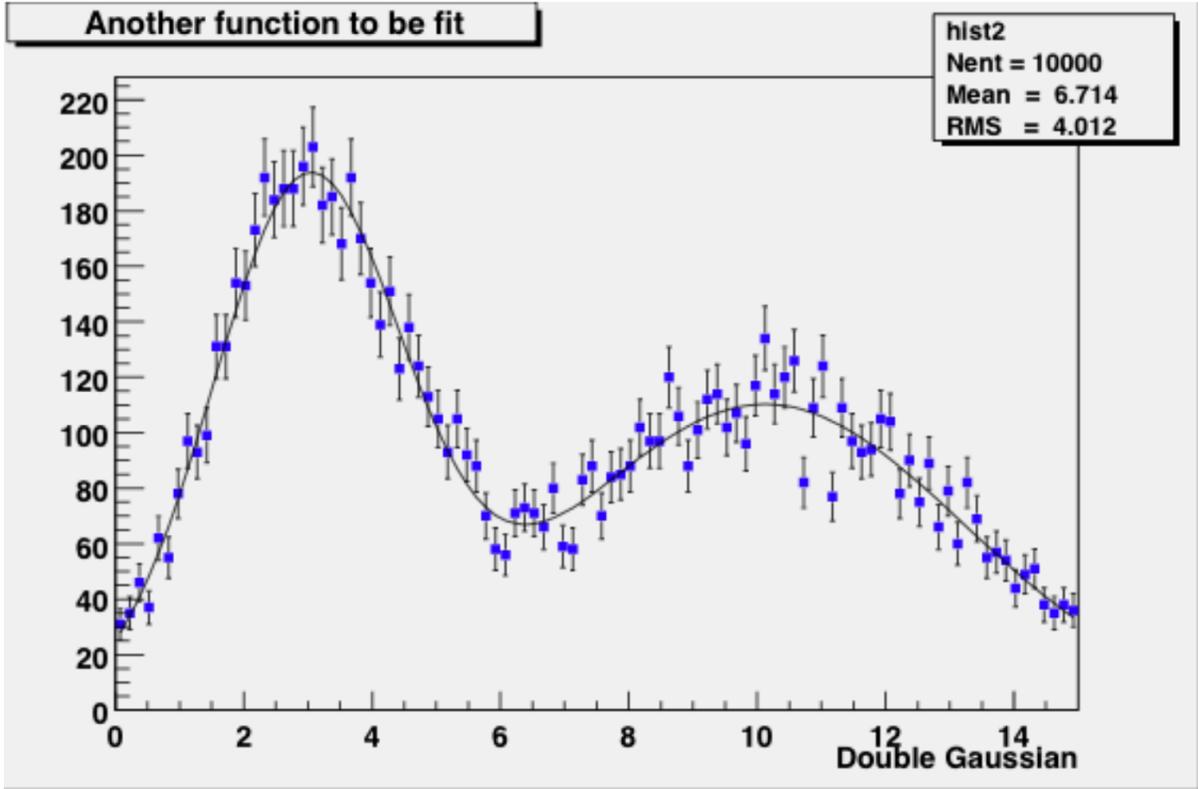
(Pré-requis : connaissance du C++)

# Plan

- TP1 – 14/11 – Initiation à ROOT
- TP2 – 21/11 – Générateur MC
- TP3 – 25/11 – Minimisation (Attention c'est un mardi)
- TP4 – 4/11 – Pseudo-expériences / limite d'exclusion

