Validation of Electromagnetic Physics Models for Parallel Computing Architectures in the GeantV project

Marilena Bandieramonte
(marilena.bandieramonte@cern.ch)
for the GeantV team

13th Oct 2016, San Francisco

Overview

*See Plenary Session 3 "Simulation - Key Issues for the Coming Decade" by Federico Carminati

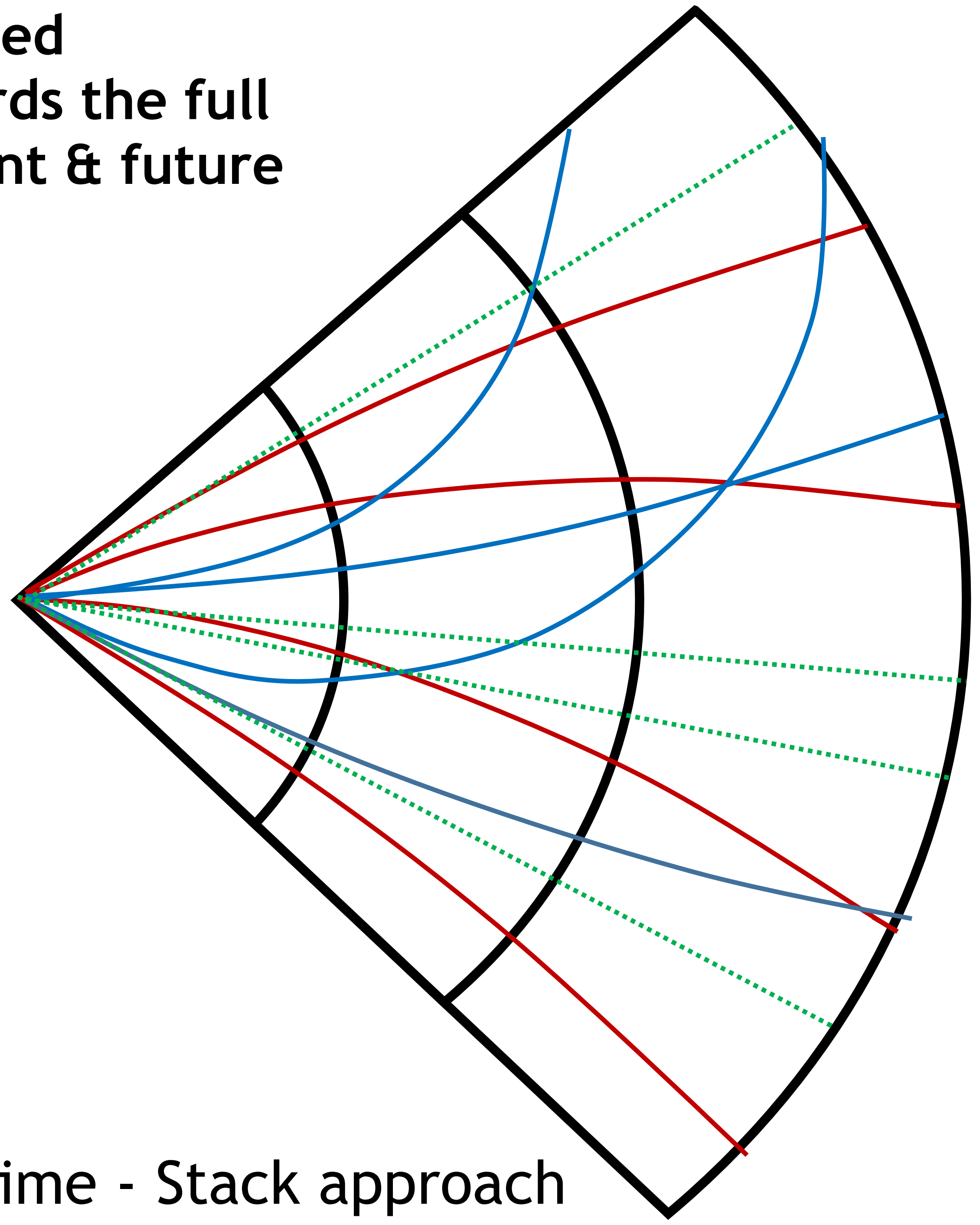
- **GeantV project: the future generation simulation software***
- **Motivation and goals for the Vectorized Physics library (VecPhys)**
    - Write EM physics models dealing with multiple tracks
      - accurate, fast and portable
    - Exploit both ***SIMD*** (vector pipeline) and ***SIMT*** (accelerators) execution models
- **Statistical Verification suite for sampling for GeantV physics models**
    - Alternative sampling techniques:
        - Alias sampling method and improvements
        - Shuffling rejection method
        - Hybrid sampling techniques
    - Investigate/try code improvements & alternative algorithm
    - Measured performances on CPU/GPU**
    - Automation of the verification tools
- **Final considerations and ongoing work**

**See Oral "Computing Performance of GeantV Physics Models" by Soon Y. Jun!

GeantV – Adapting simulation to modern hw

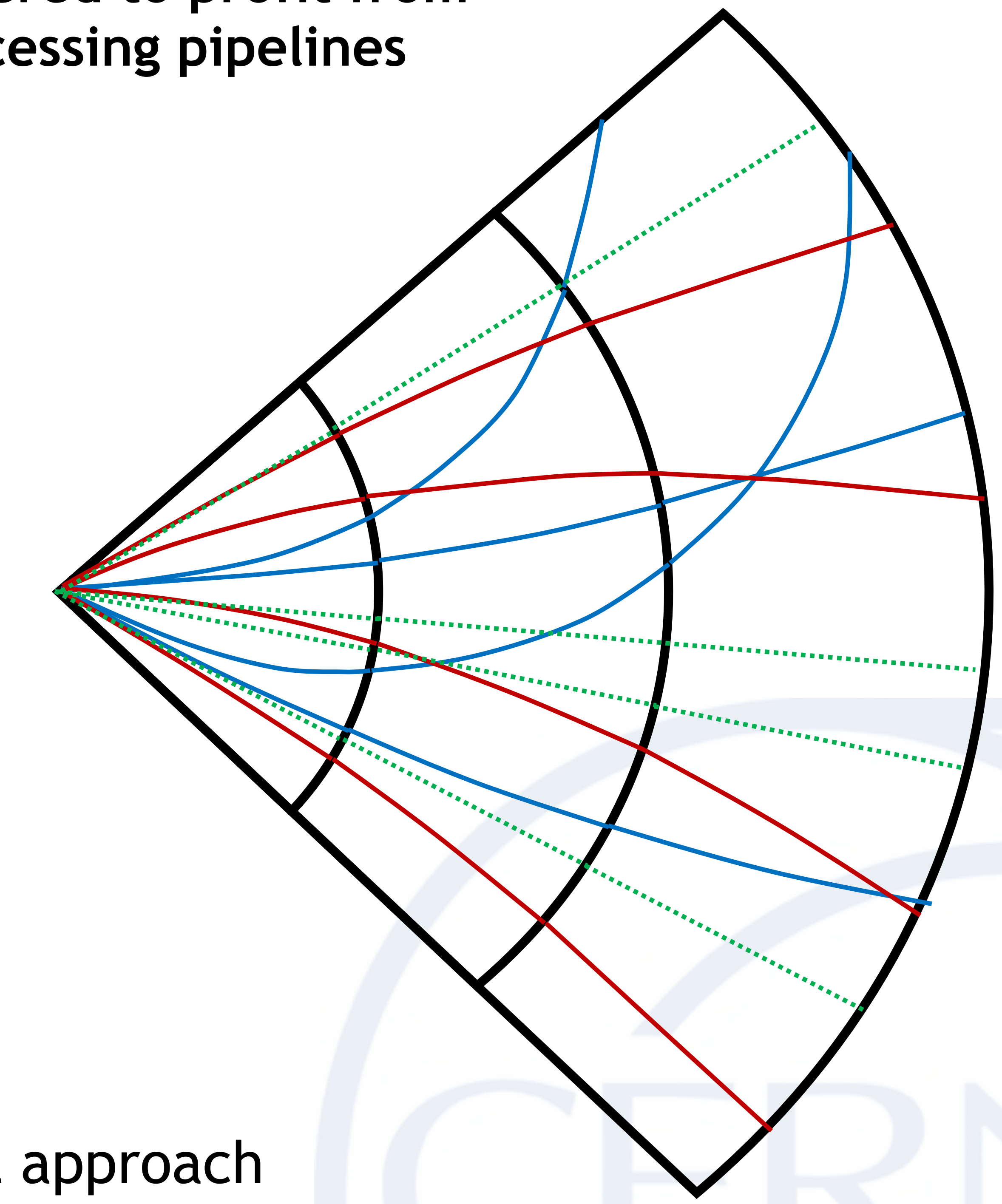
*See Plenary Session 3 "Simulation - Key Issues for the Coming Decade" by Federico Carminati

Classical simulation (G3, G4 and others)
Flexible, but limited adaptability towards the full potential of current & future hardware



- One track at a time - Stack approach
- Single event transport
- Embarrassingly parallel
- Cache coherency – low
- Vectorization – low (scalar auto-vectorization)

GeantV simulation
Engineered to profit from all processing pipelines

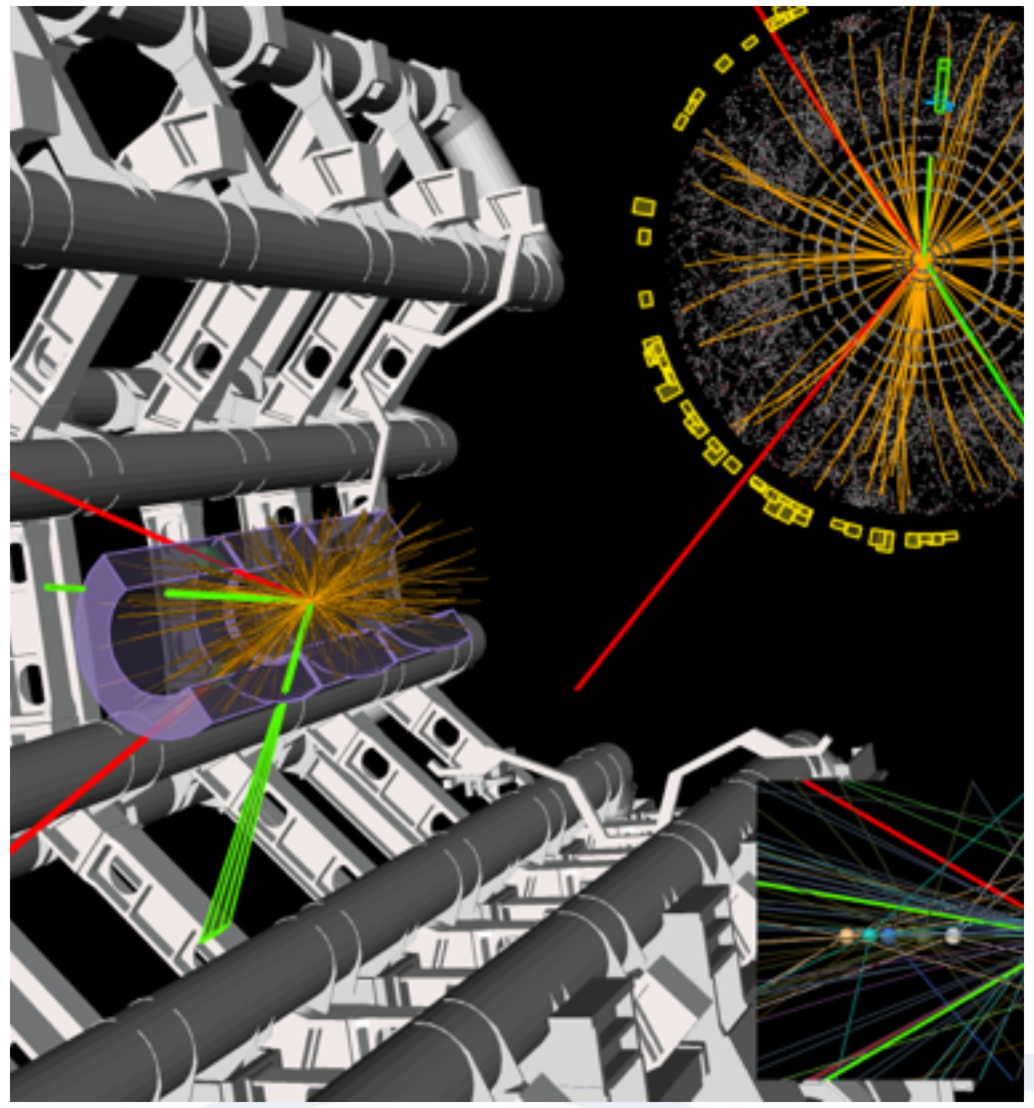


- Basket approach
- Multi event transport
- Fine-grain parallelism + threads
- Cache coherency – high
- Vectorization – high (explicit multi-particle interfaces)

credit A. Gheata

Motivation and goals

- ***80% of CPU*** time is spent on electrons and photons in Geant4 for typical collider experiments. About 30-40% is on Electromagnetic (EM) processes
    - Write EM physics models dealing with multiple tracks
        - accurate
        - fast
        - portable
    - Exploit both ***SIMD*** (vector pipeline) and ***SIMT*** (accelerators) execution models
    - Have a ***common source code**** between scalar, vector and accelerator (GPU, Xeon Phi)
- Two approaches followed for the Physics:
    - Focus on vectorisation:
        - Explore alternative **sampling techniques**
        - Validate new vectorized physics models
    - Focus on physics models themselves
        - New Bremsstrahlung and Ionization models (both useful also for Geant4)

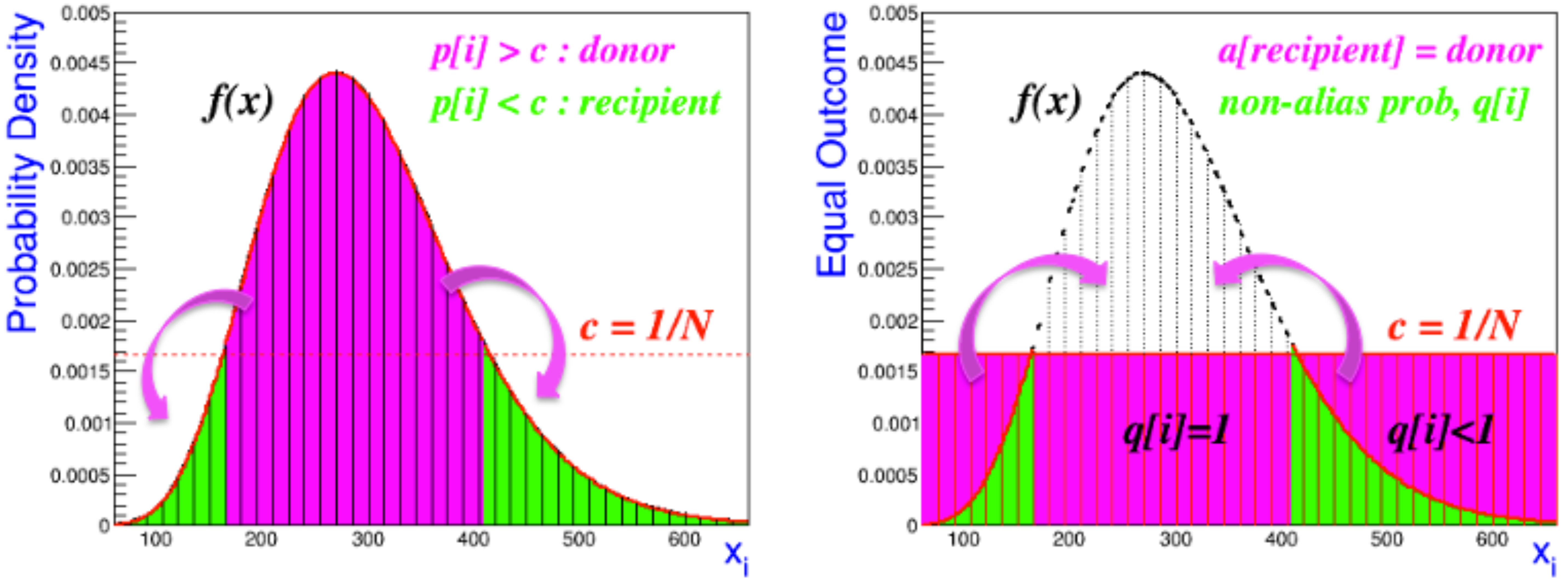


*See Oral "Accelerating Navigation in the VecGeom Geometry Modeller" by Sandro C. Wenzel

