Monte Carlo generation

- These days lots of black box tools to generate random numbers according to predefined distributions, e.g., Gaussian, etc.
- Still useful to go over some basic principles
- First: everything is based on pseudo-random number generators
  - Computer algorithms to give sequences of (pseudo) random numbers, most simply uniform between 0 and 1
  - Need a starting point ("seed") most often an integer that is used by the random number package to start the sequence
  - Good practice to specify the seed to get repeatable results
  - Use different seeds for different runs (if needed)
  - Never ever ever use irreproducible seeds (eg: time-of-day) in reconstruction codes
    - I have seen examples.....

- Want to generate a variable *x* distributed according to *f(x)dx*
- Pick random number *R* uniform btw 0 an 1. *p(R)dR = dR*
- What should *x(R)* be such that *x* is distributed as *f(x)dx* ?

$$p(R)dR = dR = \frac{dR}{dx}\,dx$$

- So: $f(x) = \frac{dR}{dx}$ $\qquad R = \int_{-\infty}^{x} f(u)du$
- Define the Cumulative Distribution Function $F(x) = \int_{-\infty}^{x} f(u)du$
- Gives:

$$x(R) = F^{-1}(R)$$

If you can invert the CDF, $x(R) = F^{-1}(R)$ has pdf $f(x)dx$

Examples

- Exponential btw 0 an infinity: $f(x) = \frac{1}{\lambda}\, e^{-\frac{x}{\lambda}}$ → $x = -\lambda\, logR$
- Exponential btw x_1 and x_2: $x = -\lambda \log(\left(e^{-\frac{x_2}{\lambda}} - e^{-\frac{x_1}{\lambda}}\right) R + e^{-\frac{x_2}{\lambda}})$

More Examples of Direct Method

$f(x)$	Range	Solution
$\frac{n-1}{x^n}$, $n>1$	$0,\infty$	$x = R^{\frac{1}{1-n}}$
$\frac{1}{x^2}$	$0,\infty$	$x = 1/R$
$(n+1)x^n$ $n \neq -1$	$0,1$	$x = R^{\frac{1}{n+1}}$
$3x^2$	$0,1$	$x = \sqrt{R}$
$\frac{\sin\theta}{2}$	$0,\pi$	$\cos\theta = 1-2R$
$\frac{1}{\pi}\frac{\Gamma/2}{(E-E_0)^2+(\Gamma/2)^2}$	$-\infty,\infty$	$E = E_0 + \frac{\Gamma}{2}\tan\left[\pi\left(R-\frac{1}{2}\right)\right]$

1. Normal Distribution

$$f(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Can be generated in pairs

$$\int f(x) f(y) = \frac{1}{2\pi} \iint e^{-\frac{x^2+y^2}{2}} dx\,dy = \int_0^\infty r e^{-r^2/2} dr \int \frac{d\phi}{2\pi}$$

for $\mu = 0, \sigma = 1$

This can be integrated and inverted to give

$$r = \sqrt{-2 \ln R_1}$$

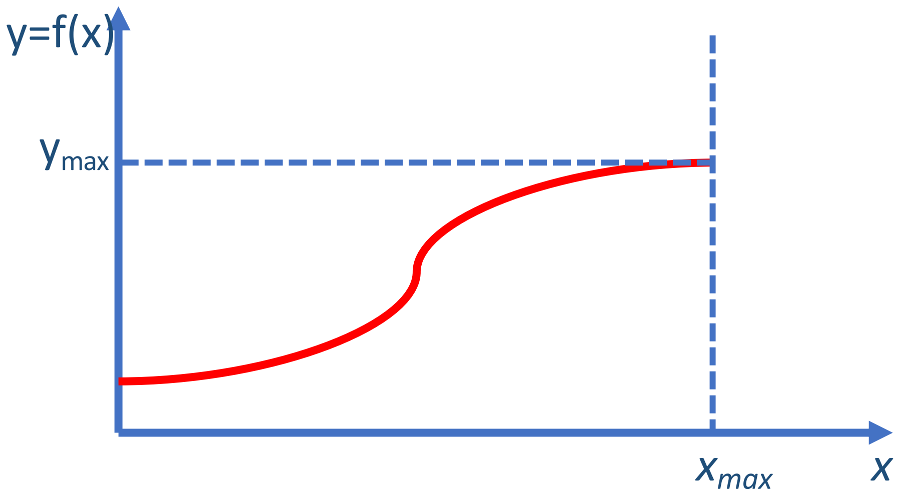
$$\phi = 2\pi R_2$$

and $R_{G1} = \mu + \sigma r \sin\phi$

$R_{G2} = \mu + \sigma r \cos\phi$

R_{G1} and R_{G2} are independent gaussian random numbers.

What if you cannot invert F(x)



Acceptance-rejection method

1. Choose $x=R_1 * x_{max}$
2. Choose $y=R_2 * y_{max}$
3. Keep if $f(x)<y$

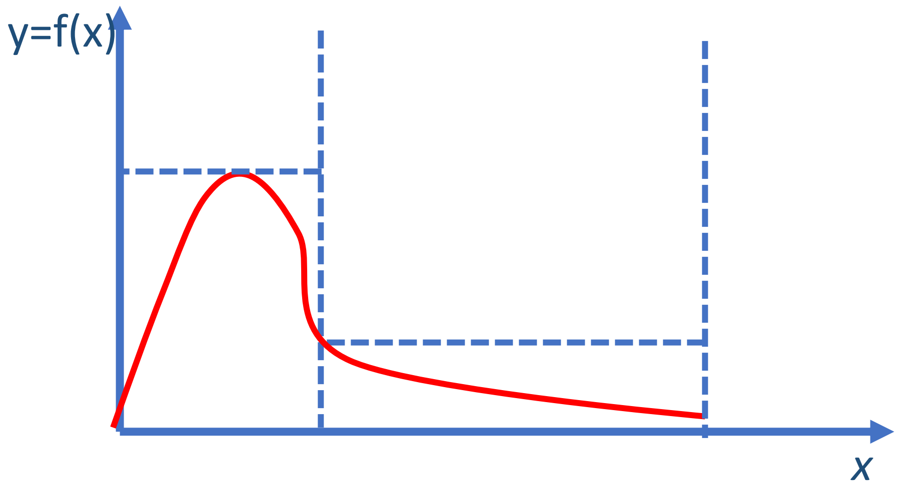
Some inefficiency is inevitable

Efficiency = $\frac{1}{x_{max}} \frac{1}{y_{max}} \int_0^{x_{max}} f(x)dx$

To minimize inefficiency, can break it up into two or more x-regions

Method well suited for histograms.

