Inferring the ACIS sub-pixel grade distribution

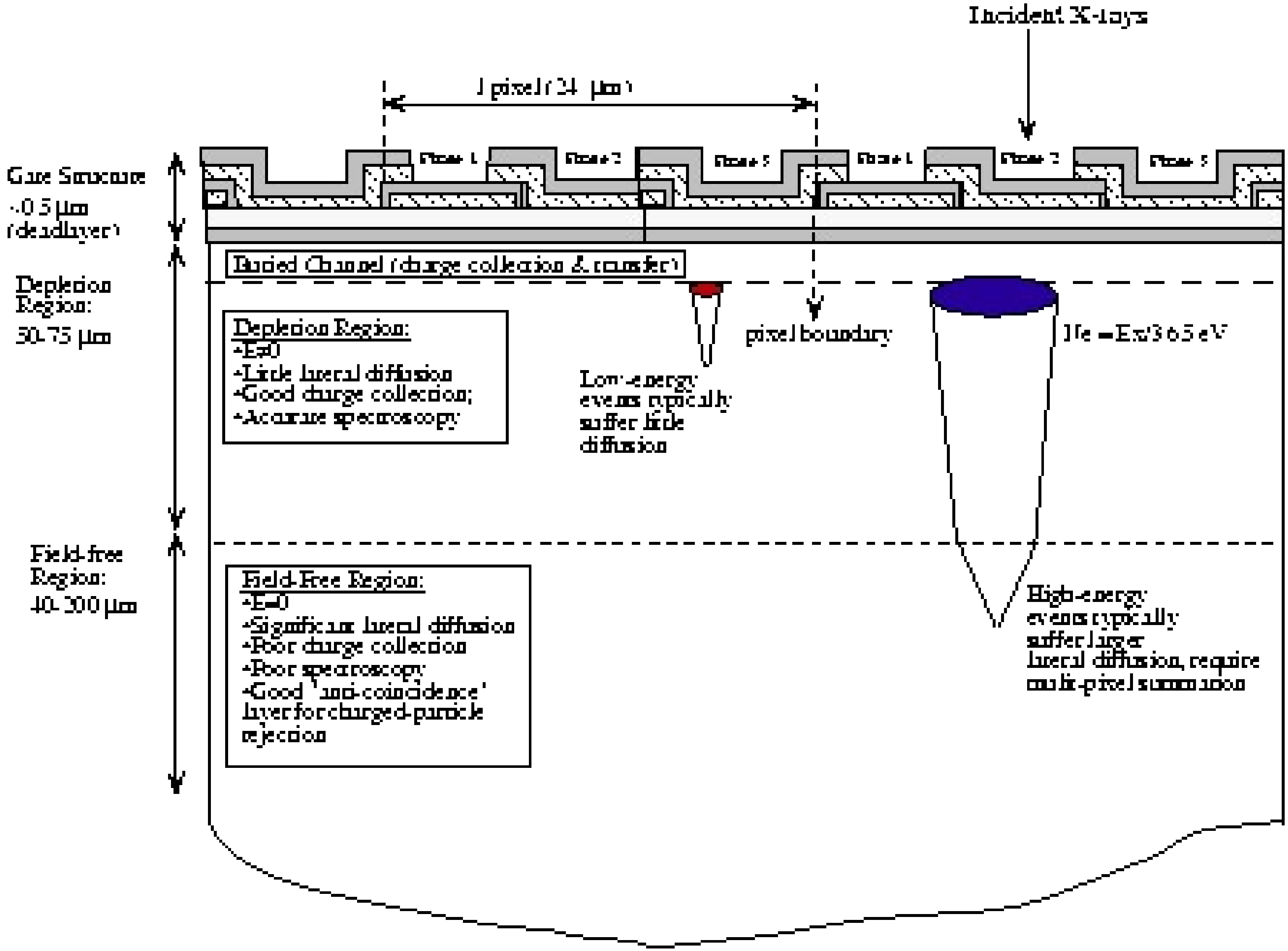
Hans Moritz Günther
MIT
hgunther@mit.edu

Outline

- **How CCDs work**
- **What is a "grade"**
- **Why do I want to know the sub-pixel grade distribution?**
- **Three ways to determine the grade distribution**
- **Other potential applications**
- **Fitting the grade distribution**
  - This is the part where I would like advice.

