UNIVERSITY OF CALIFORNIA
SANTA CRUZ

DATA ANALYSIS OF PIENU EXPERIMENT RESULTS

A dissertation submitted in partial satisfaction of the
requirements for the degree of

Applied Physics, B.S.

in

DARK MATTER RESEARCH

by

Se Rang Seo (Simon)

June 2024

The Dissertation of Se Rang Seo (Simon)
is approved:

Professor Bruce Schumm, Chair

Professor Stefania Gori

Professor Wolfgang Altmannshofer

Dean Stefano Profumo
Associate Dean for Graduate Studies

Table of Contents

List of Figures

Abstract

Data Analysis of PIENU Experiment Results

by

Se Rang Seo (Simon)

PIENU experiment studies the pion (π) decay to dark matters: axion (X or a) and sterile neutrino (N). The thesis studies the analysis done for 3 decay models: $\pi^+ \to Ne^+$, $\pi^+ \to \mu N$, and $\pi^+ \to l^+ vX$, where X = axion and N = sterile neutrino. The main tasks are: recreating signal plot, extracting and digitizing data, and mainly recreating and cross checking graphs and data fits. The methods include using digitizing plot site, python, and mathematica. The purpose of this research is to create a procedure that models a way to perform data fit, check bounds, model branching ratio, and digitize graphs for data related to other dark matter detection experiments more efficiently.

To my parents,

Yeon Hee Lee and Jeong Bong Seo,

for leaving everything behind in South Korea to help me get a better education.

Acknowledgments

I want to thank Professor Gori for allowing me to do research with her from Phys102, Modern Physics, even though my background in physics was very weak. Whenever I asked her if I were capable, she kept saying yes. Thank you Professor Wolfgang for helping me with mathematica and teaching Phys116C for as a guest lecturer. Thank you soon to be Dr. Pierce Giffin for always being nice and helpful. Thank you Ollie Jackson and Khai Luong for being the best research partners one can ever ask for. Thank you everyone in my research group for always being positive and supportive.

Chapter 1

Introduction

Pion is a bosonic, subatomic particle with spin value of 0, composed of up and down quirks making it a meson. Its most common decay is found to be pion decay to muon and muon neutrino ($\pi \rightarrow \mu v_{\mu}$), which are few of many elementary particles that we have discovered. Particle physicists hypothesize that pions have a potential to decay into dark matter particles namely, sterile neutrino (also known as heavy neutrino) and axion.

There are couple reasons why this might be the case. To name a few: baryogenesis (study of balance between matter and antimatter after the big bang), large scale structure formation (formation of galaxies), big bang nucleosynthesis (the creation of nuclei shortly after the big bang), and couple others. All this to say that the theory behind dark matter particles stems from the attempt to discover every particle that might have been created since and from the big bang. [2]

As for my research, its main focus is digitizing 3 different pion decays: pion to sterile neutrino and positron($\pi \rightarrow e^+ N$), pion to muon and sterile neutrino($\pi^+ \rightarrow \mu N$), and 3 body pion decay to ($\pi^+ \rightarrow l^+ v X$, l=e, μ and X=neutral boson, like axion). [1, 2, 3] There has been experiments already done by PIENU collaboration. They have done the pion decay experiment in 2010, and it explored the pion decay to dark matter and will be the basis for my data analysis. It might be important to mention that PIONEER is another research group whose experiment will happen in the near future, and it studies the same decay model but with a different design of the detector than PIENU's.

To study and analyze the particle decays that PIENU has experimented, I needed to know the derivation of the equations that related the pion mass/energy with its particle decays, in this case sterile neutrino (N) and positron (e^+). This was not only to understand the fundamental concept of particle physics, but also be able to to use these equations for Monte Carlo and for replicating graphs where E_e^+ is needed in digitizing and creating graphs.

Starting with the energy-momentum relation, I solve for E_N and E_{e^+} by solving for p_v in terms of only m_{e^+}, m_π, and m_N:

$$E_N = \sqrt{m_N^2 c^4 + p_v^2}$$

With c=1 and squaring both sides:

$$E_N^2 = m_N^2 + p_N^2 \tag{1.1}$$

and

$E = m$, if $p = 0$ (particle is at rest)

Then:

$$m_\pi = E_{e^+} + E_N$$

$$m_\pi = \sqrt{p_{e^+}^2 + m_{e^+}^2} + \sqrt{p_N^2 + m_N^2}$$

Through conservation of energy, we know that $p_N - p_{e^+} = 0$ because v and e^+ decay from the same source of pion, and they decay anti-parallel to each other. So, $p_N = p_{e^+}$.

$$m_\pi = \sqrt{p_N^2 + m_{e^+}^2} + \sqrt{p_N^2 + m_N^2}$$

Through some algebra, we find P_N to be:

$$P_N = P_{e+} = \frac{m_{e+}^4 + m_N^4 + m_\pi^4 - 2m_{e+}^2 m_N^2 - 2m_{e+} m_\pi^2 - 2m_N^2 m_\pi^2}{2m_\pi^2}$$

Plugging this back into (1.1) to solve for E_N:

$$E_N^2 = m_N^2 + \frac{m_{e+}^4 + m_N^4 + m_\pi^4 - 2m_{e+}^2 m_N^2 - 2m_{e+} m_\pi^2 - 2m_N^2 m_\pi^2}{2m_\pi^2}$$

$$E_N = \frac{-m_{e+}^2 + m_N^2 + m_\pi^2}{2m_\pi}$$

Since m_{e^+} is very small and since $m_\pi > m_N$, $\pm$ goes away. And, for E_{e^+} :

$$E_{e^+}^2 = p_{e^+}^2 + m_{e^+}^2$$

Doing same steps, plugging in p_{e^+}:

$$E_{e^+} = \frac{m_{e+}^2 - m_N^2 + m_\pi^2}{2m_\pi} \tag{1.2}$$

We can also derive m_N using this equation. Solving for m_N, we get:

$$m_N = \sqrt{m_\pi^2 + m_{e^+}^2 - 2m_\pi E_{e^+}} \tag{1.3}$$

With this knowledge, I will be able to understand the graphs that's been produced from PIENU experiment, which use positron energy (MeV) as a connection to sterile neutrino.

The goal of my research is to extract data from PIENU graphs from several papers and use them to reproduce and cross check using various methods to digitize data such as digital plotting site, python, and mathematica. Then, my research partners, Ollie and Khai, will perform data fitting. Finally, we use this entire procedure or model of verification on other research that either focuses on discovering axionlike particle or different decays to sterile neutrino.

Chapter 2

Previous Work

PIENU collaboration did their experiment using a square compartment in the center of the steel wall as a detecting platform as shown in Fig 2.1. They shot a beam of pion into a plate where pions stay at rest, and the pion decayed into different combinations of particles. Since positron is already known, the square detector measures the possible dark matter in terms of the energy of positron (E_{e^+}). Without looking at the result, we can hypothesize certain uncertainties that might interfere with the actual signal, for example detection of other decays to positron or the detector being a square. Although it might sound trivial, the difference between PIENU and PIONEER is mostly the fact that PIONEER uses a sphere detector instead of a square one as shown in Fig. 2.2.

I used the results analyzed and produced in 3 different papers that used PIENU experiment dataset where each paper looked at the research in different lenses of different pion decay. Each paper studied 1 pion decay model of these 3: $\pi^+ \rightarrow Ne^+$, $\pi^+ \rightarrow \mu N$,

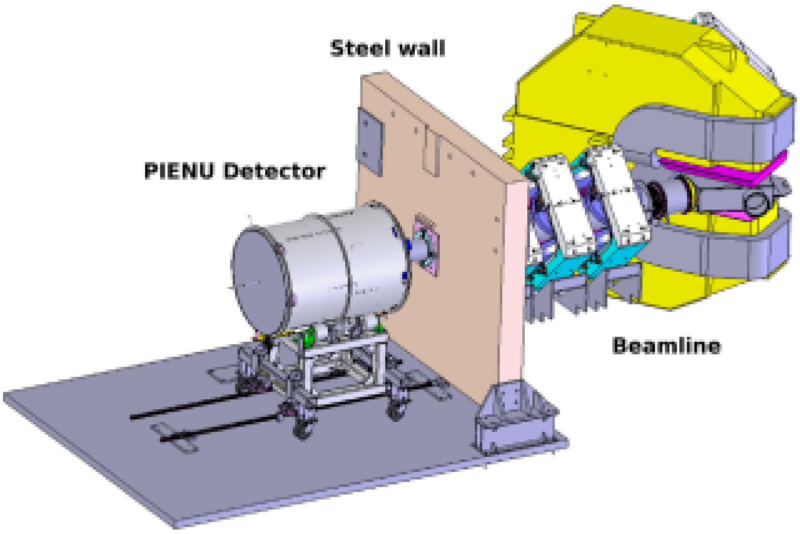


Figure 2.1: This is the PIENU detector. The small square in the center of the steel wall receives pion beam from the beamline and reads e^+ from the predicted $\pi \rightarrow e^+ N$.

