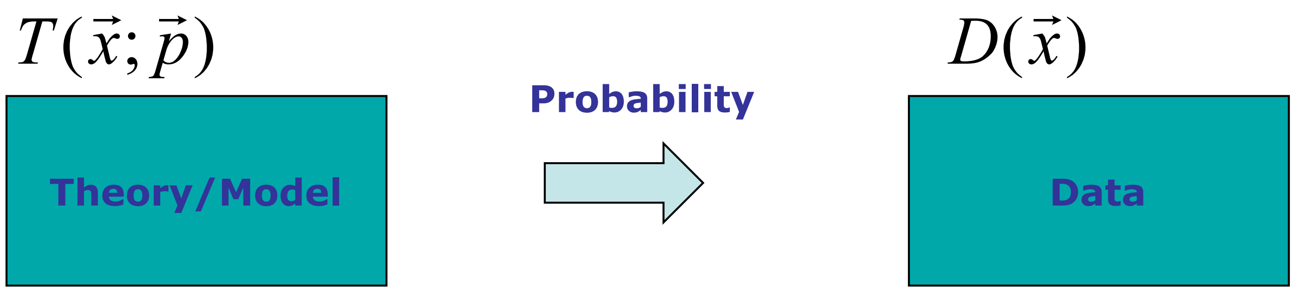
Use of likelihood fits in HEP

Gerhard Raven
NIKHEF and VU Amsterdam

Some basics of parameter estimation
Examples, Good practice, …
Several real-world examples
of increasing complexity…

Parts of this talk taken (with permission) from
http://www.slac.stanford.edu/~verkerke/bnd2004/data_analysis.pdf

Parameter Estimation

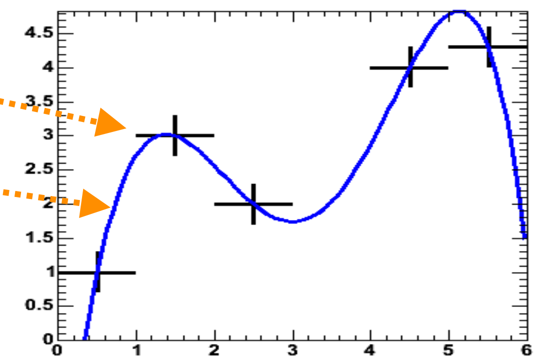


- Given the theoretical distribution parameters $\vec{p}$, what can we say about the data

- **Need a procedure to estimate $\vec{p}$ from $D(\vec{x})$**
  - Common technique – fit!

A well known estimator – the χ^2 fit

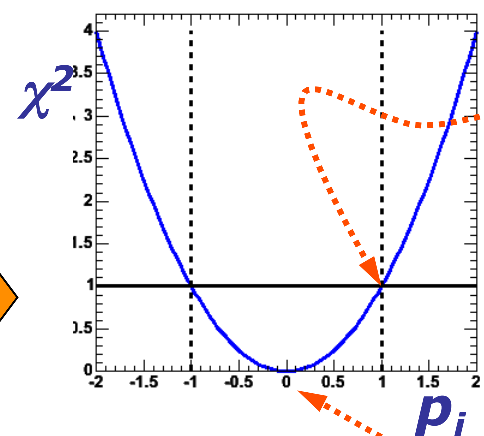
- Given a set of points $\{(\vec{x}_i, y_i, \sigma_i)\}$ and a function $f(\boldsymbol{x},\boldsymbol{p})$ define the χ^2



$$\chi^2(\vec{p}) = \sum_i \frac{(y_i - f(\vec{x};\vec{p}))^2}{\sigma_y^2}$$

- Estimate parameters by minimizing the $\chi^2(p)$ with respect to all parameters p_i
  - *In practice, look for*

$$\frac{d\chi^2(p_i)}{dp_i} = 0$$



Error on p_i is given by χ2 variation of +1

Value of p_i at minimum is estimate for p_i

- Well known: but why does it work? Is it always right? Does it always give the best possible error?

Back to Basics – What is an estimator?

- An ***estimator*** is a ***procedure*** giving a value for a parameter or a property of a distribution as a function of the actual data values, e.g.

$$\hat{\mu}(x) = \frac{1}{N}\sum_i x_i$$

← **Estimator of the mean**

$$\hat{V}(x) = \frac{1}{N}\sum_i (x_i - \vec{\mu})^2$$

← **Estimator of the variance**

- A perfect estimator is
  - Consistent: $\lim_{n\to\infty}(\hat{a}) = a$
  - Unbiased – *With finite statistics you get the right answer on average*
  - Efficient: $V(\hat{a}) = \left\langle (\hat{a} - \langle \hat{a} \rangle)^2 \right\rangle$

    ← This is called the **Minimum Variance Bound**
  - ***There are no perfect estimators!***

Another Common Estimator: Likelihood

NB: Functions used in likelihoods must be Probability Density Functions:

$$\int F(\vec{x};\vec{p})d\vec{x} \equiv 1, \quad F(\vec{x};\vec{p}) > 0$$

- Definition of Likelihood
  - given $\boldsymbol{D(\vec{x})}$ and $\boldsymbol{F(\vec{x};\vec{p})}$

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

  - For convenience the negative log of the Likelihood is often used

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

- Parameters are estimated by maximizing the Likelihood, or equivalently minimizing *–ln(L)*

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

Variance on ML parameter estimates

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

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

  - I.e. variance is estimated from 2nd derivative of *–log(L)* at minimum
  - Valid if estimator is **efficient** and **unbiased**!

From Rao-Cramer-Frechet inequality

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

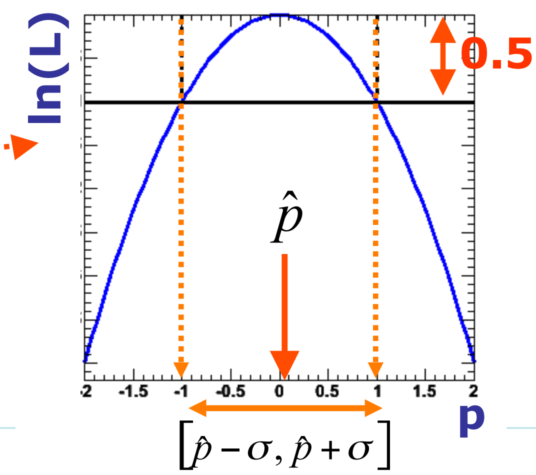
b = bias as function of p, inequality becomes equality in limit of efficient estimator

- **Visual interpretation** of variance estimate
  - Taylor expand *log(L)* around maximum

$$\ln L(p) = \ln L(\hat{p}) + \left. \frac{d \ln L}{dp} \right|_{p=\hat{p}} (p - \hat{p}) + \frac{1}{2} \left. \frac{d^2 \ln L}{d^2 p} \right|_{p=\hat{p}} (p - \hat{p})^2$$

$$= \ln L_{\max} + \left. \frac{d^2 \ln L}{d^2 p} \right|_{p=\hat{p}} \frac{(p - \hat{p})^2}{2}$$

$$= \ln L_{\max} + \frac{(p - \hat{p})^2}{2\hat{\sigma}_p^2} \Rightarrow \boxed{\ln L(p \pm \sigma) = \ln L_{\max} - \frac{1}{2}}$$



Properties of Maximum Likelihood estimators

- In general, Maximum Likelihood estimators are
  - **Consistent** (gives right answer for $N \rightarrow \infty$)
  - **Mostly unbiased** (bias $\propto 1/N$, 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 $(\hat{p})^2 = \widehat{(p^2)}$

Use of 2nd derivative of −log(L) for variance estimate is usually OK

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

More about maximum likelihood estimation

- It's not 'right' it is just sensible

- It does not give you the 'most likely value of p' –
  it gives you *the value of p for which this data is most likely*

- Numeric methods are often needed to find the maximum of *ln(L)*
  - Especially difficult if there is >1 parameter
  - Standard tool in HEP: MINUIT

- Max. Likelihood does **not** give you a goodness-of-fit measure
  - If assumed $F(\vec{x};\vec{p})$ is not capable of describing your data for any $\vec{p}$, the procedure will *not* complain
  - The absolute value of L tells you *nothing*!

Properties of χ^2 estimators

- Properties of χ^2 estimator follow from properties of ML estimator

$$F(x_i;\vec{p}) = \frac{\exp\left[-\frac{1}{2}\left(\frac{y_i - f(x_i;\vec{p})}{\sigma_i}\right)^2\right]}{\sqrt{2\pi}\sigma_i}$$

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

Take log,
Sum over all points $\boldsymbol{x_i}$

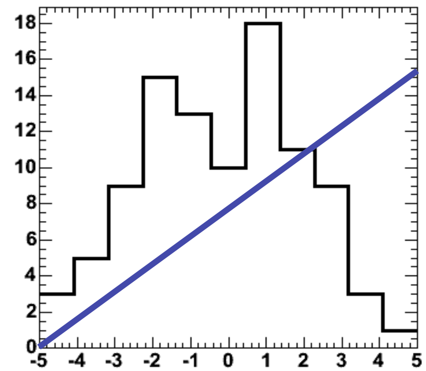
$$\ln L(\vec{p}) = -\tfrac{1}{2}\sum_i \left(\frac{y_i - f(x_i;\vec{p})}{\sigma_i}\right)^2 = -\tfrac{1}{2}\chi^2$$

The Likelihood function in p for given points $x_i(\sigma_i)$ 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 σ_i is truly Gaussian**
  - i.e. need $n_i > 10$ if y_i follows a Poisson distribution
- Bonus: Goodness-of-fit measure – $\chi^2 \approx 1$ per d.o.f

Estimating and interpreting Goodness-Of-Fit

- Fitting determines best set of parameters of *a given model* to describe data
  - **Is 'best' good enough?,** i.e.
  - Is it an adequate description, or are there significant and incompatible differences?



'Not good enough'

- Most common test: **the χ^2 test**

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

  - If f(x) describes data then $\chi^2 \approx N$, if $\chi^2 >> N$ something is wrong
  - How to quantify meaning of 'large χ^2'?

How to quantify meaning of 'large χ^2'

- Probability distr. for χ^2 is given by

$$\chi^2 = \sum_i \left(\frac{y_i - \mu_i}{\sigma_i} \right)^2 \quad \Longrightarrow \quad p(\chi^2, N) = \frac{2^{-N/2}}{\Gamma(N/2)} \chi^{N-2} e^{-\chi^2/2}$$

- To make judgement on goodness-of-fit, relevant quantity is integral of above:

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

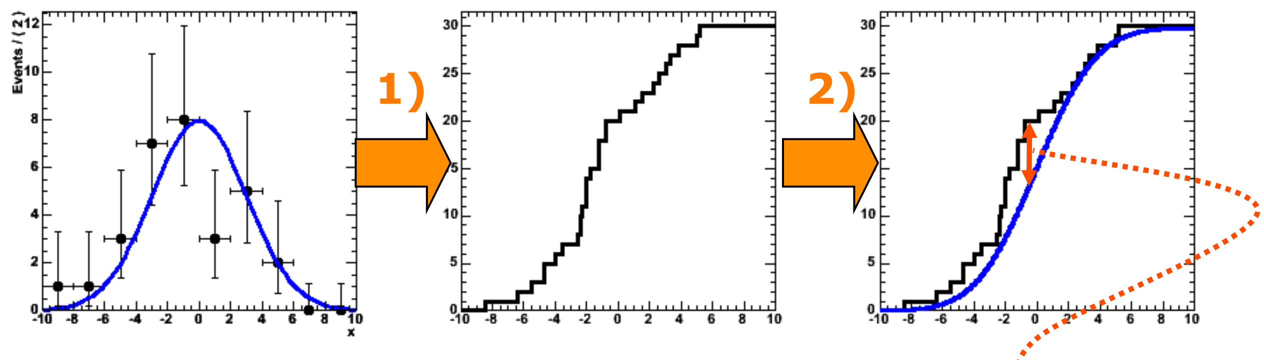
- **What does χ^2 probability $P(\chi^2,N)$ mean?**
  - It is the probability that a function which does genuinely describe the data on N points would give a χ^2 probability *as large or larger* than the one you already have.
    - Since it is a probability, it is a number in the range [0-1]

Goodness-of-fit – χ^2

- Example for χ^2 probability
  - Suppose you have a function $\boldsymbol{f(\vec{x};\vec{p})}$ which gives a χ^2 of 20 for 5 points (histogram bins).
  - Not impossible that $\boldsymbol{f(\vec{x};\vec{p})}$ 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 PAW '`call prob(chi2,ndf)`'
  - To calculate the probability in ROOT '`TMath::Prob(chi2,ndf)`'
  - For large N, sqrt($2\chi^2$) has a Gaussian distribution with mean sqrt(2N-1) and σ=1

Goodness-of-fit – Alternatives to χ^2

- When sample size is very small, it may be difficult to find sensible binning – Look for binning free test
- Kolmogorov Test
  1) Take all data values, arrange in increasing order and plot cumulative distribution
  2) Overlay cumulative probability distribution



- GOF measure:

$$d = \sqrt{N} \cdot \max\left|\mathrm{cum}(x) - \mathrm{cum}(p)\right|$$

- ‘d’ large → bad agreement; ‘d’ small – good agreement
- Practical tip: in ROOT: `TH1::KolmogorovTest(TF1&)` calculates probability for you

Maximum Likelihood or χ^2?

- χ^2 fit is fastest, easiest
  - 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
  - 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
  - Much faster than full Maximum Likihood
  - Correct Poisson treatment of low statistics bins
  - Misses information with feature size < bin size
  - Has bias proportional to 1/N

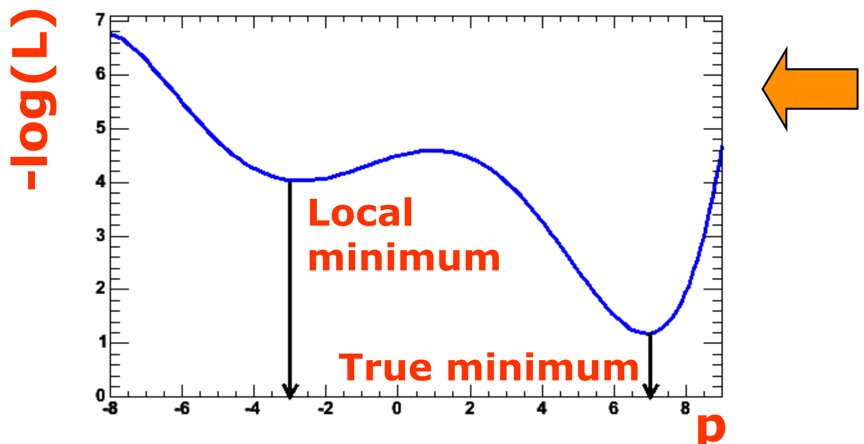
$$-\ln L(p)_{\text{binned}} = \sum_{\text{bins}} n_{\text{bin}} \ln F(\vec{x}_{\text{bin-center}}; \vec{p})$$

Practical estimation – Numeric χ^2 and -log(L) minimization

- For most data analysis problems minimization of χ^2 or –log(L) cannot be performed analytically
  - Need to rely on numeric/computational methods
  - In >1 dimension generally a difficult problem!
- But no need to worry – Software exists to solve this problem for you:
  - Function minimization workhorse in HEP many years: MINUIT
  - MINUIT does function minimization and error analysis
  - It is used in the PAW,ROOT fitting interfaces behind the scenes
  - It produces a lot of useful information, that is sometimes overlooked
  - Will look in a bit more detail into MINUIT output and functionality next

Numeric χ^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
  - This may come as a disappointment to some…
  - So you need to supply good starting values for your parameters



Reason: There may exist multiple (local) minima in the likelihood or χ^2

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

Multi-dimensional fits – Benefit analysis

- Fits to multi-dimensional data sets offer opportunities but also introduce several headaches

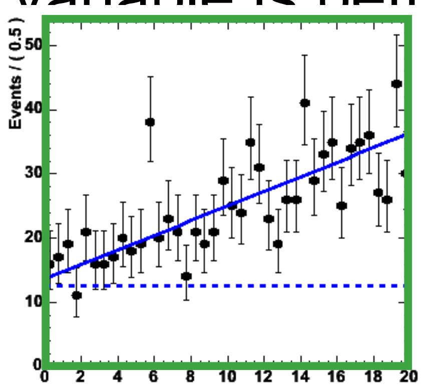
Pro

- Enhanced in sensitivity because more data and information is used simultaneously
- Exploit information in correlations between observables

Con

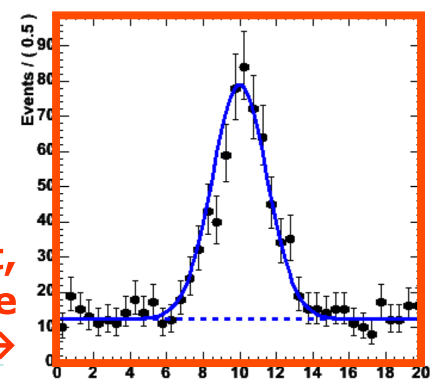
- More difficult to visualize model, model-data agreement
- More room for hard-to-find problems
- Just a lot more work

- It depends very much on your particular analysis if fitting a variable is better than cutting on it



← No obvious cut, may be worthwile to include in n-D fit

Obvious where to cut, probably not worthwile to include in n-D fit →



Ways to construct a multi-D fit model

- Simplest way: take product of N 1-dim models, e.g

$$FG(x,y) = F(x) \cdot G(y)$$

  - Assumes x and y are uncorrelated in data. If this assumption is unwarranted you may get a wrong result: Think & Check!

