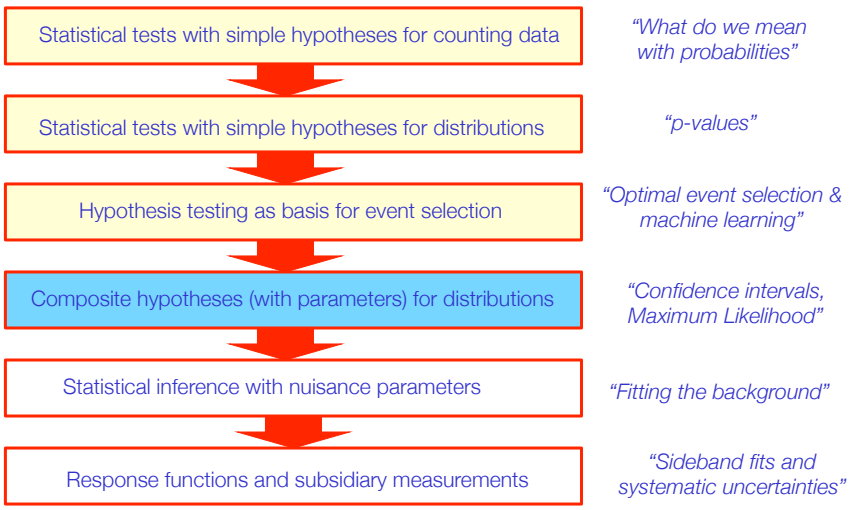
3 *'Composite hypotheses'*

Wouter Verkerke, NIKHEF

Roadmap for this course

- Tomorrow we with start with *hypothesis with parameters*



Introduce concept of composite hypotheses

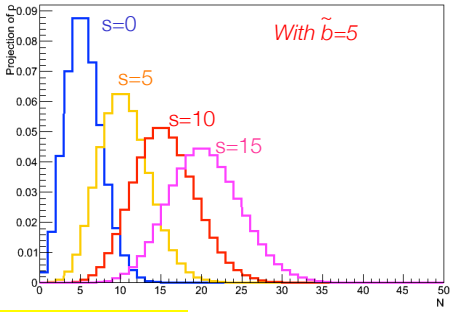
- In most cases in physics, a hypothesis is not “simple”, but “composite”
- **Composite hypothesis** = Any hypothesis which does *not* specify the population distribution completely
- Example: counting experiment with signal and background, that leaves signal expectation unspecified

Simple hypothesis

$$L = Poisson(N \mid \tilde{s} + \tilde{b})$$

$$L(s) = Poisson(N \mid s + \tilde{b})$$

Composite hypothesis



(My) notation convention: all symbols with ~ are constants

A common convention in the meaning of model parameters

- A common convention is to recast signal rate parameters into a normalized form (e.g. w.r.t the Standard Model rate)

Simple hypothesis

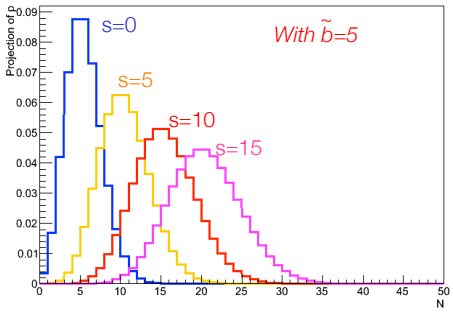
$$L = Poisson(N \mid \tilde{s} + \tilde{b})$$

$$L(s) = Poisson(N \mid s + \tilde{b})$$

Composite hypothesis

$$L(\mu) = Poisson(N \mid \mu \cdot \tilde{s} + \tilde{b})$$

Composite hypothesis
with normalized rate parameter



‘Universal’ parameter interpretation makes it easier to work with your models

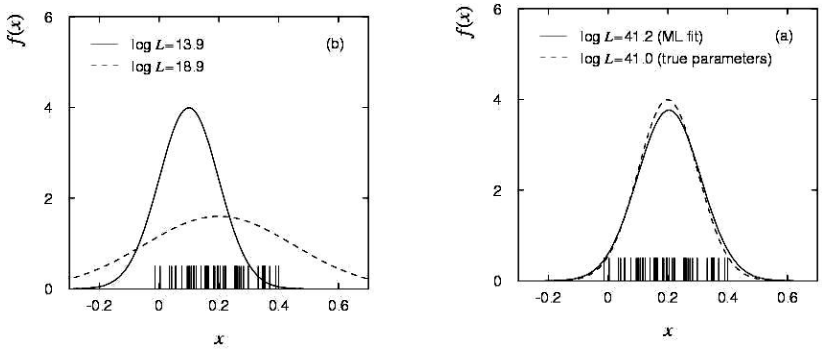
μ=0 → no signal
μ=1 → expected signal
μ>1 → more than expected signal

What can we do with composite hypothesis

- With simple hypotheses – inference is restricted to making statements about P(D|hypo) or P(hypo|D)
- With composite hypotheses – many more options
- 1 Parameter estimation and variance estimation
  - What is value of ***s*** for which the observed data is most probable?
  - What is the variance (std deviation squared) in the estimate of ***s?*** } s=5.5 ± 1.3
- 2 Confidence intervals
  - Statements about model parameters using frequentist concept of probability
  - s<12.7 at 95% confidence level
  - 4.5 < s < 6.8 at 68% confidence level
- 3 Bayesian credible intervals
  - Bayesian statements about model parameters
  - s<12.7 at 95% credibility

Parameter estimation using Maximum Likelihood

- Likelihood is high for values of p that result in distribution similar to data



- Define the maximum likelihood (ML) estimator to be the procedure that finds the parameter value for which the likelihood is maximal.

Parameter estimation – Maximum likelihood

- Practical estimation of maximum likelihood performed by minimizing the negative log-Likelihood

$$L(\vec{p}) = \prod_i f(\vec{x}_i;\vec{p})$$

$$-\ln L(\vec{p}) = -\sum_i \ln F(\vec{x}_i;\vec{p})$$

  - Advantage of log-Likelihood is that contributions from events can be summed, rather than multiplied (computationally easier)

- In practice, find point where derivative of –logL is zero

$$\left.\frac{d\ln L(\vec{p})}{d\vec{p}}\right|_{p_i=\hat{p}_i} = 0$$

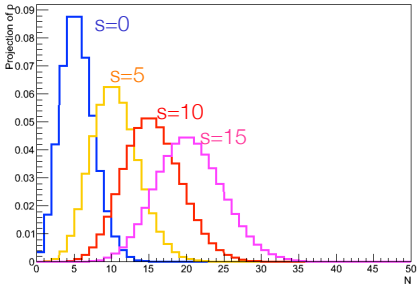
- Standard notation for ML estimation of p is $\hat{p}$

Example of Maximum Likelihood estimation

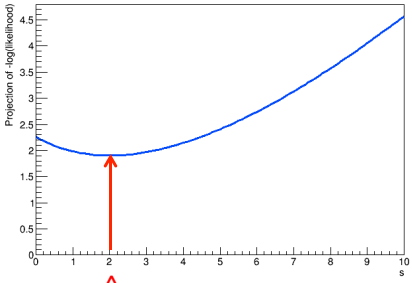
- Illustration of ML estimate on Poisson counting model

$$L(N \mid s) = Poisson(N \mid s + \tilde{b})$$

-log *L*(*N*|*s*) versus *N* [s=0,5,10,15]



-log *L*(*N*|*s*) versus *s* [N=7]



$\hat{s}=2$

- Note that Poisson model is discrete in N, *but continuous in s!*

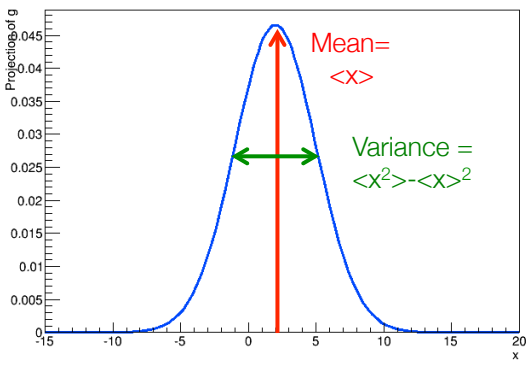
Properties of Maximum Likelihood estimators

- In general, Maximum Likelihood estimators are
  - Consistent (gives right answer for N→∞)
  - Mostly unbiased (bias ∝1/N, may need to worry at small N)
  - Efficient for large N (you get the smallest possible error)
  - Invariant: (a transformation of parameters will Not change your answer, e.g $(\hat{p})^2 = \widehat{(p^2)}$

- MLE efficiency theorem: the MLE will be *unbiased* and *efficient* if an unbiased efficient estimator exists
  - Proof not discussed here
  - Of course this does not guarantee that any MLE is unbiased and efficient for any given problem

Estimating parameter variance

- Note that 'uncertainty' on a parameter estimate is an ambiguous statement
- Can either mean an interval with a stated confidence or credible, level (e.g. 68%), or simply assume it is the square-root of the variance of a distribution



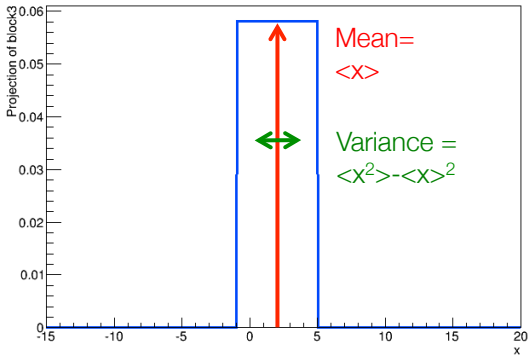
For a Gaussian distribution mean and variance map to parameters for *mean* and $sigma^2$

and interval defined by $\sqrt{V}$ contains 68% of the distribution (='1 sigma' by definition)

Thus for Gaussian distributions all common definitions of 'error' work out to the same numeric value

Estimating parameter variance

- Note that 'error' or 'uncertainty' on a parameter estimate is an ambiguous statement
- Can either mean an interval with a stated confidence or credible, level (e.g. 68%), or simply assume it is the square-root of the variance of a distribution



For other distributions intervals by √V do not necessarily contain 68% of the distribution

Estimating variance on parameters

- Variance on of parameter can also be estimated from Likelihood using the variance estimator

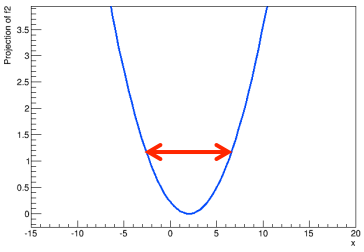
$$\hat{\sigma}(p)^2 = \hat{V}(p) = \left(\frac{d^2 \ln L}{d^2 p} \right)^{-1}$$

From Rao-Cramer-Frechet inequality

$$V(\hat{p}) \geq \frac{1 + \frac{db}{dp}}{\left(\frac{d^2 \ln L}{d^2 p} \right)}$$

b = bias as function of p, inequality becomes equality in limit of efficient estimator

- Valid if estimator is efficient and unbiased!
- Illustration of Likelihood Variance estimate on a Gaussian distribution



$$f(x \mid \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2 \right]$$

$$\ln f(x \mid \mu, \sigma) = -\ln\sigma - \ln\sqrt{2\pi} + \frac{1}{2}\left(\frac{x-\mu}{\sigma} \right)^2$$

$$\left. \frac{d \ln f}{d\sigma} \right|_{x=\mu} = \frac{-1}{\sigma} \quad \Rightarrow \quad \left. \frac{d^2 \ln f}{d^2 \sigma} \right|_{x=\mu} = \frac{1}{\sigma^2}$$

