

# **Applied Mechanism Design and Big Data**

**→ sub-topic: Statistical Data Analysis**

Max Baak

# **Brief summary of lecture 1**

# Lecture 1: Advantages of modeling by hand

- Full control (over contents of the model).
  - You determine all ingredients of the model yourself.
    - E.g. What you want when underlying model is known.
- Assess the uncertainty on the fitted model
  - Can do error propagation of uncertainties on fit parameters.
- Simulation
  - Easy to test impact of parameters in model.
  - Often need for knobs that can be tuned.
    - E.g. test impact of policy changes on a population.
- Hypothesis testing
  - Likelihood fits to data have high(-est) sensitivity for classification.
  - Well-defined formulas to calculate p-values.

# Lecture 1: Properties of the Poisson distribution

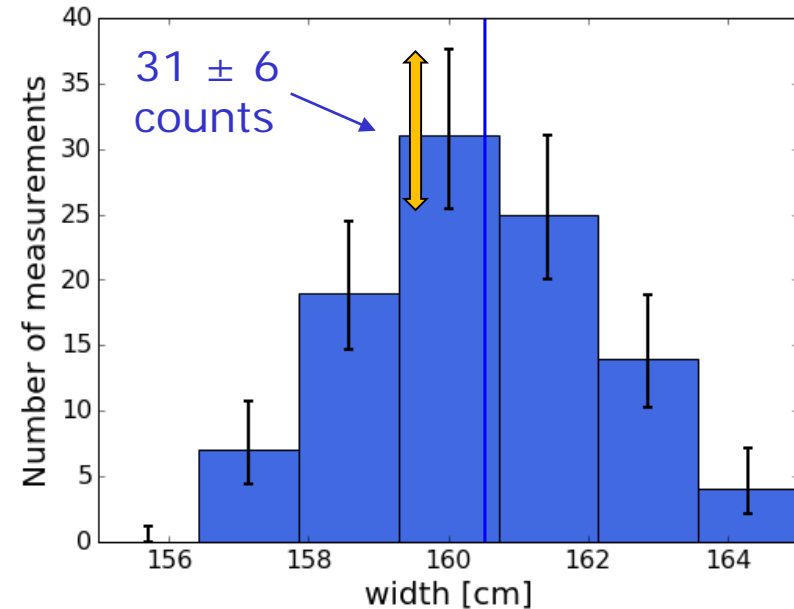
- Poisson distribution:

$$P(r; \lambda) = \frac{e^{-\lambda} \lambda^r}{r!}$$

- Mean, variance:

$$\langle r \rangle = \lambda$$

$$V(r) = \lambda \quad \Rightarrow \quad \sigma = \sqrt{\lambda}$$



- → For a sample with N observations, we expect:
  - $\lambda \approx N$
  - a statistical spread of  $\sqrt{N}$
- Relative uncertainty drops with:  $1/\sqrt{N}$

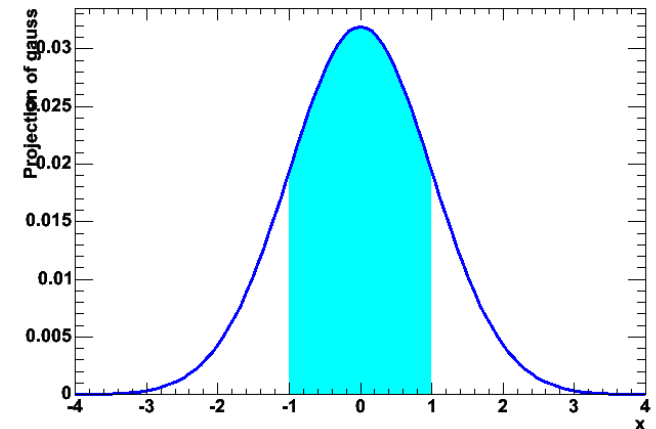
# Lecture 1: Gaussian distribution

- For large samples of data (N records):  
Binomial  $\rightarrow$  Poisson  $\rightarrow$  Gaussian distribution
  - E.g. collect multiple experiments (N) into one bin.*

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

- Mean* and *Variance*

$$\begin{aligned}\langle x \rangle &= \int_{-\infty}^{+\infty} x P(x; \mu, \sigma) dx = \mu \\ V(x) &= \int_{-\infty}^{+\infty} (x - \mu)^2 P(x; \mu, \sigma) dx = \sigma^2 \\ \sigma &= \sigma\end{aligned}$$



- For N (>40) records collected in one bin (from Poisson):
  - Statistical uncertainty  $\sigma_{\text{stat}} = \sqrt{N}$

# Lecture 1: Measurement uncertainties

- Doing an experiment  $\rightarrow$  making measurements ( $X_i$ )
- Measurements not perfect  $\rightarrow$  imperfection quantified in the *resolution* or *error* or *measurement uncertainty* ( $\sigma_i$ )
  - Notation:  $X_i \pm \sigma_{i,\text{stat}} \pm \sigma_{i,\text{sys}}$
- Common language to quote errors
  - Gaussian standard deviation =  $\text{sqrt}(V(x))$
  - 68% probability that true values is within quoted errors
- *Central Limit Theorem:*  
*Errors are usually Gaussian if they quantify a result that is based on sum of (many) independent measurements*

## Headline news (VK: 7 sep 2016)

# Alweer zo'n psychologisch lachertje

Opnieuw blijkt een beroemd psychologisch experiment bij nader inzien gebakken lucht. Dat is het lot van tweederde van de herhaalde proeven. Een heel wetenschapsgebied loopt leeg.

Van onze verslaggever  
**Maarten Keulemans**

**AMSTERDAM** Voor wie dacht dat een mens gelukkiger wordt door de mond in een glimlach te plooiën: uit een grootscheepse herhaling van het achterliggende experiment blijkt 'he-le-maal niets', zegt onderzoeksleider Eric-Jan Wagenmakers (UvA).

Het glimlach-effect werd in 1988 beschreven door een team onder leiding van de Duitse psycholoog Fritz Strack. Laat vrijwilligers een potlood tussen de tanden klemmen zodat hun mond een glimlach vormt, betoogde Strack na experimenten met 92 studenten, en ze beoordelen cartoons als grappiger dan iemand die de pen tussen de lippen houdt en daarmee een prutgericht trekt.

De studie werd de hoeksteen van de 'gezichtsfeedback-hypothese', de gedachte dat een emotionele gelaatsuitdrukking de stemming kan bepalen. Stracks studie werd al 1.440 keer door collega's geciteerd en wordt nog steeds als hulpmiddelje gebruikt door sommige therapeuten.

Maar het blijkt dus allemaal nep. Nota bene op uitnodiging van Strack zelf gingen 17 laboratoria in de VS, Canada en Europa ertoe over om de experimenten over te doen. In geen van de laboratoria ging de vrolijkheid meetbaar omhoog. 'Niemand vindt iets. En bij elkaar opgeteld vinden we ook niets', zegt Wagenmakers.

'Ik ging het project in met de gedachte dat we wel iets zouden vinden, maar dat is niet gebeurd. Het is een beetje



Klassiek, nu door de mand gevallen psychologie-experiment: potlood tussen de tanden maakt vrolijker dan tussen de lippen. Niet dus.



# Headline news – continued (VK: 7 sep 2016)

'Ik ging het project in met de gedachte dat we wel iets zouden vinden, een kleiner effect', zegt de Rotterdamse psychologiehoogleraar Rolf Zwaan, een van de deelnemers. 'Niet leuk', vindt Zwaan de uitkomst. 'Het laat je toch achter met een beetje een leeg gevoel.'

In een reactie meldt Strack 'heel verast' te zijn. De inmiddels 65-jarige psycholoog noemt een reeks technische kritiekpunten, maar krijgt op de psychologische blogs nauwelijks bijval. 'Onze replicatie is veel nauwkeuriger uitgevoerd dan Stracks eigen experiment', zegt Zwaan. 'Ik vind zijn reactie niet erg constructief.'

De psychologie is schoon schip aan het maken, nadat in het vakgebied een reeks wantoestanden aan het licht kwam – met als opvallendste de jarenlange fraude van sociaal-psycholoog Diederik Stapel. Sindsdien is men diverse sleutelexperimenten ter controle aan het herhalen. Vorige maand trok onderzoeksfinancier NWO daarvoor nog 3 miljoen euro uit.

Gevolg is dat de ene na de andere klassieker sneuvelt. Wie denkt aan ouderdom, gaat bij nader inzien toch niet langzamer lopen. Het 'knuffelhormoon' oxytocine in de neus sprayen blijkt geen effect te hebben op vertrouwen. Toen psychologen vorig jaar honderd studies overdeden, bleek dat liefst tweederde in de herhaling andere uitkomsten gaf.

Advertentie

Lees ook **kwaliteit**  
op het perron

**V** Zet de Vop  
uw telefoon

Kijk op [volkskrant.nl/digitaal](http://volkskrant.nl/digitaal)

Klassiek, nu door de mand gevallen psychologie-experiment: potlood tussen de tanden maakt vrolijker dan tussen de lippen. Niet dus.

## 'De wetenschap staat in brand, er is alle reden voor paniek'

**Interview**  
**Eric-Jan Wagenmakers,**  
**hoogleraar UvA**

Steeds weer blijkt dat sociaal-psychologische experimenten niet zijn te herhalen. Dat is echt een heel serieus probleem.

Van onze verslaggever  
**Maarten Keulemans**

**D**e mondhoeken optrekken in een glimlach maakt dus toch niet gelukkig. Methodoloog Eric-Jan Wagenmakers (44, UvA) leidde de replicatie die er gehakt van maakt – en denkt dat de hele wetenschap een ernstig probleem heeft.

**Je gezicht geforceerd in een glimlach trekken is dus toch niet genoeg om gelukkiger te worden. Heeft Fritz Strack, de ontdekker van het effect, gelogen?**

'Dat denk ik niet. Er kan van alles aan de hand zijn. Zo was er bij zijn experimenten altijd iemand aanwezig om te kijken of het experiment goed verliep. Dat kan de uitkomsten hebben beïnvloed. Misschien dat het gezicht van de waarnemer net iets meer in de



**De bom is ontploft in de sociale psychologie, maar we gaan het voelen in veel meer onderzoeksgebieden**



lach gaat staan als zo'n vrijwilliger daar zit met een potlood tussen de tanden. Dat kan de vrolijkheid hebben vergroot.'

**Het was anders wel een fors effect, wat Strack vond. De potloodbijters vonden die cartoons écht veel grappiger.**

'Ik ben geneigd te denken dat het komt door hoe ze toen onderzoek deden. Ze legden niet vooraf vast wat ze precies wilden bestuderen. Het lijkt erop dat ze gewoon de experimenten hebben gedaan en daarna de krenten uit de pap hebben gehaald: kijk, een effect! Dat zien we vaker. De data zijn niet zo sterk, maar wat zou het mooi zijn als het klopt! En dan vind je ook wat.'

**Intussen verschijnen er haast dagelijks studies die voortborduren op dit effect. Zo verspreidt zo'n niet bestaand feit zich als een veenbrand.**

'Ons doel was om het beroemdste experiment te repliceren: het experiment waarmee het allemaal is begonnen. Onze hoop is dat andere onderzoekers hierdoor worden wakker geschud en het netter gaan doen. Want nu zijn we elkaar gewoon voor de gek aan het houden.'

**Dat klinkt ernstig.**

'Dat is het ook. We zien voortdurend dat als we een bepaald onderzoek pakken, we het niet kunnen repliceren. Het is tijd voor paniek. De wetenschap staat in brand.'

**Voorbeeldje?**

'Neem het vermeende effect van videospelletjes op agressie: het lijkt erop dat het niet bestaat. Studies die zouden uitwijzen dat je slimmer wordt door het spelen van bepaalde braintraining-spelletjes lijken niet replicerbaar. En dat is denk ik nog maar het begin. De bom is ontploft in de sociale psychologie, maar de gevolgen ervan gaan we in veel meer onderzoeksgebieden voelen.'

**Zoals?**

'Ik durf de stelling wel aan dat het in de neurowetenschappen minstens zo erg is. Mensen met een grotere amygdala hebben meer Facebook-vrienden; dat soort inzichten. Je hebt te maken met dure experimenten met weinig proefpersonen, en veel flexibiliteit in de statistische analyses. En ze zijn nog nauwelijks getoetst door ze te herhalen.'

**Fritz Strack schrijft in een pinnige reactie dat er van alles rammelt aan uw replicatie.**

'Verplaats je eens in zijn positie: zijn nalatenschap, zijn grote bijdrage aan de wetenschap, blijkt niet te repliceren. Dus ik snap zijn emotie. Maar wat ik mis, is de enige verstandige reactie. Als dit echt zo'n overduidelijk effect is, zet dan zelf een experiment op. Laat maar zien hoe je het dan wel aantoonst.'



Side note: Importance of  $\sqrt{N}$ 

# Merendeel psychologische studies houdt geen stand

Van onze verslaggevers  
**Maarten Keulemans,**  
**Margreet Vermeulen**

**AMSTERDAM** Het meeste gepubliceerde psychologische onderzoek lijkt gebaseerd op drijfzand. Zo'n 60 tot 65 procent van de studies levert wezenlijk andere uitkomsten op als de experimenten waarop ze zijn gebaseerd, worden herhaald met andere proefpersonen.

De afgelopen vier jaar is een internationaal conglomeraat van 270 psychologen – onder wie 45 Nederlanders – in de weer geweest met het nauwgezet herhalen van honderd onderzoeken die in 2008 verschenen in drie vooraanstaande psychologische vakbladen.

Vandaag verschijnen de uitkomsten in *Science*: slechts van 39 procent van de oorspronkelijke studies werden de belangrijkste resultaten met succes gereproduceerd. Bovendien leverde 83 procent van alle experimenten bij herhaling minder sterke uitkomsten op.

Als een onderzoek niet met succes kan worden herhaald, betekent dat vaak dat het oorspronkelijke resultaat een toevalstreffer was. Of dat de onderzoekers vooringenomen waren of fouten hebben gemaakt. Fraude is ook een mogelijkheid, maar daarvan is in de onderzochte studies niets gebleken.

Initiatiefnemer en psychologie-hoogleraar Brian Nosek denkt wel te

snappen waarom er zo veel uitkomsten niet herhaalbaar zijn. 'Nieuwe, positieve en mooie resultaten hebben meer kans om door de selectie en in de vakbladen te komen', vertelt hij telefonisch vanuit Virginia. 'Daardoor kan het gebeuren dat negatieve resultaten juist worden weggelaten.'

De psychologische studies in kwestie gaan over wetenschappelijke detailzaken in de sociale en de cognitieve

**'Misschien klopt 80 procent van de psychologische literatuur niet'**

psychologie: therapieën of diepe inzichten staan niet op het spel. Niettemin spreekt Stanford-methodoloog John Ioannidis, niet betrokken bij het replicatieproject, van een zwarte dag. 'Het aantal mislukte replicaties is erg hoog. En dan is dit nog een steekproef van wat je kunt omschrijven als de beste studies, uit de beste vakbladen. Dat doet vermoeden dat van de hele psychologische literatuur misschien wel 80 procent of meer niet klopt.'

De psychologen zelf zien de positieve kant. 'De psychologie is de eerste discipline die op deze manier wordt getest', aldus voorzitter Alan Kraut van

de Amerikaanse psychologenvereniging APS. 'Ik ben trots dat we onze verantwoordelijkheid nemen.' Uit eerdere, enigszins vergelijkbare replicatiestudies bleek dat ook veel andere vakken met het probleem worstelen.

Onderzoek herhalen geldt als de gouden standaard van de wetenschap. Het gebeurt alleen te weinig, omdat 'replicatie' geldt als saai en ondankbaar werk. Bovendien kan ook replicatie vertekende resultaten opleveren als alleen gelukte replicaties worden gepubliceerd. Juist daarom is het herhaalproject van de psychologen – systematisch een hele reeks studies overdoen – 'enorm belangrijk', zegt Ioannidis.

Extra urgentie kreeg de zaak door de megafraude van hoogleraar sociale psychologie Diederik Stapel. 'Het is zeker zo dat het Stapel-incident de interesse in reproduceerbaarheid heeft vergroot', zegt Nosek. 'Maar ons idealisme om het vakgebied betrouwbaarheid te geven staat bovenaan.'

In Leiden is hoogleraar Bernhard Hommel een van de 'slachtoffers': zijn experiment dat zou aantonen dat tweetalige mensen zich beter kunnen concentreren, leverde in de herhaling niets meer op. 'Toch is Hommel positief: 'Dit is wat de wetenschap aanwakt. Zo heb je weer voortgang.'

**PAGINA 12-13**

Hoe kunnen psychologen het zo verkeerd doen?

spectaculaire experimenten – is al bijna gewoon. Net als andere maatregelen: onderzoek doen in grote, internationale groepen en een onderzoek eerst zelf repliceren voordat je het opschrijft, bijvoorbeeld.

De replicatiestudie van de psychologen – een steekproef van honderd studies opnieuw doen – is een volgende stap. In de VS is men al bezig met replicatieproject nummer twee: honderd experimenten uit het kan-keronderzoek herhalen.

Toch is daarmee de kous niet af, denken de meeste experts. Al helemaal niet in een politiek-economisch klimaat waar de wetenschap steeds meer wordt gerund als bedrijf: concurreren met elkaar, publiceer veel, haal klinkende resultaten.

'Het probleem', zegt Van Assen, 'is dat het voor individuele onderzoekers nog steeds loont als ze onderzoek doen met kleine steekproeven en zonder preregistratie.'

Het mag dan stapje voor stapje steeds beter gaan; het zal nog lang duren voordat de wetenschap de gemakkelijke weg naar succes helemaal verruult voor de koninklijke weg.

**Maarten Keulemans**

# Roadmap for this course

## 1. Statistics basics

- Probability theory
- Probability distributions

## 2. **Parameter estimation**

- Error propagation
- Simulation
- Model fitting (bulk)

Today's course

## 1. Pitfalls in (big) data analysis

- Spurious correlations in Big Data
- Data quality assessment

## 1. Hypothesis Testing

# Error propagation

## Error propagation – one variable

- Suppose we have  $f(x) = ax + b$
- How do you calculate  $V(f)$  from  $V(x)$ ?

$$\begin{aligned} V(f) &= \langle f^2 \rangle - \langle f \rangle^2 \\ &= \langle (ax + b)^2 \rangle - \langle ax + b \rangle^2 \\ &= a^2 \langle x^2 \rangle + 2ab \langle x \rangle + b^2 - a \langle x \rangle^2 - 2ab \langle x \rangle - b^2 \\ &= a^2 (\langle x^2 \rangle - \langle x \rangle^2) \\ &= a^2 V(x) \end{aligned} \quad \leftarrow \text{i.e. } \sigma_f = |a| \sigma_x$$

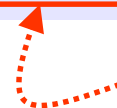
- More general:  $V(f) = \left( \frac{df}{dx} \right)^2 V(x) \quad ; \quad \sigma_f = \left| \frac{df}{dx} \right| \sigma_x$

– But only valid if *linear approximation is good in range of error*

# Error propagation – summing 2 variables

- Consider  $f = ax + by + c$

$$V(f) = a^2 \left( \langle x^2 \rangle - \langle x \rangle^2 \right) + b^2 \left( \langle y^2 \rangle - \langle y \rangle^2 \right) + 2ab \left( \langle xy \rangle - \langle x \rangle \langle y \rangle \right)$$
$$= a^2 V(x) + b^2 V(y) + \underline{2ab \operatorname{cov}(x, y)}$$

 Familiar 'add errors in quadrature'  
**only valid in absence of correlations,**  
i.e.  $\operatorname{cov}(x, y) = 0$

- More general

$$V(f) = \left( \frac{df}{dx} \right)^2 V(x) + \left( \frac{df}{dy} \right)^2 V(y) + 2 \left( \frac{df}{dx} \right) \left( \frac{df}{dy} \right) \operatorname{cov}(x, y)$$
$$\sigma_f^2 = \left( \frac{df}{dx} \right)^2 \sigma_x^2 + \left( \frac{df}{dy} \right)^2 \sigma_y^2 + 2 \left( \frac{df}{dx} \right) \left( \frac{df}{dy} \right) \rho \sigma_x \sigma_y$$

But only valid if *linear approximation*  
*is good in range of error*

**The correlation coefficient**  
 $\rho$   $[-1, +1]$  is 0 if  $x, y$  uncorrelated

# Error propagation – multiplying, dividing 2 variables

- Now consider  $f = x \cdot y$

$$V(f) = y^2 V(x) + x^2 V(y) \quad (\text{math omitted})$$

$$\left( \frac{\sigma_f}{f} \right)^2 = \left( \frac{\sigma_x}{x} \right)^2 + \left( \frac{\sigma_y}{y} \right)^2$$

- Result similar for  $f = x / y$

- Other useful formulas

$$\frac{\sigma_{1/x}}{1/x} = \frac{\sigma_x}{x}$$

**Relative error on  
x, 1/x is the same**

;

$$\sigma_{\ln(x)} = \frac{\sigma_x}{x}$$

**Error on log is just  
fractional error**



## Error propagation – Making predictions

- Suppose linear model:  $f(x) = ax + b$
- ... which has been fit to a dataset of observables  $x, y$

$$\{ (x_1, y_1), (x_2, y_2), \dots (x_N, y_N) \}$$

... resulting in fitted values for parameter  $a$  and  $b$ .

– Fit returns:  $\sigma_a, \sigma_b, \rho_{a,b}$

(details of fit discussed later)

- Now we wish to predict  $f(x)$  for certain value of  $x$ .

What is the uncertainty on  $f(x)$  ?