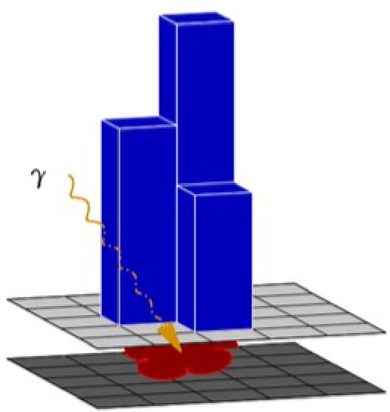
How CCDs work

- **For mathematicians:**
  - We need approximations in the process we model.
  - I want to convince you that we need to determine things from observed data, because we can't from the CCD specs.
- **For X-ray astronomers: I hope you know all that.**

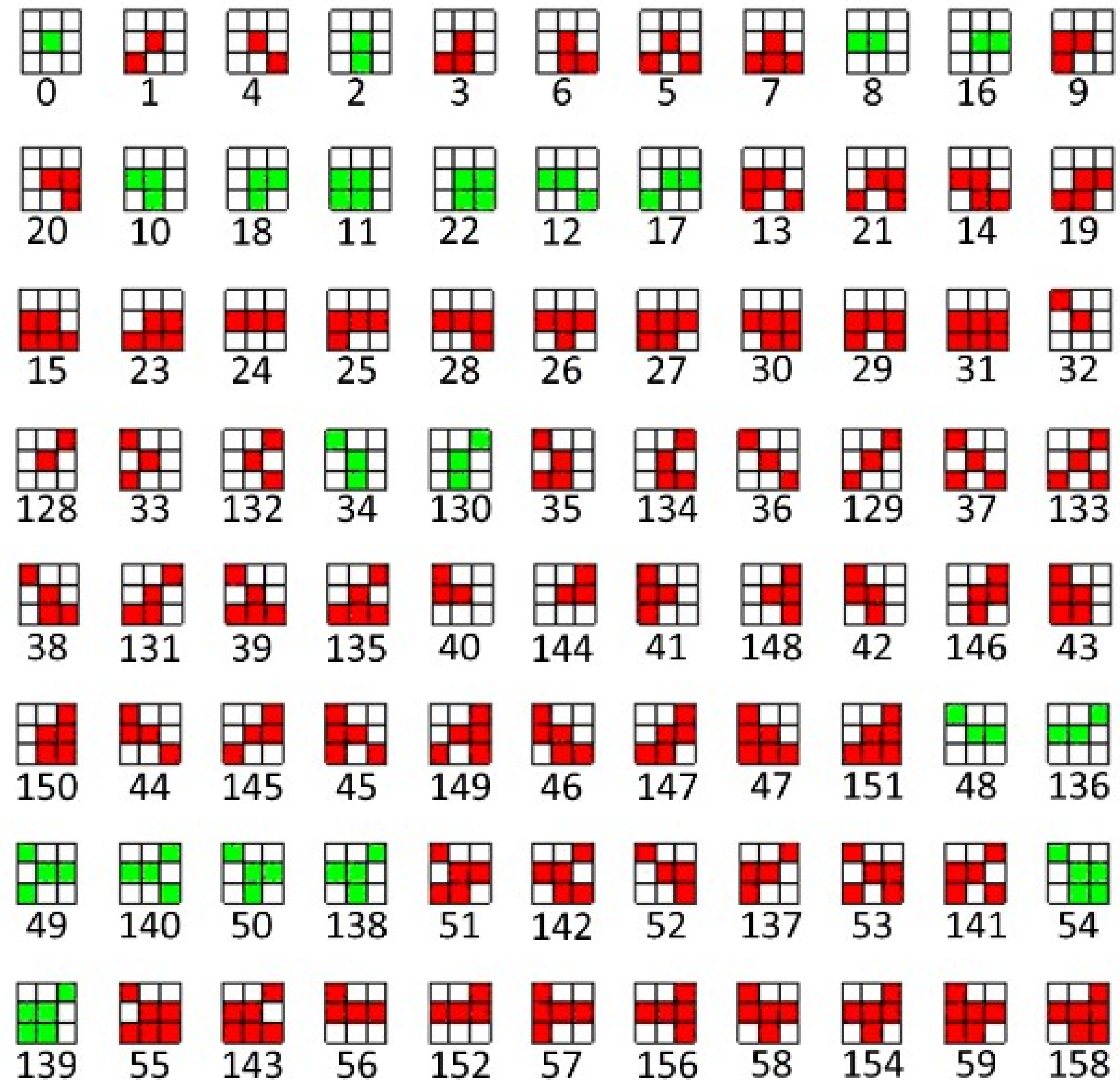


Incident X-rays
1 pixel (24 µm)
Gate Structure
~0.5 µm
(deadlayer)
Buried Channel (charge collection & transfer)
Depletion
Region:
30-75 µm
Depletion Region:
•E≠0
•Little lateral diffusion
•Good charge collection;
•Accurate spectroscopy
Low-energy
events typically
suffer little
diffusion
pixel boundary
Ne = Ex/3.65 eV
Field-free
Region:
40-300 µm
Field-Free Region:
•E=0
•Significant lateral diffusion
•Poor charge collection
•Poor spectroscopy
•Good 'anti-coincidence'
layer for charged-particle
rejection
High-energy
events typically
suffer larger
lateral diffusion, require
multi-pixel summation

What is an event “grade”?

