Unconstrained minimisation of a Log-Likelihood fit to extract Neutrino oscillations parameters

CID: 01727810. Word count: 2314

***Abstract*—This investigation aims to determine muon-neutrino oscillations parameters for a set of data in the 23 quadrant (θ_{23} and Δm^2_{23}), by finding the minimum of a Negative Log Likelihood fit, assuming a Poisson distribution of events in each energy bin.**

We implement various minimisation methods in 1D and 2D. We also extend our methods to 3D, by including the neutrino cross-section (α). We find the correlation between the parameters drops with α, from $r_{\Delta m^2 \theta} = 0.338$ to 0.091, however, our values remain within 1σ for θ_{23} and 2σ for Δm^2_{23}.

The best parameters in our investigation are given by the Quasi Newton method, $\theta_{23} = 0.709 \pm 0.013$rad, $\Delta m^2_{23} = 2.743 \pm 0.038 \times 10^{-3}$ GeV^2/c^4, $\alpha = 1.271 \pm 0.006$ GeV^{-1}, corresponding to a p-value $= 1.25\%$ in a χ^2 test with our data.

Methods examined: 1D/2D - parabolic minimisers. 2D/3D - Gradient descent, Newton, Quasi Newton and Simulated annealing. We test, implement and determine the relative advantages of each method, and finally propose improvements.

I. Introduction

A. Neutrino mixing

NEUTRINOS are fundamental particles that are hypothesised to be masless in the Standard model[1]. Three different flavours exist, namely the electron (v_e), muon (v_μ) and tau (v_τ) neutrinos. However, research conducted in the latter half of the 20th century has revealed neutrino oscillations. This is a quantum mechanical effect, in which the neutrino flavour is not fixed, and oscillates. This is only possible if neutrinos have mass.

B. Our Investigation

We attempt to extract atmospheric neutrino oscillation parameters from a set of data, by minimising a negative log likelihood (NLL) fit. This is done by using un-oscillated simulated event predictions and the probability of survival of a muon neutrino (v_μ).

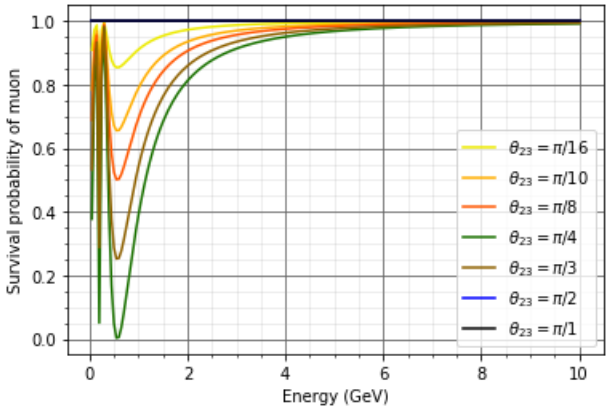
II. Theory

A. Survival probability

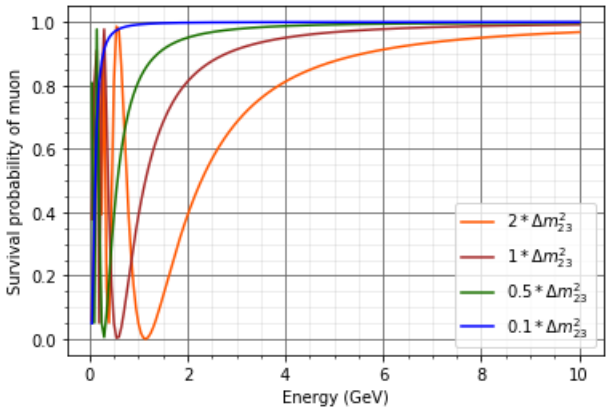
If a pure beam of v_μ flavoured neutrinos are produced a distance L (km) away from the detector, we can calculate the survival probability of each neutrino,

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

Where E is it's energy (GeV), θ_{23} is the mixing angle, describing how different the flavour states are from the mass states, and Δm^2_{23} is the mass squared difference (GeV^2/c^4), controlling the relative phase of the two mass wave-functions [1]. The variation of these values in (1) is plotted in Fig. 1. We see that when $\Delta m^2_{23} \to 0$ or $\theta_{23} \to n\pi$ (for n in $\mathbb{Z}$), there's no oscillations.



(a) Variation with θ_{23}. Maximum oscillations at $\theta = \pi/4$.



(b) Variation with Δm^2_{23}. As Δm^2_{23} increases and at larger E, v_μ is unlikely to change flavours. At smaller E, the frequency of oscillation increases.

Fig. 1: Survival probability against E. We fix $\theta_{23} = \pi/4$, $\Delta m^2_{23} = 2.4 \times 10^{-3}$ and $L = 295$.

B. NLL

The data-set can be assumed to be a discrete measurements at each energy bin, and hence follow a Poisson distribution. The oscillated event rate prediction ($\lambda_i(\mathbf{u})$), is calculated by multiplying (1), with the unoscillated event prediction, at each bin. The observed number of events at each bin is m_i. The NLL formula for a Poisson distribution is,

$$NLL(\mathbf{u}) = \sum_{i=1}^{n}\left[\lambda_i(\mathbf{u}) - m_i + m_i \ln\left(\frac{m_i}{\lambda_i(\mathbf{u})}\right)\right] \quad (2)$$

where Stirling's approximation $\ln(m!) \approx m\ln(m) - m$ has been used [2], to reduce the number of operations needed. The parameters $\mathbf{u} = (\theta_{23}, \Delta m^2_{23})$. We plot (2) in Fig. 2.

