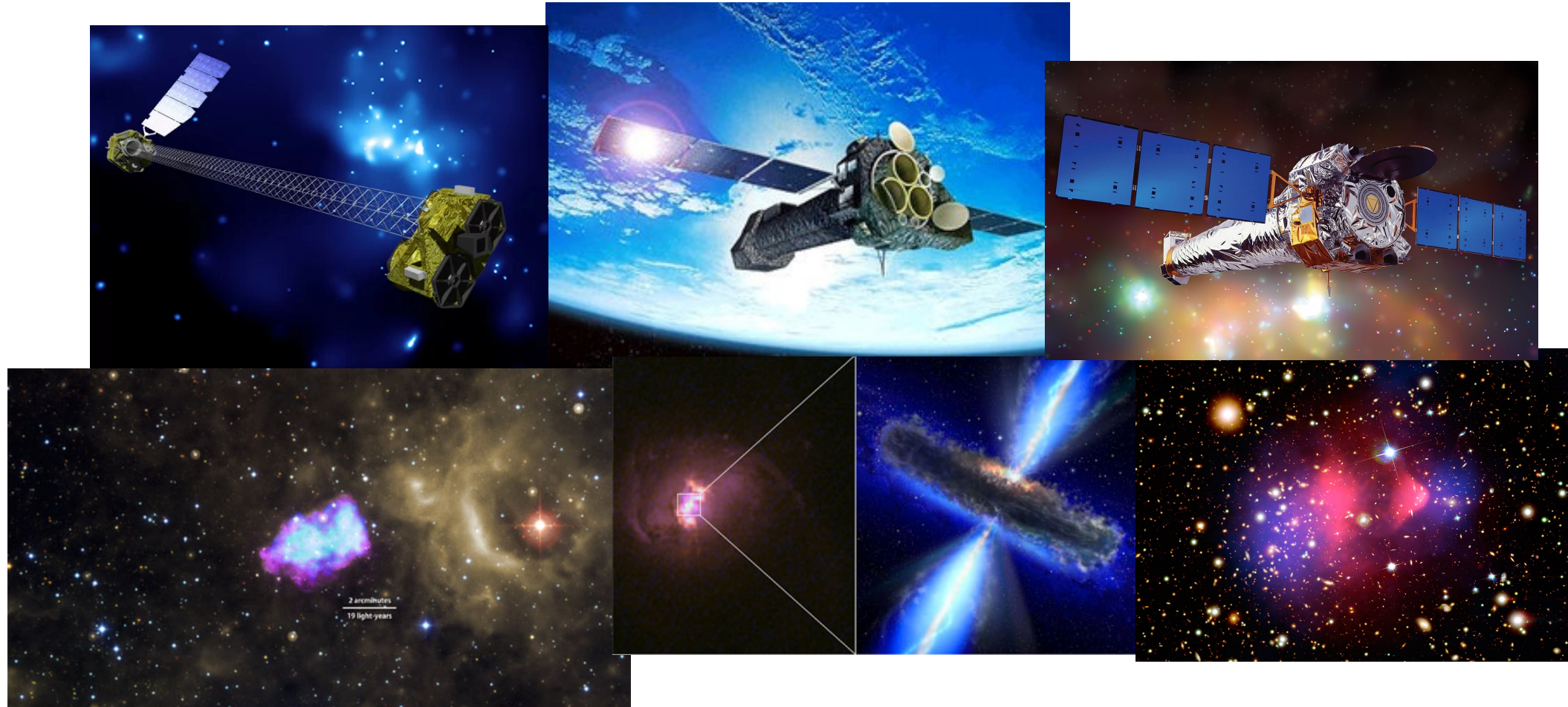


# Lecture 3: Fitting models to data

Rafael Martínez-Galarza

Harvard-Smithsonian Center for Astrophysics



# Previously on... astrostats

- We have studied the basic rules of probability and learned how to estimate probabilities in terms of a frequency or repetition of an experiment
- We have learned what random variables are, and we have seen that we can characterize them with distributions.
- We have seen that PDFs are very common in astronomy, as they are related to the very problem of measurements that have uncertainties. But also they appear naturally in nature (e.g., the IMF).

# This lecture:

- We will learn how to use statistics in order to compare our models of Nature with data obtained with telescopes.
- We will learn how the problem of fitting a model to a set of data is philosophically different for frequentists and Bayesians.
- We will learn how we can evaluate how good our fit is to a particular set of data, given some assumptions.
- We will learn how to sample the full posterior distribution of model parameters, and fully characterize the uncertainties.
- We will apply this knowledge to the modeling of SNR spectra.



# Bayes' Rule

Rev. Thomas Bayes

## Derivation of the Bayes' Theorem

We start with the product rule of probabilities. We know that:

$$P(A|B) = \frac{P(A, B)}{P(B)}$$

And also:

$$P(B|A) = \frac{P(A, B)}{P(A)}$$

Rearranging and combining these two equations, we get:

$$P(A|B)P(B) = P(A|B)P(A)$$

Now we divide both sides by  $P(B)$ :