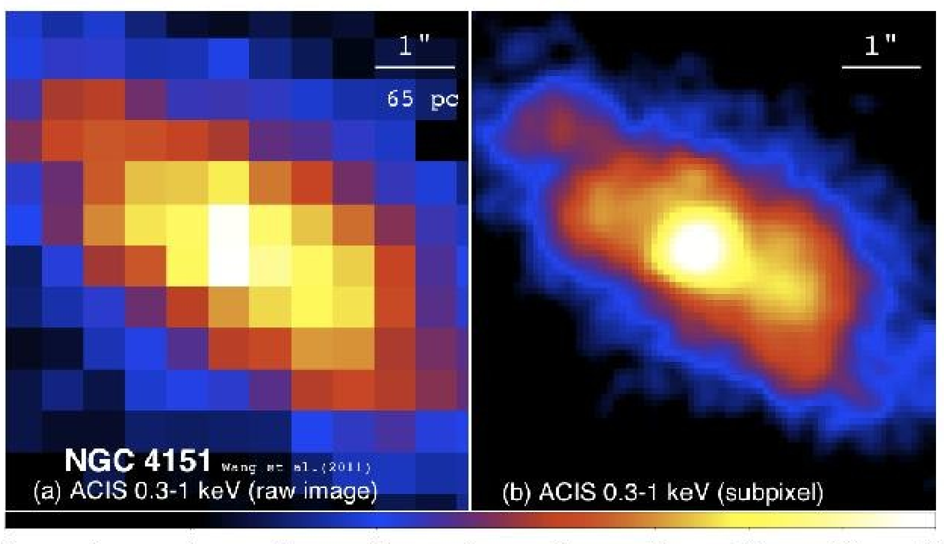


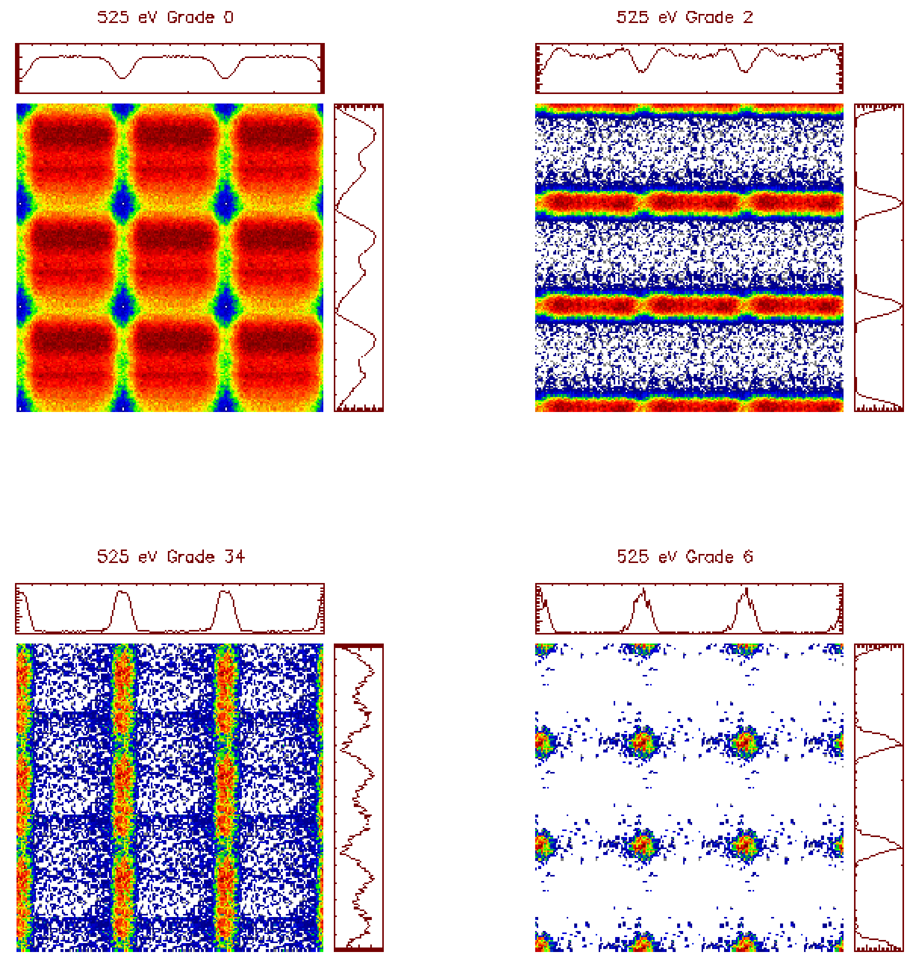
32	64	128
8	0	16
1	2	4



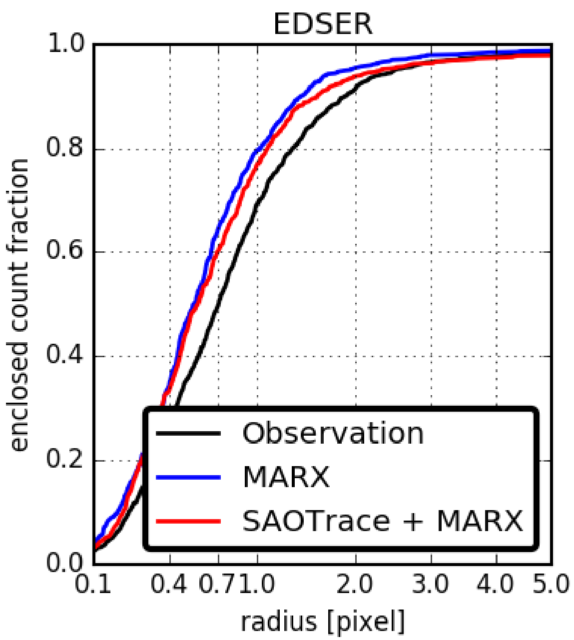