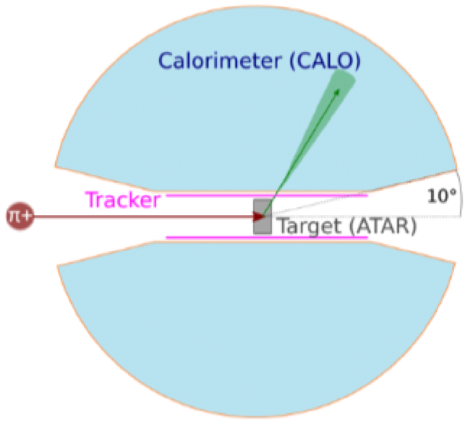


Figure 2.2: This is the model of the PIONEER detector whose experiment will happen in the near future. The pion is stopped in the center of this sphere where it is shot with a beam of energy for pion decay detection. [4]

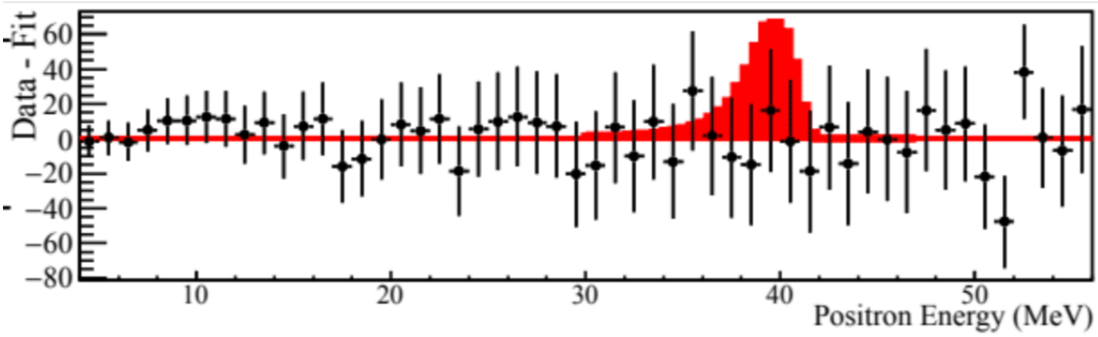


Figure 2.3: The red graph is the signal graph of sterile neutrino detection from pion decay. The black lines represent the uncertainty. [1]

and $\pi^+ \to l^+ v X$ $(X = axion)$.

The first pion decay that's presented is $\pi^+ \to e^+ N$. This studies the pion decay to positron and sterile neutrino. The resulting signal data along with the uncertainty from the PIENU experiment is plotted in Fig. 2.3. The red curve is the theoretical plot of sterile neutrino detection where $E_e^+ = 40$ MeV and using Eqn. 1.3 $m_N = 91.18$ MeV as shown in the x-axis corresponding. The signal peaks at E_{e^+}=4.1 MeV for all graphs modeling different decays. The y-axis is the data fit for this experiment which corresponds to the number of events. The black vertical lines are the uncertainties or the statistical errors, and their center points are the actual data found in the experiment.

To give a better idea, if the theory were perfect, then the black lines should be formed around the red plot which would then show the correspondence between the theory graph and the data from the experiment. Hence, the conclusion of this graph is that the PIENU experiment didn't show any signs of heavy neutrino detection from pion decay.

For the next two papers, I started working with the model for $\pi^+ \to \mu N$ shown in Fig. 2.4. But since, the graphical representations consisting of x-axis, y-axis, legends, colors, and meanings are all very similar for Fig. 2.4 and Fig. 2.5. The x-axis is T_μ which is the total kinetic energy of $\pi^+ \to \mu^+ \to e^+$, and the y-axis is the events. [3] The graph is measured where T_μ <1.2 MeV to study the lower limit. The green plot is the Gaussian graph centered at 4.1 MeV, the blue plot is the quadratic background, and the red plot is the sum of the green and blue plots. The black plot is the actual data with the uncertainty which was smeared in the plotting process to make the statistical error bars look more smooth. The plot is graphed in logarithmic scale because the number spikes up after 1.2 MeV. The difference between the top one and the bottom is that the top one has the smeared effect which changes the way I perform data fitting.

The researchers who modeled these two graphs for $\pi^+ \to \mu N$ decay do not provide us with the data fit graph. But, for the next 2 graphs, data fit is provided which makes it easier for me and my professor, Professor Altmannshofer, to cross check with each other. The conclusion of this paper also agrees with the previous analysis that no sterile neutrino was detected in the PIENU experiment.

Lastly, the final model is a 3 body pion decay of sterile neutrino, lepton, and neutral boson ($\pi^+ \to l^+ vX$), showed in Fig. 2.5. This three body decay was mentioned but not particularly explained in Fig. 2.5. This is to show that all of these analysis study the same thing with slight difference of the statistical error measuring from 1.3 MeV to 3.4 MeV = T_μ.

Finally, the data fit graphs for Fig. 2.5 is shown in Fig. 2.6. These models are created by subtracting the signal plot (red plot) by the background residuals (green and blue). There are different ways that could have done, but most likely they used Mathematica for its sophisticated way to plot and model graphs. The y-axis is the data fit where it allows us to see the comparison between the black plot (actual data) and the red plot (signal plot). As mentioned, the plot is measured from 1.4 MeV to 3.4 MeV. This shows that they don't match very well, so the detection of sterile neutrino is in the lower side.

Unlike the other plots, the x-axis for the top figure for Fig. 2.5 and Fig.2.6 subtracts 17 MeV from T_μ to make the calculation of energy because there was integrated energy of positron and the energy from the late decay positron. The average kinetic energy of this pion decay was 17 MeV. [3] Since the graph studies $\pi^+ \to l^+ vX$, the plot might have slight differences than the others. But overall, it is very similar to the other graphs. One major difference is that this graph includes the negative MeV from -4.0 MeV to 2.0 MeV. The experiment is only accurate if the pion is fully at rest so that momentum of the decaying particles is equal, which was derived earlier in the introduction. This also accounts to the subtraction of any potential kinetic energy calculated.

In total, there was the one PIENU experiment that studied the pion decay at rest, using square detector. There were many papers released to study their results, but this paper focuses on the 3 different pion decays from 3 different papers that produced very similar plots with the same conclusion that no sterile neutrino was detected. From there, my research group will perform data fits, digitize plots, and use the overall results to cross check that we all get the same results.

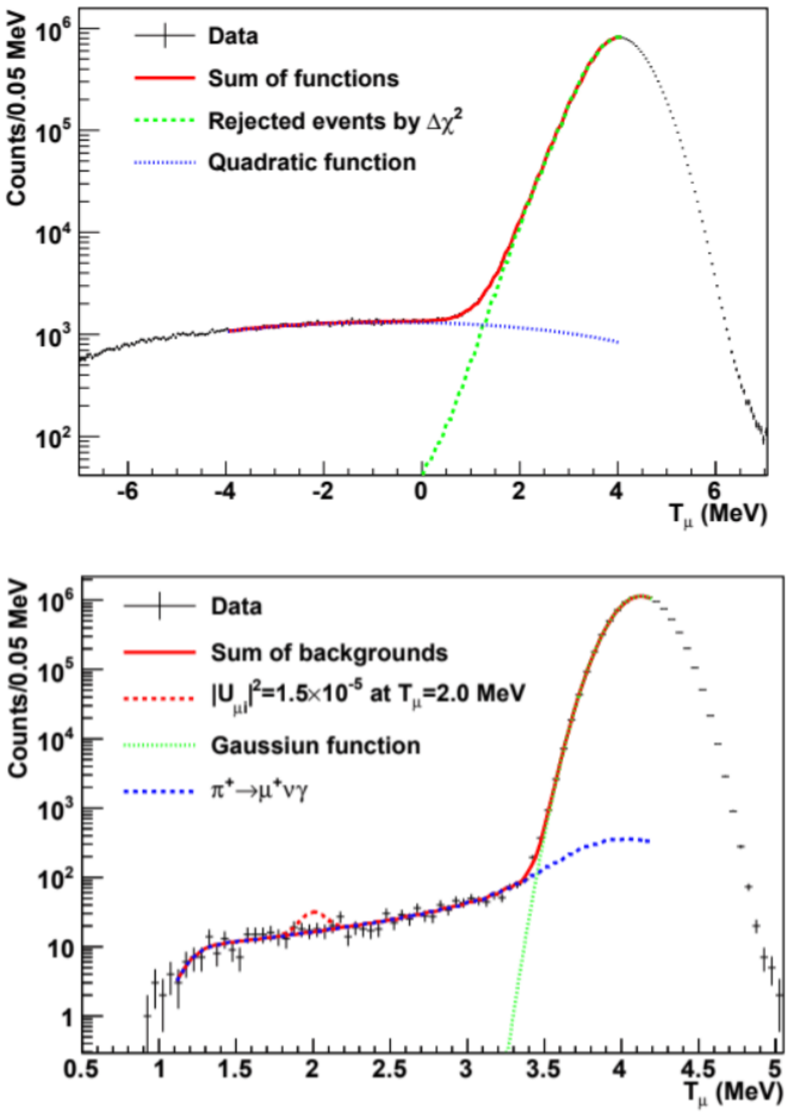


Figure 2.4: Signal graph of $\pi^+\!\rightarrow\!\mu^+\!\rightarrow\!e^+$ decays or $\pi^+\!\rightarrow\!\mu^+v$. T_μ is the total kinetic energy of these decays and the counts is the number of runs. This graph has T_μ <1.2 MeV. The green is the Gaussian peaked at 4.1 MeV and blue is the quadratic background, the red is the fit plot of the backgrounds, and the black plot is the data with statistical error. The peak is at T_μ=4.1 MeV. The uncertainty is plotted with a smearing effect. [2]

