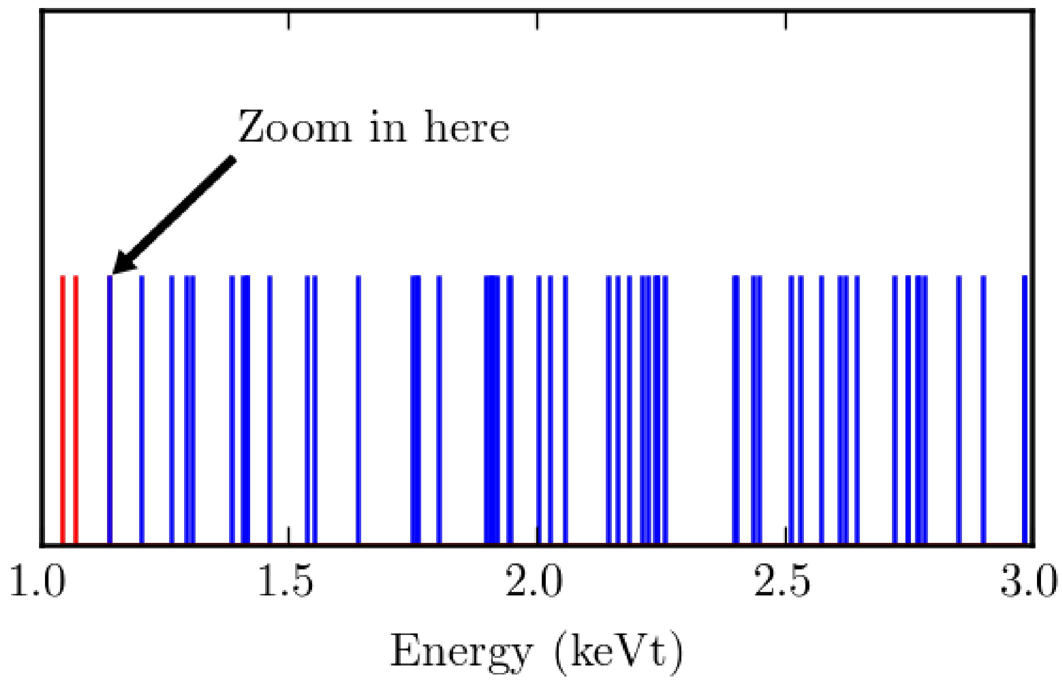
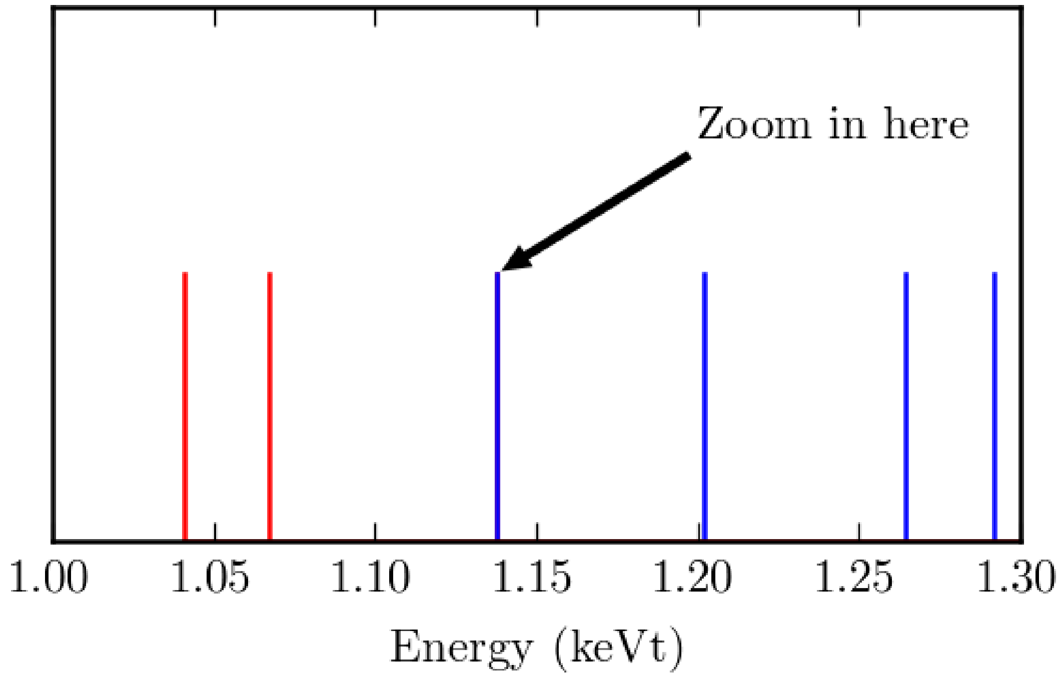
# CDMSlite Run 2a trigger turnon

