A Log Likelihood fit for extracting D^0 lifetime

Seonwook Park

December 17, 2012

Abstract

We use a catalog of lifetime data for D^0 mesons to calculate an average value for the lifetime of a D^0 meson. The Go programming language is used to accomplish this task. We first characterise the expected decay relations to a negative log likelihood (NLL) function. We then write a 1-dimensional minimiser for the NLL function using the parabolic method to find the average lifetime to be $0.404 \pm 0.079 ps$. Taking background radiation into account, we calculate a new average by writing a multi-dimensional minimiser utilising the Quasi-Newton method, with the DFP update method. The average lifetime of a D^0 meson, using given data and taking background into account, is $0.410 \pm 0.079 ps$. We also note that background spread contributed a factor of 0.016 to the provided measurements. The final result is within errors of current measurements in Physics, $0.4101 \pm 0.0015 ps$[1].

1 Introduction and Theory

When many D^0 particles decay, the momentum $\vec{p}$, and displacement $\vec{x}$ can be measured. These measurements can be used to calculate the particle decay time t as shown in equation 1.

$$t = m\frac{\vec{x} \cdot \vec{p}}{p^2} \tag{1}$$

It is expected that the decay times are distributed exponentially, with experimental effects resulting in a relationship as shown in equation 2, where $G(\sigma)$ is a Gaussian of width σ.

$$f(t) = \frac{1}{\tau} e^{-\frac{t}{\tau}} \otimes G(\sigma) \tag{2}$$

Equation 2 can be solved to become:

$$f(t) = \frac{1}{2\tau} exp\left(\frac{\sigma^2}{2\tau^2} - \frac{t}{\tau}\right) erfc\left(\frac{1}{\sqrt{2}}\left(\frac{\sigma}{\tau} - \frac{t}{\sigma}\right)\right), \tag{3}$$

where $erfc()$ is the complementary error function. This equation reflects the probability density of a value τ being the lifetime for a measurement of (t, σ). Combining the probabilities together for a set of data with n points, it can be shown that the likelihood of τ is:

$$\mathcal{L}(\tau) = \prod_{i=1}^{i=n} f(\tau, t_i, \sigma_i). \tag{4}$$

1.1 1-D Minimisation

Equation 4 can be shown as a sum by taking a $\log$ of both sides. This *Negative Log Likelihood (NLL)* function is shown in equation 5. The minimum of this function would indicate which τ is most likely, leading to the determination of the average lifetime of D^0.

$$NLL(\tau) = -\sum_{i=1}^{i=n} \log f(\tau, t_i, \sigma_i) \tag{5}$$

The parabolic method can be used for this case, where minimisation is one dimensional. The iterative step required is shown in equation 6, where $A = \frac{y_{n+1} - y_n}{x_{n+1} - x_n}$ and $B = \frac{1}{x_{n+1} - x_n}\left[\frac{y_{n+2} - y_n}{x_{n+2} - x_n} - \frac{y_{n+1} - y_n}{x_{n+1} - x_n}\right]$.

$$x_{n+2} = \frac{1}{2B}\left[\left(x_n + x_{n+1}\right)B - A\right] \tag{6}$$

1.2 2-D Minimisation

As with any experiment, there are background effects in the measurements. This effect can be considered as a Gaussian, as shown in equation 7.

$$f_{bg}\left(t\right) = G(\sigma) = \frac{1}{\sigma\sqrt{2\pi}} exp\left(-\frac{1}{2}\frac{t^2}{\sigma^2}\right) \tag{7}$$

The probability density function in equation 3 must reflect the effects of equation 7. However, the proportion of both effects are unknown and therefore a new variable must be introduced, resulting in equation 8.

$$f_{total}\left(t\right) = af\left(t\right) + (1-a)f_{bg}\left(t\right) \tag{8}$$

To find the minimum of this function, a multi-dimensional minimiser must be used, such as the Quasi-Newton DFP method shown in equation 9, where $\mathbf{G}_n$ is an inverse Hessian updated using equation 10 ($\vec{\delta}_n = \vec{x}_{n+1} - \vec{x}_n$, $\vec{\gamma}_n = \vec{\bigtriangledown} f\left(\vec{x}_{n+1}\right) - \vec{\bigtriangledown} f\left(\vec{x}_n\right)$).

$$\vec{x}_{n+1} = \vec{x}_n - \alpha_n \mathbf{G}_n \vec{\bigtriangledown} f\left(\vec{x}_n\right) \tag{9}$$

$$\mathbf{G}_{n+1} = \mathbf{G}_n + \frac{\vec{\delta}_n \otimes \vec{\delta}_n}{\vec{\gamma}_n^T \vec{\delta}_n} - \frac{\mathbf{G}_n \vec{\gamma}_n \vec{\gamma}_n^T \mathbf{G}_n^T}{\vec{\gamma}_n^T \mathbf{G}_n \vec{\gamma}_n} \tag{10}$$