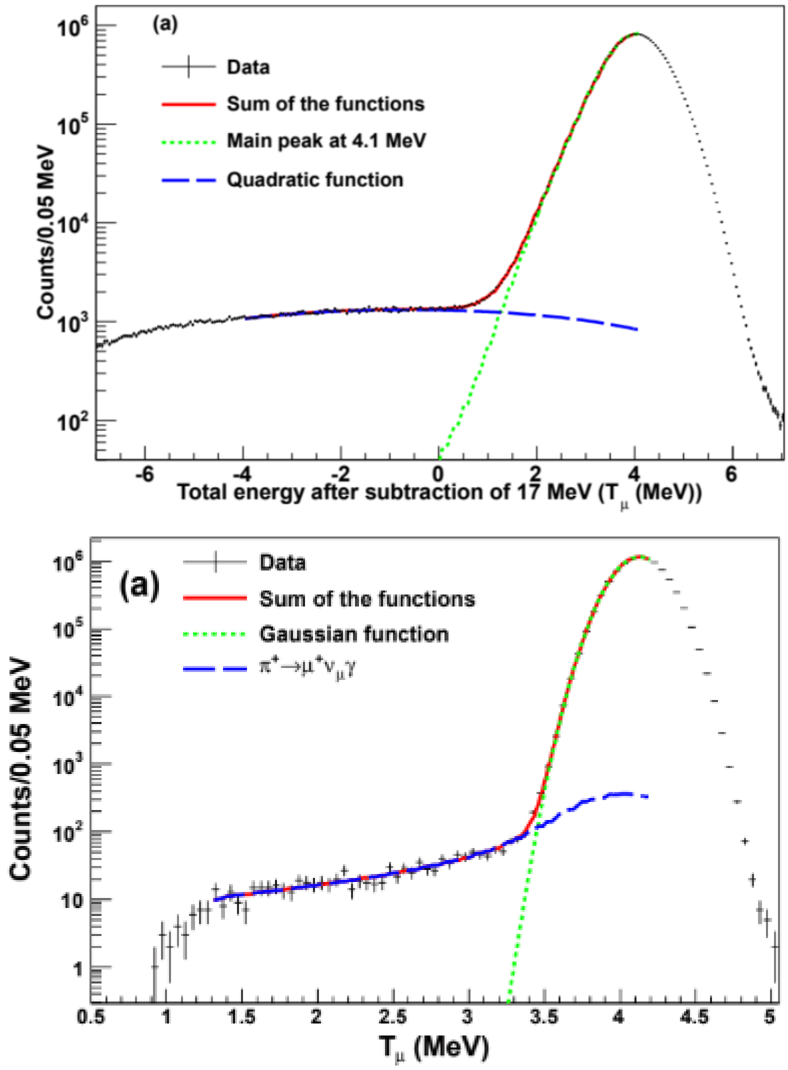
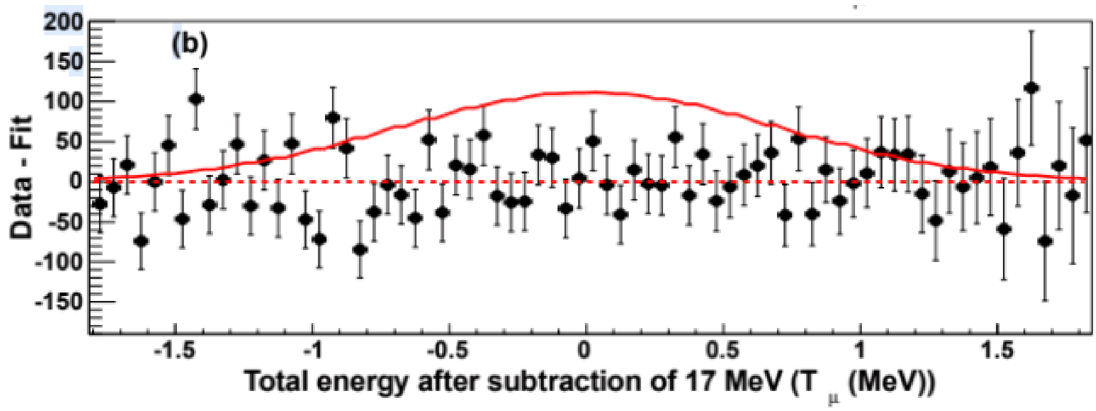


Figure 2.5: The figure represents similar features as Fig. 2.4. However, this model is for $\pi^+ \to l^+ vX$ which is the three body decay to axion (χ). The top figure is also affected by the smeared affect. [3]



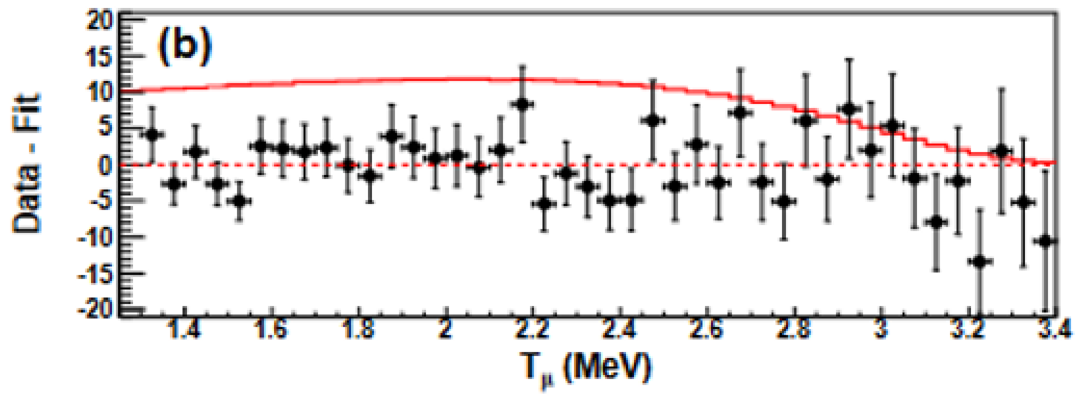


Figure 2.6: These are data fits done for Fig. 2.5 in respective order. These are useful to cross check and compare my data fit results with theirs. The x-axis is the kinetic energy of the muon in units of MeV. The subtraction of 17 MeV as mentioned is to sustain the energy at rest of pion. [3]

Chapter 3

My Work

From using the mentioned figures and graphs, I began my first significant assignment in replicating the graph of Fig. 2.3 using Python. The Gaussian equation was given by:

$$\delta(E - E_0) = Ne^{\frac{-(E-E_0)^2}{2\sigma^2}} \tag{3.1}$$

σ is initially given to me by $\sigma = \frac{\sigma_0}{100}$.

As shown in Fig. 3.1, I've defined Eqn. 3.1.

There are 2 variables that I'll be changing which are E=E_{e^+} and σ. E_0 is also a variable but I will fix it to be around 40 MeV which will shift the Gaussian peak to 40 MeV in the x-axis and change the mass of m_π accordingly. Changing σ change the width of the Gaussian, and with the right N, will represent the number of particles.

In Fig. 3.2, I make 2 other functions that define E_{e^+} and m_N since those 2 variables depend on m_π, m_{e^+}, and m_N. Although these are functions, they are given and known constants. What we are measuring is the change in the mass of sterile neutrino

```python
def f(E_pos,m_neutrino,sigma_numerator):
    sig = (sigma_numerator/100) * E_pos_0(m_neutrino)

    return np.exp((-(E_pos - E_pos_0(m_neutrino))**2)/(2*(sig**2)))
```

Figure 3.1: This equation is the bases of my Gaussian plot. The most important variable is *sigma_numerator* because it changes the width of the Gaussian. This will be important because it also changes the area underneath the curve which represents the normalizing factor.

```python
#input neutrino, returns energy of positron
def E_pos_0(m_neutrino):
    return (m_pos**2 + m_pion**2 - m_neutrino**2)/(2*m_pion)

#input energy positron, returns mass of neutrino
def m_neutrino(E_pos):
    return np.sqrt(m_pion**2 + m_pos**2 -2*m_pion*E_pos)
```

Figure 3.2: There are 2 equations that I defined in Eqn. 1.2 and 1.3 respectively for E_{e^+} and m_N.

by changing the mass of the positron since both of these masses come from the mass of pion which is fixed at 139.57 MeV. Also, keep in mind that the mass is equivalent to energy in this entire research.

Finally, I create a function to plot the Gaussian using these data points as shown in Fig. 3.3. The main function is plt.plot which graphs the Gaussian based on the parameters. The x-axis represents E_{e^+} and the y-axis goes up to 1, and it will require normalizing factor to increase the height. The area underneath will represent the number of positrons. The left of Fig. 3.4 shows the function call of Gaussian where E_{e^+}=40 MeV, m_N=91.18 MeV, and σ_0=1.5. But, as you can see it's too skinny compared to the Fig. 2.3. This means that our σ_0 is too small, and has to be a larger number.

```python
#graph with energy of positron
def graph_E_pos_input(E_pos_input,sigma_numerator):
    plt.xlim([4,60])
    #define mass of neutrino depending on energy of positron inputed
    neutrino_mass = m_neutrino(E_pos_input)

    plt.plot(E_pos,f(E_pos,neutrino_mass,sigma_numerator))
    plt.axhline(color = 'black')
    plt.fill_between(E_pos,f(E_pos,neutrino_mass,sigma_numerator),
                     where = [(E_pos>0) and (E_pos<100) for E_pos in E_pos], color = 'green' , alpha = 0.3)
    plt.xlabel('Positron Energy (MeV)', fontsize=16)
    plt.ylabel('Events', fontsize=16)
    plt.title(f'Energy of Positron = {E_pos_input} MeV', fontsize=16)
```

Figure 3.3: This is the code for plotting the Gaussian to replicate the red plot of Fig. 2.3.