Alias Sampling Method for N Discrete Outcomes

- Recast a N-discrete p.d.f to N equal probable events, each with likelihood 1/N = c *(A. J. Walker, 1974)* – effectively vectorizable

- Reproduce the original distribution by one trial sampling:




```
Alias index: a[recipient] = donor
Non-alias probability: q[i]
For any random (xi), accept if rand(0,1) < q[i] or take the alias
```

credit S. Y. Jun

Statistical Validation suite

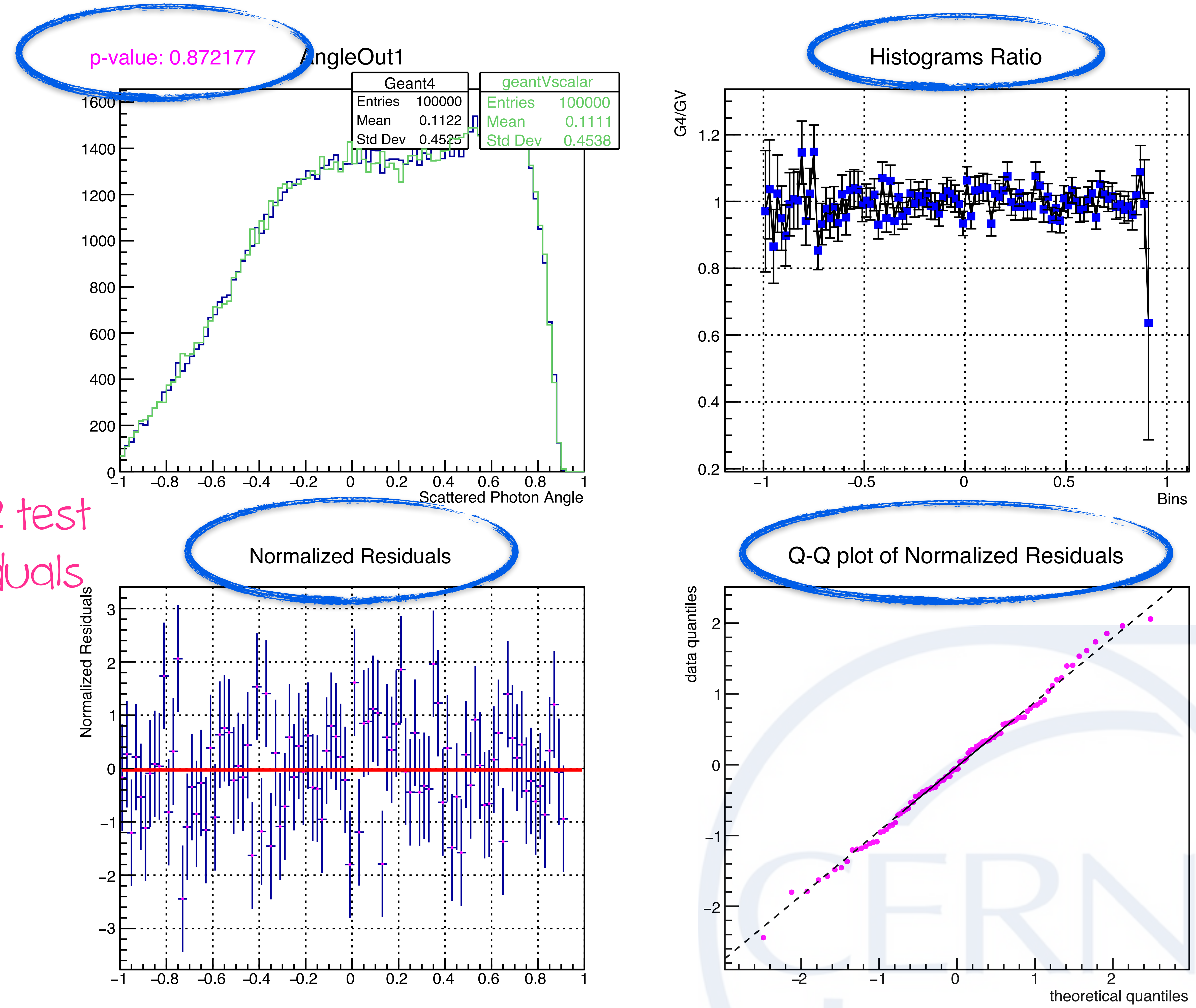
- Pearson χ^2-test for comparing weighted and unweighted histograms

- Analysis of residuals

- QQ-Plots

chi^2 test

residuals



G4/GV ratio

QQ-Plot for normalized res

EM physics processes under development

Primary	Process	Model	Secondaries	Survivor
γ	Compton Scattering	Klein-Nishina	e^-	γ
	Pair-Production	Bethe-Heitler	e^- e^+	-
	Photo-Electric Effect	Sauter-Gavrila	e^-	-
e^-	Ionization	Moller-Bhabha	e^-	e^-
	Bremsstrahlung	Seltzer-Berger	γ	e^-
	Multiple Scattering	Goudsmit-Saunderson	-	e^-

Gamma processes

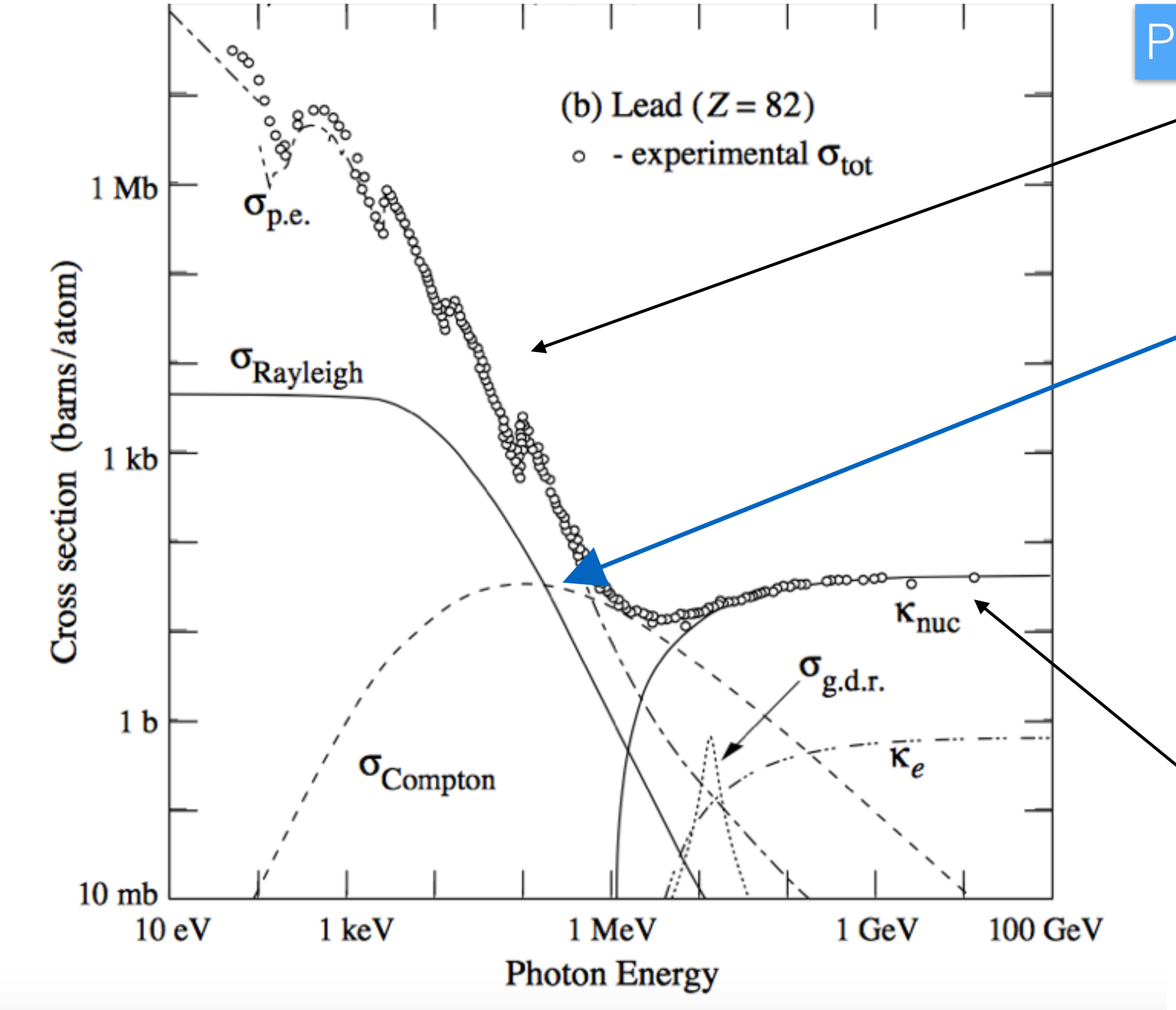


Photo-electric effect* — SauterGavrila

Compton** — KleinNishina

Pair production*** — Bethe-Heitler

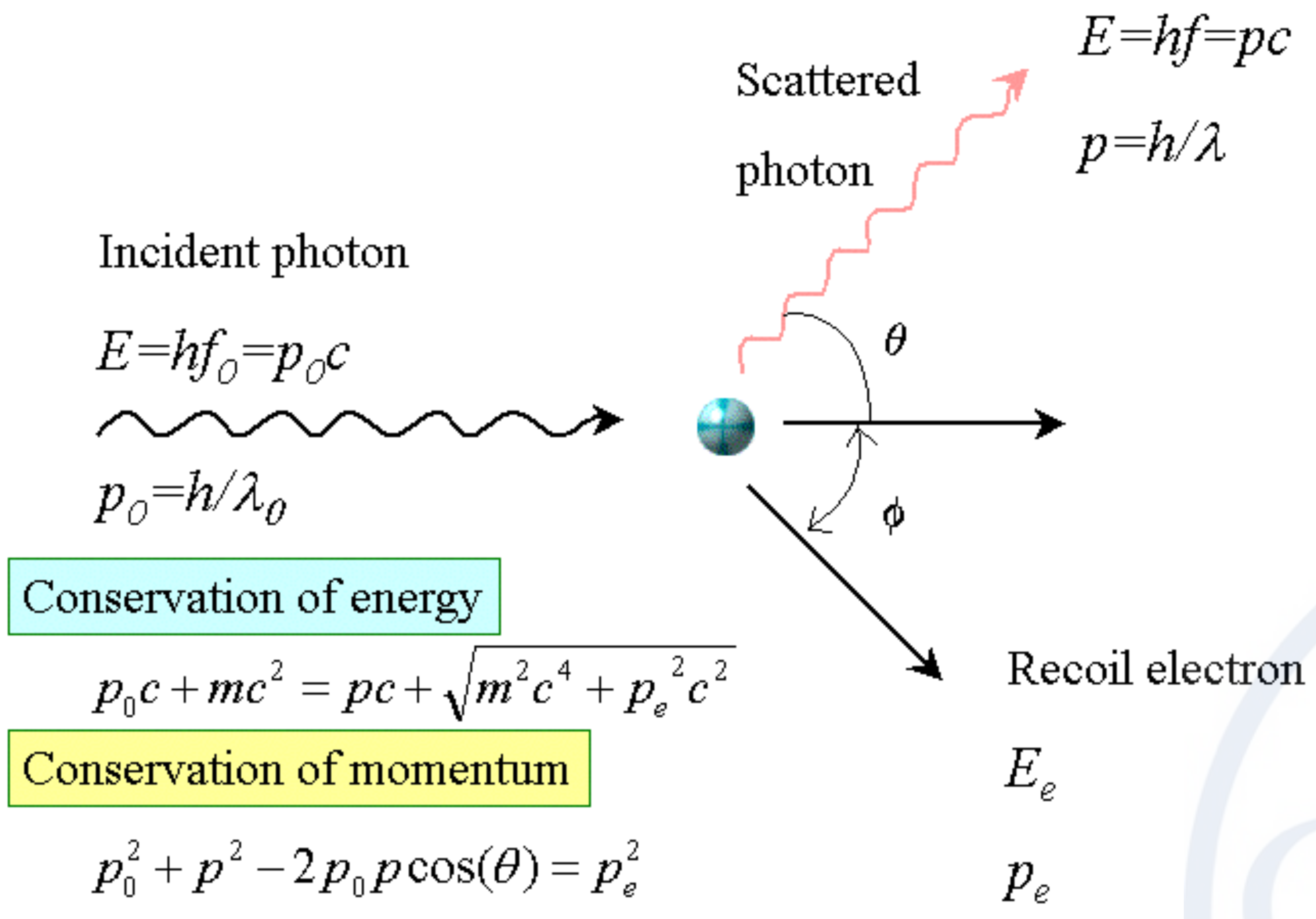
* Photo-electric should not be validated with Geant4 in the energy range, (E/m_electron) < 50 ~ 25 MeV (above this threshold, Geant4 returns cos(theta) =1 as a good approximation)