# Qu'est-ce que ROOT ?

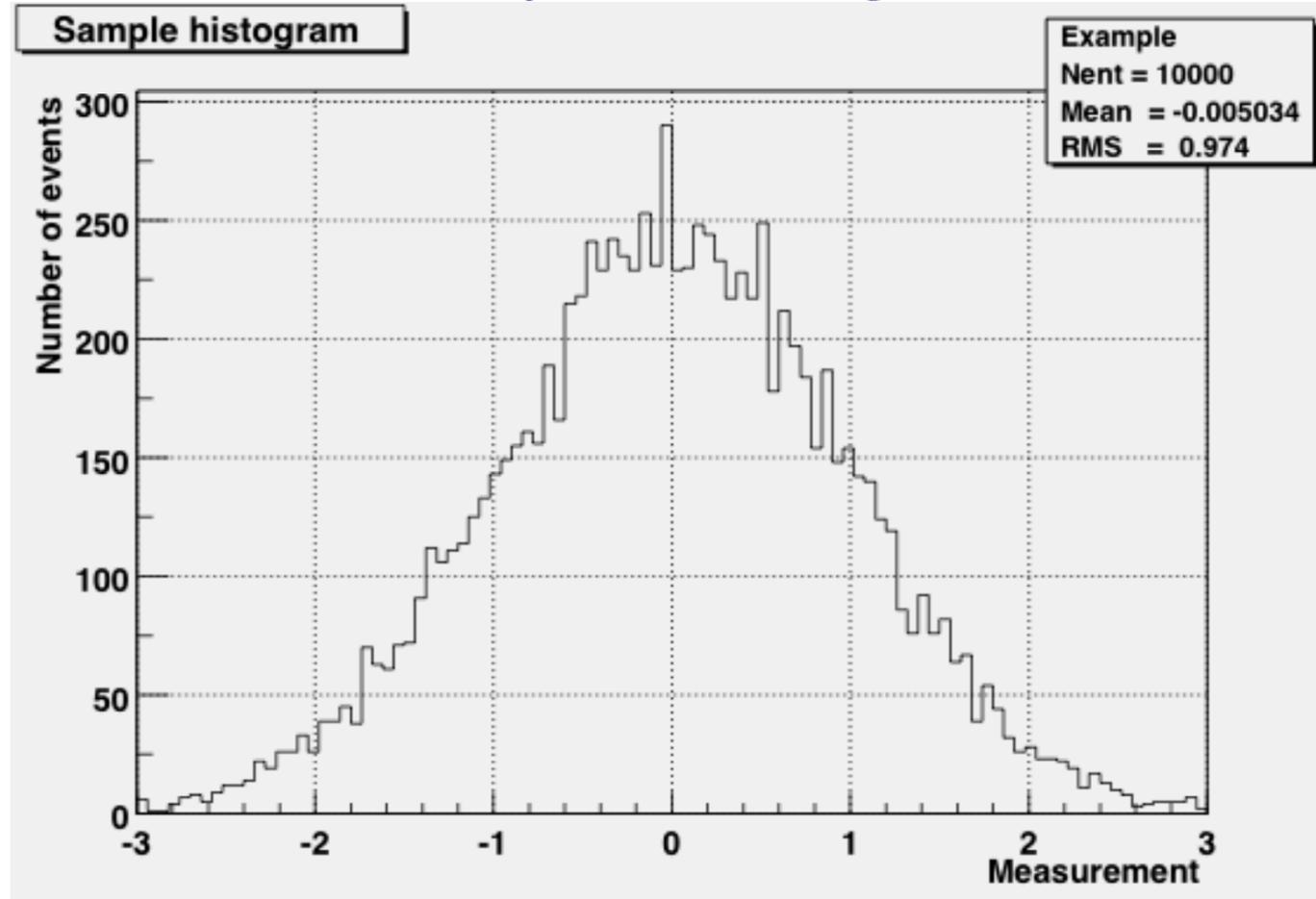
- **ROOT est ...**
- Un framework logiciel scientifique modulaires.
- Fournit toutes les fonctionnalités nécessaires traiter que grandes quantités de données, faire de l'analyse statistique, de la visualisation, et du stockage.
- Principalement écrit en C++ (mais intégré avec d'autres langages comme python et R)



Taches typiques que vous aurez à réaliser :

- Prendre des variables dans un n-tuple, réaliser des calculs et faire des histogrammes.
- Qu'est-ce que
  - un histogramme ?
  - Un n-tuple ?
  - Réaliser des calculs ?