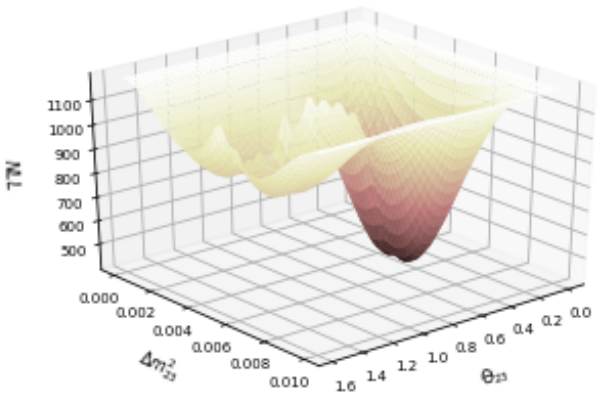


Fig. 2: NLL as a function of $\mathbf{u}$. Fixed $L = 295$.

C. Neutrino Cross-Section

The neutrino cross-section should be included in our analysis as it is known to scale proportionally with λ_i [2],

$$\lambda_i^{\text{new}}(\mathbf{u}') = \alpha \lambda_{ij}^{\text{old}}(\mathbf{u}) E_j. \tag{3}$$

Now $\mathbf{u}' = (\theta_{23}, \Delta m^2_{23}, \alpha)$.

We will investigate the importance of this term as the investigation progresses. Fig. 3 shows λ as a function of α.

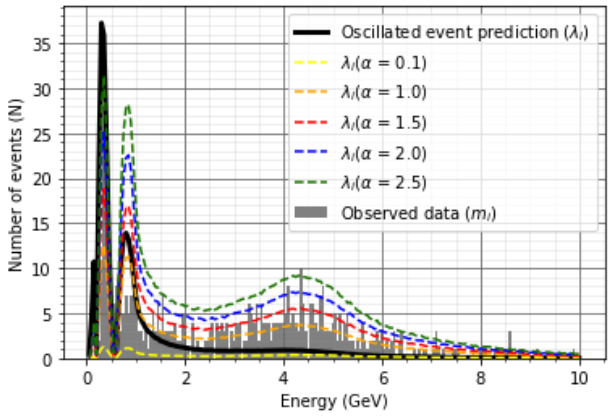


Fig. 3: Histogram of event rate (m_i), $\lambda^{new}(\mathbf{u}')$ and $\lambda^{old}(\mathbf{u})$, as a function of E. Fixed $\theta_{23} = \pi/4$, $\Delta m^2_{23} = 2.4 \times 10^{-3}$ and $L = 295$. It is clear that taking into account α reduces the difference between the λ_i and m_i, especially at higher E.

III. Methods

A. 1D Minimisation

1) Parabolic method: The method minimises one parameter at a time, by fitting a second order Lagrange polynomial (L_2) at three initial starting guesses; x_i for $i = 1, 2, 3$. We then obtain the minimum of the parabola (x_3), going through $NLL(x_i)$,

$$x_3 = \frac{1}{2} \frac{\left(x_2^2 - x_1^2\right) y_0 + \left(x_0^2 - x_2^2\right) y_1 + \left(x_1^2 - x_0^2\right) y_2}{\left(x_2 - x_1\right) y_0 + \left(x_0 - x_2\right) y_1 + \left(x_1 - x_0\right) y_2} \tag{4}$$

x_3 replaces x_i corresponding the maximum value of f. We repeat this process until our convergence criteria (ϵ) is satisfied:

$$|x_i - x_3| \leq \epsilon \tag{5}$$

This convergence criteria is extended to other methods, where x_3 is the new estimate of the minimum calculated.

B. 2D non-simultaneous minimisation

1) Univariate method: In 2D, we use the parabolic method to find the minimum of each parameter successively, with 6 initial guesses. This method is very susceptible to getting stuck at local minimums, when we vary the initial guesses for the parameters.

To avoid this, we start with a large convergence criteria (ϵ_i), to allow the parameter search to be more extensive. Once this criteria was satisfied, ϵ_i is reduced by a factor (m) and the method optimises the second parameter.

while $\epsilon_i > \epsilon$ **do**
  Run both parabolic methods with ϵ_i
  $\epsilon_i = \epsilon_i / m$
end while

This is performed successively until we satisfy ϵ.

C. 2D and 3D Simultaneous minimisation

1) Gradient method: The Gradient method computes the grad of our function at each step, $\vec{\nabla} f(\vec{x}_n)$. This is done so that the change in x is in a descent direction,

$$\vec{x}_{n+1} = \vec{x}_n - \alpha_{step} \vec{\nabla} f(\vec{x}_n). \tag{6}$$

with $\alpha_{step} << 1$ [2]. We iterate until (5) is satisfied.

This method is slow, so we attempt to improve its speed of convergence by adding momentum (ρ) [3].

$$\vec{x}_{n+1} \mathrel{+}= \rho \cdot (\vec{x}_n - \vec{x}_{n-1}). \tag{7}$$

As this uses information from the previous step, the first step won't have any momentum contribution.

2) Newton method: The Newton method considers the local curvature at each point instead, and we iterate using,

$$\vec{x}_{n+1} = \vec{x}_n - [\mathbf{H}(\vec{x}_n)]^{-1} \cdot \vec{\nabla} f(\vec{x}_n) \tag{8}$$

where the Hessian matrix ($\mathbf{H}$) is given by,

$$H_{ij}(\vec{x}) = \frac{\partial^2 f(\vec{x})}{\partial x_i \partial x_j} \tag{9}$$

For this method to work, we require a H that's positive definite; symmetric, with eigenvalues > 0 [4]. If this is not the case, the inverse of (9) may either not be defined, or not be a descent direction, so the method would fail or we'd would move away from the minimum [5].

We attempt to improve the method by enforcing the direction of $-\vec{\nabla} f$. We also ensure (9) is positive definite, as its not allowed to proceed if this is not the case. This allows us to initialise the method with values further away from the expected minimum.