** Compton should be validated for E > 10KeV (to be consistent with Geant4)

*** Pair production: should be validated in the energy range, E > 2*m_electron ~ 1 MeV

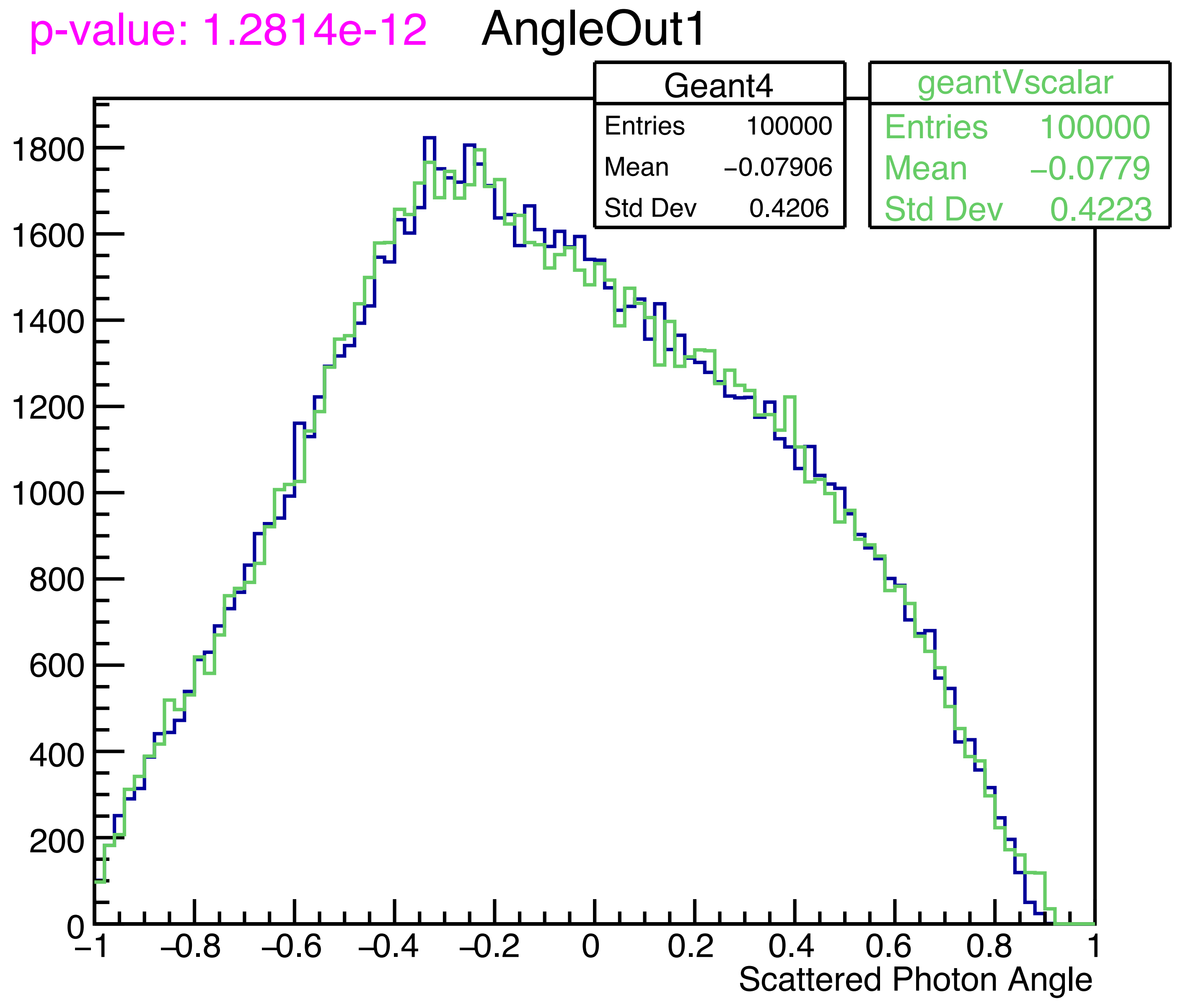
Klein-Nishina Validation

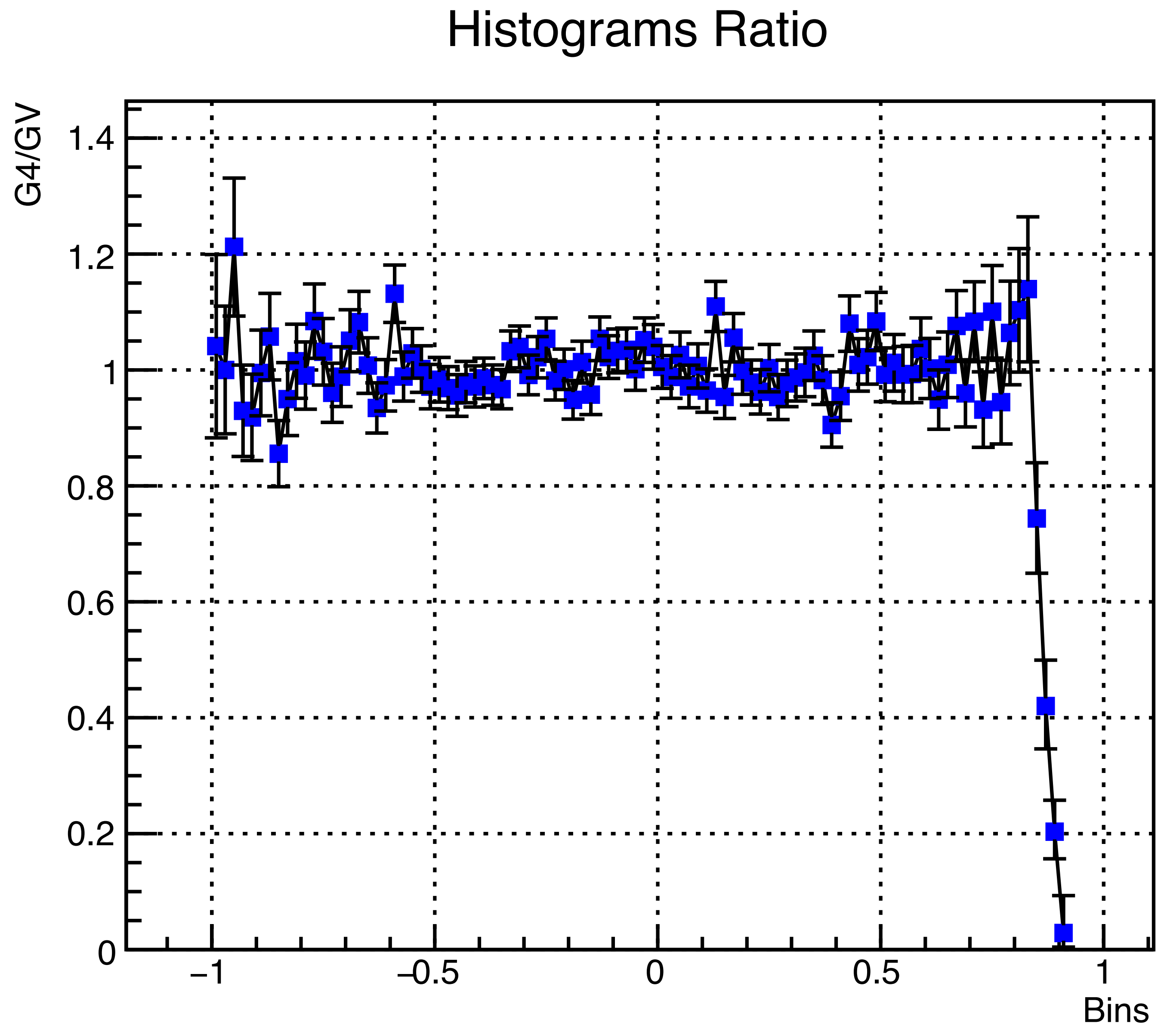
Compton Scattering

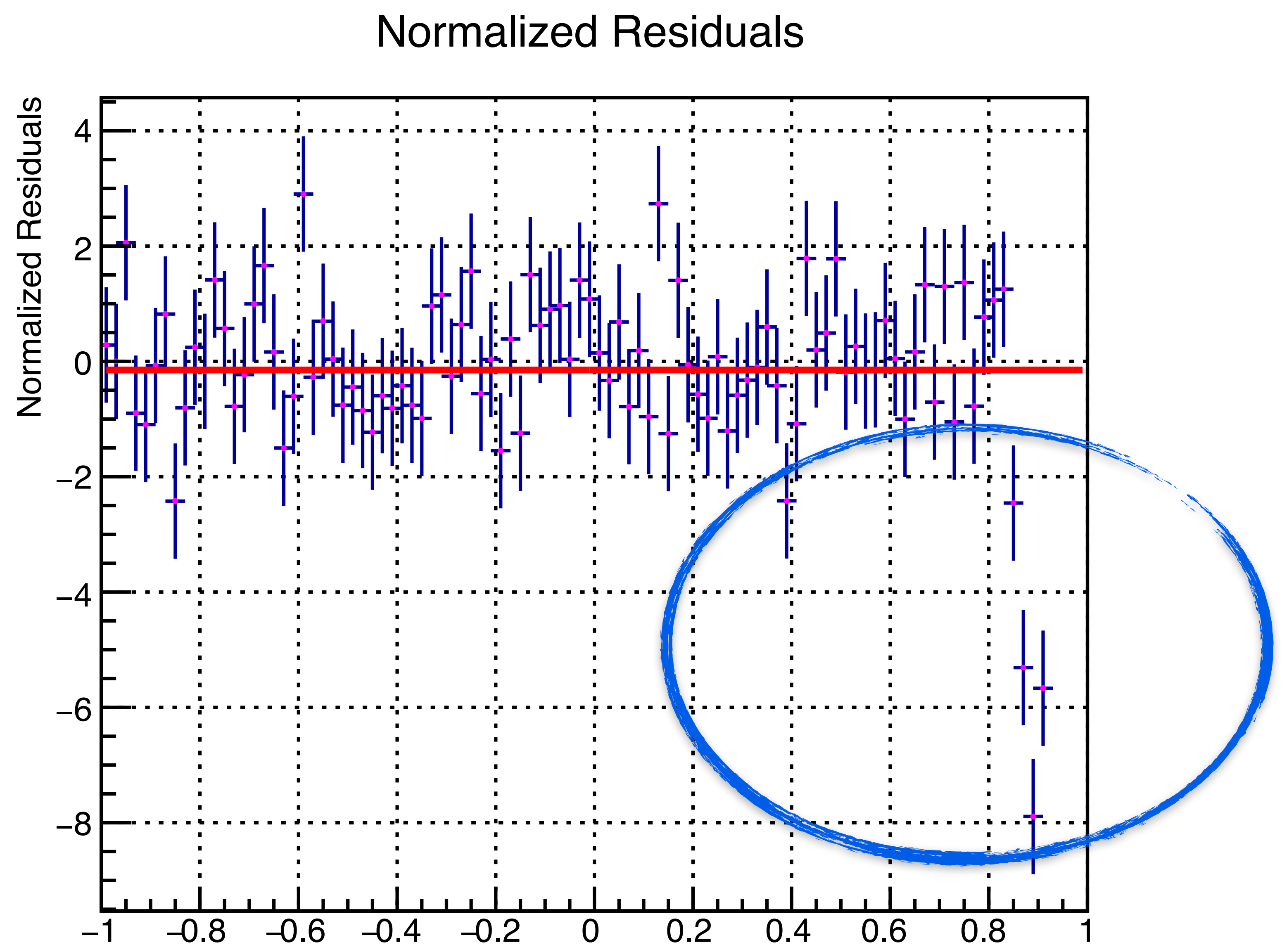


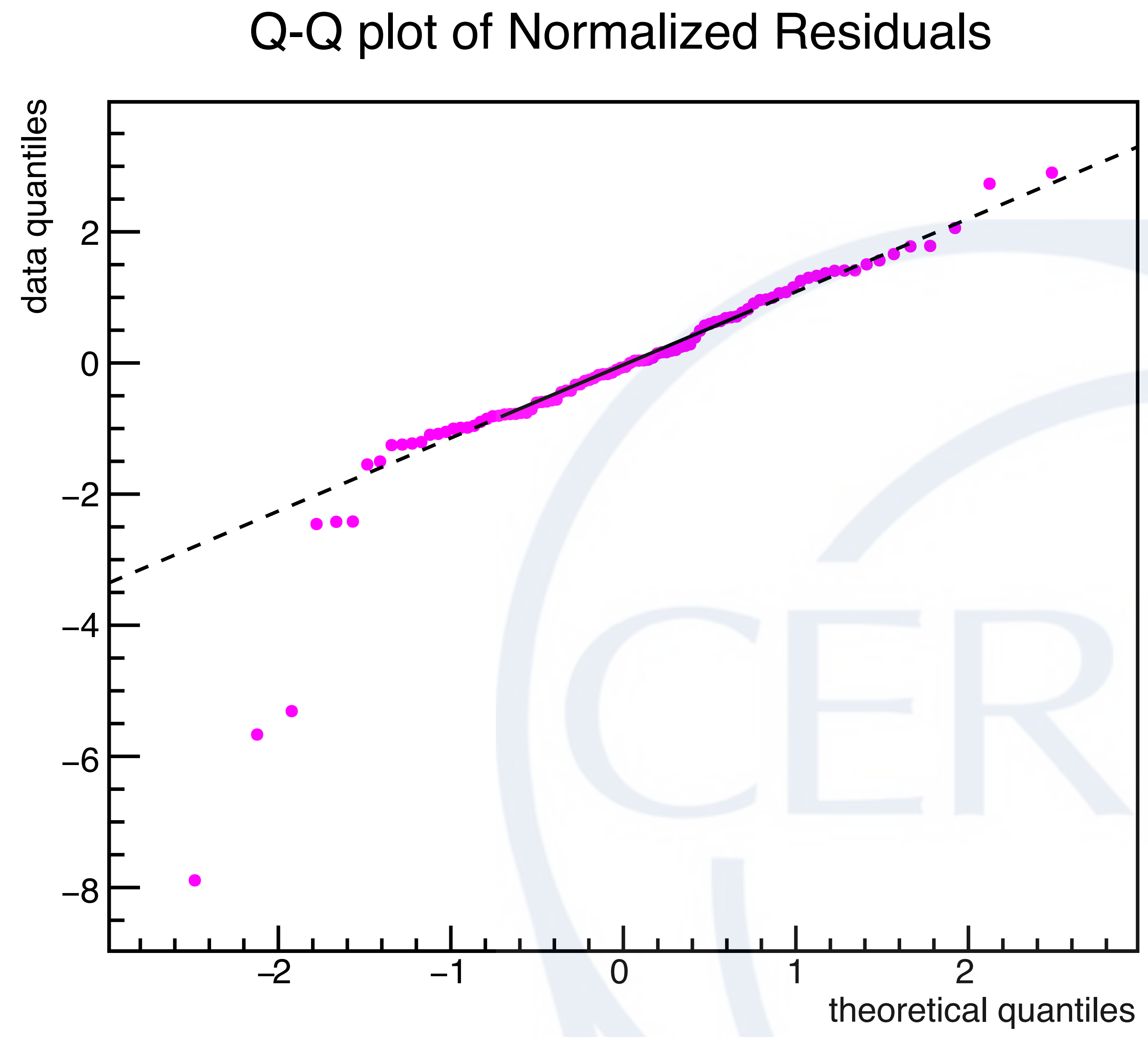
KleinNishina Pearson chi2Test: ***first results***

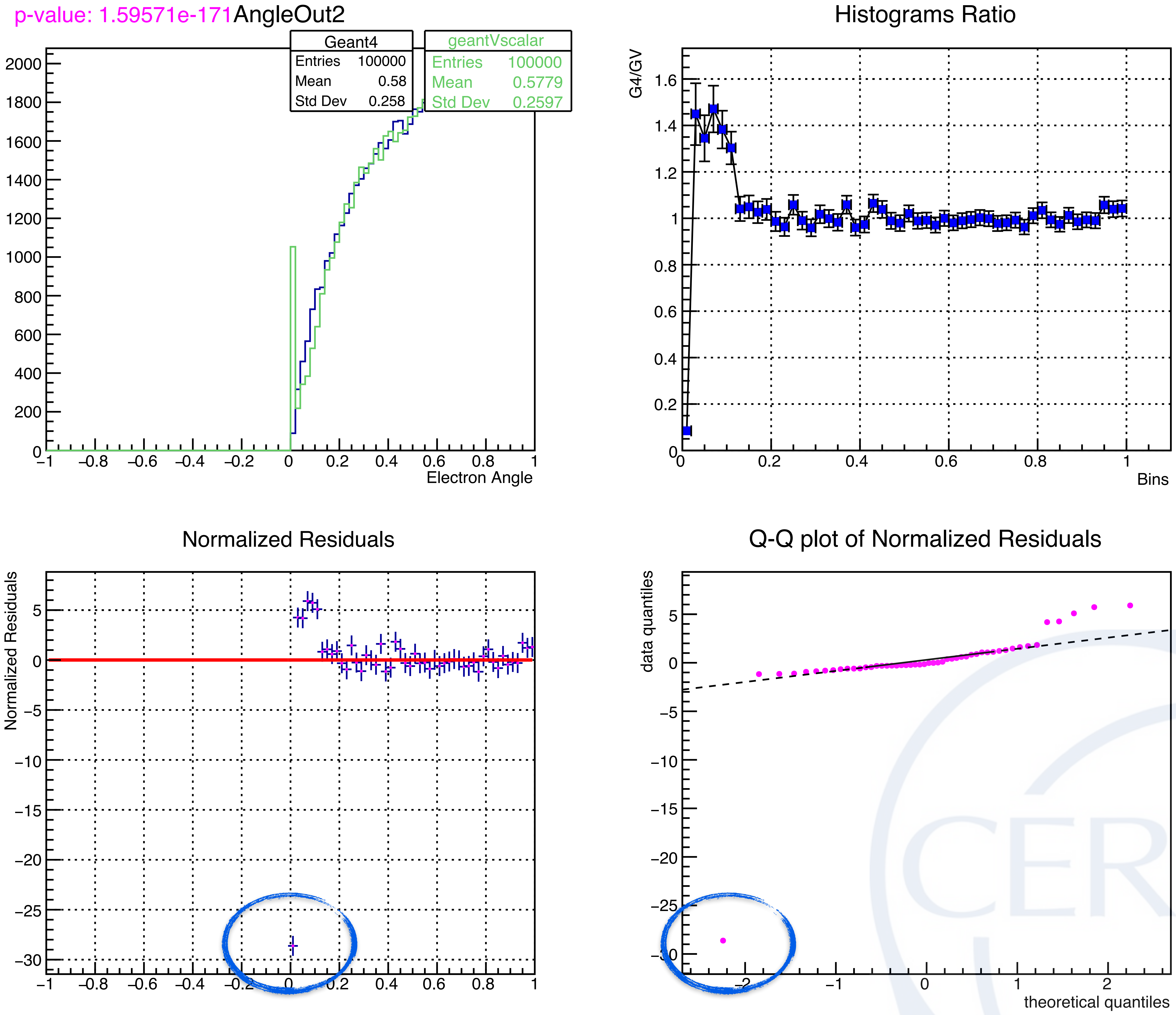
EnergyIn	*ValidationQ*	*Chi-2 test p-value*
0.01 MeV	EnergyOut1	0.650629
	EnergyOut2	0.664124
	AngleOut1	1.2814e-12
	AngleOut2	1.59571e-171
0.1 MeV	EnergyOut1	0.342675
	EnergyOut2	0.209657
	AngleOut1	0.17035
	AngleOut2	1.75073e-68
1 MeV	EnergyOut1	0.241563
	EnergyOut2	0.172393
	AngleOut1	0.379543
	AngleOut2	0.104473
10 MeV	EnergyOut1	7.19275e-32
	EnergyOut2	7.90652e-21
	AngleOut1	0.720494
	AngleOut2	6.33771e-15
100 MeV	EnergyOut1	0.522992
	EnergyOut2	0.0483391
	AngleOut1	0.0463506
	AngleOut2	0.0137113
1000 MeV	EnergyOut1	5.03277e-06
	EnergyOut2	2.58997e-07
	AngleOut1	0
	AngleOut2	0
10000 MeV	EnergyOut1	5.20263e-308
	EnergyOut2	0
	AngleOut1	0
	AngleOut2	0


p-value: 1.2814e-12
AngleOut1
Geant4
Entries 100000
Mean −0.07906
Std Dev 0.4206
geantVscalar
Entries 100000
Mean −0.0779
Std Dev 0.4223
Scattered Photon Angle


