A log-likelihood fit for extracting neutrino oscillation parameters

Anson Wing Long Poon, 21st December 2020

***Abstract*— In this project, the neutrino mixing phenomenon and the concept of negative log-likelihood were studied. Four multidimensional minimising techniques, including univariate, gradient, Newton and Quasi-Newton method, were investigated and were used to minimise the negative log-likelihood function to obtain the best estimates of the mixing angle, squared mas difference and neutrino cross-section. In the first part of the experiment, the cross-section was neglected and the best estimates of the mixing angle and the squared mass difference were determined to be 0.726 ± 0.0014 and 0.0019646 ± 0.0000332 eV^2. Then the cross-section was taken into account and the best estimates of the mixing angle, the squared mass difference and the neutrino cross-section were determined to be 0.68438 ± 0.01593, 0.0020442 ± 0.0000374 eV^2 and 1.2928 ± 0.05014.**

I. *Introduction.*

Neurtinos are one of the fundamental particles in the Standard model [1], with some strange properties. Due to their peculiar properties, neutrinos have been referred to as ghost particles [2] and our understanding of neutrinos has changed over time. For instance, they have long been assumed to be massless according to the Standard model but in recent years physicists rejected this idea due to neutrino mixing [1]. There are also large scale ongoing experiments on neutrinos, including the Deep Underground Neutrino Experiment at Fermilab [3]and the T2K experiment in Japan [1]. Many also believe that neutrinos hold the key to understand many physical phenomena in the universe, from the Big Bang to the nuclear fusion reaction of the Sun[2]. Due to their weakly interactive nature, it has been difficult for scientists to collect data on neutrinos [1]. The aim of the experiment is to study the experimental neutrino data and to estimate the key parameters of neutrinos by comparing the experimental neutrino flux to the theoretically expected neutrino flux through a negative log-likelihood fit.

II. Theory

A. *Neutrino mixing*

There are three flavours of neutrinos: electron neutrino, muon neutrino and tau neutrino. Neutrino mixing refers to the phenomenon that neutrinos can oscillate between these three flavour eigenstates over time [4]. Although in the real world, neutrinos oscillate between all three flavours, in this project we are investigating only the oscillation between the muon and tau neutrinos. The probability of a muon neutrino not decaying into a tau neutrino is known as the 'survival probability' [1],

$$P(\nu_\mu \to \nu_\mu) = 1 - \sin^2(2\theta_{23}) \sin^2\left(\frac{1.267\Delta m^2_{23} L}{E}\right) \quad (1)$$

where θ_{23} is an arbitrary parameter called the mixing angle, Δm^2_{23} (eV) is the squared mass difference between the tau and muon neutrinos, L is the distance the neutrinos travel (km) before detection and E is the energy of neutrinos.

B. *Negative log-likelihood fit*

To estimate the values of the mixing angle and the square mass difference, the negative log-likelihood fit was used. The likelihood function is defined as,

$$\mathcal{L} = \prod_{i=1}^{n} \mathcal{P}(\mathbf{u}; m_i) \quad (2)$$

where $\mathbf{u} = (\theta_{23},\ \Delta m^2_{23})$, m_i is the observed number of events in energy bin i and $\mathcal{P}(\mathbf{u}; m_i)$ is the probability of the measurement m.

The likelihood there corresponds to the probability of observing the given experimental data, for an individual set of parameters[5]. The values of **u**, that maximises the likelihood function, are the best estimates of u. Since a natural log function is a single-valued function, the maximum of the location of the log-likelihood is the same as the likelihood. Moreover, to simplify the mathematical calculating we use the negative log-likelihood instead of the likelihood. Since the natural log function is single-valued and increasing [5], the location of the minimum negative log-likelihood is the same as that of the maximum likelihood [1],

$$\begin{aligned} \mathrm{NLL}(\mathbf{u}) &= -\ln\left(\prod_{i=1}^{n} \mathcal{P}(\mathbf{u}; m_i)\right) \\ &= -\sum_{i=1}^{n} \ln\left(\mathcal{P}(\mathbf{u}; m_i)\right) \end{aligned} \quad (3)$$