Does not work if x is unbounded



c. Combination of direct and Acceptance–Rejection Methods

Example: $f(x) = A\, x^{1/2} e^{-3x/2}$

$0 \le x < \infty$

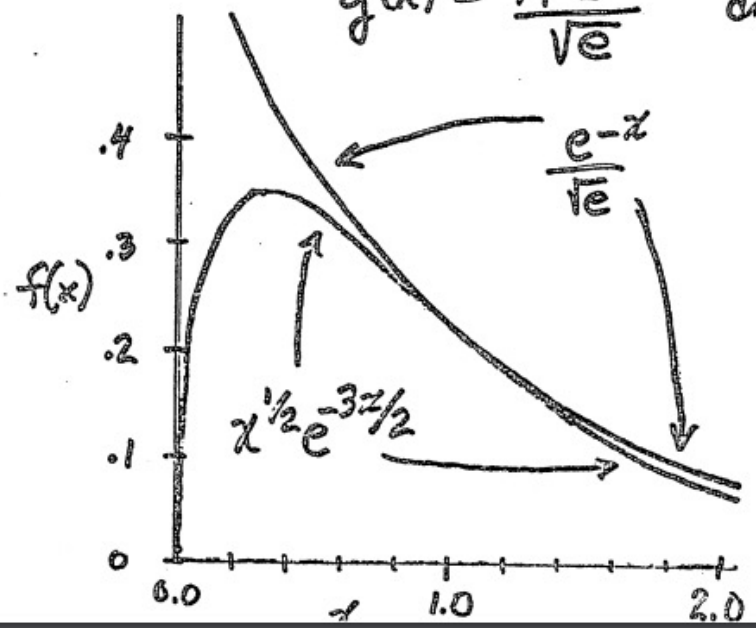
The direct method would be difficult at best (I don't know how.)

Simple Acceptance–Rejection is impossible

Look for a function with a finite area that is always larger than $f(x)$

In this case

$g(x) = \dfrac{A e^{-x}}{\sqrt{e}}$ does this



Procedure:

1. Generate x according to $f(x) = e^{-x}$, $x = -\ln R_1$
2. Keep this if $f(x) > \dfrac{R_2 e^{-x}}{\sqrt{e}}$

The efficiency of this method (fraction of attempts accepted)

is $\sqrt{e} A = \dfrac{\sqrt{e\pi}}{2\,(3/2)^{3/2}} = 0.795$

Markov Chain Monte Carlo

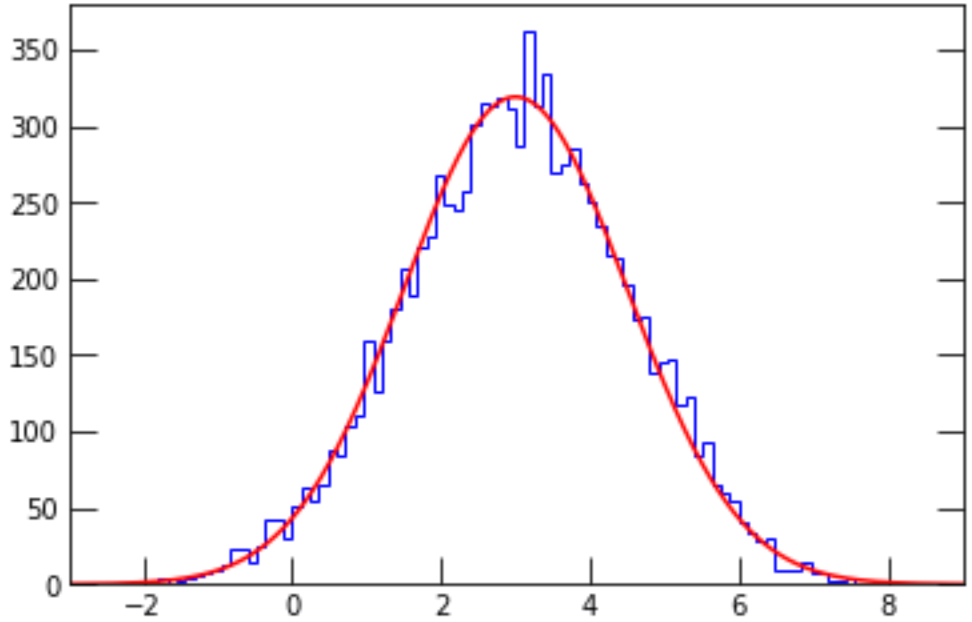
A sequence $\overrightarrow{x_1}, \overrightarrow{x_2}, \ldots.. \overrightarrow{x_N}$ where the probability of $\overrightarrow{x_{N+1}}$ only depends on $\overrightarrow{x_N}$

- A random walk is an example
- Can be multidimensional

Here is how one can generate a sequence of $\overrightarrow{x_i}$ according to a pdf $f(\vec{x})$

1. Pick arbitrary $\overrightarrow{x_1}$
2. Decide on $\overrightarrow{x_2}$ based on a proposal.
   - For example, a proposal could be $\overrightarrow{x_2^{prop}} = \overrightarrow{x_1} \pm R\,\vec{\delta}$
     - This is a symmetrical proposal, prob of proposing 2 from 1 is the same as proposing 1 from 2
     - Metropolis conditions
3. Calculate $p = \min(1, \frac{f\left(\overrightarrow{x_2^{prop}}\right)}{f(\overrightarrow{x_1})})$
4. Throw a random number btw 0 and 1 to accept the proposal with probability *p.*
   - If proposal is accepted, then $\overrightarrow{x_2} = \overrightarrow{x_2^{prop}}$ Otherwise $\overrightarrow{x_2} = \overrightarrow{x_1}$
5. Rinse and repeat

- After many trials the sequence can be shown to be a good sampling of $f(\vec{x})$
- There is an arbitrariness about starting point. Should be in bulk of pdf.
- Typically throw away the first few members of the chain ("burn in")
- Here are the results of a toy 1D exercise as follows
  - f(x) = Gauss(μ=3, σ=1.5)
  - Proposal: uniform random steps between -1 and 1
  - Start at x=0
  - Length of chain = 11K
  - Throw away first 1K



- The histogram is the MC draws
- The red curve is Gauss(μ=3, σ=1.5)

http://hep.ucsb.edu/people/claudio/Phys250/MarkovChain/mchain.html
http://hep.ucsb.edu/people/claudio/Phys250/MarkovChain/mchain.ipynb

Recall p-value (frequentist) for $\mu S+B$ where we conventionally normalize the signal expectation by a multiplicative factor μ ("signal strength")