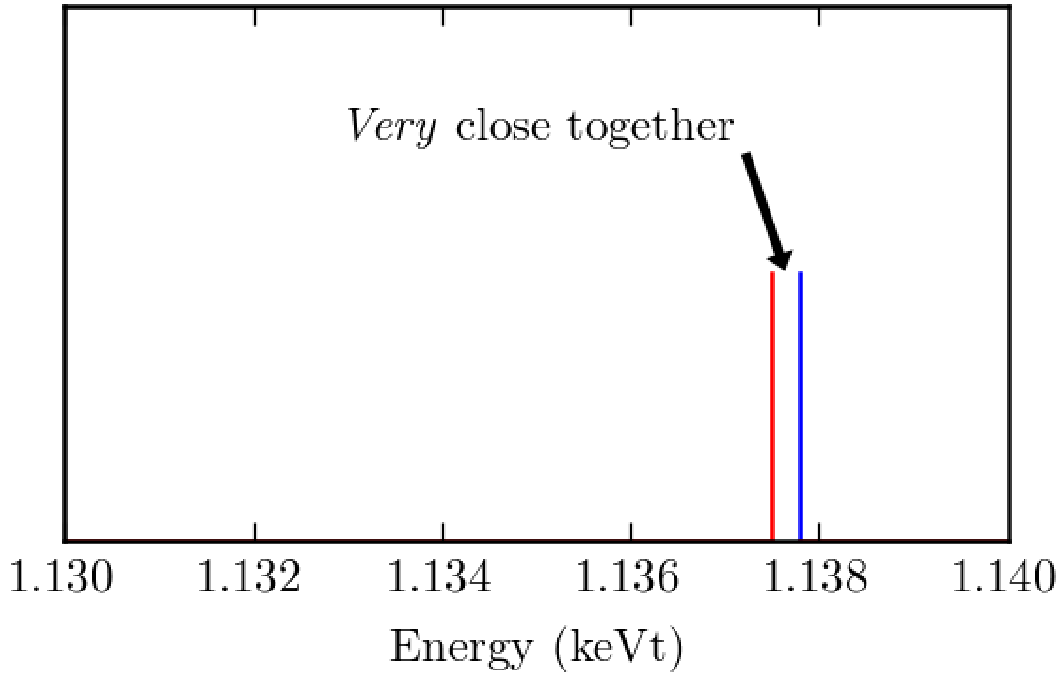
Jon Wilson, July 30, 2015

# The problem

We have a set of events that we are using to estimate the trigger turn on. Some of them pass the trigger requirements, and some of them fail. In Run 2a, there are only three events that fail, and they are all at lower energy than all of the events that pass. Worse, the highest failing and lowest passing events are very close to one another in energy. Red marks events that failed, and blue marks events that pass.




Zoom in here
1.00
1.05
1.10
1.15
1.20
1.25
1.30
Energy (keVt)



*Very* close together
1.130
1.132
1.134
1.136
1.138
1.140
Energy (keVt)

# How we estimate trigger turn-on

Previous estimates have used a likelihood approach. For each event, assign a probability for it to pass. Make that probability a sigmoid function (`erf`) of the energy, with two parameters, $\mu$ and $\sigma$, the center of the turnon and the width of the turnon, respectively. Then, since you know whether each event passed or failed, you have a likelihood that you can scan in $\mu$ and $\sigma$.