Since neutrinos are weakly interacting with only O(100) observed events and that the neutrino events are independent and have a fixed probability over time, the number of events is expected to follow a Poisson distribution [1],

$$\mathcal{P}(\mathbf{u}; m_i) = \frac{\lambda^m e^{-\lambda}}{m!} \quad (4)$$

Hence, by substituting eq(4) int eq(3) and using Stirling's approximation,

$$\mathrm{NLL}(\mathbf{u}) = -\sum_{i=1}^{n} \lambda_i(\mathbf{u}) - m_i + m_i \ln\left(\frac{m_i}{\lambda_i(\mathbf{u})}\right) \quad (5)$$

where lambda is the oscillated event rate prediction.
Therefore, the **u** that corresponds to the minimum of eq(5) is the best estimates of θ_{23} and Δm^2_{23}.

However, the neutrino cross-sections can affect the oscillated event rate prediction and the effect can be approximated as follow [1],

$$\lambda_i^{\text{new}}(\mathbf{u}) = \lambda_i^{\text{old}}(\mathbf{u}) \cdot \alpha \cdot E \quad (6)$$

where α is the neutrino cross-section and E is the energy of the neutrino.

In the project, we first minimised the negative log-likelihood function with respect to the mixing angle and squared mass difference, neglecting the effect of the neutrino cross-section. Then at the end, we revisited the minimisation problem with the cross-section.

III. Methods

First, the position of the minimum was estimated by minimising the mixing angle, while fixing Δm^2_{23} = 0.0024, using the one dimensional parabolic minimiser, from the periodic nature of the negative log-likelihood function was also observed. A contour plot was then created to get a better estimate of the minimum for both the mixing angle and the squared mass difference (see figure 1.1). Using the contour plot, the initial point was chosen to pass into four minimising algorithms: univariate method, gradient method, Quasi-Newton method and Newton method, to determine the minimum. Finally, we used the same four algorithms to determine the minimum for the three dimensional case, taking into account the cross-section as well as the mixing angle and squared mass difference. Moreover, before applying these minimisers to the negative log-likelihood, they were validated with analytical functions with known minima.

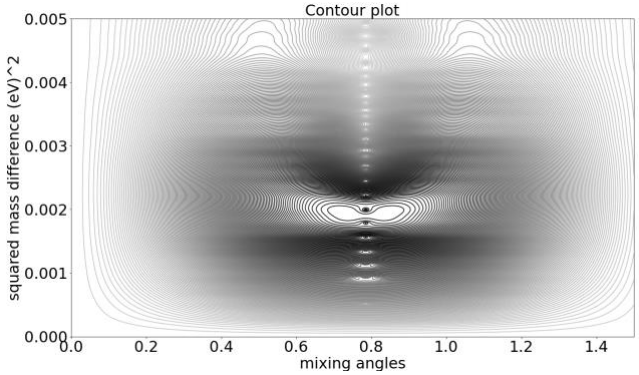

Figure 1.1. A contour plot showing the approximate locations of the minima.

From figure 1.1, it is clear that the first minimum occurs between 0.6 and 0.8 for the mixing angle at about 0.002 for the squared mass difference. Therefore the initial point of all the minimisers were taken as (0.6, 0.0024).

A. Parabolic minimiser

In the parabolic method, three guesses, x_0, x_1 and x_2, close to the minimum were passed into the algorithm, which must satisfy the condition: f(x_0) > f(x_1) < f(x_2). A parabola was then fitted through these three points using Lagrange second order polynomials, from which the minimum, x_3 can be calculated [6],

$$x_3 = \frac{(x_2^2 - x_1^2)f_0 + (x_0^2 - x_2^2)f_1 + (x_1^2 - x_0^2)f_2}{2[(x_2 - x_1)f_0 + (x_0 - x_2)f_1 + (x_1 - x_0)f_{2]}} \quad (7)$$

Then the procedure was repeated with the three x values that correspond to the three lowest f values. x_3 would therefore converge towards the minimum after each iteration.

B. Univariate method

The univariate method is the same as the parabolic minimiser but in multiple dimensions. The univariate method minimised one variable after another and then this procedure was repeated until it reached the minimum point.