Why do I care about the grade distribution?

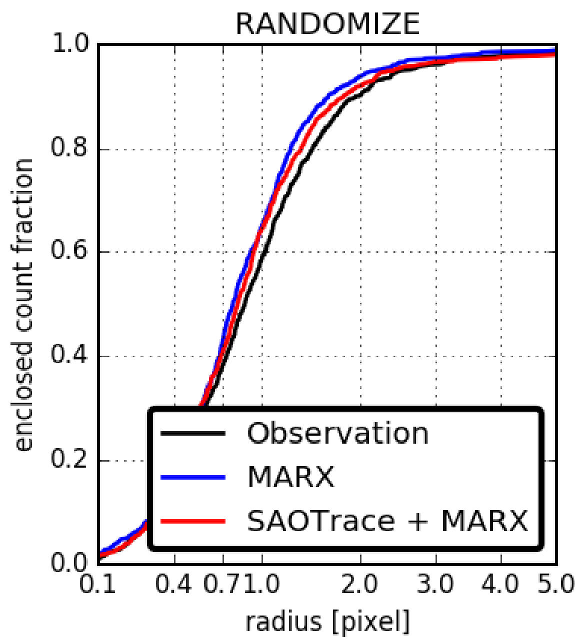
- **Energy depend sub-pixel event repositioning**