- Harder way: explicitly model correlations by writing a 2-D model, eg.:

$$H(x,y) = \exp\left[-\left((x+y)/2\right)^2\right]$$

- Hybrid approach:
  - Use conditional probabilities

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

Probability for y $\int G(y)dy \equiv 1$

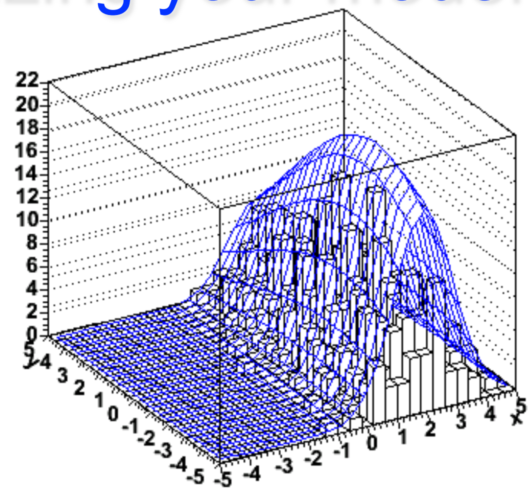
Probability for x, given a value of y

$\int F(x,y)dx \equiv 1$ for all values of y

Multi-dimensional fits – visualizing your model

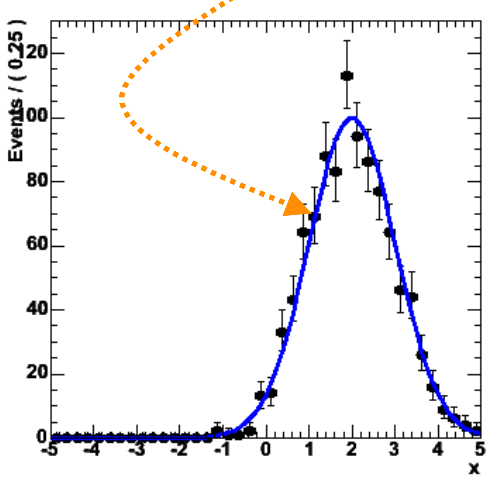
- Overlaying a 2-dim PDF with a 2D (lego) data set doesn't provide much insight

  "You cannot do quantitative analysis with 2D plots"
  (Chris Tully, Princeton)

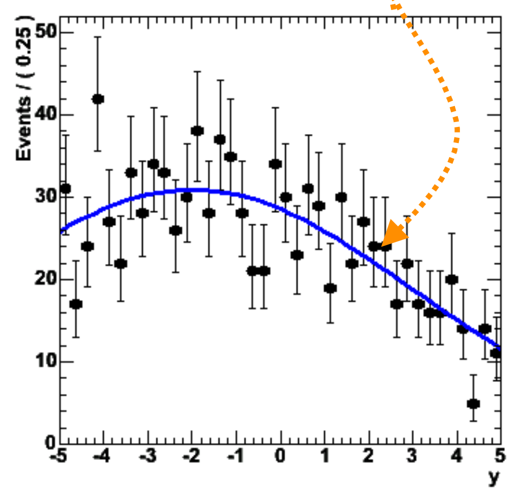


- 1-D projections usually easier

$$f_y(x) = \int F(x,y)\,dy$$



$$f_x(y) = \int F(x,y)\,dx$$

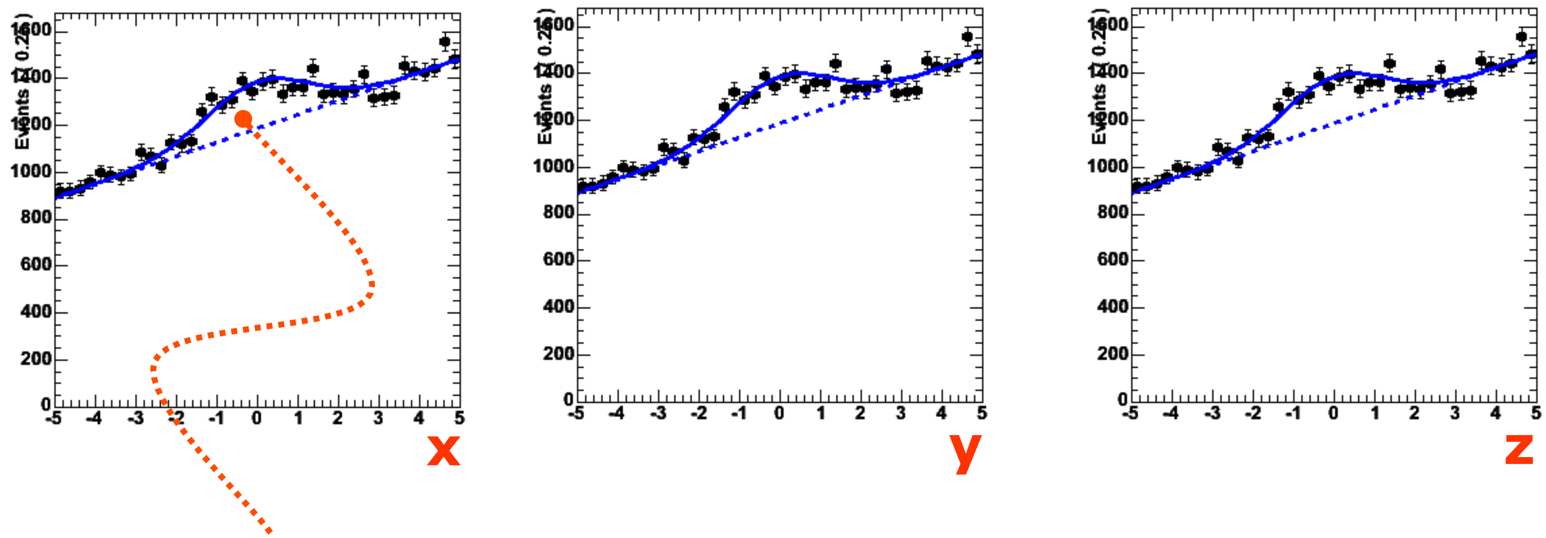


x-y correlations in data and/or model difficult to visualize

Multi-dimensional fits – visualizing your model

- However: plain 1-D projections often don't do justice to your fit
  - Example: 3-Dimensional dataset with 50K events, 2500 signal events
  - Distributions in x,y and z chosen identical for simplicity

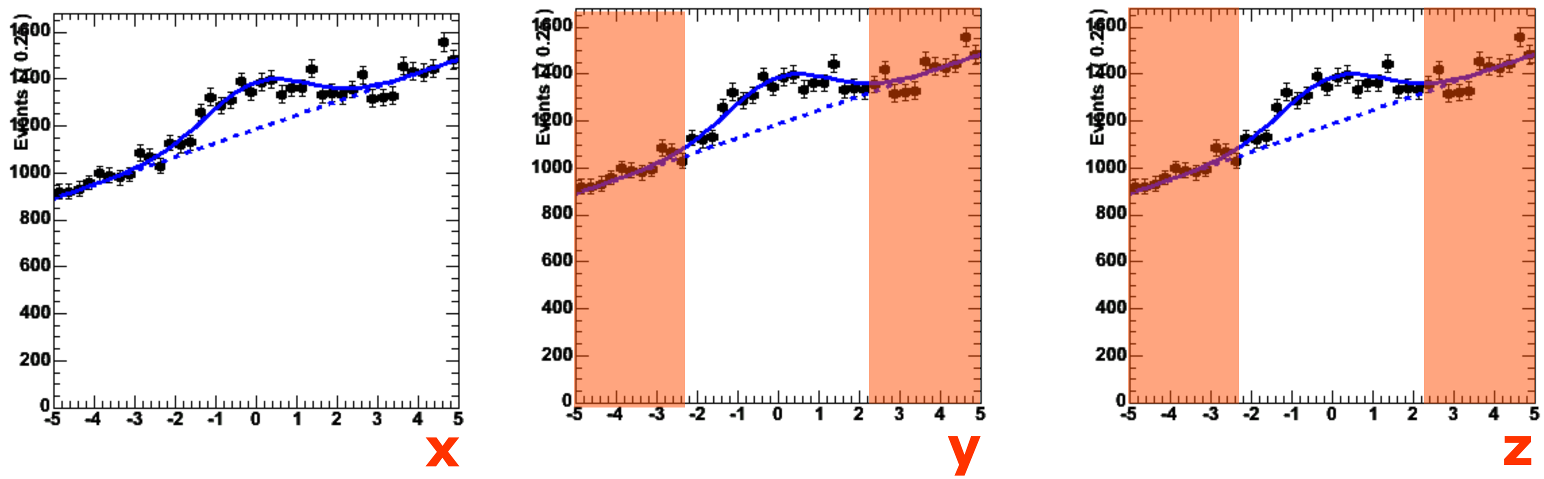
- Plain 1-dimensional projections in x,y,z



- Fit of 3-dimensional model finds N_{sig} = 2440±64
  - Difficult to reconcile with enormous backgrounds in plots

Multi-dimensional fits – visualizing your model

- Reason for discrepancy between precise fit result and large background in 1-D projection plot
  - Events in shaded regions of y,z projections can be discarded without loss of signal



- Improved projection plot: show only events in x projection that are likely to be signal in (y,z) projection of fit model
  - Zeroth order solution: make box cut in (x,y)
  - Better solution: cut on signal probability according to fit model in (y,z)

Multi-dimensional fits – visualizing your model

- Goal: Projection of model and data on x, with a cut on the signal probability in (y,z)
- First task at hand: calculate signal probability according to PDF using only information in (y,z) variables
  - Define 2-dimensional signal and background PDFs in (y,z) by integrating out x variable (and thus discarding any information contained in x dimension)

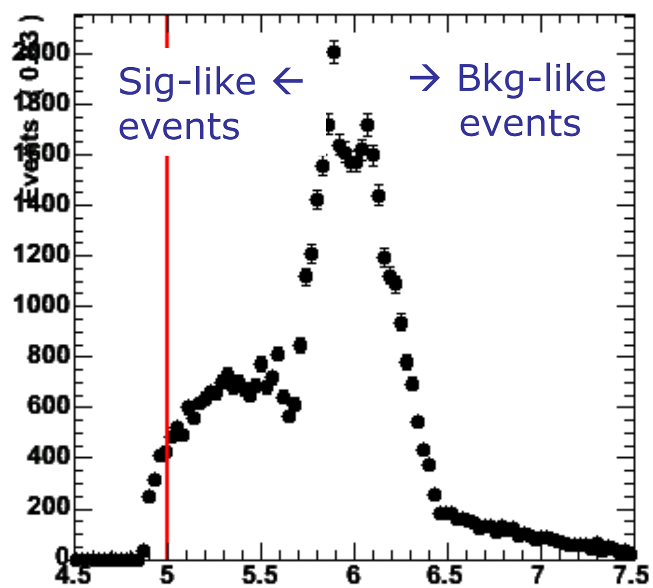
$$F_{SIG}(y,z) = \int S(x,y,z)dx$$

$$F_{BKG}(y,z) = \int B(x,y,z)dx$$

  - Calculate signal probability P(y,z) for all data points (x,y,z)

$$P_{SIG}(y,z) = \frac{F_{SIG}(y,z)}{F_{SIG}(y,z) + F_{BKG}(y,z)}$$

  - Choose sensible cut on P(y,z)



-log(P_{SIG}(y,z))

Plotting regions of a N-dim model – Case study

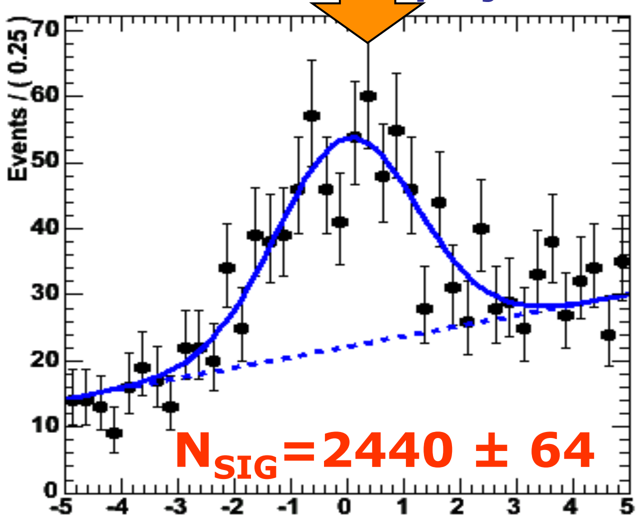
- Next: plot distribution of data, model with cut on $P_{SIG}(y,z)$
  - Data: Trivial
  - Model: Calculate projection of selected regions with Monte Carlo method

1) Generate a toy Monte Carlo dataset $D_{TOY}(x,y,z)$ from $F(x,y,z)$

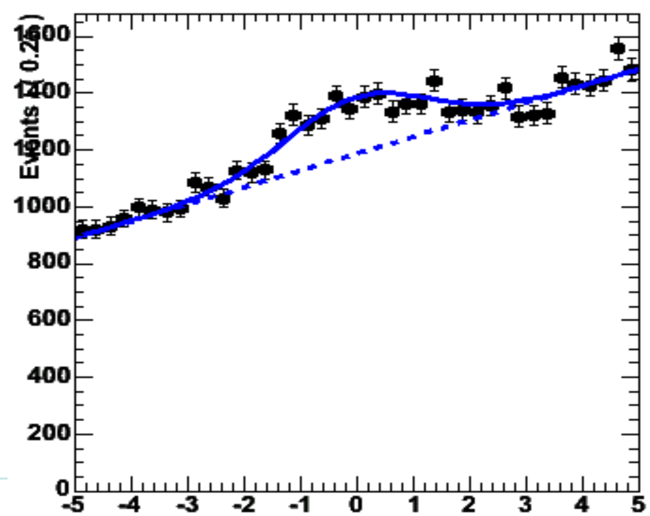
2) Select subset of D_{TOY} with $P_{SIG}(y,z)<C$

3) Plot $$f_C(x) = \sum_{D_{TOY}} F(x, y_i, z_i)$$

Likelihood ratio projection



Plain projection (for comparison)



Alternative: 'sPlots'

- Again, compute signal probability based on variables y and z
- Plot x, *weighted* with the above signal probability
- Overlay signal PDF for x

PRL 93(2004)131801

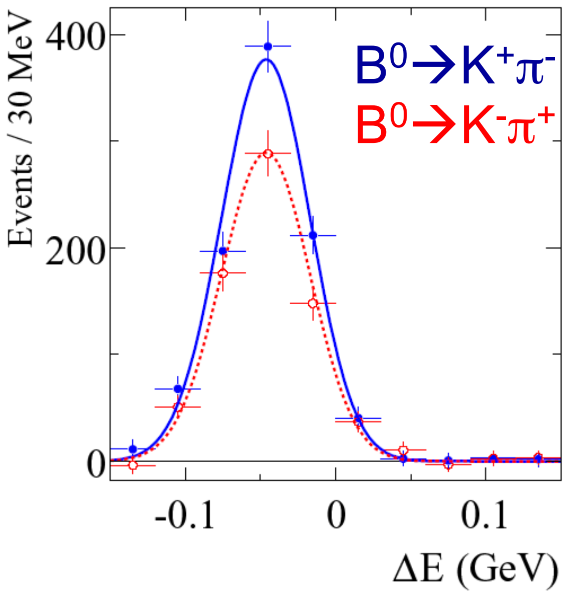


FIG. 2: Distributions of ΔE in data (points with error bars) and the PDFs (curves) used in the maximum likelihood fit for $K^+\pi^-$ (solid circles and solid curve) and $K^-\pi^+$ (open circles and dashed curve). The data are weighted using the background-subtraction technique of Ref. [15] (see text).

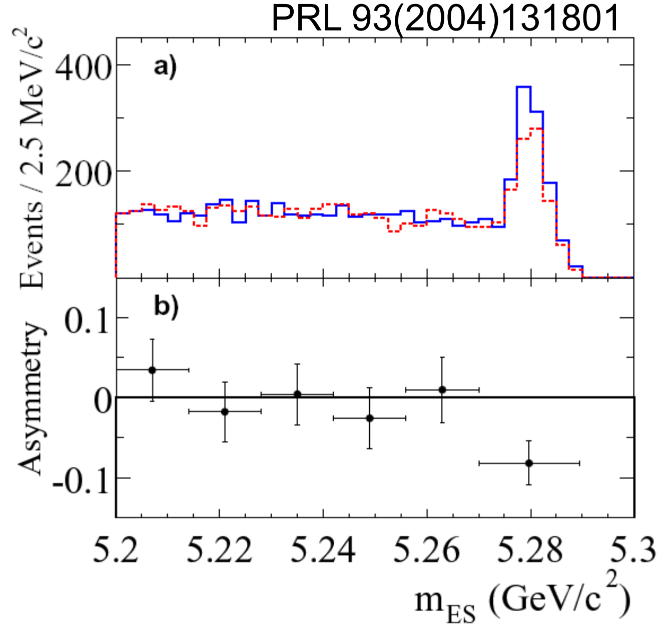


FIG. 3: (a) Distribution of $m_{\rm ES}$ enhanced in $K^+\pi^-$ (solid histogram) and $K^-\pi^+$ (dashed histogram). (b) Asymmetry $\mathcal{A}_{K\pi}$ calculated for ranges of $m_{\rm ES}$. The asymmetry in the highest $m_{\rm ES}$ bin is somewhat diluted by the presence of background.