Histograms Ratio
G4/GV
Bins


Normalized Residuals
Normalized Residuals

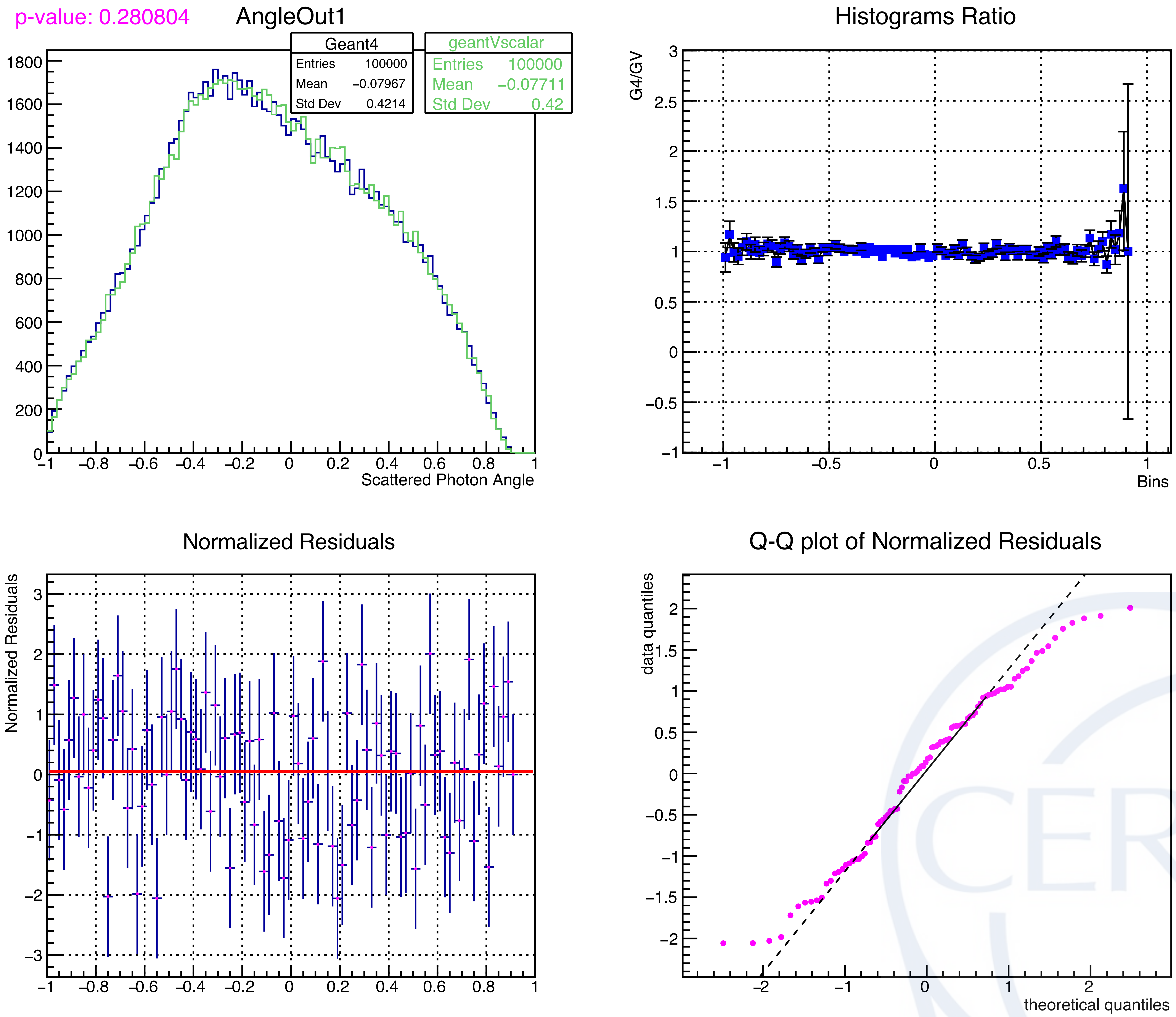

Q-Q plot of Normalized Residuals
data quantiles
theoretical quantiles


p-value: 1.59571e-171
AngleOut2
Geant4
Entries 100000
Mean 0.58
Std Dev 0.258
geantVscalar
Entries 100000
Mean 0.5779
Std Dev 0.2597
Electron Angle
Histograms Ratio
G4/GV
Bins
Normalized Residuals
Normalized Residuals
Q-Q plot of Normalized Residuals
data quantiles
theoretical quantiles

EnergyIn	*ValidationQ*	*Chi-2 test p-value*
0.01 MeV	EnergyOut1	0.31306
	EnergyOut2	0.795741
	AngleOut1	0.280804
	AngleOut2	0.141958
0.1 MeV	EnergyOut1	0.857909
	EnergyOut2	0.858931
	AngleOut1	0.560302
	AngleOut2	1.5663e-15
1 MeV	EnergyOut1	0.0103478
	EnergyOut2	0.0475544
	AngleOut1	0.0167979
	AngleOut2	5.14327e-56
10 MeV	EnergyOut1	0.817173
	EnergyOut2	0.875291
	AngleOut1	0.872177
	AngleOut2	0.995689
100 MeV	EnergyOut1	0.011512
	EnergyOut2	0.124762
	AngleOut1	0.0264691
	AngleOut2	0.387263
1000 MeV	EnergyOut1	5.03277e-06
	EnergyOut2	2.58997e-07
	AngleOut1	0
	AngleOut2	0
10000 MeV	EnergyOut1	5.20263e-308
	EnergyOut2	0
	AngleOut1	0
	AngleOut2	0

E_in=0.01MeV

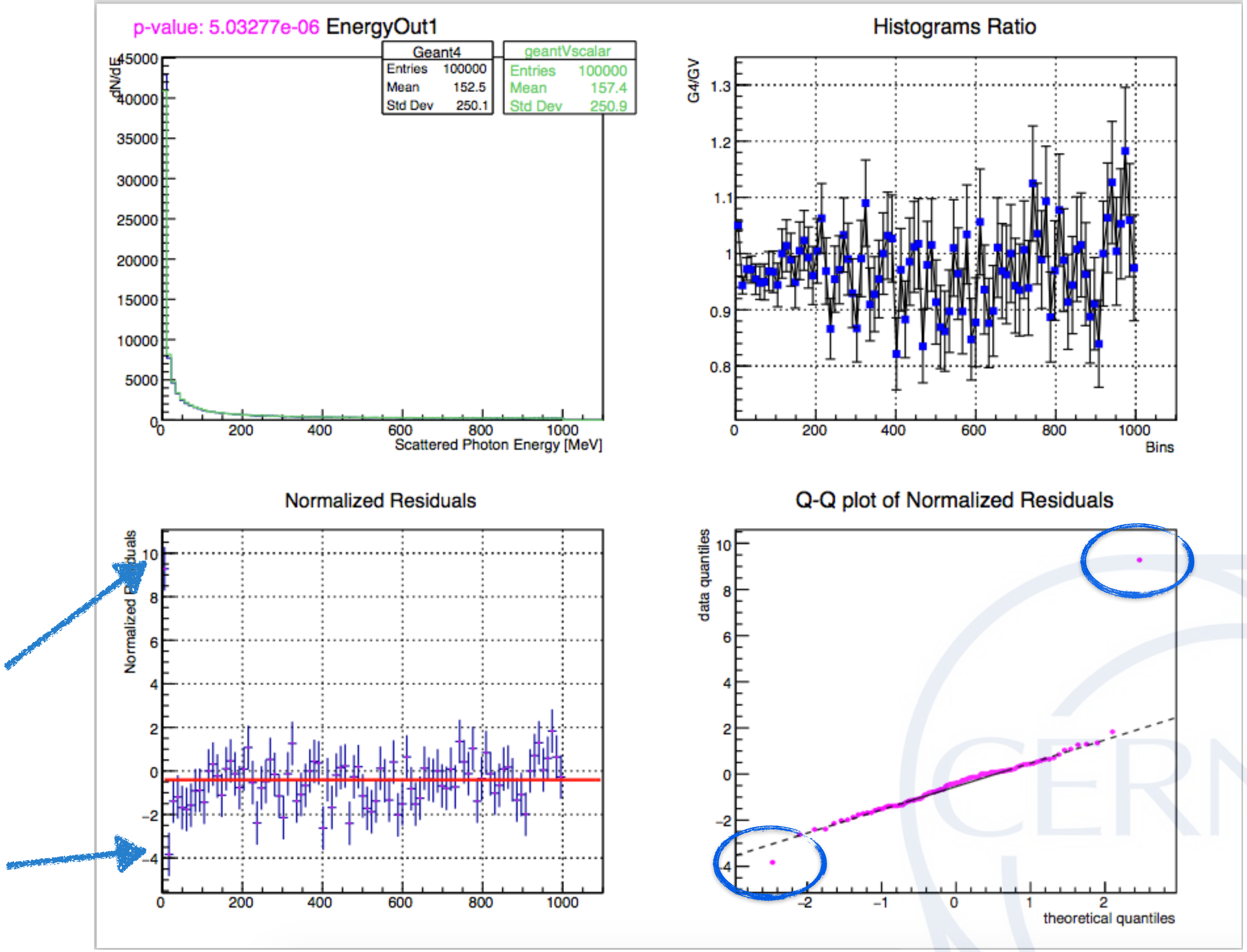
Fixed



EnergyIn	ValidationQ	Chi-2 test p-value
0.01 MeV	EnergyOut1	0.131912
	EnergyOut2	0.492766
	AngleOut1	0.878847
	AngleOut2	0.590701
0.1 MeV	EnergyOut1	0.07002
	EnergyOut2	0.0333841
	AngleOut1	0.120768
	AngleOut2	0.897628
1 MeV	EnergyOut1	0.228669
	EnergyOut2	0.220132
	AngleOut1	0.0470413
	AngleOut2	4.68076e-46
10 MeV	EnergyOut1	0.492207
	EnergyOut2	0.29053
	AngleOut1	0.289294
	AngleOut2	0.139839
100 MeV	EnergyOut1	0.564205
	EnergyOut2	0.817869
	AngleOut1	0.357838
	AngleOut2	0.572742
1000 MeV	EnergyOut1	0.379965
	EnergyOut2	0.219433
	AngleOut1	1.18777e-163
	AngleOut2	1.52104e-193

In the case of Klein-Nishina at higher energies sampling a continuos variable discretising the pdf, bias the outcomes

E_in=1000 MeV

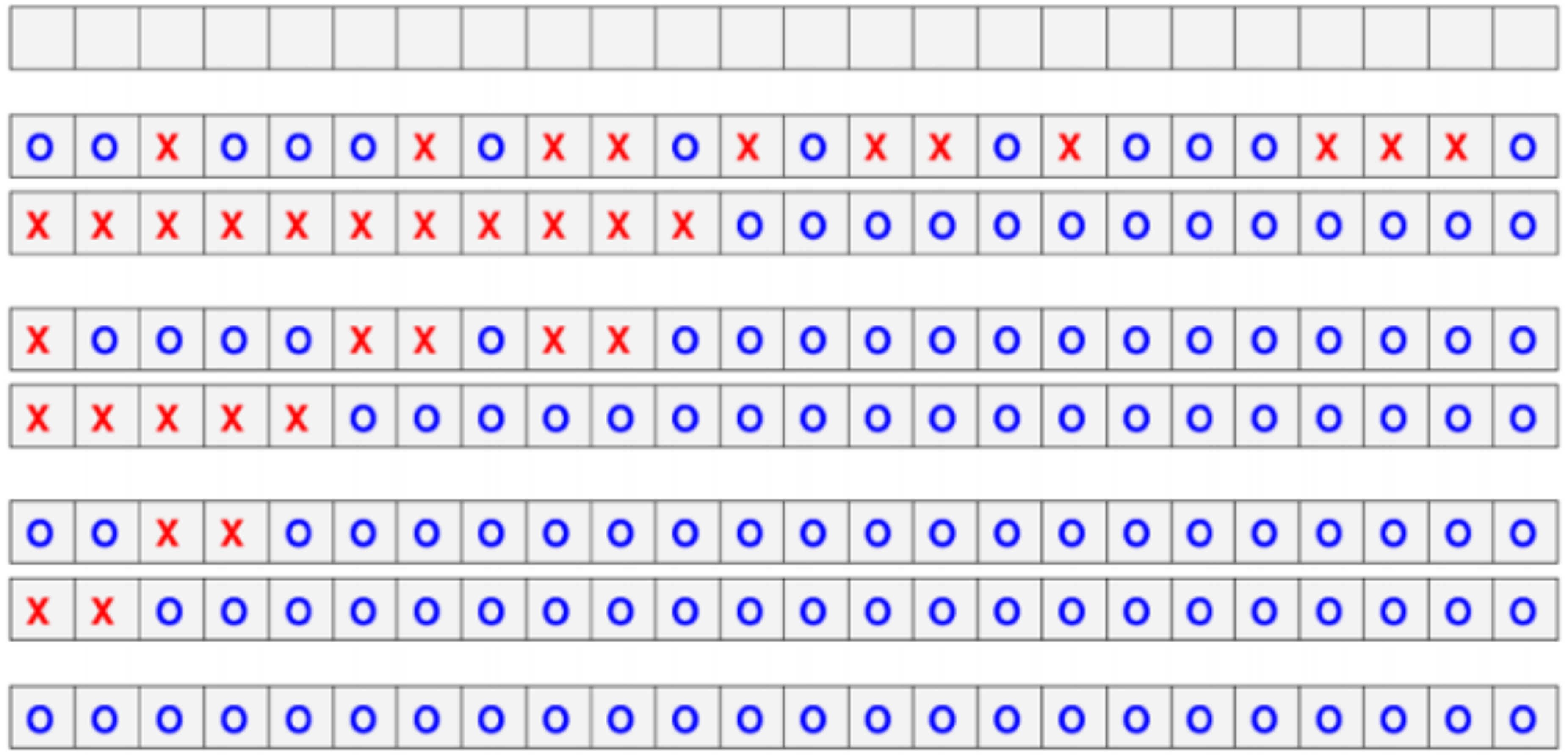


p-value: 5.03277e-06 EnergyOut1
Geant4
Entries 100000
Mean 152.5
Std Dev 250.1
geantVscalar
Entries 100000
Mean 157.4
Std Dev 250.9
dN/dE
Scattered Photon Energy [MeV]
Histograms Ratio
G4/GV
Bins
Normalized Residuals
Q-Q plot of Normalized Residuals
data quantiles
theoretical quantiles

Alternative sampling methods

- **Limitation of the discrete alias sampling method**
  - The alias method with a finite bin size is subject to have a biased outcomes if pdf is neither near constant not linear within a bin (sampling a continuos variable discretising the pdf).
    - *Improvement of* the Alias sampling (adaptive binning, transformation)

- **Alternative techniques using the composition and rejection**
  - Parallel (vector) + Sequential (scalar) loop over the vector width
  - Shuffling (try and unpack, overhead for reorganising data)



  - Hybrid (mixture of different methods in the parameter space)

See talk "Computing Performance of GeantV Physics Models" by Soon Y. Jun)!

Hybrid KleinNishina Pearson chi2Test results

EnergyIn	*ValidationQ*	*Chi-2 test p-value*
1 MeV	EnergyOut1	0.580762
	EnergyOut2	0.187933
	AngleOut1	0.355483
	AngleOut2	0.823083
10 MeV	EnergyOut1	0.00633212
	EnergyOut2	0.00199365
	AngleOut1	0.0351077
	AngleOut2	0.100245
100 MeV	EnergyOut1	0.834322
	EnergyOut2	0.903547
	AngleOut1	0.795684
	AngleOut2	0.809488
250 MeV	EnergyOut1	0.930848
	EnergyOut2	0.856883
	AngleOut1	0.0656924
	AngleOut2	0.721054
500 MeV	EnergyOut1	0.937476
	EnergyOut2	0.841945
	AngleOut1	0.0737359
	AngleOut2	0.114897
1000 MeV	EnergyOut1	0.523284
	EnergyOut2	0.777374
	AngleOut1	0.638076
	AngleOut2	0.656735
10000 MeV	EnergyOut1	0.73755
	EnergyOut2	0.615561
	AngleOut1	0.247631
	AngleOut2	0.360034
100000 MeV	EnergyOut1	0.121317
	EnergyOut2	0.19909
	AngleOut1	0.881816
	AngleOut2	0.86112

Upcoming work and further developments

- Integration of Compton VecPhys Process with the GeantV Scheduler
- Investigation of *different sampling techniques* and *improvement of* the Alias sampling (adaptive binning, transformation)
- Extend validation for intermediate energies (around 100 MeV and in the middle of the bins)
- Validation against experimental data.

