

Additional tools - TStatToolkit

# Why TStatToolkit ?

- Not all of the desired functionality is implemented in ROOT
  - A lot of (non trivial) statistical code repeated in the user (e.g TPC calibration <-> TRD calibration, comparisons,tracking) code
- TStatToolkit
  - The place to hold simple common algorithms
  - Algorithms implemented as static functions (as in TMath)
- Where to find it
  - `$ALICE_ROOT/STAT/TStatToolkit.h`

# Algorithms on arrays

- Truncated mean
- LTM -Least Trimmed Mean (EvaluateUni)
- LTM with external information (EvaluateUniExternal)
- Frequency – Calculates occurrence of given integers (Freq)

```
static void EvaluateUni(Int_t nVectors, Double_t *data, Double_t &mean,  
                        Double_t &sigma, Int_t hh);
```

```
static void EvaluateUniExternal(Int_t nVectors, Double_t *data, Double_t &mean,  
                                Double_t &sigma, Int_t hh, Float_t externalfactor=1);
```

```
static Int_t Freq(Int_t n, const Int_t *inlist, Int_t *outlist, Bool_t down);
```

```
static Double_t FitGaus(Float_t *arr, Int_t nBins, Float_t xMin, Float_t xMax,  
                        TVectorD *param=0, TMatrixD *matrix=0, Bool_t verbose=kFALSE);
```

# Algorithms on histograms

```
static void TruncatedMean(TH1F * his, TVectorD *param, Float_t down=0, Float_t up=1.0,  
                           Bool_t verbose=kFALSE);
```

```
static void LTM(TH1F * his, TVectorD *param=0 , Float_t fraction=1, Bool_t verbose=kFALSE);
```