- See http://arxiv.org/abs/physics/0402083 for more details on sPlots

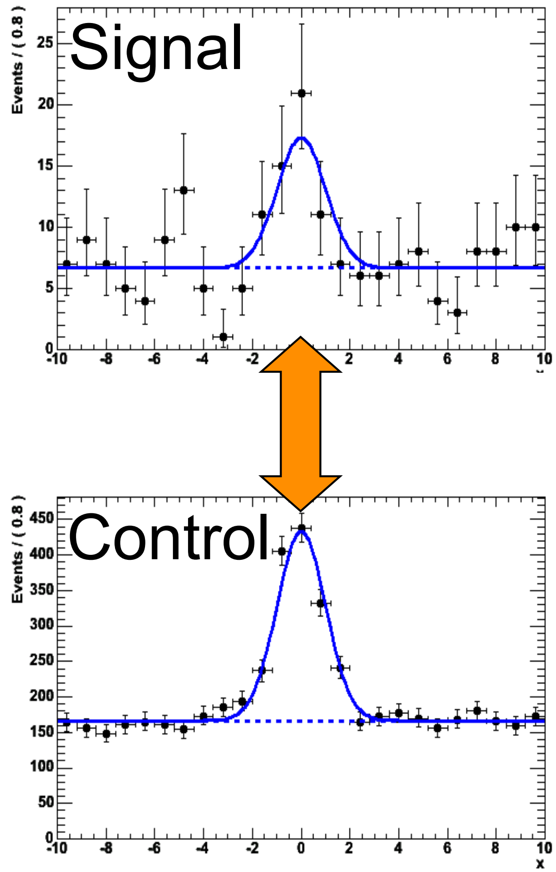
Multidimensional fits – Goodness-of-fit determination

- Warning: Goodness-of-fit measures for multi-dimensional fits are difficult
  - Standard χ^2 test does not work very will 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**
  - Some references to recent progress:
    - PHYSTAT2001, PHYSTAT2003

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 – Error propagation between samples

- 0^{th} order solution:
  - **Fit control sample first, signal sample second** – signal shape parameters fixed from values of control sample fit
  - Signal fit will give correct parameter estimates
  - But error on signal will be underestimated because uncertainties in the determination of the signal shape from the control sample are not included
- 1^{st} order solution
  - **Repeat fit on signal sample at $p \pm \sigma_p$**
  - Observe difference in answer and add this difference in quadrature to error:

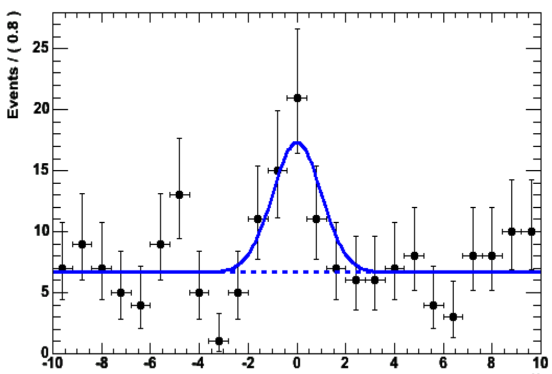
$$\sigma_{tot}^2 = \sigma_{stat}^2 + (N_{sig}^{p-\sigma_p} - N_{sig}^{p+\sigma_p})^2 / 2$$

  - Problem: Error estimate will be incorrect if there is >1 parameter in the control sample fit and there are correlations between these parameters
- Best solution: a simultaneous fit

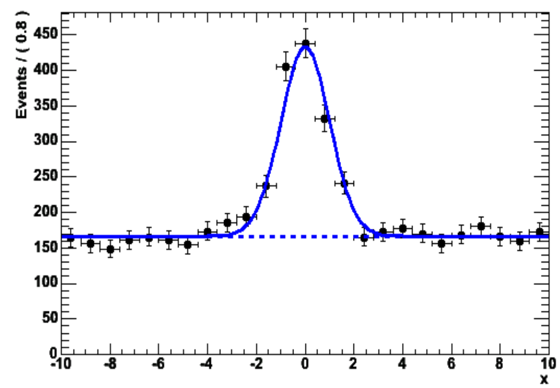
Practical fitting – Simultaneous fit technique

- given data $D_{sig}(\mathbf{x})$ and model $F_{sig}(\mathbf{x};\mathbf{p_{sig}})$ and data $D_{ctl}(\mathbf{x})$ and model $F_{ctl}(\mathbf{x};\mathbf{p_{ctl}})$
  - construct $\chi^2_{sig}(\mathbf{p_{sig}})$ and $\chi^2_{ctl}(\mathbf{p_{ctl}})$ and

$D_{sig}(x)$, $F_{sig}(x;p_{sig})$



$D_{ctl}(x)$, $F_{ctl}(x;p_{ctl})$



- Minimize $\chi^2\,(\mathbf{p_{sig}},\mathbf{p_{ctl}}) = \chi^2_{sig}(\mathbf{p_{sig}}) + \chi^2_{ctl}(\mathbf{p_{ctl}})$
  - All parameter errors, correlations automatically propagated

Practical Estimation – Verifying the validity of your fit

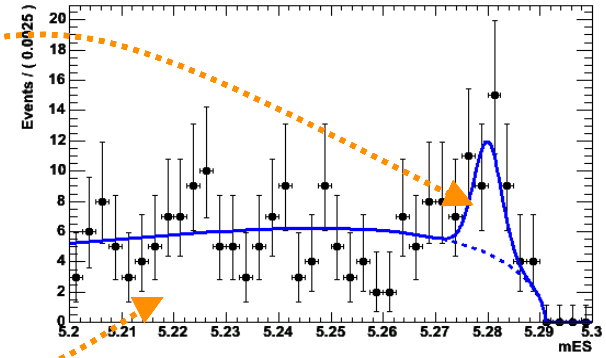
- How to validate your fit? – You want to demonstrate that
  1) Your fit procedure gives on average the correct answer **'no bias'**
  2) The uncertainty quoted by your fit is an accurate measure for the statistical spread in your measurement **'correct error'**

- **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**

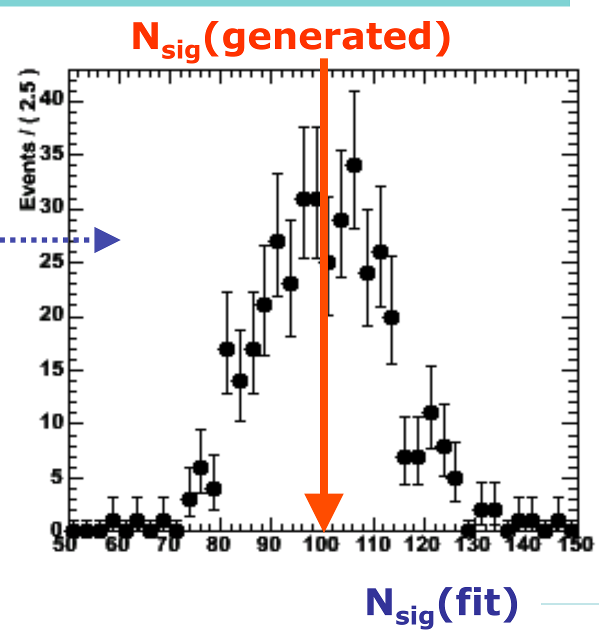
Fit Validation Study – Practical example

- 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)



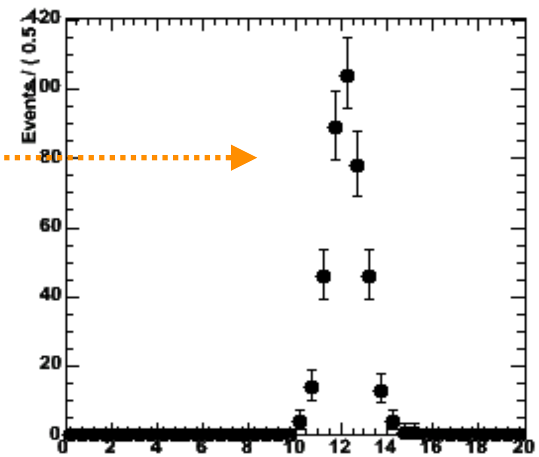
$$F(m; N_{\text{sig}}, N_{\text{bkg}}, \vec{p}_S, \vec{p}_B) = N_{\text{sig}} \cdot G(m; p_S) + N_{\text{bkg}} \cdot A(m; p_B)$$

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



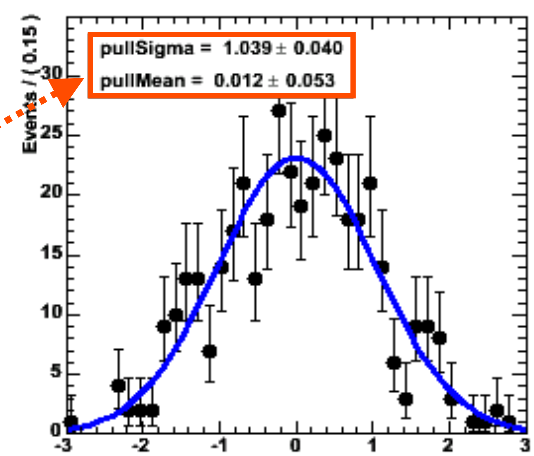
Fit Validation Study – The pull distribution

- What about the validity of the error?
  - Distribution of error from simulated experiments is difficult to interpret…
  - We don't have equivalent of N_{sig}(generated) for the error



$\sigma(N_{sig})$

- Solution: look at the ***pull distribution***
  - Definition: $$\mathrm{pull}(N_{sig}) = \frac{N_{sig}^{fit} - N_{sig}^{true}}{\sigma_N^{fit}}$$
  - 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



pull (N_{sig})

Fit Validation Study – Low statistics example

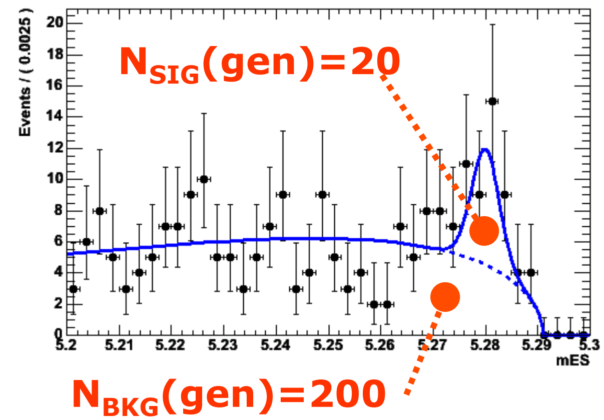
- Special care should be taken when fitting small data samples
  - Also if fitting for small signal component in large sample

- Possible causes of trouble
  - χ^2 estimators may become approximate as Gaussian approximation of Poisson statistics becomes inaccurate
  - ML estimators may no longer be efficient
    → error estimate from 2nd derivative may become inaccurate
  - Bias term proportional to 1/N of ML and χ^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

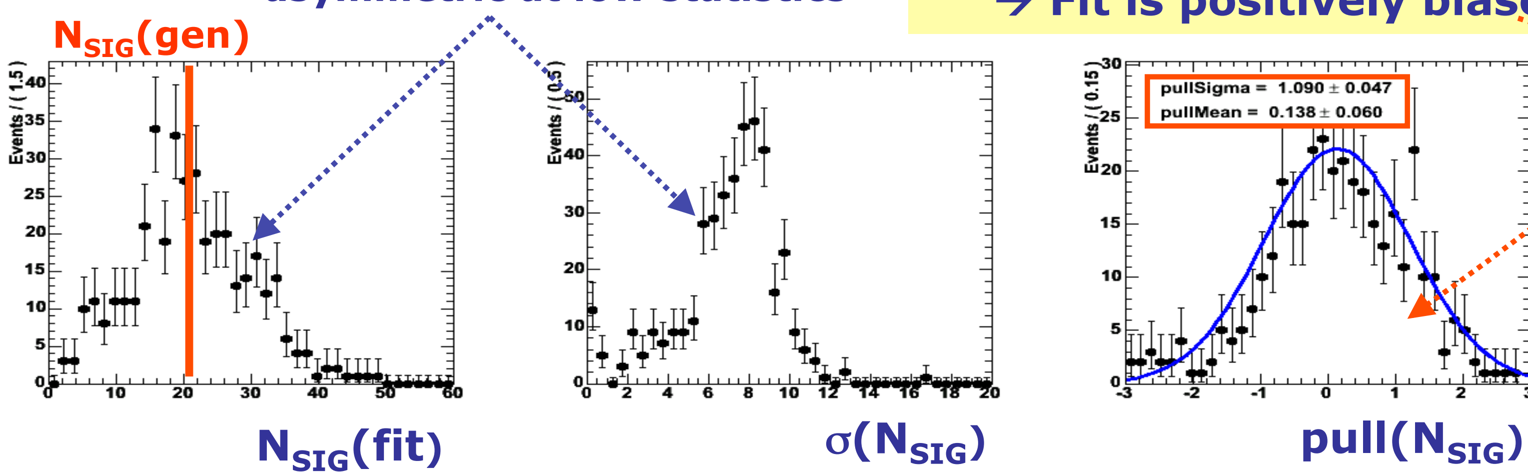
- Low statistics example:
  - Scenario as before but now with 200 bkg events and **only 20 signal events** (instead of 100)



- Results of simulation study

Distributions become asymmetric at low statistics

Pull mean is 2.3σ away from 0
→ Fit is positively biased!



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

Fit Validation Study – How to obtain 10.000.000 simulated events?

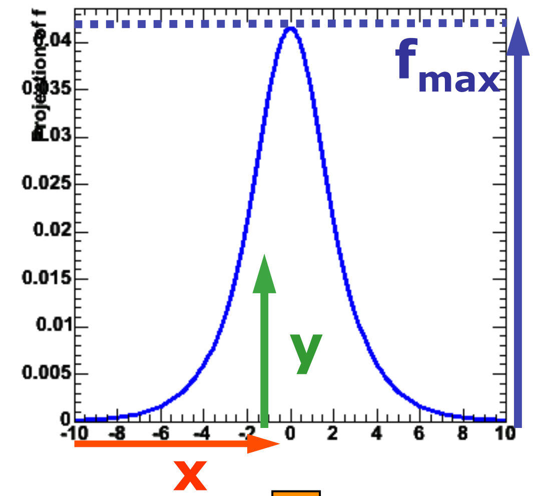
- Practical issue: usually you need very large amounts of simulated events for a fit validation study
  - Of order 1000x number of events in your fit, easily >1.000.000 events
  - Using data generated through a full GEANT-based detector simulation can be prohibitively expensive

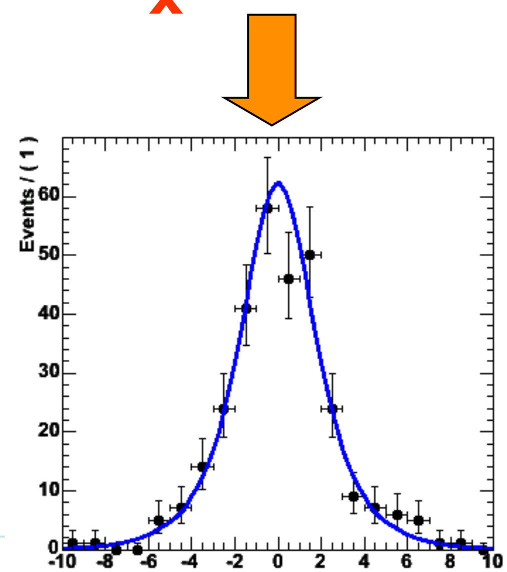
- Solution: Use events sampled directly from your fit function
  - Technique named '*Toy Monte Carlo*' sampling
  - Advantage: Easy to do and very fast
  - Good to determine fit bias due to low statistics, choice of parameterization, boundary issues etc
  - Cannot be used to test assumption that went into model (e.g. absence of certain correlations). Still need full GEANT-based simulation for that.

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)**



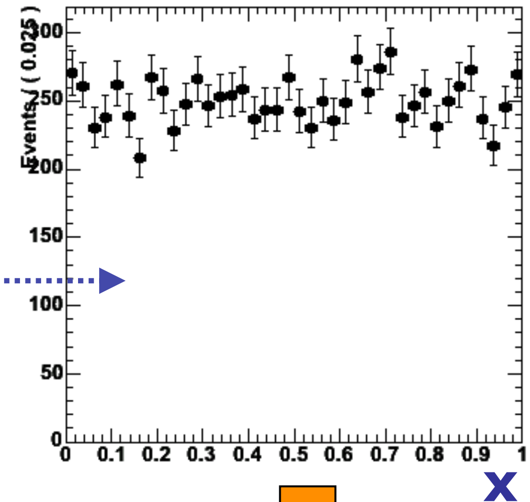


- PRO: Easy, always works
- CON: It can be inefficient if function is strongly peaked. Finding maximum empirically through random sampling can be lengthy in >2 dimensions

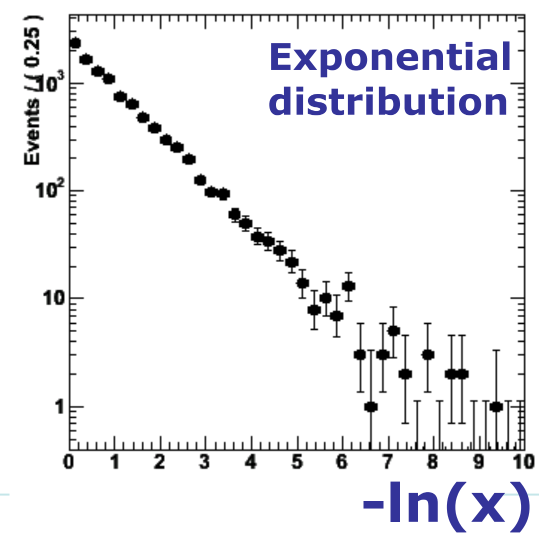
Toy MC generation – Inversion method

◆ 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)