Summary

- **Focus on alternative sampling techniques**
    - The actual implementation of the Alias sampling introduces discretization errors, especially when the sampled distributions are very steep
        - Found and fixed problems for Alias-Compton up to 100MeV
        - Investigating the threshold energy (boundary between Alias and Shuffling sampling methods 100 MeV at the moment)
- **We have built a verification suite which can identify deficiencies of algorithms or errors in implementation**
    - robust test/verification suite which compares all the relevant physical quantities of output particles
- ***Tools automation* is advanced, easing future validations**
    - P-value Tables automatically generated
    - Statistical analysis and graphs automatically generated

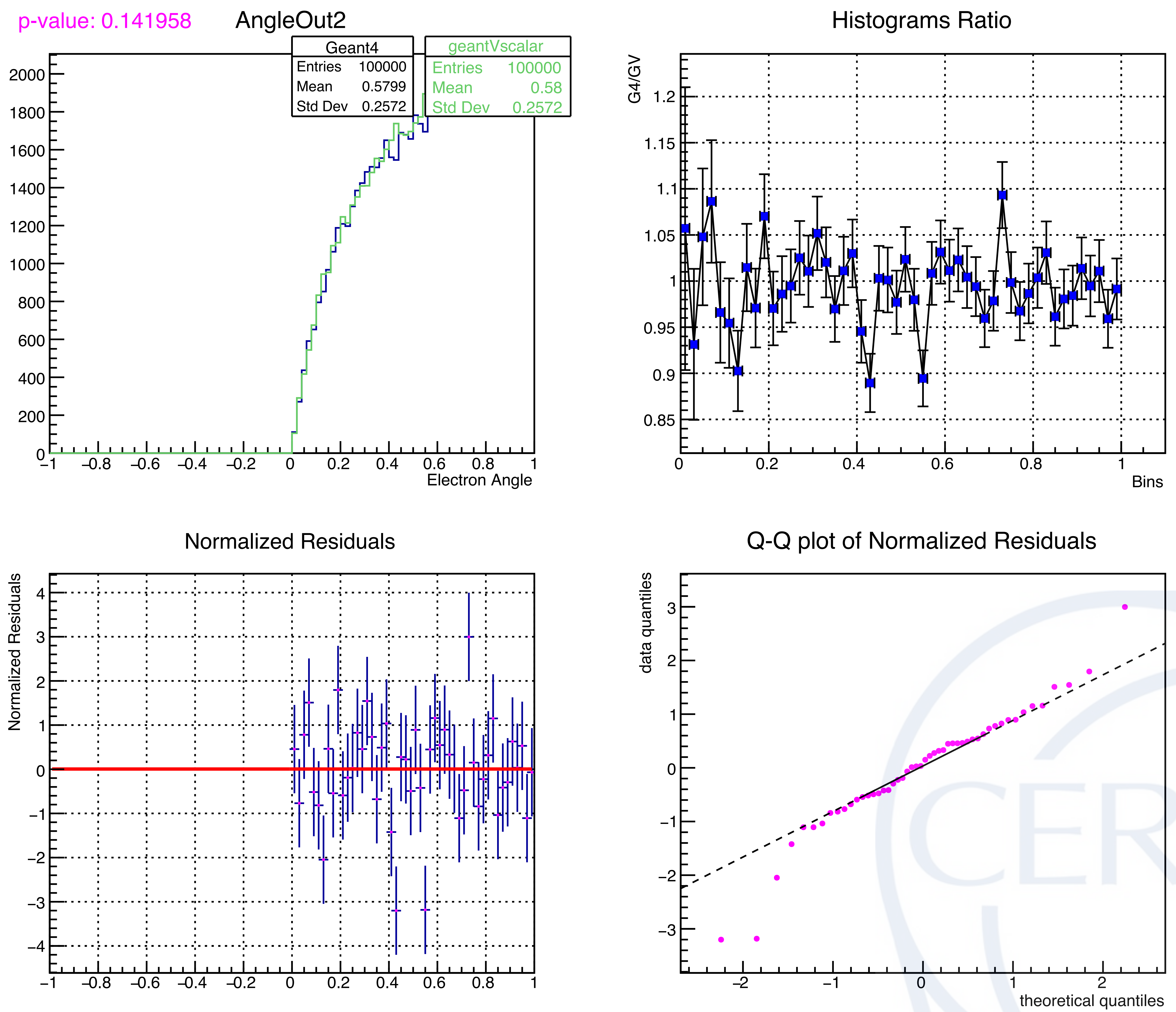
Questions?

Marilena Bandieramonte
(marilena.bandieramonte@cern.ch)
for the GeantV team

Backup

E_in=0.01MeV

Fixed



Pearson χ^2-test for comparing weighted and unweighted histograms

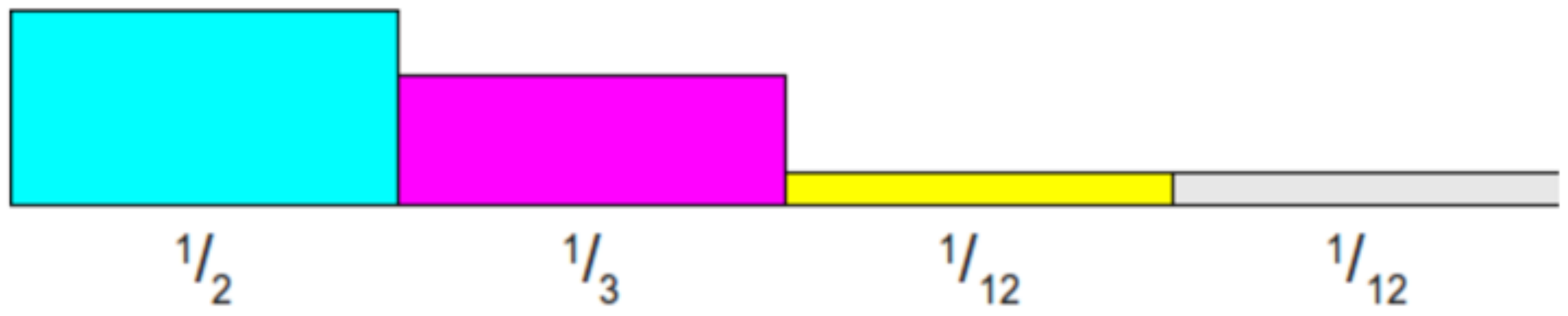
- Comparison of two histograms expect hypotheses that two histograms represent identical distributions. To make a decision **p-value** should be calculated.

- The hypotheses of identity is rejected if the p-value is lower than some significance level.

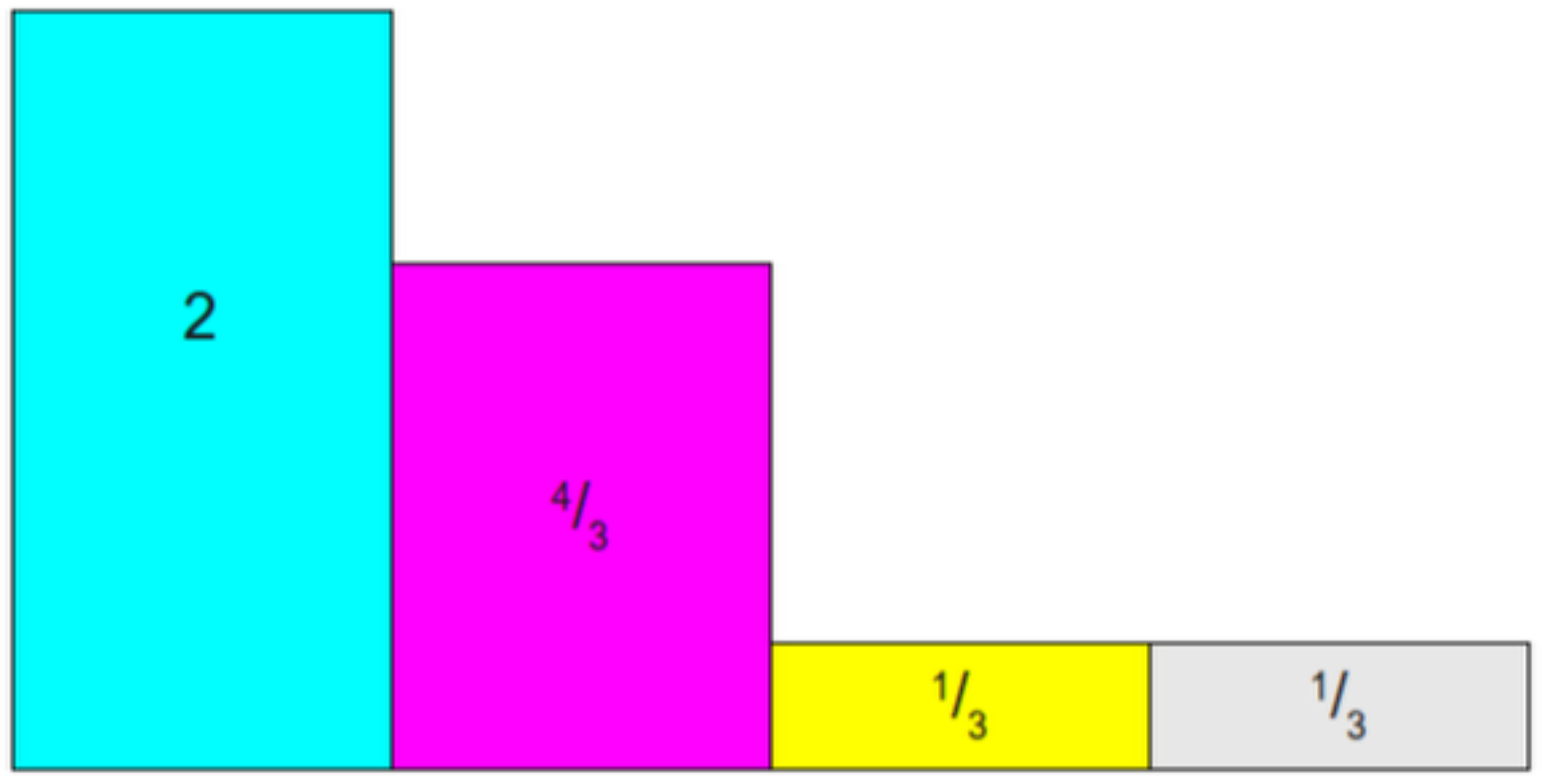
  - Traditionally significance levels 0.1, 0.05 and 0.01 are used.
  - **Chosen threshold 0.05 -> If p-value < 0.05 hypothesis of identity is rejected!**

- The comparison procedure should include an **analysis of the residuals** which is often helpful in identifying the bins of histograms responsible for a significant overall #chi^{2} value.

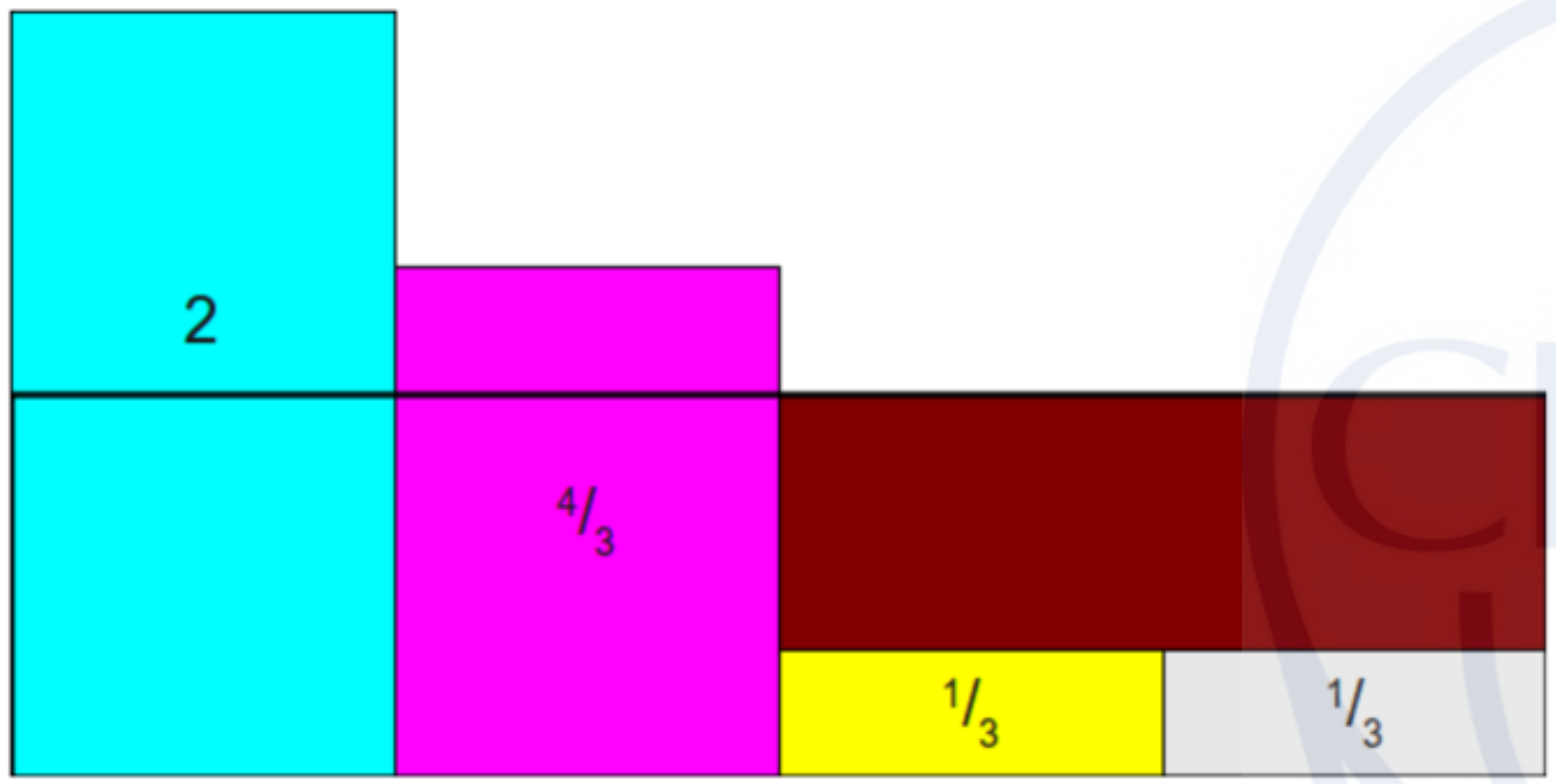
Alias method



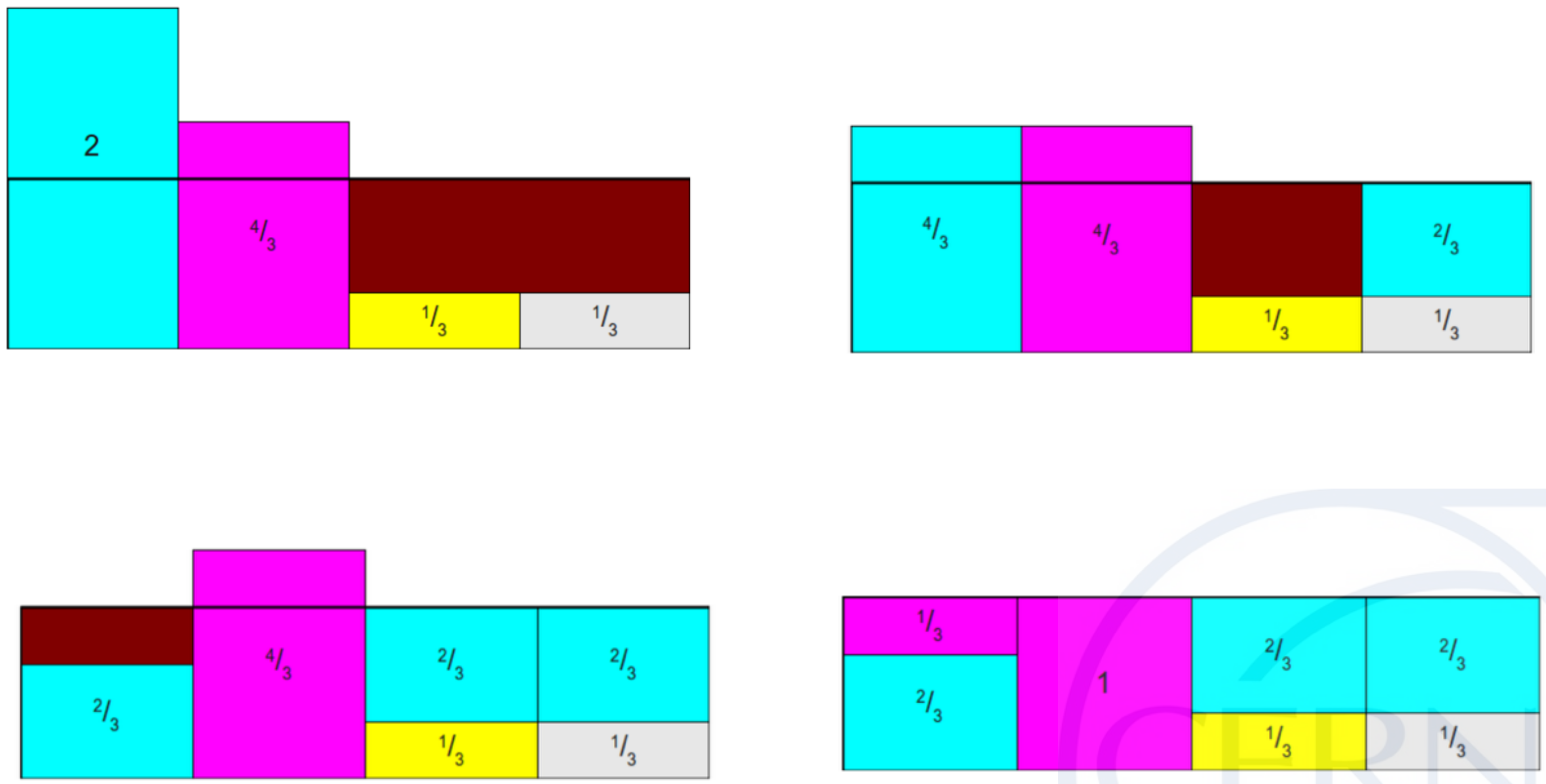
Initial pdf (equal likelihood=1/4)