Relation between Likelihood and χ^2 estimators

- Properties of χ^2 estimator follow from properties of ML estimator using *Gaussian probability density functions*

$$F(x_i, y_i, \sigma_i; \vec{p}) = \prod_i \exp\left[-\left(\frac{y_i - f(x_i;\vec{p})}{\sigma_i}\right)^2\right]$$

Gaussian Probability Density Function in p for single measurement y±σ from a predictive function f(x|p)

Take log,
Sum over all points (x_i, y_i, σ_i)

$$-\ln L(\vec{p}) = \tfrac{1}{2}\sum_i \left(\frac{y_i - f(x_i;\vec{p})}{\sigma_i}\right) = \tfrac{1}{2}\chi^2$$

The Likelihood function in p for given points $x_i(s_i)$ and function $f(x_i;p)$

- The χ^2 estimator follows from ML estimator, i.e it is
  - Efficient, consistent, bias 1/N, invariant,
  - But only in the limit that the error **on** x_i is truly Gaussian

What can we do with composite hypothesis

- With simple hypotheses – inference is restricted to making statements about P(D|hypo) or P(hypo|D)
- With composite hypotheses – many more options
- 1 Parameter estimation and variance estimation
  - What is value of ***s*** for which the observed data is most probable?
  - What is the variance (std deviation squared) in the estimate of ***s?***

} s=5.5 ± 1.3

- 2 Confidence intervals
  - Statements about model parameters using frequentist concept of probability
  - s<12.7 at 95% confidence level
  - 4.5 < s < 6.8 at 68% confidence level
- 3 Bayesian credible intervals
  - Bayesian statements about model parameters
  - s<12.7 at 95% credibility

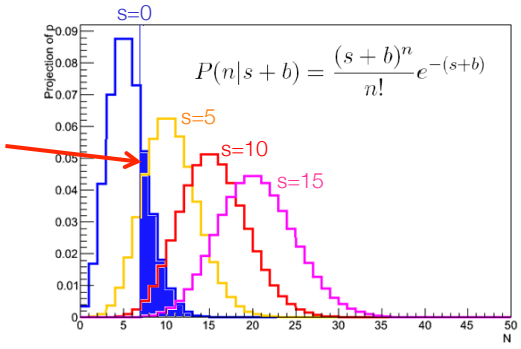
Interval estimation with fundamental methods

- Can also construct parameters intervals using 'fundamental' methods explored earlier (Bayesian or Frequentist)
- Construct Confidence Intervals or Credible Intervals with defined probabilistic meaning, independent of assumptions on normality of distribution (Central Limit Theorem) → "95% C.L."
- With fundamental methods you greater flexibility in types of interval. E.g when no signal observed → usually wish to set an upper limit (construct 'upper limit interval')

Reminder - the Likelihood as basis for hypothesis testing

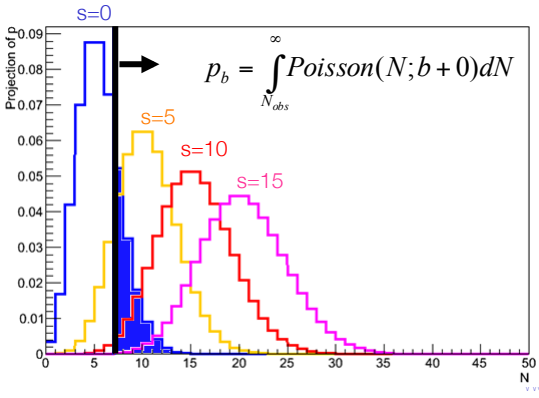
- A probability model allows us to calculate the probability of the observed data under a hypothesis
- This probability is called the Likelihood

P(obs|theo) is called the Likelihood



Reminder - Frequentist test statistics and p-values

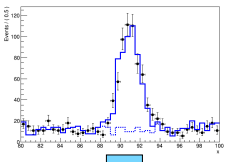
- Definition of 'p-value': *Probability to observe this outcome or more extreme in future repeated measurements is x%,* if hypothesis is true
- Note that the definition of p-value assumes an explicit ordering of possible outcomes in the 'or more extreme' part



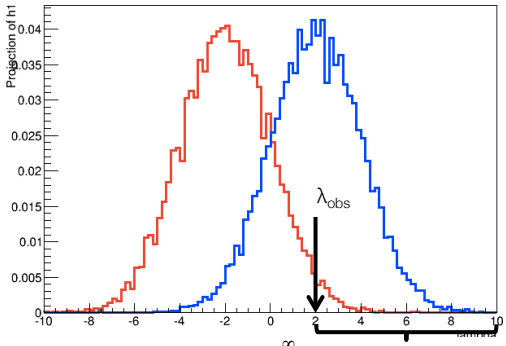
$$p_b = \int_{N_{obs}}^{\infty} Poisson(N; b+0)\,dN \quad (= 0.23)$$

P-values with a likelihood ratio test statistic

- With the introduction of a (likelihood ratio) test statistic, hypothesis testing of models of arbitrary complexity is now reduced to the same procedure as the Poisson example



$$\lambda(\vec{N}) = \frac{L(\vec{N} \mid H_{s+b})}{L(\vec{N} \mid H_b)}$$



$$p - value = \int_{\lambda_{obs}}^{\infty} f(\lambda \mid H_b)$$

- *Except that we generally don't know distribution f(λ)…*

A different Likelihood ratio for composite hypothesis testing

- On *composite hypotheses,* where both null and alternate hypothesis map to values of μ, we can define an alternative likelihood-ratio test statistics that has better properties

'simple hypothesis' 'composite hypothesis' Hypothesis μ that is being tested

$$\lambda(\vec{N}) = \frac{L(\vec{N} \mid H_0)}{L(\vec{N} \mid H_1)} \quad \Rightarrow \quad \lambda_\mu(\vec{N}_{obs}) = \frac{L(\vec{N} \mid \mu)}{L(\vec{N} \mid \hat{\mu})}$$

'Best-fit value'

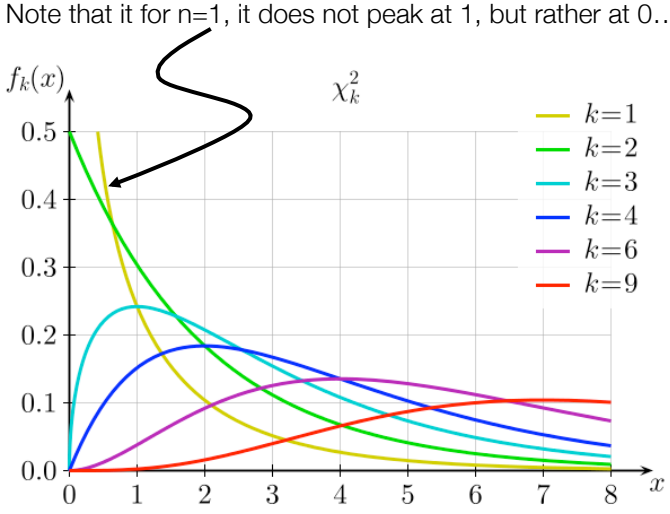
- **Advantage: distribution of new λ_μ has known asymptotic form**
- Wilks theorem: distribution of $-\log(\lambda_\mu)$ is asymptotically distribution as a χ^2 with N_{param} degrees of freedom*

  *Some regularity conditions apply

- → Asymptotically, we can *directly* calculate p-value from λ_μ^{obs}

What does a χ^2 distribution look like for n=1?

- Note that it for n=1, it does not peak at 1, but rather at 0…



Composite hypothesis testing in the asymptotic regime

- For 'histogram example': what is p-value of null-hypothesis

'likelihood assuming zero signal strength'

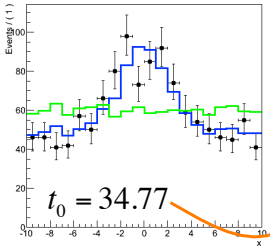
$$t_0 = -2\ln\frac{L(data\,|\,\mu=0)}{L(data\,|\,\hat{\mu})}$$

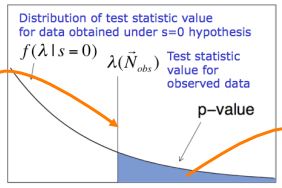
$\hat{\mu}$ is best fit value of μ

'likelihood of best fit'

$-\log\mu$

On signal-like data t_0 is large





Wilks: f(λ|0) → χ^2 distribution

P-value = TMath::Prob(34.77,1)
= 3.7×10^{-9}

Composite hypothesis testing in the asymptotic regime

- For 'histogram example': what is p-value of null-hypothesis

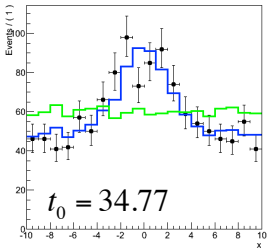
'likelihood assuming zero signal strength'

$$t_0 = -2\ln\frac{L(data\,|\,\mu=0)}{L(data\,|\,\hat{\mu})}$$

$\hat{\mu}$ is best fit value of μ

'likelihood of best fit'

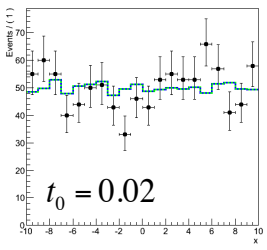
On signal-like data t_0 is large



P-value = TMath::Prob(34.77,1)
= 3.7×10^{-9}

Use Wilks Theorem

On background-like data t_0 is small

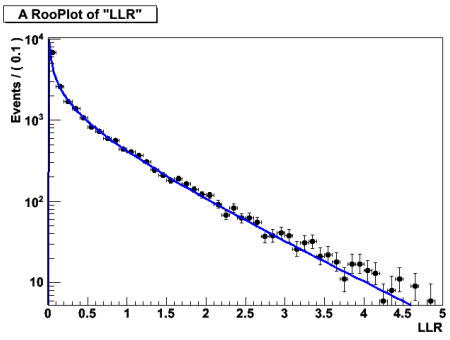


P-value = TMath::Prob(0.02,1)
= 0.88

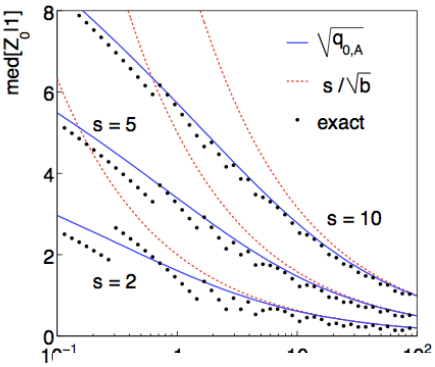
How quickly does $f(\lambda_{\mu}|\mu)$ converge to its asymptotic form

- Pretty quickly –

Here is an example of likelihood function for 10-bin distribution with 200 events



Here is an example for event counting at various s,b



$$\sqrt{q_{0,A}} = \sqrt{2\left((s+b)\ln(1+s/b) - s\right)}\,.$$

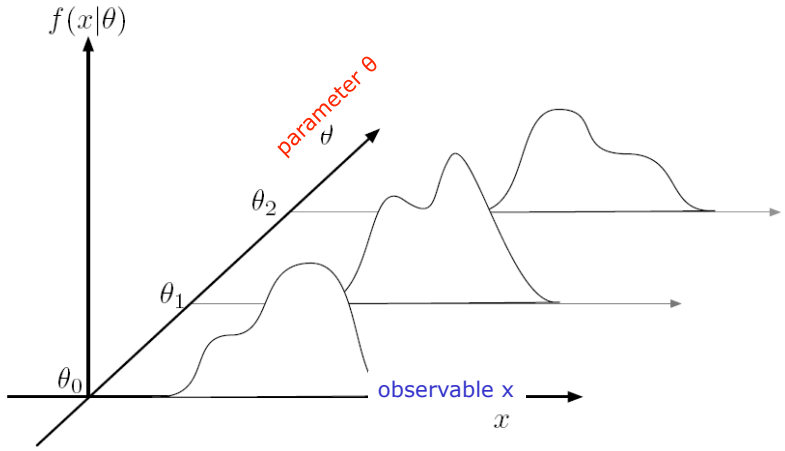
From hypothesis testing to confidence intervals