- p_μ = prob. of getting an equally significant or more significant result than observed
  - e.g., for counting experiment $p_\mu = P(N_\mu > N_{obs} \mid \mu S + B)$
  - Or for a more complicated test statistics q_μ: $p_\mu = P(q_\mu > q_\mu^{obs} \mid \mu S + B)$
- A frequentist 95% upper limit on μ is set where $p_\mu = 0.05$
- When the background fluctuates very low, can exclude at 95% CL very small μ. Even μ=0.
- Technically correct (frequentist). But not desirable. Enter CL_s

$$\mathrm{CL_S}(\mu) \equiv \frac{\mathrm{CL_{S+B}}}{\mathrm{CL_B}} \equiv \frac{p(q_\mu \geq q_\mu^{\mathrm{obs}} \mid \mu\mathrm{S+B})}{p(q_\mu \geq q_\mu^{\mathrm{obs}} \mid \mathrm{B})}$$

- Sometimes written as $CL_S = \frac{p_\mu}{1-p_B}$ where p_B is the p-value for the background only (I find this notation confusing).
- A 95% limit is then set where $CL_S(\mu) = 0.05$
- Note $CL_S(\mu=0) = 1$. Can never exclude $\mu=0$.
- The strictly frequentist p-value gets renormalized.
- $CL_S(\mu)$ is a decreasing function of μ

Test statistics and calculation of CLs

- The current convention is that the test statistics is based on the profile likelihood, but the devil is in the details

$$q_\mu^{\mathrm{LEP}} = -2\log\frac{L(\mathrm{data}|\mu{=}0)}{L(\mathrm{data}|\mu)}$$

LEP style:

- Nuisances fixed to nominal value

$$q_\mu^{\mathrm{TEV}} = -2\log\frac{L(\mathrm{data}|\mu{=}0,\hat{\theta}_0)}{L(\mathrm{data}|\mu,\hat{\theta}_\mu)}$$

Tevatron style:

- Nuisances profiled for μ and μ=0

$$q_\mu^{\mathrm{LHC}} = -2\log\frac{L(\mathrm{data}|\mu,\hat{\theta}_\mu)}{L(\mathrm{data}|\hat{\mu},\hat{\theta})}$$

LHC style:

- Nuisances profiled for μ and $\hat{\mu}$
- Constraint $\mu > 0$
- q_μ=0 when $\hat{\mu} > \mu$

Test statistics and calculation of CLs

- The current convention is that the test statistics is based on the profile likelihood, but the devil is in the details
- Generation of toy Monte Carlo to get the p-values

$$q_\mu^{\mathrm{LEP}} = -2\log\frac{L(\mathrm{data}|\mu{=}0)}{L(\mathrm{data}|\mu)}$$

$$q_\mu^{\mathrm{TEV}} = -2\log\frac{L(\mathrm{data}|\mu{=}0,\hat{\theta}_0)}{L(\mathrm{data}|\mu,\hat{\theta}_\mu)}$$

$$q_\mu^{\mathrm{LHC}} = -2\log\frac{L(\mathrm{data}|\mu,\hat{\theta}_\mu)}{L(\mathrm{data}|\hat{\mu},\hat{\theta})}$$

LEP style:

- Nuisances fixed to nominal value
- Toys: nuisances randomized according to their pdfs

Tevatron style:

- Nuisances profiled for μ and μ=0
- Toys: fixed to post-fit value on μ

LHC style:

- Nuisances profiled for μ and $\hat{\mu}$
- Constraint $\mu > 0$
- q_μ=0 when $\hat{\mu} > \mu$
- Toys: fixed to post fit value on μ

Test statistics and calculation of CLs

$$q_\mu^{\rm LHC} = -2\log\frac{L(\text{data}|\mu,\hat{\theta}_\mu)}{L(\text{data}|\hat{\mu},\hat{\theta})}$$

LHC style:

- Nuisances profiled for μ and $\hat{\mu}$
- Constraint $\mu > 0$
- **q_μ=0 when $\hat{\mu} > \mu$**

The reason for setting $q_\mu = 0$ for $\hat{\mu} > \mu$ is that when setting an upper limit, one would not regard data with $\hat{\mu} > \mu$ as representing less compatibility with μ than the data obtained, and therefore this is not taken as part of the rejection region of the test. That is, the upper limit is obtained by testing μ against the alternative hypothesis consisting of lower values of μ. From the definition of the test statistic one sees that higher values of q_μ represent greater incompatibility between the data and the hypothesized value of μ.

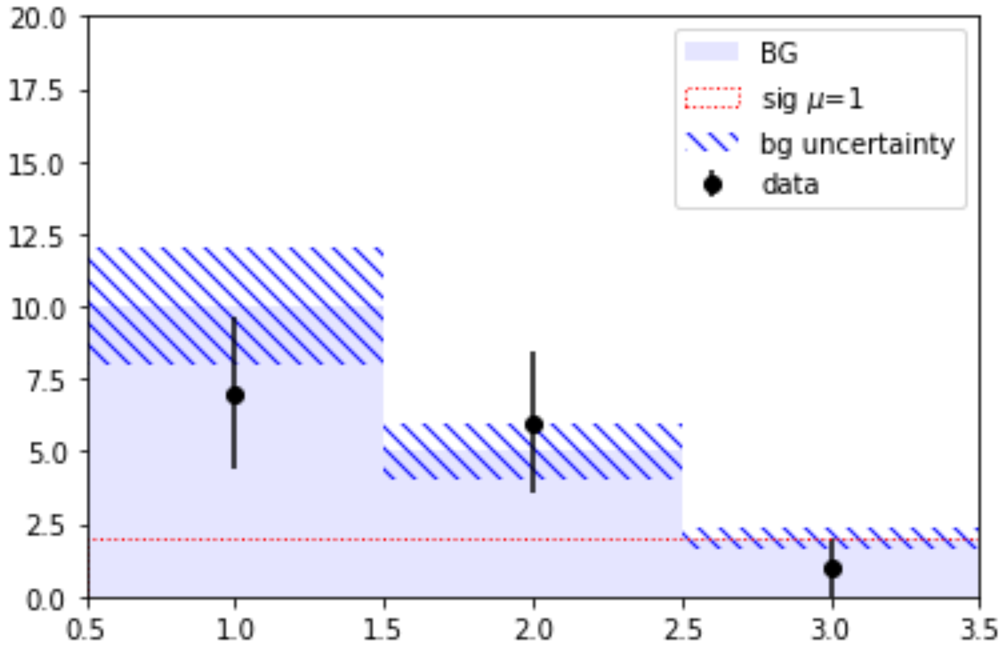
One should note that q_0 is not simply a special case of q_μ with $\mu = 0$, but rather has a different definition (see (12) and (14)). That is, q_0 is zero if the data fluctuate downward ($\hat{\mu} < 0$), but q_μ is zero if the data fluctuate upward ($\hat{\mu} > \mu$). With that caveat in mind, we will often refer in the following to q_μ with the idea that this means either q_0 or q_μ as appropriate to the context.