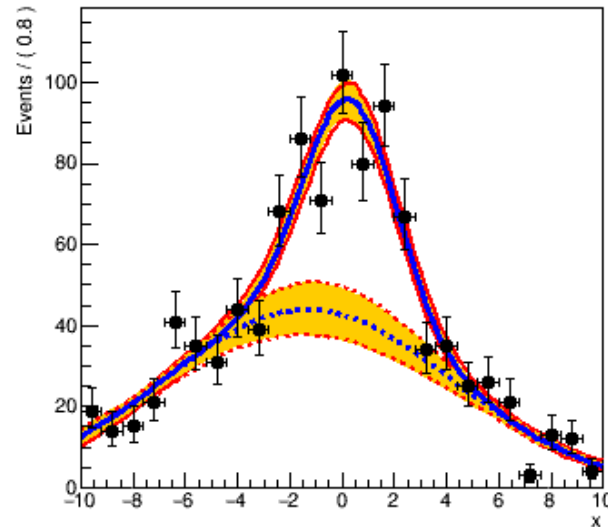
$$V(f) = \left( \frac{df}{da} \right)^2 V(a) + \left( \frac{df}{db} \right)^2 V(b) + 2 \left( \frac{df}{da} \right) \left( \frac{df}{db} \right) \text{cov}(a, b)$$

$$\sigma_f^2 = x^2 \sigma_a^2 + \sigma_b^2 + 2x \rho \sigma_a \sigma_b$$

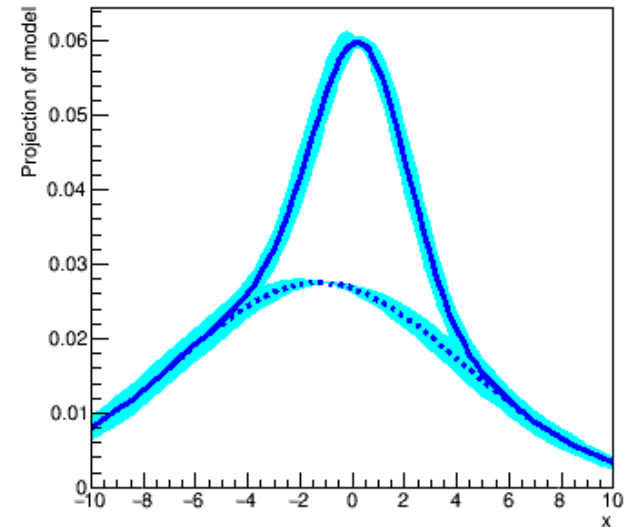
# Error propagation & visualization

- Error propagation works for any formula!
- Also when extrapolating  $x$  *beyond* fitted range.
- Ideal for making predictions *including model uncertainties!*

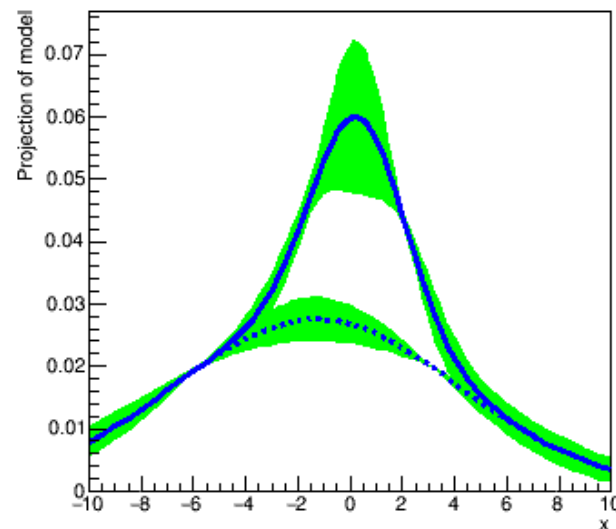
P.d.f with visualized 1-sigma error band



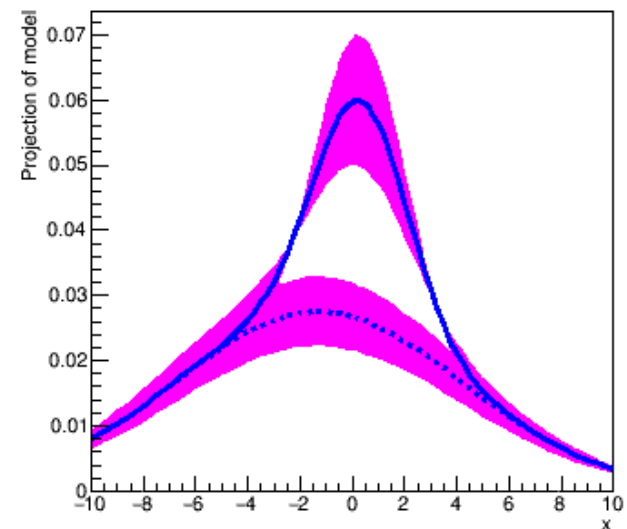
Visualization of 2-sigma partial error from (m,m2)



Visualization of 2-sigma partial error from (s,s2)



Visualization of 2-sigma partial error from fsig



# Simulation

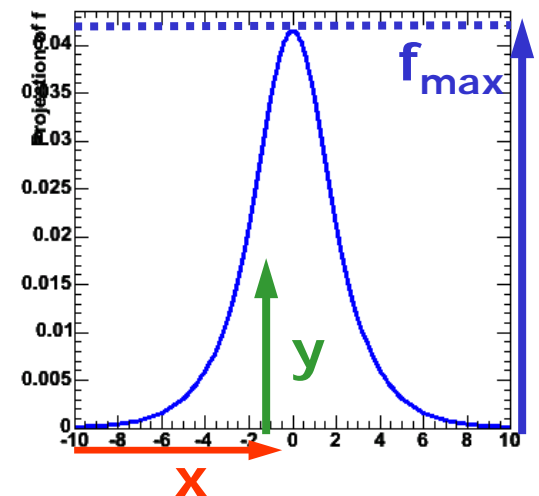
## How to obtain 10.000.000 simulated records?

- Practical issue: very normal to need very large amounts of simulated events for a fit validation study
  - Of order 1000x number of events in your fit, easily >1.000.000 events
- Solution: Use events sampled directly from your fit function
  - Technique named '*Toy Monte Carlo*' sampling
  - Advantage: Easy to do and very fast
  - Good to determine fit bias due to low statistics, choice of parameterization, boundary issues etc.

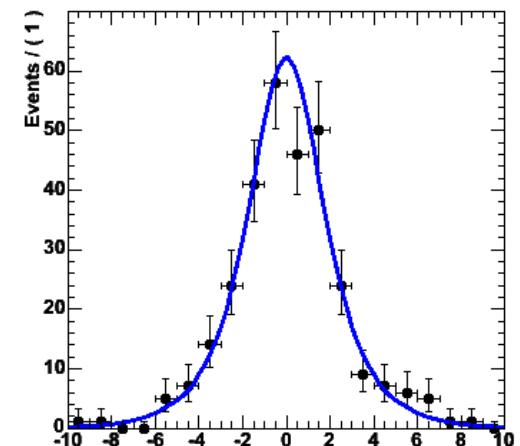
# Toy MC generation – Accept/reject sampling

- *How to sample events directly from your fit function?*
- Simplest: accept/reject sampling

- 1) Determine maximum of function  $f_{\max}$
- 2) Throw random number  $x$
- 3) Throw another random number  $y$
- 4) If  $y < f(x)/f_{\max}$  keep  $x$ ,  
otherwise return to step 2)



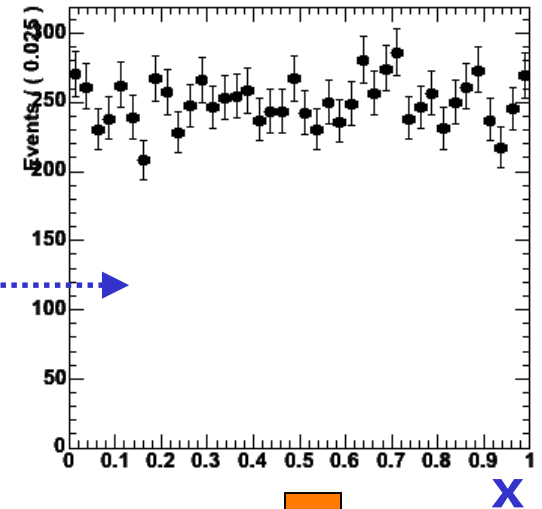
- PRO: Easy, always works
- CON: It can be inefficient if function is strongly peaked.  
Finding maximum empirically through random sampling can be lengthy in  $>2$  dimensions



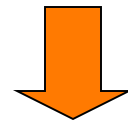
# Toy MC generation – Inversion method

- Fastest: function inversion

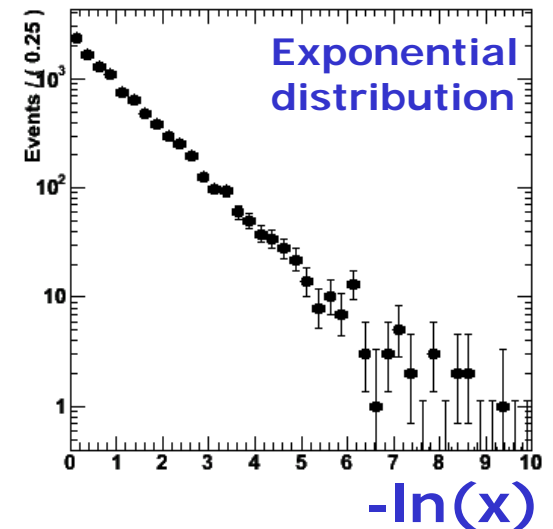
- 1) Given  $f(x)$  find inverted function  $F(x)$  so that  $f(F(x)) = x$
- 2) Throw uniform random number  $x$
- 3) Return  $F(x)$



Take  $-\log(x)$



- PRO: Maximally efficient
- CON: Only works for invertible functions

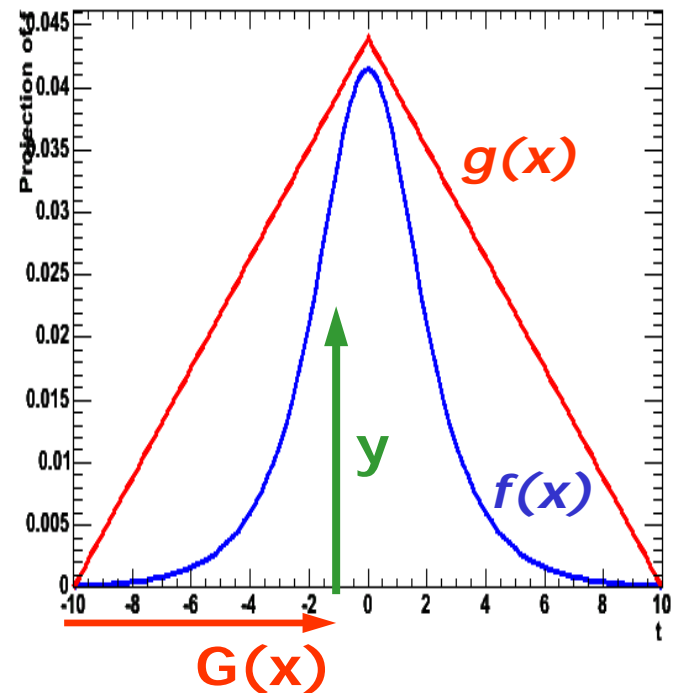




# Toy MC Generation in a nutshell

- Hybrid: Importance sampling

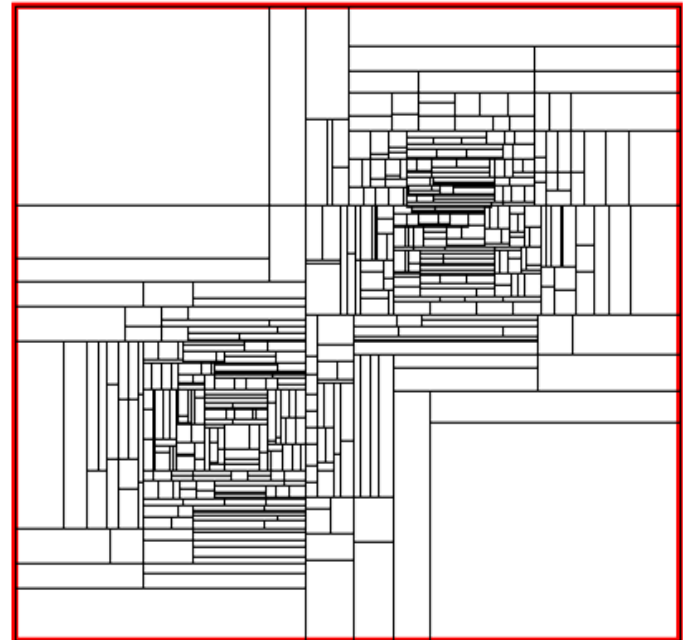
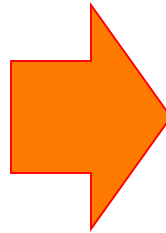
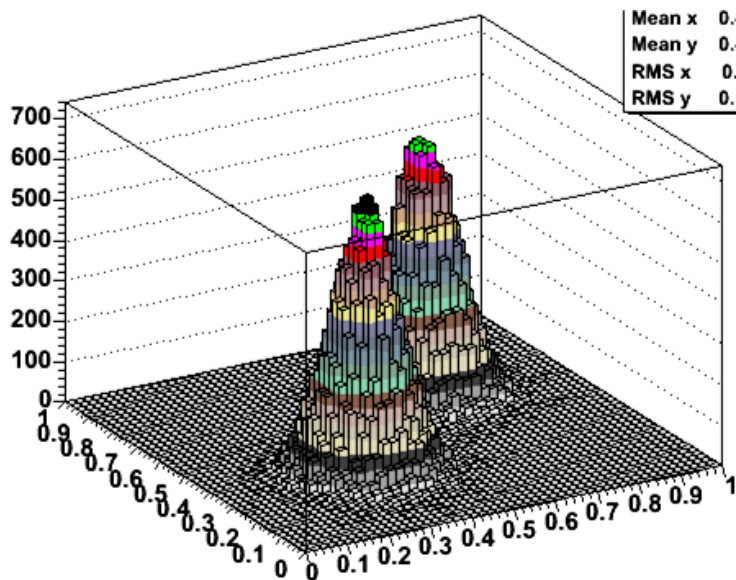
- 1) Find 'envelope function'  $g(x)$  that is invertible into  $G(x)$  and that fulfills  $g(x) \geq f(x)$  for all  $x$
- 2) Generate random number  $x$  from  $G$  using inversion method
- 3) Throw random number ' $y$ '
- 4) If  $y < f(x)/g(x)$  keep  $x$ , otherwise return to step 2



- PRO: Faster than plain accept/reject sampling  
Function does not need to be invertible
- CON: Must be able to find invertible envelope function

# Toy MC Generation in a nutshell

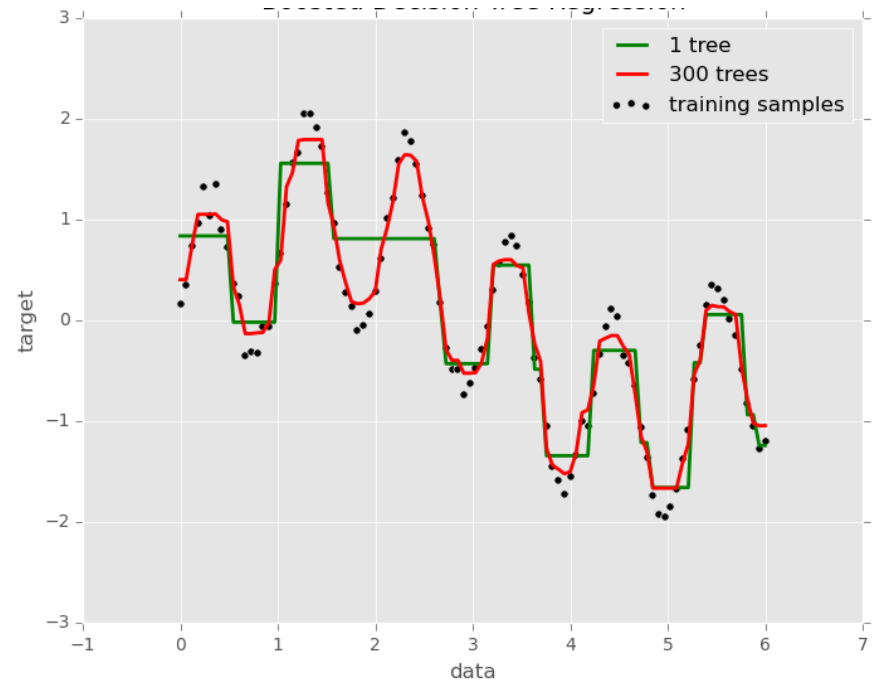
- General algorithms exists that can construct empirical envelope function
  - Divide observable space recursively into smaller boxes and take uniform distribution in each box
  - Example shown below from FOAM algorithm



# Parameter estimation (practicum)

# Model building and fitting the data (1/2)

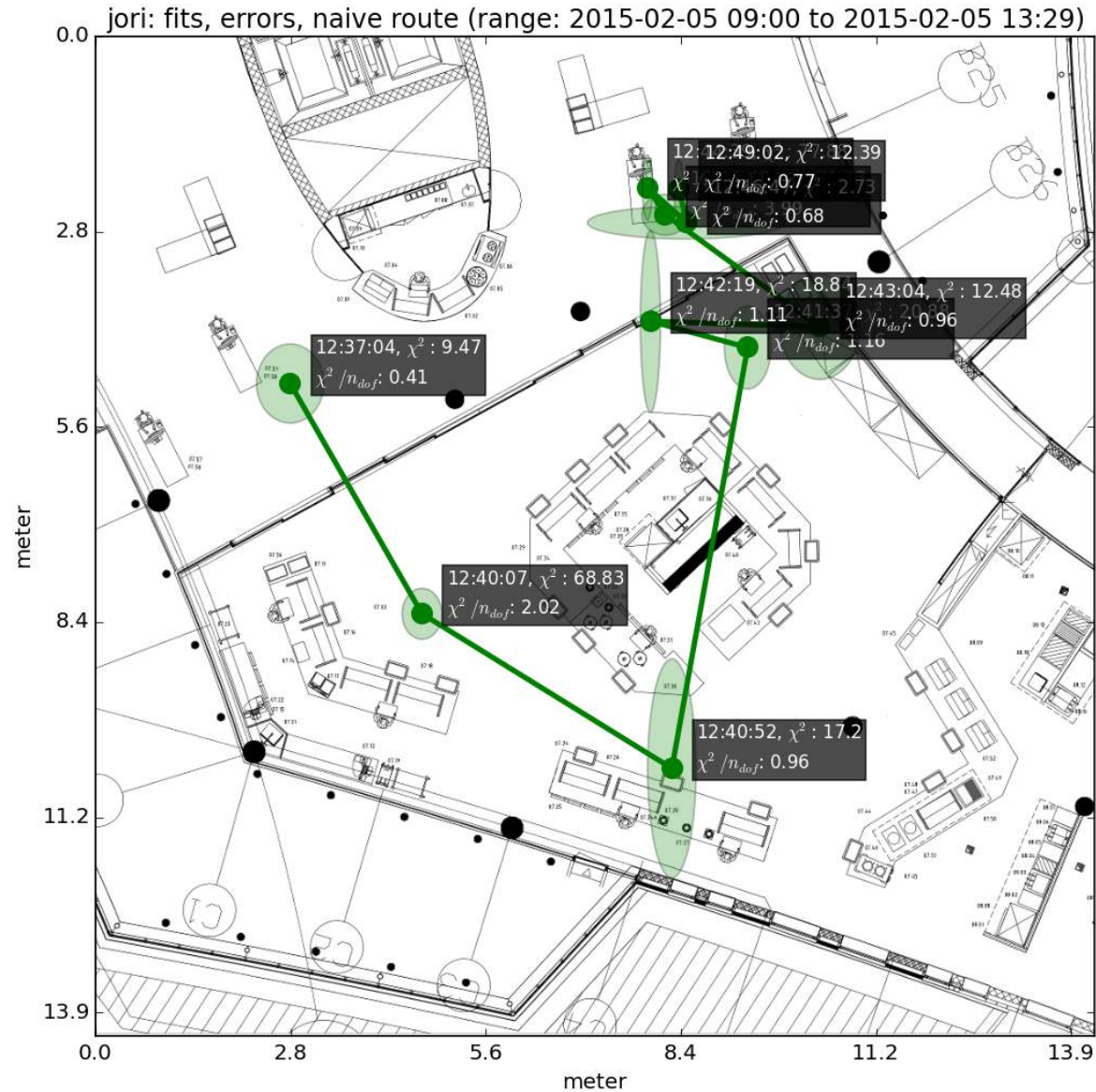
- Goal: find best model and model parameters that describe the data
  - **x** = observables  
(measured quantities)
  - **p** = model parameters  
(model/theory parameters)
- Example: determine amplitudes, frequencies, phases that best fit the measured data points (X,y).



$$f(x; \vec{p}) = A_1 \sin(\omega_1 x + \phi_1) + A_2 \sin(\omega_2 x + \phi_2)$$

$$\vec{p} = (A_1, A_2, \omega_1, \omega_2, \phi_1, \phi_2)$$

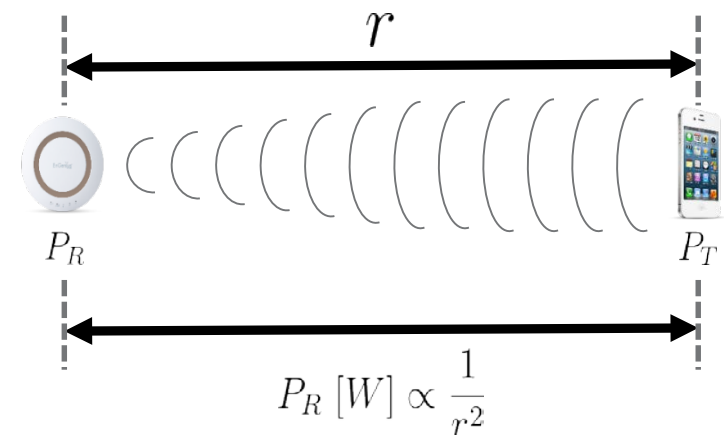
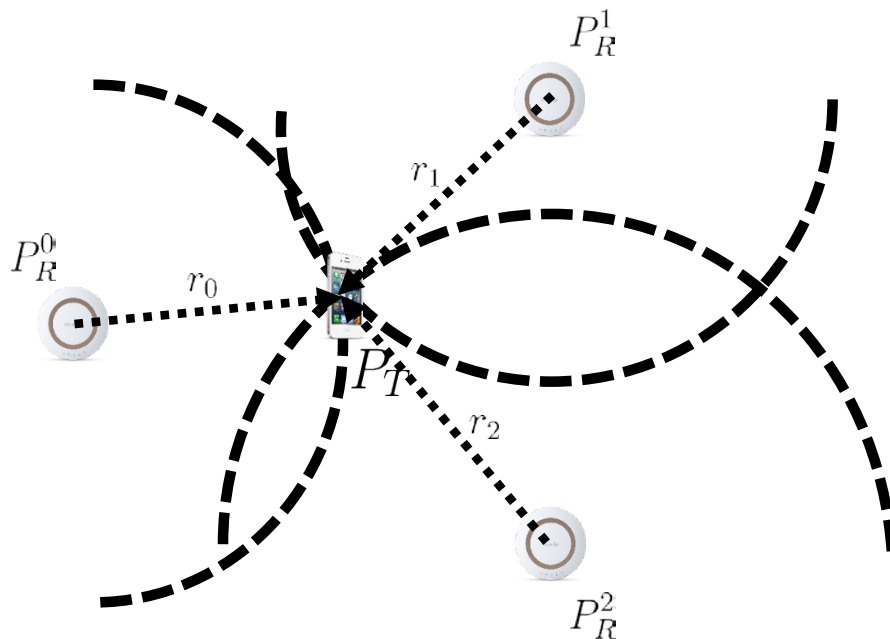
# Tracking a smart phone through a restaurant



→ Part of Computer Practicum exercises!