- Next step for composite hypothesis is to go from p-values for a hypothesis defined by fixed value of μ to *an interval statement on μ*
- Definition: A interval on μ at X% confidence level is defined such that the true of value of μ is contained X% of the time in the interval.
  - Note that the output is *not* a probabilistic statement on the true s value
  - The true μ is fixed but unknown – each observation will result in an estimated interval $[\mu_-,\mu_+]$. X% of those intervals will contain the true value of μ
  - Coverage = guarantee that probabilistic statements is true (i.e. repeated future experiments do reproduce results in X% of cases)
- Definition of confidence intervals does not make any assumption on shape of interval

  → Can choose one-sided intervals ('limits'), two-sided intervals ('measurements'), or even disjoint intervals ('complicated measurements')

Exact confidence intervals – the Neyman construction

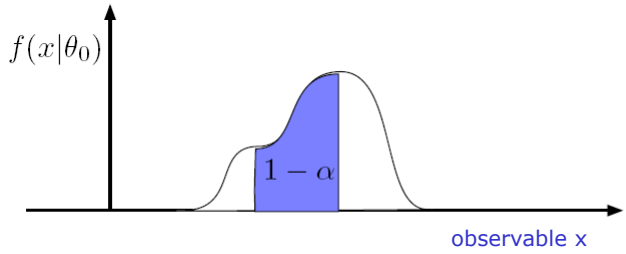
- Simplest experiment: one measurement (x), one theory parameter (θ)
- For each value of parameter θ, determine distribution in in observable x



How to construct a Neyman Confidence Interval

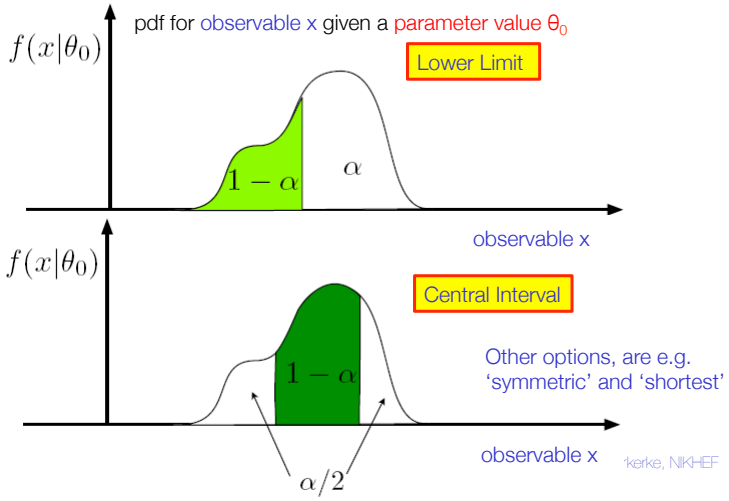
- Focus on a slice in θ
  - For a 1-α% confidence Interval, define ***acceptance interval*** that contains 100%-α% of the distribution

pdf for observable x
given a parameter value θ_0



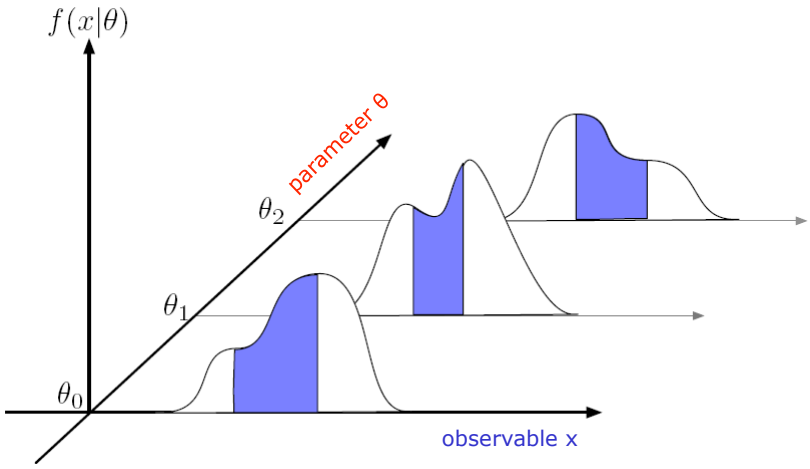
How to construct a Neyman Confidence Interval

- Definition of acceptance interval is not unique
  → Choose shape of interval you want to set here.
  - Algorithm to define acceptance interval is called 'ordering rule'



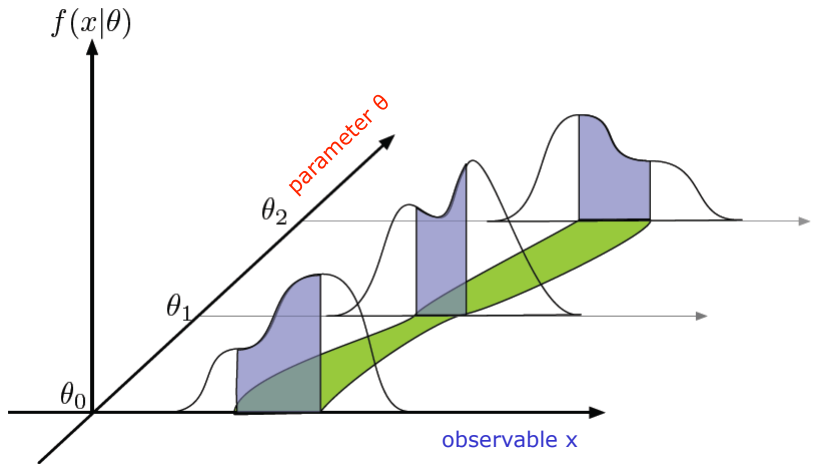
How to construct a Neyman Confidence Interval

- Now make an acceptance interval in observable x
  for each value of parameter θ



How to construct a Neyman Confidence Interval

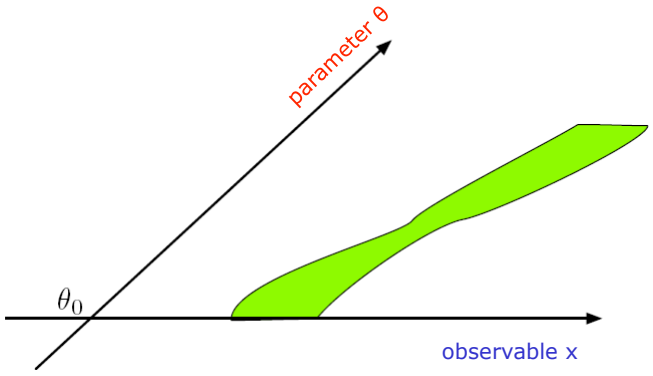
- This makes the confidence belt



Wouter Verkerke, NIKHEF

How to construct a Neyman Confidence Interval

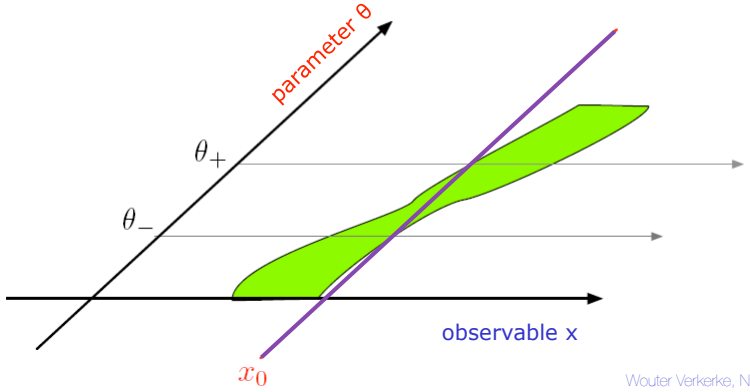
- This makes the confidence belt



Wouter Verkerke, NIKHEF

How to construct a Neyman Confidence Interval

- The confidence belt can constructed *in advance of any measurement*, it is a property of the model, not the data
- Given a measurement x_0, a confidence interval $[\theta_+,\theta_-]$ can be constructed as follows
- The interval $[\theta_-,\theta_+]$ has a 68% probability to cover the true value



What confidence interval means & concept of coverage

- A confidence interval is an interval on a parameter that contains the true value X% of the time
- This is a property of the procedure, and should be interpreted in the concept of repeated identical measurements:

  Each future measurement will result a confidence interval that has somewhat different limits every time
  ('confidence interval limits are a random variable')

  But procedure is constructed such that true value is in X% of the intervals in a series of repeated measurements
  (*this calibration concept is called 'coverage*'. The Neyman constructions guarantees coverage)
- **It is explicitly <u>not</u> a probability statement on the true value** *you are trying to measure. In the frequentist the true value is fixed (but unknown)*

On the interpretation of confidence intervals

Why isn't everyone a Bayesian ?

My suspicion: it is because most people do not understand the frequentist approach. Frequentist statements and Bayesian statements are thought to be about the same logical concept, and the frequentist statement does not require a prior, so …

A. L. Read, *Presentation of search results: the CL_s technique,* J. Phys. G: Nucl. Part. Phys. **28** (2002) 2693-2704.

nearly all physicists tend to misinterpret frequentist results as statements about the theory given the data.

Frequentist statements are not statements about the model – only about the data in the context of the model. This is not what we wanted to know … At least not the ultimate statement.

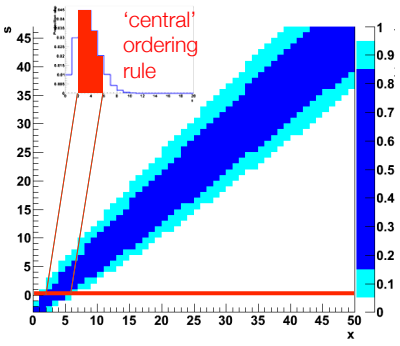
SOS 12

Wouter Verkerke, NIKHEF

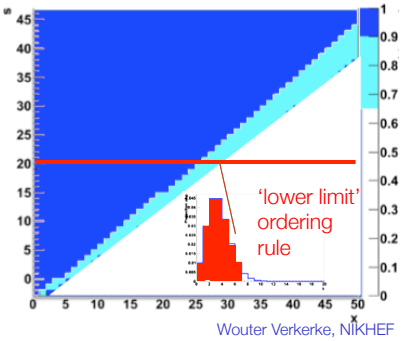
The confidence interval – Poisson counting example

- Given the probability model for Poisson counting example: for every hypothesized value of *s*, plot the expected distribution *N*

Confidence belt for
68% and 90% central intervals



Confidence belt for
68% and 90% lower limit



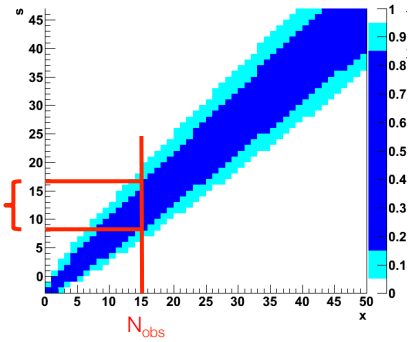
Wouter Verkerke, NIKHEF

Wouter Verkerke, NIKHEF

The confidence interval – Poisson counting example

- Given confidence belt and observed data, confidence interval on parameter is defined by belt intersection

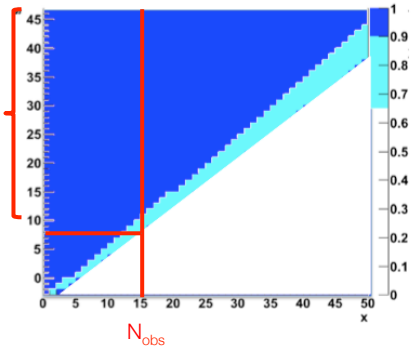
Confidence belt for
68% and 90% central intervals



N_{obs}

Central interval on s at 68% C.L.