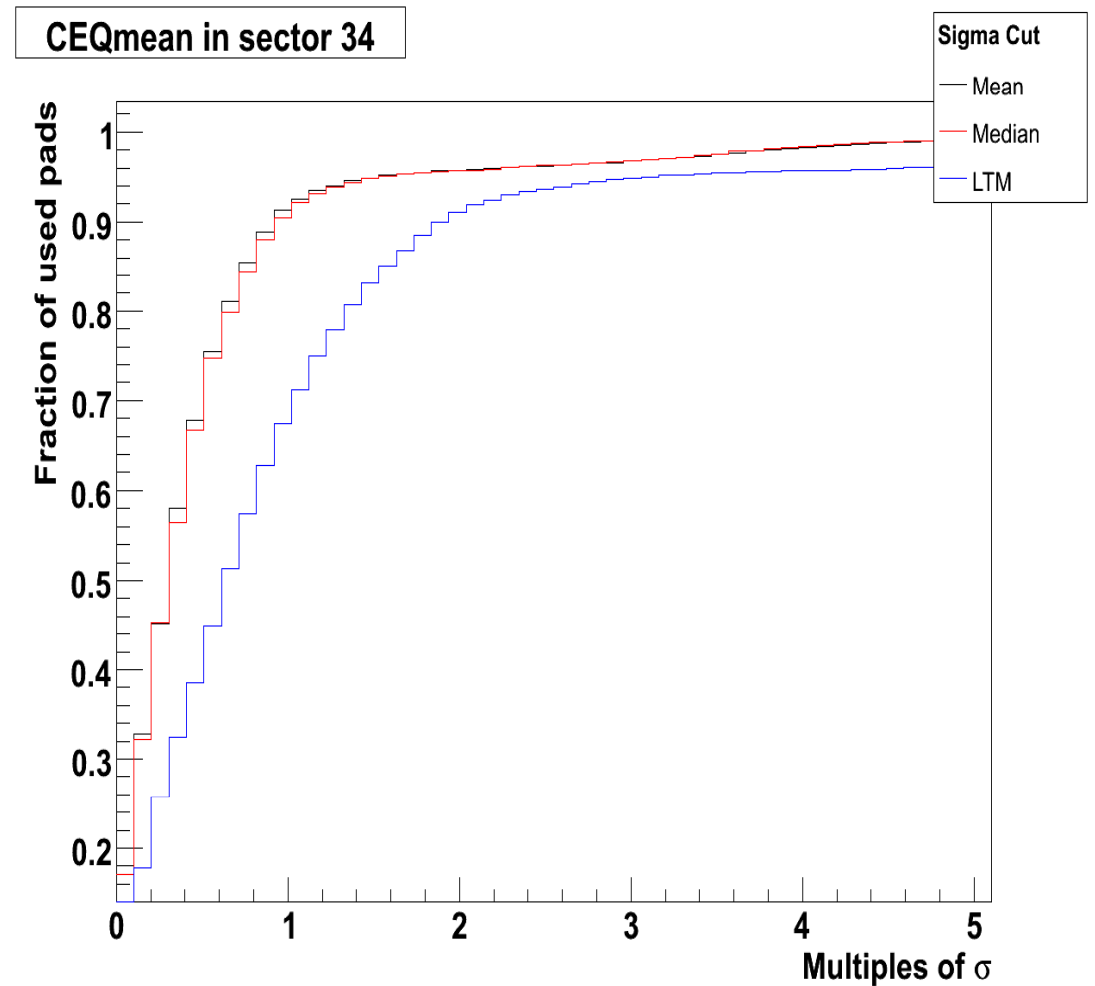
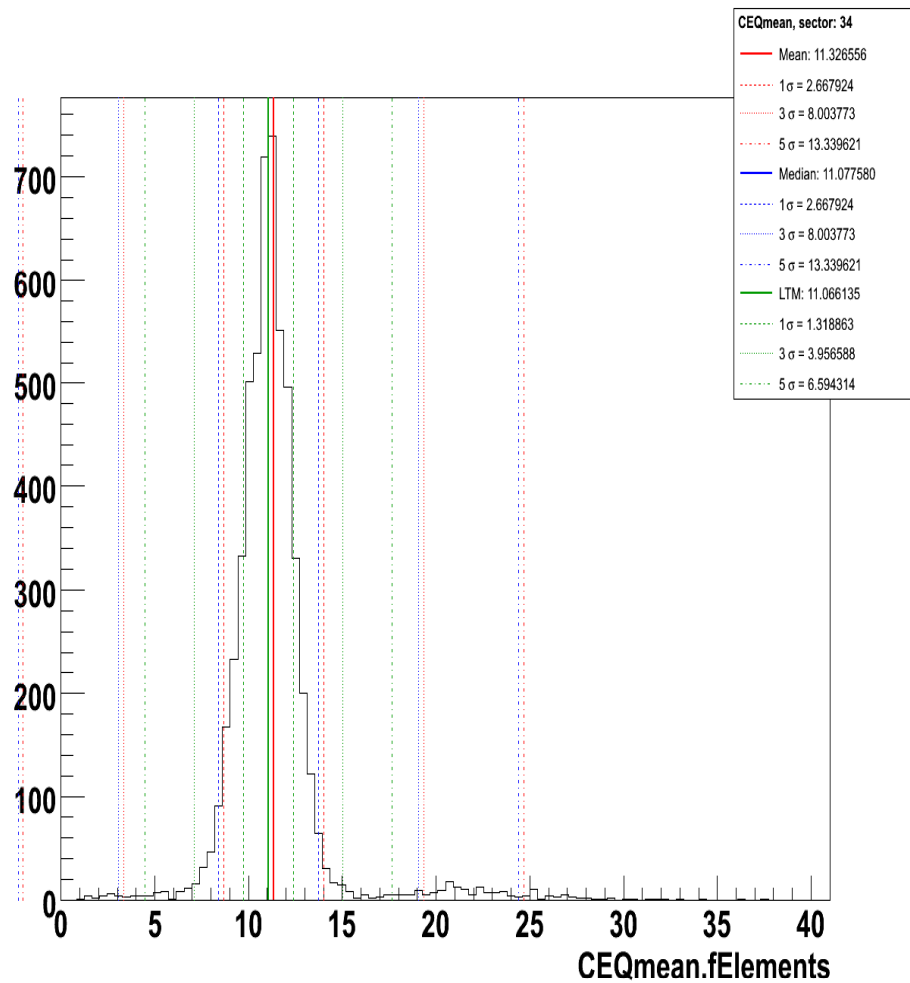
```
static Double_t FitGaus(TH1F* his, TVectorD *param=0, TMatrixD *matrix=0, Float_t xmin=0,  
                          Float_t xmax=0, Bool_t verbose=kFALSE);
```

```
static TGraph2D * MakeStat2D(TH3 * his, Int_t delta0, Int_t delta1, Int_t type);
```

```
static TGraph * MakeStat1D(TH3 * his, Int_t delta1, Int_t type);
```