https://link.springer.com/article/10.1140%2Fepjc%2Fs10052-011-1554-0

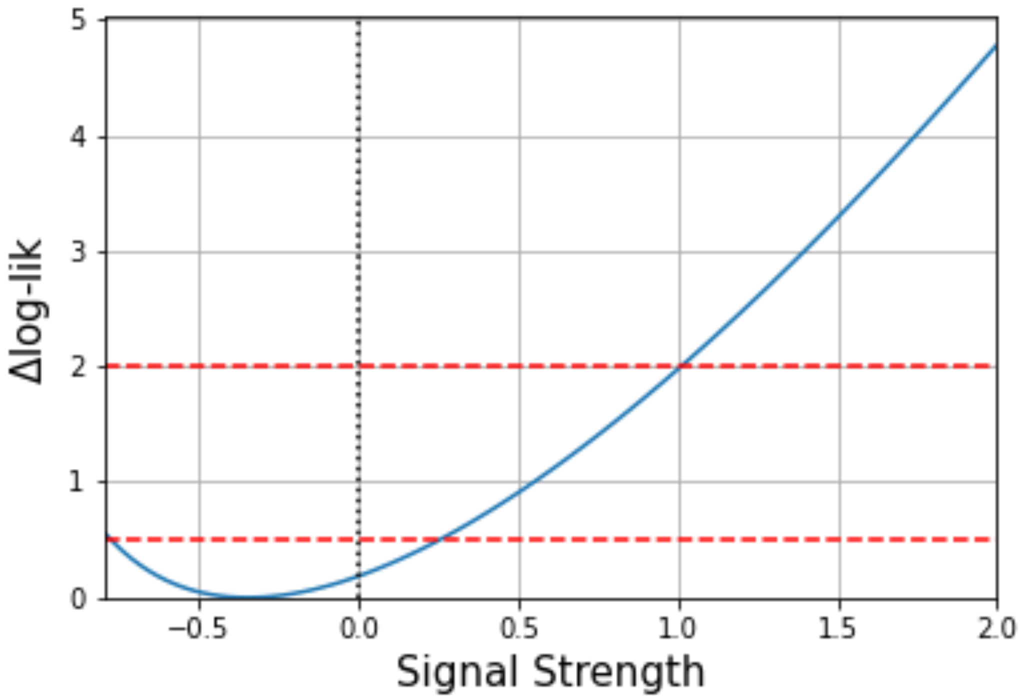
Example

http://hep.ucsb.edu/people/claudio/Phys250/CLs/Toy_v4.html
http://hep.ucsb.edu/people/claudio/Phys250/CLs/Toy_v4.ipynb

```
# This is some fake data in 3 bins
data = np.array([ 7,  6, 1])  # observed
sig  = np.array([ 2,  2, 2])  # signal predicted with mu=1
bg   = np.array([10,  5, 2])  # bg predicted
err  = 0.2*bg                 # bg uncertainty
```



Fits to negative signal strength:



Procedure to get the limit using the LHC-style CL_S

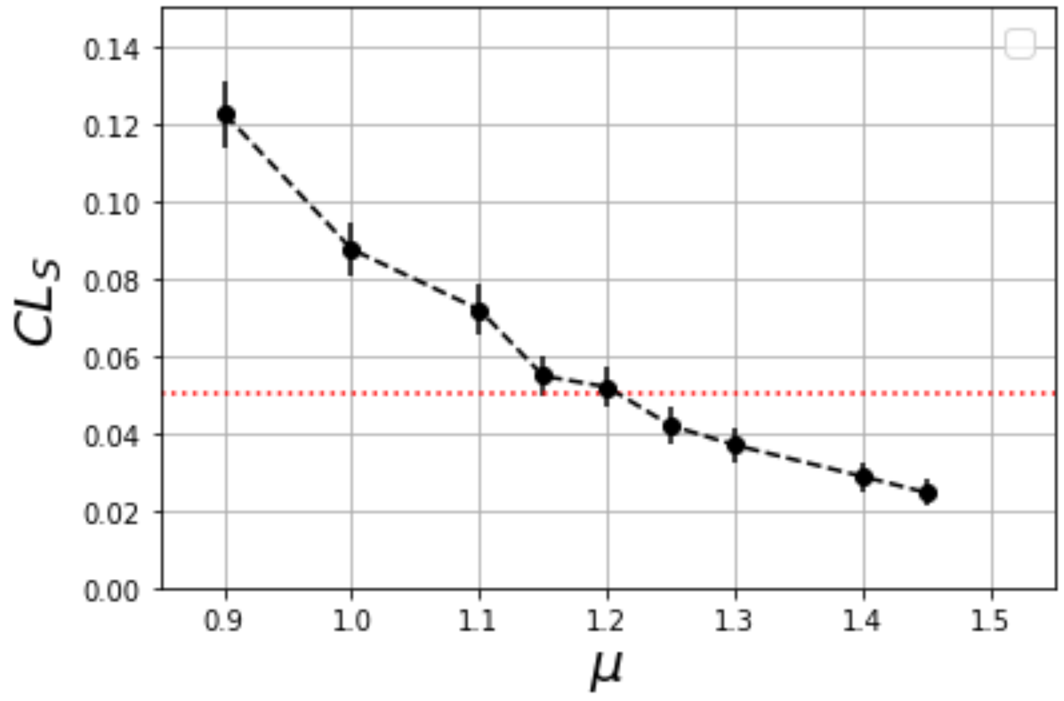
- Scan values of μ. At each μ

1. Fit <u>data</u> to μS+B (μ fixed but > 0)
   - get best fit nuisances (ie: 3 values of background bins)
   - get q_μ^{obs}
2. Generate many signal toys with strength μ (draw from three Poisson…)
3. Generate many background toys using nuisances from (1). (draw from three Poissons..)
4. Add signal and background toys to get many fake-data-with-signal (s+b) toys
5. Fit the fake data toys to μS+B hypothesis
   - μ is now a free parameter (but μ>0)
   - get q_μ^{s+b}
6. Fit the background only toys to the μS+B hypothesis
   - μ is now a free parameter (but μ>0)
   - get q_μ^{b}
7. Calculate $CL_S(\mu)$

$$\mathrm{CL_S}(\mu) = \frac{\mathrm{CL_{S+B}}}{\mathrm{CL_B}} = \frac{q_\mu^{\mathrm{s+b}} \geq q_\mu^{\mathrm{obs}}}{q_\mu^{\mathrm{b}} \geq q_\mu^{\mathrm{obs}}}$$

Procedure to get the limit using the LHC-style CL_S (continued)