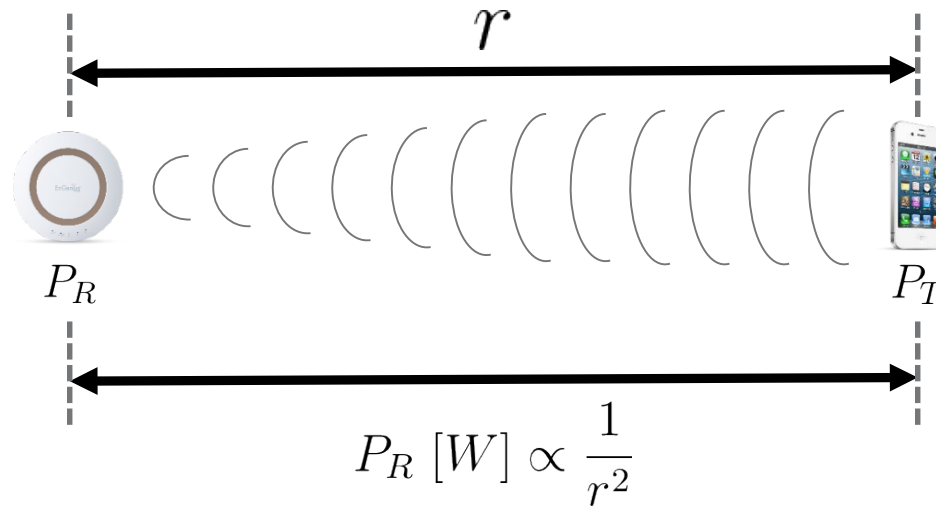
## Model building and fitting the data (2/2)

- Goal: find best model parameters that describe the data
  - $\mathbf{x}$  = observables (measured quantities)
  - $\mathbf{p}$  = model parameters (model/theory parameters)
- Example: determine location of phone ( $x, y$ ) that best fits the measured signal strengths ( $P_R$ ).





# Friis: Free Space Transmission Equation



This is going to be our model

$$P_R [dBm] = P_T + 10 \log \left( \frac{c}{4\pi f r} \right)^2$$

- $P_R$  – transmission power at receiver
- $P_T$  – transmission power
- $n = 2$  – line of sight, i.e. no obstacles
- $c$  – speed of light
- $f$  – frequency, 2.4 GHz or 5 GHz
- $r$  – distance from transmitter to receiver

Valid for:

- Line – of – sight: No obstacles that may lead to reflections, refractions, etc.
- $r > 0.4 \text{ m}$

(Details in computer practicum!)

# Parameter estimation (general)

## Followed here: $\chi^2$ approach

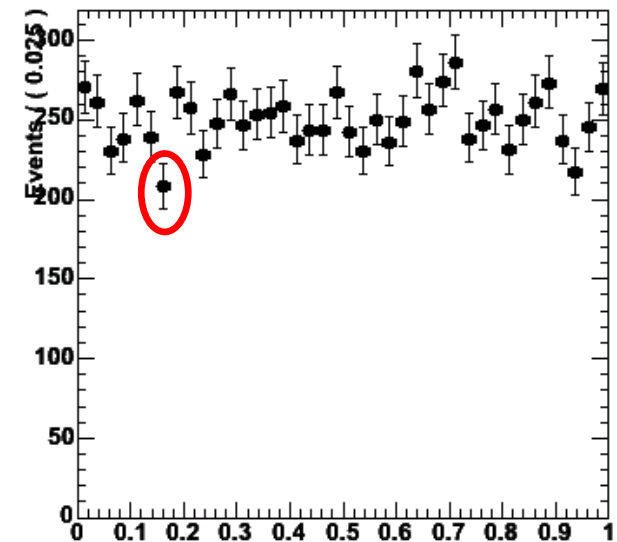
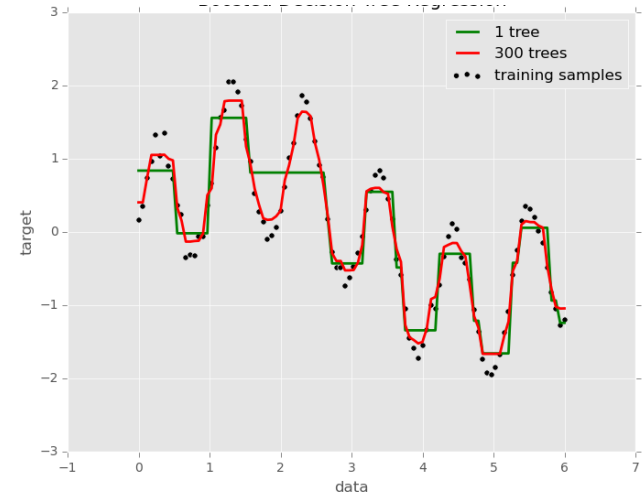
- Each individual data point / each bin of data distribution described with a Gaussian (G) probability function:

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

- Only in the limit that the error on  $x_i$  is truly Gaussian
  - i.e. need  $n_i > 10$  if  $y_i$  follows a Poisson distribution
- Function  $f(x_i, p)$  describes functional behaviour over the bins. E.g.:

$$f(x; \vec{p}) = A_1 \sin(\omega_1 x + \phi_1) + A_2 \sin(\omega_2 x + \phi_2)$$

$$f(x; \vec{p}) = Ax + B$$



x

# Definition of likelihood

- The *likelihood* is the value of a probability mass/density function **evaluated at the measured value of the observable(s)**
  - Note that likelihood is only function of parameters, not of observables

$$L(\vec{p}) = F(\vec{x} \equiv \vec{x}_{data}; \vec{p})$$

- Likelihood of seeing set of measurements? I.e. of seeing meas A and meas B and meas C and ... etc.
  - *Multiplication* of individual “chances”
- For a dataset that consists of multiple, *independent* data points, the product is taken:

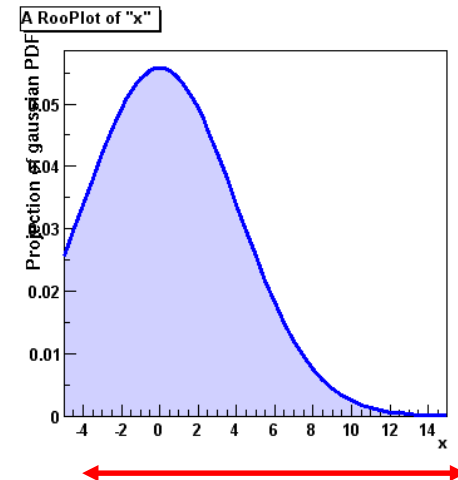
$$L(\vec{p}) = \prod_i F(\vec{x}_i; \vec{p}), \quad \text{i.e.} \quad L(\vec{p}) = F(x_0; \vec{p}) \cdot F(x_1; \vec{p}) \cdot F(x_2; \vec{p}) \dots$$

# General approach (for unbinned data)

- Approach so far ( $\chi^2$  fit) very empirical:  
Function  $f(x,y)$  can be any arbitrary function, with Gaussian probability function.
- General: we can characterize data distributions with *probability (density) functions*  $F(x;p)$ 
  - Many statistical techniques (Likelihood, Bayesian, Frequentist) require a more formal approach to data modeling through probability (density) functions
- Properties
  - Normalized to unity with respect to observable(s)  $x$
  - Positive definite –  $F(x;p) \geq 0$  for all  $(x,p)$

$$\int F(\vec{x}; \vec{p}) d\vec{x} \equiv 1$$

$$F(\vec{x}; \vec{p}) \geq 0$$



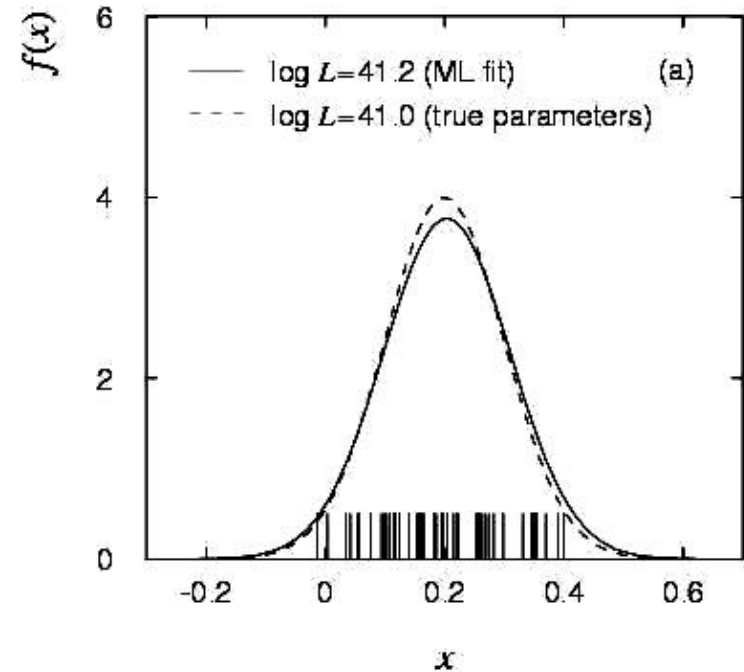
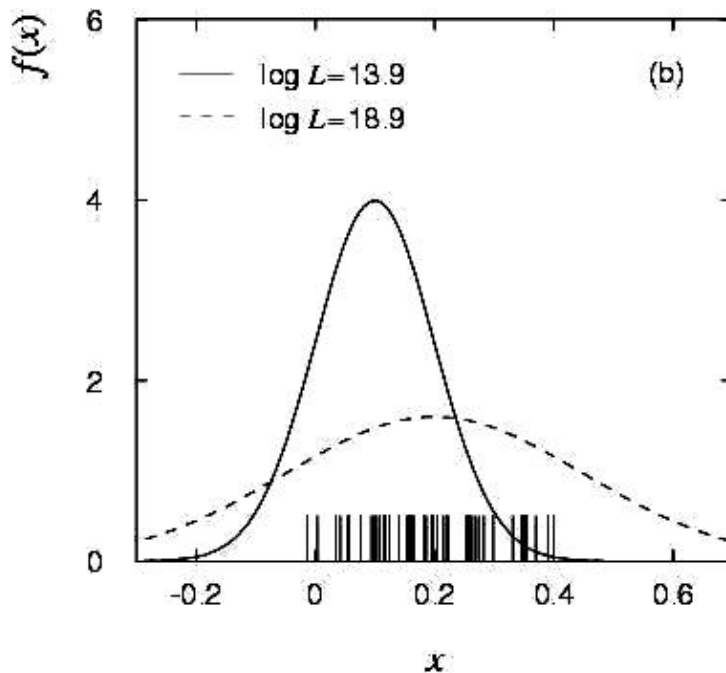
# Probability, Probability Density, and Likelihood

- For discrete observables we have probabilities instead of probability densities
  - Unit Normalization requirement still applies
- Poisson *probability*  $P(n|\mu) = \mu^n \exp(-\mu)/n!$ 
  - In Poisson case, suppose  $n=3$  is observed.  
Substituting  $n=3$  into  $P(n|\mu)$  yields the  
*likelihood function*  $L(\mu) = \mu^3 \exp(-\mu)/3!$
- Gaussian *probability density function (pdf)*  $p(x|\mu, \sigma)$ :  
 $p(x|\mu, \sigma)dx$  is differential of probability  $dP$ .
- Key point is that  $L(\mu)$  is *not* a probability density in in the fit parameters, such as  $\mu$ .
  - It is not a density!
  - However, it is *proportional* to the total probability.



# Parameter estimation using Maximum Likelihood

- Likelihood is high for values of  $\mathbf{p}$  that result in distribution similar to data



- Define the **maximum likelihood** (ML) estimator(s) to be the parameter value(s) for which the likelihood is maximum.

# Parameter estimation – Maximum likelihood

- Computational issues
  - For convenience the **negative log of the Likelihood** is often used as addition is numerically easier than multiplication

$$-2 \ln L(\vec{p}) = -\sum_i \ln F(\vec{x}_i; \vec{p})$$

- Maximizing  $L(p)$  equivalent to minimizing  $-2\log L(p)$
- In practice, find point  $p$  where derivatives are zero:

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

- Maximizing Likelihood is equivalent to *minimizing*  $-2\log L$

# Relation between Likelihood and $\chi^2$ estimator

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

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

Probability Density Function in  $p$  for single data point  $x_i(\sigma_i)$  and function  $f(x_i; p)$



Take log,  
Sum over all points  $(x_i, y_i, \sigma_i)$

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

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

- The  $\chi^2$  estimator follows from ML estimator
  - Only in the limit that the error on  $x_i$  is truly Gaussian
  - i.e. need  $n_i > 10$  if  $y_i$  follows a Poisson distribution
- Maximizing Likelihood is equivalent to *minimizing* the  $\chi^2$

$$\chi^2 = -2 \ln L(\vec{p})$$

# Variance on ML parameter estimates

- The ML estimator for the **parameter variance** is

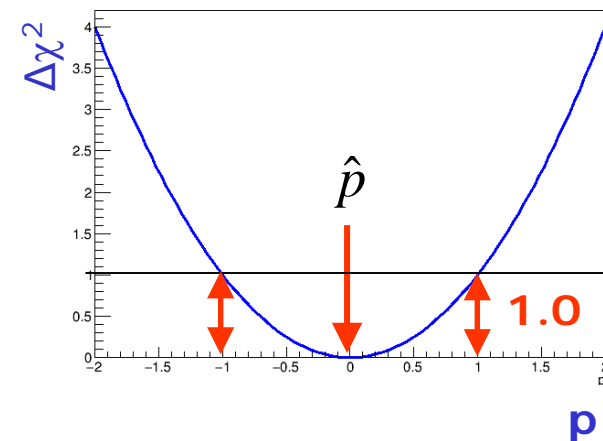
$$\hat{\sigma}(p)^2 = \hat{V}(p) \equiv \left( \frac{d^2 \ln L}{d^2 p} \right)^{-1} = 2 \left( \frac{d^2 \chi^2}{d^2 p} \right)^{-1}$$

- I.e. variance is estimated from 2<sup>nd</sup> derivative of  $-\log(L)$  at minimum

- Visual interpretation** of variance estimate

- Taylor expand  $\chi^2$  around minimum

$$\begin{aligned} \chi^2(p) &= \chi^2(\hat{p}) + \left. \frac{d\chi^2}{dp} \right|_{p=\hat{p}} (p - \hat{p}) + \frac{1}{2} \left. \frac{d^2 \chi^2}{d^2 p} \right|_{p=\hat{p}} (p - \hat{p})^2 \\ &= \chi^2_{\min} + \left. \frac{d^2 \chi^2}{d^2 p} \right|_{p=\hat{p}} \frac{(p - \hat{p})^2}{2} \\ &= \chi^2_{\min} + \frac{(p - \hat{p})^2}{\hat{\sigma}_p^2} \Rightarrow \chi^2(p \pm \sigma) = \chi^2_{\min} + 1 \end{aligned}$$



## $\chi^2$ and similar definitions

- Maximizing likelihood is equivalent to *minimizing* the  $\chi^2$  (in case of Gaussian uncertainties).
- Uncertainties  $\sigma_i$  are sometimes dropped from the  $\chi^2$  function, or set to 1.
  - In case measurement uncertainties  $\sigma_i$  are the same for all measurements  $i$ .
  - The measurement uncertainties are unknown.
  - Minimization function becomes: 
$$g(\vec{p}) = \sum_i (y_i - f(x_i; \vec{p}))^2$$
    - = Euclidian distance squared.
    - (Will get back to this.)
- Also known as: “least squares method”, or
- when fitting a linear function: “linear regression”
  - Easy to solve analytically.

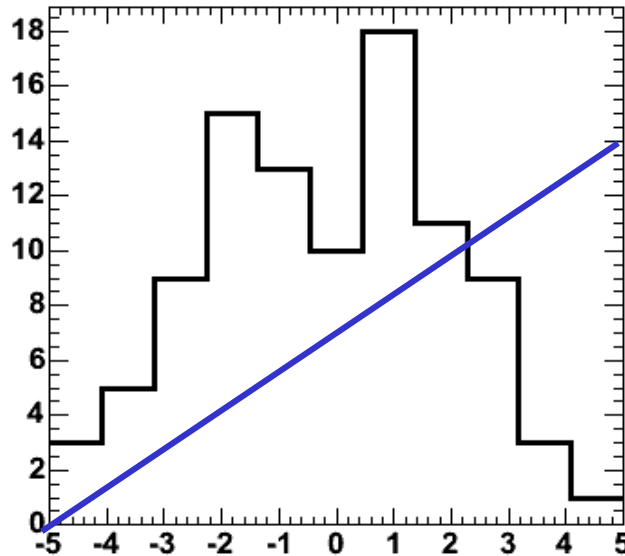
# $\chi^2$ versus unbinned Maximum Likelihood

- $\chi^2$  fit is fast and easy
  - Works fine at high statistics
  - Gives absolute goodness-of-fit indication
  - Make (incorrect) Gaussian error assumption on low statistics bins
  - Has bias proportional to  $1/N$
  - Misses information with feature size  $<$  bin size
- Full Maximum Likelihood estimators most robust
  - ***Describe each record individually. No bins of records.***
  - No Gaussian assumption made at low statistics
  - No information lost due to binning
  - Gives best error of all methods (especially at low statistics)
  - No intrinsic goodness-of-fit measure, i.e. no way to tell if 'best' is actually 'pretty bad'
  - Has bias proportional to  $1/N$
  - Can be computationally expensive for large  $N$
- Binned Maximum Likelihood
  - *In between solution of two methods above.*
  - Correct Poisson treatment of low statistics bins

# Judging Goodness of Fit

## Goodness of fit

- If assumed fit model is not capable of describing your data for any  $p$ , the *fit procedure* will often not complain.
- First step: always evaluate fit result by eye.



'Not a good fit'

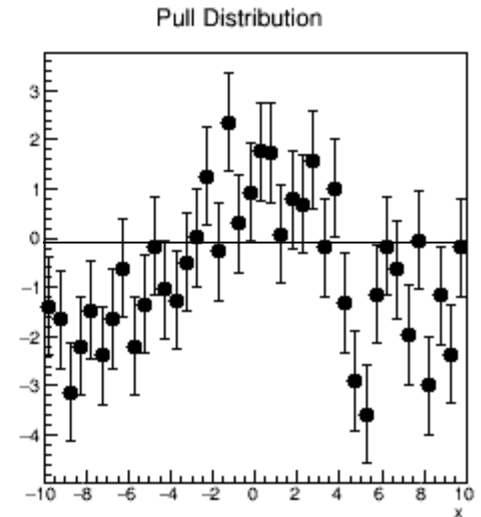
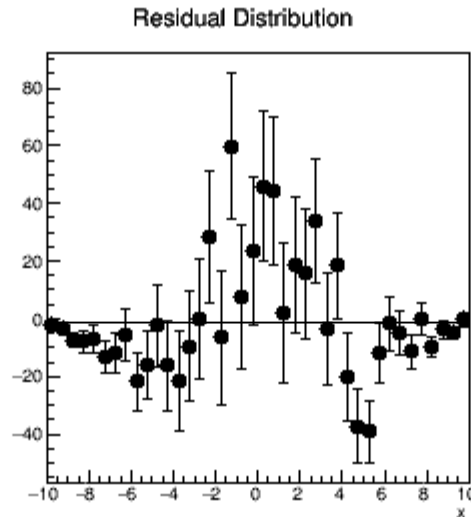
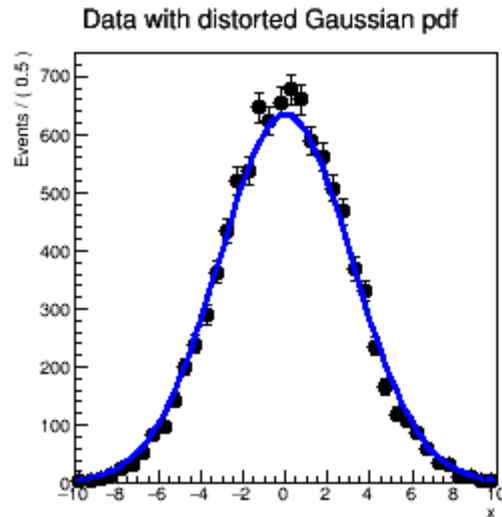


# Visual inspection (“chi by eye”)

Useful checks:

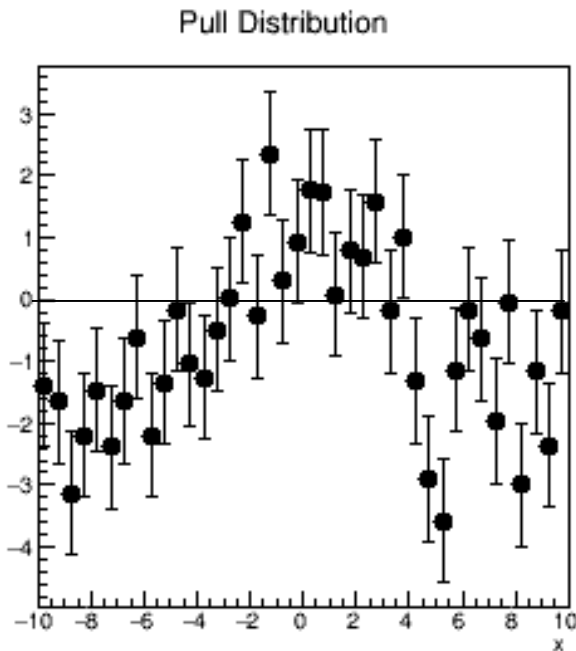
- Residual distribution. For each bin:  $y_i - f(x_i; \vec{p})$
- Pull distribution = “normalized residuals”.