C. Gradient method

First, an initial point, $\vec{x}_0$, was passed into the algorithm, from which the gradient vector, was determined by forward difference approximation. Then we found the next point by moving this initial point against the direction of the gradient vector by a small increment [6],

$$\vec{x}_{n+1} = \vec{x}_n - \alpha\vec{\nabla} f(\vec{x}_n) \quad (8)$$

where the gradient vector was approximated by the forward difference scheme so that the gradient vector was calculated most efficiently [6],

$$\frac{\partial f}{\partial x_i} \approx \frac{f(x_1, \dots, x_i + h, \dots, x_N) - f(x_1, \dots, x_i, \dots, x_N)}{h} \quad (9)$$

where h<<1.

Since the gradient vector corresponds to the direction of the steep increase in function value [8]. By iterating, the point would eventually converge to the minimum.

For the negative log-likelihood function, since the squared mass difference is O(0.01) smaller than the mixing angle. Therefore, to make the gradient method converge faster and more evenly, the squared mass difference was scaled by the ratio between the two.

D. Newton method

The Newton method is derived from the Taylor expansion at the point up to second order by including the curvature which is represented by the Hessian matrix [6],

$$H_{ij}(\vec{x}) = \frac{\partial^2 f(\vec{x})}{\partial x_i \, \partial x_j} \quad (10)$$

where delta is the vector that points from the current point to the minimum. By acting the gradient operator to the Taylor expansion and rearranging, the iterative equation can be derived [6],

$$\vec{x}_{n+1} = \vec{x}_n - \mathbf{H}_n{}^{-1} \cdot \vec{\nabla} f(\vec{x}_n) \quad (11)$$

Therefore, by iterating eq(11.18), the point would eventually converge towards the minimum.

E. *Quasi-Newton method*

The Quasi-Newton method uses the same concept as the Newton method, where the inverse Hessian is approximated instead of calculated by inversing it [6],

$$\vec{x}_{n+1} = \vec{x}_n - \alpha \mathbf{G}_n \cdot \vec{\nabla} f(\vec{x}_n) \tag{12}$$

This is believed to save computational power for inversing a matrix. The G matrix can be updated using the Davidon-Fletcher-Powell algorithm [6],

$$\mathbf{G}_{n+1} = \mathbf{G}_n + \frac{(\vec{\delta}_n \otimes \vec{\delta}_n)}{\vec{\gamma}_n \cdot \vec{\delta}_n} - \frac{\mathbf{G}_n \cdot (\vec{\gamma}_n \otimes \vec{\gamma}_n) \cdot \mathbf{G}_n}{\vec{\gamma}_n \cdot \mathbf{G}_n \cdot \vec{\gamma}_n} \tag{13}$$

where $\mathbf{G}_0$ was taken to be the identity matrix.
Similar to the gradient method, the squared mass difference was also scaled by the ratio between the two to ensure a fast and even convergence.

F. *Step sizes*

It is really important to choose the right step size for the gradient and Quasi-Newton method. If the step size is too small, it would overshoot and miss the minimum, and as a result it would not converge. However, if the step size is too small, both methods would take too long to converge.

Therefore, the values of the negative log likelihood in the first 100 iterations were determined for step sizes of 10^{-3}, 10^{-4}, 10^{-5} and 10^{-6} (see figure 1.2 and 1.3).

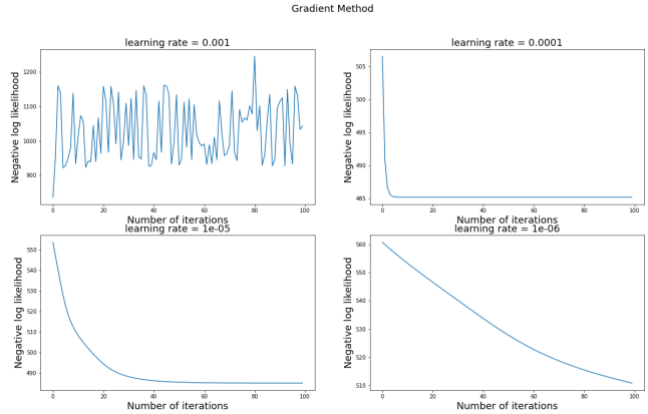

Figure 1.2. Comparison of step sizes. These plots show if the gradient method converges with a certain step size and how fast it converges.