α_n is the step length, determined using the two Wolfe conditions, shown in equation 11 where $p_n = -\mathbf{G}_n \vec{\bigtriangledown} f\left(\vec{x}_n\right)$.

$$\begin{aligned} f\left(x_n + \alpha_n p_n\right) &\leqslant f\left(x_n\right) + 10^{-4}\alpha_n p_n^T \vec{\bigtriangledown} f\left(\vec{x}_n\right) \\ p_n^T \vec{\bigtriangledown} f\left(\vec{x}_n + \alpha_n p_n\right) &\geqslant 0.9 p_n^T \vec{\bigtriangledown} f\left(\vec{x}_n\right) \end{aligned} \tag{11}$$

2 Method

All programming is done with the Go programming language. To characterise the problem, a histogram of data points was first plotted using the plotinum plotting library[1]. This was done with varying number of bins to better understand the given data.

2.1 1-D NLL Minimisation

The fit function stated in equation 3 was coded. Unfortunately, due to the lack of a Go language implementation of GSL, parts of GSL related to the $erfc$ function had to be ported[2]. The input variables, τ and σ were varied to aid in understanding the effects of such variations.

The NLL function in equation 5 was coded, and used in a parabolic minimiser which was programmed with steps laid out in equation 6. A good initial estimate was made by inspecting a graph of the NLL function over varying τ. The minimisation is considered to be converged when $\Delta\tau_n < 1 \times 10^{-7}$. This was determined by inspecting calculated values.

[1] plotinum - A plotting library for the Go programming language (http://code.google.com/p/plotinum)

[2] Not included in submitted code. Ported code can be found at https://github.com/swook/gogsl. Features Go native error handling.

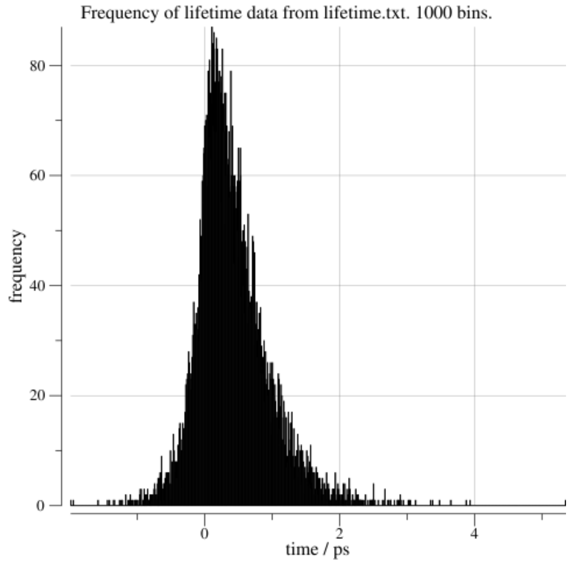


Figure 1: Histogram of lifetimes from provided data.

2.2 Accuracy of calculated minimum value

To find the error of the calculated minimum value, a method was written to find τ^+ and τ^- values for when $NLL(\tau)$ is 0.5 greater than the minimum NLL value. This is because $\tau^+ - \tau^-$ corresponds to one standard deviation. This required the usage of the Newton-Raphson method (eq 12) for finding roots, which uses the central difference scheme (CDS) to determine the gradient $f'(x_n)$ at different points. Convergence was determined to be when $\Delta x_n < 1 \times 10^{-12}$, by inspection.

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{12}$$

The number of samples used for parabolic minimisation can be varied to observe effects on the calculated minimum value and the according standard deviations. Plotting this variation allows for the determination of a rough relation between sample size and uncertainty, and thus allows for estimations of required sample sizes for desired levels of uncertainty.

2.3 2-D NLL Minimisation

Background effects are taken into account by attempting to minimise equation 8. The Quasi-Newton DFP method is used as outlined in section 1.2. As this requires numerous matrix operations, a matrix library was created in the go language with basic operations such a matrix initialisation, multiplication, and addition.[3].

As equation 10 requires the calculation of gradients, the central difference scheme is used again to approximate gradients. The initial estimate solutions are similar to those in section 2.1. This is because background effects are expected to be minimal. The Quasi-Newton minimiser is determined to have converged when the gradient no longer changes, and thus $\vec{\bigtriangledown} f(\vec{x}_{n+1}) = \bigtriangledown f(\vec{x}_n)$.