Confidence belt for
68% and 90% lower limit



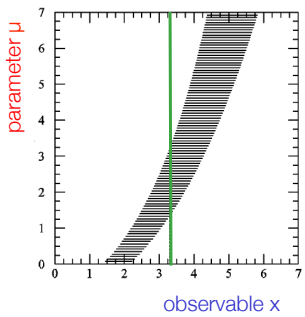
N_{obs}

Lower limit on s at 90% C.L.

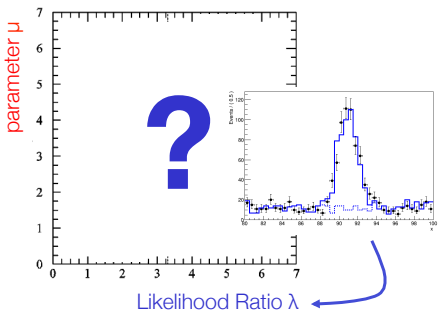
Confidence intervals using the Likelihood Ratio test statistic

- Neyman Construction on Poisson counting looks like 'textbook' belt.
- In practice we'll use the Likelihood Ratio test statistic to summarize the measurement of a (multivariate) distribution for the purpose of hypothesis testing.
- Procedure to construct belt with LR is identical:
  obtain distribution of λ for every value of μ to construct confidence belt

x=3.2



$\lambda_\mu(x,\mu)$



The asymptotic distribution of the likelihood ratio test statistic

- Given the likelihood ratio

$$t_\mu = -2\log\lambda_\mu(x) = -2\log\frac{L(x\,|\,\mu)}{L(x\,|\,\hat{\mu})}$$

  Q: What do we know about asymptotic distribution of λ(μ)?

- A: Wilks theorem → Asymptotic form of f(t|μ) is a χ^2 distribution

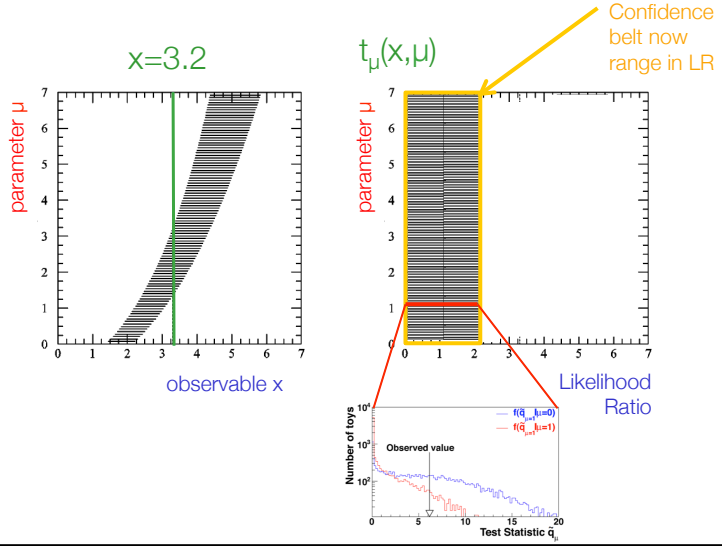
$$f(t_\mu|\mu) = \chi^2(t_\mu, n)$$

  Where
  μ is the hypothesis being tested and
  n is the number of parameters (here 1: μ)

- **Note that $f(t_\mu|\mu)$ is independent of μ!**
  → Distribution of t_μ is the *same* for every 'horizontal slice' of the belt

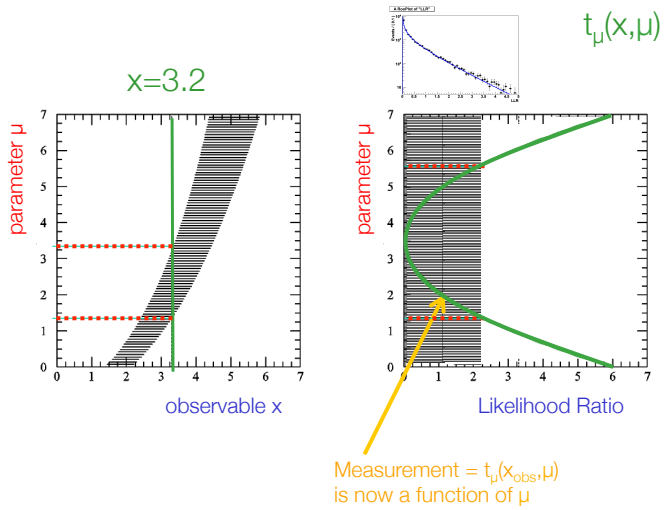
Confidence intervals using the Likelihood Ratio test statistic

- Procedure to construct belt with LR is identical:
  obtain distribution of λ for every value of μ to construct belt



What does the observed data look like with a LR?

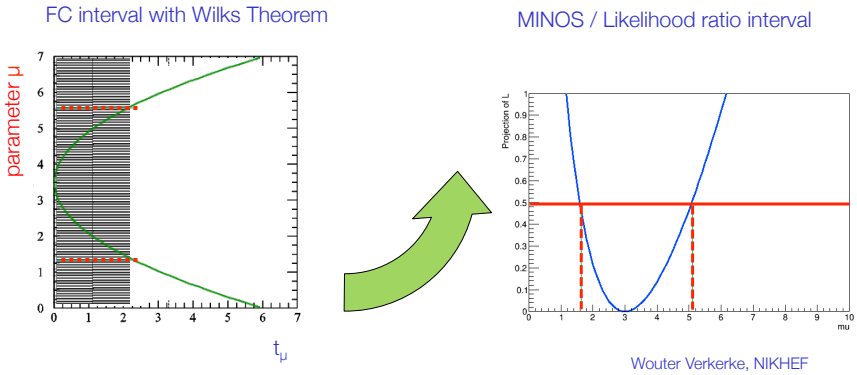
- Note that while belt is (asymptotically) independent of parameter μ, observed quantity now is dependent of the assumed μ



Connection with likelihood ratio intervals

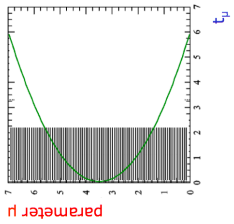
- If you assume the asymptotic distribution for t_μ,
  - Then the confidence belt is exactly a box
  - And the constructed confidence interval can be simplified to finding the range in μ where $t_\mu = ½ \cdot Z^2$

  → **This is exactly the MINOS error**



Recap on confidence intervals

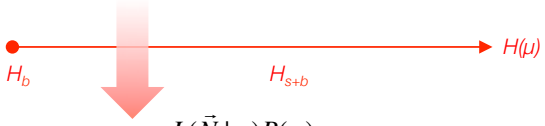
- Confidence intervals on parameters are constructed to have precisely defined probabilistic meaning
  - This calibration is called "coverage"
    The Neyman Construction has coverage by construction
  - This is different from parameter variance estimates
    (or Bayesian methods) that don't have (a guaranteed) coverage
  - For most realistic models confidence intervals are calculated using
    (Likelihood Ratio) test statistics to define the confidence belt
- Asymptotic properties
  - In the asymptotic limit (Wilks theorem),
    Likelihood Ratio interval converges to a
    Neyman Construction interval
    (with guaranteed coverage) "Minos Error"
    *NB: the likelihood does **not** need to be parabolic for Wilks theorem to hold*
  - Separately, in the limit of normal distributions the likelihood becomes exactly parabolic and the ML Variance estimate converges to the Likelihood Ratio interval



Bayesian inference with composite hypothesis

- With change L→L(μ) the prior and posterior model probabilities become probability density functions

$$P(H_{s+b} \mid \vec{N}) = \frac{L(\vec{N} \mid H_{s+b})P(H_{s+b})}{L(\vec{N} \mid H_{s+b})P(H_{s+b}) + L(\vec{N} \mid H_b)P(H_b)}$$



$$P(\mu \mid \vec{N}) = \frac{L(\vec{N} \mid \mu)P(\mu)}{\int L(\vec{N} \mid \mu)P(\mu)\,d\mu}$$

Posterior probability *density*

Prior probability *density*

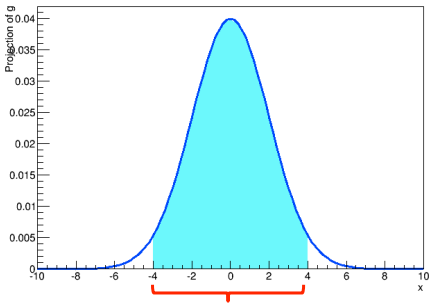
$$P(\mu \mid \vec{N}) \propto L(\vec{N} \mid \mu)P(\mu)$$

NB: Likelihood is not a probability density

Bayesian credible intervals

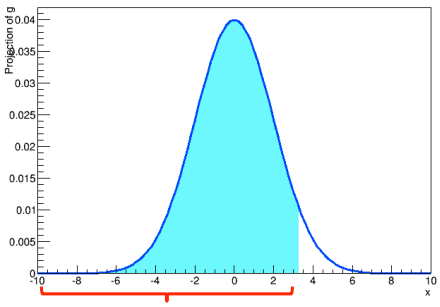
- From the posterior density function, a credible interval can be constructed through integration

Posterior on μ



95% credible central interval

Posterior on μ

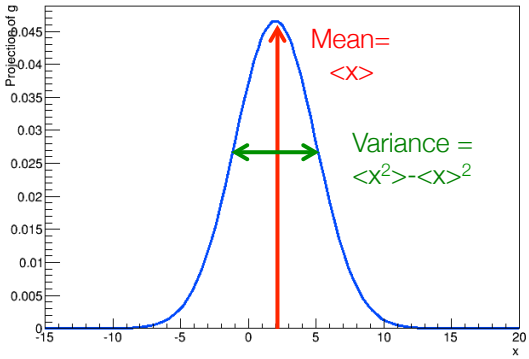


95% credible upper limit

- Note that Bayesian interval estimation require *no minimization* of –logL, just integration

Bayesian parameter estimation

- Bayesian parameter estimate is the posterior mean
- Bayesian variance is the posterior variance

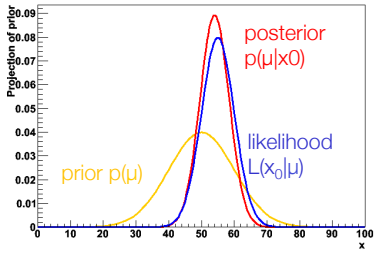


$$\hat{\mu} = \int \mu P(\mu \mid N) d\mu$$

$$\hat{V} = \int (\hat{\mu} - \mu)^2 P(\mu \mid N) d\mu$$

Choosing Priors

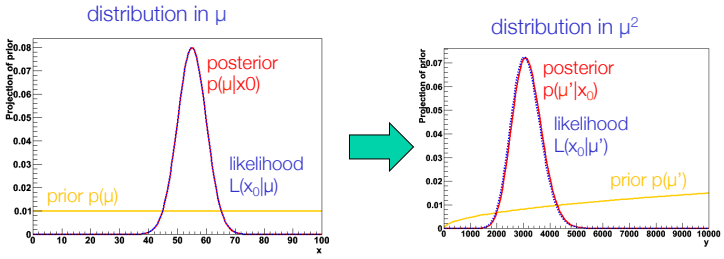
- As for simple models, Bayesian inference always in involves a prior
  → now a prior probability density on your parameter
- When there *is* clear prior knowledge, it is usually straightforward to express that knowledge as prior density function
  - Example: prior measurement of μ = 50 ± 10



  - **Posterior represents updated belief** → It incorporates information from measurement *and* prior belief
  - But sometimes we only want to publish result of *this* experiment, or there is no prior information. What to do?