3) Quasi Newton: We approximate the Inverse hessian to $\mathbf{G_n}$, where the steps are given by,

$$\vec{x}_{n+1} = \vec{x}_n - \alpha_{step}\mathbf{G}_n \cdot \vec{\nabla} f\left(\vec{x}_n\right). \tag{10}$$

The updated matrix is defined by the Davidon-Fletcher-Powell algorithm,

$$\mathbf{G}_{n+1} = \mathbf{G}_n + \frac{\vec{\delta}_n \cdot \vec{\delta}_n^T}{\vec{\gamma}_n \cdot \vec{\delta}_n} - \frac{\mathbf{G}_n \cdot \left(\vec{\gamma}_n \vec{\gamma}_n^T\right) \cdot \mathbf{G}_n}{\vec{\gamma}_n \cdot \mathbf{G}_n \cdot \vec{\gamma}_n} \tag{11}$$

Where $\vec{\gamma}_n = \vec{\nabla} f(\vec{x}_{n+1}) - \vec{\nabla} f(\vec{x}_n)$ and $\vec{\delta}_n = \vec{x}_{n+1} - \vec{x}_n$.

The Hessian will be positive definite by construction, if the initial $\mathbf{G_0}$ is and $\vec{\delta}_n^T \cdot \vec{\gamma}_n > 0$. This means $f(x_{n+1}) < f(x_n)$ for $\alpha_{step} << 1$. We will use $\mathbf{G_0} = \mathbb{1}$.

4) Monte-Carlo minimisation - Simulated Annealing: The last method we investigate, uses a metropolis algorithm to generate random numbers following a Gaussian distribution of the initial guess, to find the global minimum.

We will use the analogy that the evaluation of our cost function, f corresponds to an energy, E.

When we generate a new guess E_2, we accept it, if its less than our previously accepted value E_1. If its more, we accept it with a probability dependent on a Boltzmann distribution, at a the system's temperature (T).

if $E_2 \leq E_1$ **then**
  $p_{\text{acc}} = 1$
end if
if $E_2 > E_1$ **then**

$$p_{\text{acc}} = \exp(-\Delta E/T) \tag{12}$$

end if

D. Errors

1) Parabolic curvature: The inverse of the second derivative of L_2 is used for our Parabolic error in 1D, which is sufficiently accurate and isn't computationally demanding.

2) Secant method: An improved method to find the error of our minimum, is to find the parameter values that shifts the NLL by 0.5. We do this by finding the roots of,

$$F(\mathbf{u}) = NLL(\mathbf{u}) - (NLL_{min} + 0.5), \tag{13}$$

using an automatic Secant method. Where NLL_{min} is the value of the NLL at the minimum. This could alternatively be done by iterating or using the last parabolic fit. These methods would be less efficient and accurate, as these assume a symmetric error - the Secant method may give us different uncertainties depending in the direction of the scan (Δx^+ and Δx^-). This method will be used to find the errors for the Univariate and Gradient descent.

3) Co-variance matrix: This method becomes necessary with correlated variables [6];

$$E_{ij} = r_{ij}\sigma_i\sigma_j \tag{14}$$

Where σ_i is the uncertainty on variable i, and r_{ij} is the Pearson correlation coefficient, which determines the correlation between two variables (i, j). In our investigation $(H^{-1})_{ij} = E_{ij}$, which we will use to find the errors for the Quasi Newton and Newton methods.

The diagonal terms have $r_{ii} = 1$, so $E_{ii} = \sigma_i^2$. Using the off-diagonal terms, $r_{ij} = E_{ij}/\sigma_i\sigma_j$.

IV. Testing

To verify our gradient methods work, we test $f_1(x, y, z) = (xyz)^3$ and $f_2(x, y, z) = sin(x)cos(y) + e^{z^2}$.

The Secant method, and minimising methods are also tested, some shown in Fig. 4 and 4c. The minimum of (1) was also tested, agreeing with $\theta = \pi/4$. We can therefore proceeded to implementing the method in our investigation.

V. Trust-regions

A. Scaling

The parameters θ_{23}, Δm^2_{23} and α, are of $\mathcal{O}(-1)$, $\mathcal{O}(-3)$, $\mathcal{O}(1)$. This will become a problem as the same perturbation will have different effects on the NLL. We solve this by scaling our values to $\mathcal{O}(1)$, and re-scaling them before evaluating the NLL and δ_n in (11).

B. Step size and convergence criteria

Gradients are calculated using a Central Difference Scheme for which we apply a perturbation of $h = 10^{-3}$ to values of $\mathcal{O}(1)$. This provides a balance between the error $\mathcal{O}(10^{-6})$) and computational expense of having a smaller h. A small $\alpha_{step} = 10^{-6}$ in (6) and (10) is chosen to satisfy the Wolfe conditions [5]. If it's too small, the convergence will take a long time. If it's too large, we may not be able to convergence onto a minimum, as it may skip it.

For the Secant method, we apply $\epsilon = 10^{-6}$. For all minimisation methods, $\epsilon = 10^{-5}$, as this is a fractional difference of the change in NLL. The Univariate method has $\epsilon_i = 10^{-1}$ and $m = 10$.

C. Initial guesses

We plot the variation of the NLL as a function of each parameter, in Fig. 5, to find the expected global minimum.

VI. Results

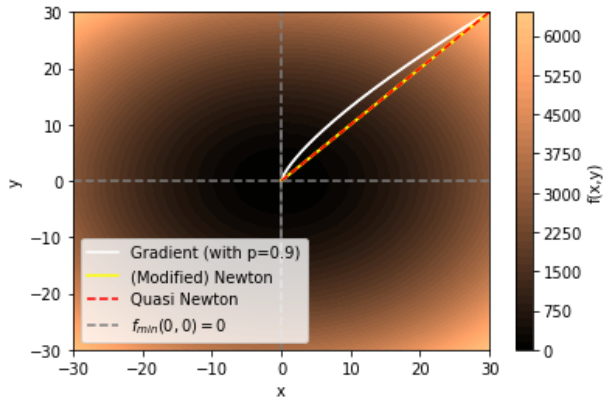
A. 1D parabolic

We initialise the parabolic method with $\theta_{23} = [0.0, 0.5, 0.7]$ or $[0.5, 1.0, 1.5]$ to obtain the two minimums, and $\Delta m^2_{23} = [0.002, 0.003, 0.004]$. We obtain $\theta_{23} = 0.712 \pm 0.011$rad and 0.859 ± 0.011rad, and $\Delta m^2_{23} = 2.644 \pm 0.017 \times 10^{-3}$.