Pull, for each bin: 
$$\chi_i = \frac{y_i - f(x_i; \vec{p})}{\sigma_i}$$

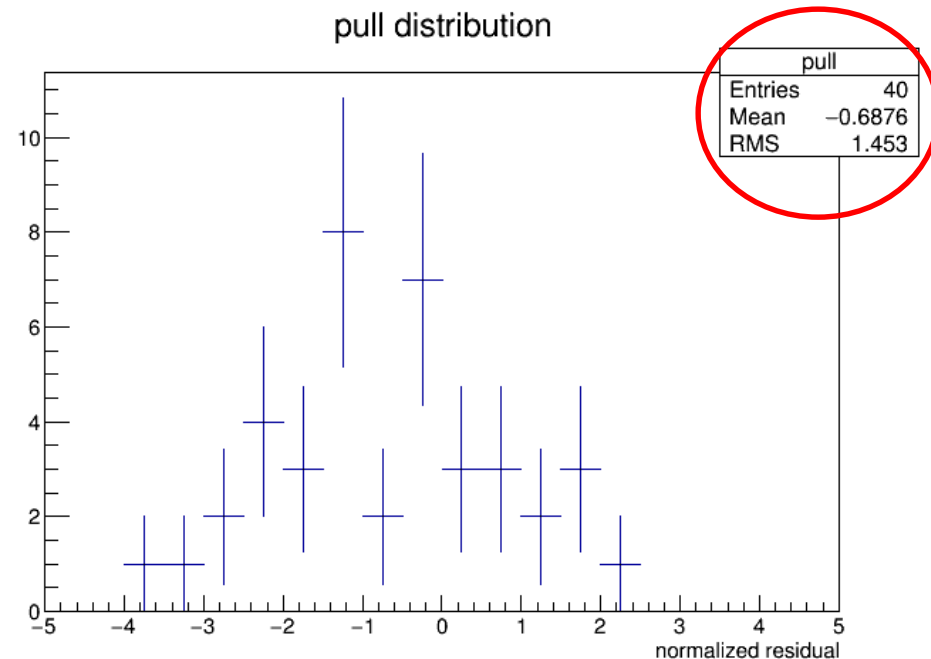
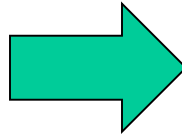


# Pull distribution – normalized residuals

- Pull distribution = distribution of normalized residuals



Projected  
on y-axis



- Properties of pull:
  - Mean is 0 if there is no bias
  - Width is 1 if error is correct

# Estimating and interpreting Goodness-Of-Fit

- Most common test: **the  $\chi^2$  test**

$$\chi^2 = \sum_i \left( \frac{y_i - f(\vec{x}_i; \vec{p})}{\sigma_i} \right)^2$$

- Take  $\chi^2$  value **after** minimization
- N = total number of data points / bins (i)
- If  $f(x)$  describes data then  $\chi^2 \approx N$ , if  $\chi^2 \gg N$  something is wrong

- How to quantify meaning of 'large  $\chi^2$ '?

- *What you really want to know: the **probability** that a **function** which does **genuinely describe the data** on N points would give a  **$\chi^2$  probability as large or larger** than the one you already have.*
- How to make a well calibrated statement for intermediate N

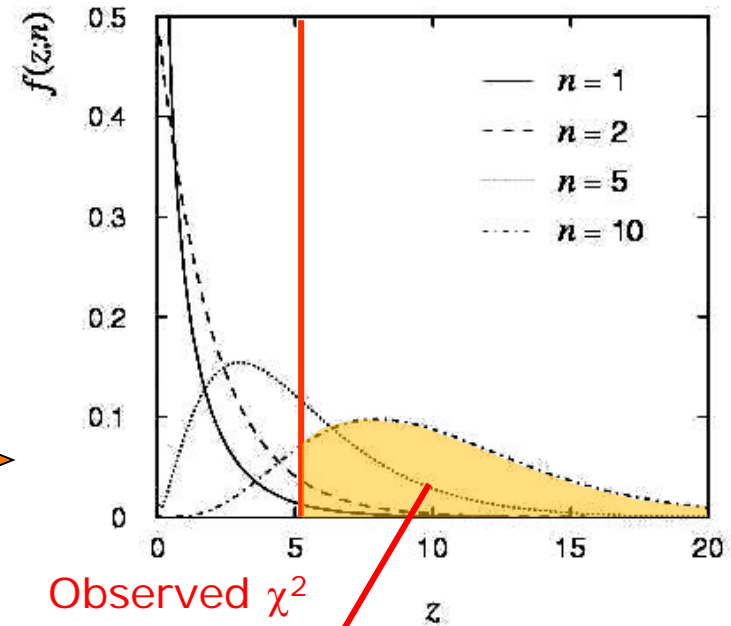
# How to quantify meaning of 'large $\chi^2$ '

- Probability distr. for  $\chi^2$  is given by

$$\chi^2 = \sum_i \left( \frac{y_i - \mu_i}{\sigma_i} \right)^2$$



$$p(\chi^2, N) = \frac{2^{-N/2}}{\Gamma(N/2)} \chi^{N-2} e^{-\chi^2/2}$$



Observed  $\chi^2$   
for  $n=10$

P = integral over shaded area

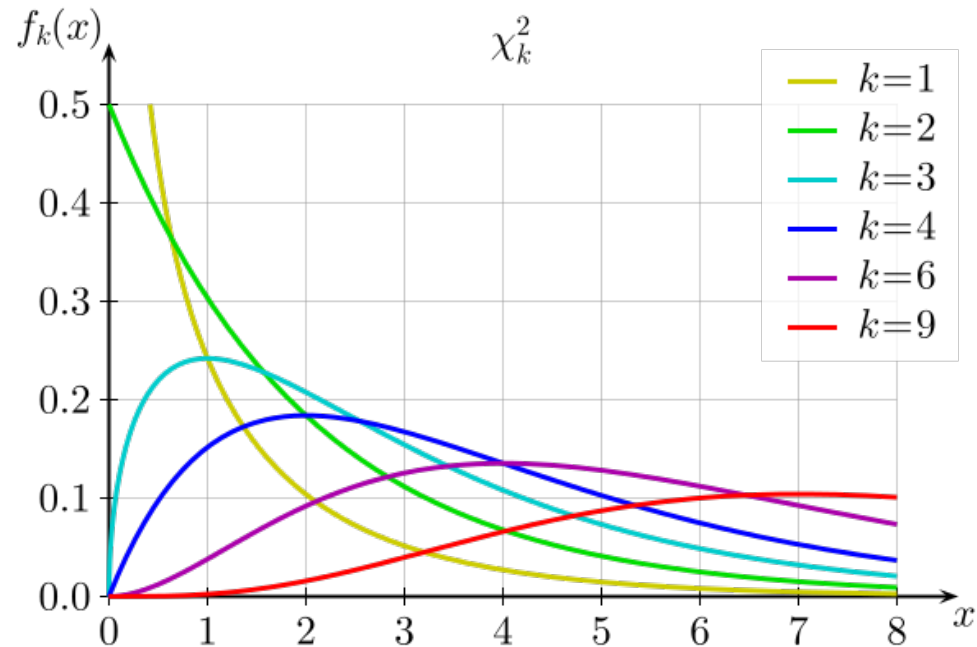
$$P(\chi^2; N) = \int_{\chi^2}^{\infty} p(\chi'^2; N) d\chi'^2$$

- Good news: Integral of  $\chi^2$  pdf is analytically calculable!

# Properties of $\chi^2$ distribution

- Mean:
- Variance:

$$N_{dof}$$
$$2 \times N_{dof}$$



- Goodness-of-fit measure:  $\chi^2 \approx 1$  per degree of freedom
  - Normalized  $\chi^2 = \chi^2_{N_{dof}} = \chi^2 / N_{dof}$
- If  $\chi^2_{N_{dof}} \gg 1$ : fit model is likely incorrect
  - Model is wrong, or measurement uncertainties too small.
- If  $\chi^2_{N_{dof}}$  close to zero: fit model works too well.
  - Model is too flexible (overfitting, or uncertainties too large.)

# Goodness-of-fit – $\chi^2$

- Example for  $\chi^2$  probability

- Suppose you have a function **f(x;p)** which gives a  $\chi^2$  of 20 for 5 points (histogram bins).
- Not impossible that **f(x;p)** describes data correctly, just unlikely

- How unlikely?  $\int_{20}^{\infty} p(\chi^2, 5) d\chi^2 = 0.0012$

- Note: If function has been fitted to the data

- Then you need to account for the fact that parameters have been adjusted to describe the data

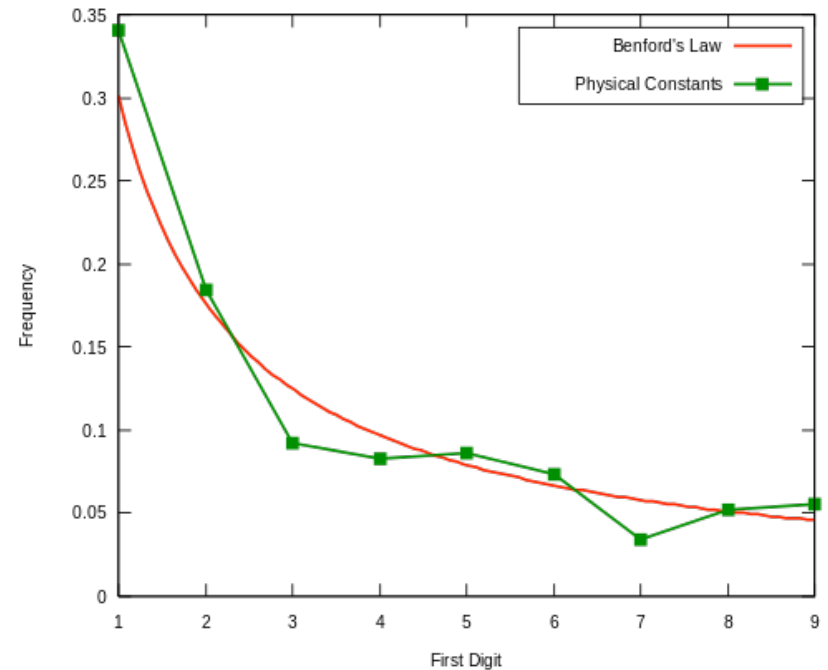
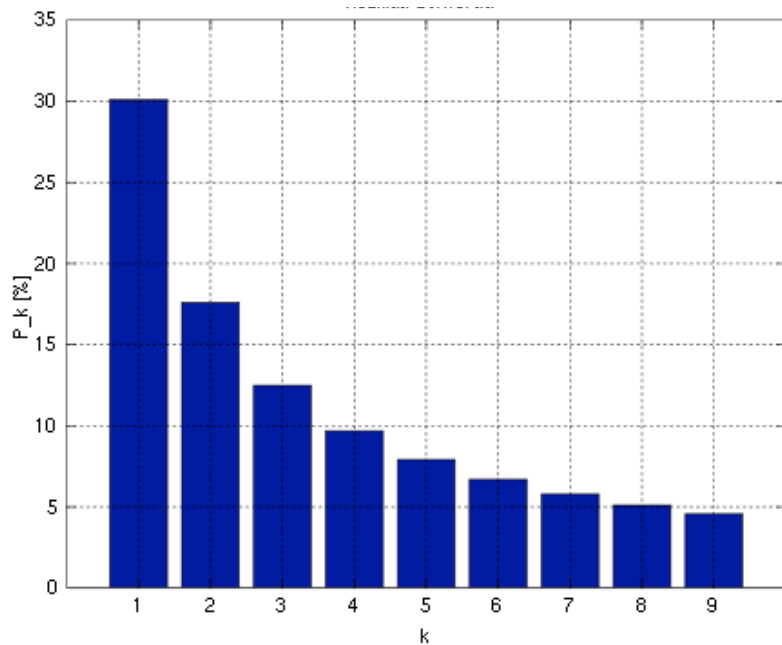
$$N_{\text{d.o.f.}} = N_{\text{data}} - N_{\text{params}}$$

- Practical tips

- To calculate the probability in '**Prob(chi2, N<sub>dof</sub>)**'
- Clearly,  $N_{\text{dof}} \geq 1$  in order to have any residual  $\chi^2$  value.
  - For  $N_{\text{dof}} = 0$ ,  $\chi^2 = 0$ .

# Example: Benford's law

- Law of first digits.



- Number of degrees of freedom =  $9 - 1 = 8$

## Example: Benford's law in finance data

- “Fact and Fiction in EU-Governmental Economic Data”
  - Analysis of National deficit data.
  - <http://www.cesruc.org/uploads/soft/130301/1-1303011Z221.pdf>
- Benford's law of first digits in (financial) numbers:

**Table 2** First significant digit distribution for the aggregate dataset

Digit	Benford	EU-27
1	0.3010	0.2990
2	0.1761	0.1810
3	0.1249	0.1323
4	0.0969	0.1014
5	0.0792	0.0765
6	0.0669	0.0663
7	0.0580	0.0543
8	0.0512	0.0467
9	0.0458	0.0424



# Benford's law (continued)

**Table 3**  $\chi^2$  statistic for euro countries

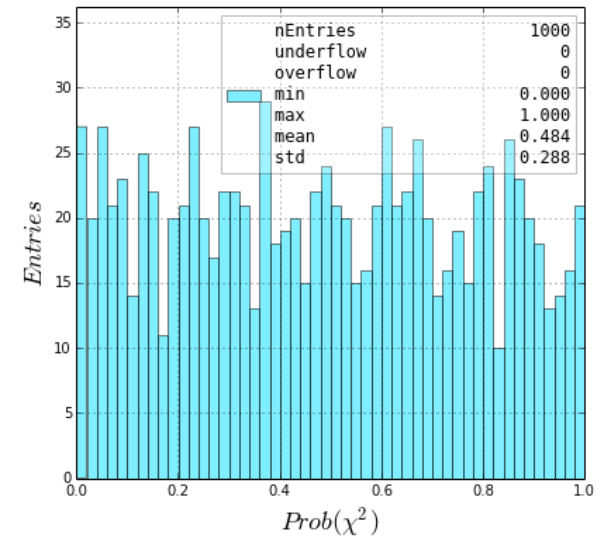
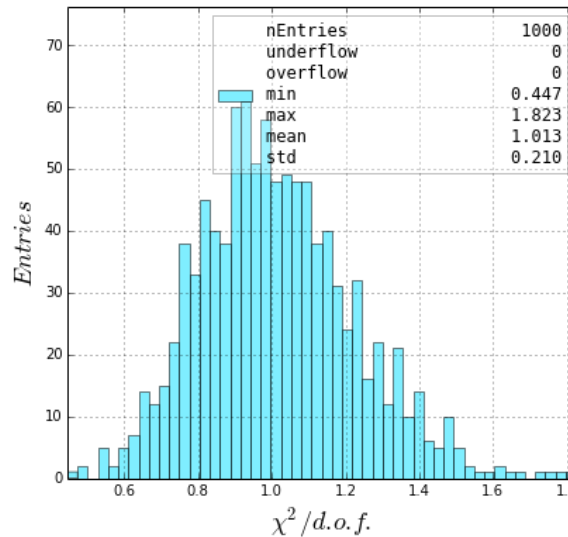
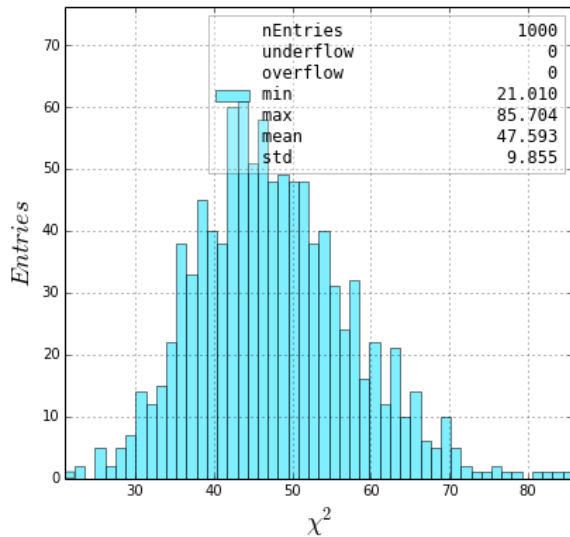
	Mean size	1999	2000	2001	2002	2006	2007	2008	2009	Mean $\chi^2$
Greece	135.55	7.70	26.33**	16.55**	20.50**	14.53	15.85**	27.88**	27.61**	17.74
Belgium	148.73	14.10	14.14	11.67	18.01**	16.98**	6.48	8.09	12.40	17.21
Austria	139.36	14.73	32.00**	27.13**	7.84	14.96	4.69	7.76	9.89	15.25
Ireland	137.73	14.51	15.00	9.78	18.34**	9.14	18.74**	17.04**	23.90**	14.60
Finland	150.00	6.29	11.66	17.16**	19.01**	17.50**	13.48	19.38**	18.34**	13.78
Slovakia	132.91	4.37	21.01**	13.19	11.30	14.45	18.64**	23.82**	8.89	12.40
Germany	149.18	6.68	12.88	8.89	19.21**	16.35**	14.05	15.89**	16.33**	12.37
Italy	134.27	16.45**	13.55	5.47	13.13	14.57	14.76	9.12	12.45	12.37
Cyprus	117.55	7.80	12.73	17.07**	9.74	9.65	8.46	7.08	16.96**	11.96
Slovenia	134.64	12.86	5.69	19.34**	10.46	4.98	9.99	17.20**	6.35	11.44
Spain	137.09	11.50	9.92	13.63	10.88	14.70	2.50	6.04	11.09	11.36
France	149.00	7.75	11.28	23.33**	8.39	11.30	8.73	6.62	21.24**	11.11
Malta	114.91	16.66**	9.94	2.80	4.96	15.57**	21.31**	7.57	9.65	10.78
Luxembourg	123.82	23.25**	16.99**	3.86	8.28	9.36	3.34	6.89	8.87	10.33
Portugal	149.91	28.60**	12.43	22.64**	5.50	6.09	4.37	4.95	10.14	10.19
Netherlands	139.64	6.46	6.06	11.08	3.32	10.51	12.89	5.03	9.62	7.83

# Estimation of measurement uncertainty

- Remember: goodness-of-fit measure –  $\chi^2 \approx 1$  per d.o.f
- If you don't know your data uncertainties (like in a least squares fit), then the  $\chi^2_{\min}$  value of your fit provides an estimate.
- Rough estimate of  $\sigma_i$  is obtained by setting  $\chi^2_{\text{Ndof}} = 1$ , by hand.
- → The correction factor  $\approx 1 / \sigma^2$
- *Computer practicum: estimate resolution of wifi sensors.*

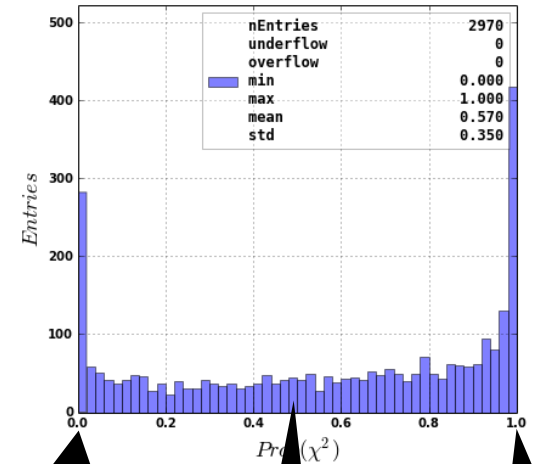
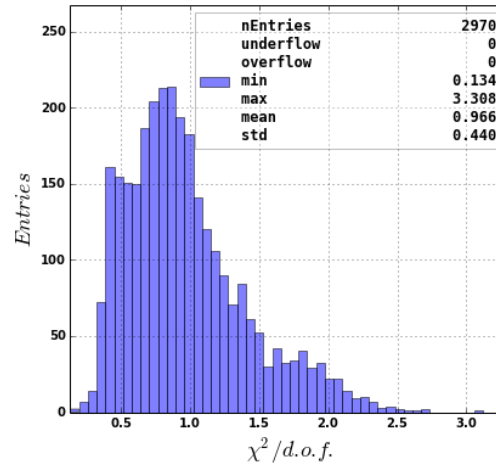
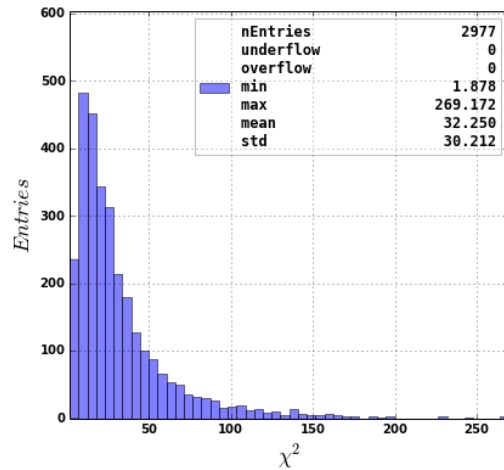
# Goodness-of-fit – $\chi^2$ goodness of fit

- Probability distribution should then (ideally) be *uniform*.
- Below: results from ideal Monte Carlo simulation.



- $E[\chi^2]$  gives the number of degree of freedom: 50 measurements – 3 parameters = 47
- $E[\chi^2/d.o.f.] = 1$
- $Prob(\chi^2)$  is the probability of finding a  $\chi^2$  that is equal or worse than this  $\chi^2$ . Indicates how well our model describes the data.
  - Peak at zero means that our model describes the data “poorly”
  - Peak at one means that our model is too “good”
  - Another problem: slope in p-value distribution.

# Actual wifi-tracking data – $\chi^2$ goodness of fit



Model doesn't  
describe data well

Uniform, OK-ish

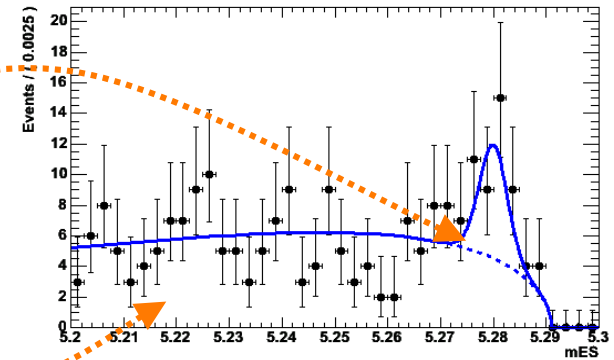
Too well ...