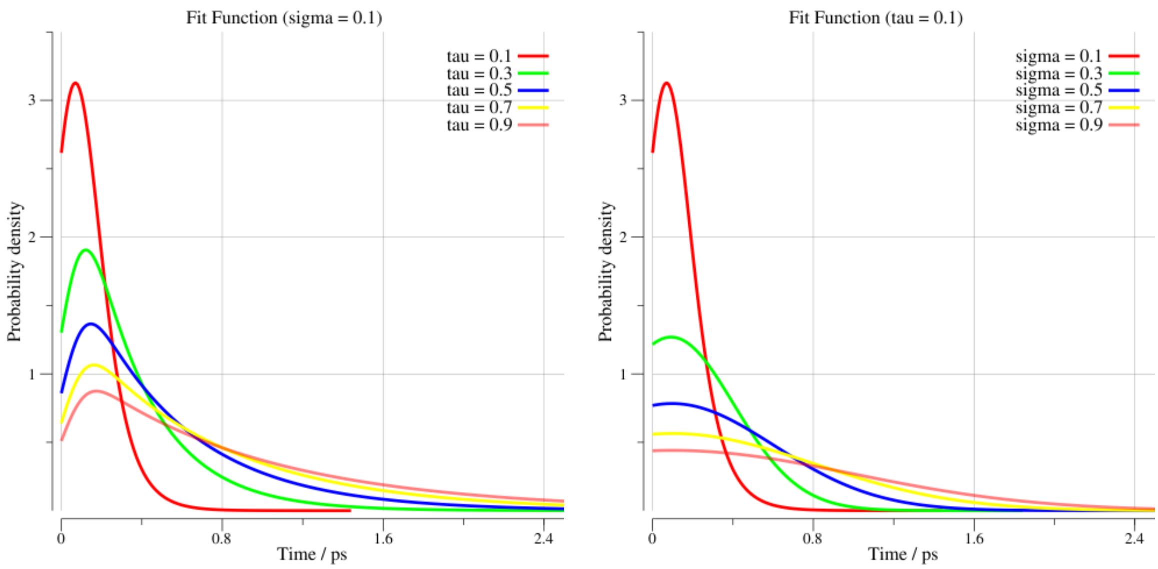
For every iteration of this minimisation, α_n must be re-determined using the Wolfe conditions in equation 11. This can be done via a backtracking line search where a trial α is reduced by a factor $0 < k < 1$ until conditions are met. In this project, a k value of 0.5 was used.

3 Results and Discussion

3.1 Data inspection

On inspecting a histogram of the provided data (figure 1), it can be seen that the distribution could be modelled as a form of a Gaussian. This explains the convolution with a Gaussian that is performed in equation 2 when forming

[3]Not included in submitted code but can be found at https://github.com/swook/go.iccp. Features parsing of matrix literals.



(a) $\sigma = 0.1$,with varying τ (b) $\tau = 0.1$,with varying σ

Figure 2: Varying σ and τ in turn to see effects on fit function from equation 3

a model for these measurements.

Figure 2 shows the effects of varying σ or τ on the fit function defined in equation 3. It can be seen that with varying τ, the maximum point shifts position, while with varying σ, this happens less, and instead the peaks spread out with higher σ. Comparing the graphs with figure 1 allows for a careful guess that average lifetime τ would be approximately 0.5. This is because the fit function most closely resembles the histogram plot with the mentioned τ value.

3.2 1-D Minimisation

Inspecting a graph of the NLL function (figure 3) from equation 5, it can be said that the minimum should be approximately 0.4. With the estimate, the parabolic minimiser was run, yielding a minimum NLL value of 6220.4 and τ of $0.404ps$. As mentioned in section 2.2, the Newton-Raphson method is used to find the 1σ width for τ. It was calculated to be $0.079ps$, resulting in a average lifetime estimate for D^0 at $0.404 \pm 0.079ps$.

The current accepted measurement of D^0 lifetime, as mentioned in the Particle Data Group's article, Review of particle physics‘[1], is $0.4101 \pm 0.0015ps$. Though the calculated errors are large and therefore have the accepted measurement within range, this accepted value is higher than our current estimate when disregarding standard deviation. It was noted though, that background has not yet been taken into account, meaning that the calculated mean τ will change in value.

The 1σ width was found for varying sample sizes. This was done by repeating section 2.1 with different number of data points being used. Figure 4 shows a linear relationship between the number of samples used, and the uncertainty for the corresponding minimum calculated using the parabolic method. A linear fit was performed, yielding a fit equation of $y = -5 \times 10^{-7}x + 0.083$. For a 1σ width of $10^{-15}s$, an approximate required sample size can be calculated from the linear fit equation to yield 1.66×10^5 samples. This value is 16.6 times the number of samples which were provided for this experiment.