Choosing Priors

- Common but thoughtless choice: a flat prior
  - Flat implies choice of metric. Flat in x, is not flat in x^2



- Flat prior implies choice on of metric
  - A prior that is flat in μ is not flat in μ^2
  - **'Preferred metric' has often no clear-cut answer.**
    (E.g. when measuring neutrino-mass-squared, state answer in m or m^2)
  - **In multiple dimensions even complicated** (prior flat in x,y or is prior flat in r,ϕ?)

Is it possible to formulate an 'objective' prior?

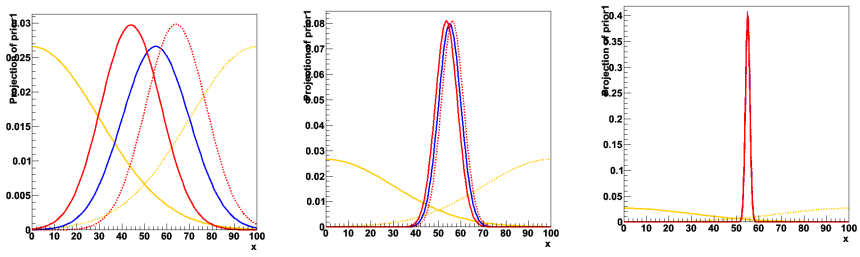
- *Can one define a prior p(μ) which contains as little information as possible, so that the posterior pdf is dominated by the likelihood?*
  - A bright idea, vigorously pursued by physicist Harold Jeffreys in in mid-20thcentury:
  - This is a really *really* thoughtless idea, recognized by Jeffreys as such, but dismayingly common in HEP: just choose p(μ) uniform in whatever metric you happen to be using!
- "Jeffreys Prior" answers the question using a prior uniform in a metric related to the Fisher information.

$$I(\theta) = -E\left[\frac{\partial^2}{\partial\theta^2}\log f(x\,|\,\theta)\middle|\theta\right]$$

  - Unbounded mean μ of gaussian: p(μ) = 1
  - Poisson signal mean μ, no background: p(μ) = 1/√μ
- Many ideas and names around on non-subjective priors
  - Advanced subject well beyond scope of this course.
  - Many ideas (see e.g. summary by Kass & Wasserman), but very much an open/active in area of research

Sensitivity Analysis

- Since a Bayesian result depends on the prior probabilities, which are either personalistic or with elements of arbitrariness, it is widely recommended by Bayesian statisticians to study the sensitivity of the result to varying the prior.
- Sensitivity generally decreases with precision of experiment



- Some level of arbitrariness – what variations to consider in sensitivity analysis

Likelihood Principle

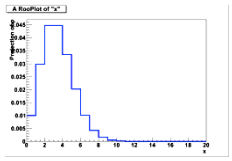
- As noted above, in both **Bayesian** methods and **likelihood-ratio** based methods, the probability (density) for obtaining the *data at hand is used (via the likelihood function), but probabilities for obtaining other data are not used!*
- In contrast, in typical **frequentist** calculations (e.g., a p-value which is the probability of obtaining a value as extreme or *more extreme than that observed), one uses probabilities of data not seen.*
- This difference is captured by the *Likelihood Principle**:

  If two experiments yield likelihood functions which are proportional, then Your inferences from the two experiments should be identical.

Wouter Verkerke, NIKHEF

[B.Cousins HPCP]

The "Karmen Problem"



- Simple counting experiment:
  - You expected precisely 2.8 background events with a Poisson distribution
  - You count the total number of observed events N=s+b
  - You make a statement on s, given N_{obs} and b=2.8
- You observe N=0!
  - Likelihood: L(s) = $(s+b)^0$ exp(-s-b) / 0! = exp(-s) exp(-b)
- Likelihood –based intervals
  - LR(s) = exp(-s) exp(-b)/exp(-b)= exp(-s) → **Independent of b!**
  - Bayesian integral also independent of factorizing exp(-b) term
- So for zero events observed, likelihood-based inference about signal mean **s** *is independent of expected b.*
- For essentially all frequentist confidence interval constructions, the fact that n=0 is less likely for b=2.8 than for b=0 results in *narrower* confidence intervals for μ as b increases.
  - Clear violation of the L.P.

Likelihood Principle Example #2

- Binomial problem famous among statisticians
- Translated to HEP: You want to know the trigger efficiency *e*.
  - You count until reaching n=4000 zero-bias events, and note that of these, m=10 passed trigger.

    Estimate e = 10/4000, compute binomial confidence interval for e.
  - Your colleague (in a different sample!) counts zero-bias events until m=10 have passed the trigger. She notes that this requires n=4000 events.

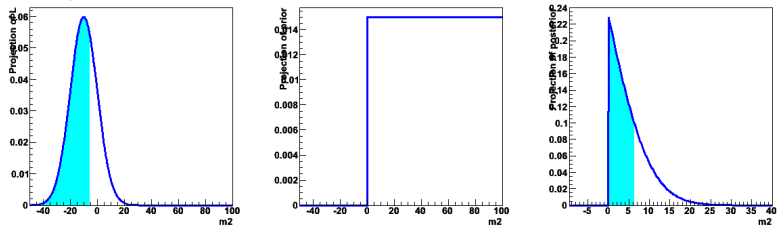
    Intuitively, e=10/4000 *over-estimates* e because she stopped *just* upon reaching 10 passed events. (The relevant distribution is the negative binomial.)
- Each experiment had a different *stopping rule*. Frequentist confidence intervals depend on the stopping rule.
  - It turns out that the likelihood functions for the binomial problem and the negative binomial problem differ only by a constant!
  - So with same n and m, (the strong version of) the L.P. demands *same* inference about e from the two stopping rules!

[B.Cousins HPCP]

Using priors to exclude unphysical regions

- Priors provide simple way to exclude unphysical regions
- Simplified example situations for a measurement of m_ν^2
  1. Central value comes out negative (= unphysical).
  2. Even upper limit (68%) may come out negative, e.g. m^2<-5.3,
  3. What is inference on neutrino mass, given that is must be >0?

$p(\mu|x_0)$ with flat prior | $p'(\mu)$ | $p(\mu|x_0)$ with p'(μ)



- Introducing prior that excludes unphysical region ensure limit in physical range of observable (m^2<6.4)
- NB: Previous considerations on appropriateness of flat prior for domain m^2>0 still apply

Using priors to exclude unphysical regions

- Do you want publish (only) results restricted to the physical region?
  - It depends very much to what further analysis and/or combinations is needed…
- An interval / parameter estimate that in includes unphysical still represents the best estimate of *this* measurement
  - Straightforward to combined with future measurements, new combined result might be physical (and more precise)
  - You need to decide between 'reporting outcome of this measurement' vs 'updating belief in physics parameter'
- Typical issues with unphysical results in confidence intervals
  - 'Low fluctuation of background' → 'Negative signal' → 95% confidence interval excludes *all* positive values of cross-section.
  - Correct result (it should happen 5% of the time), but people feel 'uncomfortable' publishing such a result
- Can you also exclude unphysical regions in confidence intervals?
  - No concept of prior…But yes, it can be done!

Physical boundaries frequentist confidence intervals

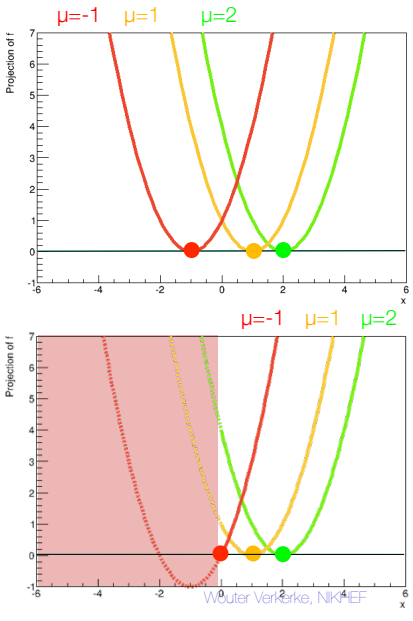
- Solution is to modify the statistic to avoid unphysical region

$$t_\mu(x) = -2\log\frac{L(x\,|\,\mu)}{L(x\,|\,\hat{\mu})}$$

Introduce "physical bound" μ>0

$$\tilde{t}_\mu(x) = \begin{cases} -2\log\dfrac{L(x\,|\,\mu)}{L(x\,|\,\hat{\mu})} & \forall\,\hat{\mu} \geq 0 \\ -2\log\dfrac{L(x\,|\,\mu)}{L(x\,|\,0)} & \forall\,\hat{\mu} < 0 \end{cases}$$

If μ<0, use 0 in denominator
→ *Declare data maximally compatible with hypothesis μ=0*



Physical boundaries in frequentist confidence intervals

- What is effect on *distribution* of test statistic?

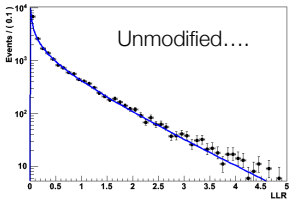
$$t_\mu(x) = -2\log\frac{L(x\,|\,\mu)}{L(x\,|\,\hat{\mu})}$$

Introduce "physical bound" μ>0

$$\tilde{t}_\mu(x) = \begin{cases} -2\log\dfrac{L(x\,|\,\mu)}{L(x\,|\,\hat{\mu})} & \forall\,\hat{\mu} \geq 0 \\ -2\log\dfrac{L(x\,|\,\mu)}{L(x\,|\,0)} & \forall\,\hat{\mu} < 0 \end{cases}$$

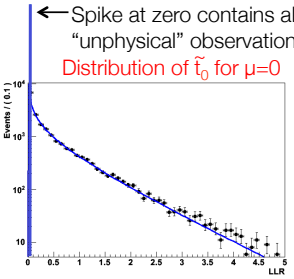
If μ<0, use 0 in denominator
→ *Declare data maximally compatible with hypothesis μ=0*

Distribution of $\tilde{t}_0$ for μ=2



Spike at zero contains all "unphysical" observations

Distribution of $\tilde{t}_0$ for μ=0



Physical boundaries frequentist confidence intervals

- What is effect on *acceptance interval* of test statistic?