Possible reasons why fit is sometimes "off":

- Assume line – of – sight measurements, i.e. ignore scattering, reflections. These can lead to destructive/constructive interference
- Some sensors are different than others
- Sensors are mis-aligned, i.e. their coordinates are "off".
- Directionality of the sensors
- ...

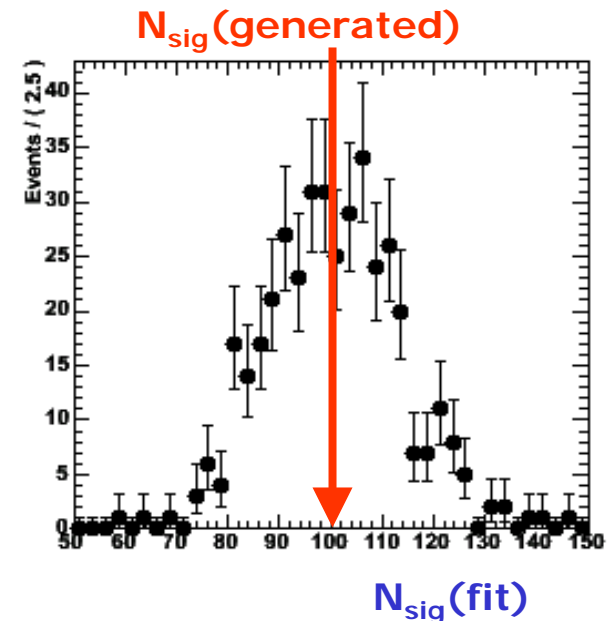
# Fit Validation Study – Practical example

- Example fit model in 1-D
  - Signal component is Gaussian centered here
  - Background component is Argus function (models phase space near kinematic limit)



$$F(m; N_{\text{sig}}, N_{\text{bkg}}, \vec{p}_S, \vec{p}_B) = N_{\text{sig}} \cdot G(m; p_S) + N_{\text{bkg}} \cdot A(m; p_B)$$

- Fit parameter under study:  $N_{\text{sig}}$ 
  - Results of simulation study:  
1000 experiments  
with  $N_{\text{SIG}}(\text{gen}) = 100$ ,  $N_{\text{BKG}}(\text{gen}) = 200$
  - Distribution of  $N_{\text{sig}}(\text{fit})$  .....→
  - This particular fit looks unbiased...



# Fit Validation Study – The pull distribution

- What about the validity of the error?

- Distribution of error from simulated experiments is difficult to interpret...
- We don't have equivalent of  $N_{\text{sig}}(\text{generated})$  for the error

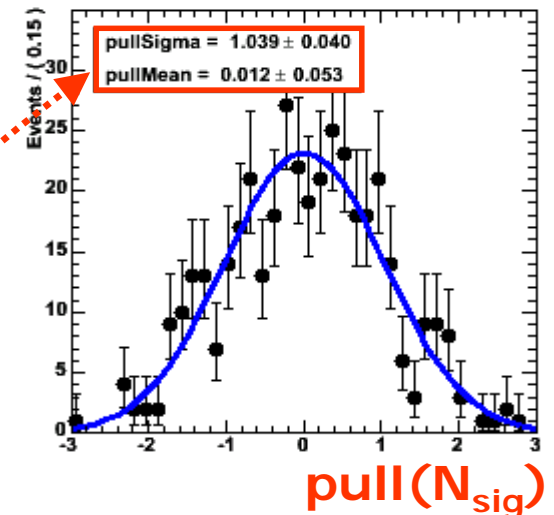
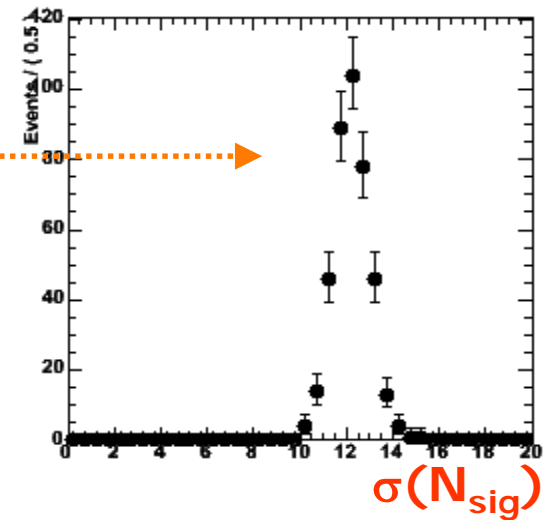
- Solution: look at the *pull distribution*

- Definition: 
$$\text{pull}(N_{\text{sig}}) = \frac{N_{\text{sig}}^{\text{fit}} - N_{\text{sig}}^{\text{true}}}{\sigma_N^{\text{fit}}}$$

- Properties of pull:

- Mean is 0 if there is no bias
- Width is 1 if error is correct

- In this example: no bias, correct error within statistical precision of study

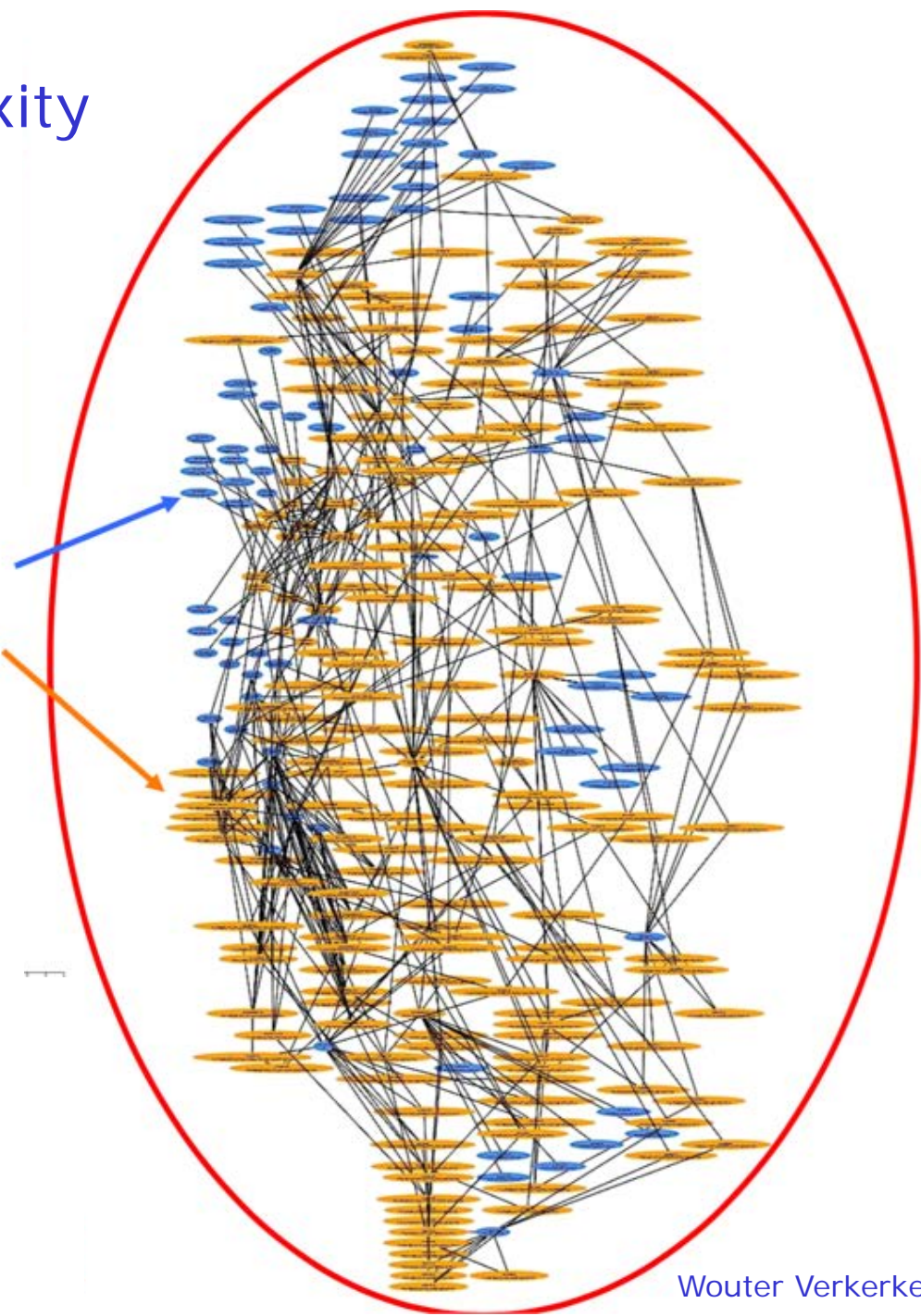


# **Model building & Dimensionality**

# Model building complexity

- Fit models can be complex!

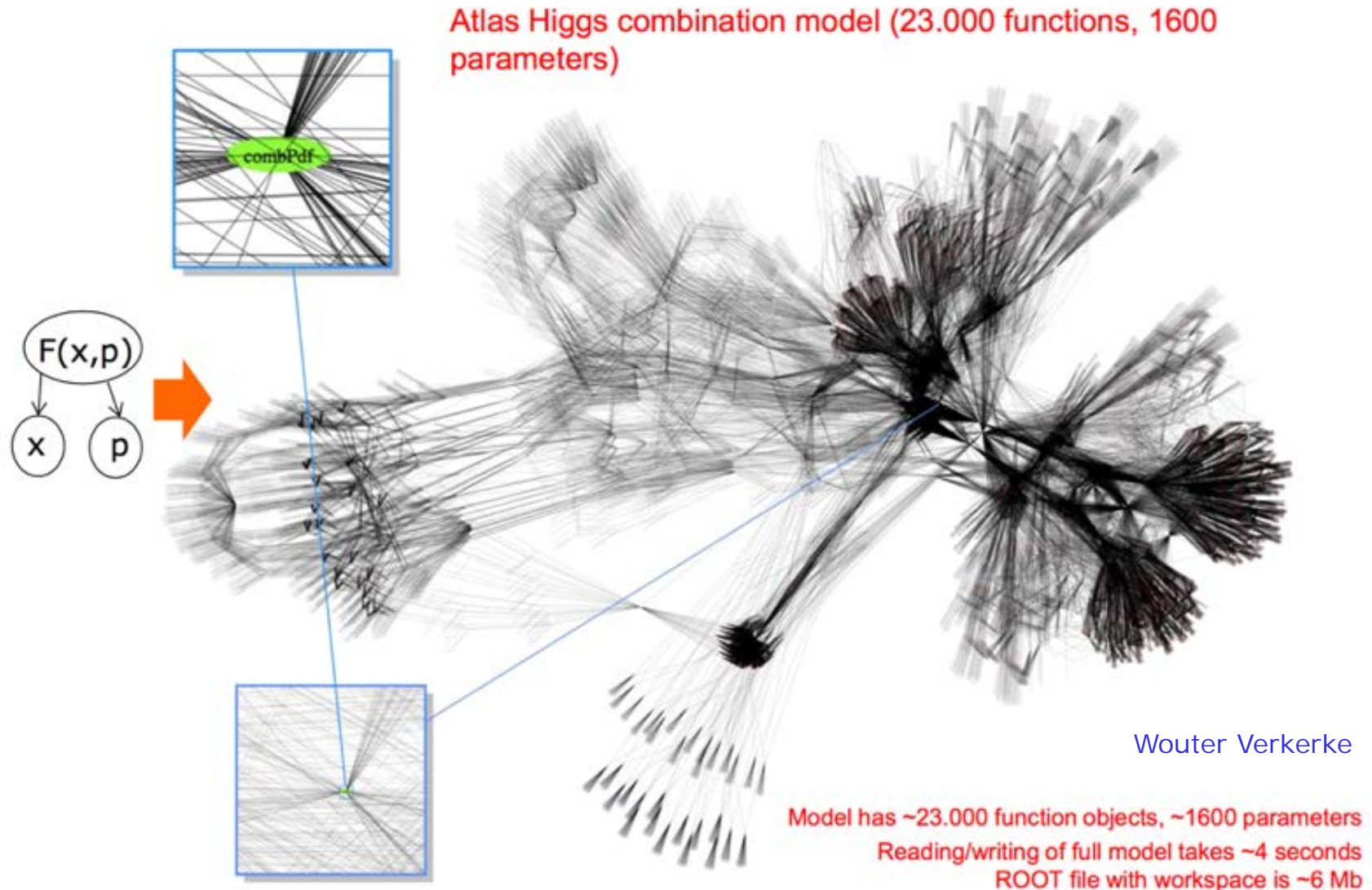
variables  
function objects



- [Graphical example of model complexity to describe data of (one channel of) Higgs particle decay.]



# Example of model building complexity: Statistical combination of all Higgs decay channels



*The discovery of the Higgs has been one of the most complex data analysis challenges performed ever!*

# Degrees of Freedom & Dimensionality

Every fit model, also Machine Learning alg, has free parameters that have to be fit / trained / estimated from data. E.g:

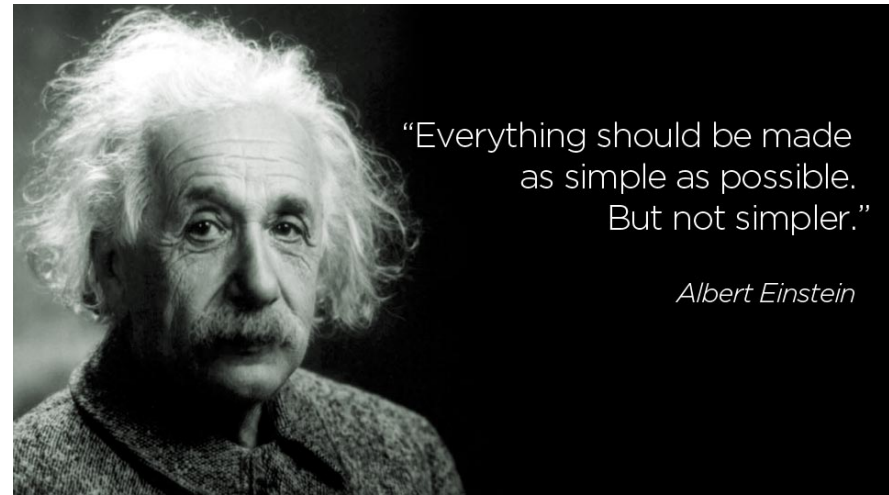
- Mean and variance in a Gaussian probability distribution
- Slope and offset of a linear trend
- Connection weights in a Neural Network
- Fourier coefficients of a periodic signal, etc.

***In general, with every added parameter the model can describe more realities, and more closely describe the data. E.g:***

- Straight line vs. parabola
- Single layer vs. multi-layer perceptron
- First order linear differential equation (exponential decrease) vs. second order (exponential decrease and periodic signals)

***But with every added parameter the model can also model more easily statistical fluctuations/noise in the data!***

# Dimensionality



- You need sufficient data to fit/train/estimate all of free parameters of the model.
- **Dimensionality: you can't have more parameters in your model than the number of data point that are being fit.**
- The more parameters your model contains, with the same amount of data, the more closely they could describe reality ... ("over-fitting")
- ... but the more uncertain your parameter estimates of them will be.  
... and (quite possibly) your entire model.
- **General model building tip: keep it simple!**

# Dimensionality

Don't dress up [your model] like these guys



But remember the true meaning of KISS = Keep It Simple, Stupid!

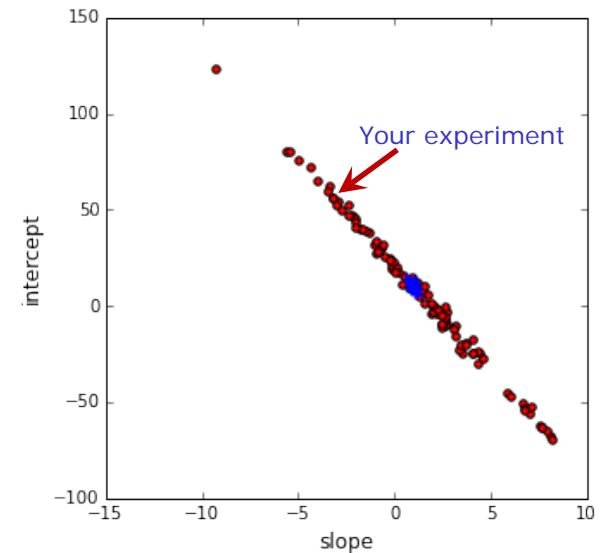
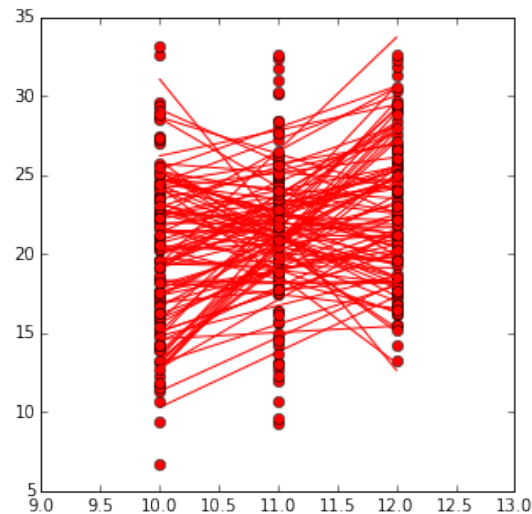
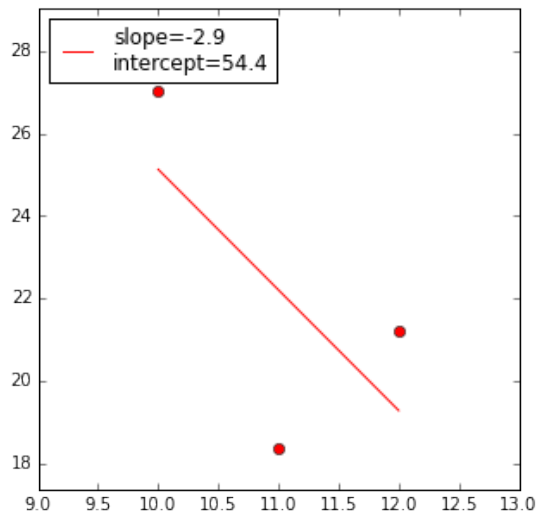
Avoid many free parameter models whenever possible (e.g. machine learning), by building a model from principles and/or by making assumptions.

E.g.

- The IQ of people is approximately Gaussian (2 parameters)
- The number of web visits is a Poisson distribution (1 parameter)
- The time until a return visit is a Weibull distribution (1 or 2 parameters)

# Dimensionality

- For every parameter introduced you need enough data to estimate/fit it
- Example: linear regression (2 parameters) on 3 data points

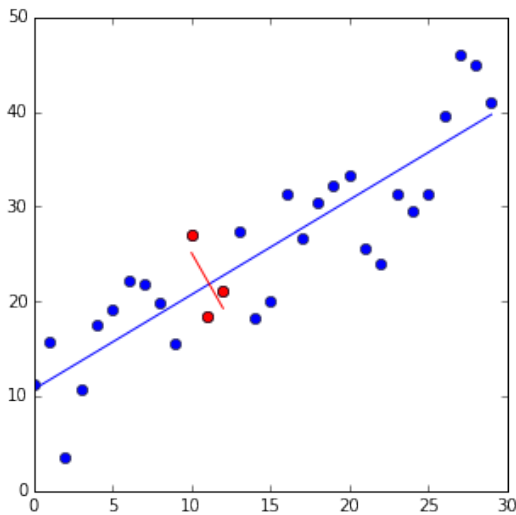


The fit works, but:

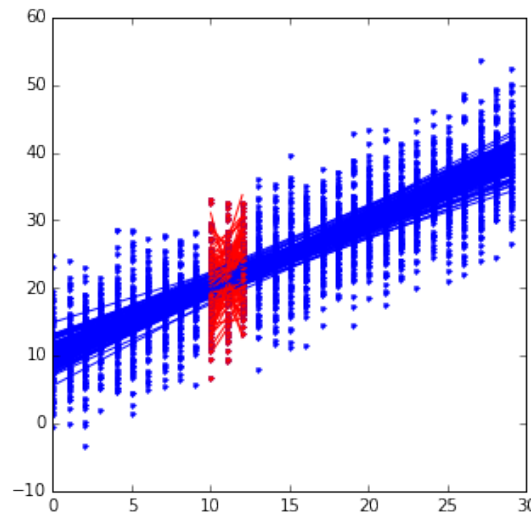
- If you would have repeated the experiment, completely different lines would have come up.
- The fit returns very large fitted uncertainties on the slope and intercept.

# Dimensionality

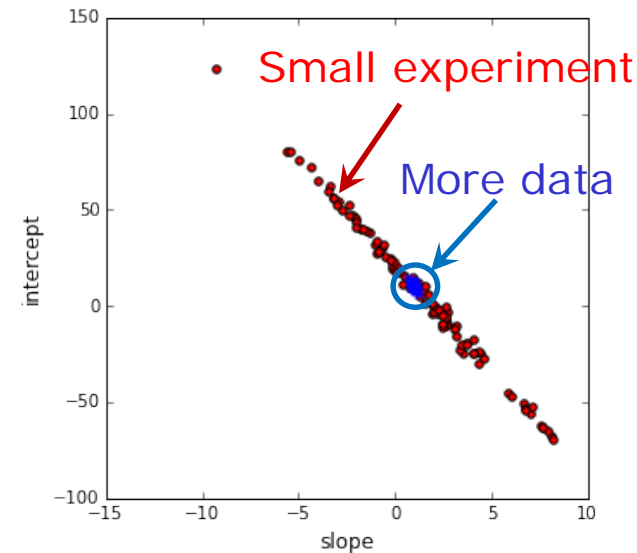
- For every parameter introduced you need enough data to estimate/fit it.
- Example: linear regression (2 parameters) on 3 data points



If you would have had more data, a completely different line would have come up.



But if you would have had more data and repeated the experiment, similar lines would have come up. Which means small uncertainties on the slope and intercept.