# Anatomy of a histogram

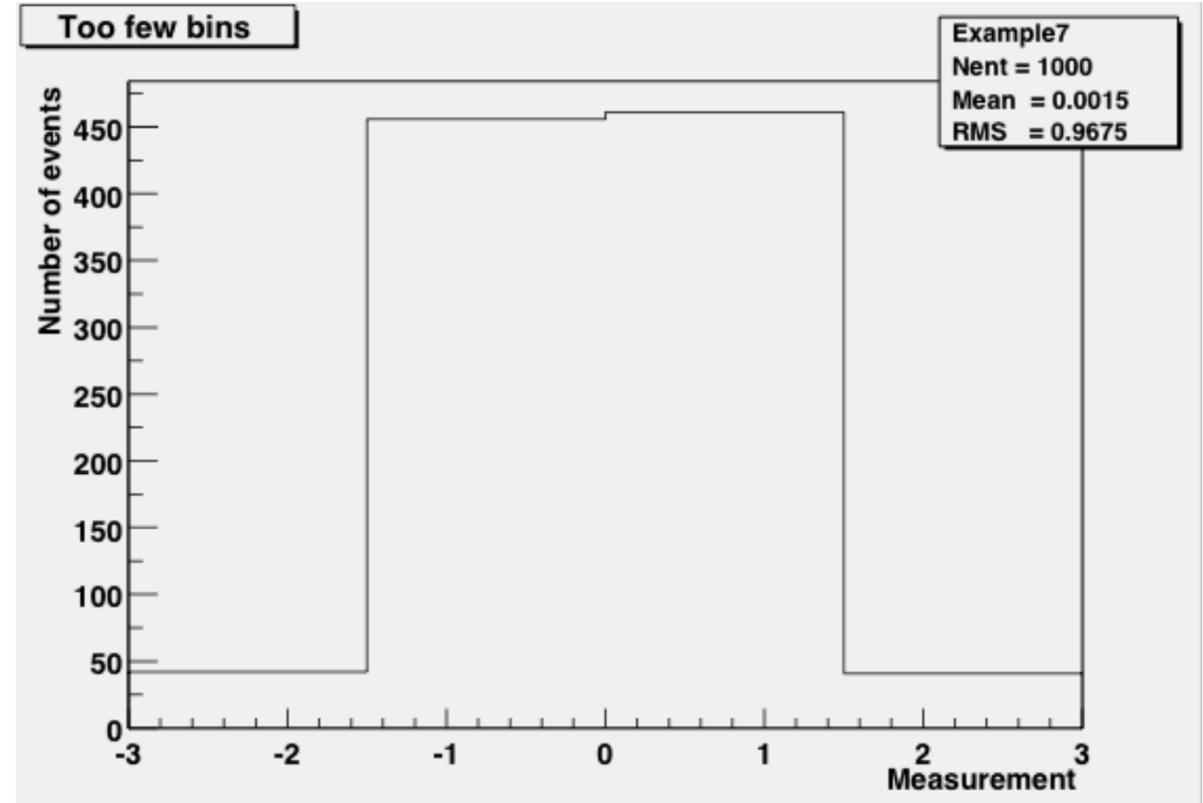
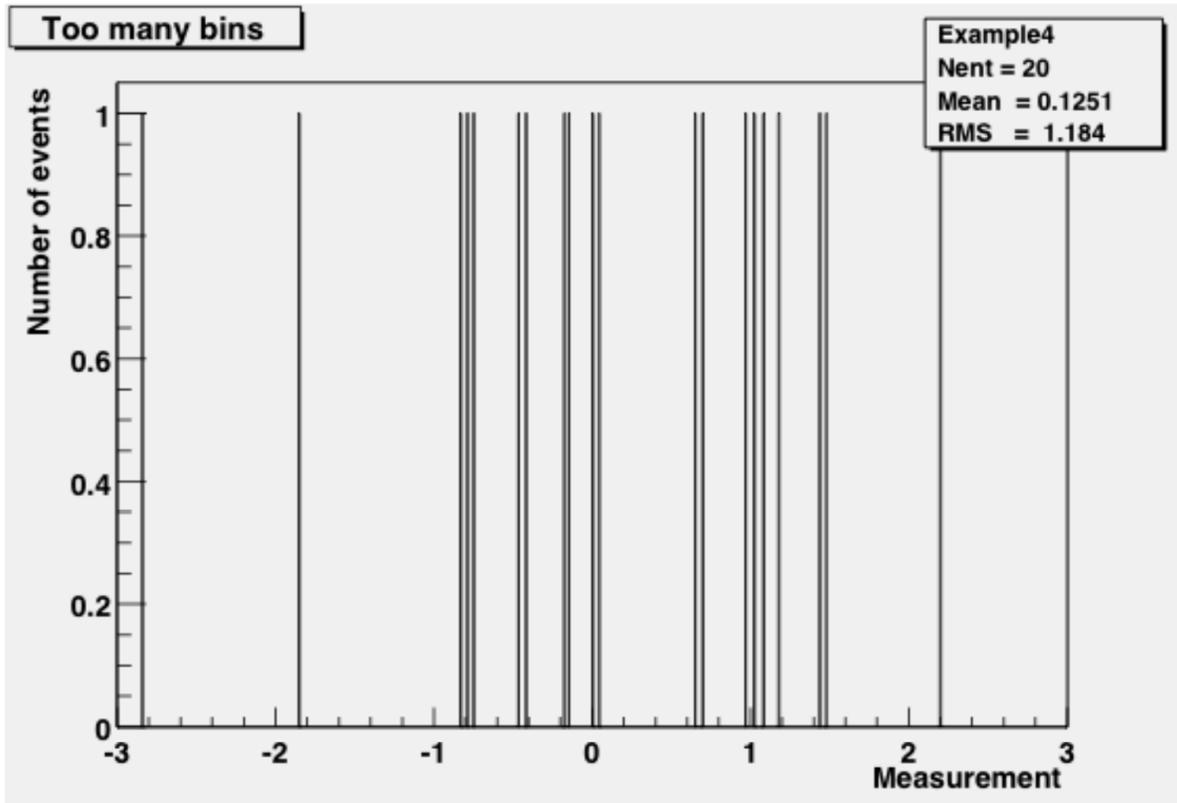