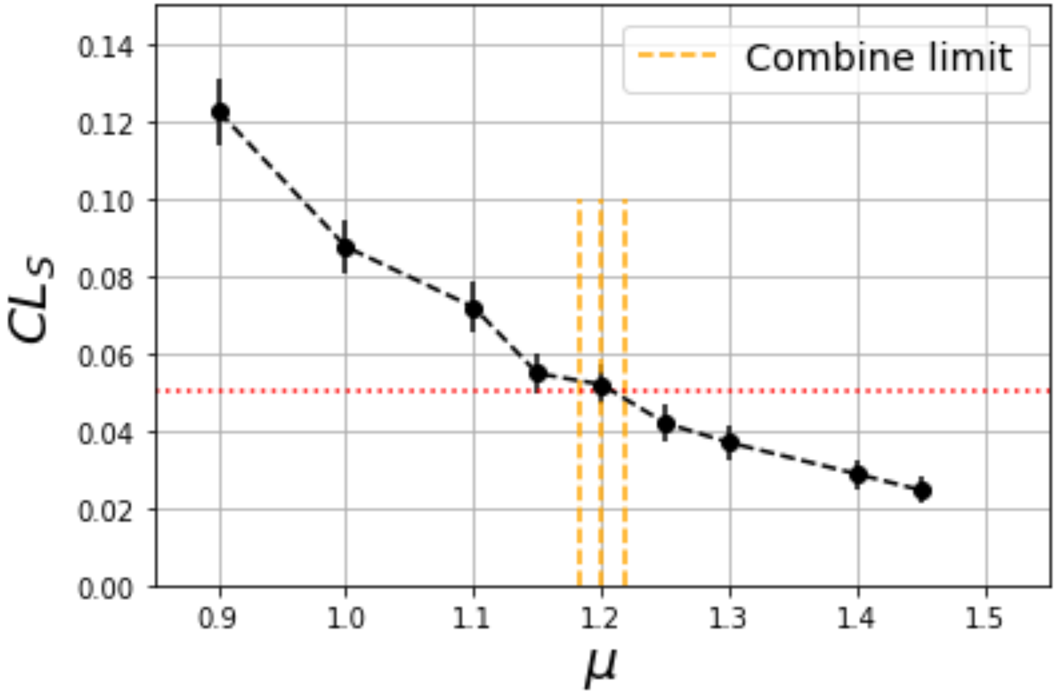
8. Plot $CL_S(\mu)$ and fit it to find the point at which it crosses 0.05



I did not bother to fit it.. It’s around 1.20

Procedure to get the limit using the LHC-style CL_S (continued)

8. Plot $CL_S(\mu)$ and fit it to find the point at which it crosses 0.05

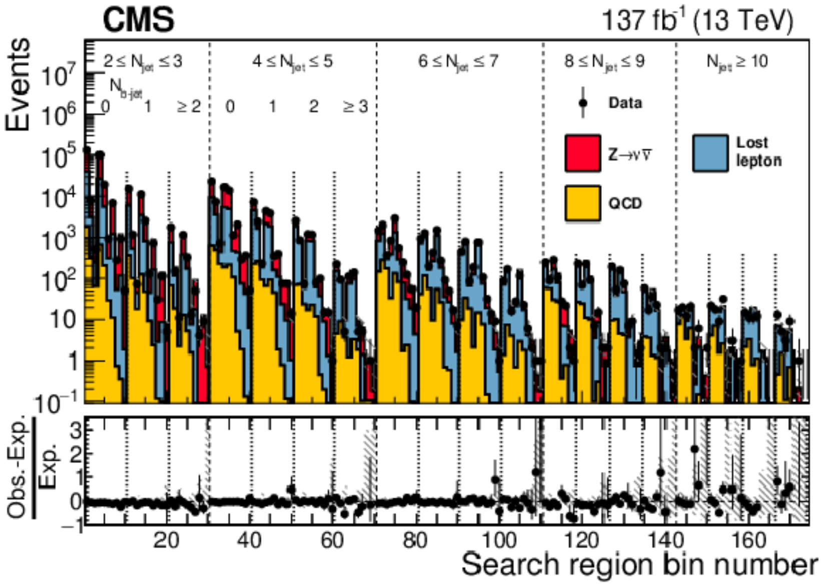


I did not bother to fit it.. It's around 1.20

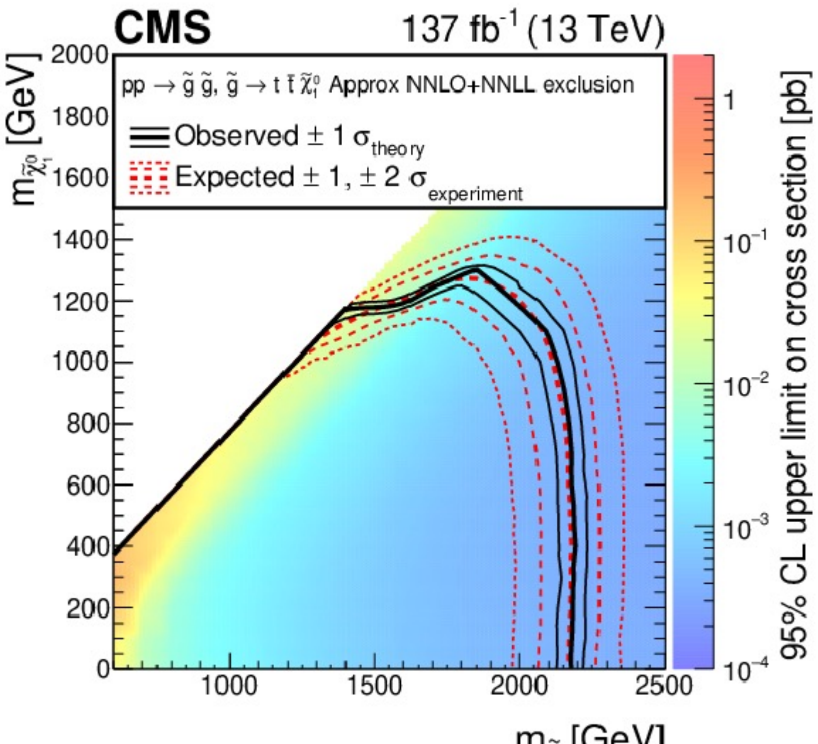
Compare with standard CMS code ("Combine")

```
 -- Hybrid New --
Limit: r < 1.19991 +/- 0.0177488 @ 95% CL
Done in 4.70 min (cpu), 4.70 min (real)
```

Takes a lot of CPU. (This was from C++ not python)
This was a simple case.
Often need to check many different signal models
Possibly 100's of signal regions
Possibly 100's of nuisances