$$P(A|B) = \frac{P(A|B)P(A)}{P(B)}$$

This is known as the Bayes' rule



“An Essay towards solving a Problem  
in the Doctrine of Chances”, 1763

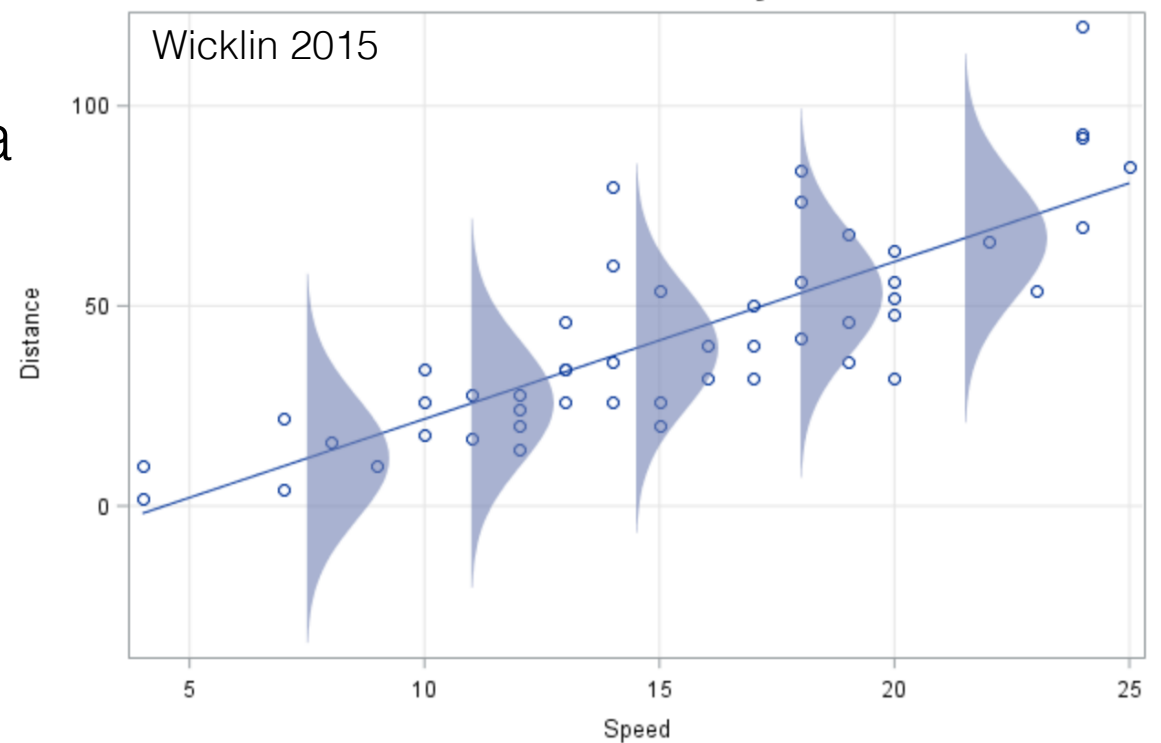
The Bayes' theorem is just an implication of the rule of probabilities. But its interpretation has a deep impact in the way we interpret probabilities, as we will see in the next lecture. In astronomy, Bayesian inference has received lots of attention over the last years.

# What is a model?

- A (physical) model is an abstract representation of reality. It is the **framework of ideas and concepts** from which we interpret our observations or experimental results.
- As such, a physical model is capable of **making predictions** about the reality that it represents. For example, there is a model of reality in which a fair coin will end up heads half of the time.
- But models (and I cannot emphasize this enough) are NOT reality itself, for **reality itself is not accessible to us**. Reality as we perceive it is a model created by our brain based on sensorial signals.
- Statistics and probability are the tools we use to test these models against observations obtained in experiments.

# What is data?

- Frequentist statistics is one answer to this philosophical question. It treats data as **a sample from an existing population**.
- Data then represent an underlying reality that we are trying to characterize.
- In this sense, data are just particular samples randomly generated from a **distribution of probability** that has some parameters. Let us refer to these as  $\theta$ .
- To know the true value  $\theta^*$ , we would need the entire population, not just our sample. But we only have the sample, and therefore all we can hope for is an **estimate**  $\hat{\theta}$ .



Two different types of parameters:

The parameters  $\lambda$  of the model predict the expected values of each datapoint

The parameters  $\theta$  for the distribution from which is datapoint is generated

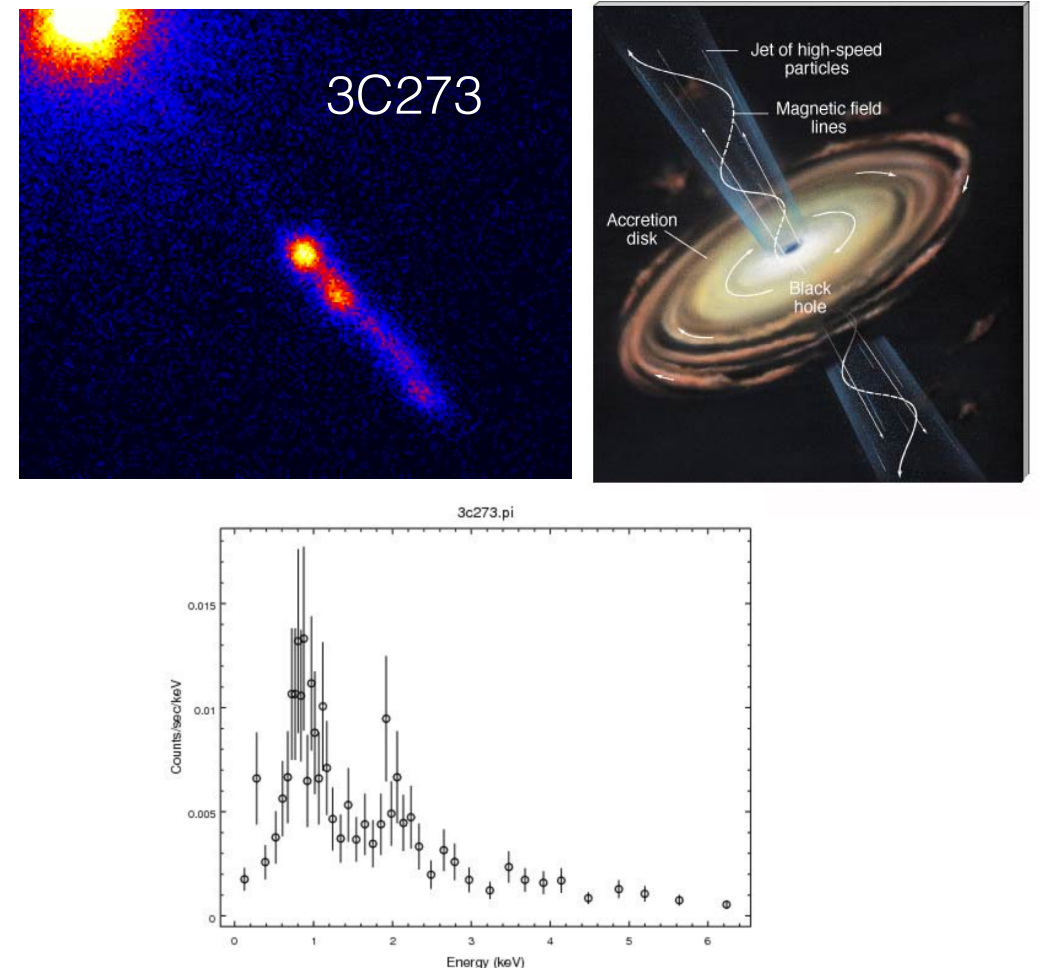
# Reprise: the likelihood

Remember: **likelihood** is a way to relate our model of reality to data. One possible model of reality: Aldebaran has a visible magnitude of 0.03.