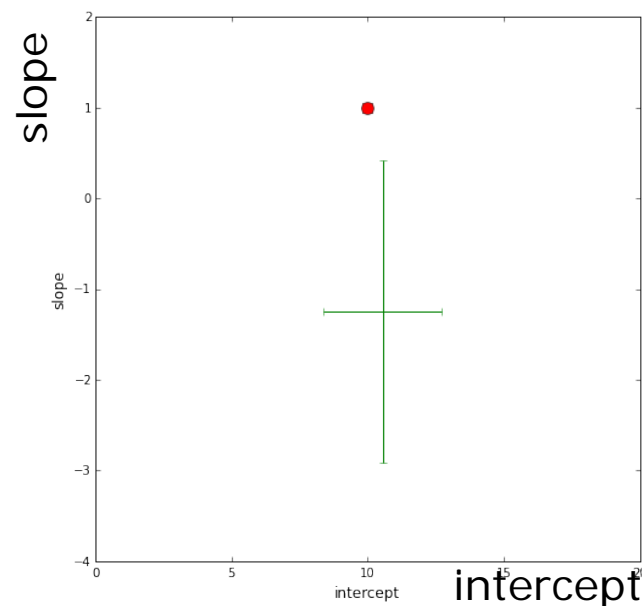
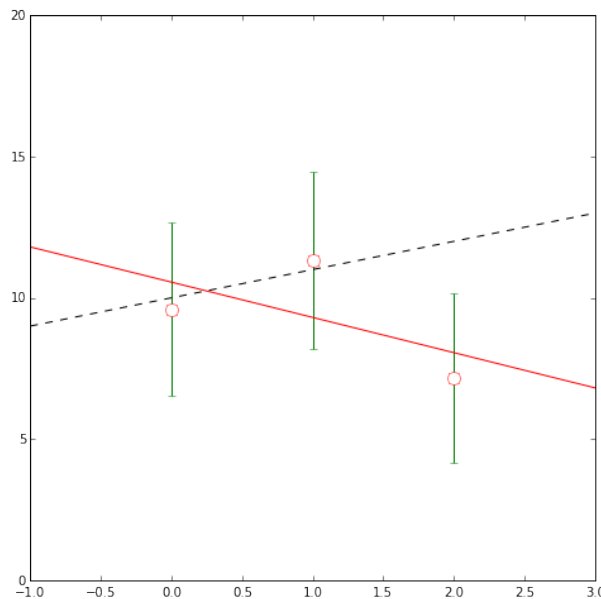
# Fit parameter uncertainties

**Remember: every data point (= measurement) has an uncertainty**

In example below:

- The dotted line is a linear reality. Its intercept and slope are the dot in the right hand plot
- Three data points are generated from the linear reality with noise/fluctuations. The error bars are the uncertainties on the generated points due to the noise.
- The red line is a chi-square fit through the generated points. Its intercept and slope are in the right hand plot *together with their fit uncertainties*.

***Need to include fit parameter uncertainties to judge if result found is consistent with underlying, true model.***



*(parameter correlation set to zero in example)*

# Parameter estimation in practice

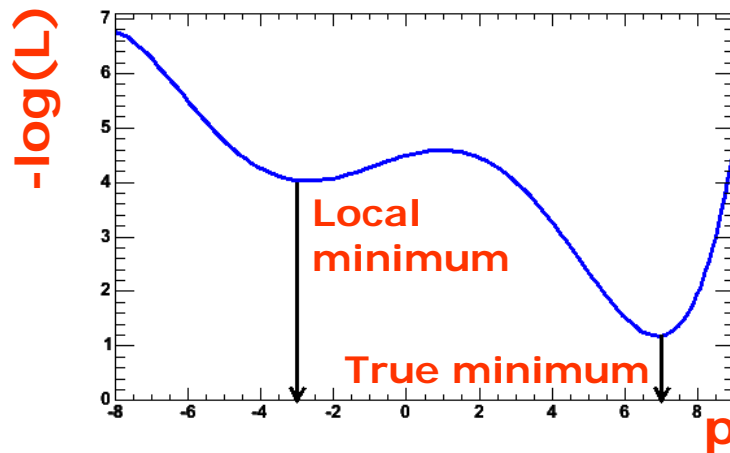


## Practical estimation – Numeric $\chi^2$ and $-\log(L)$ minimization

- For most data analysis problems minimization of  $\chi^2$  or  $-\log(L)$  **cannot be performed analytically**
  - [Only analytically for linear models.]
  - Need to rely on numeric/computational methods
  - In  $>1$  dimension **generally a difficult problem!**
- But no need to worry – great software exists to solve this problem for you:
  - In SciPy, ROOT, NumAlg, etc.
  - **Function minimization workhorse in HEP many years: MINUIT**
  - MINUIT does function minimization and error analysis
    - It produces a lot of useful information, that is sometimes overlooked
- Next: will look in a bit more detail into typical fitter output and functionality.

## Numeric $\chi^2$ / $-\log(L)$ minimization – Proper starting values

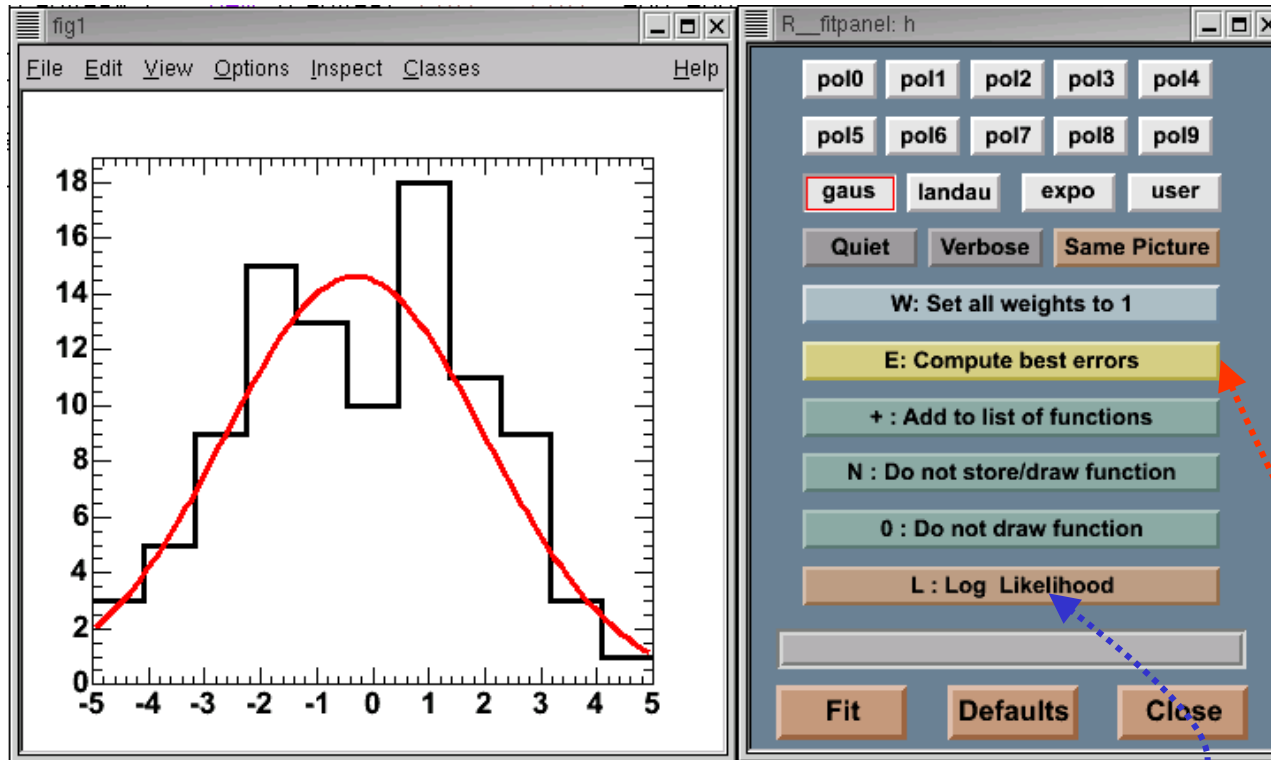
- For all but the most trivial scenarios it is not possible to automatically find reasonable starting values of parameters
  - So you need to supply good starting values for your parameters



*Reason: There may exist multiple (local) minima in the likelihood or  $\chi^2$*

- Supplying good initial uncertainties on your parameters helps too
- Reason: Too large error will result in minimizer coarsely scanning a wide region of parameter space. It may accidentally find a far away local minimum

# What happens behind the scenes



Example of  
interactive fit  
in ROOT

- What happens in MINUIT behind the scenes
  - 1) Find minimum in  $-\log(L)$  or  $\chi^2$  – MINUIT function MIGRAD
  - 2) Calculate errors on parameters – MINUIT function HESSE
  - 3) Optionally do more robust error estimate – MINUIT function MINOS

# Logging output: Minuit function MIGRAD

- Purpose: find minimum

Progress information,  
watch for errors here

\*\*\*\*\*

\*\* 13 \*\*MIGRAD 1000 1

\*\*\*\*\*

(some output omitted)

MIGRAD MINIMIZATION HAS CONVERGED.  
MIGRAD WILL VERIFY CONVERGENCE AND ERROR MATRIX.  
COVARIANCE MATRIX CALCULATED SUCCESSFULLY

FCN=257.304 FROM MIGRAD STATUS=CONVERGED 31 CALLS 32 TOTAL  
EDM=2.36773e-06 STRATEGY= 1 ERROR MATRIX ACCURATE

EXT PARAMETER

NO. NAME

VALUE

ERROR

STEP

FIRST

SIZE

DERIVATIVE

1 mean

8.84225e-02

3.23862e-01

3.58344e-04

-2.24755e-02

2 sigma

3.20763e+00

2.39540e-01

2.78628e-04

-5.34724e-02

ERR DEF= 0.5

EXTERNAL ERROR MATRIX. NDIM= 25 NPAR 2 ERR DEF=0.5

1.049e-01 3.338e-04

3.338e-04 5.739e-02

PARAMETER CORRELATION COEFFICIENTS

NO. GLOBAL

1

2

1 0.00430 1.000 0.004

2 0.00430 0.004 1.000

Parameter values and approximate  
errors reported by MINUIT

Error definition (in this case 0.5 for  
a likelihood fit)

# Logging output: Minuit function MIGRAD

- Purpose: find minimum

\*\*\*\*\*

\*\* 13 \*\*MIGR

\*\*\*\*\*

(some output o

MIGRAD MINIMIZ

MIGRAD WILL VERI

COVARIANCE MATRIX CALCULATED SUCCESSFULLY

FCN=257.304

FROM MIGRAD

STATUS=CONVERGED

31 CALLS

32 TOTAL

EDM=2.36773e-06

STRATEGY= 1

ERROR MATRIX ACCURATE

EXT PARAMETER

STEP

FIRST

NO. NAME

VALUE

ERROR

SIZE

DERIVATIVE

1 mean

8.84225e-02

3.23862e-01

3.58344e-04

-2.24755e-02

2 sigma

3.20763e+00

2.39540e-01

2.78628e-04

-5.34724e-02

ERR DEF= 0.5

EXTERNAL ERROR MATRIX.

NDIM= 25

NPAR= 2

ERR DEF=0.5

1.049e-01 3.338e-04

3.338e-04 5.739e-02

PARAMETER CORRELATION COEFFICIENTS

NO. GLOBAL

1

2

1 0.00430 1.000 0.004

2 0.00430 0.004 1.000

Value of  $\chi^2$  or likelihood at minimum

(NB:  $\chi^2$  values are not divided by  $N_{\text{d.o.f}}$ )

Approximate  
Error matrix  
And covariance matrix

# Logging output: Minuit function MIGRAD

- Purpose: find minimum

*Status:*

Should be 'converged' but can be 'failed'

*Estimated Distance to Minimum*  
should be small  $O(10^{-6})$

*Error Matrix Quality*

should be 'accurate', but can be  
'approximate' in case of trouble

\*\*\*\*\*

\*\* 13 \*\*MIGRAD 1000

\*\*\*\*\*

(some output omitted)

MIGRAD MINIMIZATION HAS CONVERGED

MIGRAD WILL VERIFY CONVERGENCE AND ERROR MATRIX.

COVARIANCE MATRIX CALCULATED SUCCESSFULLY

FCN=257.304 FROM MIGRAD STATUS=CONVERGED 31 CALLS 32 TOTAL  
EDM=2.36773e-06 STRATEGY= 1 ERROR MATRIX ACCURATE

EXT PARAMETER

NO.	NAME	VALUE	ERROR	STEP SIZE	FIRST DERIVATIVE
1	mean	8.84225e-02	3.23862e-01	3.58344e-04	-2.24755e-02
2	sigma	3.20763e+00	2.39540e-01	2.78628e-04	-5.34724e-02

ERR DEF= 0.5

EXTERNAL ERROR MATRIX. NDIM= 25 NPAR= 2 ERR DEF=0.5

1.049e-01 3.338e-04

3.338e-04 5.739e-02

PARAMETER CORRELATION COEFFICIENTS

NO.	GLOBAL	1	2
1	0.00430	1.000	0.004
2	0.00430	0.004	1.000

# Logging output: Minuit function MINOS

- MINOS errors are calculated by 'hill climbing algorithm'.
  - In one dimension find points where  $\Delta\chi^2 = +1$ .
  - In  $>1$  dimension find contour with  $\Delta\chi^2 = +1$ . Errors are defined by bounding box of contour.
  - In  $>>1$  dimension can be very time consuming, but more in general more robust.

```
*****
**    23 **MINOS          1000
*****
FCN=257.304 FROM MINOS      STATUS=SUCCESSFUL      52 CALLS          94 TOTAL
                        EDM=2.36534e-06    STRATEGY= 1      ERROR MATRIX ACCURATE

EXT  PARAMETER
NO.   NAME      VALUE      PARABOLIC
                        ERROR
  1  mean      8.84225e-02  3.23861e-01
  2  sigma     3.20763e+00  2.39539e-01
                        ERR DEF= 0.5
```

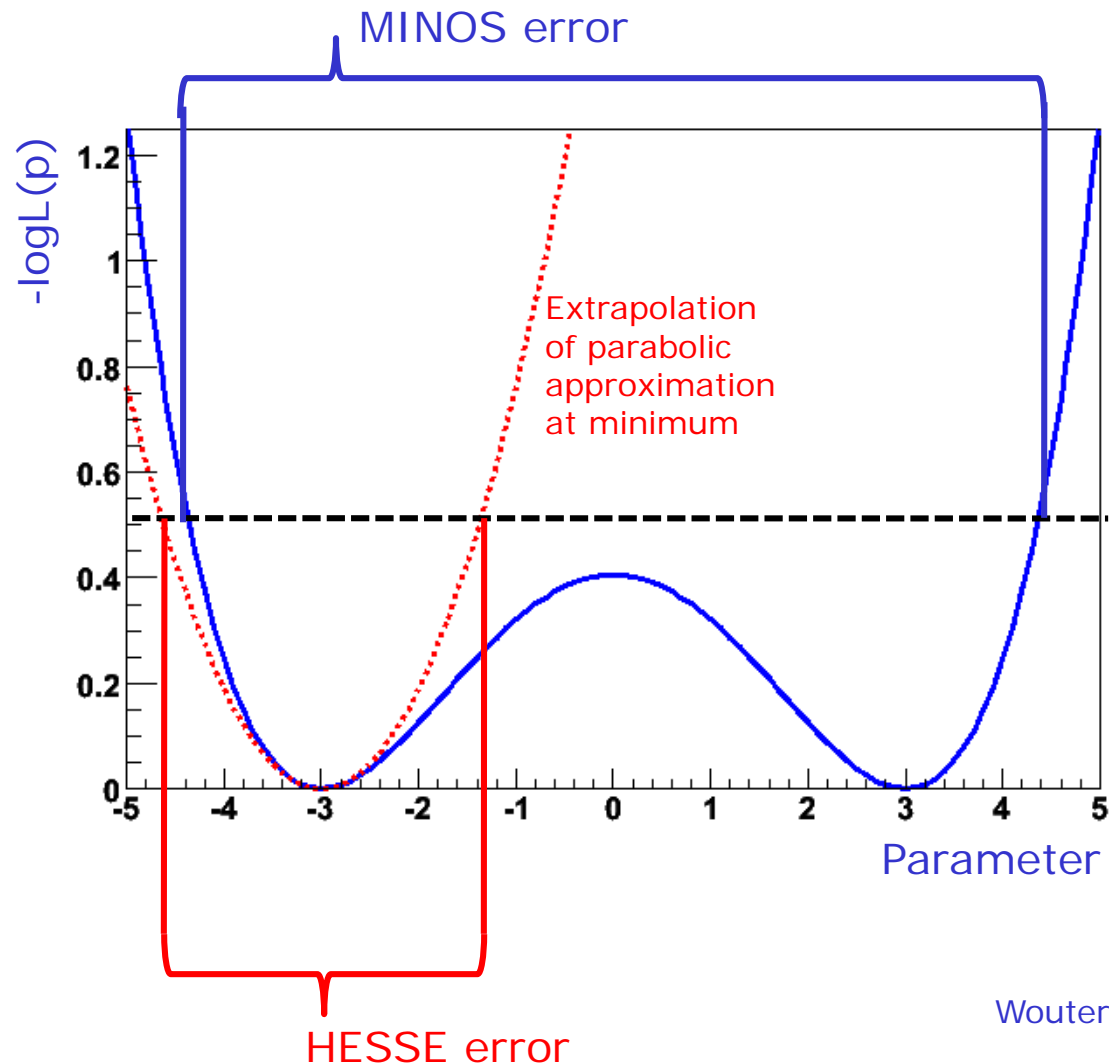
MINOS ERRORS	
NEGATIVE	POSITIVE
-3.24688e-01	3.25391e-01
-2.23321e-01	2.58893e-01

Symmetric error  
(repeated result  
from HESSE)

MINOS error  
Can be asymmetric  
(in this example the 'sigma' error  
is slightly asymmetric)

## Illustration of difference between HESSE and MINOS errors

- 'Pathological' example likelihood with multiple minima and non-parabolic behavior. (Beware!)





# Practical estimation – Fit converge problems

- Sometimes fits don't converge because, e.g.
  - MIGRAD unable to find minimum
  - HESSE finds negative second derivatives (which would imply negative errors)
- Reason is usually numerical precision and stability problems, but
  - The **underlying cause** of fit stability problems is usually by **highly correlated parameters** in fit
- HESSE correlation matrix in primary investigative tool

PARAMETER	CORRELATION COEFFICIENTS		
NO.	GLOBAL	1	2
1	0.99835	1.000	0.998
2	0.99835	0.998	1.000

*Signs of trouble...*

- In limit of 100% correlation, the usual **point solution** becomes a **line solution** (or surface solution) in parameter space.  
*Minimization problem is no longer well defined*

# Mitigating fit stability problems – Polynomials

- **Warning:** Regular parameterization of polynomials  $a_0 + a_1x + a_2x^2 + a_3x^3$  nearly always results in strong correlations between the coefficients  $a_i$ .
  - *Fit stability problems, inability to find right solution common at higher orders*
- **Solution:** Use existing parameterizations of polynomials that have (mostly) uncorrelated variables
  - **Example: Chebychev polynomials**

$$T_0(x) = 1$$

$$T_1(x) = x$$

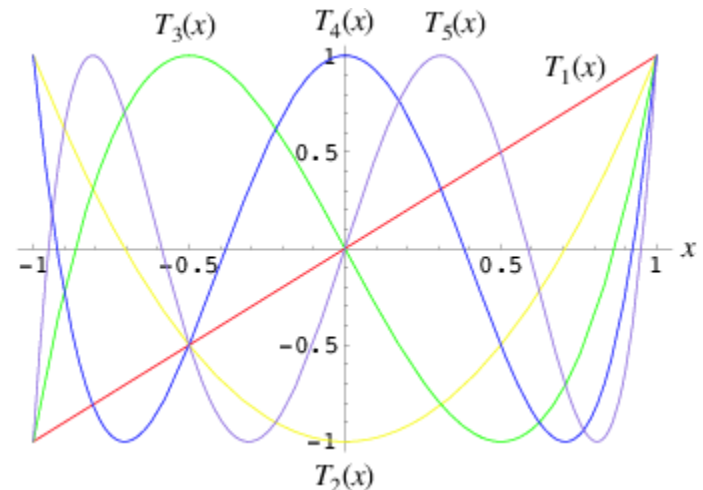
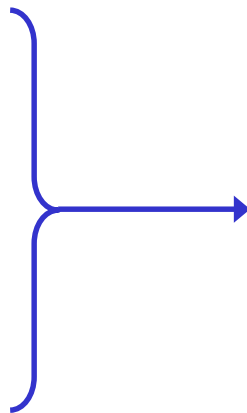
$$T_2(x) = 2x^2 - 1$$

$$T_3(x) = 4x^3 - 3x$$

$$T_4(x) = 8x^4 - 8x^2 + 1$$

$$T_5(x) = 16x^5 - 20x^3 + 5x$$

$$T_6(x) = 32x^6 - 48x^4 + 18x^2 - 1.$$



## The lesson

- You can learn an awful lot of useful information by taking a close look at the output from your fit / minimizer.
- I urge you all to do this for your practicum exercises.
- Don't turn off your logging output when doing a fit.