Scaled probabilities so that a prob of 1/4 would weight 1



Alias method



Algorithm: Alias Method

- **Initialization**:
  1. Create arrays $Alias$ and $Prob$, each of size n.
  2. Create a balanced binary search tree T.
  3. Insert $n \cdot p_i$ into T for each probability i.
  4. For $j = 1$ to $n - 1$:
     1. Find and remove the smallest value in T; call it p_l.
     2. Find and remove the largest value in T; call it p_g.
     3. Set $Prob[l] = p_l$.
     4. Set $Alias[l] = g$.
     5. Set $p_g := p_g - (1 - p_l)$.
     6. Add p_g to T.
  5. Let i be the last probability remaining, which must have weight 1.
  6. Set $Prob[i] = 1$.
- **Generation**:
  1. Generate a fair die roll from an n-sided die; call the side i.
  2. Flip a biased coin that comes up heads with probability $Prob[i]$.
  3. If the coin comes up "heads," return i.
  4. Otherwise, return $Alias[i]$.

Improvements of the algorithm - Vose's algorithm

This algorithm was originally described in the paper "A Linear Algorithm For Generating Random Numbers With a Given Distribution" by Michael Vose

The idea behind Vose's algorithm is to maintain two worklists, one containing the elements whose height is less than 1 and one containing the elements whose height is at least 1, and to repeatedly pair the first elements of each worklist.

On each iteration, we consume the element from the "small" worklist, and potentially move the remainder of the element from the "large" worklist into the "small" worklist.

The algorithm maintains several invariants:
- The elements of the "small" worklist are all less than 1.
- The elements of the "large" worklist are all at least 1.

Algorithm	Initialization Time		Generation Time		Memory Usage	
	Best	Worst	Best	Worst	Best	Worst
Alias Method	$O(n \log n)$		$\Theta(1)$		$\Theta(n)$	
Vose's Alias Method	$\Theta(n)$		$\Theta(1)$		$\Theta(n)$	

Ref: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=92917

Algorithm: (Unstable) Vose's Alias Method

- **Initialization**:
  1. Create arrays $Alias$ and $Prob$, each of size n.
  2. Create two worklists, $Small$ and $Large$.
  3. Multiply each probability by n.
  4. For each scaled probability p_i:
     1. If $p_i < 1$, add i to $Small$.
     2. Otherwise ($p_i \geq 1$), add i to $Large$.
  5. While $Small$ is not empty:
     1. Remove the first element from $Small$; call it l.
     2. Remove the first element from $Large$; call it g.
     3. Set $Prob[l] = p_l$.
     4. Set $Alias[l] = g$.
     5. Set $p_g := p_g - (1 - p_l)$.
     6. If $p_g < 1$, add g to $Small$.
     7. Otherwise ($p_g \geq 1$), add g to $Large$.
  6. While $Large$ is not empty:
     1. Remove the first element from $Large$; call it g.
     2. Set $Prob[g] = 1$.
- **Generation**:
  1. Generate a fair die roll from an n-sided die; call the side i.
  2. Flip a biased coin that comes up heads with probability $Prob[i]$.
  3. If the coin comes up "heads," return i.
  4. Otherwise, return $Alias[i]$.

Is it a stable version?

Unfortunately, the above algorithm, as written, is not numerically stable.
Two sources of inaccuracy:

- The computation to determine whether or not a probability belongs to the Small or Large group may be inaccurate. Specifically, it may be possible that scaling up the probabilities by a factor of n may cause probabilities equal to 1/n to end up being slightly less than 1 (ending up in the Small list rather than in the Large one)

- The computation that subtracts the appropriate probability mass from a larger probability is not numerically stable and may introduce significant rounding errors.

A Vose's Alias method stable implementation

We will update the inner loop of the algorithm so that it terminates whenever either of the two worklists are empty, so we don't accidentally end up looking at nonexistent elements from the Large worklist.

Second, when one worklist is empty, we'll set the remaining probabilities of the elements in the other worklist to all be 1 since, mathematically, this should only occur if all of the remaining probabilites are precisely equal to 1.

Finally, we'll replace the computation that updates the large probabilities with a slightly more stable computation.

Algorithm: Vose's Alias Method

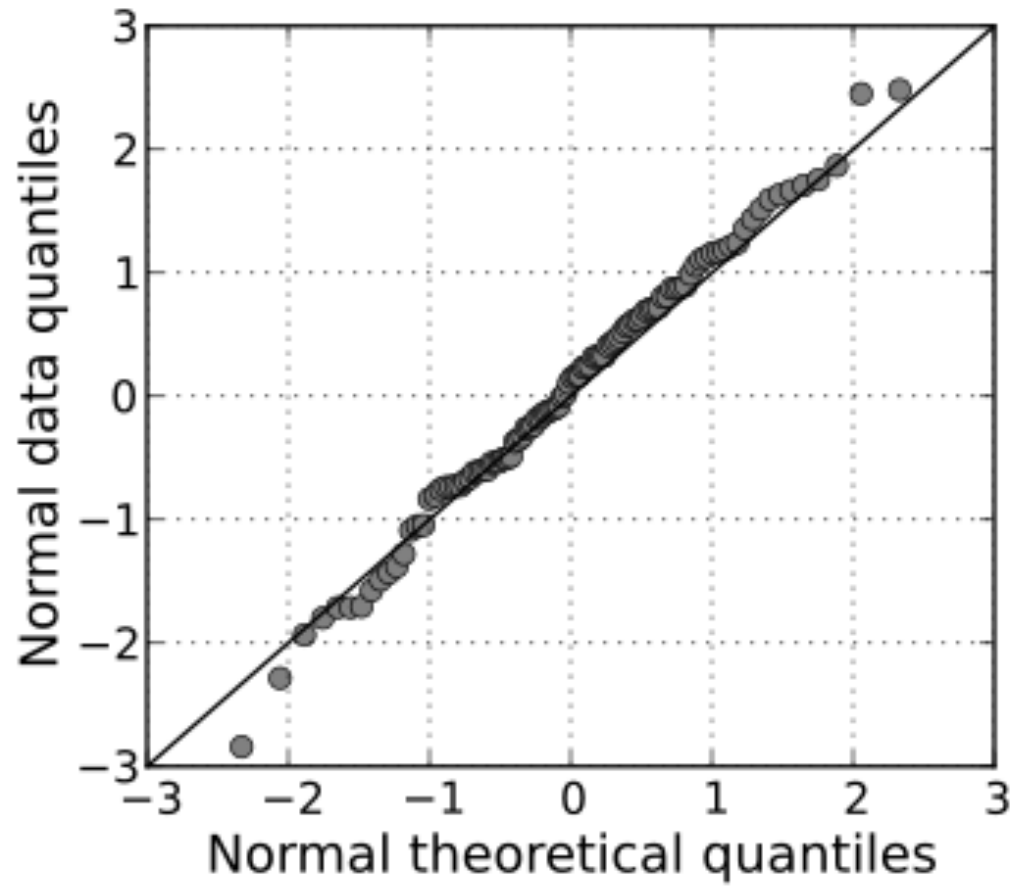
- **Initialization**:
  1. Create arrays $Alias$ and $Prob$, each of size n.
  2. Create two worklists, $Small$ and $Large$.
  3. Multiply each probability by n.
  4. For each scaled probability p_i:
     1. If $p_i < 1$, add i to $Small$.
     2. Otherwise ($p_i \geq 1$), add i to $Large$.
  5. While $Small$ and $Large$ are not empty: *($Large$ might be emptied first)*
     1. Remove the first element from $Small$; call it l.
     2. Remove the first element from $Large$; call it g.
     3. Set $Prob[l] = p_l$.
     4. Set $Alias[l] = g$.
     5. Set $p_g := (p_g + p_l) - 1$. *(This is a more numerically stable option.)*
     6. If $p_g < 1$, add g to $Small$.
     7. Otherwise ($p_g \geq 1$), add g to $Large$.
  6. While $Large$ is not empty:
     1. Remove the first element from $Large$; call it g.
     2. Set $Prob[g] = 1$.
  7. While $Small$ is not empty: *This is only possible due to numerical instability.*
     1. Remove the first element from $Small$; call it l.
     2. Set $Prob[l] = 1$.
- **Generation**:
  1. Generate a fair die roll from an n-sided die; call the side i.
  2. Flip a biased coin that comes up heads with probability $Prob[i]$.
  3. If the coin comes up "heads," return i.
  4. Otherwise, return $Alias[i]$.

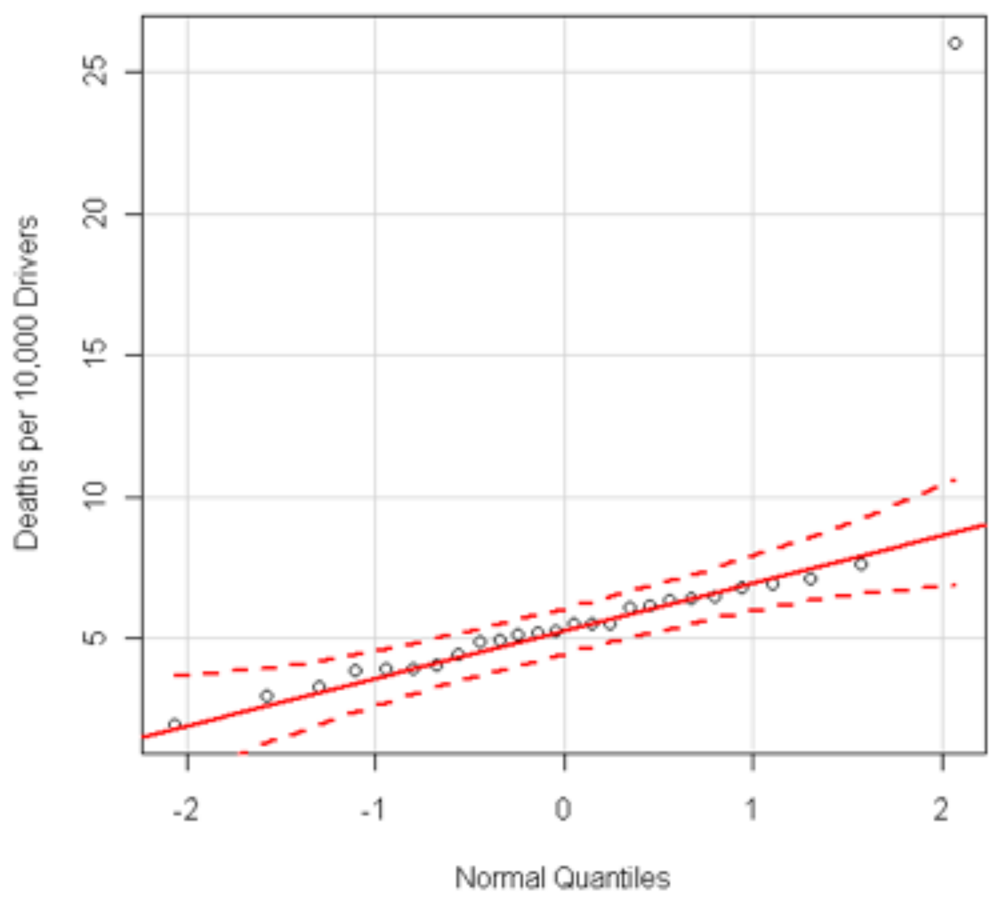
Q-Q plots

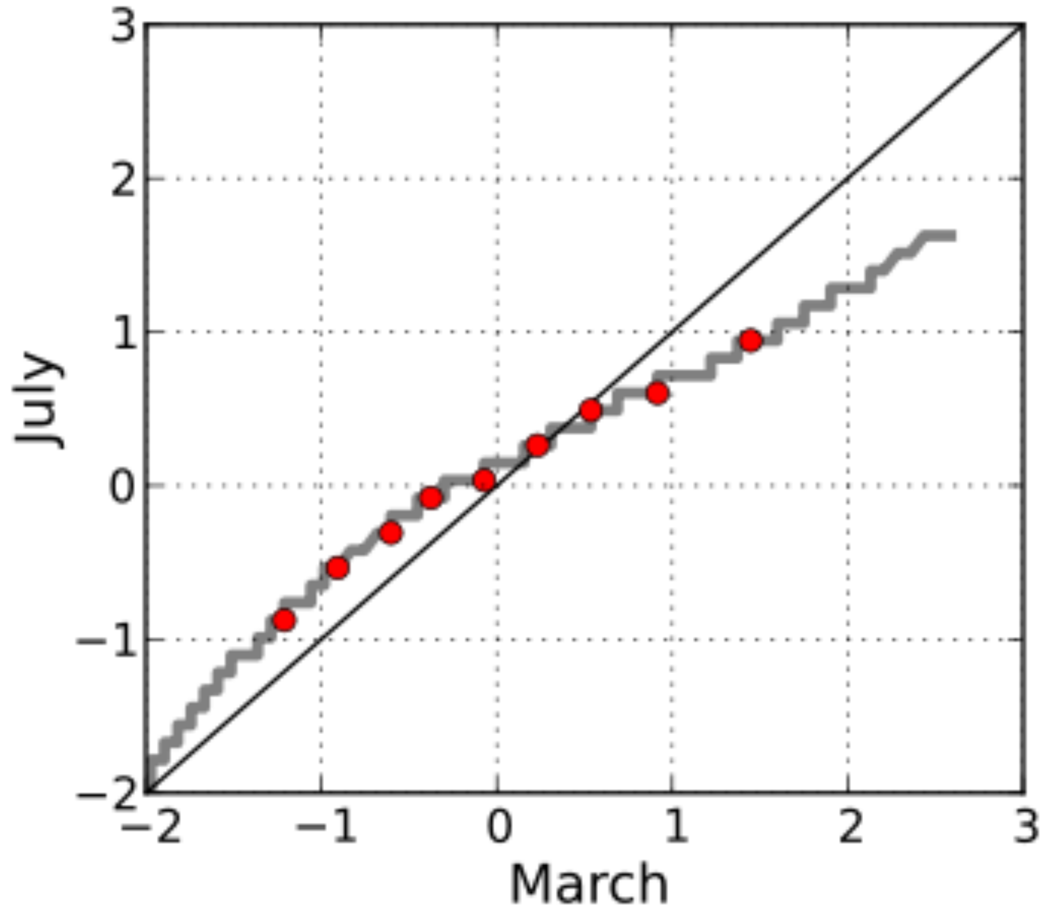
- In statistics, a **Q–Q plot** ("Q" stands for *quantile*) is a probability plot, which is a graphical method for comparing two probability distributions by plotting their quantiles against each other.