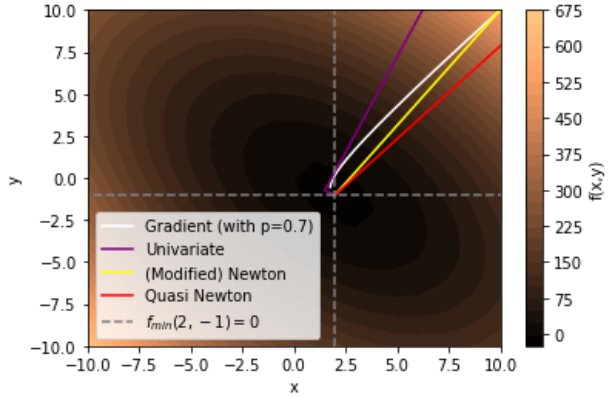
B. 2D Univariate non-simultaneous minimisation

The Univariate method is initialised with the same parameters as the parabolic methods. We find $\theta_{23} = 0.715^{+0.010}_{-0.012}$, $\theta_{23} = 0.856^{+0.012}_{-0.010}$rad, and $\Delta m^2_{23} = 2.604^{+0.041}_{-0.048} \times 10^{-3}$, independently of the order the parameters are optimised.

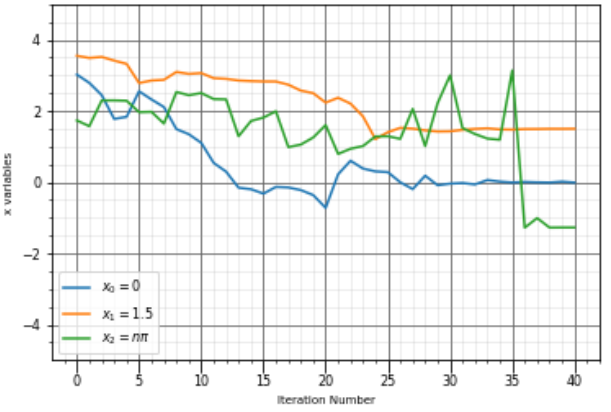
To understand the effect of the varying θ_{min} on our error, we plot Fig. 6. We notice, the error of $+\Delta\theta_{23}$ may increase as $\theta_{min} \to \pi/4$.



(a) $f(x,y) = 4x^2 + 3y^2 + 2$ converging to $f(0,0) = 0$

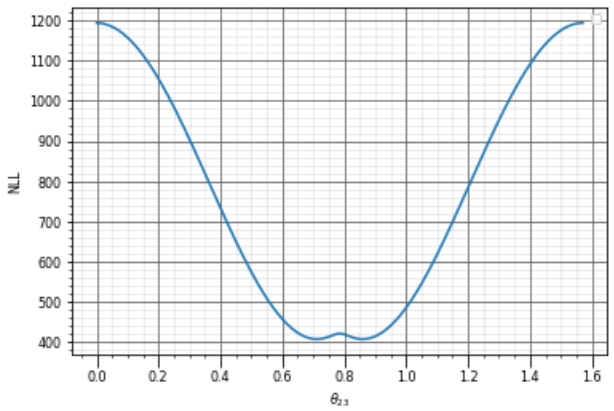


(b) $f(x,y) = 2x^2+2xy+2y^2-6x$ converging to $f(2,-1) = 0$.

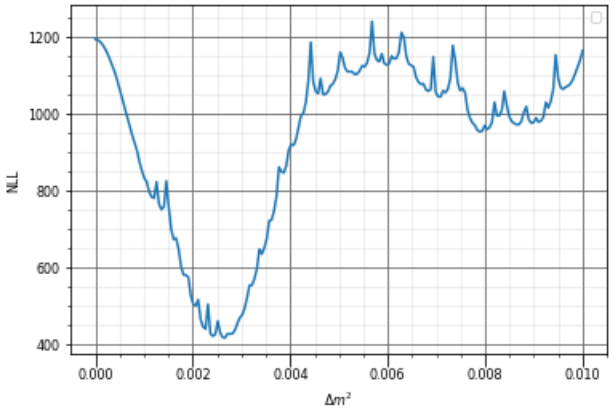


(c) $f(x_1,x_2,x_3) = x_1^2 + (2x_2-3)^2 + sin(x_3)^2$ for simulated annealing, converging to $f(0,3/2,n\pi) = 0$. $T_i = 20, T_f = 0$, with $n = 10k$ iterations per $\Delta T = 0.5$.

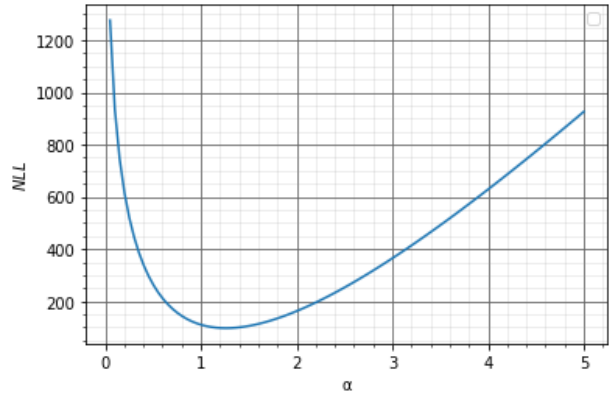
Fig. 4: Example of three test functions for our minimisation methods.



(a) As a function of θ_{23}. We expect two minimums around 0.7 and 0.9, neighbouring the point symmetry of $\approx \pi/4$ in (1).