3.3 2-D Minimisation

The Quasi-Newton DFP minimiser implemented as outlined in section 2.3 was run using initial $\tau = 0.4$ and $a = 0.1$. The calculated result is $\tau = 0.410$, $a = 0.984$, and corresponding $NLL = 6218.4$.

The error determination method carried out in 2.2 was used to calculate a 1σ width for the calculated results. This calculated standard deviation is 0.0792, yielding an average D^0 lifetime calculation of $0.410 \pm 0.079ps$, which is well within the errors of the currently accepted value of $0.4101 \pm 0.0015ps$. This shows that the additional consideration of the background has yielded a more accurate lifetime calculation, proving that the models used are

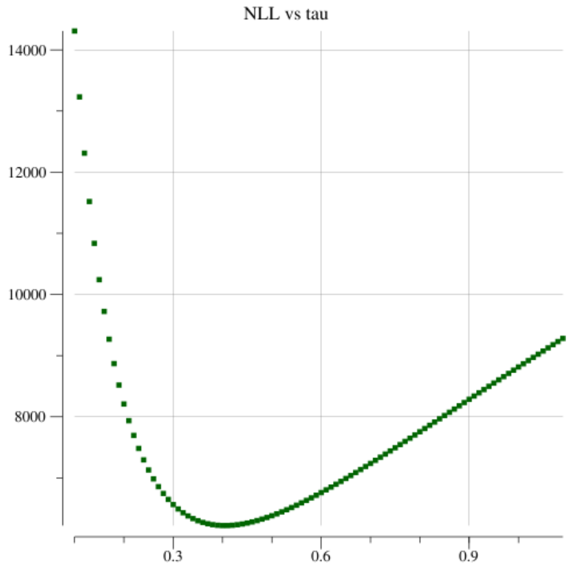


Figure 3: NLL function against average lifetime, τ

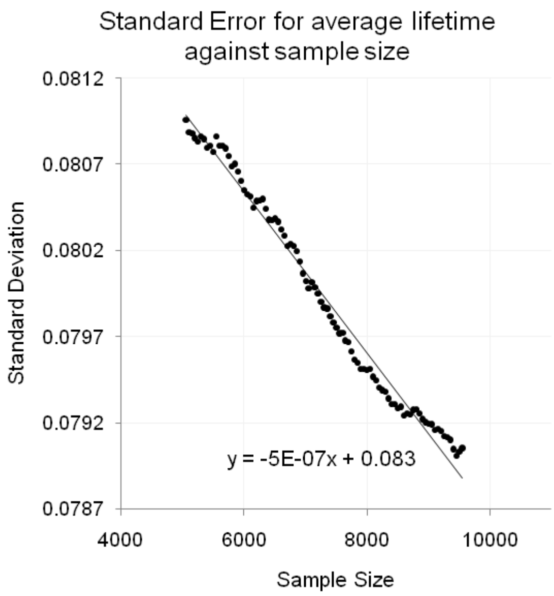


Figure 4: 1σ width against sample size

appropriate. It is also determined, from the value of a, that the proportion of background influences in the provided data is 0.016, or 1.6% of the data.

4 Conclusion

An average lifetime calculation was attempted from a set of given lifetime data. A one-dimensional fit was first attempted, using a negative log likelihood function to model the lifetimes of D^0 mesons, with the parabolic method. Initial estimates were made by plotting the NLL function beforehand. A standard deviation for the calculated mean lifetime was found by seeking τ values when NLL is 0.5 greater than its value at the minimum. This yielded a mean τ value of $0.404 \pm 0.079ps$.

An improvement was made to the model by accounting for background. This was done by adding an unknown factor of a Gaussian to the fit function. Due to the added unknown variable, a multi-dimensional minimiser had to be implemented. The Quasi-Newton DFP method was used, using a custom matrix library, and central difference scheme methods. This calculation yielded a mean τ value of $0.410 \pm 0.079ps$. This value is closer to the accepted value of $0.4101 \pm 0.0015ps$, though when considering errors, both values (background-considered, and not-considered) are within range.

An improvement to this problem, would be an increase in the number of data points used. It was found that with an increase of sample size, there is a linear decrease in error for mean τ. For an error of 10^{-15}, approximately 1.66×10^5 would be required. This would mean that not only our calculated mean lifetimes will become more accurate, but the error margin will become smaller, allowing for higher precision.

References

[1] J. Beringer et al. (Particle Data Group). Review of particle physics. *Phys. Rev. D*, 86:010001, Jul 2012.

Approximate word count: 1590 words