174 signal regions



A dozen (about) 2D models

Toy Monte Carlo limits are most often not practical

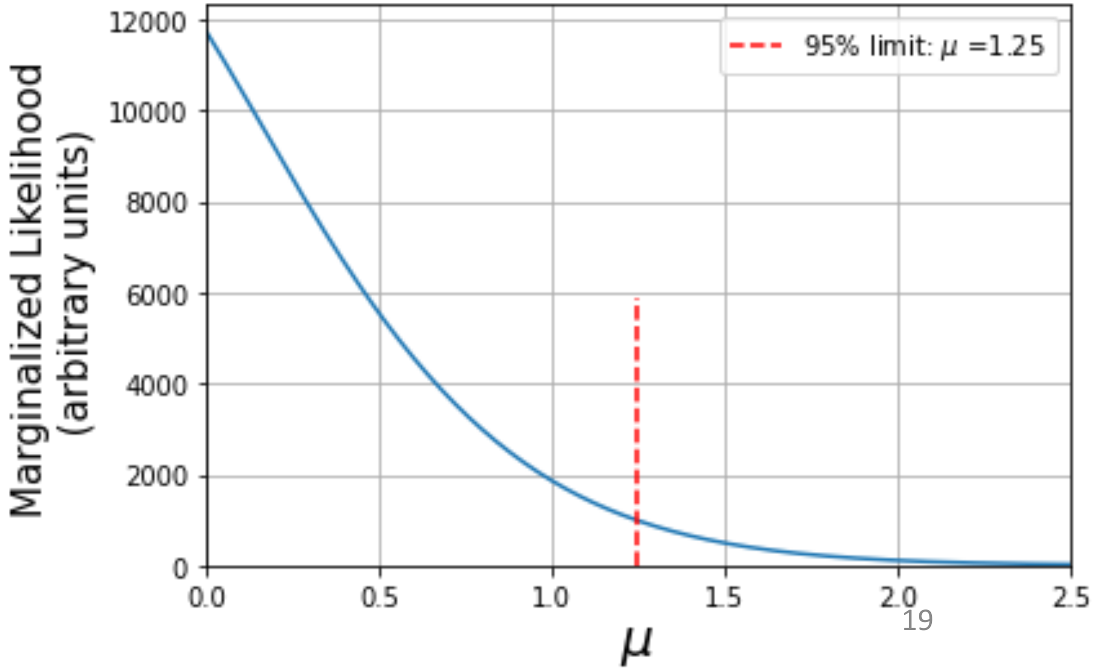
An alternative is to go Bayesian, with flat prior

- Remember Bayesian: marginalize the likelihood by integrating over the nuisance parameters. Multiply by the prior (in this case $\Theta(\mu=0)$) to get a (up to a normalization constant) a posterior pdf $p(\mu)d\mu$
- $p(\mu) \propto \Theta(\mu = 0) \int d\vec{\theta}\; L(data|\mu, \vec{\theta})\, p(\vec{\theta})$
- It is convenient to do the integration using MC methods

$$\int f(x)p(x)dx = \frac{1}{N}\sum f(x_i)$$

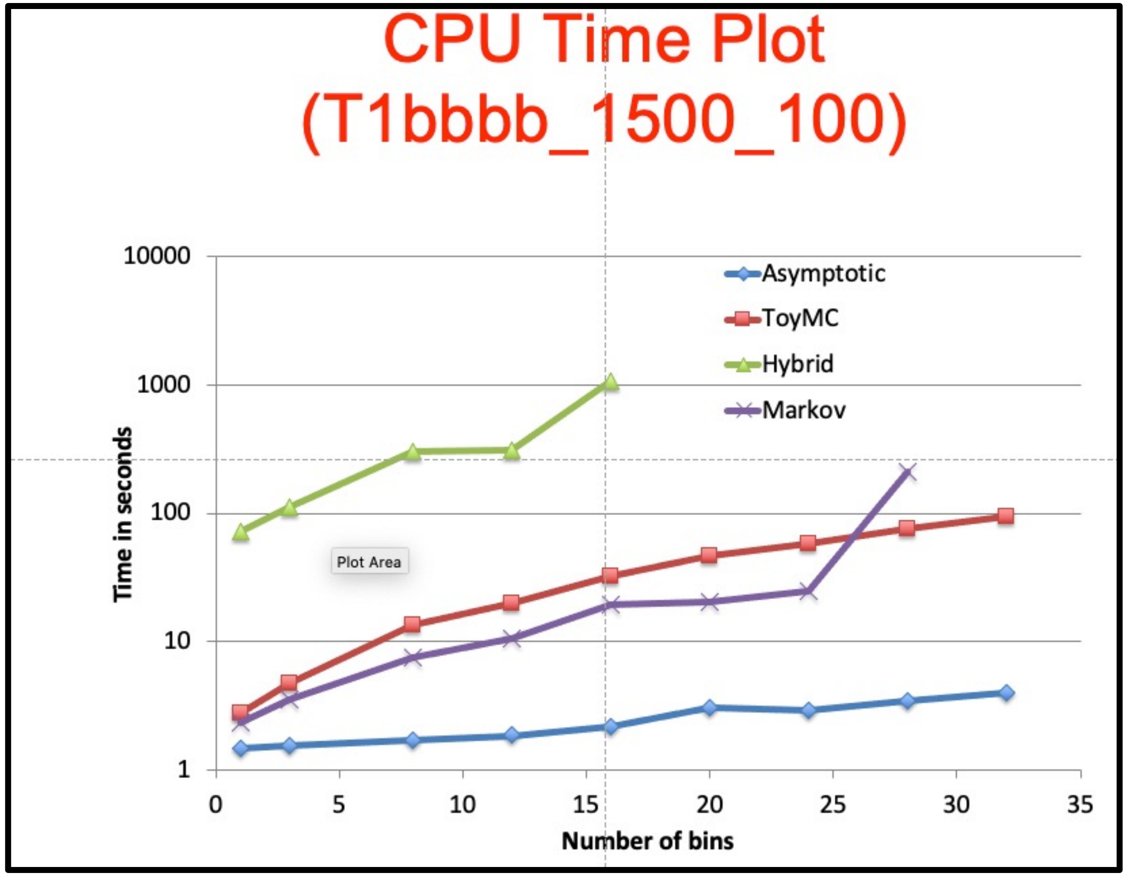
Where x_i are N values picked according to *p(x)dx*