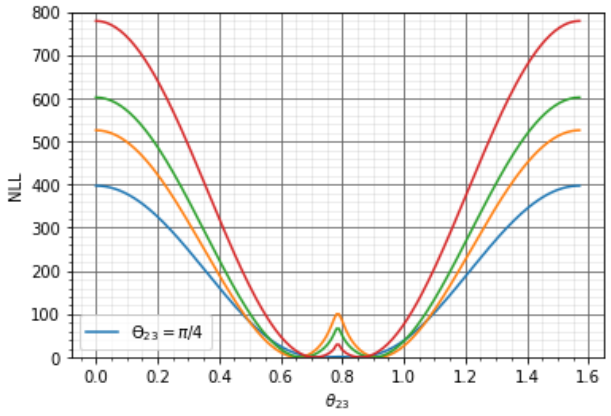


(b) As a function of Δm^2_{23}. We determine a minimum around $\Delta m^2_{23} \approx 2.5$.

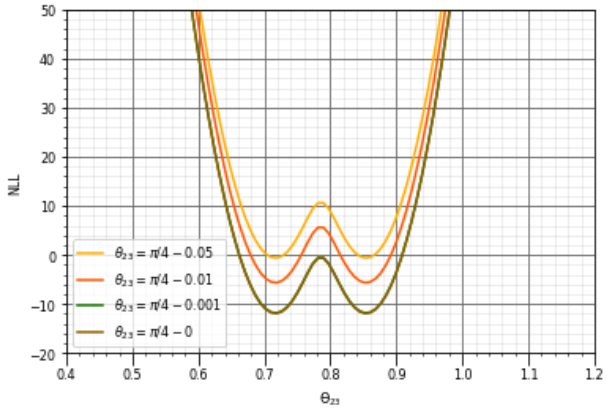


(c) As a function of α. We determine a minimum around $\alpha \approx 1$.

Fig. 5: NLL as a function of each parameter. We fix $\theta_{23} = \pi/4$, $\Delta m^2_{23} = 2.4 \times 10^{-3}$, $L = 295$.



(a) Variation in the shape of the NLL. The spike defining the symmetry disappears when $\theta_{23} = \pi/4$.



(b) Variation of (13). The spike defining the upper root disappears when $\theta_{23} = \pi/4$.

Fig. 6: NLL plot as a function of θ_{23}, as $\theta_{min} \to \pi/4$.

C. 2D Simultaneous minimisation

A summary of the results is presented in Table I, and plotted in Fig. 7. We have focused our finds on the lower θ_{23}, which will be justified later using Simulated Annealing.

We pick $\rho = 0.8$ which is stable and yet still gives a significant reduction to the iteration number. Further information is presented in the Appendix, Fig. 13a.

	2D simultaneous methods	
Method	θ_{23} (rad)	Δm^2_{23} (10^{-3} GeV$^2/c^4$)
Gradient	$0.713^{+0.012}_{-0.010}$	$2.601^{+0.042}_{-0.050}$
Grad. ρ=0.8	$0.714^{+0.012}_{-0.010}$	$2.603^{+0.043}_{-0.049}$
Q.N.	0.714 ± 0.011	2.603 ± 0.050
Newton	0.714 ± 0.011	2.600 ± 0.047
Mod.N.	0.713 ± 0.011	2.600 ± 0.047

TABLE I: Table of results for all 2D simultaneous methods. The initial guesses to our methods are $[\theta_{23}, \Delta m^2_{23}] = [0.5, 2 \times 10^{-3}]$. The Newton method uses $\Delta m^2_{23} = 2.6 \times 10^{-3}$.

1) Correlation 2D: We examine further the inverse hessian of the Newton method, at the minimum, to calculate the correlation between our two parameters, $r_{\theta \Delta m^2} = 0.338$. Due to the large correlation, we attempt to find a better estimate for the Gradient uncertainty, that uses (13).

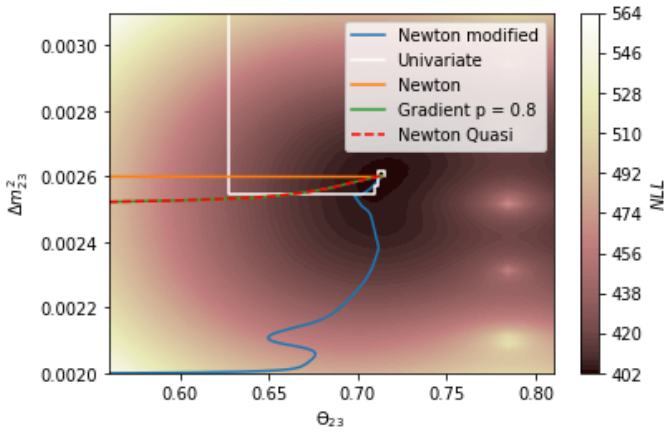


Fig. 7: NLL contour plot showing convergence for all 2D methods.

Using a Monte-Carlo algorithm, we generate new parameter guesses. If these lie within $(NLL + 0.5) \pm 0.01$, we plot them in Fig. 8, and find the min. and max. values. The reconsidered results become, $\theta_{23} = 0.713^{+0.012}_{-0.011}$ and $\Delta m^2_{23} = 2.601^{+0.045}_{-0.053} \times 10^{-3}$. The errors have increased by a very small amount, suggesting the Secant method remains valid.

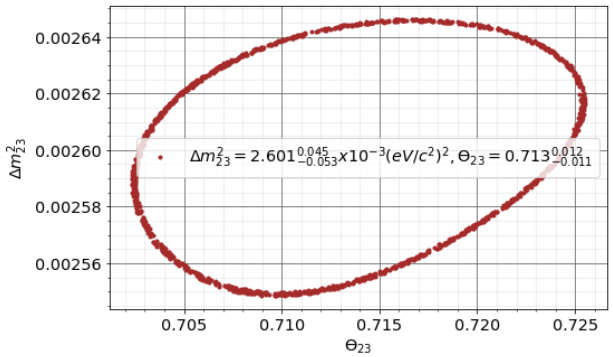


Fig. 8: Plot of generated random parameters distributed according to a Gaussian probability density function, around $\mu = (\theta_{min}, \Delta m^2_{min})$, and σ =associated uncertainty.