A more complex model: X-rays observed towards quasar 3C273 are **synchrotron emission** due to relativistic charged particles been accelerated by the quasar's magnetic field of intensity  $B$ .

We can always test how likely the data we observe are given that model, whose **parameters**  $\lambda$  are the charge of the accelerated particles,  $B$ , etc. In this case:

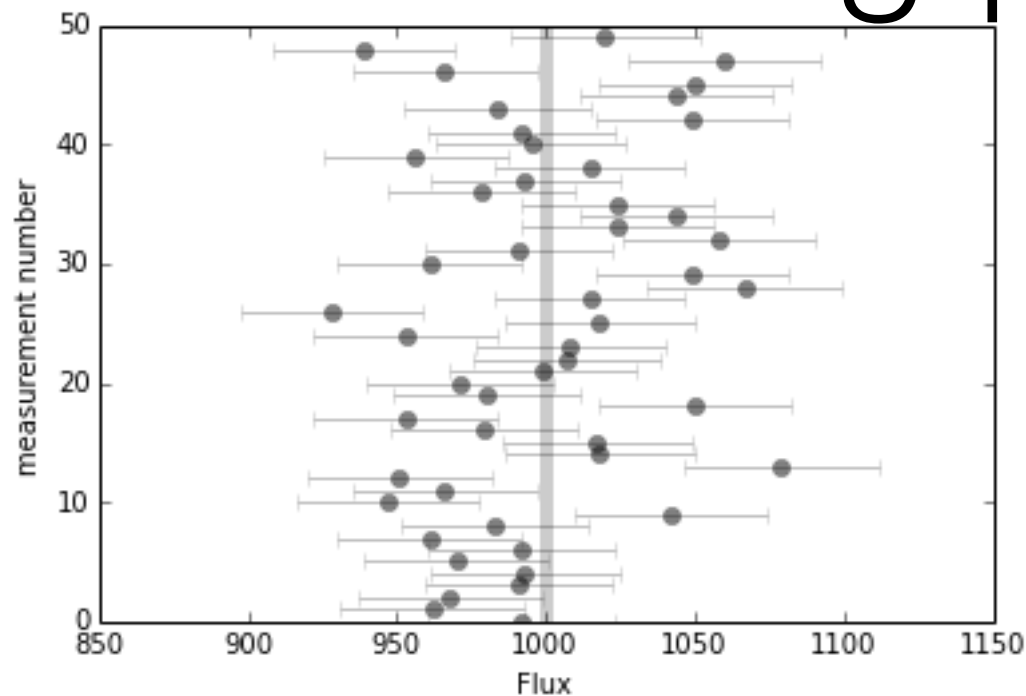
$$L(\lambda) = \prod_{i=1}^n P(x_i | \lambda)$$



Frequentist: the parameters  $\lambda$  adopt fixed true values. How likely is our spectrum **given those true values?**

In fact, what parameter values maximize  $L(\lambda)$ ?

# The frequentist view of counting photons: MLE



- Maximum Likelihood Estimate refers to the method of maximizing the likelihood (which is a PDF over the possible values of the flux).
- We can do this either analytically or computationally.
- MLE assumes nothing about other sources of information about the model parameters.

The likelihood for a normal model:

$$\log \mathcal{L} = -\frac{1}{2} \sum_{i=1}^N \left[ \log(2\pi e_i^2) + \frac{(F_i - F_{\text{true}})^2}{e_i^2} \right]$$

We try to find the  $F_{\text{true}}$  that maximizes this likelihood. In this case we can do it analytically by setting its derivative to zero:

$$F_{\text{est}} = \frac{\sum w_i F_i}{\sum w_i}; \quad w_i = 1/e_i^2$$

If all errors are equal, we get the mean, as expected

$$F_{\text{est}} = \frac{1}{N} \sum_{i=1}^N F_i$$



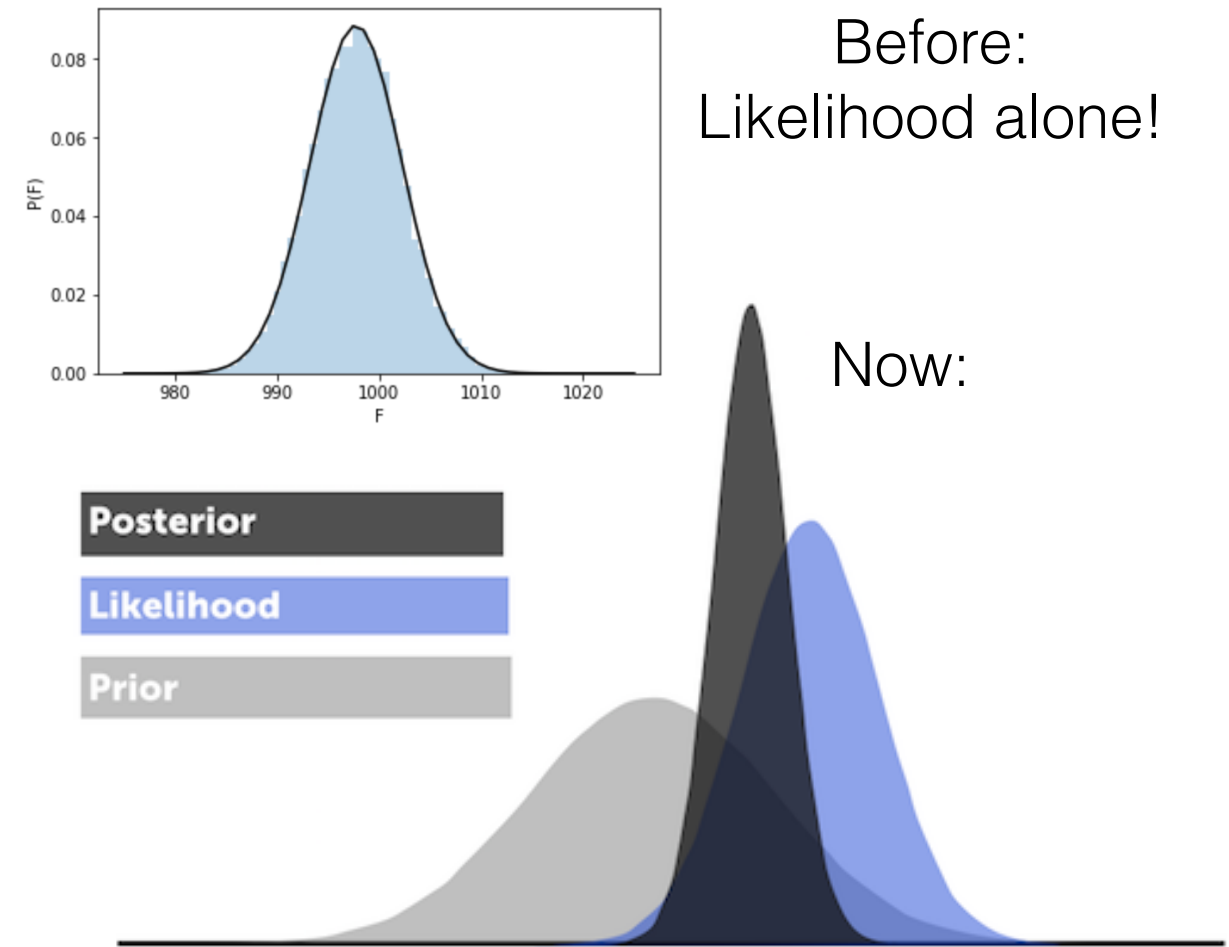
# The world of Bayes

From a philosophical point of view, there is a different way to understand this problem. What if instead of the **likelihood of the data given a true model**, we ask the question: **what is the probability of the flux being  $F_{\text{true}}$ , given these data?** For frequentists, this has no meaning whatsoever:

$$P(F_{\text{true}} | D)$$

In Bayesian statistics, probability is not about frequencies anymore; **it is about a degree of belief!**

In this framework, the model parameters are not fixed true values anymore, but random variables with an associated probability; **we infer the physics, not data!**



**The Bayes' Rule:**

$$P(F_{\text{true}} | D) = \frac{P(D | F_{\text{true}}) P(F_{\text{true}})}{P(D)}$$

This comes from the rules of probability.

# What are all those terms?

$$P(F_{\text{true}} \mid D) = \frac{P(D \mid F_{\text{true}}) P(F_{\text{true}})}{P(D)}$$

- $P(F_{\text{true}} \mid D)$ : The **posterior**, or the probability of the model parameters given the data: this is the result we want to compute.
- $P(D \mid F_{\text{true}})$ : The **likelihood**, which is proportional to the  $\mathcal{L}(D \mid F_{\text{true}})$  in the frequentist approach, above.
- $P(F_{\text{true}})$ : The **model prior**, which encodes what we knew about the model prior to the application of the data  $D$ .
- $P(D)$ : The **data probability**, which in practice amounts to simply a normalization term.

The prior allows to include other information in the computation, information that can come from previous experiments or constraints. The prior measures what we believe the parameters should be **BEFORE** we have made the current measurement.

Calculating posteriors exactly is difficult!

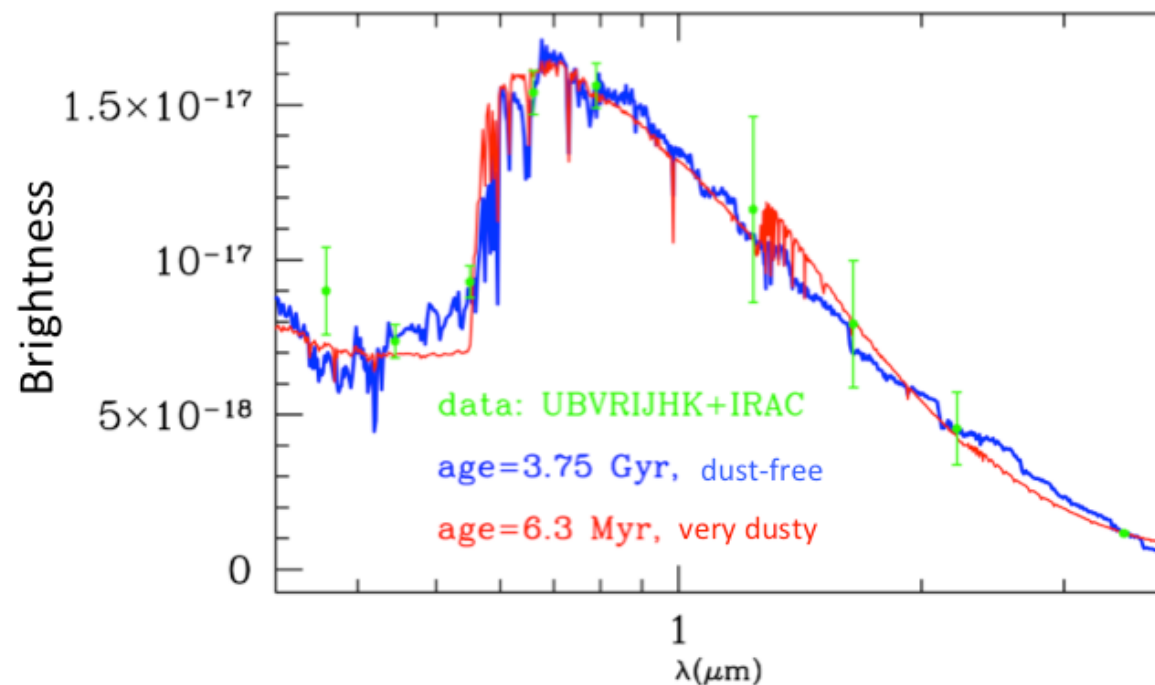
# Notebook: Frequentists vs Bayesian

Fitting models to data



# Model fitting (the frequentist view)

Suppose you want to fit a set of photometric data with a model of your choice.