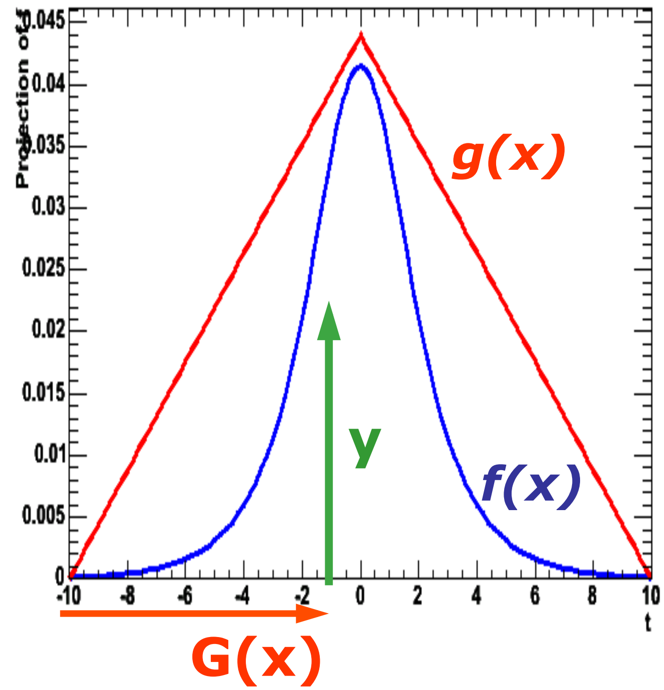


- PRO: Maximally efficient
- CON: Only works for invertible funct

Toy MC Generation in a nutshell

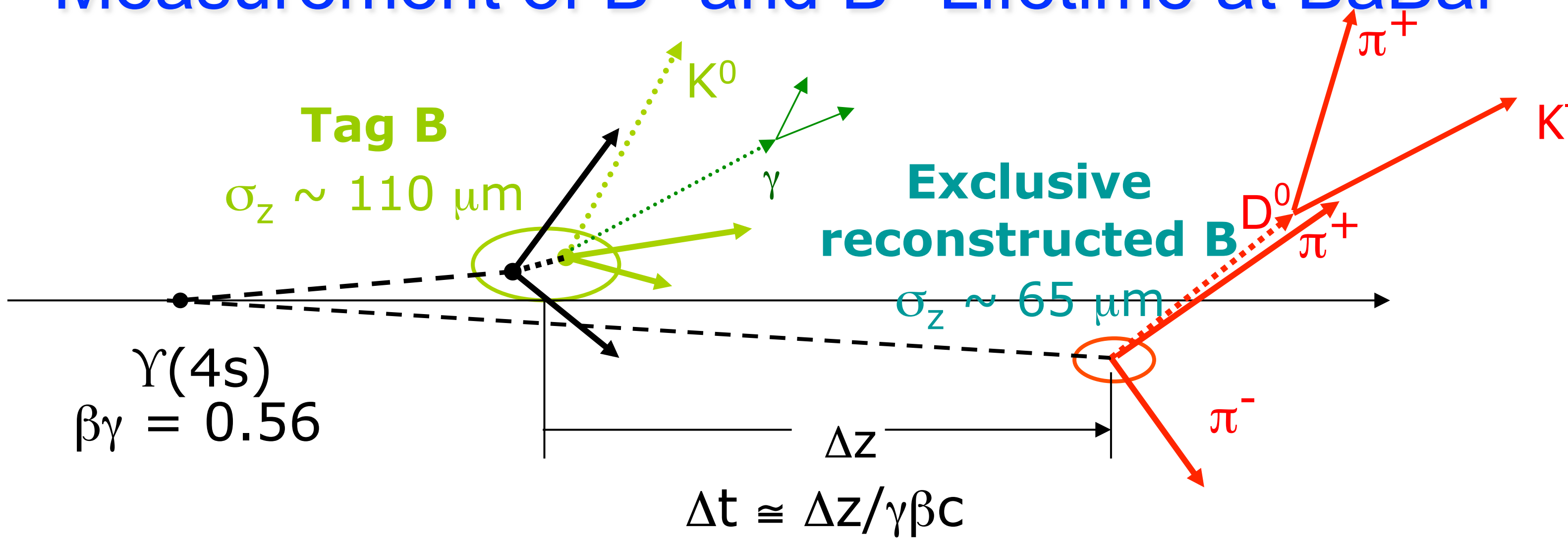
- Hybrid: Importance sampling

1) Find 'envelope function' g(x) that is invertible into G(x) and that fulfills g(x)>=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

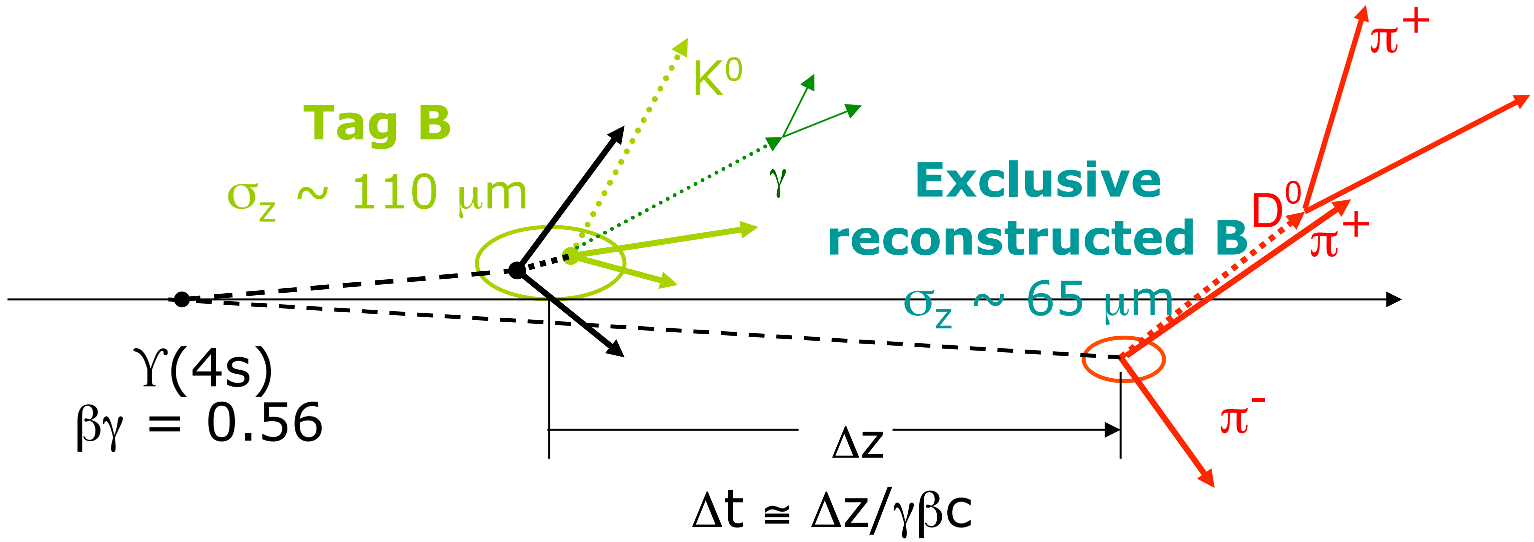


- PRO: Faster than plain accept/reject sampling
  Function does not need to be invertible
- CON: Must be able to find invertible envelope function

A ‘simple’ real-life example: Measurement of B^0 and B^+ Lifetime at BaBar



Measurement of B^0 and B^+ Lifetime at BaBar

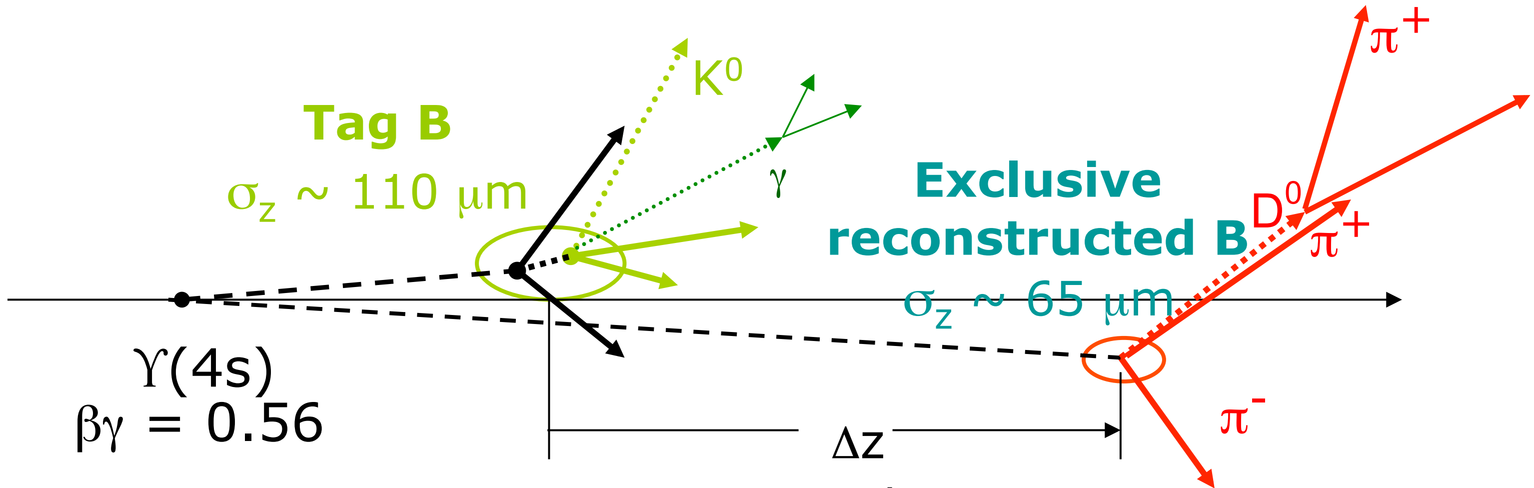


3. Reconstruct inclusively the vertex of the "other" B meson (B_{TAG})

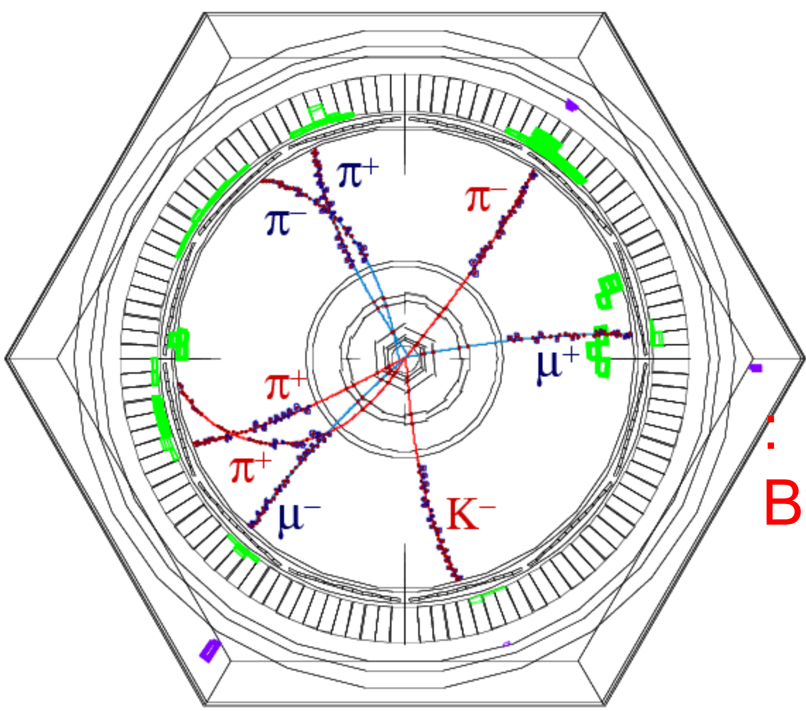
1. Fully reconstruct one B meson (B_{REC})
2. Reconstruct the decay vertex

4. compute the proper time difference Δt
5. Fit the Δt spectra

Measurement of B^0 and B^+ Lifetime at BaBar



$\Delta t \cong \Delta z/\gamma\beta c$



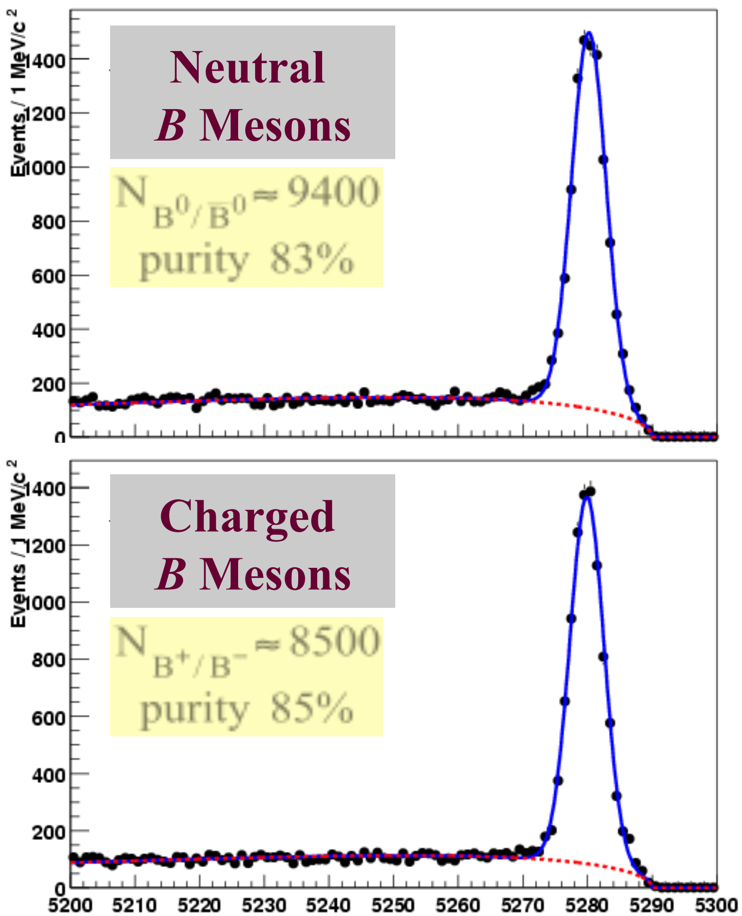
1. Fully reconstruct one B meson (B_{REC})
2. Reconstruct the decay vertex

$B^0 \rightarrow D^{*+} \pi^-$

$\rightarrow D^0 \pi^+$

$\rightarrow K^- \pi^+$

Measurement of B^0 and B^+ Lifetime at BaBar



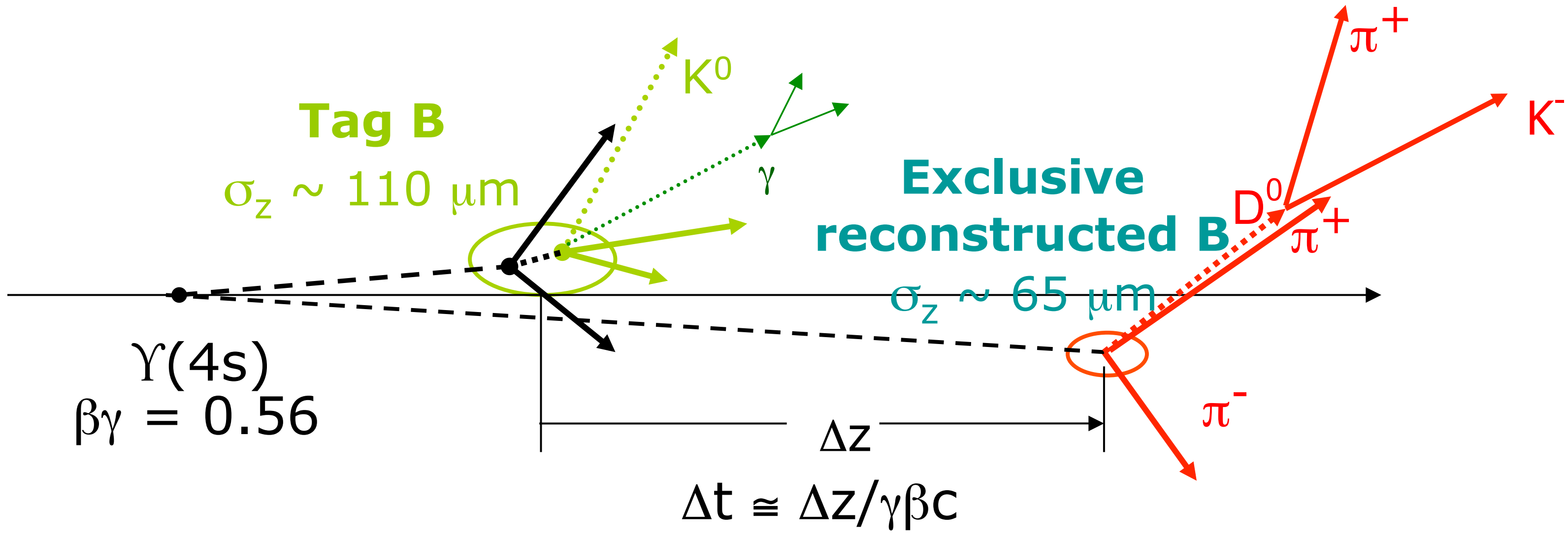
Energy substituted mass [MeV/c^2]

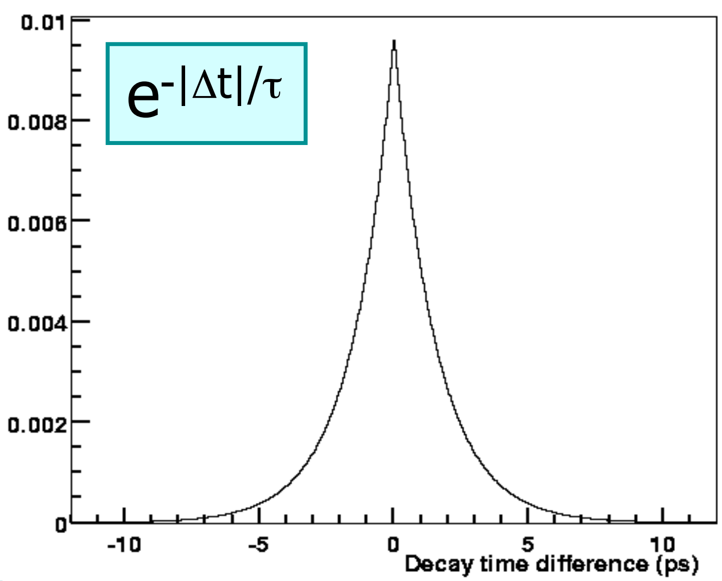
$$m_{ES} = \sqrt{\left(\frac{\sqrt{s}}{2}\right)^2 - \left(p_B^{cm}\right)^2}$$

1. Fully reconstruct one B meson (B_{REC})
   a) Classify signal and background

Utilize that @ BaBar, in case of signal, One produces exactly 2 B mesons, so their energy (in the center-of-mass) is half the center-of-mass energy of the collider

Signal Propertime PDF





$$F(t_{rec}, t_{tag}; \tau) = \frac{e^{-t_{rec}/\tau}}{\tau} \cdot \frac{e^{-t_{tag}/\tau}}{\tau}$$

$$t_{rec}, t_{tag} \rightarrow \Delta t = t_{rec} - t_{tag}, \Sigma t = t_{rec} + t_{tag}$$