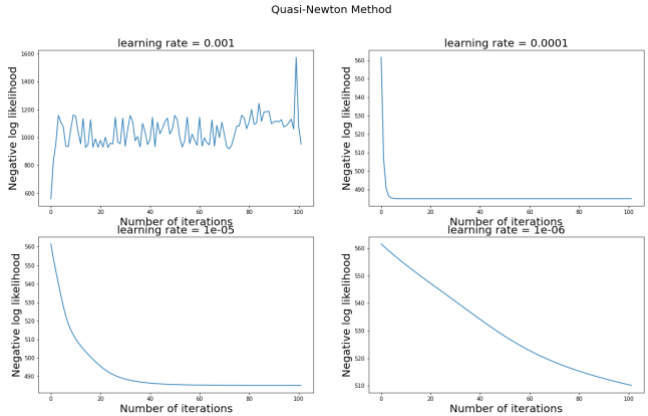

Figure 1.3. Similar to figure 1.2. These plots show if the Quasi-Newton method converges with a certain step size and how fast it converges.

From figure 1.2 and 1.3, it is clear that 0.001 is too big while 10^{-6} is too small. Therefore the step size was eventually chosen as 0.0001 to ensure fast convergence.

IV. Results

A. *One-dimensional minimiser*

In the one-dimensional case, the mixing angle was minimised with a fixed $\Delta m^2_{23} = 0.0024$. This mixing angle was determined to be 0.64354 ± 0.01085. This error was estimated by the curvature method [5],

$$\sigma_{\hat{a}} \approx \frac{1}{\sqrt{d^2 NLL / da^2 \mid \hat{a}}} \tag{14}$$

where $\sigma_{\hat{a}}$ is the standard error for the estimated parameter $\hat{a}$.

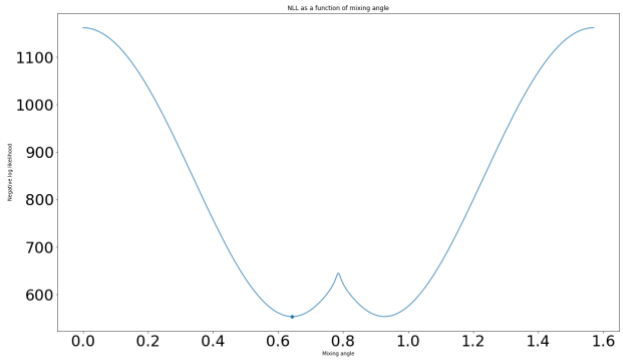

Figure 1.4. A plot of negative log likelihood against theta with a fixed square mass difference.

The curvature method is based on a Gaussian approximation near the minimum. The error estimation is therefore symmetric. As long as the minimum is analytical and smooth, the curvature method would give a good approximation to the errors. However, errors can also be calculated by find the points that correspond to NLL ± 0.5, and were determined to be (0.01084, -0.01090). However, both error methods would become inaccurate as it approaches $\pi/4$ which is the local maximum (as seen from figure 1.4), sine the Guassian approximation would become inaccurate as it leaves the minimum and the value method can become very asymmetric as it approaches from one side.

B. *Multidimensional minimiser*

The minimisers gave very similar values for the estimation of the parameters (see figure 1.4 and figure 1.5).

	Univariate	Gradient	Quasi-Newton	Newton
θ_{23}	0.72532	0.72562	0.72563	0.72636
Δm^2_{23} (eV)^2	0.0019648	0.0019646	0.0019645	0.0019589
NLL(**u**)	485.18	485.18	485.18	485.19
Time taken (s)	0.95237	0.16792	0.16481	0.20715
Number of iterations	24	23	22	8

Figure 1.5. A table showing the value of the minimum, the estimated values for the parameters, time taken and number of iterations of the two dimensional minimisers.