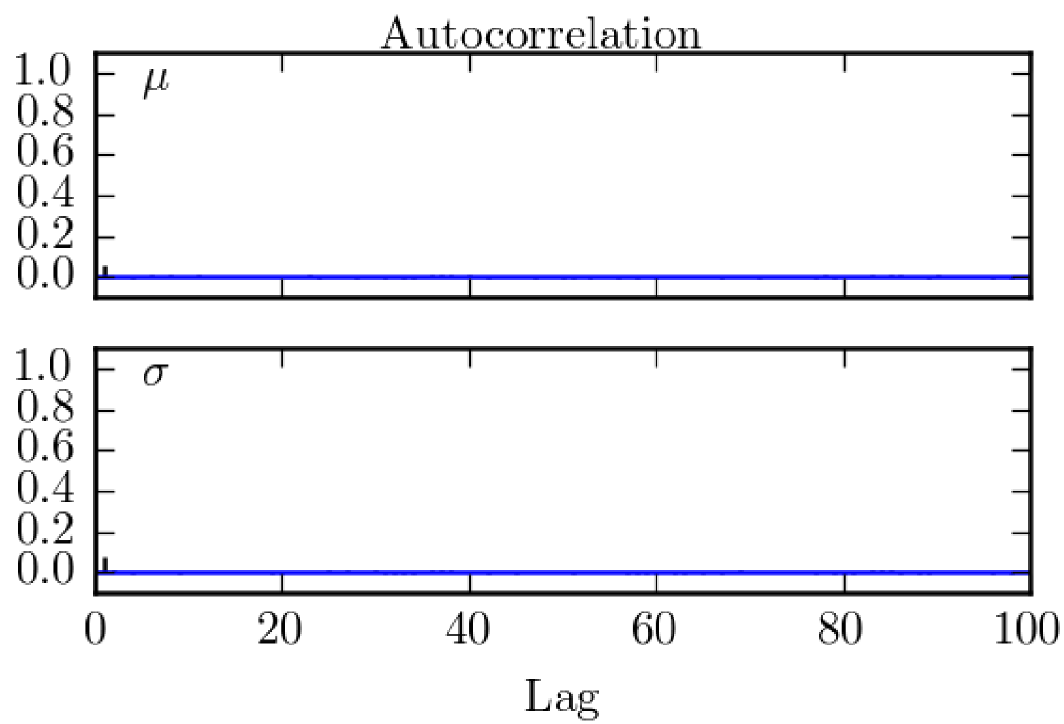
Because there is no overlap between passing and failing events, and the gap between passing and failing is tiny, the likelihood is very sharply peaked, and very strongly prefers miniscule $\sigma$ (on the order of $10^{-12}$ keVt) within a very narrow $\mu$ range.
Such a tiny $\sigma$ is unphysical. The trigger turn-on must have a width on the same order as the energy resolution. So let's put a prior and constrain it.

# How we estimate trigger turn-on, part 2

Now we build a Bayesian model. The likelihood is essentially the same, except now we have a log-normal prior on σ. This prior peaks at 0.172 keVt (best fit σ from R2b), and I gave it a width of 1.5 (unitless -- this is log-normal, so "1 standard deviation" means going up or down by a factor of 1.5).

This keeps σ from getting too small (or too large).