The figure on the right of Fig. 3.4 shows how the Gaussian changes with different σ_0. The figure on the left shows the graph with the default σ_0=1.5 and the right shows the variations of σ_0: 2.5, 5, and 10. As you can see, the width increases as σ_0 increases. The lack of number of elements in the E_{e^+} array makes the graph look jagged, but the only characteristics of this plot that we're looking at is the base of the graph in the x-axis. This determines the width of the Gaussian and can be used as a reference point to compare with Fig. 2.3. Fig. 2.3 has the detection from 30 to 42 MeV which can be identified from the beginning to the end of the Gaussian peak. This means that I have to find σ_0 that will give me a Gaussian that has similar width on the base.

The fundamental problem is that Fig. 2.3 is not a perfect Gaussian, hence not a Gaussian at all. We use Gaussian simply as an estimator of Fig. 2.3 but is not exactly the same. The only accuracy that we can come up with is that the rise and the fall of the peak has the same width, in this case the width of 12 MeV. There were few different ways to determine the most accurate σ_0, but the most accurate number was from eyeballing and testing, which turned out to be $\sigma_0 = 5$ as shown in Fig. 3.5.

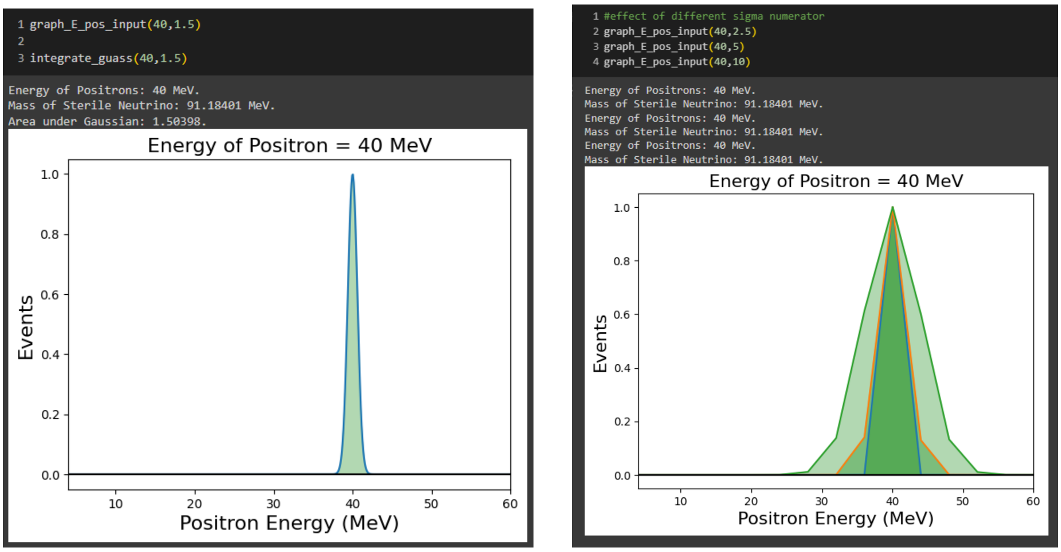


Figure 3.4: The figure on the left is the default modification of the Gaussian where $\sigma_0 = 1.5$. The figure on the right displays the change in the width of the Gaussian with different σ_0: 2.5, 5, and 10. It's rather pointy because of the lack of data points in the array. But, that is irrelevant because all we care about is the general look of the graph rather than the quality. Adding more elements, in this case E_{e^+} will only make the Gaussian more curved which is not very important..

The integration of this Gaussian tells you the number of particles in the experiment performed. To find the exact number, we had to find the normalizing factor and divide it on the equation. This is mostly done by my colleagues.

```
1 #sigma numerator = 5 to be width of 12 MeV
2 guassian_width(40,5,.5)
```

```
Energy of Positrons: 40 MeV.
Mass of Sterile Neutrino: 91.18401 MeV.
Left vertical line: x = 34.11294988742263
Right vertical line: x = 45.88705011257737
New width is 11.77410022515474 compared to speculated width: 17.
Area under Gaussian: 5.01326.
```

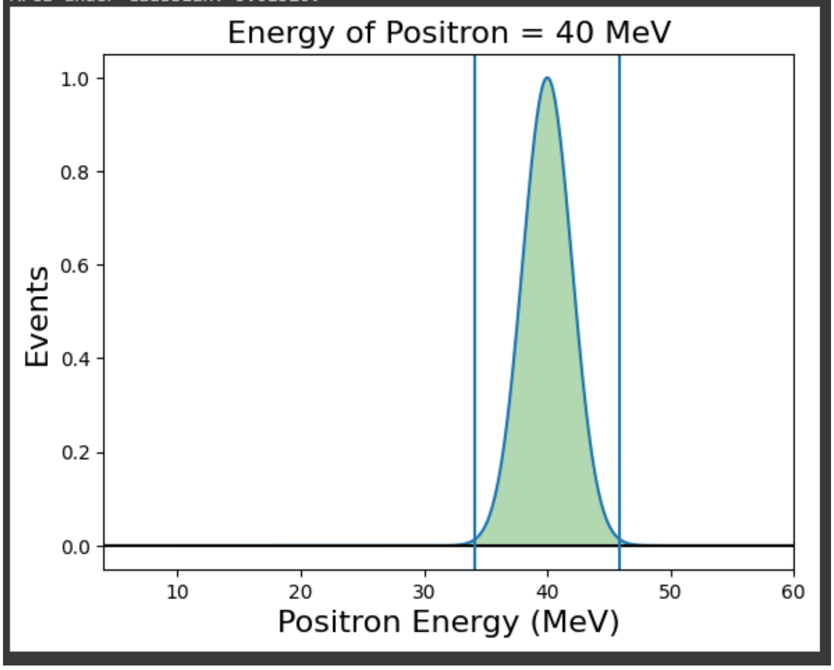


Figure 3.5: This shows the most accurate replicate of Fig. 2.3 The lines are used as a measurement to find the bottom edges of the plot to see the exact values in the base. Since Fig. 2.3 is not a symmetrical graph, it defers by half in comparison to this symmetric graph.

From here, we can use this graph as a basis for performing data fits and χ^2, which my research partners, Ollie Jackson and Khai Luong, did. From there, I switch from using Python to Mathematica. Again, I was given the graphs to digitize the plots into readable data so that I can replicate their graphs and this time, perform data fit myself using Mathematica program that Professor Altmannshofer wrote. Starting with Fig. 2.4, Professor Altmannshofer used Mathematica to extract only the black plot so that I can use plot digitizing app as shown in the left of Fig. 3.6. The data set is given in csv file which then can be used as data for the program that Altmannshofer wrote to perform data fitting.

The figure on the right of Fig. 3.6 shows the markings of the plot where the digitizing app records its coordinates into a csv file. This can be transferred to Mathematica template that Altmannshofer wrote for me to run the program to perform interpolation and data fitting. I did not save every image of the markings for the other graphs, but I've done the same procedure to extract/digitize data.

Once I got my data, I first plotted to check if it matched the graph as shown in Fig. 3.7. Each point is an x-y coordinate, x-axis being the MeV and the y-axis being the number of events. The y-axis is in log scale and all the data points seem accurate with Fig. 2.4, so my next step is to create another array which includes 3 parameters: x,y, and vertical error.

The vertical error for this plot is given to me by Professor Altmannshofer as $\sqrt{y}$. This is because the graph has the "smeared" effect, which gets rid of the crosses of uncertainty and makes the graph look more continuous. The process of making this

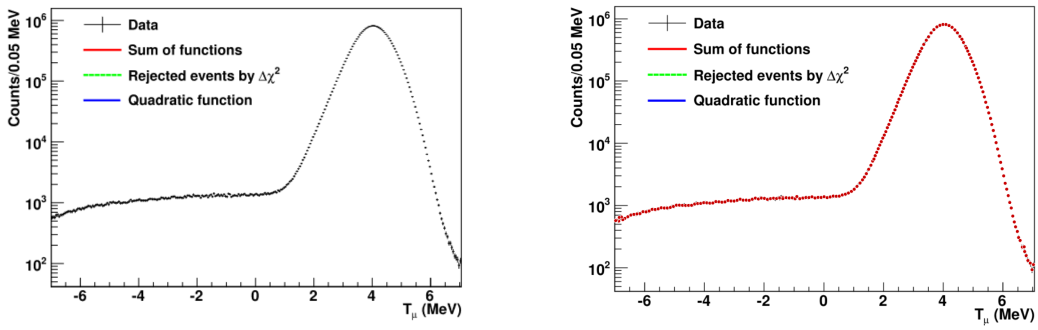


Figure 3.6: The figure on the left is the black plot of Fig. 2.4 extracted by Professor Altmannshofer using Mathematica. The figure on the right is the markings done manually using digitizing plot app. This can be transferred from csv file to programmable data in Mathematica to perform data fitting.

error array is shown in Fig. 3.8. In order to create another array of 3 parameters, I copy the x and y data points from the data made from digitizing app and add a 3rd parameter using $\sqrt{y}$ where each elements in this new array is a list of 3 elements: $\{\{x_1, y_1, \sqrt{y}_1\}, \{x_2, y_2, \sqrt{y}_2\}, ...\}$. One last thing I did before data fitting was to interpolate these data. This process is shown in Fig. 3.9. This makes the data points to have a continuous data between each point so that they include all domain.

