Analyzing Muon Lifetime Data

INTRODUCTION:

Recall from last week that muons follow an exponential decay law, i.e. If we start with N_0 muons, the number of muons left at time t, is given by the formula

$$N = N_0 e^{t/\tau}$$

where τ is the typical lifetime of the muon.

The detail of the experiment is described elsewhere, but in essence we told the Quarknet cards to tell us whenever it saw a possible muon decay event, and the time delay between the muon pulse and the electron pulse.

Not all of these events can be genuine muon decays, so we need to use a standard tool in data analysis, which is to apply *cuts* to the raw data.

The First Cut:

The first cut has already been made by this point: we tell the Quarknet card to only record events that are coincident in two detectors, which we placed above one another. This reduces the amount of noise, which may be caused, say, by radioactivity in the scintillator.

The Second Cut:

We are looking for 'double' events, which would represent a muon entering the scintillator and decaying to produce an electron. Both the muon and the electron would produce light and a PMT pulse.

The Third Cut:

We expect all muons to be coming from above (from cosmic rays), so we only want to use events in which we see a pulse in the top scintillator, then a double pulse in the lower detector. (The other way round would indicate a particle coming from the ground, unlikely to be a muon.) Fortunately the Quarknet card tells us which channel registered the double, so we can simply select them.

The Fourth Cut:

The final cut is more arbitrary - we discard all events in which the double pulses arrived less than 160ns apart... We do this because most of the noise that survived cuts 1-3 lies in this region.

The Method:

We will use Mathematica to parse the text file produced by the Quarknet card. The format of that file is as follows

```
000DACE3 53 01 0004
001562FD 53 02 0004
00252DDF 53 02 004E
000F6901 53 02 0004
00039B1C 53 01 0004
001D054E 53 02 0005
002FC4C0 53 02 0004
00022253 53 02 0105
```

Column 1 is the time, in hexadecimal multiples of 160ns, since the last event. For example, the value '000DACE3' corresponds to 896227 multiples of 160ns, i.e. 143396μs.

Column 2 is two hexadecimal characters (one byte) representing the nature of the event. The byte is laid out like this

7	*6*	*5*	*4*	*3*	*2*	*1*	*0*
Not Used	Set if a double occurrs on any channel	Not used	Set if the trigger is satisfied	Set if channel 4 registere d an event	Set if channel 3 registere d an event	Set if channel 2 registere d an event	Set if channel 1 registere d an event

In this way we can interpret the second column. Our left-hand four bits read 0x5. The corresponding bits are 0101, which means that there was a double, and that this satisfied the trigger (which we programmed into the card when we started the experiment).

The second four bits read 0x3, i.e. 0011 in binary. This means that channel 1 and channel two registered pulses in this event.

Column 3 in our data tells us which channel registered the 'double'. In this case it is a mixture between channels 1 and 2... but we need to select only channel 2, the bottom scintillator.

Column 4 is the time in multiples of 20ns between the two pulses in a 'double'. Note that there are a total of three pulses in the events that we are looking for: One in the top scintillator, one caused by a muon in the lower scintillator, shortly followed by a pulse caused by an electron in the lower scintillaor. These last two make up the 'double'.

Mathematica:

First we read in the data from a file, typically saved as something like "CAPTURE_081702_1837.TXT". Doug wrote a little program to read in our data and convert from hexadecimal, etc. It is in the notebook "muon_data_analysis.nb". The three functions are called ReadMuonFile, FromHexString and PlotDecayTime. A brief description of each is included in the notebook. Only the function PlotDecayTime will be called by you, the others are 'subfunctions' used by PlotDecayTime.

Before you can use them, the functions need to evaluated by doing shift-enter in each of the cells (Mathematica might ask you if you want to evaluate all initialiazation statements, which is the equivalent of going to each cell and typing shift-enter, so say 'yes').

At the bottom of the file is an example usage. PlotDecayTime takes three arguments which are explained here.