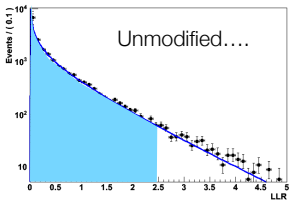
$$t_\mu(x) = -2\log\frac{L(x\,|\,\mu)}{L(x\,|\,\hat{\mu})}$$

Introduce "physical bound" μ>0

$$\tilde{t}_\mu(x) = \begin{cases} -2\log\dfrac{L(x\,|\,\mu)}{L(x\,|\,\hat{\mu})} & \forall\,\hat{\mu} \geq 0 \\ -2\log\dfrac{L(x\,|\,\mu)}{L(x\,|\,0)} & \forall\,\hat{\mu} < 0 \end{cases}$$

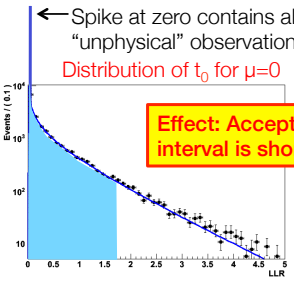
If μ<0, use 0 in denominator
→ *Declare data maximally compatible with hypothesis μ=0*

~



Spike at zero contains all "unphysical" observations

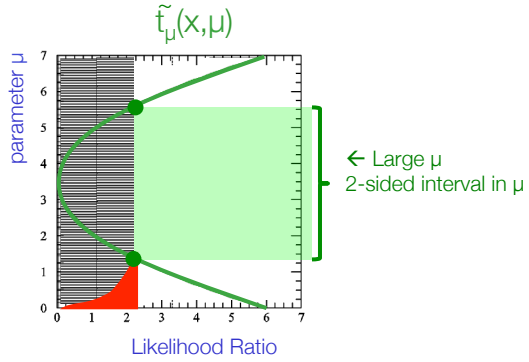
Distribution of t_0 for μ=0



Effect: Acceptance interval is shortened

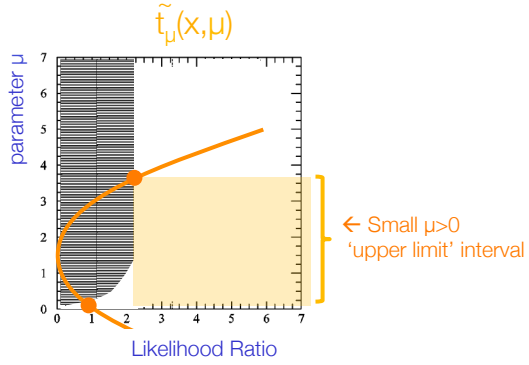
Physical boundaries frequentist confidence intervals

- Putting everything together – the confidence with modified t_μ
- Confidence belt 'pinches' towards physical boundary
- Offsetting of likelihood curves for measurements that prefer μ<0



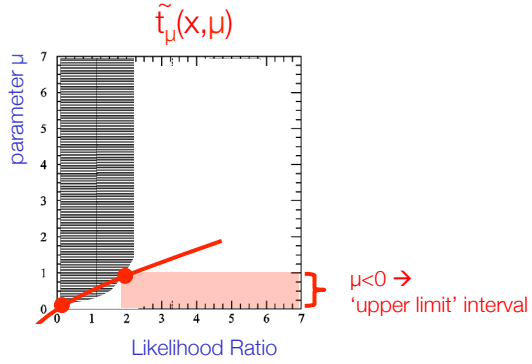
Physical boundaries frequentist confidence intervals

- Putting everything together – the confidence with modified t_μ
- Confidence belt 'pinches' towards physical boundary
- Offsetting of likelihood curves for measurements that prefer μ<0



Physical boundaries frequentist confidence intervals

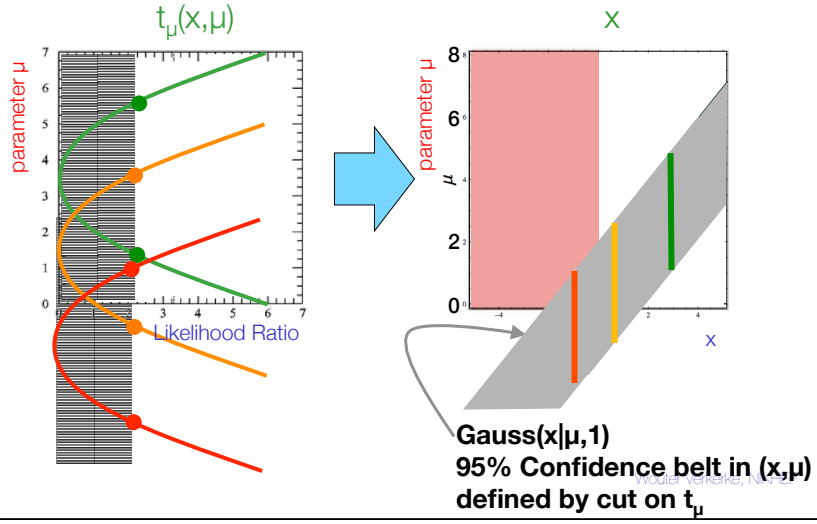
- Putting everything together – the confidence with modified t_μ
- Confidence belt 'pinches' towards physical boundary
- Offsetting of likelihood curves for measurements that prefer μ<0



Physical boundaries frequentist confidence intervals

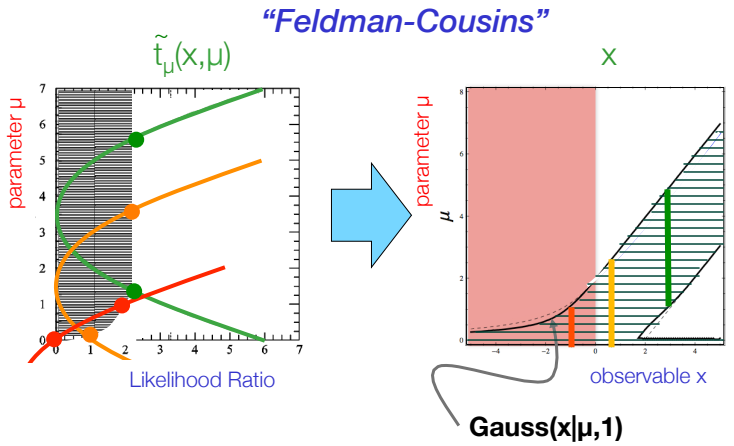
- Example for *unconstrained* unit Gaussian measurement

$$L = Gauss(x \mid \mu, 1)$$



Physical boundaries frequentist confidence intervals

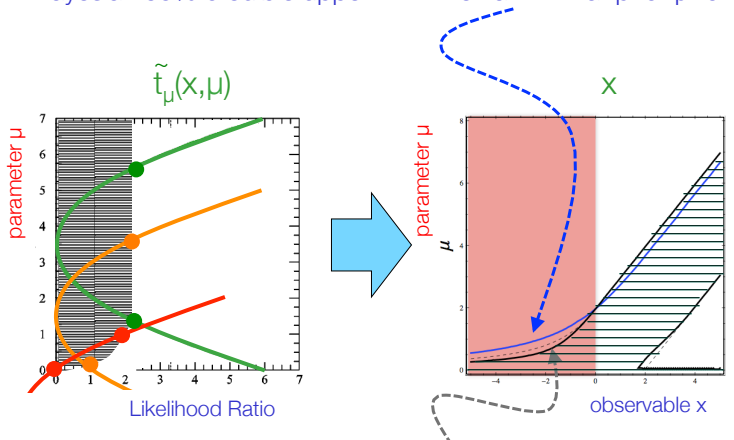
- First map back horizontal axis of confidence belt from $t_\mu(x) \rightarrow x$



Gauss(x|μ,1)
95% Confidence belt in (x,μ)
defined by cut on $\tilde{t}_\mu$

Comparison of Bayesian and Frequentist limit treatment

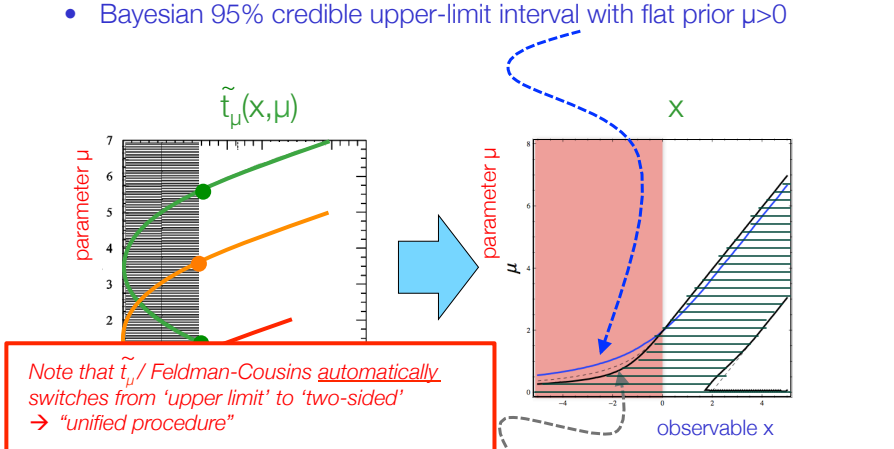
- Bayesian 95% credible upper-limit interval with flat prior μ>0



Gauss(x|μ,1)
95% Confidence belt in (x,μ)
defined by cut on t_μ for

Comparison of Bayesian and Frequentist limit treatment

- Bayesian 95% credible upper-limit interval with flat prior μ>0



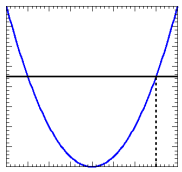
Note that $\tilde{t}_\mu$ / Feldman-Cousins automatically switches from 'upper limit' to 'two-sided'
→ "unified procedure"

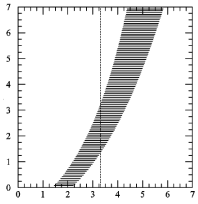
Note that Bayesian and Frequentist intervals at x>2 would agree exactly for Gaussian example if both would be taken as 'two-sided'

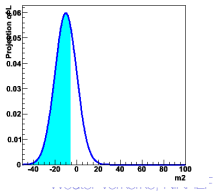
Gauss(x|μ,1)
95% Confidence belt in (x,μ) defined by cut on t_μ for

Summary

- Maximum Likelihood
  - Point and variance estimation
  - Variance estimate assumes normal distribution. No upper/lower limits
- Frequentist confidence intervals
  - Extend hypothesis testing to composite hypothesis
  - Neyman construction provides exact "coverage" = calibration of quoted probabilities
  - Strictly p(data|theory)
  - Asymptotically identical to likelihood ratio intervals (MINOS errors, *does not assume parabolic L*)
- Bayesian credible intervals
  - Extend P(theo) to p.d.f. in model parameters
  - Integrals over posterior density → credible intervals
  - Always involves prior density function in parameter space





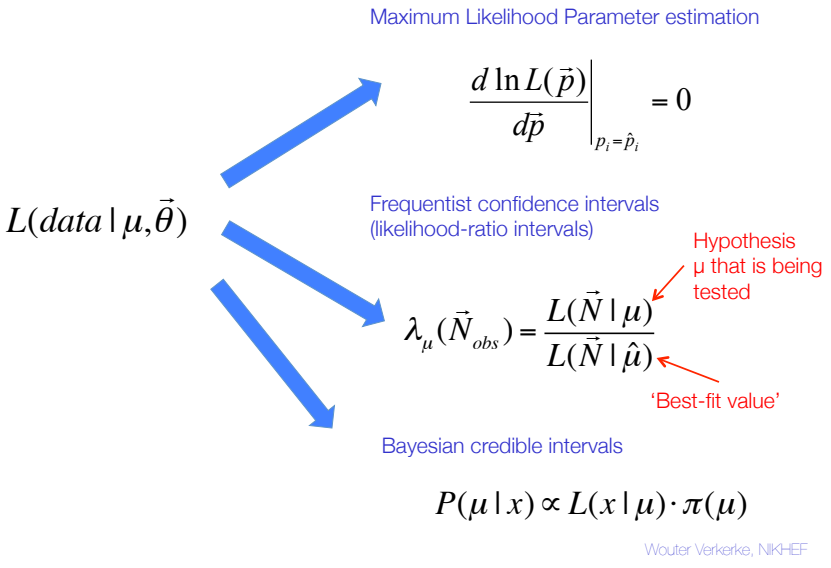


Next subject…

- Start with basics, gradually build up to complexity of

Statistical tests with simple hypotheses for counting data	*"What do we mean with probabilities"*
Statistical tests with simple hypotheses for distributions	*"p-values"*
Hypothesis testing as basis for event selection	*"Optimal event selection & machine learning"*
Composite hypotheses (with parameters) for distributions	*"Confidence intervals, Maximum Likelihood"*
Statistical inference with nuisance parameters	*"Fitting the background"*
Response functions and subsidiary measurements	*"Sideband fits and systematic uncertainties"*