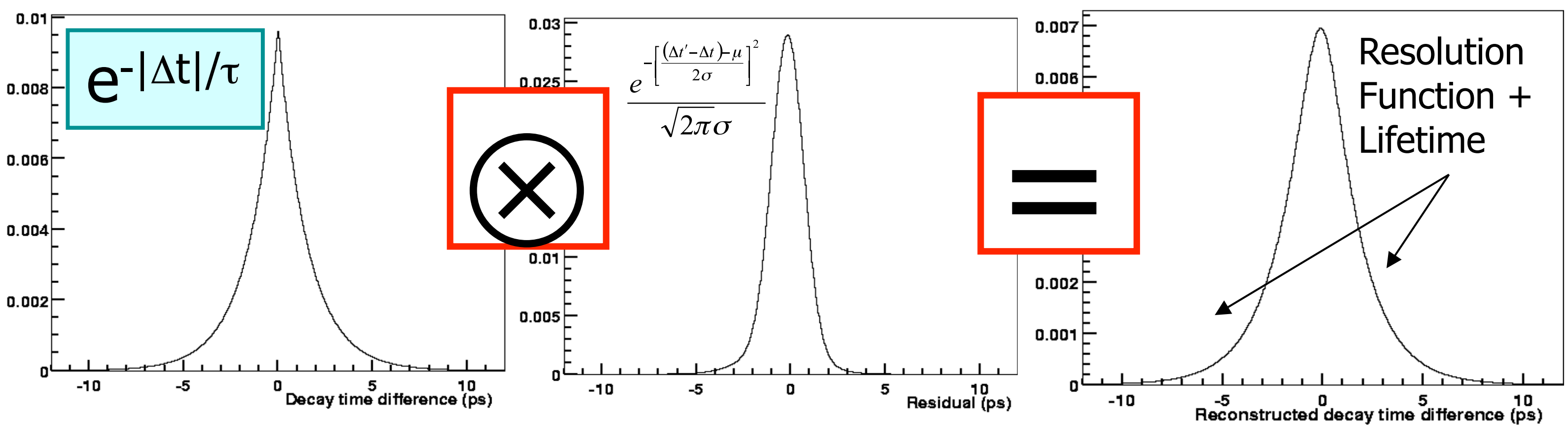
$$F(\Delta t, \Sigma t; \tau) = \frac{e^{-\Sigma t/\tau}}{2\tau^2}$$

$$F(\Delta t; \tau) = \int_{|\Delta t|}^{+\infty} d\Sigma t \ F(\Delta t, \Sigma t; \tau) = \frac{e^{-|\Delta t|/\tau}}{2\tau}$$

Including the detector response…

- Must take into account the detector response
  - Convolve ‘physics pdf’ with ‘response fcn’ (aka resolution fcn)
  - Example:

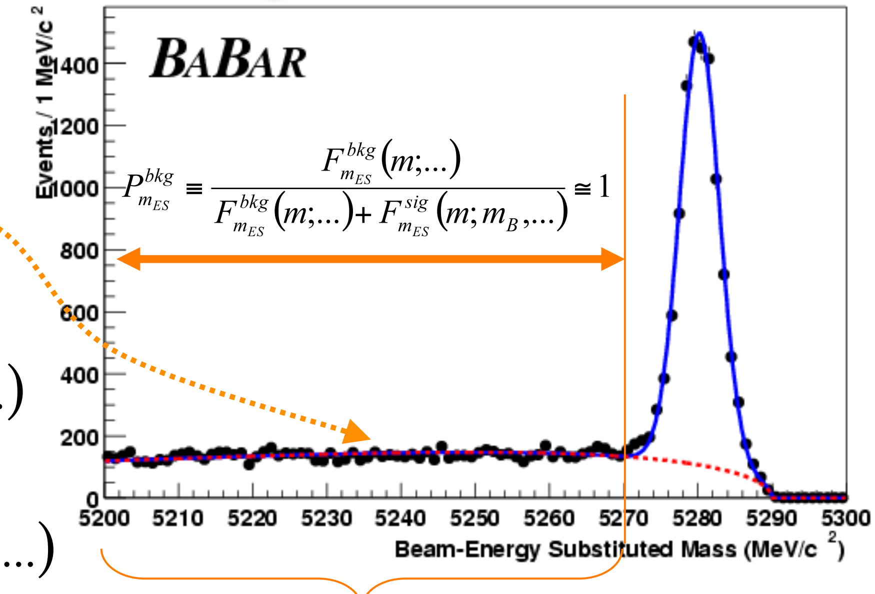
$$F(\Delta t;\tau,\mu,\sigma)=\int_{-\infty}^{+\infty} d\Delta t' \, \frac{e^{-|\Delta t'|/\tau}}{2\tau} \, \frac{e^{-\left[\frac{(\Delta t'-\Delta t)-\mu}{2\sigma}\right]^2}}{\sqrt{2\pi\sigma}}$$

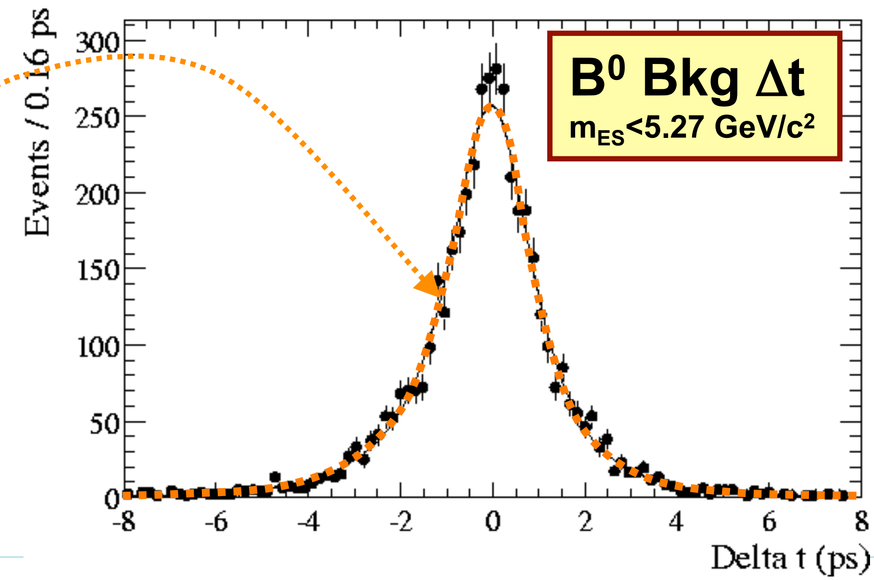


- Caveat: the real-world response function is somewhat more complicated
  - eg. additional information from the reconstruction of the decay vertices is used…

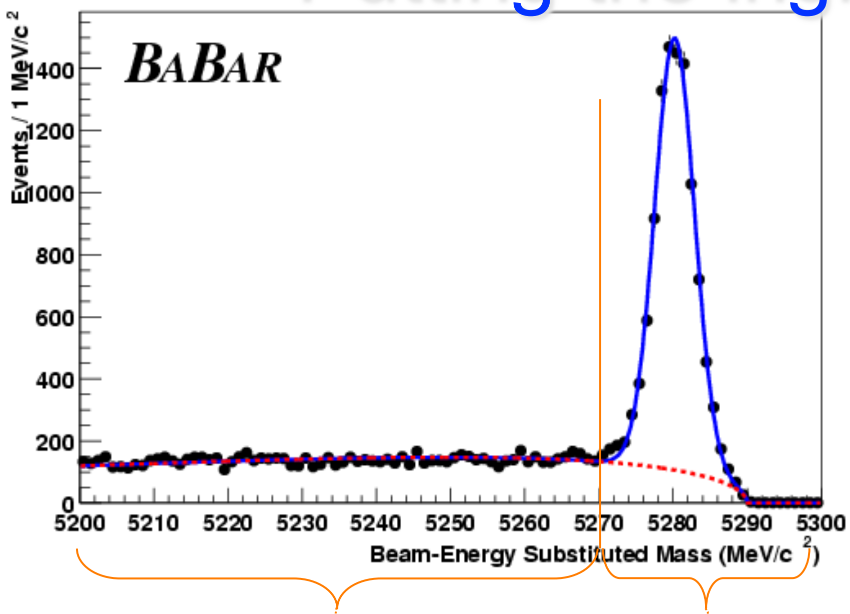
How to deal with the background?

$$F(m,\Delta t;\ldots) = F_{m_{ES}}^{sig}(m;m_B,\ldots)F_{\Delta t}^{sig}(\Delta t;\tau,\ldots)$$
$$+F_{m_{ES}}^{bkg}(m;\ldots)F_{\Delta t}^{sig}(\Delta t;\ldots)$$
$$= \frac{F_{m_{ES}}^{bkg}(m;\ldots)}{F_{m_{ES}}^{bkg}(m;\ldots)+F_{m_{ES}}^{sig}(m;m_B,\ldots)}F_{\Delta t}^{bkg}(\Delta t;\ldots)$$
$$+\frac{F_{m_{ES}}^{sig}(m;m_B,\ldots)}{F_{m_{ES}}^{bkg}(m;\ldots)+F_{m_{ES}}^{sig}(m;m_B,\ldots)}F_{\Delta t}^{sig}(\Delta t;\tau,\ldots)$$
$$= P_{m_{ES}}^{sig}(m;\ldots)F_{\Delta t}^{sig}(\Delta t;\tau,\ldots)$$
$$+P_{m_{ES}}^{bkg}(m;\ldots)F_{\Delta t}^{bkg}(\Delta t;\ldots)$$



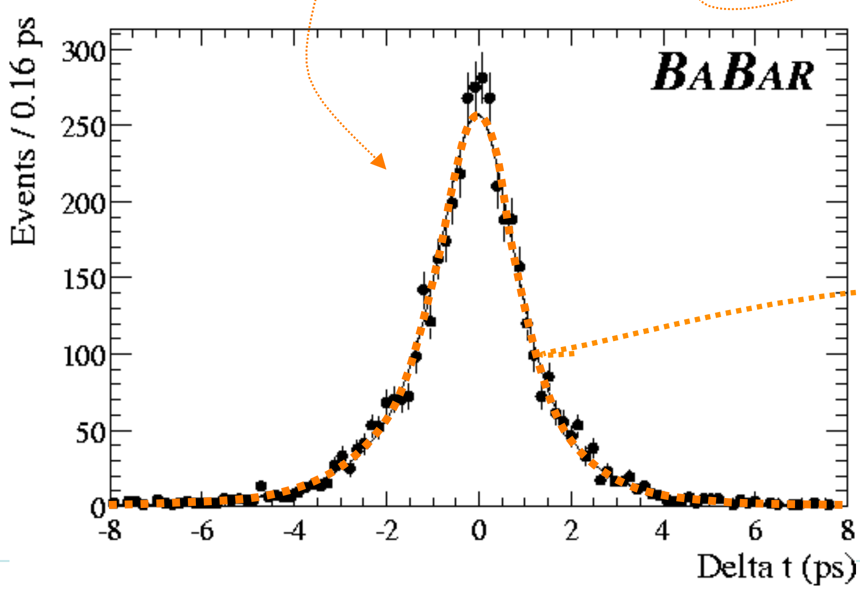


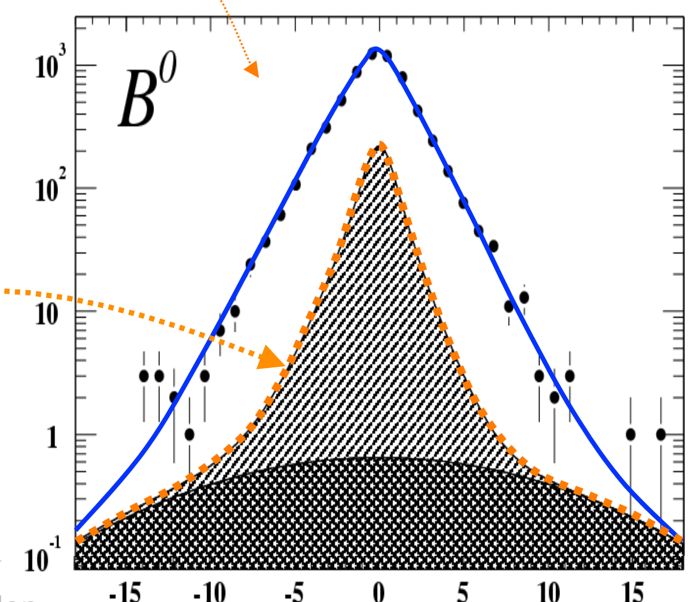
Putting the ingredients together…



$$-\ln L(\tau,\ldots) = \sum_i \ln F(m_i, \Delta t_i; \tau, \ldots)$$

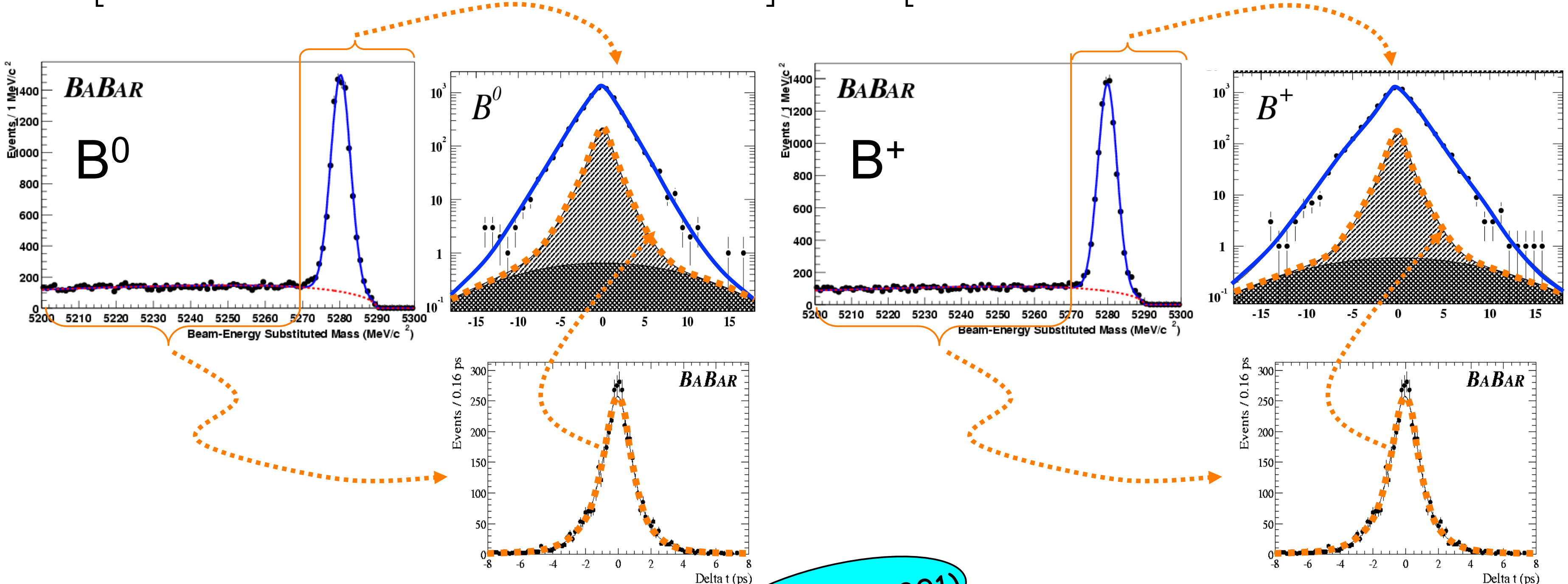
$$F(m, \Delta t; \tau, \ldots) = F^{sig}_{m_{ES}}(m;\ldots) F^{sig}_{\Delta t}(\Delta t; \tau, \ldots) + F^{bkg}_{m_{ES}}(m;\ldots) F^{bkg}_{\Delta t}(\Delta t;\ldots)$$





Measurement of B^0 and B^+ Lifetime at BaBar

$$\sum_{B^0} \ln\left[\begin{array}{l} F^{sig}_{m_{ES}}(m_i; m_B, \sigma_B) \times F^{sig}_{\Delta t}(\Delta t; \tau^0) \otimes R(\Delta t_i - \Delta t; \vec{p}_{res}) + \\ F^{bkg}_{m_{ES}}(m_i; \vec{p}^{\,bkg,B^0}_{m_{ES}}) \times F^{bkg}_{\Delta t}(\Delta t_i; \vec{p}^{\,bkg,B^0}_{\Delta t}) \end{array}\right] + \sum_{B^+} \ln\left[\begin{array}{l} F^{sig}_{m_{ES}}(m_i; m_B, \sigma_B) \times F^{sig}_{\Delta t}(\Delta t; \tau^+) \otimes R(\Delta t_i - \Delta t; \vec{p}_{res}) + \\ F^{bkg}_{m_{ES}}(m_i; p^{bkg,B^+}_{m_{ES}}) \times F^{bkg}_{\Delta t}(\Delta t_i; \vec{p}^{\,bkg,B^+}_{\Delta t}) \end{array}\right]$$



Strategy: fit mass, fix those parameters
then perform Δt fit.