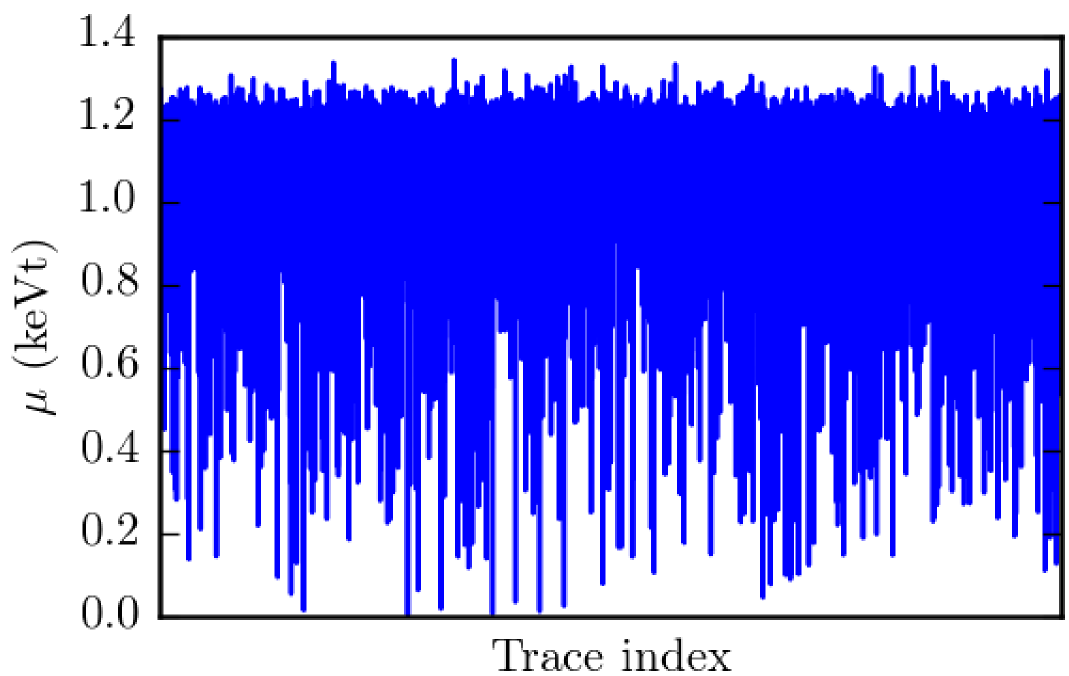
Then, we sample from the posterior using MCMC (the PyMC library, if anybody is interested in the details). Before we can look at the results, we should check to make sure that the sampler converged reasonably. First check is the autocorrelation in the trace. It should fall to zero very rapidly as the lag increases, which it does for both μ and σ.



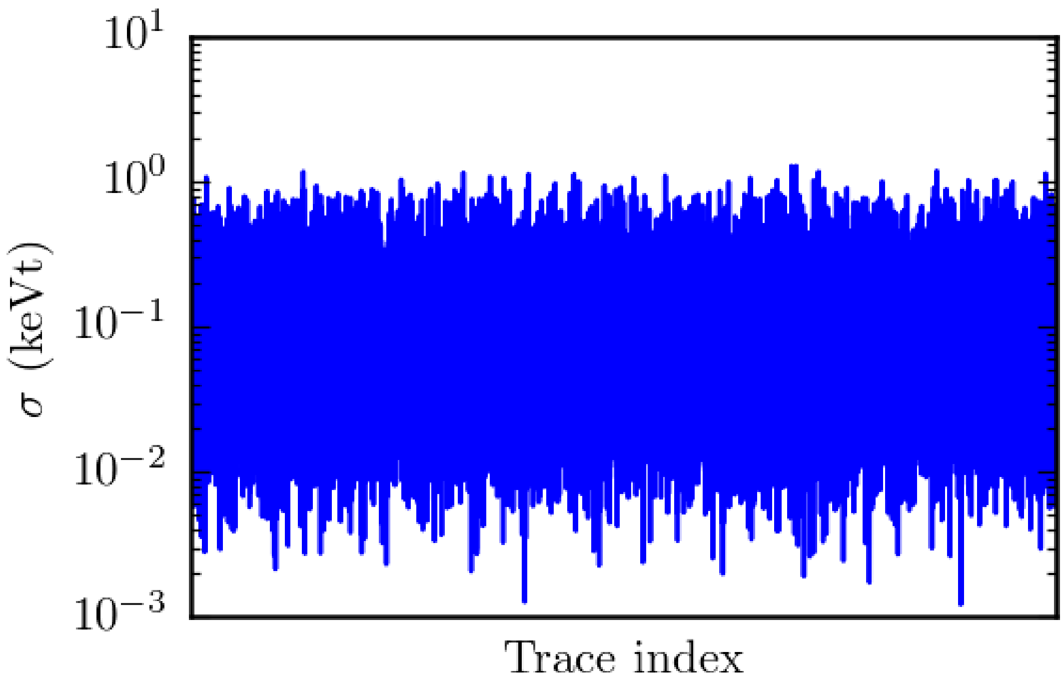
# More checking for convergence

Let's also check the MCMC trace for μ and σ directly. We should see no particular structure as a function of trace index (x axis).