Finally, in Fig. 3.10, I create a table which is basically an array where I subtract the signal by the background given as black plot - green plot - blue plot. Running that code, I get the final result shown in Fig. 3.11.

All of the following graphs follow very similar procedure from digitizing data to programming the data fit. The only difference is that for Fig. 3.12 and Fig. 3.13, I use $\frac{y_f - y_i}{2}$ as the z value instead of $\sqrt{y}$ since these figures show a vertical cross. This allows for me to extract their upper and lower ends of the uncertainty for the formula.

```
ListLogPlot[{red1904fig4, green1904fig4, blue1904fig4, black1904fig4}]
```

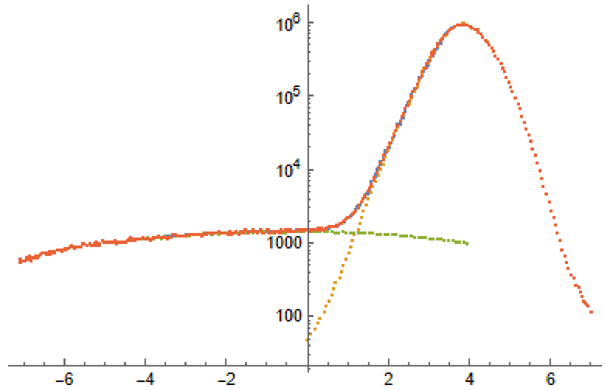


Figure 3.7: This figure is the replication of Fig. 2.4 from utilizing the digitizing app. This is plotted to check the scaling and proportions compared to the original plot so that all the results from this plot is accurate.

```
(*{x,y,sqrt(y)}*)
For[i = 1, i ≤ (Length[y1904fig4]), i += 1,
 pairs[[1]] = x1904fig4[[i]];
 pairs[[2]] = y1904fig4[[i]];
 pairs[[3]] = Sqrt[y1904fig4[[i]]];
 AppendTo[error1904fig4xyz, pairs]
]

error1904fig4xyz;
```

Figure 3.8: This code creates an array for data fit. It has 2 parameters which is x and y. The 3rd parameter accounts the vertical error bar which is $\sqrt{y}$.

```
(*Interpolating above data lists*)
(*xy*)
error1904fig4xylist = Interpolation[error1904fig4xy, InterpolationOrder → 1] // Quiet

(*blue*)
blue1904fig4list0 = Interpolation[blue1904fig4, InterpolationOrder → 1] // Quiet;
blue1904fig4list[x_] := If[-4 < x < 4, blue1904fig4list0[x], 0];

(*green*)
green1904fig4list0 = Interpolation[green1904fig4, InterpolationOrder → 1] // Quiet;
green1904fig4list[x_] := If[0 < x < 4, green1904fig4list0[x], 0];

(*red*)
red1904fig4list0 = Interpolation[red1904fig4, InterpolationOrder → 1] // Quiet;
red1904fig4list[x_] := If[-4 < x < 4, red1904fig4list0[x] - green1904fig4list[x] - blue1904fig4list[x], 0];
```

Figure 3.9: This code is showing the functions of interpolation. I interpolate all the data for data fitting. The red plot is subtracted by the backgrounds because the red plot is the ideal, theoretical plot given as the Gaussian of a signal plot.

```
(* define lists for the error bar plots, run sterile$residuals$list$Error and copy the list and name it error2101fig5xyerror *)
error1904fig4xyerror = Table[
{
{error1904fig4xyz[[ii, 1]], error1904fig4xyz[[ii, 2]] - green1904fig4list[error1904fig4xyz[[ii, 1]]] -
blue1904fig4list[error1904fig4xyz[[ii, 1]]]}
, ErrorBar[error1904fig4xyz[[ii, 3]]]
}
, {ii, 1, 305}
];
(*PIENU$data$list$error=sterile$residuals$list$Error=error1904fig4xyerror;*)
error1904fig4xyerror;
```

Figure 3.10: This code creates the array for the data fit which subtracts the experimented data with the background data.

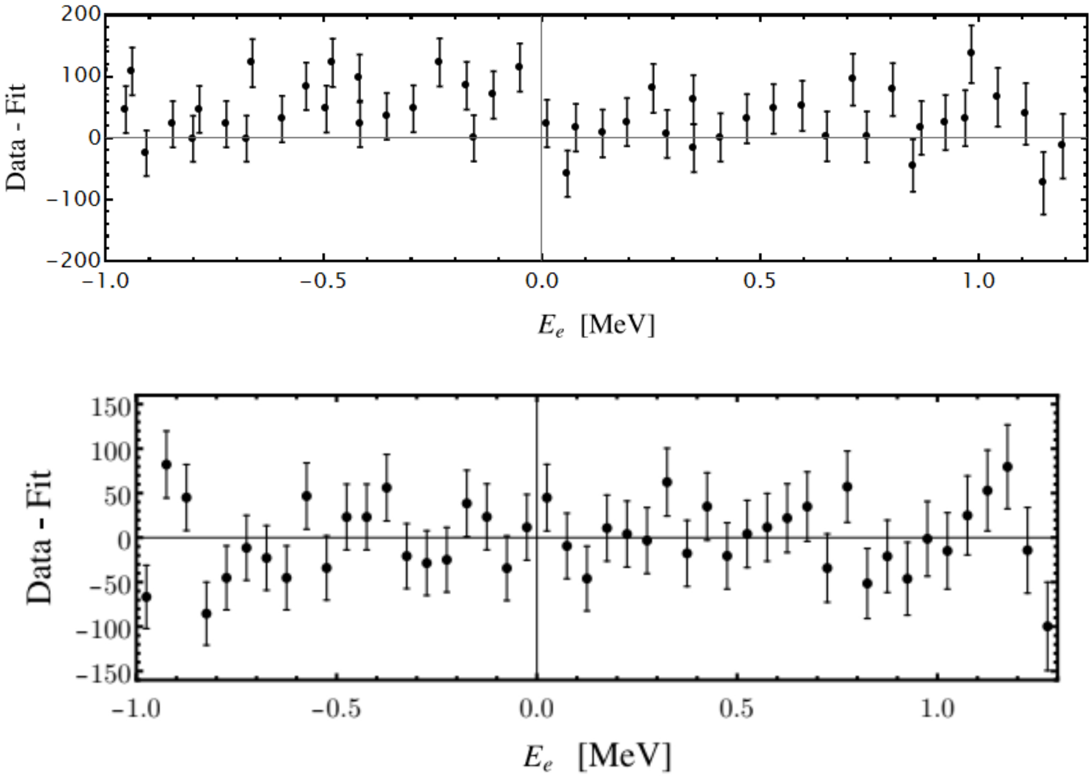


Figure 3.11: The figure on top is data fit made by me and the one below is made by Professor Altmannshofer. Although they are not exactly the same, especially between -.5 to 0 MeV, it's relatively similar. This difference is due to the method of digitizing where I manually clicked on the graph for each plot while Altmannshofer was able to code everything, making his more accurate.

The data fit for Fig. 2.9 was harder to produce because of the smeared effect. But since the paper we are using provides us with Fig. 2.9, Professor Altmannshofer can check with that data fit.

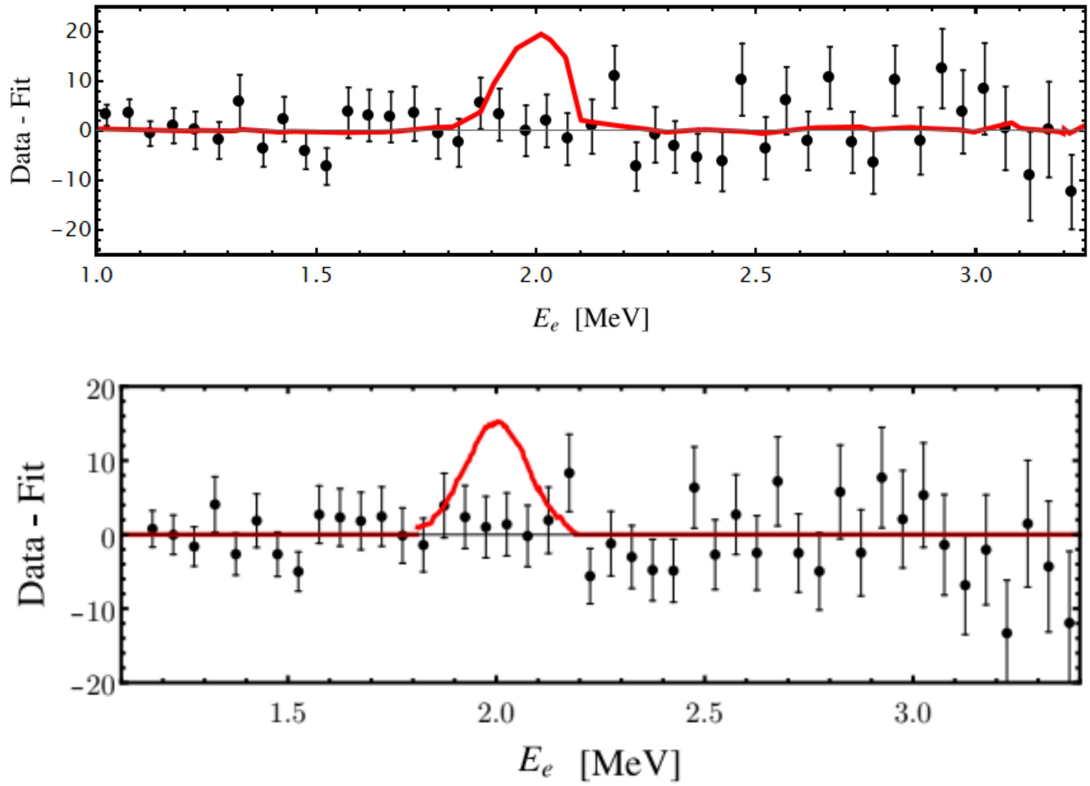


Figure 3.12: These are data fit for Fig. 2.5. The upper graph is created by me and the lower graph is created by Professor Altmannshofer. They are not exactly the same because of the different methods we used, but they are fairly similar.