From figure 1.5, it is clear that the gradient and Quasi-Newton method are the fastest, although with most number of iterations. This is because univariate function is inefficient in the way that it only minimises one variable at once, meaning the path it takes is longer than the other minimisers which also means each iteration would also take longer. Newton method involved the least number of iterations but took slightly more time than the gradient and Quasi-Newton

method. This is because the Hessian matrix involves calculating the second derivatives which would have consumed mor computational power compared to just the first derivatives as in other minimisers. Moreover, the Newton requires matrix inversion, which is not part of any other algorithms. Moreover, the results from all four algorithms agree one another up to three significant figures, except the mixing angle estimate from the univariate method. This is believed to be due to the fact that the minimisation was not simultaneous with the two variables, making one variable less accurate than the other. Moreover the implemented univariate method minimised the squared mass difference the last, which means it was possible that the final approximated minimum point was moved away from the minimum in the mixing angle direction, giving the inaccuracy in the mixing angle.

	Univariate	Gradient	Quasi-Newton	Newton
θ_{23}	0.68437	0.68438	0.68437	0.68527
Δm^2_{23} (eV)^2	0.0020443	0.0020442	0.0020442	0.0020378
α	1.2925	1.2928	1.2927	1.2904
NLL($\mathbf{u}$)	102.50	102.50	102.50	102.51
Time taken (s)	1.4027	2.7230	2.3112	0.31662
Number of iterations	20	233	185	6

Figure 1.6 A table showing the value of the minimum, the estimated values for the parameters, time taken and number of iterations of the three dimensional minimisers.

From figure 1.6, the speed of the algorithms are opposite of the two dimensional case. The univariate and Newton methods were faster than the gradient and Quasi-Newton method, which was not expected. This maybe because with the effects of cross-section, the minimum of negative log likelihood function can be better approximated as a parabola. Since the univariate method uses parabolic approximation while Quasi-Newton method takes the curvature into account, these methods can outperform the gradient and Quasi-Newton method. Moreover, the Newton method seems to disagree with the other methods in three dimensions, this maybe due to the bigger steps the Newton method takes, meaning it might have overshoot the minimum.

The errors for the estimated were approximated by the curvature of the NLL at the minimum using eq(14). Since the minimum determined by the different algorithms were very close in values, therefore the error calculated were very close as well. At the end, we decided to use the value obtained from the Quasi-Newton method as the final result because it is the most consistent with the accuracy and efficient and it also generally agrees with the other algorithms in both two and three dimensions.

	Two dimensions	Three dimensions
θ_{23}	0.726 ± 0.0014	0.68438 ± 0.01593
Δm^2_{23} (eV)^2	0.0019646 ± 0.0000332	0.0020442 ± 0.0000374
α		1.2928 ± 0.05014

Figure 1.7. This is a summary of the estimated parameters with their associated errors.

V. Conclusion

The aim of the project is to estimate the key parameters of neutrinos, including the mixing angle, squared mass difference and the neutrino cross-section. This was finally achieved by using four minimisers, from which I learned the properties of each minimisers in order to use the best suited minimiser to find the best estimates of the parameters. The estimates of the parameters can be found in figure 1.6. To improve on this project, more minimisation techniques can be used and compared. Moreover, some more computationally demanding techniques can be used to determine the minimum as well.

VI. Reference

[1] Scott, M. and Dauncey, P., 2020. Computational Physics Lecture Notes. Chapter 10.

[2] Mardar, J., 2020. What Is A Neutrino...And Why Do They Matter?. [online] PBS NewsHour. Available at: <https://www.pbs.org/newshour/science/what-is-a-neutrino-and-why-should-anyone-but-a-particle-physicist-care>

[3] Fnal.gov. 2020. Fermilab | Science | Particle Physics | Neutrinos. [online] Available at: <https://www.fnal.gov/pub/science/particle-physics/experiments/neutrinos.html> [Accessed 21 December 2020].

[4] Araujo, H., 2019. Nuclear And Particle Physics (Year 2). Imperial College London.

[5] Richards, M., 2020. Second Year Statistics Of Measurements. Imperial College London.

[6] Scott, M. and Dauncey, P., 2020. Computational Physics Lecture Notes. p.Chapter 10.

[7] *The T2K Experiment*. [online] T2k-experiment.org. Available at: <https://t2k-experiment.org>].

[8] Gerald, C. and Wheatley, P., 1989. *Applied Numerical Analysis*. Reading, Mass.: Addison-Wesley.