Sure enough, we don't see any problems in the trace for μ.



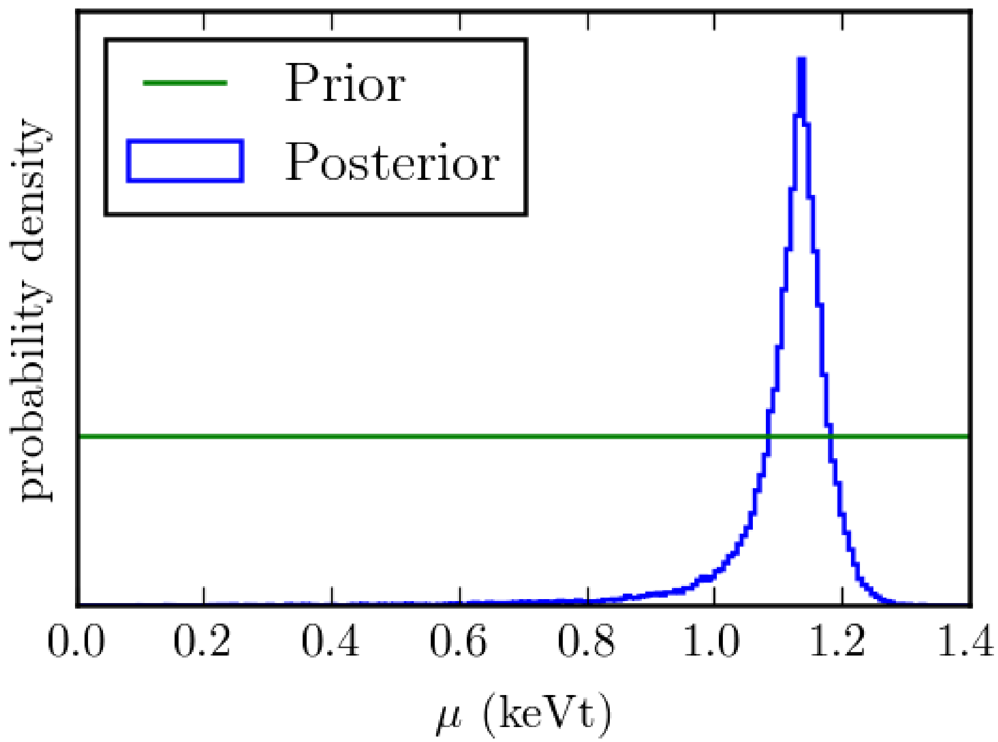
Nor do we see any structure in the trace for σ. Now we have some confidence that the MCMC sampling worked adequately, so let's start to look at the results.