D. 3D minimisation

The results for 3D minimisation is presented in Table II, and plotted in Fig. 9. A similar method to 2D is used to determine the value of $\rho = 0.5$, in 3D, however, we still see that momentum can make the method from overshoot at specific turning points (Δm^2_{23}).

1) Correlation 3D: Similarly to before, we obtain new values for the correlations; $r_{\theta \Delta m^2} = 0.091$, $r_{\theta \alpha} = 0.035$, $r_{\Delta m^2 \alpha} = 0.303$. The correlation $r_{\theta \Delta m^2}$ has decreased, hence our uncertainty calculation for the Gradient method is even more valid. Although $r_{\theta \alpha}$ is small, it would be necessary to further examine the correlation between $r_{\Delta m^2 \alpha}$, by possibly introducing a fourth parameter into our investigation.

	3D minimisation			
Method	θ_{23} (rad)	Δm^2_{23} (10^{-3} GeV$^2/c^4$)	α (GeV^{-1})	N
Gradient	$0.709^{+0.013}_{-0.012}$	2.743 ± 0.038	1.267 ± 0.001	19418
G. ρ=0.5	0.709 ± 0.013	$2.742^{+0.038}_{-0.037}$	1.268 ± 0.001	10953
Q.N.	0.709 ± 0.013	2.743 ± 0.038	1.271 ± 0.006	8001
Newton	0.709 ± 0.013	2.741 ± 0.038	1.270 ± 0.006	-
Mod.N.	0.709 ± 0.013	2.743 ± 0.038	1.270 ± 0.006	8273

TABLE II: Table of results for all 3D methods. The initial parameters used are $[\theta_{23}, \Delta m^2_{23}, \alpha] = [0.5, 2 \times 10^{-3}, 1]$. The Newton method uses $\Delta m^2_{23} = 2.6 \times 10^{-3}$.

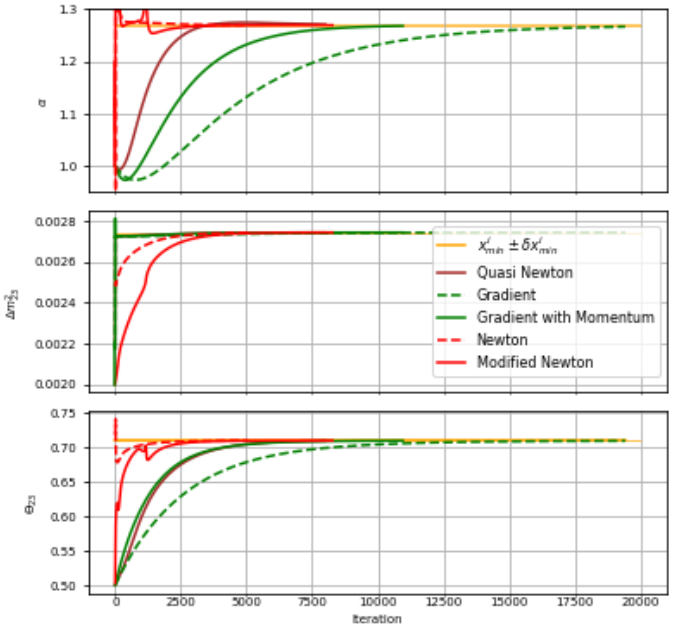


Fig. 9: Plot comparing all 3D methods against N. The yellow area represents the uncertainty in the location of the minimum.

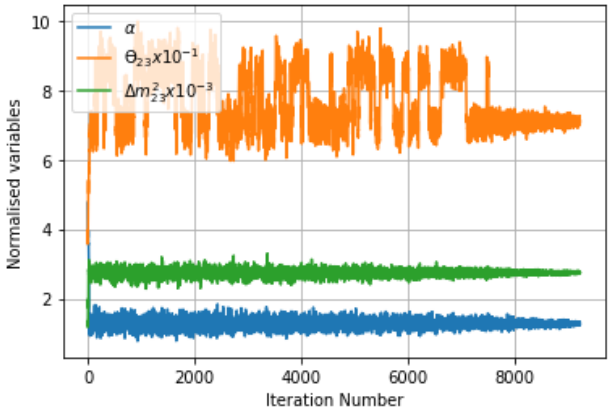
2) Monte-Carlo minimisation: The results obtained are $\theta_{23} = 0.709^{+0.015}_{-0.014}$rad, $\Delta m^2_{23} = 2.743^{+0.045}_{-0.037} \times 10^{-3}$ and $\alpha = 1.27^{+0.08}_{-0.04}$. New parameter guesses are created on a Gaussian probability density function with μ =old parameter, $\sigma = 0.2$. The convergence as T decreases is seen in Fig. 10.

This method was repeated 10 times, as the algorithm ceased to output a better estimate for the parameters. The convergence of a global minimiser onto our lower θ_{23} value reassures that our results for the other 2D and 3D methods are global minimums.

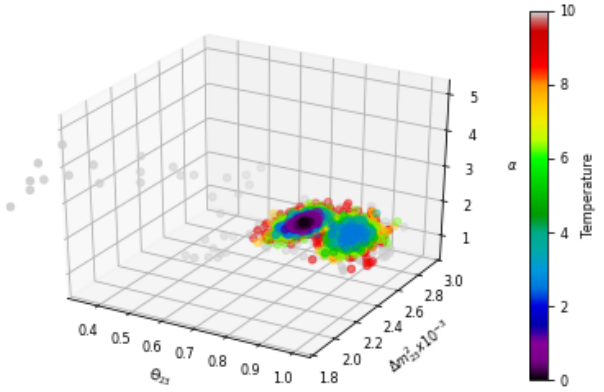
Further visualisations of the dynamics of the system is seen in Appendix Fig. 13c.

The errors for this method are obtained using a similar method to Fig. 8. We map out the values which vary the NLL_{min} by ± 0.01. The min. and max. of the 2D projection represents the associated uncertainty in our measurement, shown in Fig. 11.