**Next lecture**

# Roadmap for this course

## 1. Statistics basics

- Probability theory
- Probability distributions

## 2. Parameter estimation

- Error propagation
- Simulation
- Model fitting (bulk)

## **1. Pitfalls in (big) data analysis**

- Spurious correlations in Big Data
- Data quality assessment

Coming Monday

## 1. Hypothesis Testing

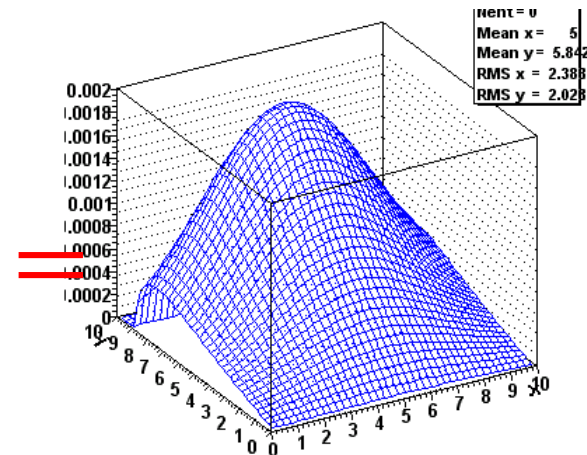
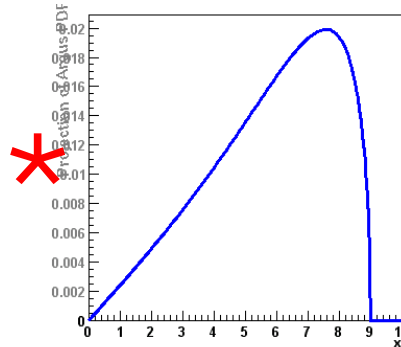
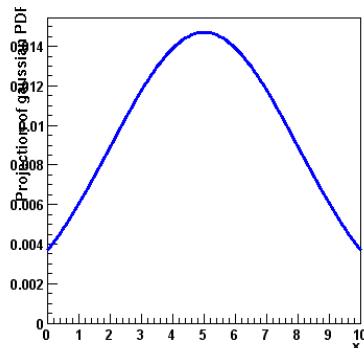
**More advanced fit models**

# Extending models to more than one dimension

- If you have data with many observables, there are two common approaches
  - Compactify information with test statistic (see previous section)
  - Describe full N-dimensional distribution with a p.d.f.
- Choice of approach largely correlated with understanding of correlation between observables and amount of information contained in correlations
  - No correlation between observables → 'Big fit' and 'Compactification' work equally well.
  - Important correlations that are poorly understood → Compactification preferred. Approach:
    1. Compactify all-but-one observable (ideally uncorrelated with the compactified observables)
    2. Cut on compactification test statistic to reduce backgrounds
    3. Fit remaining observable → Estimate from data remaining amount of background (smallest systematic uncertainty due to poor understanding of test statistic and its inputs)
  - Important correlations that are well understood → Big fit preferred

# Extending models to more than one dimension

- Bottom line: N-dim models used when either *no correlations* or *well understood correlations*
- Constructing multi-dimensional models without correlations is easy
  - Just multiply N 1-dimensional p.d.f.s.

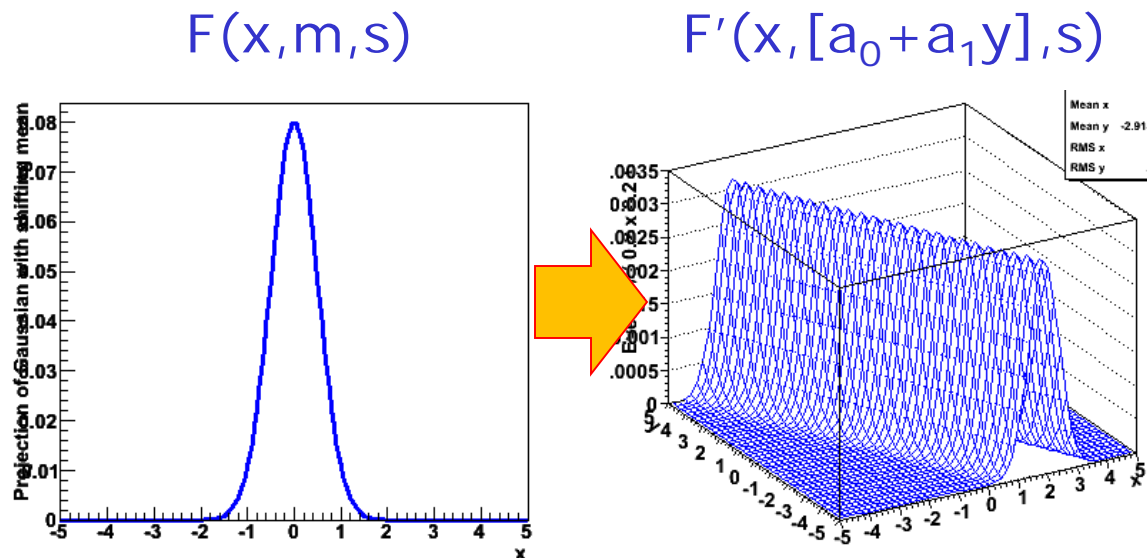


- No complex issues with p.d.f. normalization: if 1-dim p.d.f.s are normalized then product is also by construction



# Writing multi-dimensional models with correlations

- Formulating N-dim models *with* correlations may seem daunting, but it really isn't so difficult.
  - Simplest approach: start with one-dimensional model, replace one parameter  $p$  with a function  $p'(y)$  of another observable
  - Yields correction distribution of  $x$  for every given value of  $y$



- NB: Distribution of  $y$  probably *not* correct...

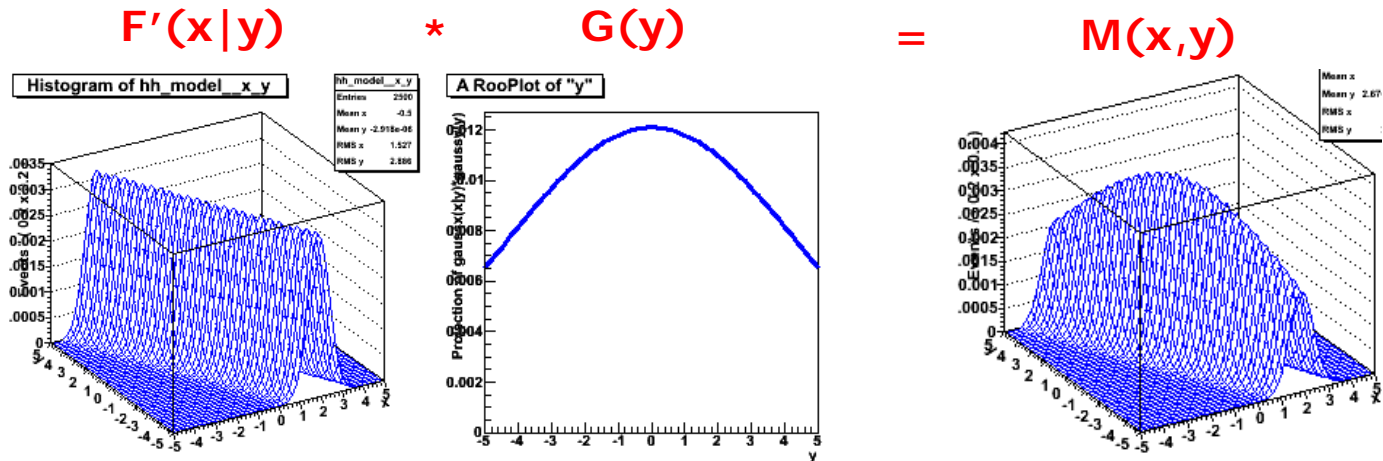
# Writing multi-dimensional models with correlations

- Solution: see  $F'(x,y,p)$  as a **conditional p.d.f.**  $F'(x|y)$ 
  - Difference is in normalization

$$\int F(x, y) dx dy \equiv 1 \quad \int F(x | y) dx \equiv 1 \quad \text{for each value of } y$$

- Then multiply with a separate p.d.f describing distribution in  $y$

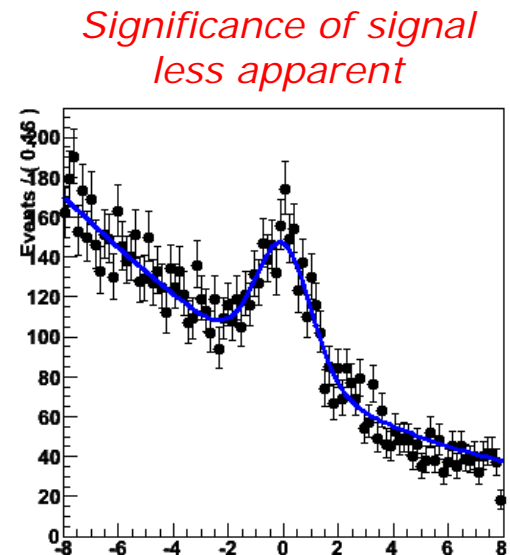
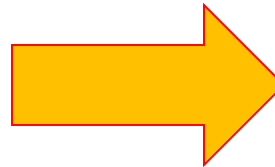
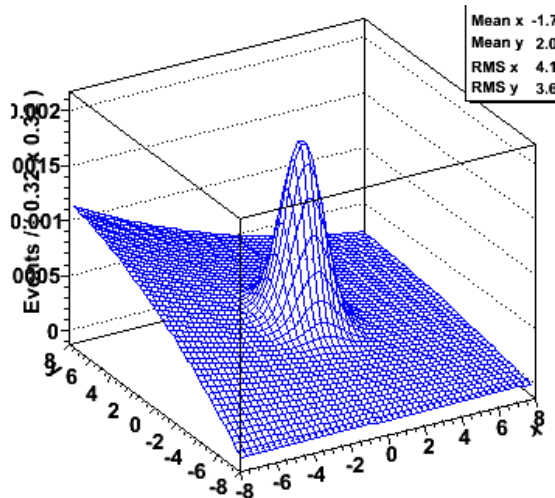
$$M(x, y) = F'(x | y) \cdot G(y)$$



- Almost **all** modeling issues with correlations can be treated this way

# Visualization of multi-dimensional models

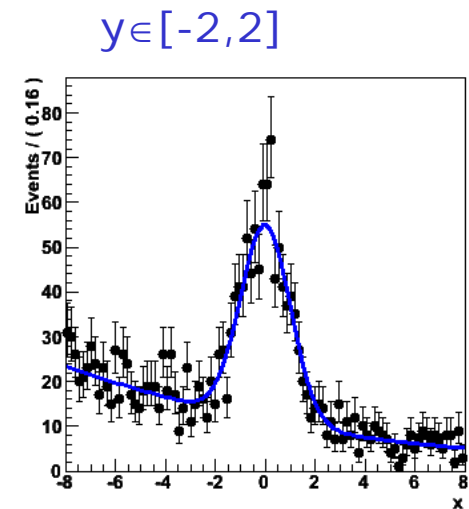
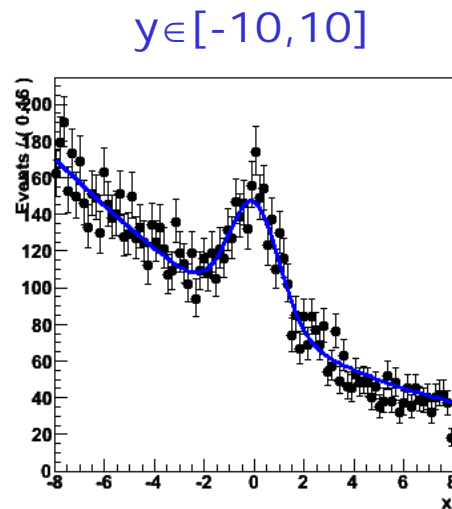
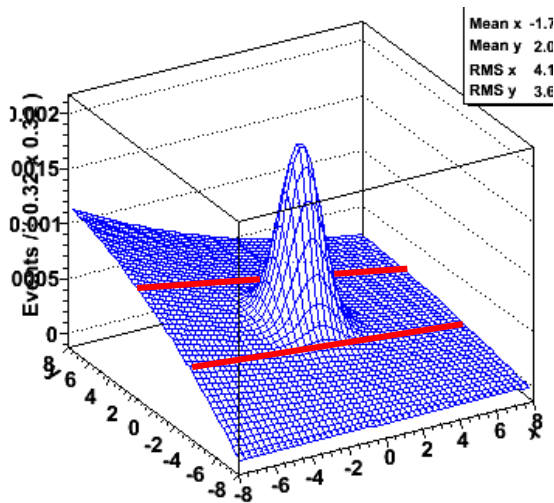
- Visualization of multi-dimensional models presents some additional challenges w.r.t. 1-D
- Can show 2D,3D distribution
  - Graphically appealing, but not so useful as you cannot overlay model on data and judge goodness-of-fit
  - Prefer to project on one dimension (there will be multiple choices)
  - But plain projection discards a lot of information contained in both model and data



*Reason: Discriminating information in y observable in both data and model is ignored*

# Visualizing signal projections of N-dim models

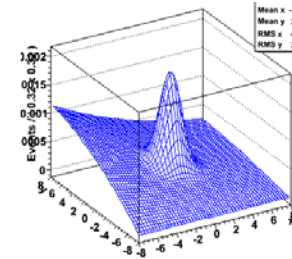
- Simplest solution, only show model and data in “**signal range**” of observable  $y$ 
  - Significance shown in “range projection” much more in line with that of 2D distribution



- Easy to define a “signal range” simple model above.  
How about 6-dimensional model with non-trivial shape?
  - Need **generic algorithm**  $\rightarrow$  Likelihood ratio plot (See Backup)

# Likelihood ratio plots

- Idea: use information on  $S/(S+B)$  ratio in projected observables to define a cut
- Example: generalize previous toy model to 3 dimensions
- Express information on  $S/(S+B)$  ratio of model in terms of integrals over model components

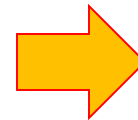


$$LR(x, y, z) = \frac{S(x, y, z)}{[S(x, y, z) + B(x, y, z)]}$$

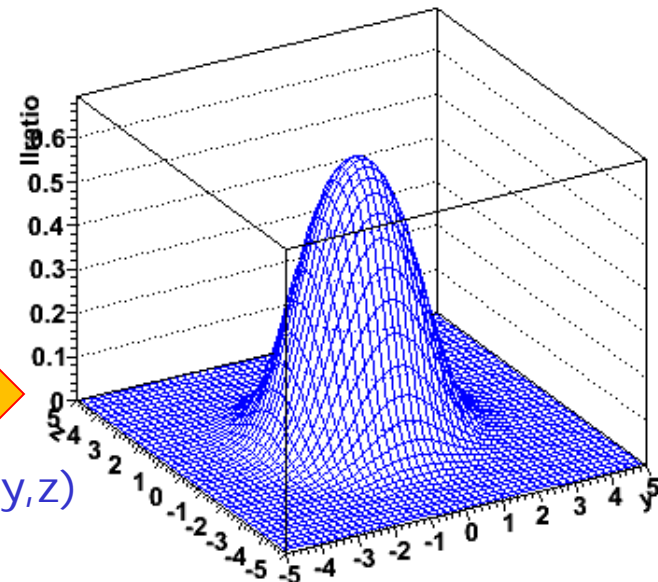


Integrate over  $x$

$$LR(y, z) = \frac{\int S(x, y, z) dx}{\int [S(x, y, z) + B(x, y, z)] dx}$$

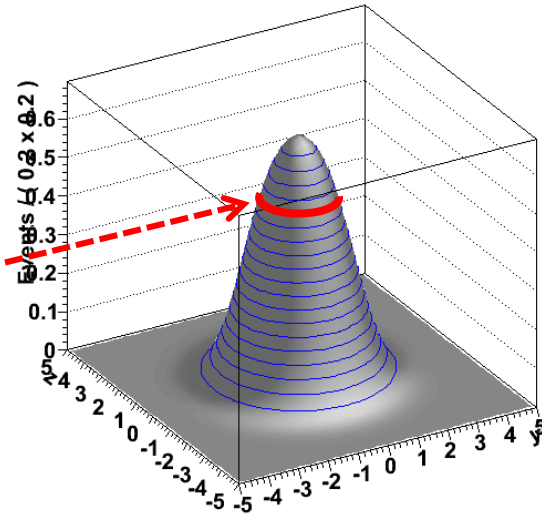


Plot LR vs  $(y, z)$



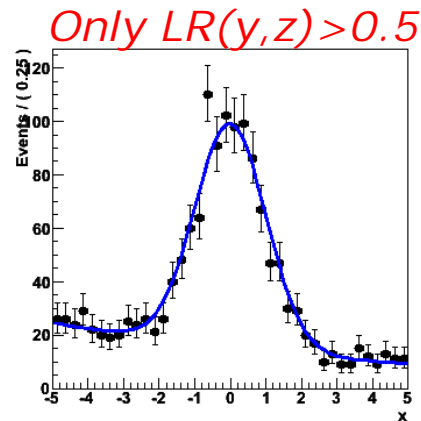
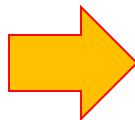
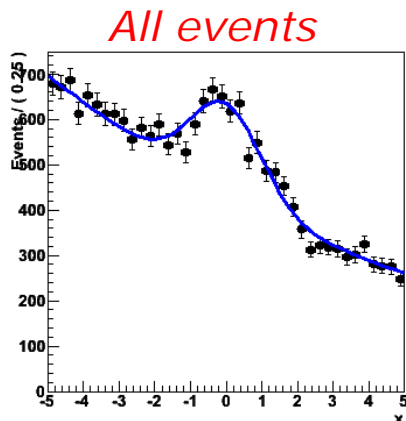
# Likelihood ratio plots

- Decide on  $s/(s+b)$  purity contour of  $LR(y,z)$ 
  - Example  $s/(s+b) > 50\%$
- Plot both data and model with corresponding cut.
  - For data: calculate  $LR(y,z)$  for each event, plot only event with  $LR > 0.5$
  - For model: using Monte Carlo integration technique:



$$\int_{LR(y,z) > 0.5} M(x, y, z) dy dz \approx \frac{1}{N} \sum_{D(y,z)} M(x, y_i, z_i)$$

Dataset with values of  $(y,z)$  sampled from p.d.f and filtered for events that meet  $LR(y,z) > 0.5$

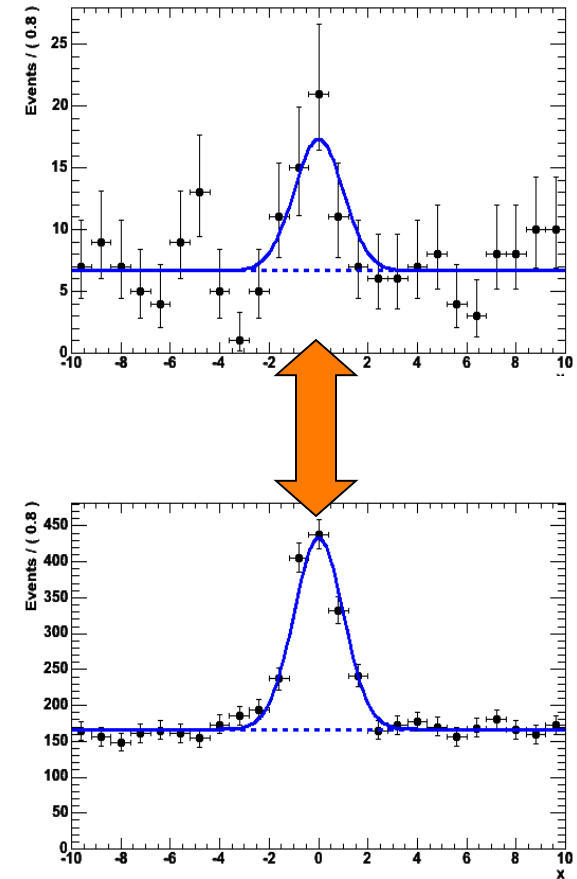


## Multidimensional fits – Goodness-of-fit determination

- Goodness-of-fit determination of  $>1$  D models is difficult
  - Standard  $\chi^2$  test does not work very well in N-dim because of natural occurrence of large number of empty bins
  - Simple equivalent of (unbinned) Kolmogorov test in  $>1$ -D does not exist
- This area is still very much a work in progress
  - Several new ideas proposed but sometimes difficult to calculate, or not universally suitable
  - Some examples
    - Cramer-von Mises (close to Kolmogorov in concept)
    - Anderson-Darling
    - ‘Energy’ tests
  - **No magic bullet here, “best” generally an ill-posed question**
  - Some references to recent progress:
    - PHYSTAT2001/2003/2005

## Practical fitting – Error propagation between samples

- Common situation: you want to fit a small signal in a large sample
  - Problem: small statistics does not constrain shape of your signal very well
  - Result: errors are large
- Idea: Constrain shape of your signal from a fit to a control sample
  - Larger/cleaner data or MC sample with similar properties
- Needed: a way to propagate the information from the control sample fit (parameter values *and* errors) to your signal fit

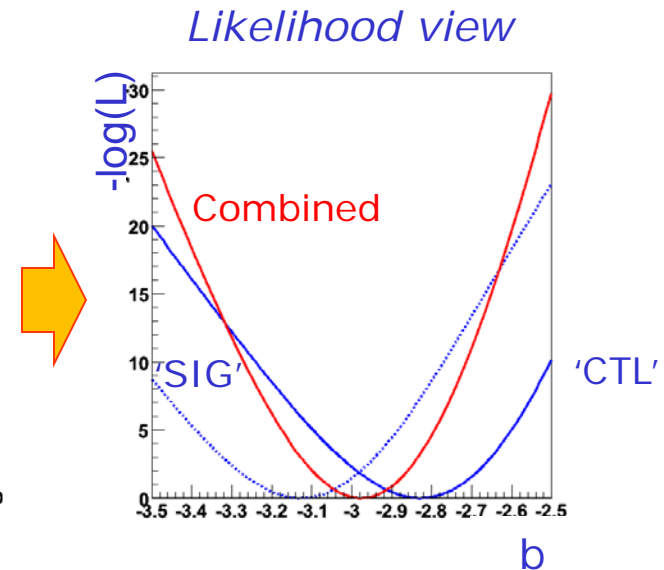
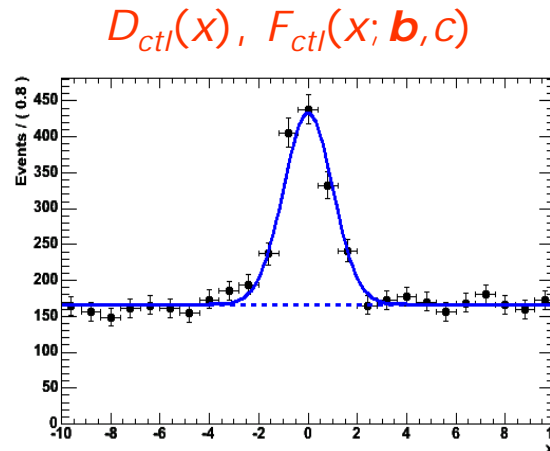
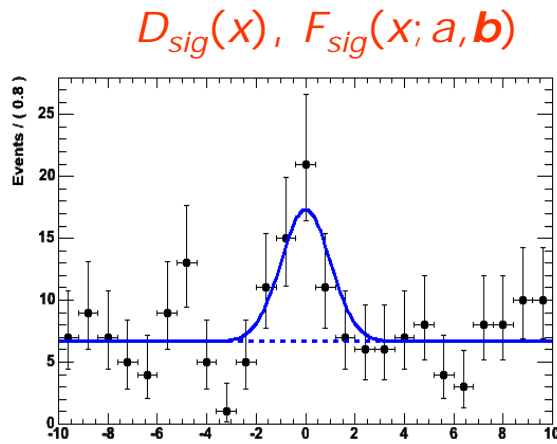