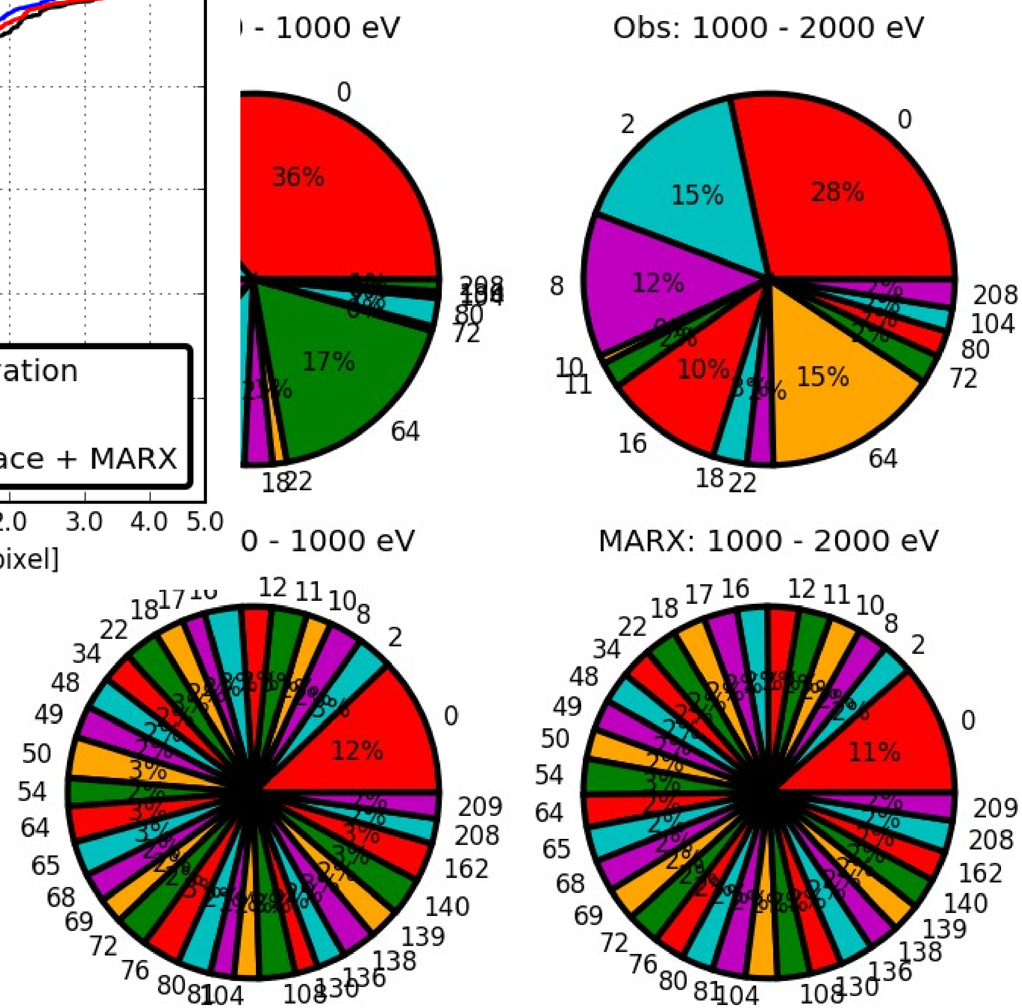




Need the distribution as simulation input

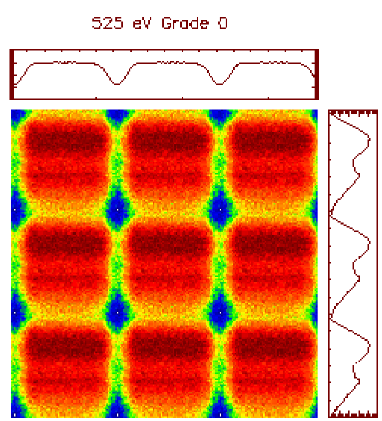


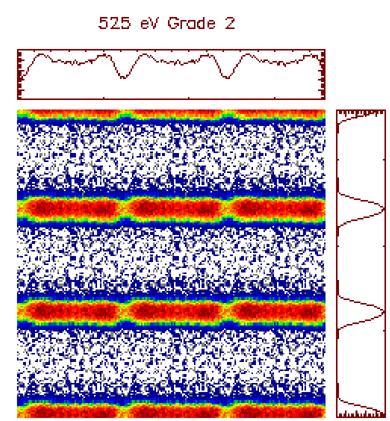




Ways to find the sub-pixel distribution

- **From calibration data with pin-hole illumination**
- **From size distribution of electron clouds**
- **Reconstruct sub-pixel distribution from observed (integrated) distribution**





Other potential applications

- **Better pile-up model**
- **Calculate fraction of background photons in region from grade distribution (particularly for faint, extended sources)**
- **Assign each photon a source/background probability and use that in fit**

Event properties

For each detected photon we know:

- energy E
- grade g
- position of pixel on the chip, with chip center coordinates x, y.

Looking for function $f(E, \hat{x}, \hat{y}) \rightarrow < p_1, p_2, p_3, ..., p_n >$
where $\hat{x}, \hat{y}$ are sub-pixel position relative to pixel center $-0.5.. + 0.5$

Properties of grade distribution

Probability to observe an event of grade g then is:

$$p(g|E,x,y) = \int_{-0.5}^{+0.5} \int_{-0.5}^{+0.5} PSF(x_0 - (x+\hat{x}), y_0 - (y+\hat{y})) f_g(E,\hat{x},\hat{y}) d\hat{x} d\hat{y}$$

We know that we get exactly one grade per event:

$$\sum_g p_g(\hat{x},\hat{y}) = 1 \vee \hat{x},\hat{y}$$

(but how do we make best use of this?)

Some simplifications

Let's do some simplifications:

- Use one energy E
- ignore X-ray background
- ignore chip type (front/back-illuminated)

My current approach

- Look at one grade g at the time.