3) Fitting the data: To compare all our findings, we perform a χ^2 squared test. We find corresponding $p - value$ of $p = 1.27\%$ for both the Newton methods and Simulated Annealing, $p = 1.28\%$ and $p = 1.29\%$ for the gradient method



(a) Iterations of the normalised parameter search.



(b) Visualisation of the variation of the parameters. At higher T, we explore a greater range, escaping local minimums. As T decreases it converges to the global minimum.

Fig. 10: Plots showing the convergence to the global minimum using the Simulated Annealing method. With initial guesses of $[\theta_{23}, \Delta m^2_{23}, \alpha]$=$[0.4, 10^{-3}, 5]$. We use $T_i = 10$ and $T_f = 1$, with $n = 6000$ iterations at each $\Delta T = 1$. Then $T_i = 1$ and $T_f = 0$, with $\Delta T = 0.1$ and the same n.

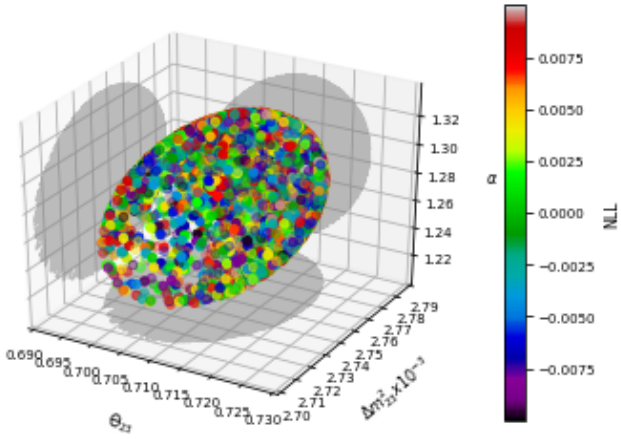


Fig. 11: Plot of generated random parameters, distributed according to a Gaussian function; $\mu = (\theta_{min}, \Delta m^2_{min}, \alpha_{min})$, and σ = standard deviation of $T = 0.1 \to 0$.

with and without momentum respectively, and $p = 1.25\%$ for

the Quasi Newton method. This means that $> 98\%$ of the time our parameters will be a good fit to the data, hence we can accept all our values to a significance level of 5%; a widely used value in neutrino research [7]. This is plotted in Fig. 12.

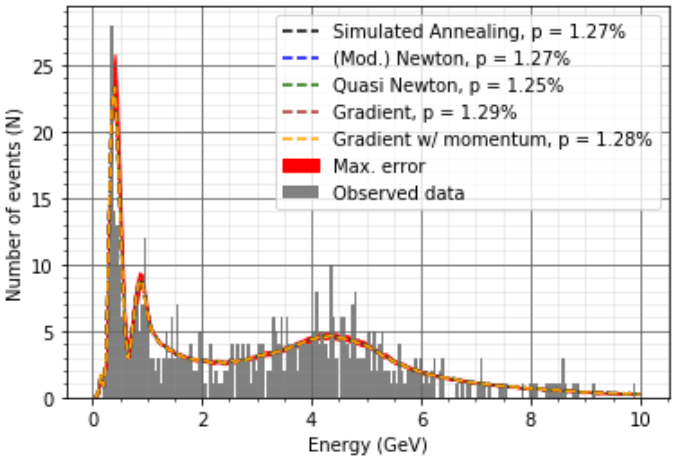


Fig. 12: Histogram of data (m_i), and fit (λ_i) with the optimum parameters found for each method and the maximum error. The calculated p-value of our χ^2 squared test is also shown, in which we have compared the experimental data, to the fit with the parameters minimising the NLL, for each method.

VII. Discussion

A. Parabolic and Univariate methods

The Univariate method is the least efficient method as it consists of more iterations of the Parabolic method. These are also inadequate, as the parameters aren't minimised simultaneously, hence the NLL change of shape might be confused between them.

B. Gradient

The gradient method is the second least efficient process, with the largest N. However, we've seen ρ reduces N by 44% in our investigation. In future investigations, we need to reconsider our method of choosing ρ, as Fig. 12 and Table II suggest that adding momentum not only increases the efficiency, but also the accuracy of the method.

The computational cost per iteration is lower than the Newton method. This becomes attractive when you don't want to deal with the second derivatives of a function, where only $\mathcal{O}(N)$ values are needed, in comparison to the second derivatives, which is $\mathcal{O}(N^2)$. Nonetheless, we've seen the accuracy of the method is lower than Newton (Fig. 12).

C. Newton and Quasi Newton

Both Newton methods give the same minimums with less N than the Gradient method. The modified Newton method presents greater irregularities as shown in Fig. 9, as we are forcing a change in direction when it attempts to go 'uphill', this could impose a greater instability in future investigations, possibly breaking down at some points. A proposition would be to modify H using Cholesky Factorization method, which is more stable [8].

A downside of using the Newton method is that it requires $\mathcal{O}(N^3)$ for matrix Inversion, as well as needing to calculate the second derivatives of (9). The modified Newton method even becomes more expensive, as it needs to spectrally decompose the Hessian to make sure its positive definite. This will generally be slower and more inefficient than the Quasi Newton method, which reduces the number of operations to $\mathcal{O}(N^k)$ where $k < 3$ [2].

The newton method was attempted to be modified differently, by adding small multiples of the identity matrix until we acquired a positive definite Hessian [5]. This took an incredible amount of iterations so wasn't implemented.

Surprisingly, the Quasi Newton, asides from presenting a higher efficiency, it also found better parameters for our fit, as shown in Fig. 12.

D. Simulated annealing

This method is ensured to find the global minimum of the NLL. However, it is inefficient as it performs thousands of iterations at each temperature, and repeating the whole process multiple times may not be feasible in larger-scale problems. To continue our investigation, we could explore how we could determine the optimum parameters (such as N and T) to increase its efficiency and minimise the running time [8].

VIII. Conclusion

To summarise, we implemented the Univariate, Gradient, Quasi-Newton, Newton and Simulated Annealing minimisation methods to find the parameters describing muon-neutrino oscillations in the 23 quadrant, that correspond to the minimum of the NLL, and their associated errors. We also proposed and implemented improvements to some of these methods.