The first argument is the name of the text file holding the four column data.

The second is a list with three parts {t_min, t_max, dt}. These tell Mathematica how to plot the data. Data with delta-T less than t_min will be discarded (this is our fourth cut.) Also data with delta-T greater than t_max will be discarded (not a cut, try playing with it, does it make much difference to change this in a certain range?). dt is the bin size for the histogram. Make it too small, and there will be mostly bins with 0 events, make it too large and we lose information. Times are in units of µs.

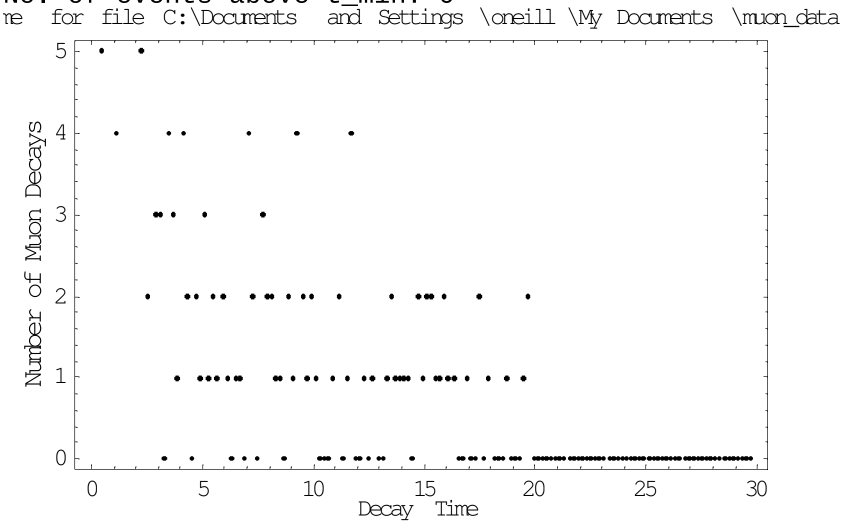
The third argument should be the channel number of the lower detector... this is the one that should be registering doubles.

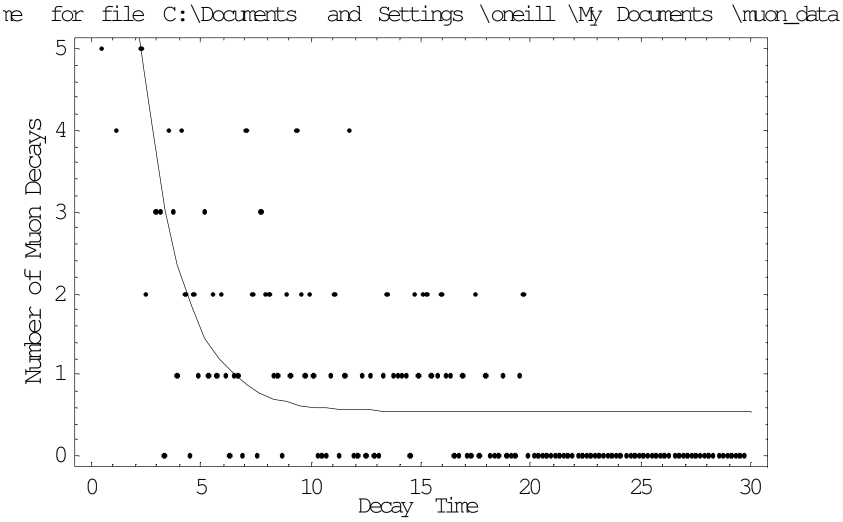
My example command was

```
mydatafile="C:\Documents and Settings\oneill\My
Documents\muondata.dat";
PlotDecayTime[mydatafile, {0.16,30,0.2}, 2]
```

Evaluating this cell gave me

```
No. of events above t_min: 0
```






```
Muon lifetime from data = 1.83259 usec
```

This was using a very small data set. We'll have to imagine that the data has an exponential shape than this case. Hopefully your data will be better.