- Your data  $D = \{y_1, y_2, \dots, y_n\}$  is defined at wavelengths  $\Lambda = \{\lambda_1, \lambda_2, \dots, \lambda_n\}$ .
- Your data has observational errors  
 $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_n\}$
- Your model depends on parameters  
 $\Theta = \{\theta_1, \theta_2, \dots, \theta_m\}$ . Example: SFR,  $A_V$ ,  $M^*$

The usual approach is to use  $\chi^2$  minimization:

$$\chi^2 = \sum_{n=1}^N \frac{(y_n - y'_n)^2}{\sigma_n^2} \quad (\text{minimize})$$

But what does this mean?

- What you are really doing is a maximum likelihood estimate (MLE) for a normal distribution:

$$P(y_n|\theta) \propto \exp\left(-\frac{(y_n - y'_n)^2}{2\sigma_n^2}\right)$$

$$P(D|\theta) \propto \prod_{n=1}^N \exp\left(-\frac{(y_n - y'_n)^2}{2\sigma_n^2}\right)$$

$$\log P(D|\theta) = \sum_{n=1}^N \left(-\frac{(y_n - y'_n)^2}{2\sigma_n^2}\right) + K = -\frac{1}{2}\chi^2 + K$$

Why Gaussian? Central Limit Theorem

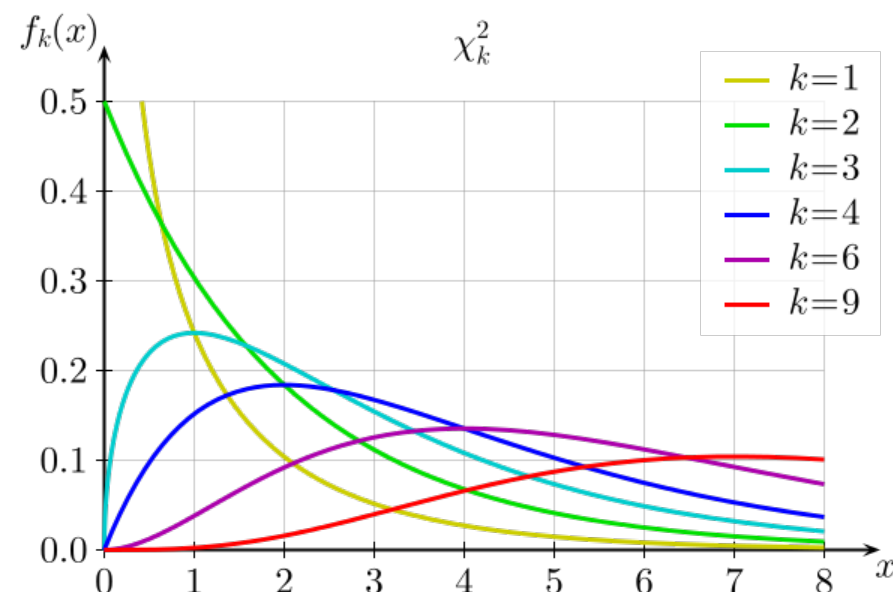
# How do you assess if your fit is good?

## the $\chi^2$ test

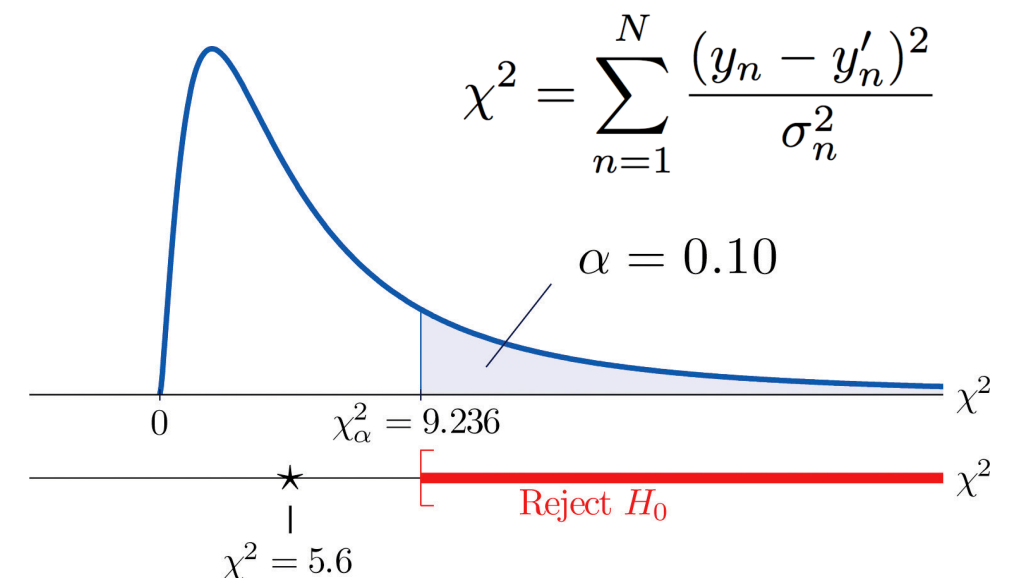
**The null hypothesis:** there is no difference between a distribution of datapoints sampled from Gaussian errors and the distribution of my points.

Accept or reject this hypothesis? Since the distribution of the sum of independent normal RVs is a  $\chi^2$  distribution, then we use it for the test.

The shape of the distribution depends on the number of degrees of freedom:



Remember the  $\chi^2$  distribution?



If the null hypothesis is true, and your data were actually drawn from Gaussian distributions, then the square of the differences between model predictions and datapoint should be distributed according to this distribution.

We reject the null hypothesis for values of  $\chi^2$  above certain critical value that is related to the significance level  $\alpha$ .