We investigated different ways of obtaining uncertainties, as well as also evaluating the Hessian matrix to obtain the correlation between different parameters. We found a decrease in the correlation between Δm^2_{23} and θ_{23}, when we took into account α, from $r = 0.338$ to 0.091, however, our estimates for the parameters remained within 1σ for θ_23 and 2σ for Δm^2_{23}.

To conclude, the best parameters to our data are given by the Quasi Newton method, $\theta_{23} = 0.709 \pm 0.013$rad, $\Delta m^2_{23} = 2.743 \pm 0.038 \times 10^{-3}$ GeV^2/c^4 and $\alpha = 1.271 \pm 0.006 \mathrm{GeV}^{-1}$. These correspond to a p-value $= 1.25\%$ in our χ^2 test. Accepting the fit with a 5% significance level.

References

[1] Y. Ashie, J. Hosaka, and K. e. a. Ishihara, "Measurement of atmospheric neutrino oscillation parameters by super-kamiokande i," *Phys. Rev. D*, vol. 71, p. 112005, Jun 2005. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevD.71.112005

[2] M. Scott and P. Dauncey, "Computational physics 2021-2022, project1: A log-likelihood fit for extracting neutrino oscillation parameters," Imperial College London, December 2021.

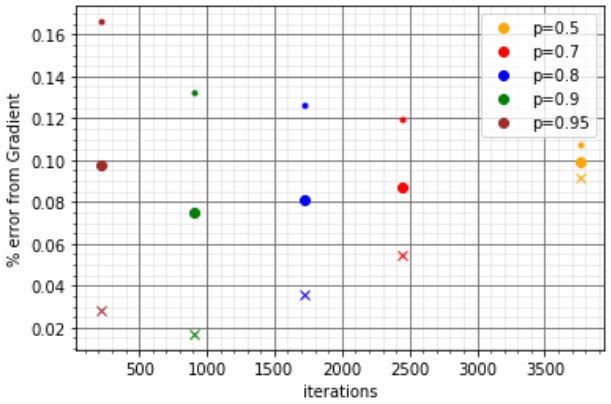
[3] V. Bushaev, "Stochastic gradient descent with momentum," Towards data science, online. Accessed: 16/12/2021. [Online]. Available: https://towardsdatascience.com/stochastic-gradient-descent-with-momentum-a84097641a5d

[4] E. Weisstein, "Positive definite matrix," Wolfram Research, online. Accessed: 16/12/2021. [Online]. Available: https://mathworld.wolfram.com/PositiveDefiniteMatrix.html
[5] J. N. J. Wright, *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering book series, 2006.
[6] M. Richards, "Second year statistics of measurement - lecture 7 the chi-squared estimation method. non-examinable: The error matrix," Imperial College London, year 1 Physics notes.
[7] L. Stanco, "Statistics and data analyses for neutrino experiments," INFN-Padova, section H: Statistical Methods for Physics Analysis in the XXI century. Online. Accessed: 15/12/2021. [Online]. Available: https://indico.cern.ch/event/648004/contributions/3032189/attachments/1695987/2729866/Stanco-neutrinos-XIIIquark-short.pdf
[8] S. Szykman, L. Schmidt, and H. Shetty, "Improving the efficiency of simulated annealing optimization through detection of productive search," 10 1997.

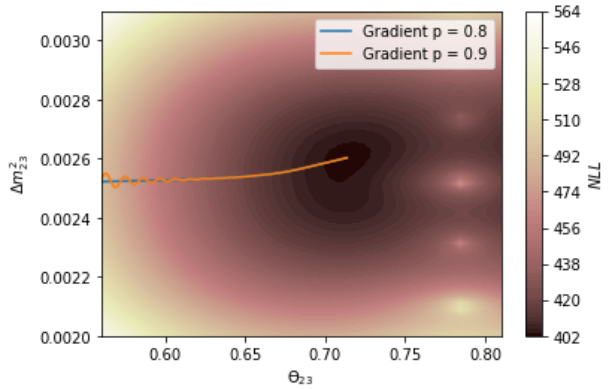
APPENDIX

We optimise our choice of ρ by minimising both the $\%$ error from the Gradient and N. As the error difference between $\rho = 0.9$ and 0.8 is $< 0.01\%$, and $\rho = 0.9$ presents a less stable value, we use $\rho = 0.8$. This is seen in Fig. 13a and 13b.

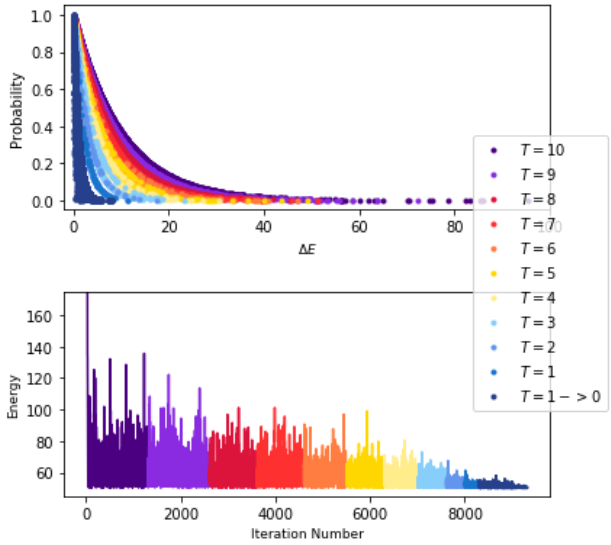
The system dynamics of simulated annealing is shown in Fig. 13c.



(a) $\%$ error against N for the Gradient method with ρ. We use [cross (x), small dot, large dot] = error in [θ_{23}, Δm^2_{23}, average error].



(b) Convergence to the minimum of both Gradient methods.



(c) Simulated Annealing dynamics. Plot of the P_{acc} and E decreasing with T.

Fig. 13: Appendix plots