# Practical fitting – Simultaneous fit technique

- given data  $D_{sig}(x)$  and model  $F_{sig}(x; a, \mathbf{b})$  and data  $D_{ctl}(x)$  and model  $F_{ctl}(x; \mathbf{b}, c)$

– Construct  $-\log[L_{sig}(a, \mathbf{b})]$  and  $-\log[L_{ctl}(\mathbf{b}, c)]$  and

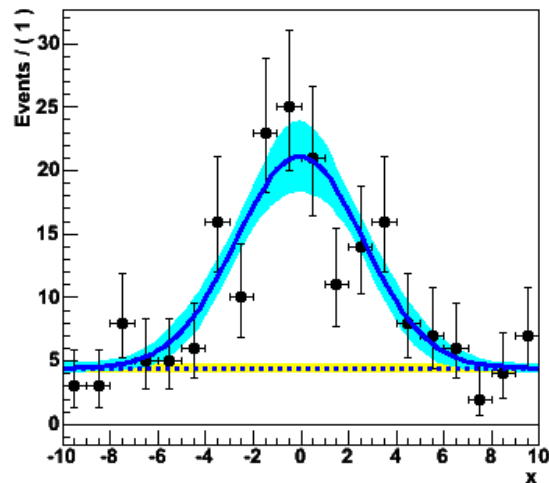


- Minimize  $-\log L(a, \mathbf{b}, c) = -\log L(a, \mathbf{b}) + -\log L(\mathbf{b}, c)$ 
  - Errors, correlations on common param.  $\mathbf{b}$  automatically propagated

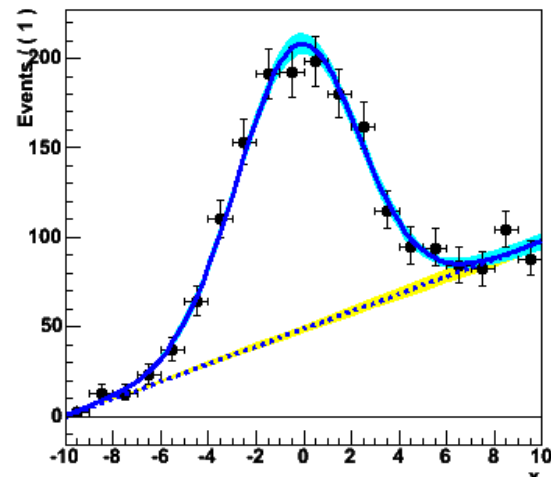
# Practical fitting – Simultaneous fit technique

- Simultaneous fit with visualization of error

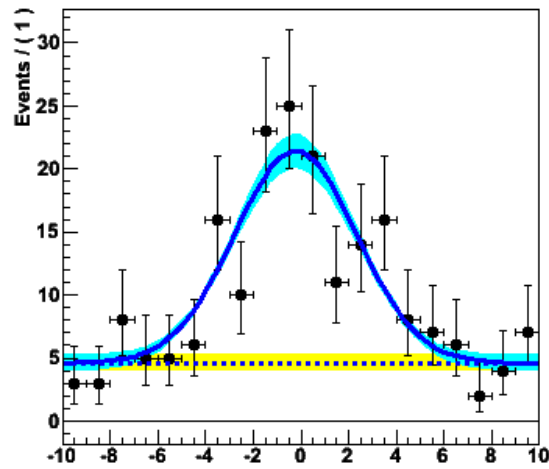
Fit to SIGNAL sample



Fit to CONTROL sample

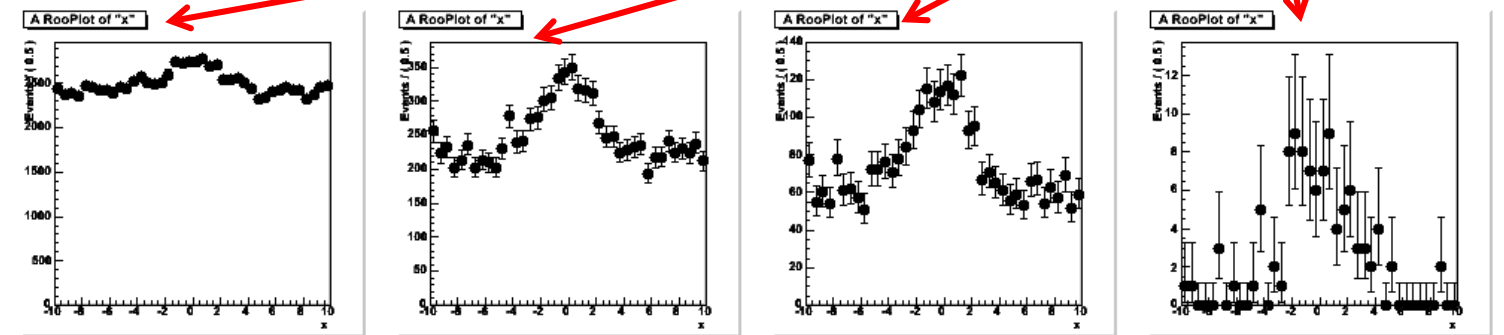
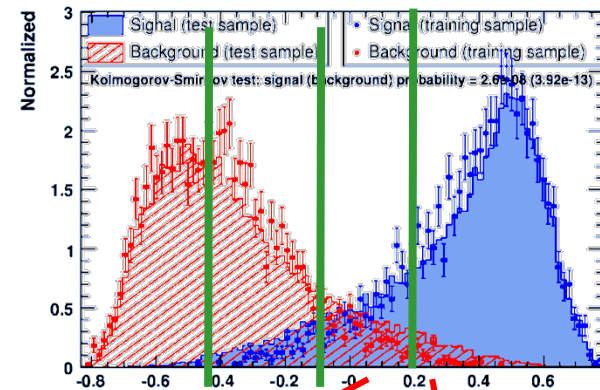


Joint fit to SIGNAL and CONTROL samples



# Another application of simultaneous fits

- You can also use simultaneous fits to samples of the same type ("signal samples") with different purity
- Go back to example of NN with one observable left out
  - Fit  $x$  after cut on  $N(x)$
  - But instead of just fitting data with  $N(x) > \alpha$ , slice data in bins of  $N(x)$  and fit *each bin*.
  - Now you exploit all data instead of just most pure data. Still no uncontrolled systematic uncertainty as purity is measured from data in each slice
  - Combine information of all slices in simultaneous fit



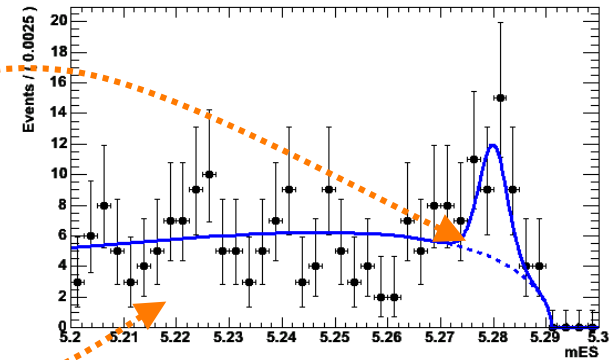
## Practical Estimation – Verifying the validity of your fit

- How to validate your fit? – You want to demonstrate that
  - 1) Your fit procedure gives on average the correct answer **'no bias'**
  - 2) The uncertainty quoted by your fit is an accurate measure for the statistical spread in your measurement **'correct error'**
- **Validation is important for low statistics fits**
  - **Correct behavior not obvious a priori due to intrinsic ML bias proportional to  $1/N$**
- Basic validation strategy – **A simulation study**
  - 1) Obtain a large sample of simulated events
  - 2) Divide your simulated events in  $O(100-1000)$  samples with the same size as the problem under study
  - 3) Repeat fit procedure for each data-sized simulated sample
  - 4) Compare average value of fitted parameter values with generated value → **Demonstrates (absence of) bias**
  - 5) Compare spread in fitted parameters values with quoted parameter error → **Demonstrates (in)correctness of error**

# Fit Validation Study – Practical example

- Example fit model in 1-D (B mass)

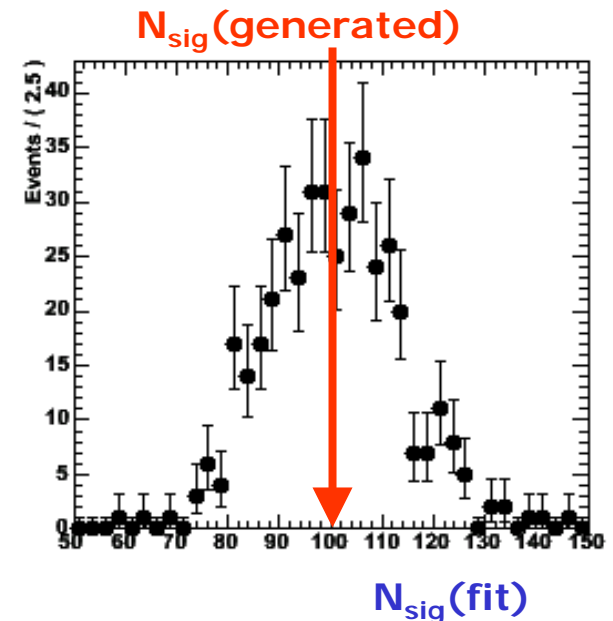
- Signal component is Gaussian centered at B mass
- Background component is Argus function (models phase space near kinematic limit)



$$F(m; N_{\text{sig}}, N_{\text{bkg}}, \vec{p}_S, \vec{p}_B) = N_{\text{sig}} \cdot G(m; p_S) + N_{\text{bkg}} \cdot A(m; p_B)$$

- Fit parameter under study:  $N_{\text{sig}}$

- Results of simulation study:  
1000 experiments  
with  $N_{\text{SIG}}(\text{gen}) = 100$ ,  $N_{\text{BKG}}(\text{gen}) = 200$
- Distribution of  $N_{\text{sig}}(\text{fit})$  .....→
- This particular fit looks unbiased...



# Fit Validation Study – The pull distribution

- What about the validity of the error?

- Distribution of error from simulated experiments is difficult to interpret...
- We don't have equivalent of  $N_{\text{sig}}(\text{generated})$  for the error

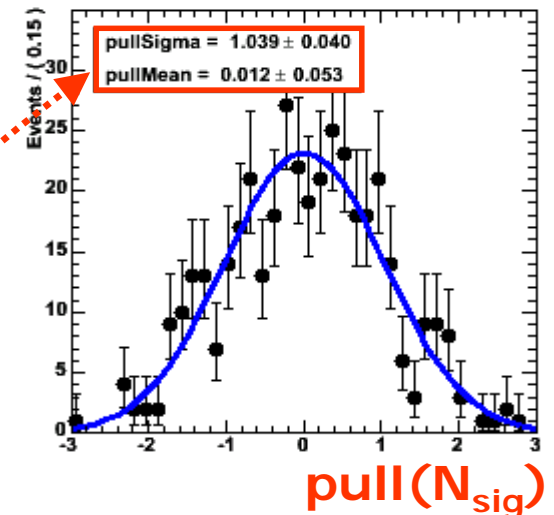
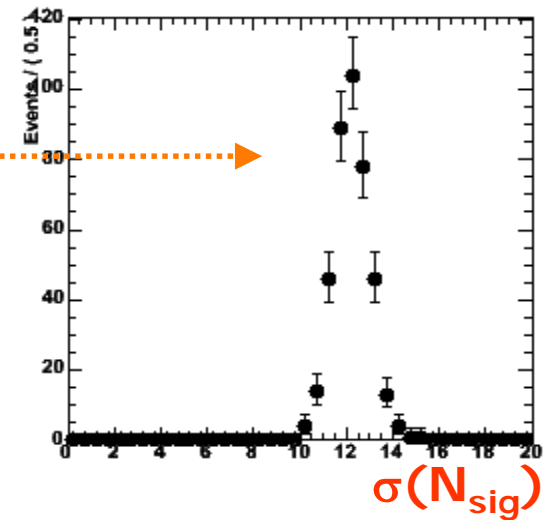
- Solution: look at the *pull distribution*

- Definition: 
$$\text{pull}(N_{\text{sig}}) = \frac{N_{\text{sig}}^{\text{fit}} - N_{\text{sig}}^{\text{true}}}{\sigma_N^{\text{fit}}}$$

- Properties of pull:

- Mean is 0 if there is no bias
- Width is 1 if error is correct

- In this example: no bias, correct error within statistical precision of study

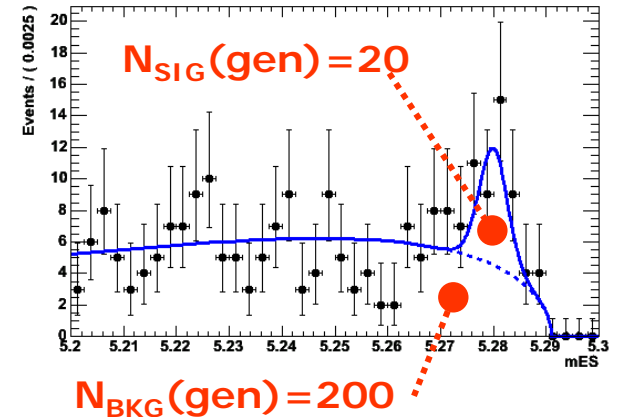


# Fit Validation Study – Low statistics example

- Special care should be taken when fitting small data samples
  - Also if fitting for small signal component in large sample
- Possible causes of trouble
  - $\chi^2$  estimators may become approximate as Gaussian approximation of Poisson statistics becomes inaccurate
  - ML estimators may no longer be efficient
    - error estimate from 2<sup>nd</sup> derivative may become inaccurate
  - Bias term proportional to  $1/N$  of ML and  $\chi^2$  estimators may no longer be small compared to  $1/\sqrt{N}$
- In general, absence of bias, correctness of error can not be assumed. How to proceed?
  - Use unbinned ML fits only – most robust at low statistics
  - Explicitly verify the validity of your fit

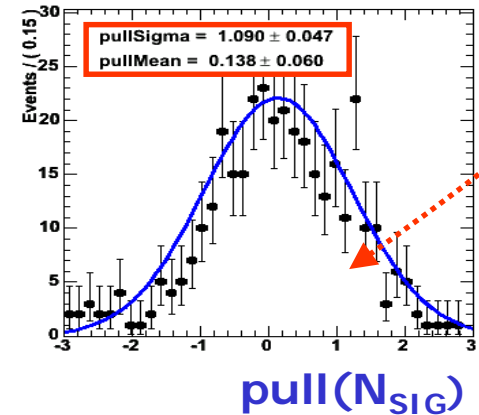
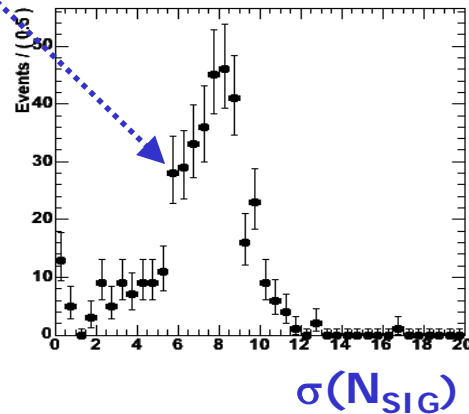
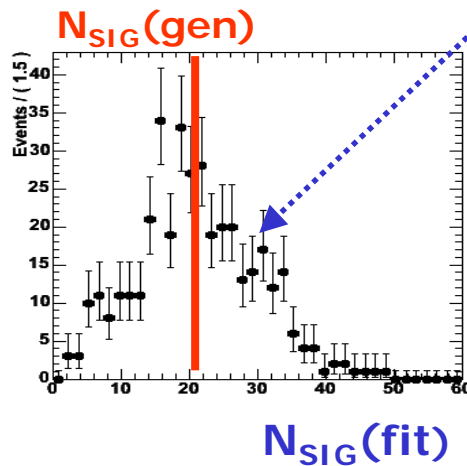
# Demonstration of fit bias at low N – pull distributions

- Low statistics example:
  - Scenario as before but now with 200 bkg events and **only 20 signal events** (instead of 100)
- Results of simulation study



Distributions become asymmetric at low statistics

→ Fit is positively biased!



- *Absence of bias, correct error at low statistics not obvious!*
  - *Small yields are typically overestimated*



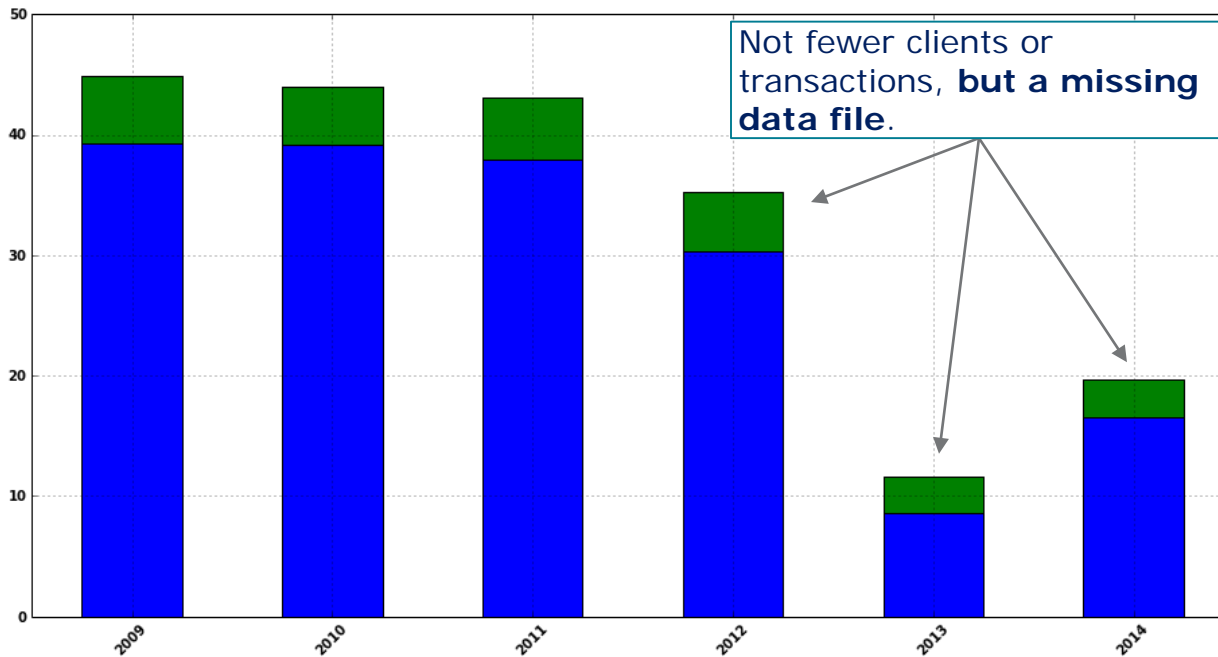
**Backup**

# Data Errors

Types of data 'errors':

- Missing files or records
- Missing or incorrect fields

Follow-up number	Date	Amount	Transaction type
20150004	Jan 13 2015	-100.00 €	Pin withdrawal
20150005	Jan 15 2015	-50.00 €	Pin payment
20150006	Jan 20 2015	2500.00 €	
20150007	Jan 22 2015	-40.00 €	Pin payment
20150008	Jan 29 2015	-1500.00 €	failed
20150008	Jan 29 2015	-1500.00 €	Bank transfer
20150009	Feb 3 2015	-250.00	Pin payment
20150010	Jan 4 2015	-150.00	Pin payment



# Data Errors

Types of data 'errors':

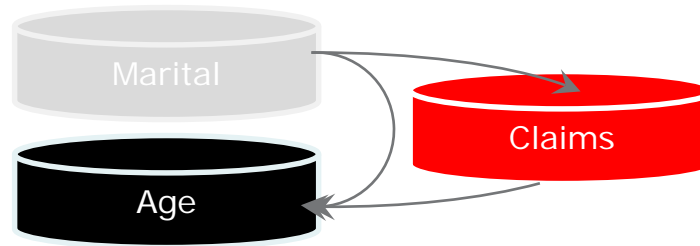
- Missing or ill understood relations between fields
- Missing or ill understood relations between records

Follow-up number	Date	Amount	Transaction type
20150004	Jan 13 2015	-100.00 €	Pin withdrawal
20150005	Jan 15 2015	-50.00 €	Pin payment
20150006	Jan 20 2015	2500.00 €	
20150007	Jan 22 2015	-40.00 €	Pin payment
20150008	Jan 29 2015	-1500.00 €	failed
20150008	Jan 29 2015	-1500.00 €	Bank transfer
20150009	Feb 3 2015	-250.00	Pin payment
20150010	Jan 4 2015	-150.00	Pin payment

# Data Errors

Types of data 'errors':

- Biased data



Enriching Age data with Marital status:

1. If name given in Age data, then get Marital status directly with name.
2. Else get name from Claims data through address, and then get Marital status with name.
3. Else marital status remains unknown.

Now what is the probability that a married person makes a claim?

**But people with claims tend to have better known marital status than others!**

# Chi-square Uncertainties

Chi square fit

- Minimize  $\chi^2 = \sum \frac{(x_i - f_i(\epsilon))^2}{\sigma_i^2}$  with respect to model parameters  $\epsilon$ .
- Here  $x_i$  are the data points,  $\sigma_i$  the uncertainties on the data points and  $f_i(\epsilon)$  is the model prediction for data point  $i$  depending on parameters  $\epsilon$ .
- If the data points  $x_i$  indeed come from your model, but can fluctuate from their expected values  $f_i(\epsilon)$  with Gaussian probability with standard deviations  $\sigma_i$ , then the value for  $\chi^2$  that can arise follows a chi-square distribution.
- On average (when taking points repeatedly)  $\chi^2$  takes on the value  $ndof = nr.of\ points - nr.of\ parameters$ , also known as the number of degrees of freedom.
- If your data points do not stem from your model, then the  $\chi^2$  tends to be too high.
- If you overestimated or underestimated your uncertainties  $\sigma_i$  then the  $\chi^2$  will be too small or large respectively.