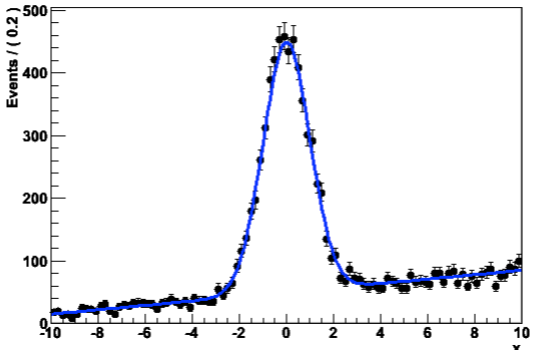
The likelihood is at the basis of many statistical techniques



1

RooFit – Focus: coding likelihood functions

- Focus on one practical aspect of many data analysis in HEP: How do you formulate your likelihood functions in ROOT
  - For 'simple' problems (gauss, polynomial) this is easy



  - But if you want to do unbinned ML fits, use non-trivial functions, or work with multidimensional functions you quickly find that you need some tools to help you

2

Introduction – Why RooFit was developed

- BaBar experiment at SLAC: Extract sin(2β) from time dependent CP violation of B decay: e^+e^- → Y(4s) → BB
  - Reconstruct both Bs, measure decay time difference
  - Physics of interest is in decay time dependent oscillation

$$f_{sig}\cdot\Big[\mathrm{SigSel}(m;\bar{p}_{sig})\cdot\big(\mathrm{SigDecay}(t;\vec{q}_{sig},\sin(2\beta))\otimes\mathrm{SigResol}(t\mid dt;\vec{r}_{sig})\big)\Big]+$$
$$(1-f_{sig})\Big[\mathrm{BkgSel}(m;\bar{p}_{bkg})\cdot\big(\mathrm{BkgDecay}(t;\vec{q}_{bkg})\otimes\mathrm{BkgResol}(t\mid dt;\vec{r}_{bkg})\big)\Big]$$

- Many issues arise
  - Standard ROOT function framework clearly insufficient to handle such complicated functions → must develop new framework
  - Normalization of p.d.f. not always trivial to calculate → may need numeric integration techniques
  - Unbinned fit, >2 dimensions, many events → computation performance important → must try optimize code for acceptable performance
  - Simultaneous fit to control samples to account for detector performance

5

RooFit core design philosophy

- Mathematical objects are represented as C++ objects

Mathematical concept		RooFit class
variable	x	`RooRealVar`
function	$f(x)$	`RooAbsReal`
PDF	$f(x)$	`RooAbsPdf`
space point	$\vec{x}$	`RooArgSet`
integral	$\int_{x_{\min}}^{x_{\max}} f(x)dx$	`RooRealIntegral`
list of space points		`RooAbsData`

6

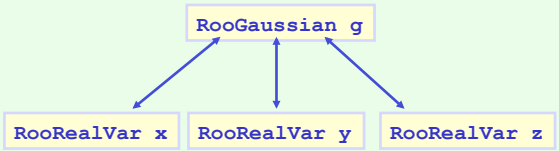
RooFit core design philosophy - Workspace

- Instead of **'double Likelihood(double paramVec[])'**, a flexible modular structure of 'programmed' functions

Math

$\text{Gauss}(x,\mu,\sigma)$

RooFit diagram



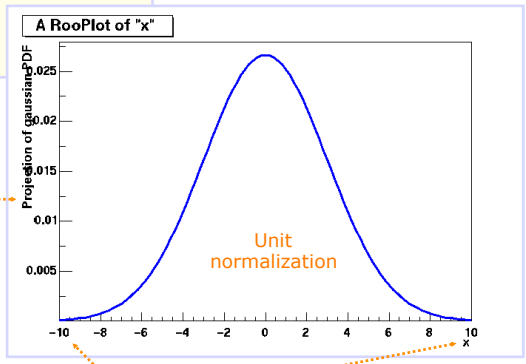
RooFit code

```
RooRealVar x("x","x",-10,10) ;
RooRealVar m("m","y",0,-10,10) ;
RooRealVar s("s","z",3,0.1,10) ;
RooGaussian g("g","g",x,m,s) ;
```

Basics – Creating and plotting a Gaussian p.d.f

Setup gaussian PDF and plot

```
// Create an empty plot frame
RooPlot* xframe = w::x.frame() ;

// Plot model on frame
model.plotOn(xframe) ;

// Draw frame on canvas
xframe->Draw() ;
```



Axis label from `gauss` title

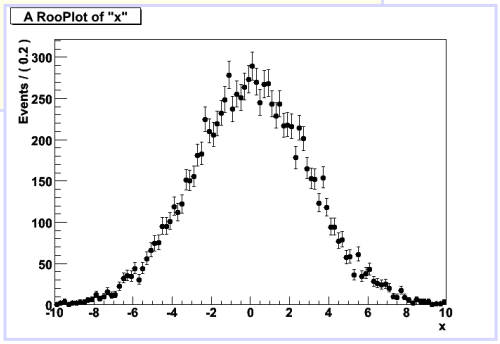
A `RooPlot` is an empty frame capable of holding anything plotted versus it variable

Plot range taken from limits of `x`

Basics – Generating toy MC events

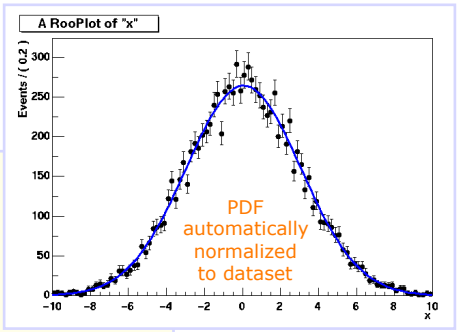
Generate 10000 events from Gaussian p.d.f and show distribution

```
// Generate an unbinned toy MC set
RooDataSet* data = w::gauss.generate(w::x,10000) ;

// Generate an binned toy MC set
RooDataHist* data = w::gauss.generateBinned(w::x,10000) ;

// Plot PDF
RooPlot* xframe = w::x.frame()
data->plotOn(xframe) ;
xframe->Draw() ;
```



Can generate both binned and unbinned datasets

Basics – ML fit of p.d.f to *unbinned* data




```
// ML fit of gauss to data
w::gauss.fitTo(*data) ;
(MINUIT printout omitted)

// Parameters if gauss now
// reflect fitted values
w::mean.Print()
RooRealVar::mean = 0.0172335 +/- 0.0299542
w::sigma.Print()
RooRealVar::sigma = 2.98094  +/- 0.0217306

// Plot fitted PDF and toy data overlaid
RooPlot* xframe = w::x.frame() ;
data->plotOn(xframe) ;
w::gauss.plotOn(xframe) ;
```

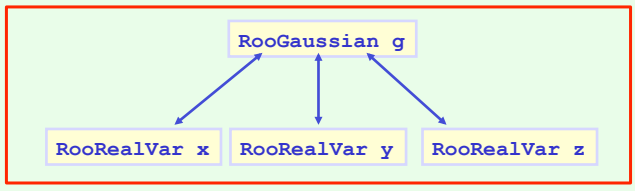

RooFit core design philosophy - Workspace

- The workspace serves a container class for all objects created

Math

$$\text{Gauss}(x,\mu,\sigma)$$

RooFit diagram

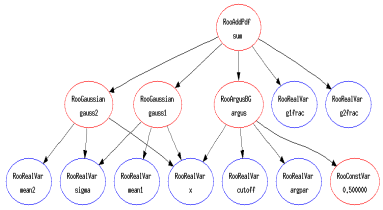


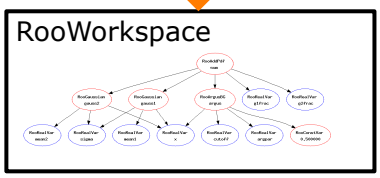
RooFit code

```
RooRealVar x("x","x",-10,10) ;
RooRealVar m("m","y",0,-10,10) ;
RooRealVar s("s","z",3,0.1,10) ;
RooGaussian g("g","g",x,m,s) ;
RooWorkspace w("w") ;
w.import(g) ;
```

The workspace

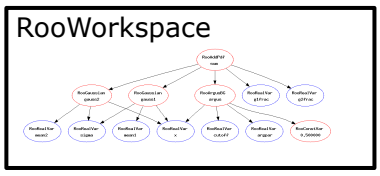
- The workspace concept has revolutionized the way people share and combine analysis
  - Completely factorizes process of building and using likelihood functions
  - You can give somebody an analytical likelihood of a (potentially very complex) physics analysis in a way to the easy-to-use, provides introspection, and is easy to modify.

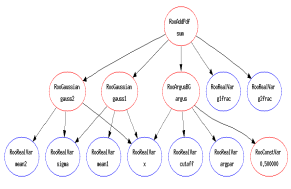




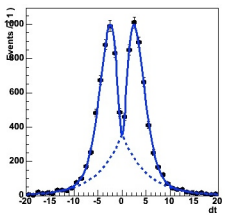

```
RooWorkspace w("w") ;
w.import(sum) ;
w.writeToFile("model.root") ;
```

Using a workspace





```
// Resurrect model and data
TFile f("model.root") ;
RooWorkspace* w = f.Get("w") ;
RooAbsPdf* model = w->pdf("sum") ;
RooAbsData* data = w->data("xxx") ;

// Use model and data
model->fitTo(*data) ;

RooPlot* frame =
        w->var("dt")->frame() ;
data->plotOn(frame) ;
model->plotOn(frame) ;
```



8

Factory and Workspace

- *One C++ object per math symbol* provides ultimate level of control over each objects functionality, but results in lengthy user code for even simple macros
- Solution: add factory that auto-generates objects from a math-like language. Accessed through factory() method of workspace
- Example: reduce construction of Gaussian pdf and its parameters from 4 to 1 line of code

```
RooRealVar x("x","x",-10,10) ;
RooRealVar mean("mean","mean",5) ;
RooRealVar sigma("sigma","sigma",3)  ;
RooGaussian f("f","f",x,mean,sigma) ;
w.import(f) ;
```

```
w.factory("Gaussian::f(x[-10,10],mean[5],sigma[3])") ;
```

6

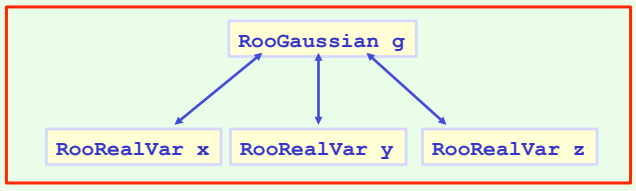
RooFit core design philosophy - Workspace