# Marginalized posterior for $\mu$

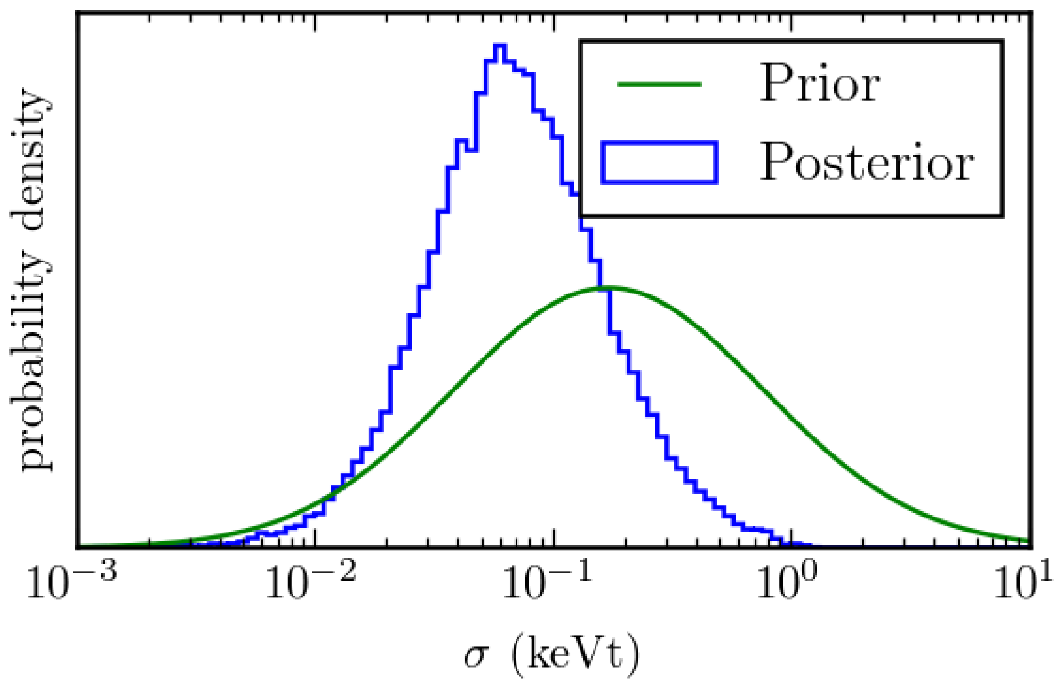
Here is the prior (flat) and marginalized posterior for $\mu$. It is pretty sharply peaked at the energy of the boundary between failing and passing events. This is expected. Has a longer tail on the low energy side than on the high energy side.

```
Out[14]:  <matplotlib.legend.Legend at 0x7f96e4efccd0>
```



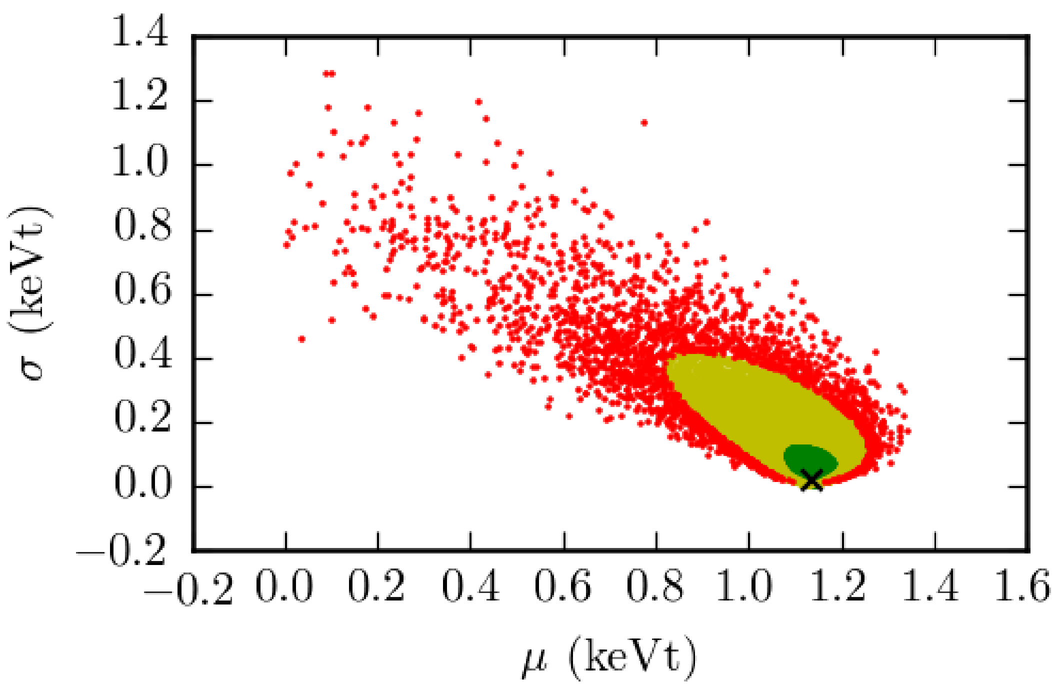
# Marginalized posterior for σ

Now we see the same for σ. Here you can see the log-normal prior. The posterior does still prefer smaller σ, but we no longer have excursions downward by many orders of magnitude to unphysical values. I think this is a success.