19 free parameters in Δt fit:
- 2 lifetimes
- 5 resolution parameters
- 12 parameters for empirical bkg description

PRL 87 (2001)

τ_0 = 1.546 ± 0.032 ± 0.022 ps

$\tau_\pm$ = 1.673 ± 0.032 ± 0.022 ps

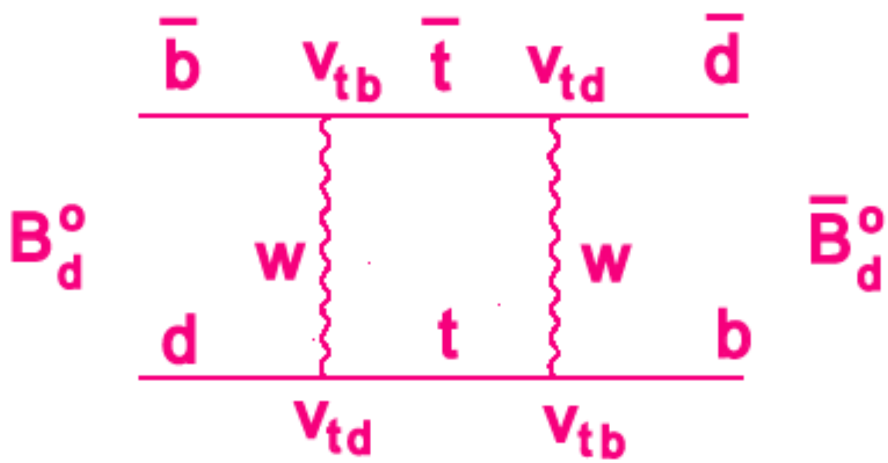
$\tau_\pm / \tau_0$ = 1.082 ± 0.026 ± 0.011

Δt RF parameterization

Common Δt response function for B^+ and B^0

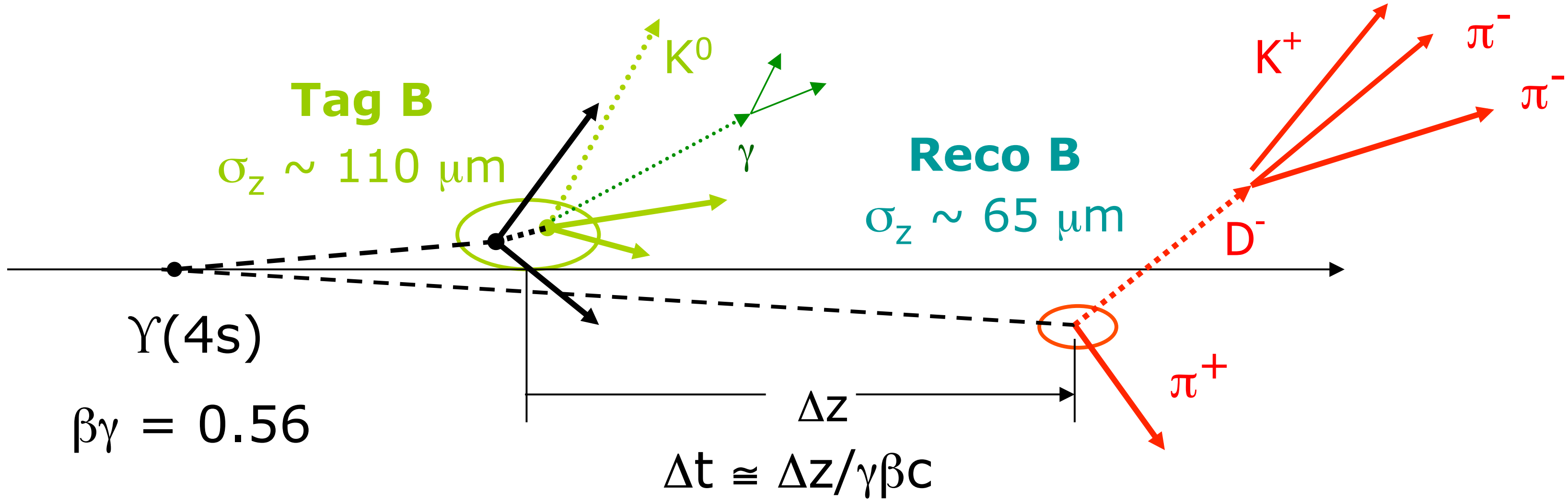
Neutral B meson mixing

- B mesons can ‘oscillate’ into B mesons – and vice versa
  - Process is describe through 2nd order weak diagrams like this:



Observation of B^0B^0 mixing in 1987 was the first evidence of a *really* heavy top quark…

Measurement of B^0B^0 mixing

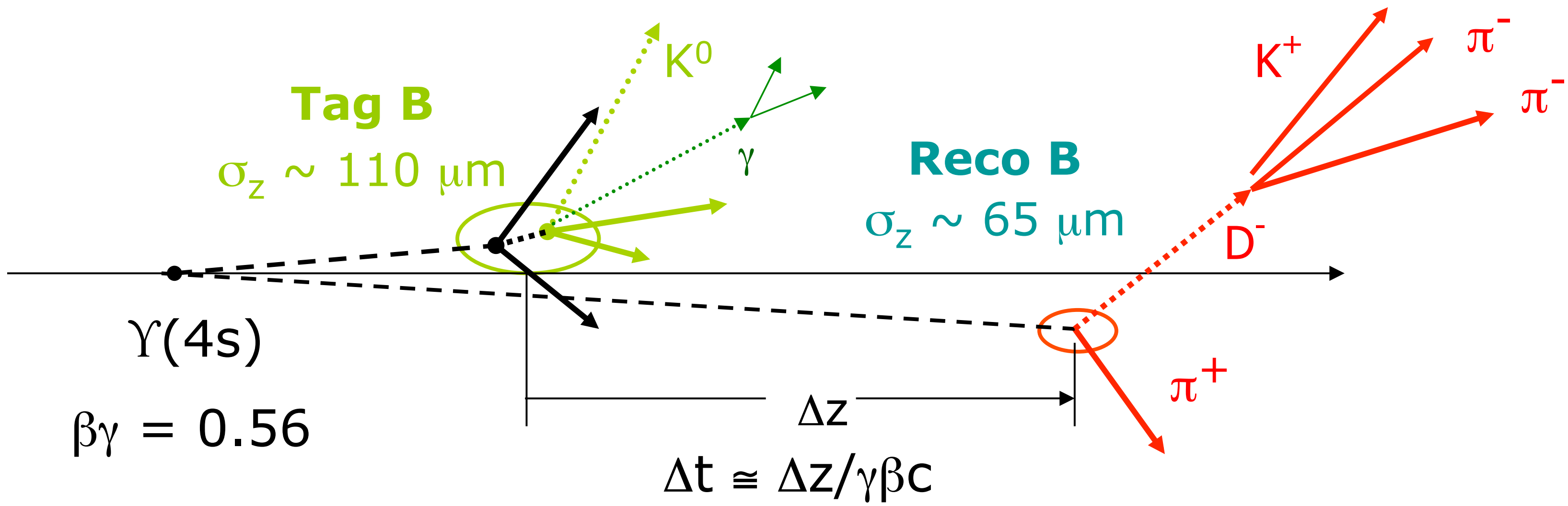


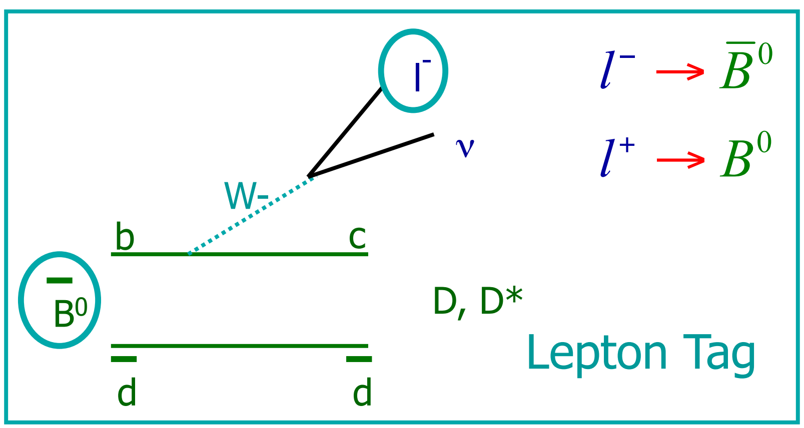
3. Reconstruct Inclusively the vertex of the "other" B meson (B_{TAG}) ✓
4. Determine the flavor of B_{TAG} to separate Mixed and Unmixed events

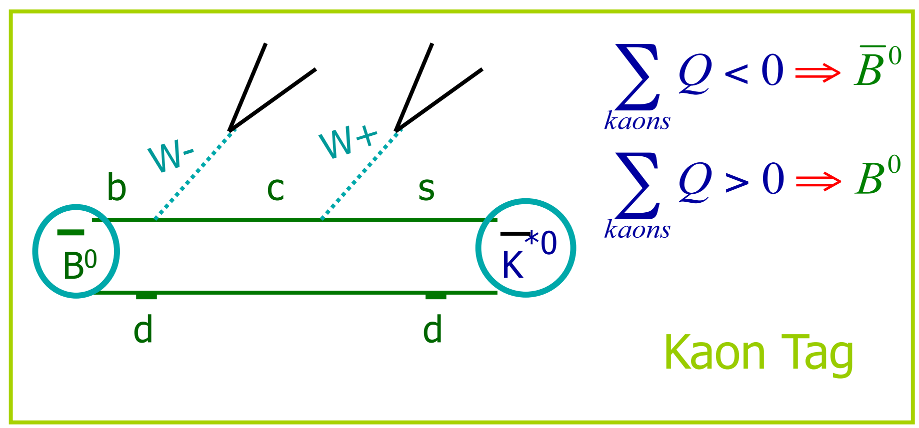
1. Fully reconstruct one B meson in flavor eigenstate (B_{REC}) ✓
2. Reconstruct the decay vertex ✓

5. compute the proper time difference Δt ✓
6. Fit the Δt spectra of mixed and unmixed events

Determine the flavour of the 'other' B







Δt distribution of mixed and unmixed events

perfect
flavor tagging & time resolution

The image cannot be displayed. Your computer may not have enough memory to open the image, or the image may have been corrupted. Restart your computer, and then open the file again. If the red x still appears, you may have to delete the image and then insert it again.

realistic
mis-tagging & finite time resolution

The image cannot be displayed. Your computer may not have enough memory to open the image, or the image may have been corrupted. Restart your computer, and then open the file again. If the red x still appears, you may have to delete the image and then insert it again.

$$f_{\substack{\text{Unmix}\\\text{Mix}}}(\Delta t) = \left\{ \frac{e^{-|\Delta t|/\tau_{B_d}}}{4\tau_{B_d}} \times \left(1 \pm (1-2w)\cos(\Delta m_d \Delta t)\right) \right\} \otimes \text{ResolutionFunction}$$

Unmixed: $B^0_{flav}\bar{B}^0_{tag}$ or $\bar{B}^0_{flav}B^0_{tag}$

Mixed: $B^0_{flav}B^0_{tag}$ or $\bar{B}^0_{flav}\bar{B}^0_{tag}$

Δm_d: oscillation frequency

w: the fraction of wrongly tagged events

Normalization and Counting…

- The PDF's used for unmixed (+) and mixed (-) events are usually:

$$f_{\pm}(\Delta t; \Gamma, \Delta m_d) = \frac{\Gamma}{4} e^{-\Gamma|\Delta t|} (1 \pm \cos \Delta m_d \Delta t)$$

- Note that $f_{\pm}$ are *not* individually normalized, but their sum is:

$$\int_{-\infty}^{\infty} d\Delta t \, [f_+(\Delta t; \Gamma, \Delta m_d) + f_-(\Delta t; \Gamma, \Delta m_d)] = 1$$

- The individual normalizations are:

$$N_{\pm} = \int_{-\infty}^{\infty} d\Delta t f_{\pm}(\Delta t; \Gamma, \Delta m_d) = \frac{1}{2} \pm \frac{1}{2} \frac{1}{1 + x_d^2}$$

where $x_d = \frac{\Delta m_d}{\Gamma}$.

- The fraction of mixed events is $\chi_d = \frac{N_-}{N_+ + N_-} = \frac{1}{2} \frac{x_d^2}{1+x_d^2}$

- If one wants to fit just mixed events, one maximizes:

$$\mathcal{L}_-(\Gamma, \Delta m_d) = \prod_{i=1}^{M} \frac{f_-(\Delta t_i; \Gamma, \Delta m_d)}{N_-(\Gamma, \Delta m_d)}$$

where M is the number of mixed events

- For mixed events, replace $f_- \leftrightarrow f_+$; $N_- \leftrightarrow N_+$ and $M \leftrightarrow U$ where U is the number of unmixed events

- When fitting just the Δt shapes:

$$\mathcal{L} = \mathcal{L}_+ \times \mathcal{L}_- = \prod_U \frac{f_-}{N_-} \times \prod_M \frac{f_+}{N_+}$$

- The likelihood to measure M mixed and U unmixed events is:

$$\begin{aligned}
\mathcal{L}_{\text{norm}} &= \frac{(M+U)!}{M!U!} (p_{\text{mix}})^M (1 - p_{\text{mix}})^U \\
&\propto \chi_d^M (1 - \chi_d)^U \\
&= \left(\frac{N_-}{N_+ + N_-}\right)^M \left(\frac{N_+}{N_+ + N_-}\right)^U \\
&= (N_-)^M (N_+)^U
\end{aligned}$$

- Combining shapes and and the normalization:

$$\begin{aligned}
\mathcal{L} &= \mathcal{L}_+ \mathcal{L}_- \mathcal{L}_{\text{norm}} \\
&\propto \left(\prod_M \frac{f_-}{N_-}\right) \times \left(\prod_U \frac{f_+}{N_+}\right) \times (N_-)^M (N_+)^U \\
&= \left(\prod_M f_-\right) \left(\prod_U f_+\right)
\end{aligned}$$

•Counting matters!

•Likelihood fit (implicitly!) uses the integrated rates unless you explicitly normalize both populations seperately

•Acceptance matters!

•unless acceptance for both populations is the same

➔ Can/Must check that shape result consistent with counting

Mixing Likelihood fit

Unbinned maximum likelihood fit to flavor-tagged neutral B sample

$$f_{\substack{\text{Unmix}\\\text{Mix}}}(\Delta t) = \left\{ \frac{e^{-|\Delta t|/\tau_{B_d}}}{4\tau_{B_d}} \times \left(1 \pm (1-2w)\cos(\Delta m_d \Delta t)\right) \right\} \otimes R$$

Fit Parameters

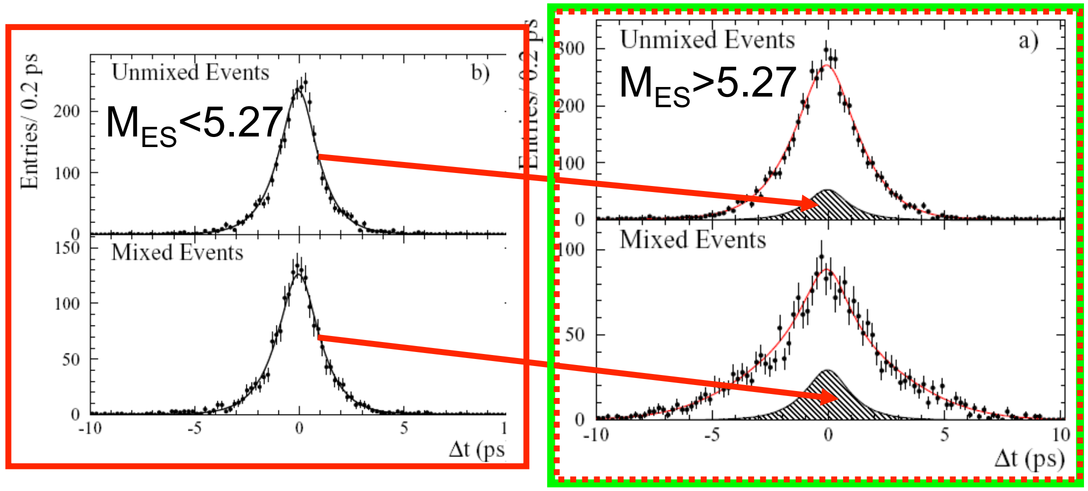
Δm_d	1
Mistag fractions for B^0 and $\overline{B^0}$ tags	8
Signal resolution function(scale factor,bias,fractions)	8+8=16
Empirical description of background Δt	19
B lifetime fixed to the PDG value	τ_B = 1.548 ps

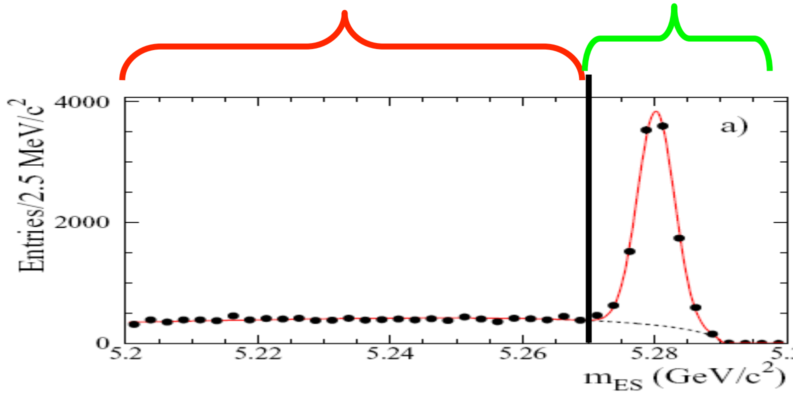
44 total free parameters ⇒ All Δt parameters extracted from data

Complex Fits?