- The workspace serves a container class for all objects created

Math

$$\text{Gauss}(x,\mu,\sigma)$$

RooFit diagram



RooFit code

```
RooRealVar x("x","x",-10,10) ;
RooRealVar m("m","y",0,-10,10) ;
RooRealVar s("s","z",3,0.1,10) ;
RooGaussian g("g","g",x,m,s) ;
RooWorkspace w("w") ;
w.import(g) ;
```

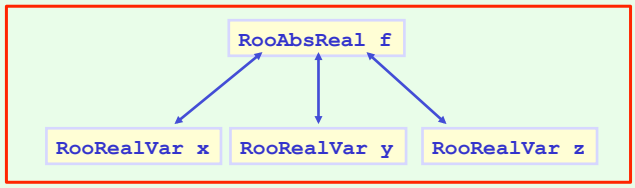
Populating a workspace the easy way – "the factory"

- The factory allows to fill a workspace with pdfs and variables using a simplified scripting language

Math

$$\text{Gauss}(x,\mu,\sigma)$$

RooFit diagram

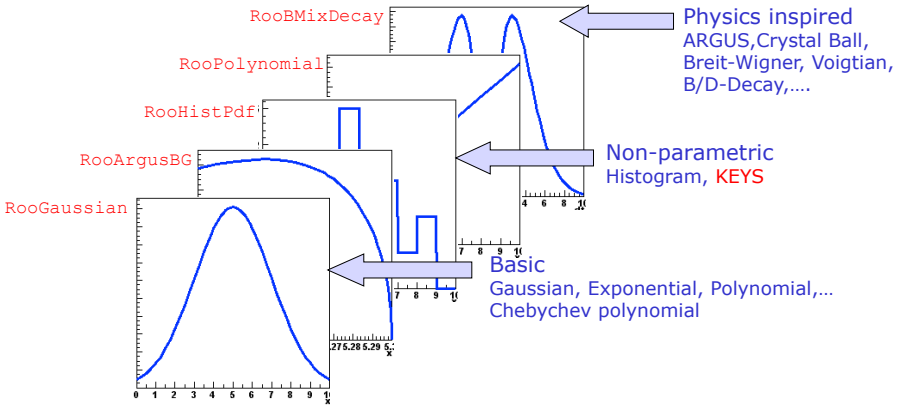


RooFit code

```
RooWorkspace w("w") ;
w.factory("RooGaussian::g(x[-10,10],m[-10,10],z[3,0.1,10])");
```


Model building – (Re)using standard components

- RooFit provides a collection of compiled standard PDF classes



Easy to extend the library: each p.d.f. is a separate C++ class

21

Model building – (Re)using standard components

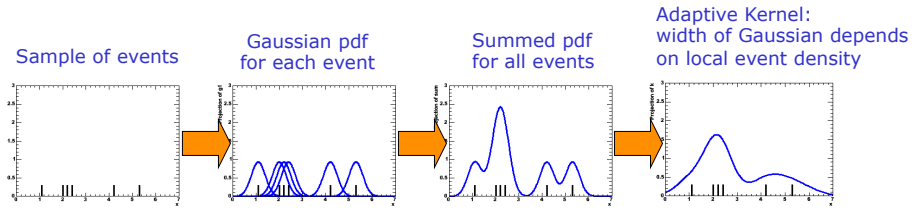
- List of most frequently used pdfs and their factory spec

Gaussian	`Gaussian::g(x,mean,sigma)`
Breit-Wigner	`BreitWigner::bw(x,mean,gamma)`
Landau	`Landau::l(x,mean,sigma)`
Exponential	`Exponental::e(x,alpha)`
Polynomial	`Polynomial::p(x,{a0,a1,a2})`
Chebychev	`Chebychev::p(x,{a0,a1,a2})`
Kernel Estimation	`KeysPdf::k(x,dataSet)`
Poisson	`Poisson::p(x,mu)`
Voigtian (=BW⊗G)	`Voigtian::v(x,mean,gamma,sigma)`

38

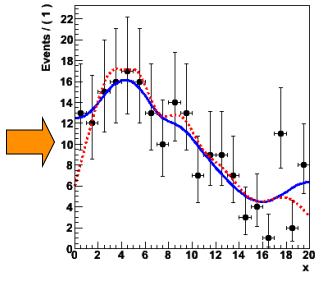
The power of pdf as building blocks – Advanced algorithms

- Example: a 'kernel estimation probability model'
  - Construct smooth pdf from unbinned data, using kernel estimation technique



- Example

```
w.import(myData,Rename("myData")) ;
w.factory("KeysPdf::k(x,myData)") ;
```



- Also available for n-D data

The power of pdf as building blocks – adaptability

- RooFit pdf classes do not require their parameter arguments to be variables, one can plug in functions as well
- Allows trivial customization, extension of probability models

class RooGaussian

$$Gauss(x \mid \mu, \sigma)$$

also class RooGaussian!

$$Gauss(x \mid \mu \cdot (1+2\alpha), \sigma)$$

Introduce a response function for a systematic uncertainty

```
// Original Gaussian
w.factory("Gaussian::g1(x[80,100],m[91,80,100],s[1])")


// Gaussian with response model in mean
w.factory("expr::m_response("m*(1+2alpha)",m,alpha[-5,5])") ;
w.factory("Gaussian::g1(x,m_response,s[1])")
```

NB: "expr" operates builds an intepreted function expression on the fly

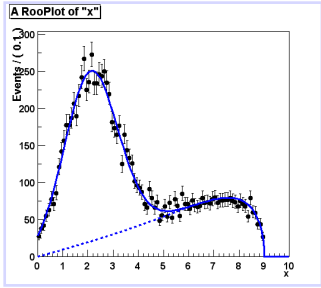
The power of building blocks – operator expressions

- Create a SUM expression to represent a sum of probability models

```
w.factory("Gaussian::gauss1(x[0,10],mean1[2],sigma[1]") ;
w.factory("Gaussian::gauss2(x,mean2[3],sigma)") ;
w.factory("ArgusBG::argus(x,k[-1],9.0)") ;

w.factory("SUM::sum(g1frac[0.5]*gauss1, g2frac[0.1]*gauss2, argus)")
```

- In composite model visualization components can be accessed by name

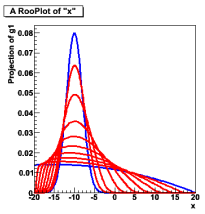
```
// Plot only argus components
w::sum.plotOn(frame,Components("argus"),
              LineStyle(kDashed)) ;
```

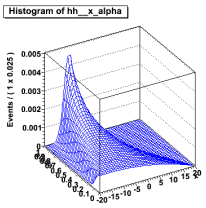


Powerful operators – Morphing interpolation

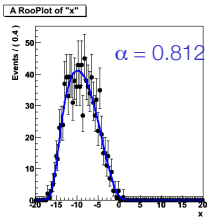
- Special operator pdfs can interpolate existing pdf shapes
  - Ex: interpolation between Gaussian and Polynomial

```
w.factory("Gaussian::g(x[-20,20],-10,2)") ;
w.factory("Polynomial::p(x,{-0.03,-0.001})") ;
w.factory("IntegralMorph::gp(g,p,x,alpha[0,1])") ;
```





Fit to data



$\alpha = 0.812 \pm 0.008$

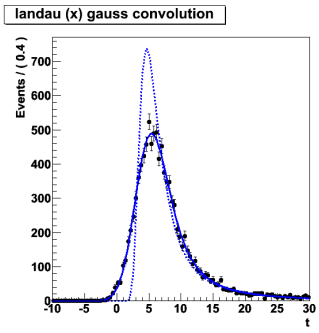
- Three morphing operator classes available
  - IntegralMorph (Alex Read).
  - MomentMorph (Max Baak).
  - PiecewiseInterpolation (via HistFactory)

Powerful operators – Fourier convolution

- Convolve any two arbitrary pdfs with a 1-line expression

```
w.factory("Landau::L(x[-10,30],5,1)") :
w.factory("Gaussian::G(x,0,2)") ;

w::x.setBins("cache",10000) ; // FFT sampling density
w.factory("FCONV::LGf(x,L,G)") ; // FFT convolution
```

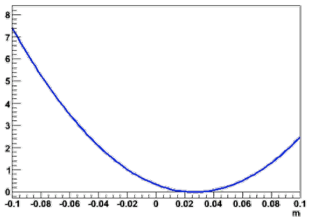
- Exploits power of FFTW package available via ROOT
  - Hand-tuned assembler code for time-critical parts
  - Amazingly fast: unbinned ML fit to 10.000 events take ~5 seconds!



Working with the likelihood function

- Plot the likelihood function versus a parameter

```
RooAbsReal* nll = w::model.createNLL(data) ;

RooPlot* frame = w::param.frame() ;
nll->plotOn(frame,ShiftToZero()) ;
```

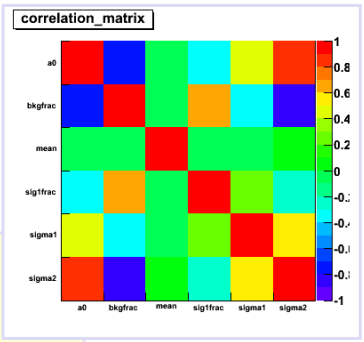


- Maximum Likelihood estimation of parameters and variance

```
RooMinimizer m(*nll) ;

// ML Parameter estimation
m.minimize("Minuit2","migrad") ;

// Variance estimation
m.hesse() ;

// Alternatively - all this in one line
pdf->fitTo(*data) ;
```

Working with covariance and correlation matrices

- Detailed information on parameter and covariance estimates can be saved for detailed information



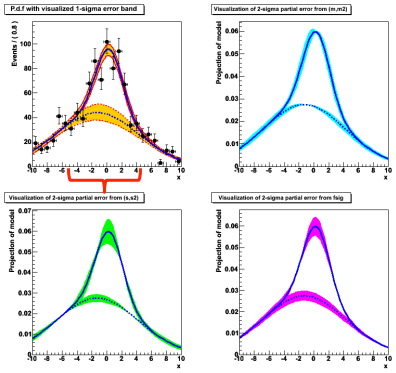

```
RooMinimizer m(*nll) ;
m.minimize("Minuit2","migrad") ;
m.hesse() ;
RooFitResult* r = m.save() ;

// Visualize correlation matrix
r->correlationHist->Draw("colz") ;

// Extract correlation,covariance matrix
TMatrixDSym cov = fr->covarianceMatrix() ;
TMatrixDSym cov = fr->covarianceMatrix(a,b) ;
```

Use covariance matrices for correlated error propagation

- Can (as visual aid) propagate errors in covariance matrix of a fit result to a pdf projection

```
w::model.plotOn(frame,VisualizeError(*fitresult)) ;
w::model.plotOn(frame,VisualizeError(*fitresult,fsig)) ;
```

  - Linear propagation on pdf projection $\Delta = \vec{E} V^{-1} \vec{E}$



- Propagated error can be calculated on arbitrary function
  - E.g fraction of events in signal range

```
RooAbsReal* fracSigRange =
     w::model.createIntegral(x,x,"sig") ;

Double_t err =
    fracSigRange.getPropagatedError(*fr);
```

Next subject…

- Start with basics, gradually build up to complexity of

Statistical tests with simple hypotheses for counting data	*"What do we mean with probabilities"*
Statistical tests with simple hypotheses for distributions	*"p-values"*
Hypothesis testing as basis for event selection	*"Optimal event selection & machine learning"*
Composite hypotheses (with parameters) for distributions	*"Confidence intervals, Maximum Likelihood"*
Statistical inference with nuisance parameters	*"Fitting the background"*
Response functions and subsidiary measurements	*"Sideband fits and systematic uncertainties"*