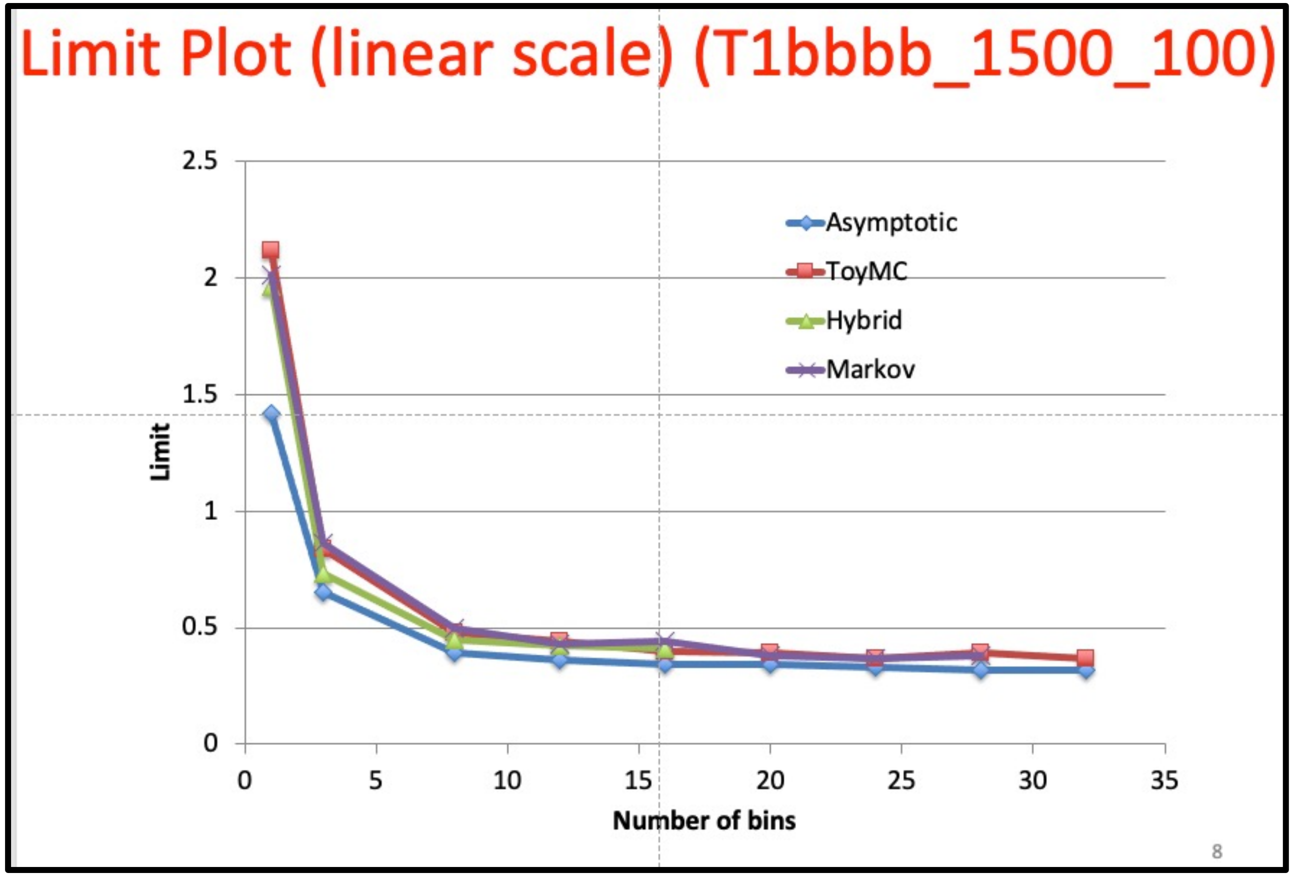
Compare with CL_S limit from toys of μ=1.20



http://hep.ucsb.edu/people/claudio/Phys250/CLs/BennetToy_Bayesian.html
http://hep.ucsb.edu/people/claudio/Phys250/CLs/BennetToy_Bayesian.ipynb

From a study I did ~ 6 years ago using standard (at the time) CMS tools





- “ToyMC” and “Markov” were different implementation of Bayesian limits
- “Hybrid” is CL_S using toy Monte Carlo
- “Asymptotic” is the asymptotic approximation to CL_S

Asymptotic CLs

Eur. Phys. J. C (2011) 71: 1554
DOI 10.1140/epjc/s10052-011-1554-0

THE EUROPEAN PHYSICAL JOURNAL C

Special Article - Tools for Experiment and Theory

Asymptotic formulae for likelihood-based tests of new physics

Glen Cowan[1], **Kyle Cranmer**[2], **Eilam Gross**[3], **Ofer Vitells**[3,a]

[1]Physics Department, Royal Holloway, University of London, Egham TW20 0EX, UK
[2]Physics Department, New York University, New York, NY 10003, USA
[3]Weizmann Institute of Science, Rehovot 76100, Israel

https://link.springer.com/article/10.1140%2Fepjc%2Fs10052-011-1554-0

Practical Statistics for the LHC

https://arxiv.org/pdf/1503.07622.pdf

Kyle Cranmer
Center for Cosmology and Particle Physics, Physics Department, New York University, USA

Asymptotic limits are based on the fact that the profile likelihood ratio

$$q_\mu = -2\log\lambda(\mu) = -2\log\frac{L(\mathrm{data}|\mu,\hat{\hat{\theta}}_\mu)}{L(\mathrm{data}|\hat{\mu},\hat{\theta})}$$

is asymptotically

$$q_\mu = \frac{(\mu-\hat{\mu})^2}{\sigma^2} + O(1/\sqrt{N})$$

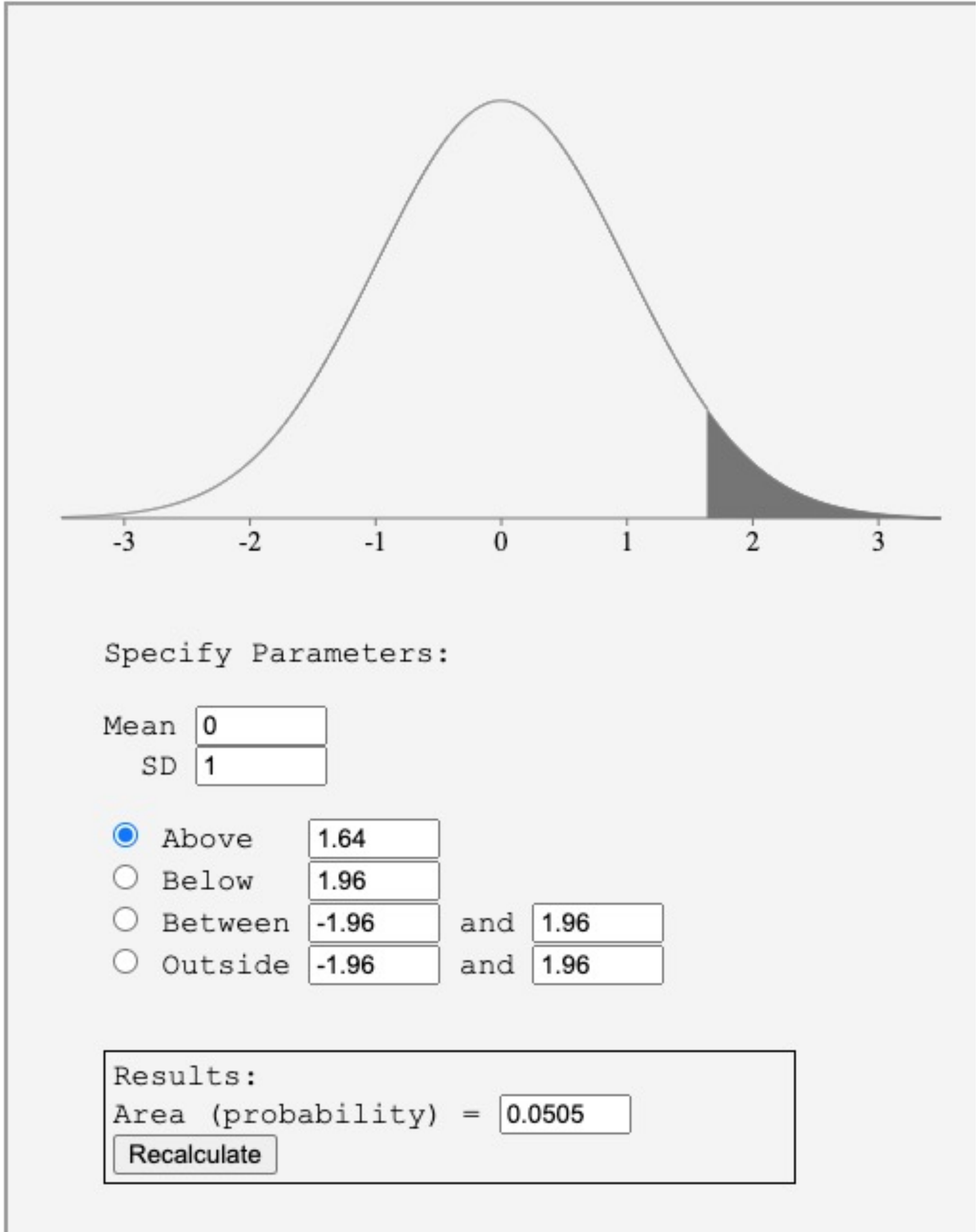
i.e. distributes as a chi-squared with one dof. With the (peculiar) LHC definition of q_μ