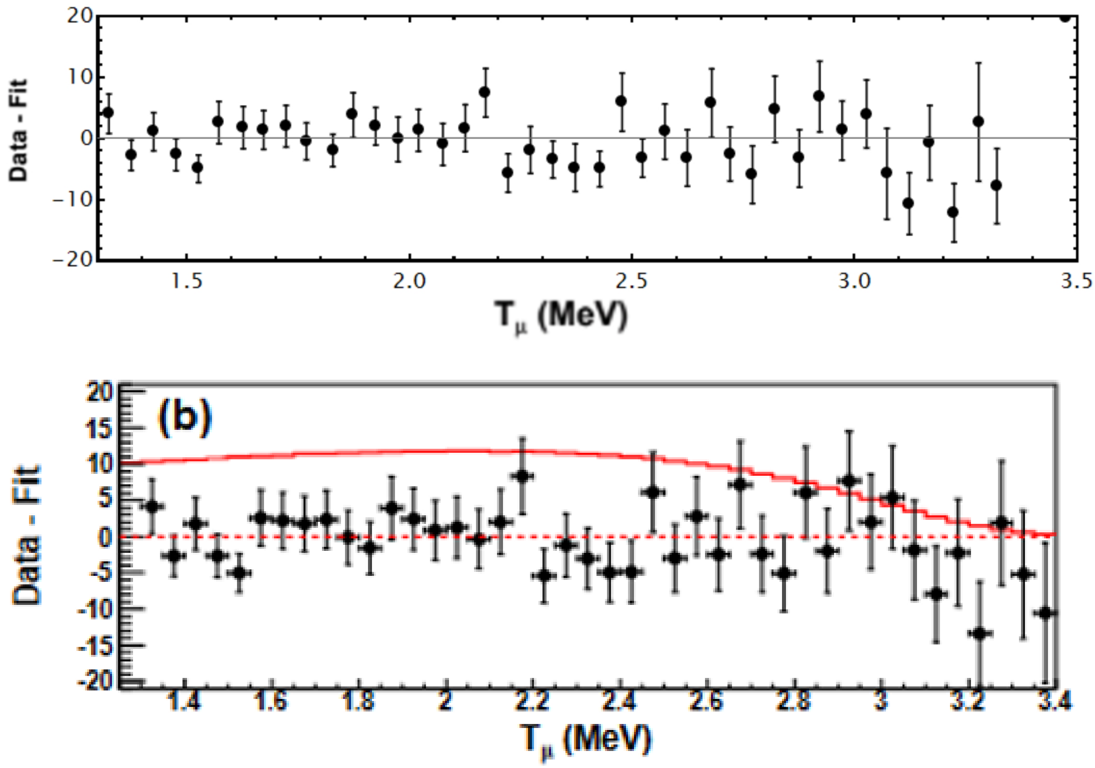


Figure 3.13: These are data fit for Fig. 2.6. The upper graph is created by me and the lower graph is created by Professor Altmannshofer.

The last assignment that I will go over is about recreating the red plot of Fig. 2.3, but it's for the $\pi \rightarrow l^+ v_l a$ decay. The l=lepton($e^+ or \mu$), v_l=neutrino, and a=axion. This will simulate the theoretical signal plot for axion decay. The method to plot this simulation is doing numerical integration using Python. All this means in my research:

Case 1 = Weak Violating: $g_{ll} = 1$, $g_{\bar{l}l} = 0$, and $g_{vl} = 0$

Case 2 = Weak Preserving: $g_{ll} = 1$, $g_{\bar{l}l} = 1$, and $g_{vl} = 0$

First looking at the equation derived by Professor Gori:

$$\begin{aligned} \frac{dB(P^+ \rightarrow l^+ v_l a)}{dE_l dE_a} &= \frac{B(P^+ \rightarrow l'^+ v_{l'})}{(1 - m_{l'}^2/m_p^2)^2} \frac{1}{\pi^2} \\ \times \Bigg[\frac{(g_{ll} - \bar{g}_{ll} + g_{vl})^2}{16 m_{l'}^2 m_l^2} &\left(2(m_p - 2E_l)E_a - (m_p - 2E_l)^2 - m_a^2 + m_l^2\right) \\ + \frac{g_{ll}^2 m_a^2}{2m_{l'}^2} &\frac{m_p^2 - 3m_p(E_a + E_l) + 2(E_a + E_l)^2}{(m_p^2 - 2m_p(E_a + E_l) + m_l^2)^2} \\ + \frac{g_{ll}(\bar{g}_{ll} - g_{vl})}{4m_{l'}^2} &\frac{m_p^2 - m_a^2 + m_l^2 - 2m_p(2E_a + E_l) + 4E_a(E_a + E_l)}{m_p^2 - 2m_p(E_a + E_l) + m_l^2} \Bigg] [5] \end{aligned} \tag{3.2}$$

Eqn. 3.2 is referred as Eqn. S-1 because of its reference to the article, but I will refer to the equation as they are shown. There are 2 more equations S-2 and S-3 shown in Eqn. 3.3 and 3.4 respectively that are required to integrate Eqn. 3.2:

$$E_- < E_a < E_+, E_\pm = \frac{1}{2} \left[m_p - E_l \pm \sqrt{E_l^2 - m_l^2} + \frac{m_a^2(m_p - E_l \mp \sqrt{E_l^2 - m_l^2})}{m_p^2 + m_l^2 - 2m_p E_e} \right] \tag{3.3}$$

$$m_l < E_l < \frac{m_p}{2} \left(1 - \frac{m_a^2}{m_p^2} + \frac{m_l^2}{m_p^2} \right) \tag{3.4}$$

All of the definitions of the variables are known: P^+=pion as a particle, E_l=energy of lepton, E_a=energy of axion, E_e=energy of positron, m_a=mass of ax-

```python
#Case 1: g_ll = 1, g_llbar = 0, g_vl = 0
#Weak Violating
def integral_case1(m_a):
  g_ll = 1
  g_llbar = 0
  g_vl = 0

  #a and b -> S-3
  b = E_l_upper(m_a)
  a = 0.511
  E_l = np.linspace(a,b,100)
  #E_l[i]
  y = []

  for i in range(0,100):
    #integrate from E- to E+ with each E_l[i]
    E_plus = ((1/2)*(m_p-E_l[i] + np.sqrt(E_l[i]**2-m_l**2) + ((m_a**2*(m_p-E_l[i]-np.sqrt(E_l[i]**2-m_l**2)))/(m_p**2+m_l**2-2*m_p*E_l[i]))))
    E_minus = ((1/2)*(m_p-E_l[i] - np.sqrt(E_l[i]**2-m_l**2) + ((m_a**2*(m_p+E_l[i]-np.sqrt(E_l[i]**2-m_l**2)))/(m_p**2+m_l**2-2*m_p*E_l[i]))))

    #integrand_f = type function for quad(f,b,a), f must be a function, not numpy/list
    integrand_f = lambda x: s_1(g_ll, g_llbar, g_vl, m_a, x,E_l[i])

    #return y plots each for loop
    y_val, error = quad(integrand_f, E_minus, E_plus)

    y.append(y_val)
```

Figure 3.14: This figure shows the Python codes for integrating Eqn. 3.2 with using the boundary conditions of Eqn. 3.4 and E_a from Eqn. 3.3. This is a numerical integration, which means that it's not integrating the Eqn. 3.2 by solving for the anti-derivatives, but rather using numbers to plug into the *quad* integrating python function.

ion, m_l=mass of lepton, and $m_{l'} = m_\mu$=105.7. Using Eqn. 3.4 as the boundary of the integral and using Eqn. 3.3 to plug into Eqn. 3.2, I integrate Eqn. 3.2 in terms of E_l. I do this numerical, not analytically because the analytic integration results in complicated functions with all sorts of boundary conditions like natural logs and more fractions. The code for the numerical integration is shown in Fig. 3.14. Fig. 3.14 shows the weak violating case while Fig. 3.15 shows the weak preserving case.

The resulting graphs are shown in Fig. 3.15 and 3.16. The x-axis is in terms of the variable that we integrated with which is E_l MeV and this is the rate of the change of branching ratio in respect to E_a. This is a resembling graph of Fig. 2.3 where the graph is not a perfect Gaussian. The difference is that the pion decay is $\pi \to l^+ va$ where a=axion instead of $\pi \to e^+ N$ where N=sterile neutrino.