# Histogram example

## Mean, Median, LTM



# TTree – TLinearFitter interface

```
static TString* FitPlane(TTree * tree, const char* drawCommand,  
                          const char* formula, const char* cuts,  
                          Double_t & chi2, Int_t & npoints,  
                          TVectorD & fitParam, TMatrixD & covMatrix,  
                          Int_t start=0, Int_t stop=10000000);
```

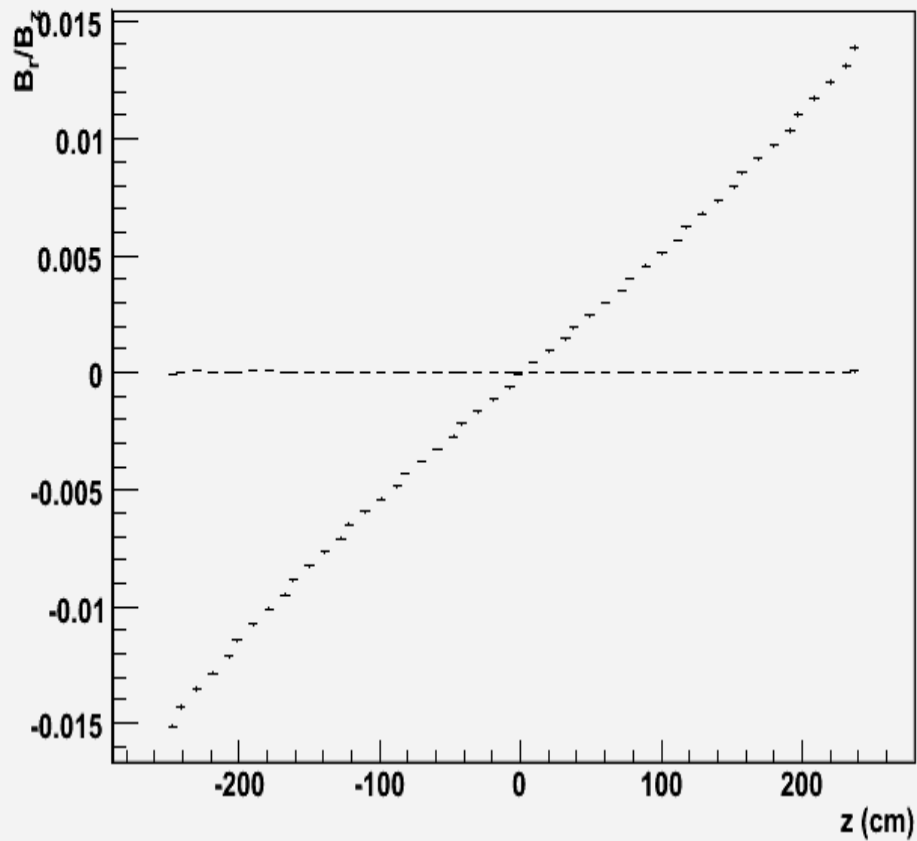
```
tree->Draw("br:z",ctpc);  
  
TString *str0 =  
 toolkit.FitPlane(tree,"br","zm++zm^2++zm^3++rm++rm^2++rm^3  
 ++rm*zm++rm*cfi++zm^2*cfi++rm*zm^2*cfi++rm*sfi++zm^2*sfi  
 ++rm*zm^2*sfi++zm^2*sfi2++zm^3*sfi++zm^3*cfi++zm^3*sfi2++z  
 m^3*cfi2",ctpc,chi2,npoints,param,covar);  
  
tree->SetAlias("br0",str0.Data());  
  
tree->Draw("br-br0:z",ctpc);
```