$$\tilde{q}_\mu = \begin{cases} \frac{\mu^2}{\sigma^2} - \frac{2\mu\hat{\mu}}{\sigma^2}, & \hat{\mu} < 0, \\ \frac{(\mu-\hat{\mu})^2}{\sigma^2}, & 0 \le \hat{\mu} \le \mu, \\ 0, & \hat{\mu} > \mu. \end{cases}$$

Then the 95% upper limit on μ is still given by this simple formula (remarkably, because ythe last equation on the previous page is not simple)

$$\mu_{up} = \hat{\mu} + \sigma\phi^{-1}(1 - \alpha)$$

Where Φ^{-1} is the quantile (inverse of the cumulative distribution) of the Gaussian and α=0.05. (ie $\phi^{-1}(1 - \alpha)$ = 1.64)


-3
-2
-1
0
1
2
3
Specify Parameters:
Mean 0
SD 1
Above 1.64
Below 1.96
Between -1.96 and 1.96
Outside -1.96 and 1.96
Results:
Area (probability) = 0.0505
Recalculate

Then the 95% upper limit on μ is still given by this simple formula (remarkably, because ythe last equation on the previous page is not simple)

$$\mu_{up} = \hat{\mu} + \sigma\phi^{-1}(1-\alpha)$$

Where Φ^{-1} is the quantile (inverse of the cumulative distribution) of the Gaussian and α=0.05. (ie $\phi^{-1}(1-\alpha)$ = 1.64)

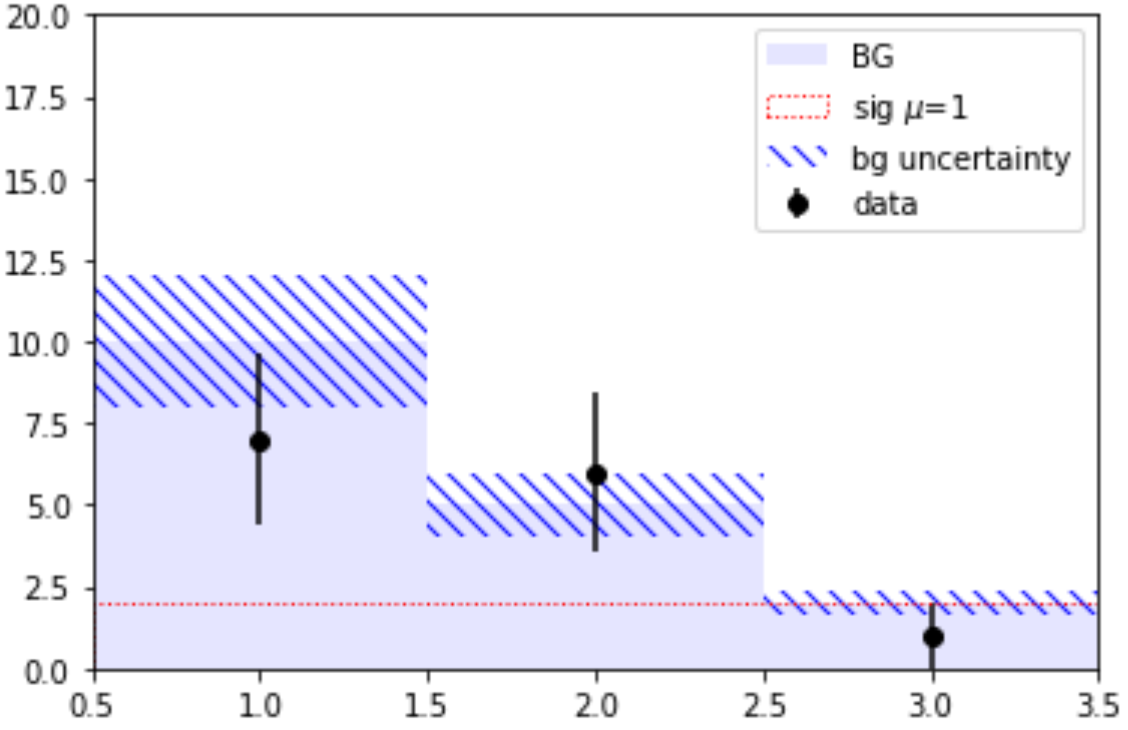
The value of σ can be estimated by the so-called “Asimov data set”, ie, a (binned) data set where the number of events in each bin is exactly the number of events expected in each bin.

- Note that then this number is not necessarily integer, but that’s OK

Then one can write down an Asimov likelihood and calculate the variance of μ by taking 2nd derivatives. Or alternatively, use the equation $q_\mu = \frac{(\mu-\hat{\mu})^2}{\sigma^2}$ for the Asimov data set.

Summary of Results for our example

- CL_S with toys $\mu < 1.20 \pm 0.02$
- Asymptotic CL_S $\mu < 1.14$
- Bayesian with flat prior $\mu < 1.25$



Full disclosure: I did not code the asymptotic, just used CMS tools