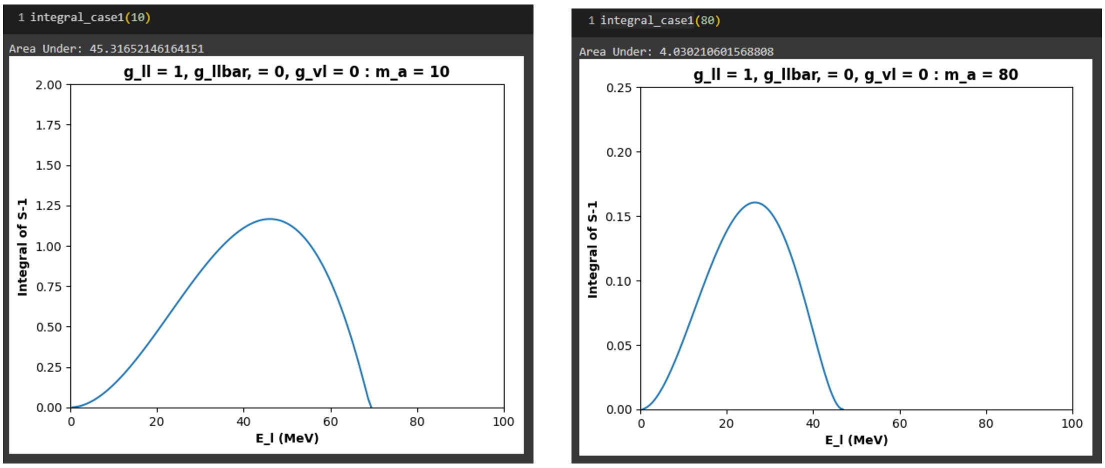


Figure 3.15: These two figures represent the theoretical plot of axion detection in $\pi \to l^+ va$. Since they are case 1, this is the weak violating case. The figure on the left is with m_a=10 MeV and the figure on the right is m_a=80 MeV. The x-axis is in E_l and the y-axis is the integral of Eqn. 3.2 in terms of E_a.

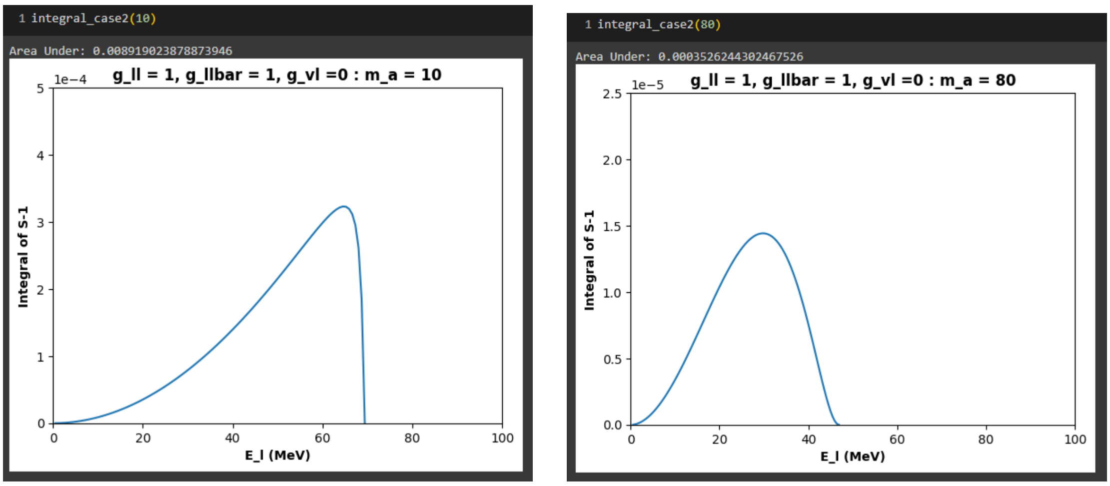


Figure 3.16: These two figures represent the same graph as Fig. 3.15 but with weak preserving case. The left shows the graph with m_a=10 MeV and the right figure is when m_a=80 MeV.

This concludes all of the significant assignments that I've done to contribute to the PIONEER collaboration. From using python to mathematica, I've done data fitting, digitizing graphs, integrating complicated equations, simulating particle detection, and many others. Like many groups, my work is but a small picture of a greater purpose. Next, I will go over how my work relates to works that my undergraduate colleagues have done in their parts of this research.

Chapter 4

Connection to Overall Research

My colleagues, Ollie Jackson and Khai Luong, performed the data fits and found the bounds of the pion decay models that were mentioned in this research. Although I've mostly worked with data fitting and digitizing graphs, most of the research is about finding the branching ratio of pion decay to sterile neutrino and axion. This also requires checking bounds to see the probability of detection such dark matters from the pion decay model that we theorize. One of those bound graphs are shown in Fig. 4.1. As the graph increases m_a, the more unlikely we'll see a dark matter.

Like before, we have to consider two cases: weak preserving and weak violating. The bounds, data fit, and branching ratios differ in each case by certain amount of order of magnitude. These variations are shown in Fig. 4.2 and Fig. 4.3. Like Fig. 4.1, as m_a increases, the graph increases which means the possibility of dark matter detection gets lower. Fig. 4.2 shows the bound on the left and the branching ratio on the right for weak preserving case. Fig. 4.3 shows the bound and branching ratio for the weak

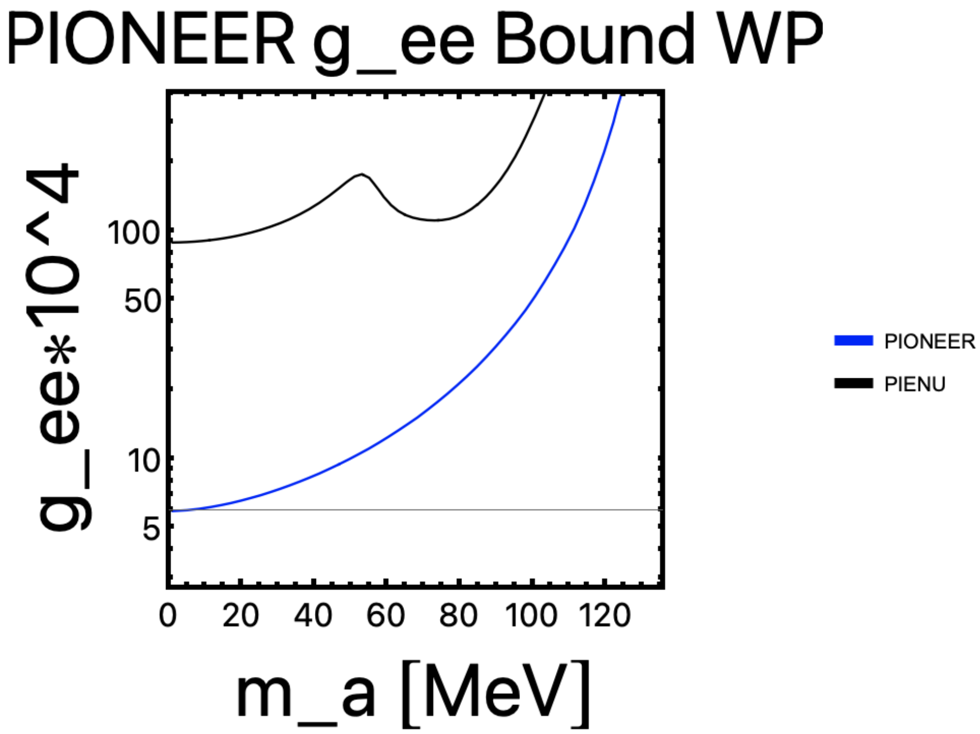


Figure 4.1: This figure shows the bounds of pion decay to positron created by Ollie Jackson. The x-axis represents the mass of the axion while the y-axis represents the $g_{ee} * 10^4$. The plot increases like an exponential function, and beyond around 100 MeV, it's impossible to decay as axion because of the limit due to conservation of energy.

violating case. They look somewhat similar, but the weak violating case scales much lower. The y-axis shows the factor of g_{ee} which is the same as g_{ll} mentioned before. This is a couple factor which is determined by the constraints of the experiment.

The coupling constraints have been analyzed by Pierce Giffin, who is a physics graduate student and my research partner, whose plot is shown in Fig. 4.4. This is just comparing 2 different couplings for 2 different cases. The weak preserving is greater than weak violating. He is also studying other decay models one of which is kaon decay to dark matter.

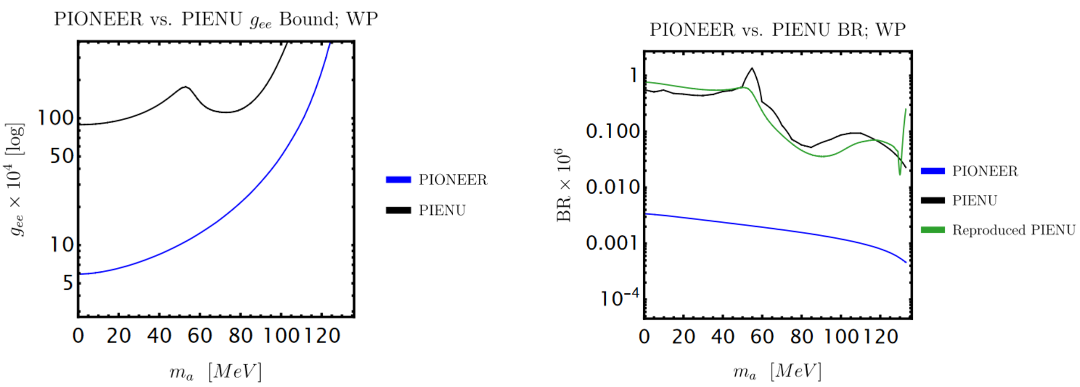


Figure 4.2: The figure on the left is the bound for weak preserving case. The y-axis is in g_{ee}=g_{ll}, where it's scaled to 10^4 to match the experimental constraints. The figure on the right is is the branching ratio of the weak preserving case. As the mass increases, the branching ratio decreases while the bound increases. This figure was made by my research partners, Ollie Jackson and Khai Luong.