## Properties of a histogram

A ROOT command that might be used to define this histogram:

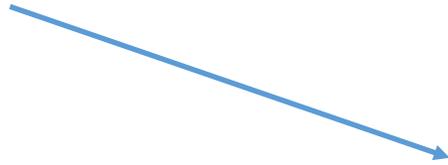
- Name or Identifier
  - Title (to be displayed on plot)
  - Number of bins
  - Lower bin limit
  - Upper bin limit
- ```
TH1F myPlot("Example", "Sample histogram", 100, -3, 3)
```

Faire un histogramme correct est un art ...



# Anatomie d'un n-tuple

- Un n-tuple est une liste ordonnée de nombres



Branches -->

Entries <--

| Row | event | ebeam  | px     | py     | pz     |
|-----|-------|--------|--------|--------|--------|
| 0   | 0     | 150.14 | 14.33  | -4.02  | 143.54 |
| 1   | 1     | 149.79 | 0.05   | -1.37  | 148.60 |
| 2   | 2     | 150.16 | 4.01   | 3.89   | 145.69 |
| 3   | 3     | 150.14 | 1.46   | 4.66   | 146.71 |
| 4   | 4     | 149.94 | -10.34 | 11.07  | 148.33 |
| 5   | 5     | 150.18 | 17.08  | -12.14 | 143.10 |
| 6   | 6     | 150.02 | 5.19   | 7.79   | 148.59 |
| 7   | 7     | 150.05 | 7.55   | -7.43  | 144.45 |
| 8   | 8     | 150.07 | 0.23   | -0.02  | 147.78 |
| 9   | 9     | 149.96 | 1.21   | 7.27   | 146.99 |
| 10  | 10    | 149.92 | 5.35   | 3.98   | 140.70 |
| 11  | 11    | 149.88 | -4.63  | -0.08  | 147.91 |

- Généralisation : un arbre ROOT (TTree) peut être une liste ordonnée de n'importe quelle collection d'objets C++.

# Pourquoi ROOT ?

- Capable de manipuler des n-tuples et des histogrammes (et des 4-vecteurs et la persistance des objets et la géométrie des détecteurs et les diagrammes de Feynman and l'algèbre linéaire and l'ajustement des fonction et les méthodes d'analyse multivariées et ...)
- Peut gérer de très grand volumes de données (des millions d'événement de physique, des fichiers jusqu'à une taille de l'ordre du TByte, processing batch et multiprocesseur) •
- Multi-plateformes (Windows, Mac, beaucoup de type d'UNIX)
- C'est du logiciel libre (open source)
- MAIS ...
  - Complicé par l'historique de sa construction
  - Interface utilisateur et documentation souvent négligées
  - Ce n'est pas une « app » au sens produit fini du terme. ROOT n'est pas facile à utiliser
  - Connaissance du C++ impérative, notamment pour faire les calculs

- Le premier TP consiste en une prise en main de root sous la forme d'un tutoriel
- Extrait d'un (excellent) tutoriel créé par M Seligman :  
<http://www.nevis.columbia.edu/~seligman/root-class/>
- Pour commencer :  
<http://www.ipnl.in2p3.fr/cours/perries/M2suba/listing.php>
- Indispensable : <https://root.cern.ch/>