- Quantiles are cutpoints dividing a set of observations into equal sized groups.

- A point (x, y) on the plot corresponds to one of the quantiles of the second distribution (y-coordinate) plotted against the same quantile of the first distribution (x-coordinate). Thus the line is a **parametric curve** with the parameter which is the (number of the) interval for the quantile.





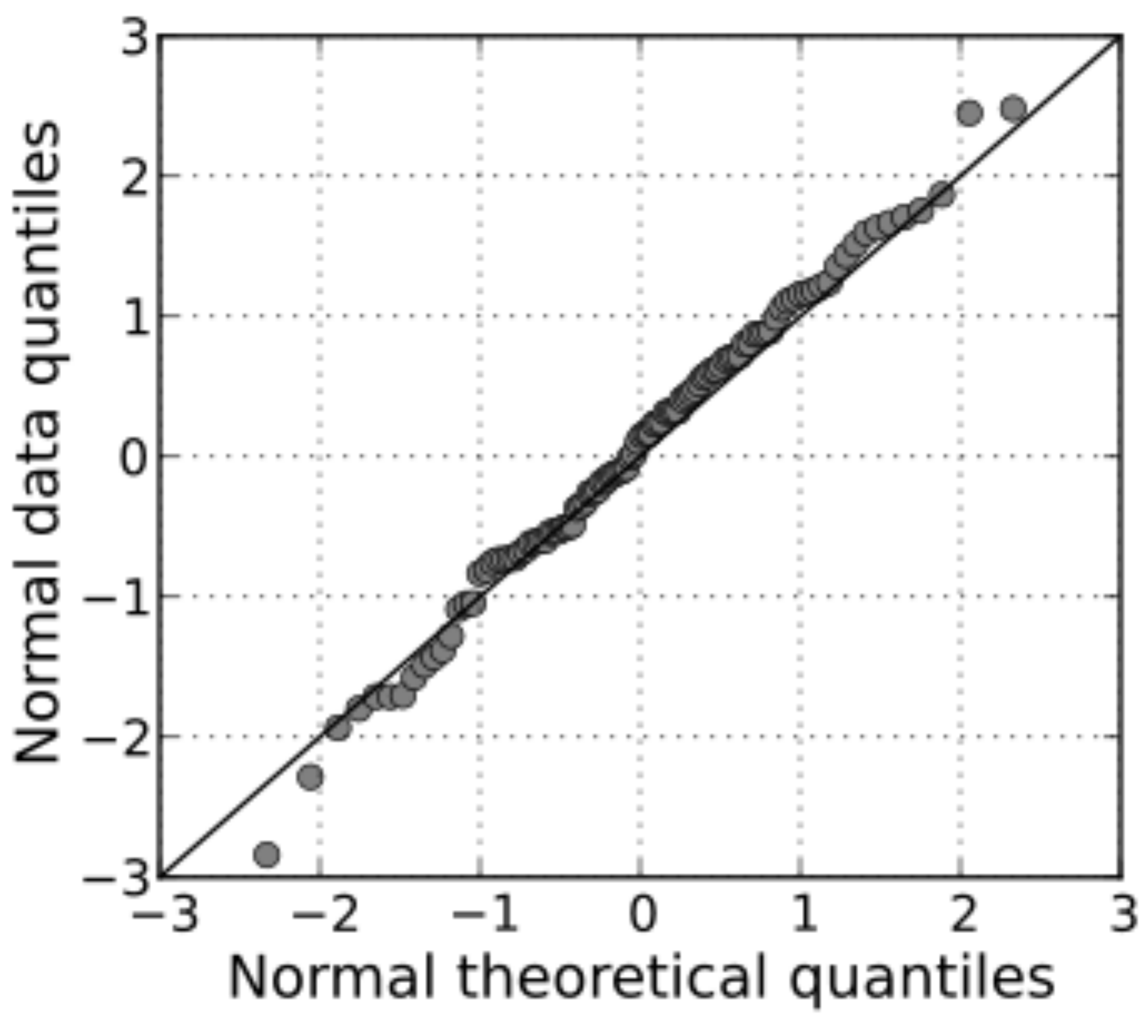


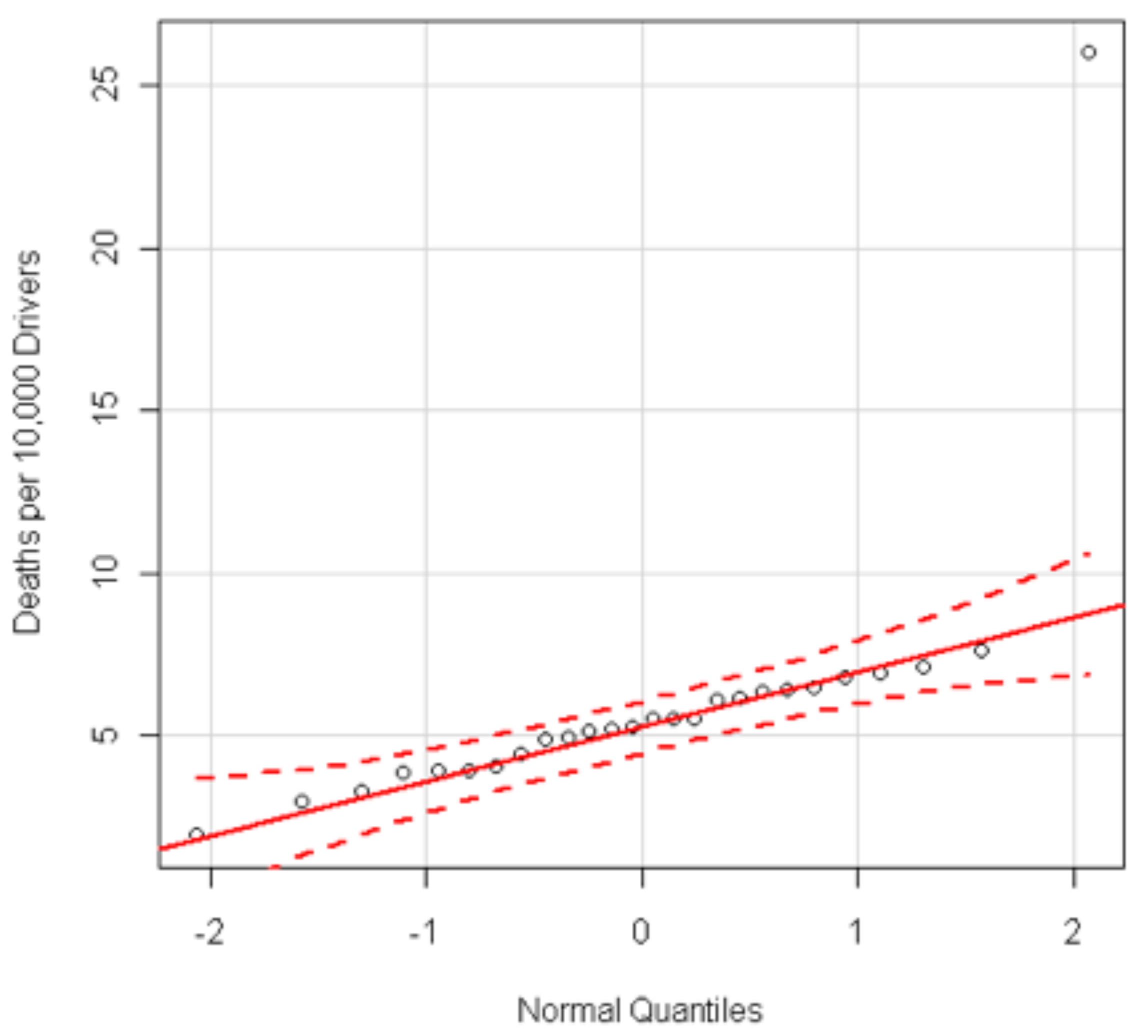
Q-Q plots

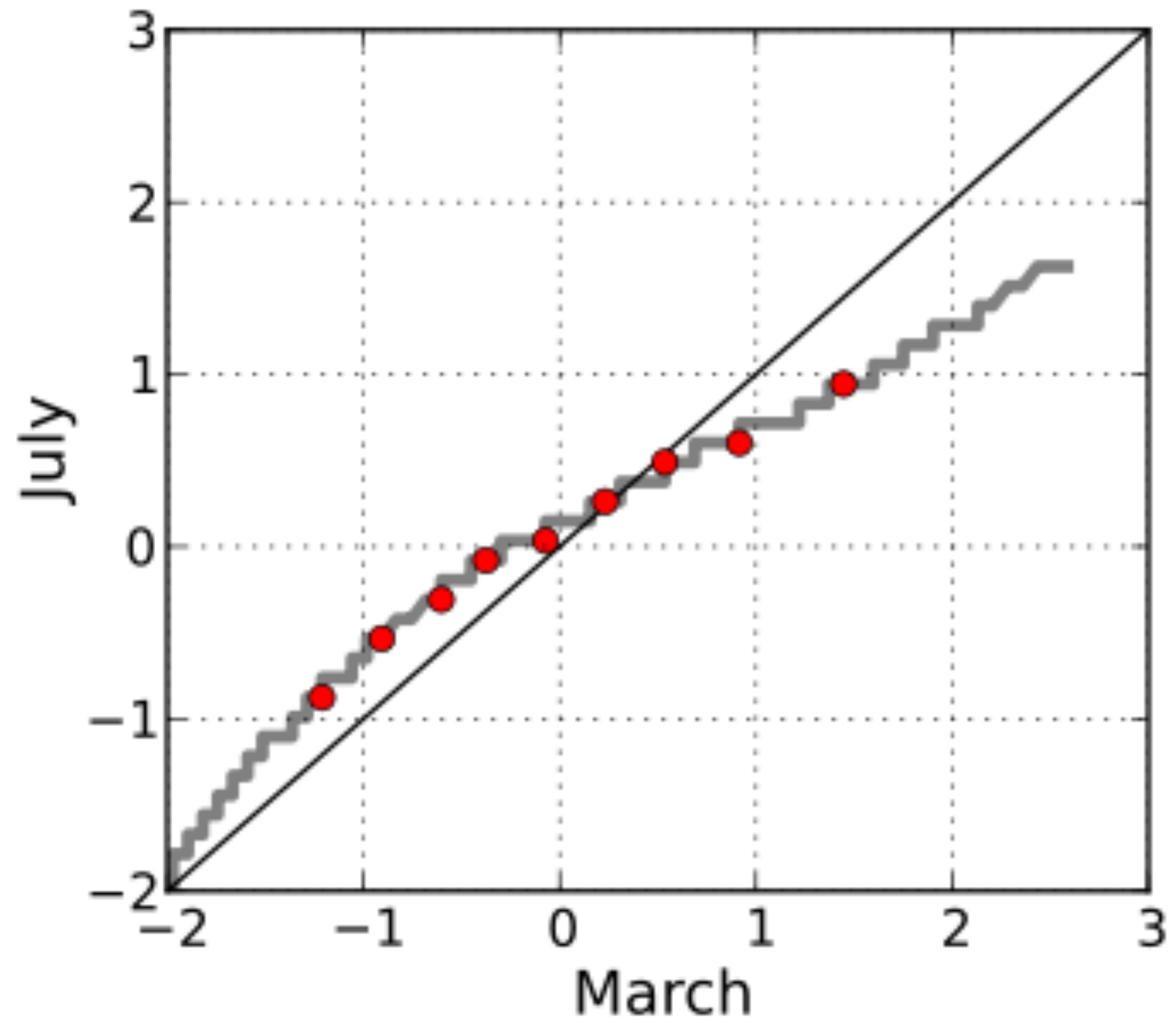
- If the two distributions being compared are **similar**, the points in the Q–Q plot will approximately lie on the line $y = x$.

- If the distributions are **linearly related**, the points in the Q–Q plot will approximately lie on a line, but not necessarily on the line $y = x$.

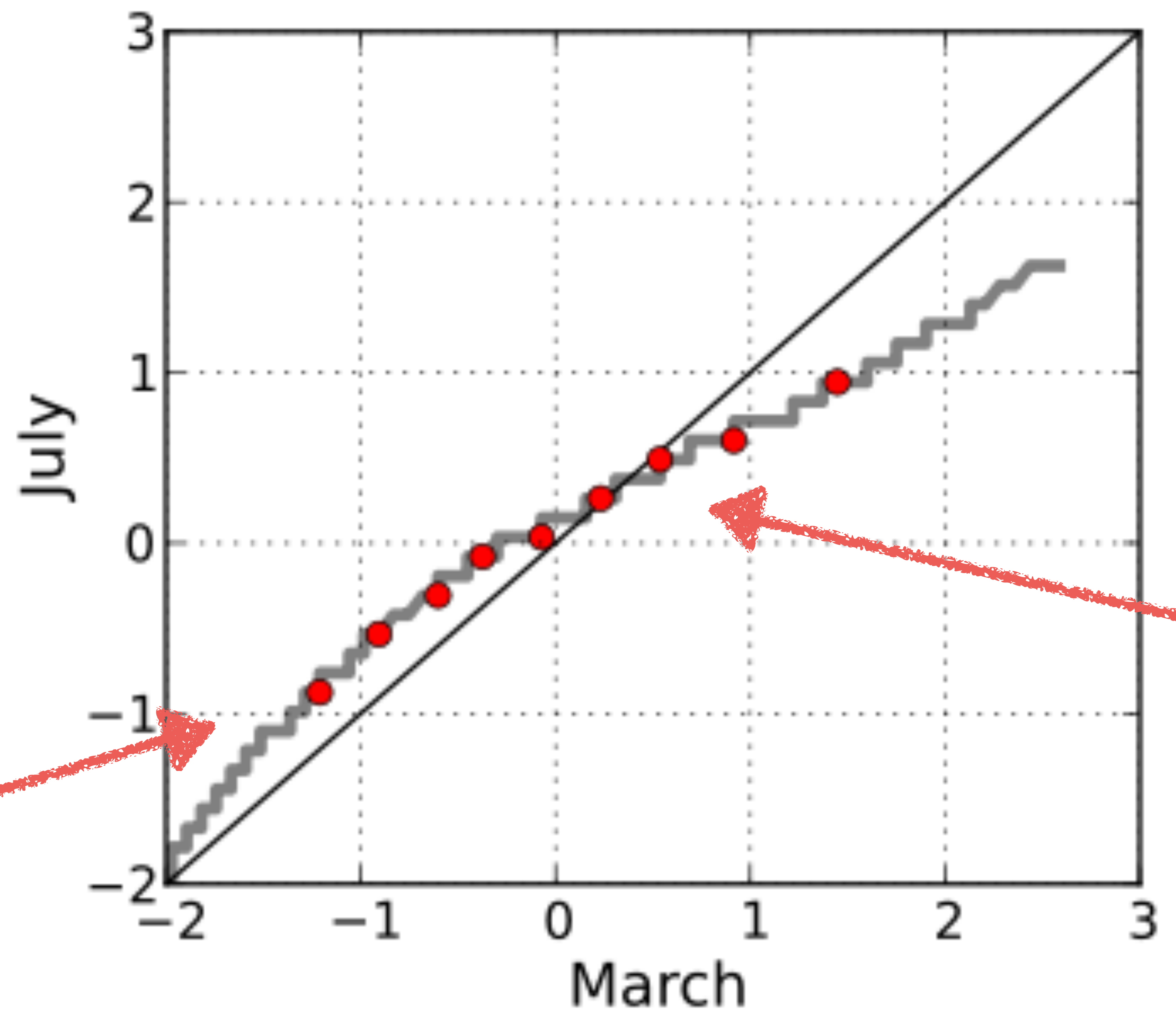
- A Q–Q plot is used to compare the shapes of distributions, providing a graphical view of how properties such as location, **scale**, and **skewness** are similar or different in the two distributions.