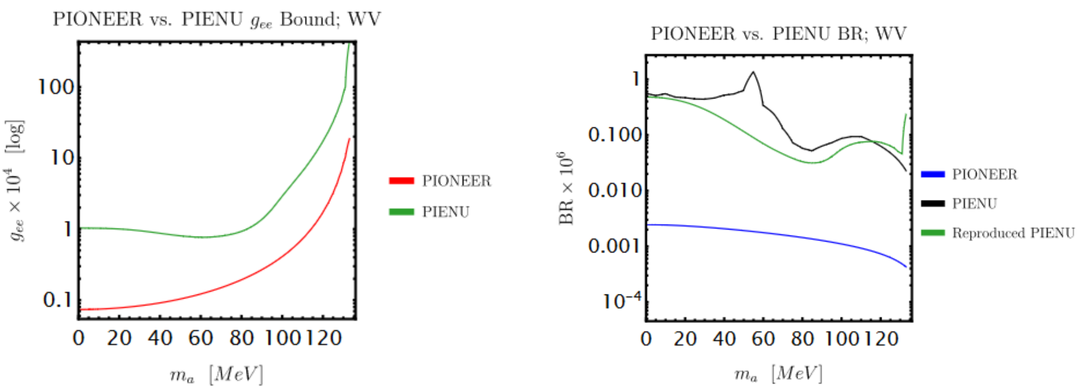


Figure 4.3: This figure is similar to Fig. 4.2 but this figure is for the weak violating case They share similar properties, but the scale is much lower.

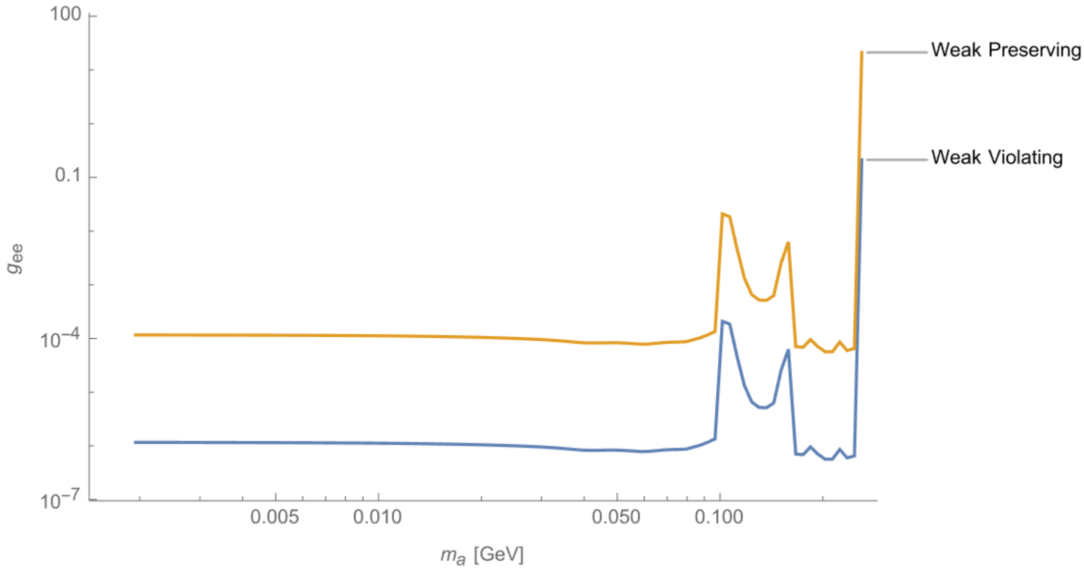


Figure 4.4: This figure shows the coupling factors for g_{ee} for weak violating and weak preserving cases. This graph was made by Pierce Giffin.

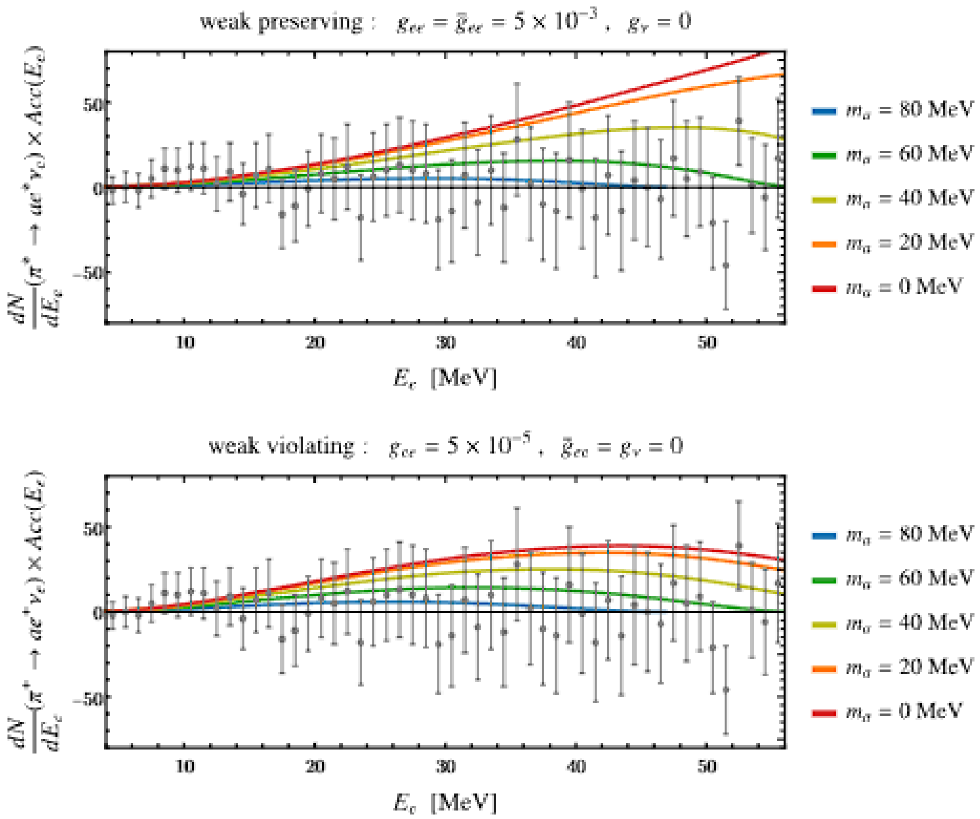


Figure 4.5: This plot is made by Ollie Jackson who studied not just the branching ratio, bounds, and data fits, but also plotted the number of particles detected in the pion decay model of $\pi \to e^+va$.

The last figure shown in Fig. 4.5 analyzes the number of particles detected in the experiment in terms of E_e^+. The upper figure shows the weak preserving case with coupling factor to be 5 x 10^{-3}, and the detection diverges as m_a increases above m_a=60 MeV. The vertical lines represent the uncertainties and the y-axis shows the number of particles under specific m_a. The lower figure shows the weak violating case with the coupling factor to be 5 x 10^{-5}. Unlike the weak preserving case, weak violating case converges even above m_a=80 MeV.

Chapter 5

Conclusion

In the end, my research is one of many parts needed to analyze the pion decay to dark matter from PIENU experiment. I've plotted the theoretical model of axion decay, recreated previously analyzed models, and performed data fits to cross check with my professor and the provided papers. By doing this, it allowed us to confirm the 3 decay model results and to create a procedure/model to perform similar methods to different dark matter studies.

I begin with recreating the $\pi \rightarrow e^+ N$ Gaussian curve using python, and collaborated with my research partners to find the normalizing factor. Then worked with Professor Altmannshofer to recreate and perform data fit on the 4 plots of PIENU experiment using Mathematica. These 4 plots were given by 2 papers which focuses on $\pi \rightarrow \mu v$ and $\pi \rightarrow e^+ va$. The conclusion of each paper was that there was no result that showed the detection of axion or sterile neutrino.

And lastly, I've also created the theoretical signal plot for the $\pi \rightarrow \mu a$ decay using Python. This was made by numerically integrating the S-1 equation using the bounds of S-2 and S-3 equations. Performing this analytically was difficult since it's a huge equation with boundary conditions from the result of a logarithmic terms.

There were few things that I've done like creating monte carlo simulation for PIONEER experiment which would happen later, plotting the S-1 equation and its double integrals, and any additional assignments that might be given since this is an on going research. From these assignments, I've learned to plot graphs, extract data, perform integrals through python, dabble into a bit of mathematica, and perform data fitting. This isn't the end of my assignments since the research is ongoing, but this concludes my contributions to this research over the course of 2 years.

Bibliography

[1] A. Aguilar-Arevalo et al. Improved search for heavy neutrinos in the decay $\pi \to e\nu$. *Phys. Rev. D*, 97(7):072012, 2018.

[2] A. Aguilar-Arevalo et al. Search for heavy neutrinos in $\pi \to \mu\nu$ decay. *Phys. Lett. B*, 798:134980, 2019.

[3] A. Aguilar-Arevalo et al. Search for three body pion decays $\pi^+ \to l^+ \nu X$. *Phys. Rev. D*, 103(5):052006, 2021.

[4] W. Altmannshofer et al. PIONEER: Studies of Rare Pion Decays. 3 2022.

[5] Wolfgang Altmannshofer, Jeff A. Dror, and Stefania Gori. New Opportunities for Detecting Axion-Lepton Interactions. *Phys. Rev. Lett.*, 130(24):241801, 2023.