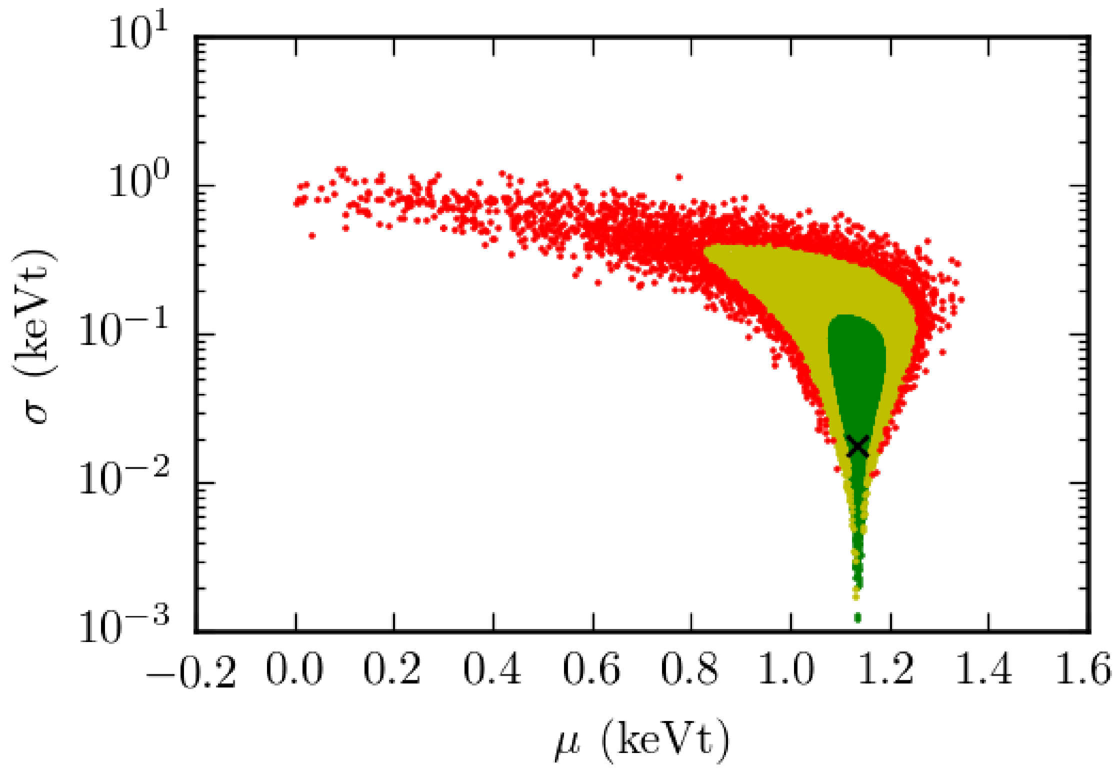
# Two-dimensional posterior

Now we see the joint posterior in μ and σ together. The green points are within the 68% credible region, the yellow and green are within the 95% credible region, and the remaining points are red. The credible regions are the "highest posterior density" definition for credible regions. The black "X" marks the maximum *a posteriori* point.



In log scale, we still see the old "funnel effect", where the range of μ narrows fast as σ decreases. But we no longer get down to very tiny σ, and the 68% and 95% credible regions certainly contain a lot of points with reasonable σ.

```
Out[19]: <matplotlib.text.Text at 0x7f96e526a710>
```



# Conclusion

This isn't a drastic change from Mark's likelihood scan, which is good. It's probably best to avoid unphysical regions, but at least it won't make much a difference if we switch to this or not.

I also have MCMC samples from Run 2b, but there isn't anything special there -- it pretty much looks just like Mark's result, as you'd expect, so I haven't presented it. If we want to use the MCMC samples instead of the likelihood scan, I'm happy to provide samples ($\mu$ and $\sigma$) from either or both.

One other possibility to deal with the unphysical values: instead of just putting a prior on $\sigma$, we could incorporate the uncertainty in the energy of the events. This idea is half-baked at the moment, but there doesn't seem to be any good reason it shouldn't work. That would probably be the best-motivated way to deal with this, but it also probably won't change the answer noticeably, so maybe isn't worth the time to do it.