Q-Q plots



Daily maximum

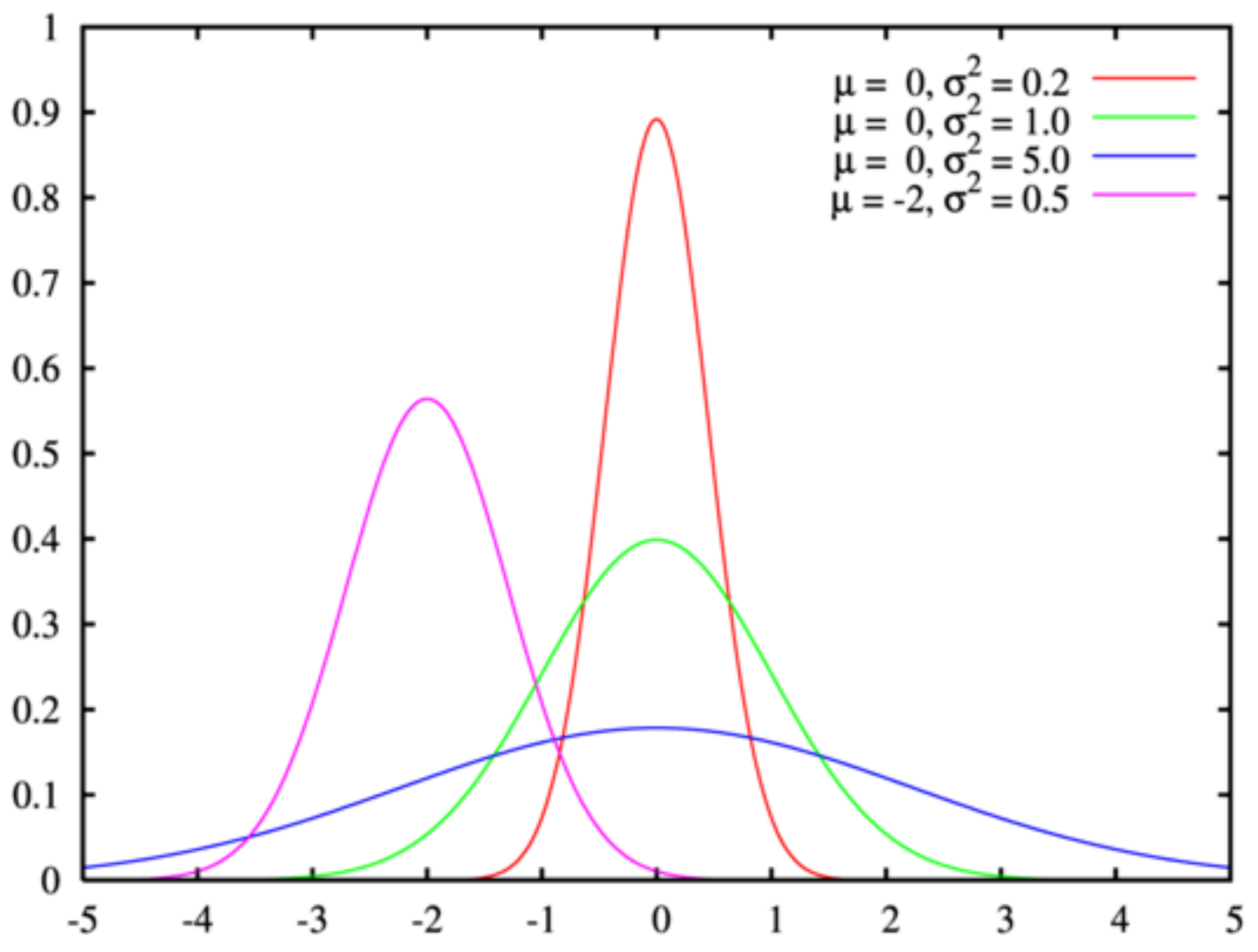
July distribution is skewed to the left compared to the March distribution

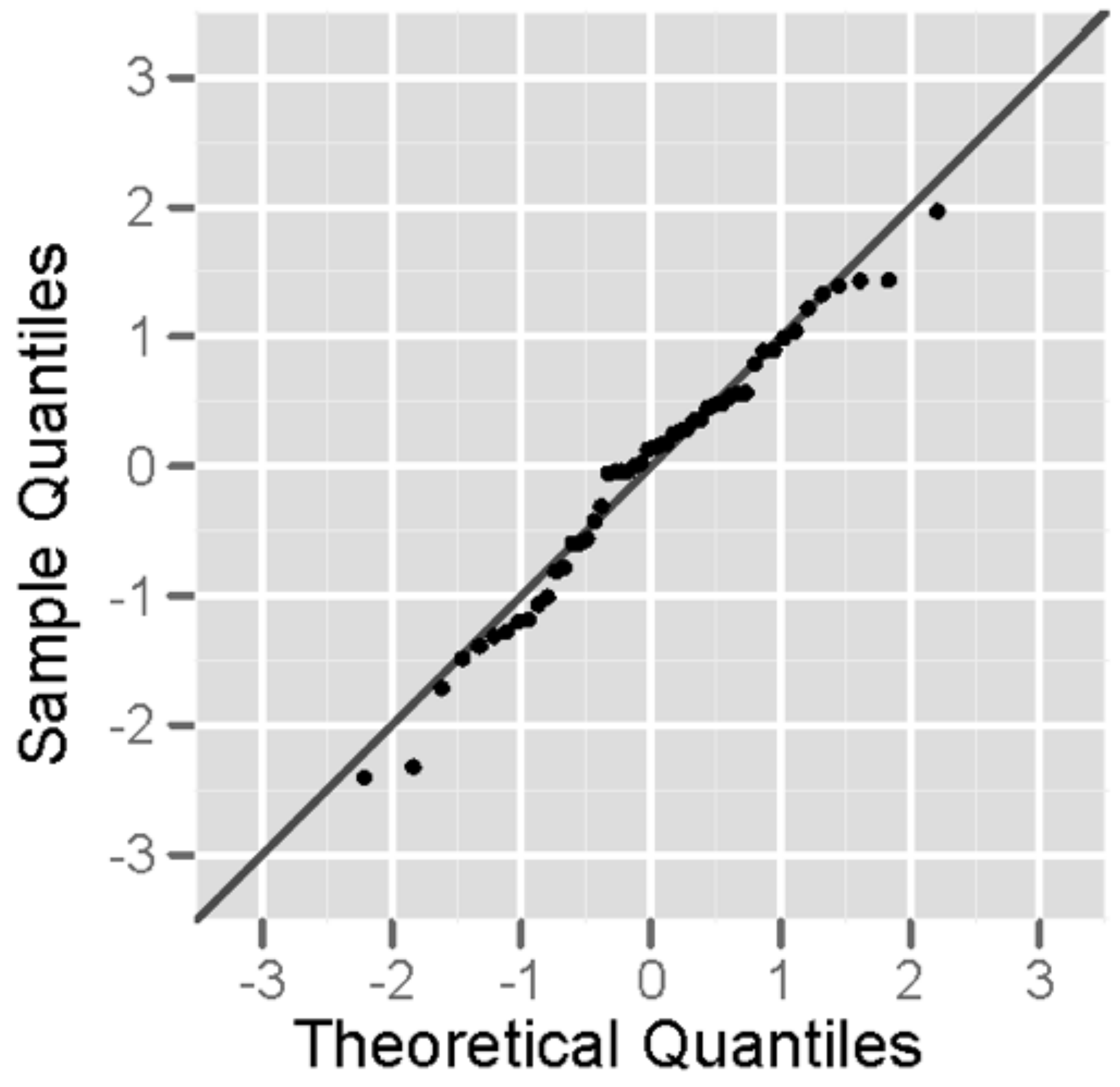
the central quantiles are more closely spaced in July than in March

A Q–Q plot comparing the distributions of standardised daily maximum temperatures at 25 stations in the US state of Ohio in March and in July. The data cover the period 1893–2001.

Normal Probability Plot

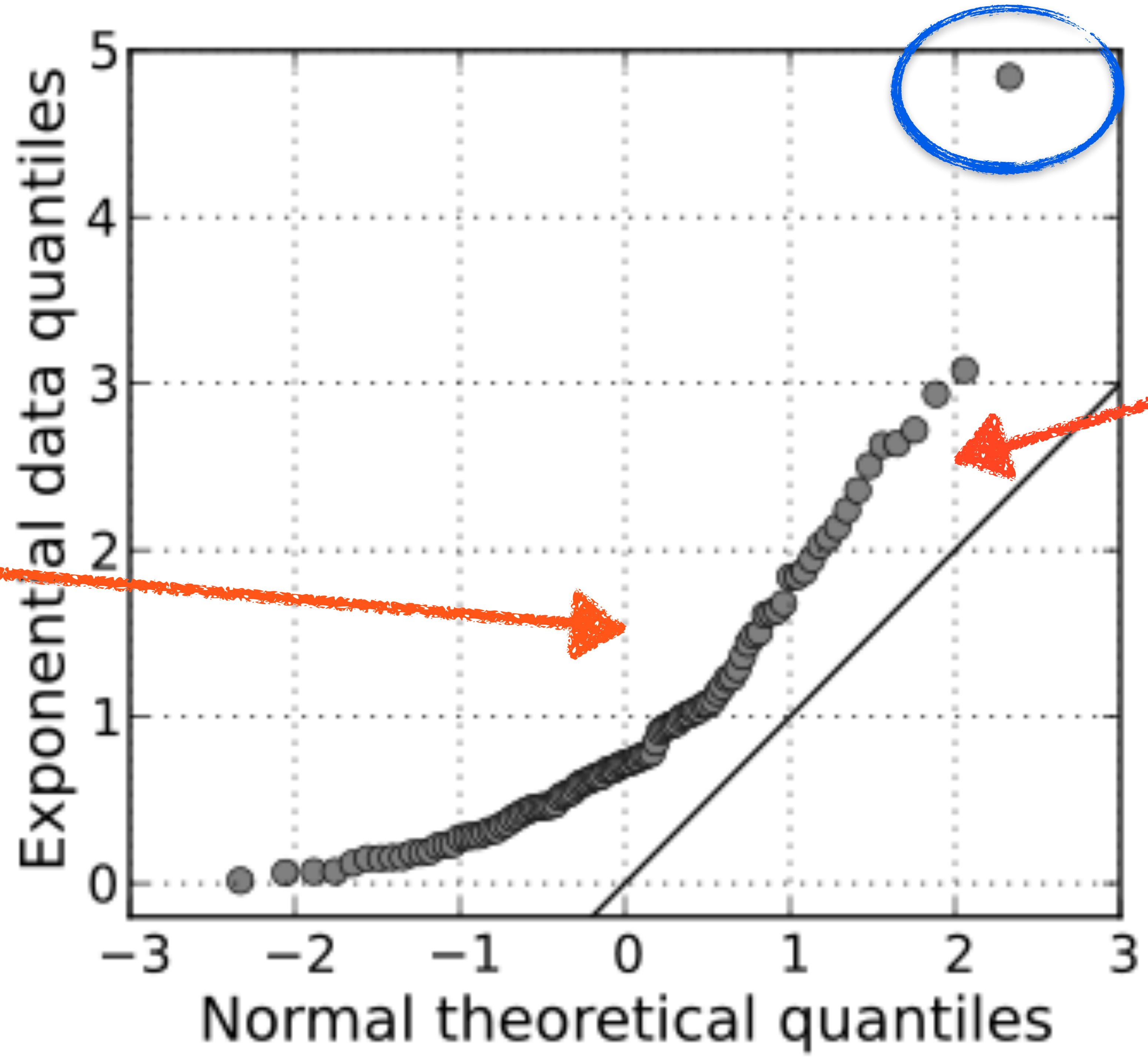
- The **normal probability plot** is a graphical technique to identify substantive departures from normality. This includes identifying outliers, skewness, kurtosis, a need for transformations, and mixtures. Normal probability plots are made of raw data, residuals from model fits, and estimated parameters.





Normal Probability Plot

The points follow a strongly nonlinear pattern, suggesting that the data are not distributed as a standard normal (X ~ N(0,1))

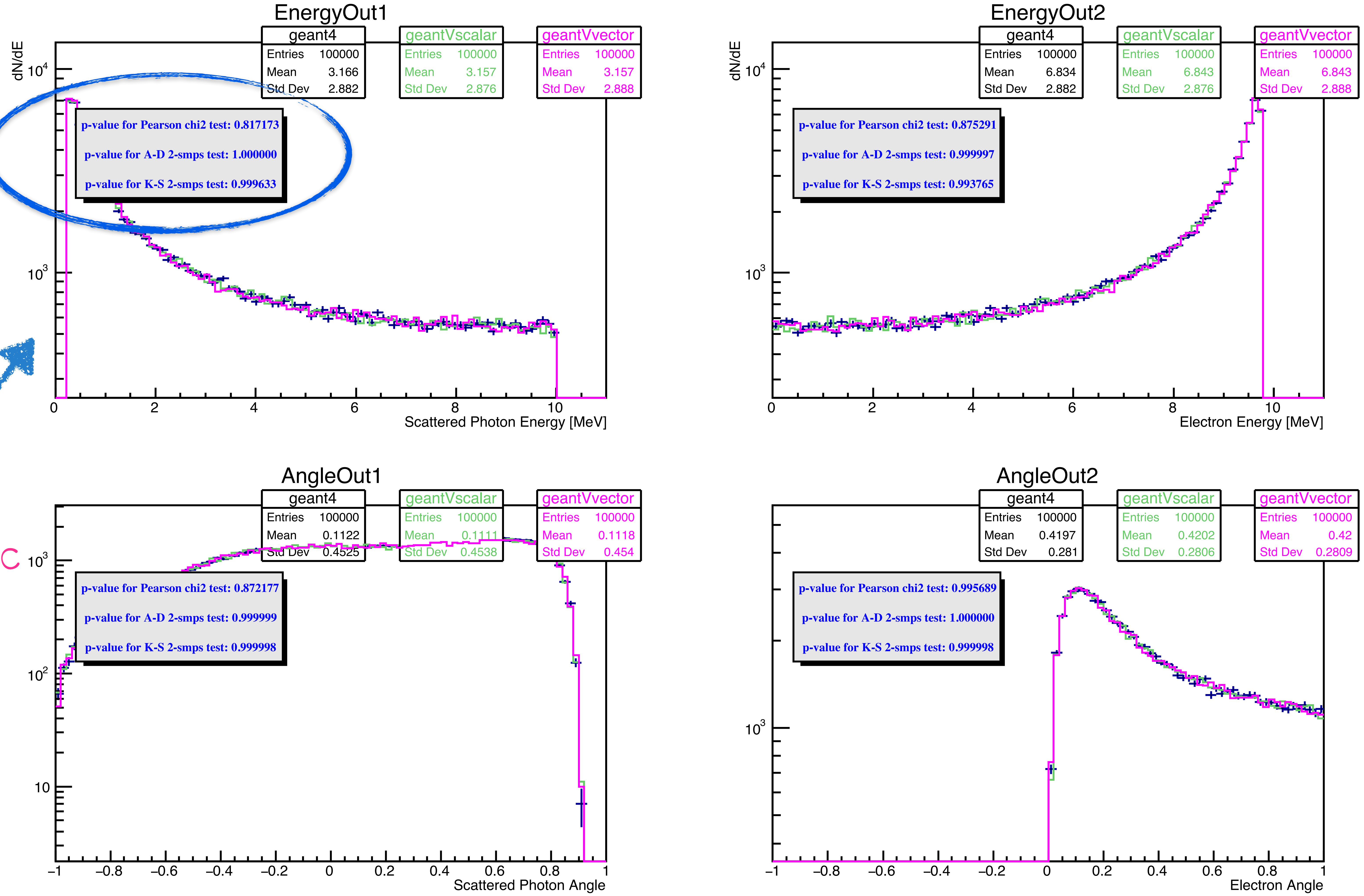


The offset between the line and the points suggests that the mean of the data is not 0. The median of the points can be determined to be near 0.7

A normal Q–Q plot of randomly generated, independent standard exponential data, (X ~ Exp(1)). This Q–Q plot compares a sample of data on the vertical axis to a statistical population on the horizontal axis.

Pearson, A-D and K-S p-values

Logarithmic scale



χ^2-test for comparing weighted and unweighted histograms

- **Residuals** are the difference between bin contents and expected bin contents. Most convenient for analysis are the normalized residuals.
- If the hypothesis of identity is valid then **normalized residuals** are approximately independent and identically distributed random variables having **N(0,1) distribution**.
- Analysis of residuals expect test of above mentioned properties of residuals. Notice that indirectly the analysis of residuals increase the power of #chi^{2} test.

References:
http://arxiv.org/pdf/0905.4221v3.pdf
http://arxiv.org/pdf/1309.4649.pdf
https://indico.cern.ch/event/217511/contribution/35/attachments/349287/486926/compar.pdf
http://cds.cern.ch/record/1116584/files/ACAT-060.pdf?version%3D1?ln=bg
http://arxiv.org/pdf/physics/0605123v1.pdf
https://root.cern.ch/root/html/tutorials/math/chi2test.C.html