Parameter	Fit Result Run 1	Fit Result Run 2	Correlation Run 1	Correlation Run 2
Δm_d [ps^{-1}]	0.516 ± 0.016			
Signal Resolution Function				
S_1 (core)	1.37 ± 0.09	1.18 ± 0.11	0.25	0.16
$b_1(\Delta t)$ `lepton` (core)	0.06 ± 0.13	-0.04 ± 0.16	0.08	0.00
$b_1(\Delta t)$ `kaon` (core)	-0.22 ± 0.08	-0.25 ± 0.09	0.03	0.00
$b_1(\Delta t)$ `NT1` (core)	-0.07 ± 0.15	-0.45 ± 0.21	-0.00	0.00
$b_1(\Delta t)$ `NT2` (core)	-0.46 ± 0.12	-0.20 ± 0.16	0.01	0.03
$b_2(\Delta t)$ (tail)	-5.0 ± 4.2	-7.5 ± 2.4	0.04	0.06
f_2(tail)	0.014 ± 0.020	0.015 ± 0.010	0.06	0.07
f_3(outlier)	0.008 ± 0.004	0.000 ± 0.014	-0.09	0.01
Signal dilutions				
$\langle D\rangle$, `lepton`	0.842 ± 0.028		0.24	
$\langle D\rangle$, `kaon`	0.669 ± 0.023		0.30	
$\langle D\rangle$, `NT1`	0.563 ± 0.044		0.11	
$\langle D\rangle$, `NT2`	0.313 ± 0.041		0.11	
ΔD, `lepton`	-0.006 ± 0.045		0.02	
ΔD, `kaon`	0.024 ± 0.033		0.01	
ΔD, `NT1`	-0.086 ± 0.068		0.00	
ΔD, `NT2`	0.100 ± 0.060		0.00	
Background properties				
τ, mixing bkgd [ps]	0.853 ± 0.036		-0.01	
$f(\tau = 0)$, mixing bkgd, `lepton`	0.05 ± 0.10		0.01	
$f(\tau = 0)$, mixing bkgd, `kaon`	0.42 ± 0.05		0.01	
$f(\tau = 0)$, mixing bkgd, `NT1`	0.33 ± 0.08		0.01	
$f(\tau = 0)$, mixing bkgd, `NT2`	0.32 ± 0.08		0.01	
Background resolution function				
S_1 (core)	1.211 ± 0.043	1.131 ± 0.046	-0.00	0.00
$b_1(\Delta t)$ (core)	-0.135 ± 0.031	-0.015 ± 0.038	-0.00	-0.00
f_3 (outlier)	0.022 ± 0.004	0.036 ± 0.007	-0.01	0.02
Background dilutions				
$\langle D\rangle$, `lepton`, $\tau = 0$	0.0 ± 2.9		-0.02	
$\langle D\rangle$, `kaon`, $\tau = 0$	0.52 ± 0.08		-0.03	
$\langle D\rangle$, `NT1`, $\tau = 0$	0.67 ± 0.27		-0.01	
$\langle D\rangle$, `NT2`, $\tau = 0$	-0.05 ± 0.13		-0.00	
$\langle D\rangle$, `lepton`, $\tau > 0$	0.34 ± 0.13		0.02	
$\langle D\rangle$, `kaon`, $\tau > 0$	0.26 ± 0.06		0.04	
$\langle D\rangle$, `NT1`, $\tau > 0$	-0.13 ± 0.11		0.01	
$\langle D\rangle$, `NT2`, $\tau > 0$	0.12 ± 0.031		0.01	

PRD 66 (2002) 032003





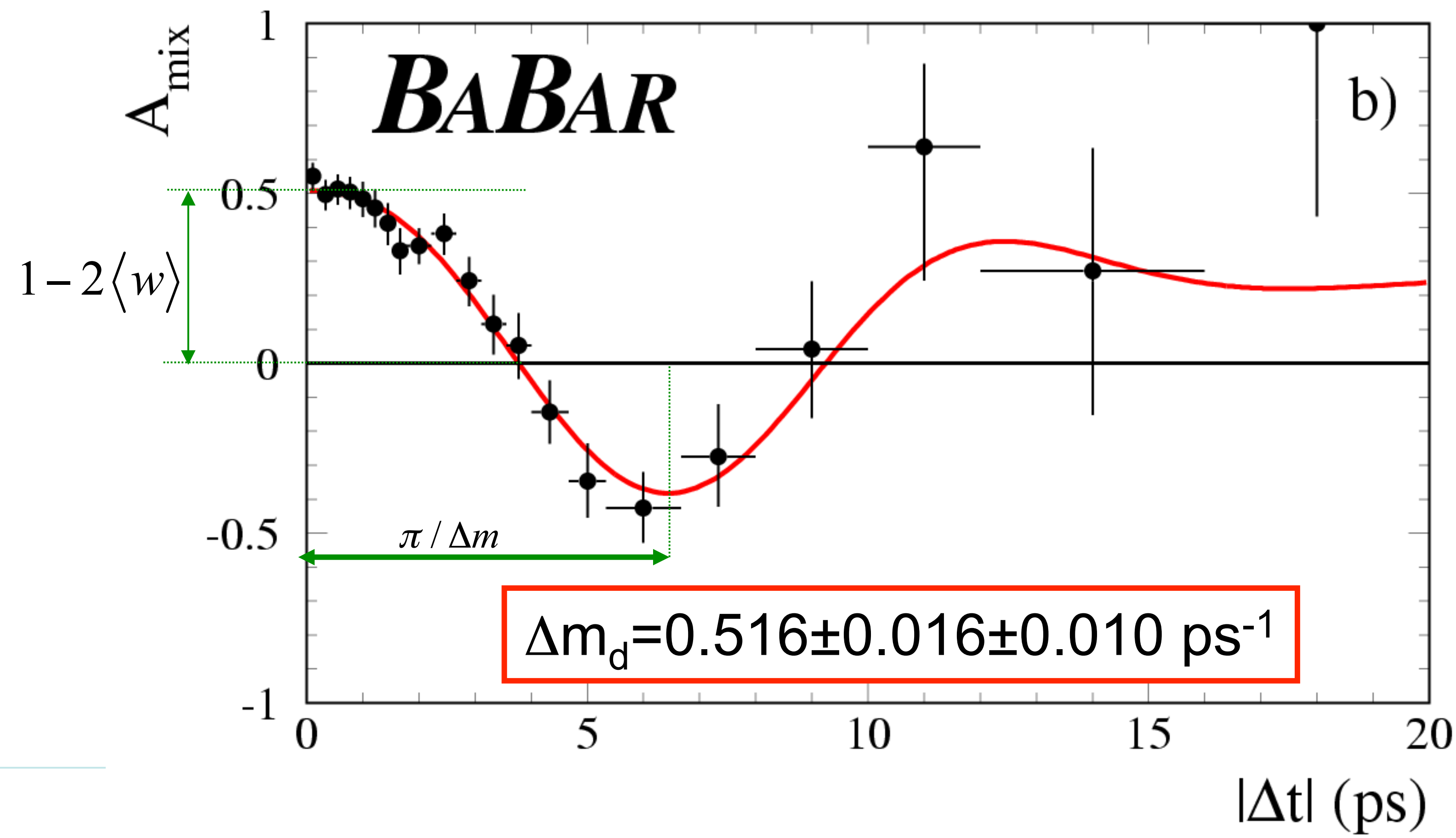
No matter how you get the background parameters, you have to know them anyway. Could equally well first fit sideband only, in a separate fit, and propagate the numbers

But then you get to propagate the statistical errors (+correlations!) on those numbers

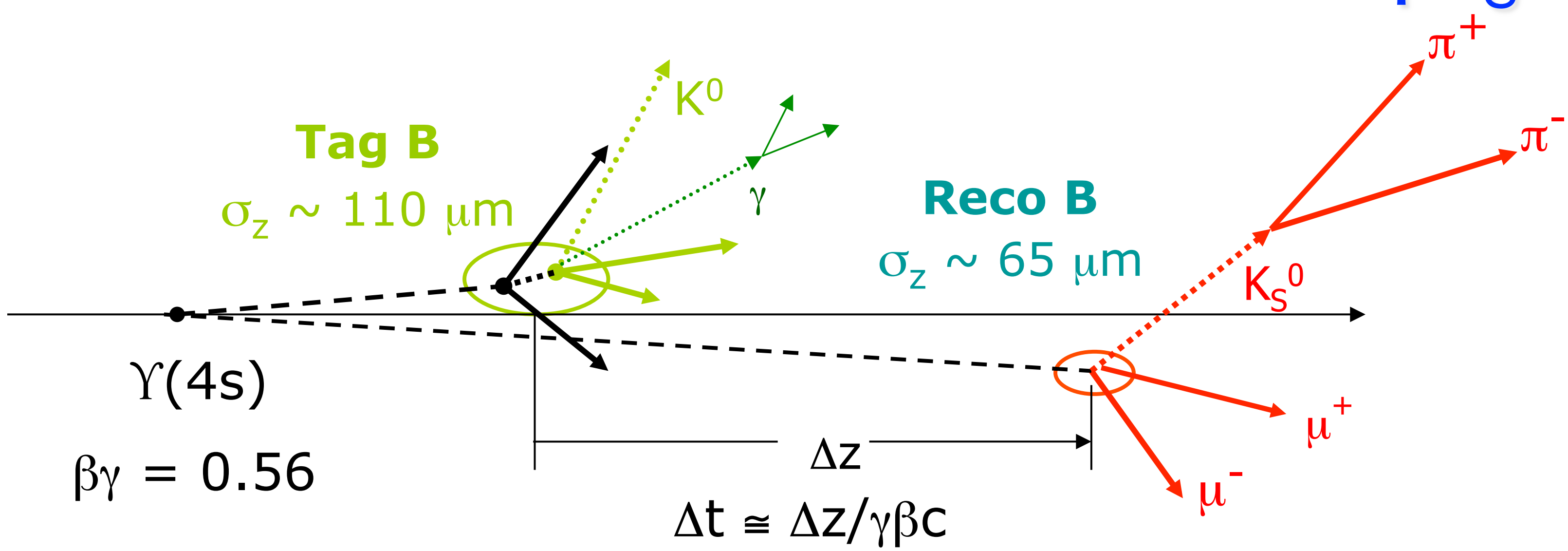
Mixing Likelihood Fit Result

PRD 66 (2002) 032003

$$A_{mix}(\Delta t) = \frac{N_{unmixed}(\Delta t) + N_{mixed}(\Delta t)}{N_{unmixed}(\Delta t) - N_{mixed}(\Delta t)}$$
$$\approx (1 - 2\langle w \rangle)\cos(\Delta m \Delta t)$$



Measurement of CP violation in $B \rightarrow J/\psi K_S$



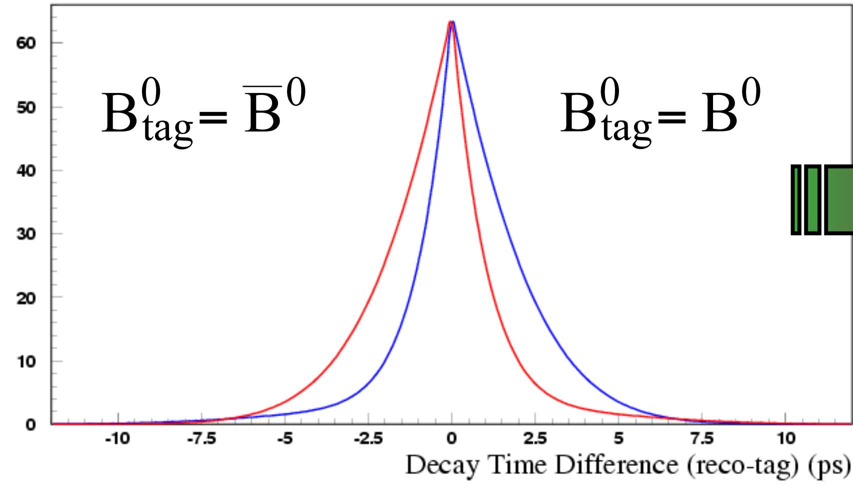
3. Reconstruct Inclusively the vertex of the "other" B meson (B_{TAG}) ✓
4. Determine the flavor of B_{TAG} to separate B^0 and $\bar{B}^0$ ✓

1. Fully reconstruct one B meson in CP eigenstate (B_{REC})
2. Reconstruct the decay vertex √

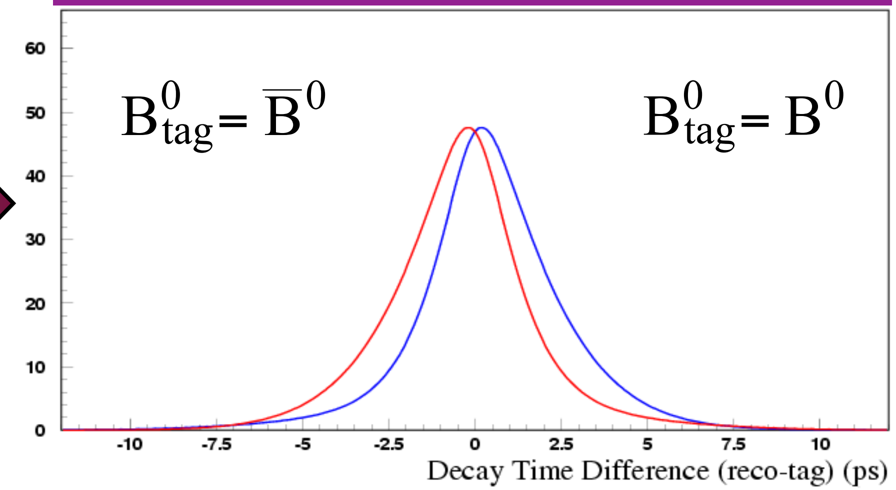
5. compute the proper time difference Δt ✓
6. Fit the Δt spectra of B^0 and $\bar{B}^0$ tagged events

Δt Spectrum of CP events

perfect
flavor tagging & time resolution



realistic
mis-tagging & finite time resolution



CP PDF

$$f_{CP,\pm}(\Delta t) = \left\{ \frac{e^{-|\Delta t|/\tau_{B_d}}}{4\,\tau_{B_d}} \times \left(1 \mp \eta_f \sin 2\beta (1-2w) \sin(\Delta m_d \Delta t)\right) \right\} \otimes R$$

Mistag fractions w
And
resolution function R

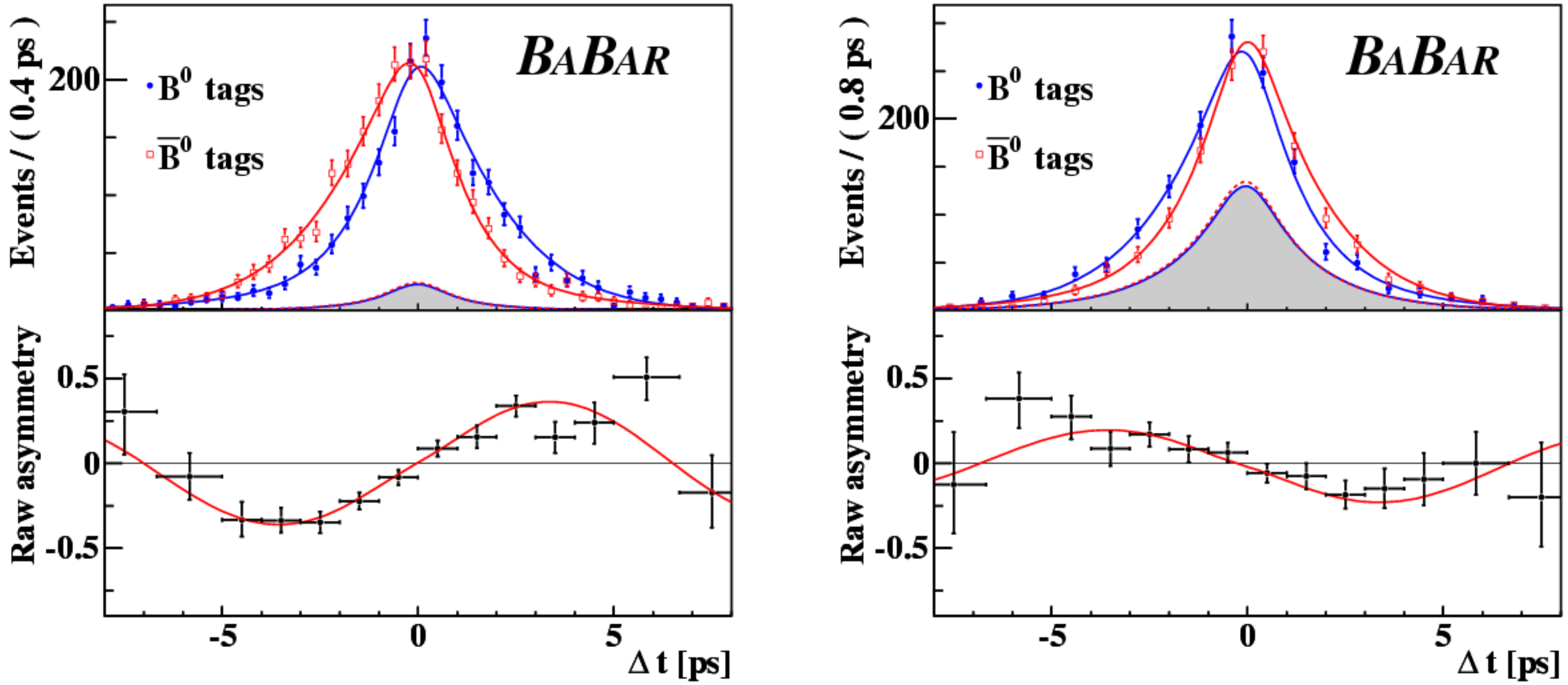
determined by the
flavor sample

Mixing PDF

$$f_{mixing,\pm}(\Delta t) = \left\{ \frac{e^{-|\Delta t|/\tau_{B_d}}}{4\,\tau_{B_d}} \times \left(1 \pm (1-2w) \cos(\Delta m_d \Delta t)\right) \right\} \otimes R$$

$e^{-|\Delta t|/\tau}$

Most recent sin2β Results: 227 BB events



$\sin 2\beta = 0.722 \pm 0.040\ (\text{stat}) \pm 0.023\ (\text{sys})$

- Simultaneous fit to mixing sample and CP sample
- CP sample split in various ways ($J/\psi\ K_S$ vs. $J/\psi\ K_L$, …)
- All signal and background properties extracted from data

CP fit parameters [30/fb, LP 2001]

TABLE XVI: Parameters for the combined likelihood fit to the B_{CP} and B_{flav} samples. The first major column contains the fit results, while the second major column contains the correlation coefficients with respect to sin2β for each fit parameter.