- Bin continuous f into discrete distributions F, e.g. grid of 3×3 or 5×5 sub-pixels.

For each event, calculate shape of PSF in the pixel where we detected something, e.g. for an event detected just to the "bottom left" of a point source:

$$PSF = \begin{pmatrix} .3 & .2 & .1 \\ .2 & .1 & .0 \\ .1 & .0 & .0 \end{pmatrix}$$

with $\sum PSF_{ij} = 1$ since we know that the event occured somewhere in the pixel.
(Let us assume $PSF(E, x, y)$ is known for now.)

My current approach

$$p_g = \begin{pmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{pmatrix}$$

So, can write likelyhood for event i as

$$L_{gi} = PSF_i \times p_g$$

and now maximize the sum of the likelihoods (or in practice, minimize the negative log likelihood) for all events of grade g

$$L_g = \sum S_i \times F_g$$

where the sum is over all events with detected grade g.

Combine results for several grades

- Get weights w_g just from observed frequency

$$p(\hat{x}, \hat{y}) = \begin{pmatrix} w_1 p_1(\hat{x}, \hat{y}) \\ w_2 p_2(\hat{x}, \hat{y}) \\ w_3 p_3(\hat{x}, \hat{y}) \\ \ldots \end{pmatrix}$$

Pros and Cons

- Practical: Easy to set up parallel fits, limited number of variables
- Practical: Need to ensure $F_g(i,j) \geq 0$ for all i,j and $\sum F_g(i,j) = 1$
- Not correct (but maybe good enough?)! Does not enforce $\sum_g F_g(i,j) = 1$ for all (i,j).
- I feel there is a lot of information I do not use, which makes me think there must be a better way.

The end (for now)

- Did run some simulations
- but not totally happy with results

Your ideas here...