# Assumptions

- We should be careful and always think of your assumptions (We often do not do it in writing papers)
- If we obtain photometry of a source at a given band many times, the different measurements will be normally distributed. (Gaussian errors)
- Your model is right. (Likelihood is probability of your data being drawn from your model)
- Photometric points in different bands are uncorrelated (Joint probability is the product of independent likelihoods).
- No x-axis errors.

$$P(D|\theta) \propto \prod_{n=1}^N \exp \left( -\frac{(y_n - y'_n)^2}{2\sigma_n^2} \right)$$

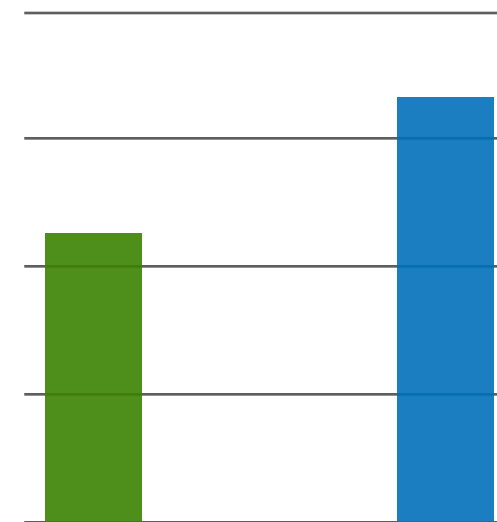
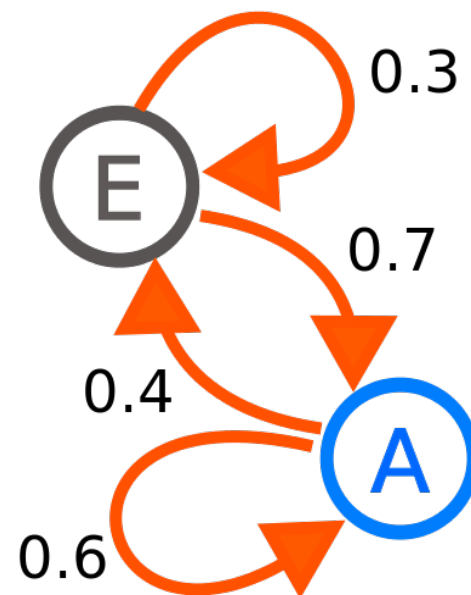
# Sampling

Depending on whether we want to obtain a point estimate for the parameters, or investigate their uncertainties, we use an **fitter** or a **sampler**.

A point estimate is known as a Maximum A Posteriori (MAP) estimate.

Markov Chain Monte Carlo (MCMC) methods consist on the construction of a Markov Chain that has the desired posterior as its equilibrium distribution.

A Markov Chain is a sequence in which next state only depends on current state.



Trick is: construct a Markov chain that looks like the posterior.

Metropolis-Hastings algorithm: a possible way to achieve this.

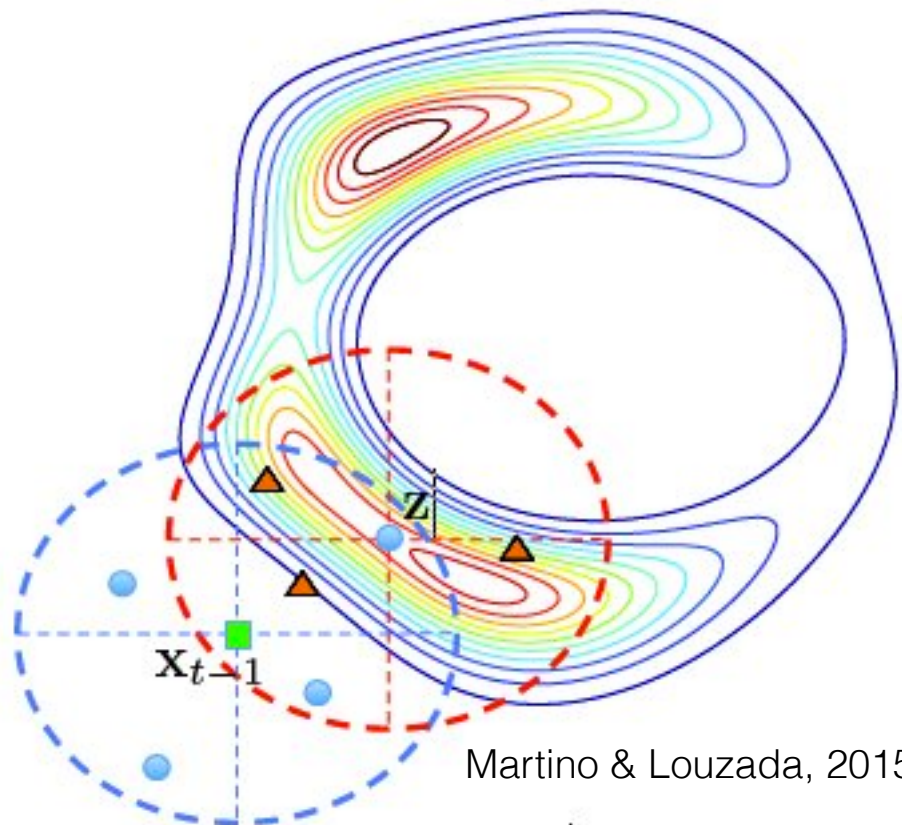
Choose next candidate according to  $g(x|x')$ , and accept or reject according to:

$$A(x'|x) = \min \left( 1, \frac{P(x')}{P(x)} \frac{g(x|x')}{g(x'|x)} \right)$$



# The Metropolis-Hastings Algorithm

The target and proposal distributions



We want to randomly hop in this probability landscape in such a way that we land more often on areas of higher probability.

How do we do it?

We define a **proposal distribution**  $g(x|x')$  that represents the probability of jumping to the next position, given the position where I am now.

I decide if I actually take the jump based on the following:

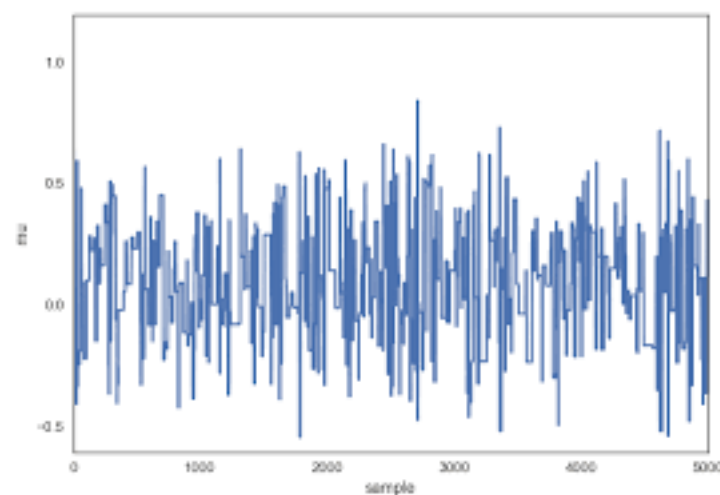
- If  $P(x) > P(x')$ , accept the sum
- If  $P(x) < P(x')$ , accept the jump with probability proportional to:

$$\frac{A(x', x)}{A(x, x')} = \frac{P(x')}{P(x)} \frac{g(x|x')}{g(x'|x)}$$

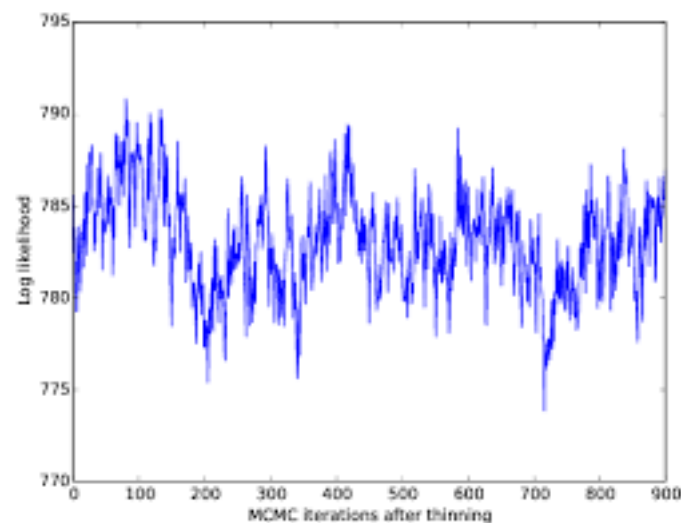
If we do this many times, we are guaranteed that the histogram of the samples will look like the target distribution (but devil is in the details.)

# Convergence of MCMC chains

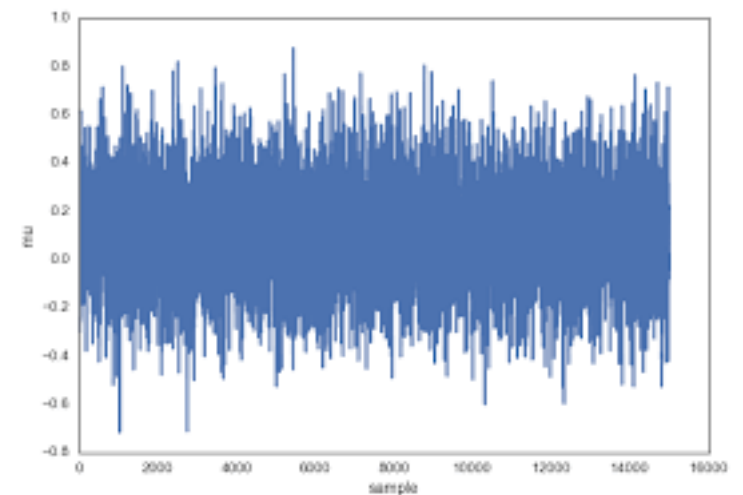
As your MCMC chain steps across the parameter space, after many iterations you expect the parameter values to converge to the target distribution. How does a converged chain look like?



Too few effective samples



Samples are correlated

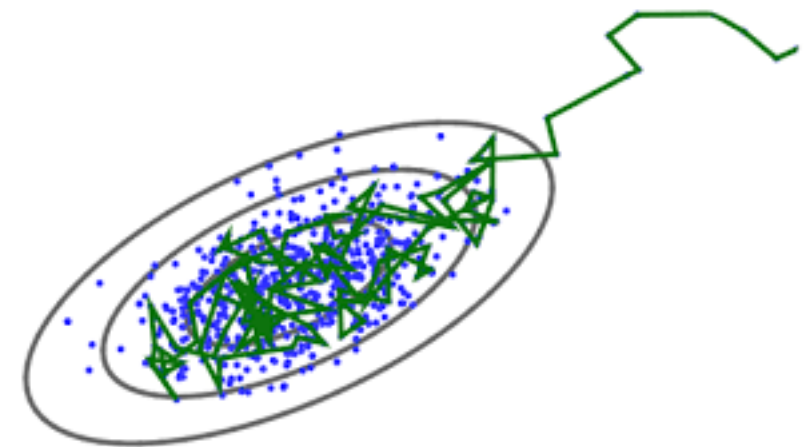


Converged

You want the acceptance rate to be between 30% and 40%

What can I tune in my MCMC sampler to get a chain to converge?

- 1) The total number of iterations
- 2) Remove early burn-in samples
- 3) De-correlate samples by skipping some of them
- 4) Change the step size (proposal dist)