Parameter	Fit Result		Correlation	
	Run 1	Run 2	Run 1	Run 2
sin2β	0.59 ± 0.14			
Signal Resolution Function				
S_1 (core)	1.2 ± 0.1	1.1 ± 0.1	0.018	0.020
$b_1(\Delta t)$ `lepton` (core)	0.07 ± 0.12	0.04 ± 0.16	0.008	0.045
$b_1(\Delta t)$ `kaon` (core)	-0.26 ± 0.08	-0.18 ± 0.09	0.002	0.021
$b_1(\Delta t)$ `NT1` (core)	-0.21 ± 0.15	-0.33 ± 0.21	0.004	0.001
$b_1(\Delta t)$ `NT2` (core)	-0.31 ± 0.11	-0.17 ± 0.15	-0.001	-0.002
$b_2(\Delta t)$ (tail)	-1.7 ± 1.5	-3.3 ± 2.8	0.001	0.006
f_2(tail)	0.08 ± 0.06	0.04 ± 0.04	0.009	0.005
f_3(outlier)	0.005 ± 0.003	0.000 ± 0.001	-0.001	0.000
Signal dilutions				
$\langle D\rangle$, `lepton`	0.82 ± 0.03		-0.042	
$\langle D\rangle$, `kaon`	0.65 ± 0.02		-0.083	
$\langle D\rangle$, `NT1`	0.56 ± 0.04		-0.015	
$\langle D\rangle$, `NT2`	0.30 ± 0.04		-0.032	
ΔD, `lepton`	-0.02 ± 0.04		0.010	
ΔD, `kaon`	0.04 ± 0.03		0.005	
ΔD, `NT1`	-0.11 ± 0.06		0.014	
ΔD, `NT2`	0.12 ± 0.05		-0.008	
Background properties				
τ, mixing bkgd [ps]	1.3 ± 0.1		-0.001	
$f(\tau=0)$, *CP* bkgd	0.60 ± 0.12		-0.011	
$f(\tau=0)$, mixing bkgd, `lepton`	0.31 ± 0.10		-0.001	
$f(\tau=0)$, mixing bkgd, `kaon`	0.65 ± 0.04		-0.001	
$f(\tau=0)$, mixing bkgd, `NT1`	0.62 ± 0.06		-0.001	
$f(\tau=0)$, mixing bkgd, `NT2`	0.64 ± 0.04		-0.001	
Background resolution function				
S_1 (core)	1.5 ± 0.1	1.3 ± 0.1	0.004	-0.003
$b_1(\Delta t)$ core [ps]	-0.16 ± 0.03	0.02 ± 0.04	0.000	-0.001
f_2(outlier)	0.016 ± 0.004	0.017 ± 0.005	-0.001	0.000
Background dilutions				
$\langle D\rangle$, `lepton`, $\tau=0$	0.33 ± 0.27		0.003	
$\langle D\rangle$, `kaon`, $\tau=0$	0.45 ± 0.03		0.008	
$\langle D\rangle$, `NT1`, $\tau=0$	0.25 ± 0.10		0.002	
$\langle D\rangle$, `NT2`, $\tau=0$	0.11 ± 0.06		0.003	
$\langle D\rangle$, `lepton`, $\tau>0$	0.33 ± 0.14		0.000	
$\langle D\rangle$, `kaon`, $\tau>0$	0.24 ± 0.06		0.000	
$\langle D\rangle$, `NT1`, $\tau>0$	0.05 ± 0.14		-0.001	
$\langle D\rangle$, `NT2`, $\tau>0$	0.09 ± 0.09		0.000	

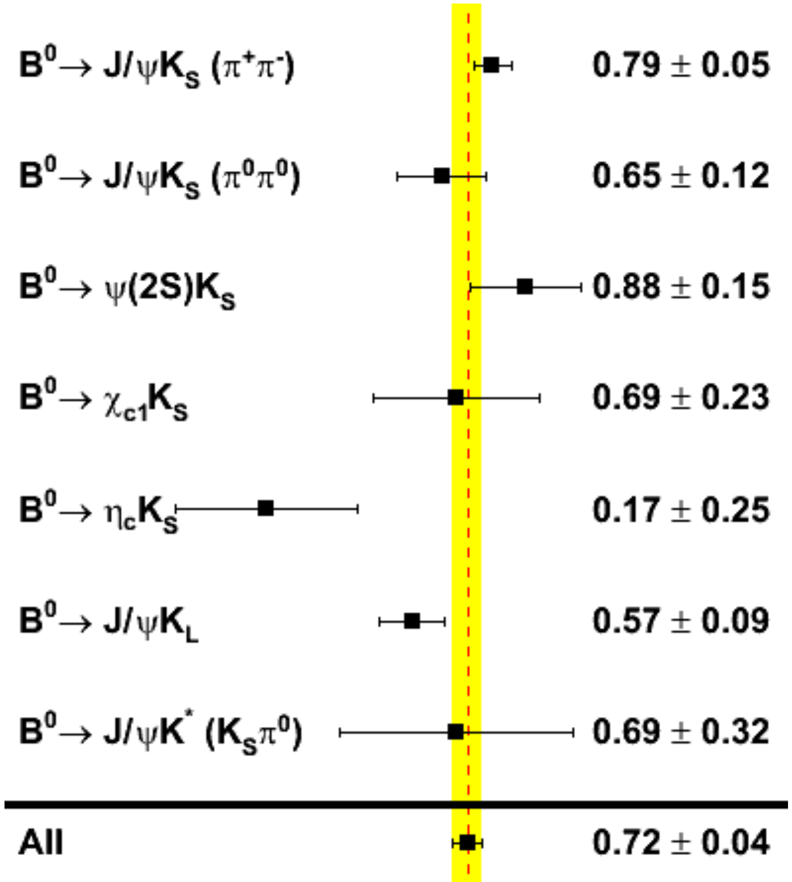
- Compared to mixing fit, add 2 parameters:
  - CP asymmetry sin(2β),
  - prompt background fraction CP events)
- And removes 1 parameter:
  - Δm
- And include some extra events…

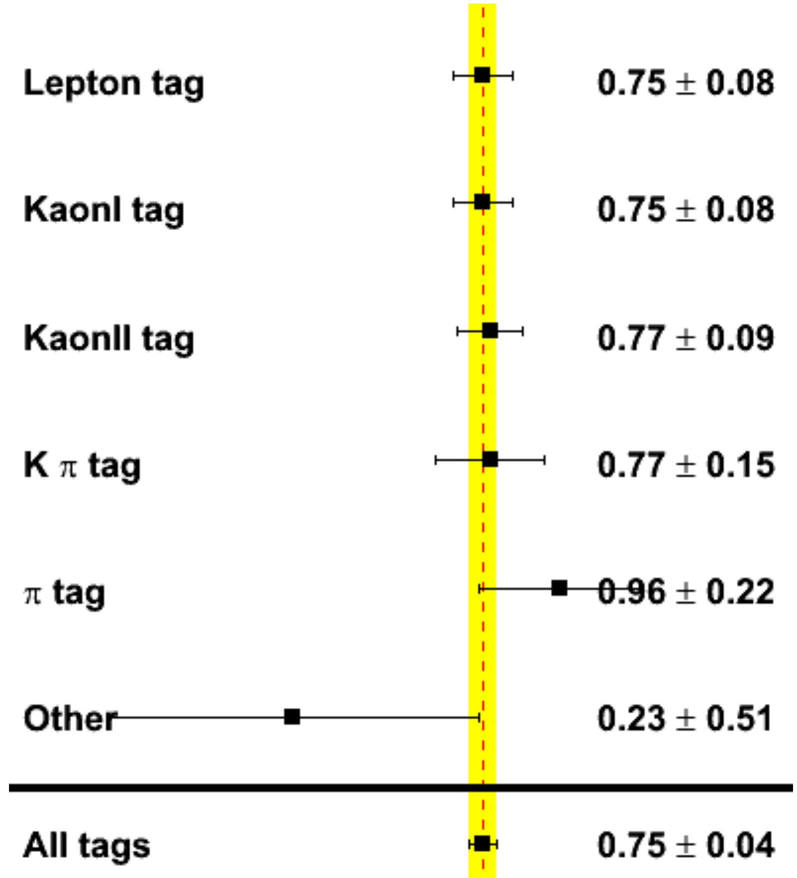
- Total 45 parameters
  - 20 describe background
    - 1 is specific to the CP sample
  - 8 describe signal mistag rates
  - 16 describe the resolution fcn
  - And then of course sin(2b)

- Note:
  - back in 2001 there was a split in run1/run2, which is the cause of doubling the resolution parameters (8+3=11 extra parameters!)

CP fit is basically the mixing fit, with a few more events (which have a slightly different physics PDF), and 2 more parameters…

Consistent results when data is split by decay mode and tagging category





X^2=11.7/6 d.o.f.
Prob (χ^2)=7%

X^2=1.9/5 d.o.f.
Prob (χ^2)=86%

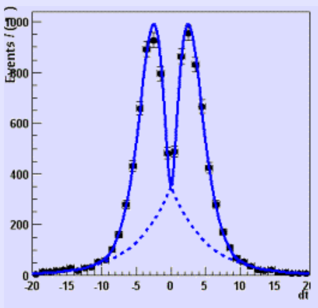
Commercial Break

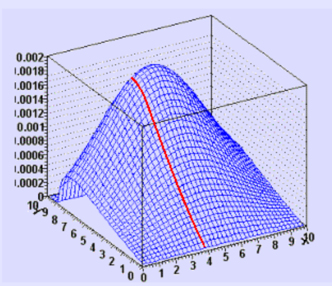
This talk comes with free software that helps you do many labor intensive analysis and fitting tasks much more easily

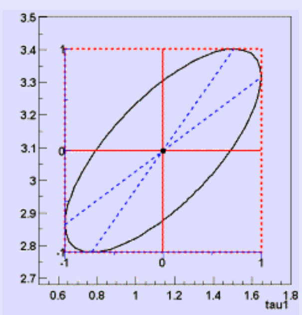
RooFit

A general purpose tool kit for data modeling

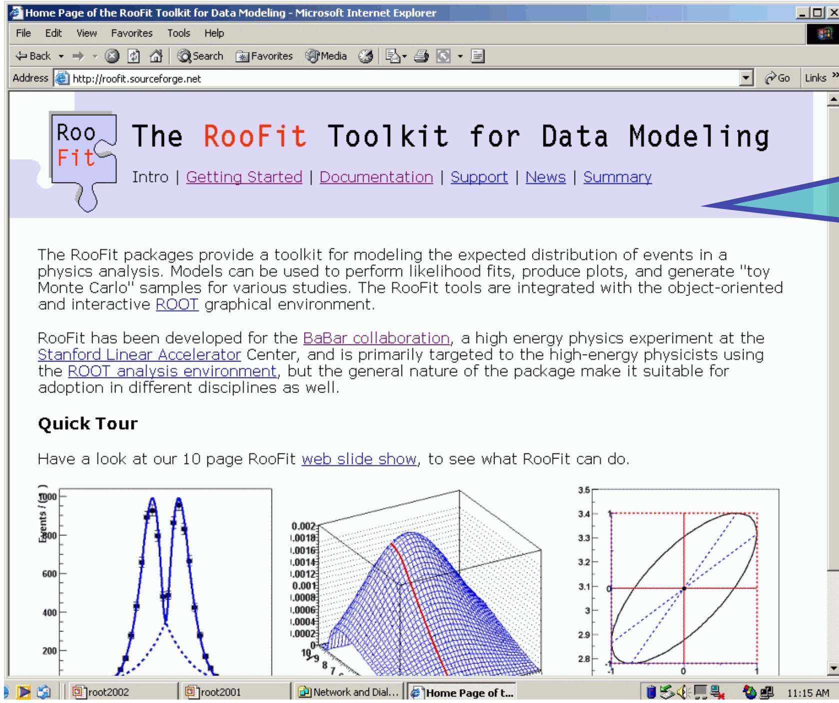
Wouter Verkerke (NIKHEF)
David Kirkby (UC Irvine)



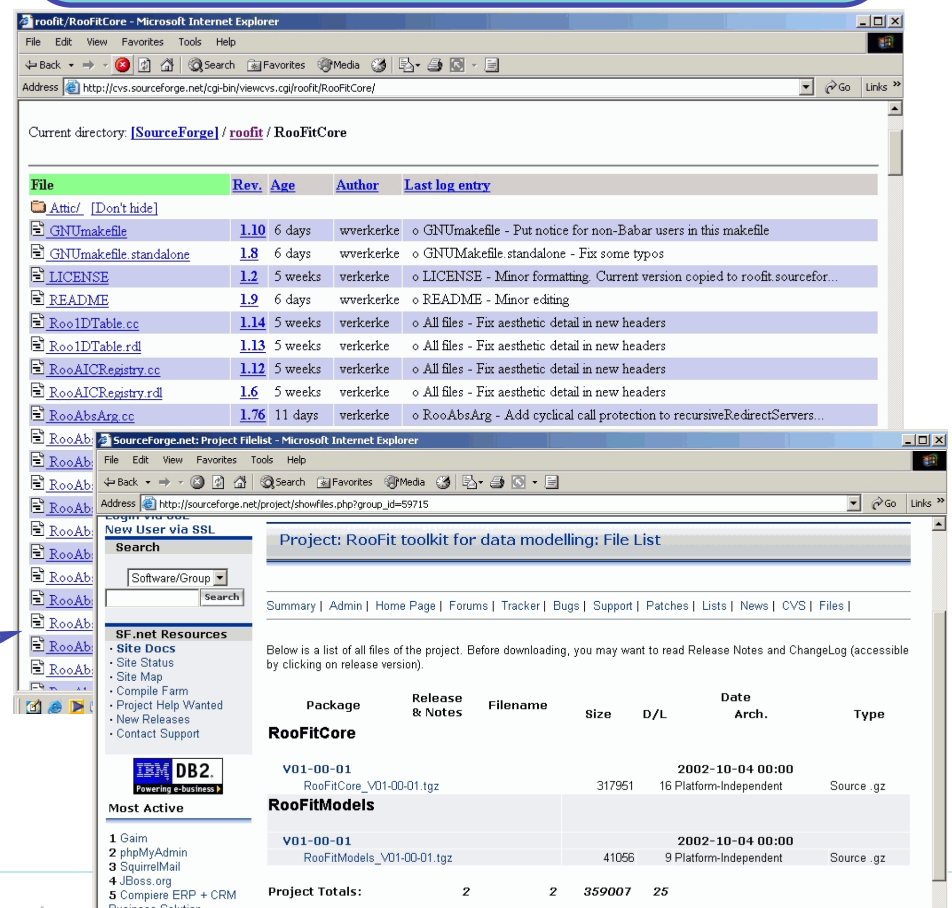




RooFit at SourceForge - `roofit.sourceforge.net`



RooFit available at SourceForge to facilitate access and communication with all users



Code access

- *–CVS repository via pserver*
- *–File distribution sets for production versions*

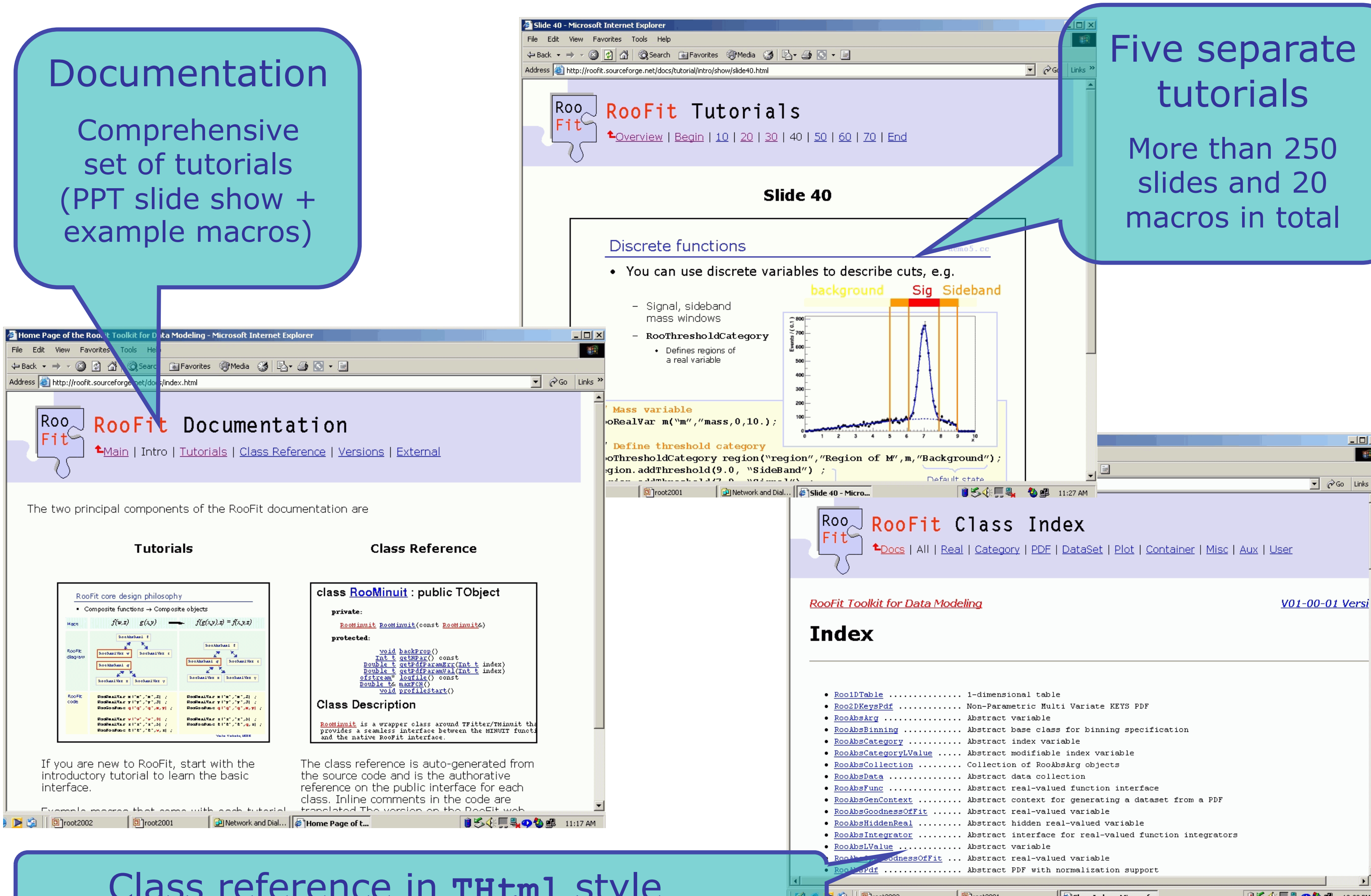
RooFit at SourceForge - Documentation

Documentation

Comprehensive set of tutorials (PPT slide show + example macros)

Five separate tutorials

More than 250 slides and 20 macros in total



Class reference in THtml style

The End

- Some material for further reading
  - http://www.slac.stanford.edu/~verkerke/bnd2004/data_analysis.pdf
  - R. Barlow, *Statistics: A Guide to the Use of Statistical Methods in the Physical Sciences*, Wiley, 1989
  - L. Lyons, *Statistics for Nuclear and Particle Physics*, Cambridge University Press,
  - G. Cowan, *Statistical Data Analysis*, Clarendon, Oxford, 1998
    (See also his 10 hour post-graduate web course: http://www.pp.rhul.ac.uk/~cowan/stat_course)