Example: fitting of the magnetic field  
using polynoms  
and visualization of the fit quality results

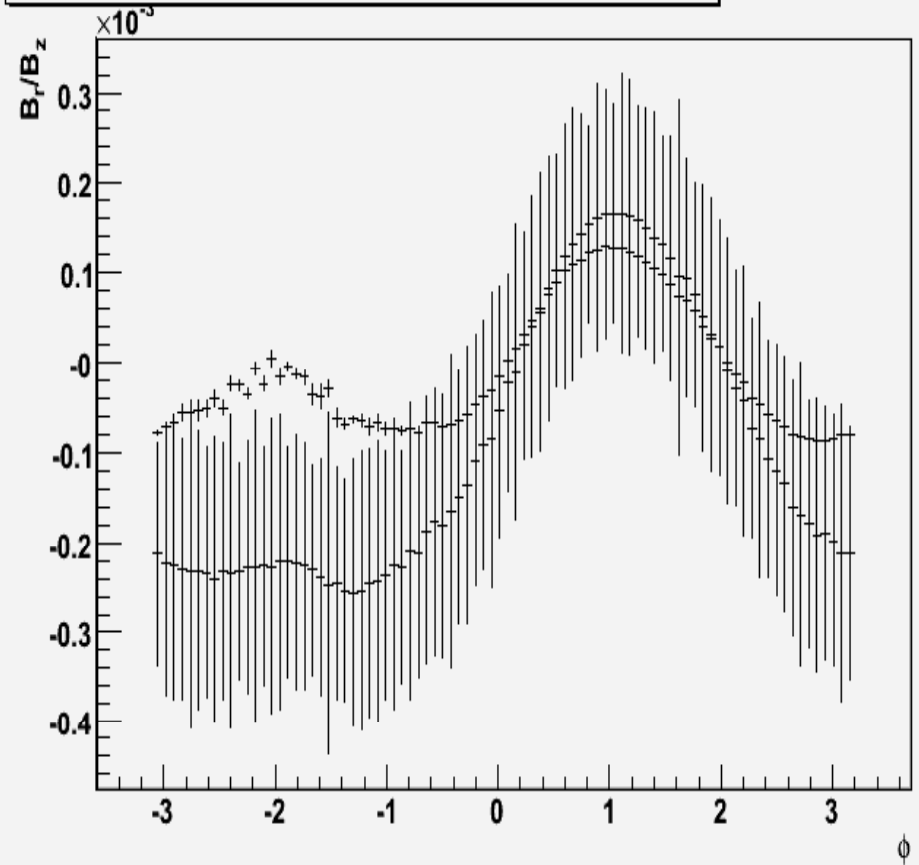
VERY IMPORTANT FOR FAST  
PROTOTYPING

# Magnetic field fitting example

br:z {r>90&&r<250}



br:fi {(abs(z)<60&&abs(r)<150)&&(r>90&&r<250)}



# Conclusion

- A lot of (non trivial) statistical code repeated in the user (e.g TPC calibration <-> TRD calibration, comparisons,tracking) code
- TStatToolkit
  - The place to hold simple common algorithms
  - \$ALICE\_ROOT/STAT directory
- The number of (common) algorithms will increase in time
  - Some of them can move to the Root framework
  - Tries to keep minimal dependencies
    - base, hist, graf, minuit, tree