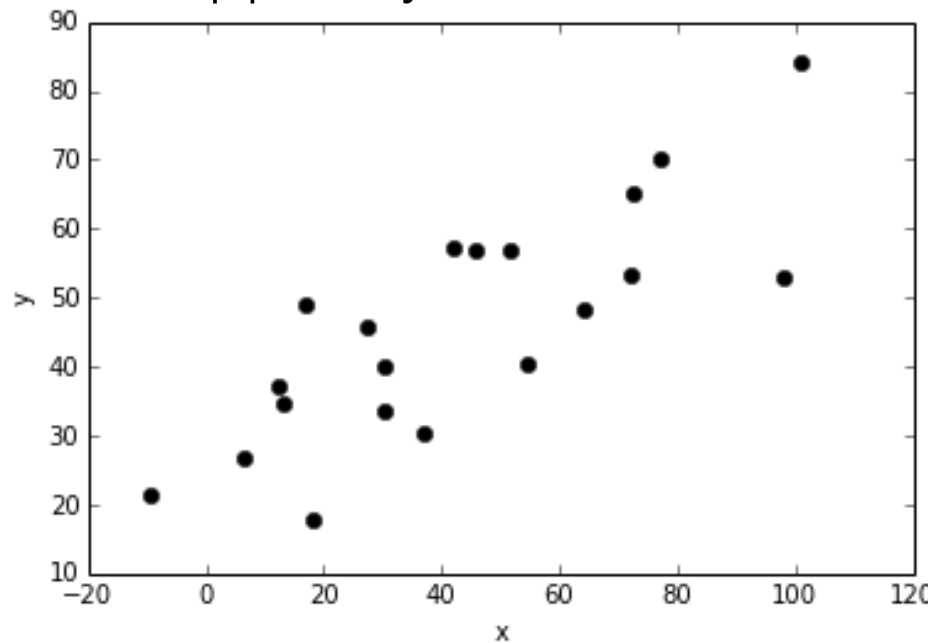


Example:  
Metropolis-Hastings

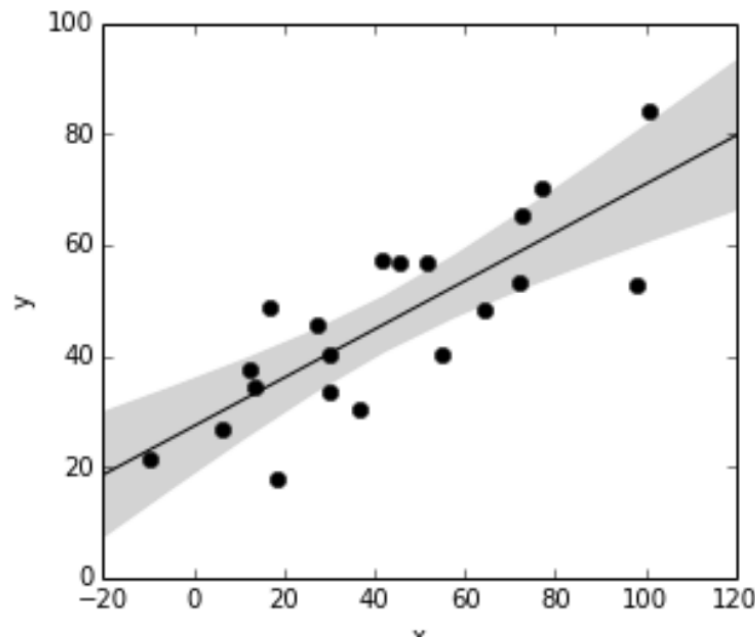
see Notebook

# Fitting vs Sampling

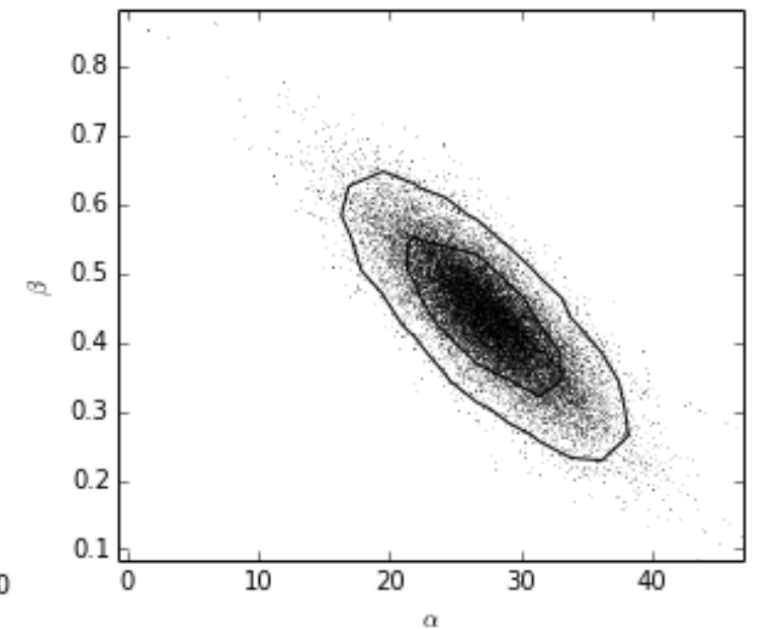
Suppose you want to fit this:



Here is a fit



Samples from posterior

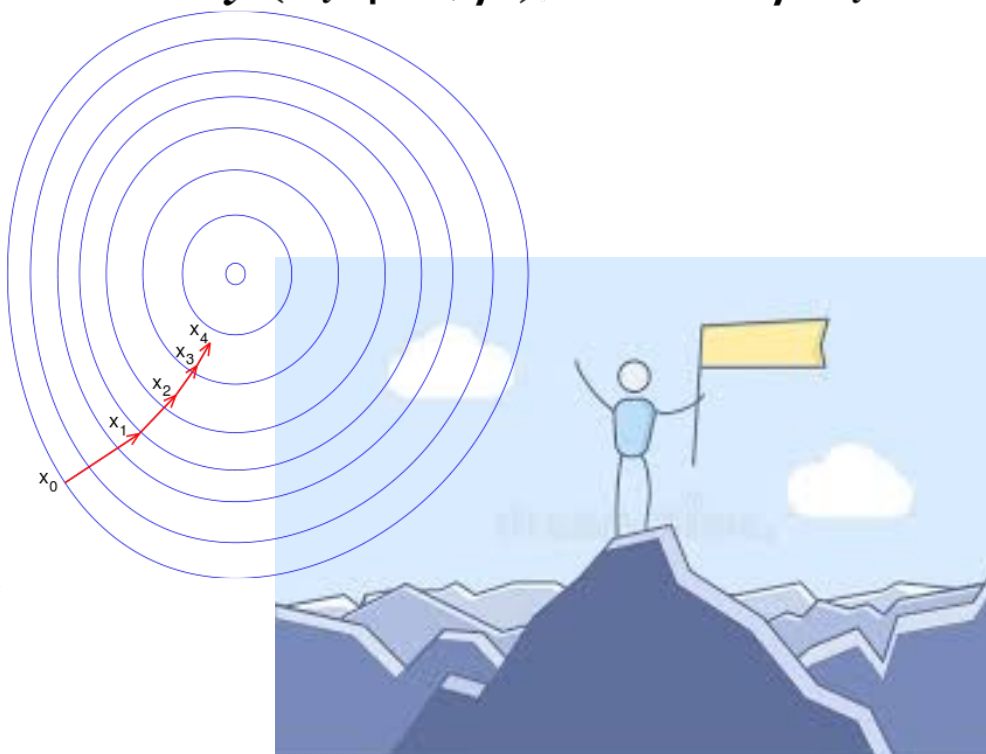


$$\hat{y}(x_i | \alpha, \beta) = \alpha + \beta x_i$$

Fitting is about climbing the probability surface until we are sufficiently close to the region of highest probability.

Sampling is reproducing the posterior as accurately as possible via some stochastic process

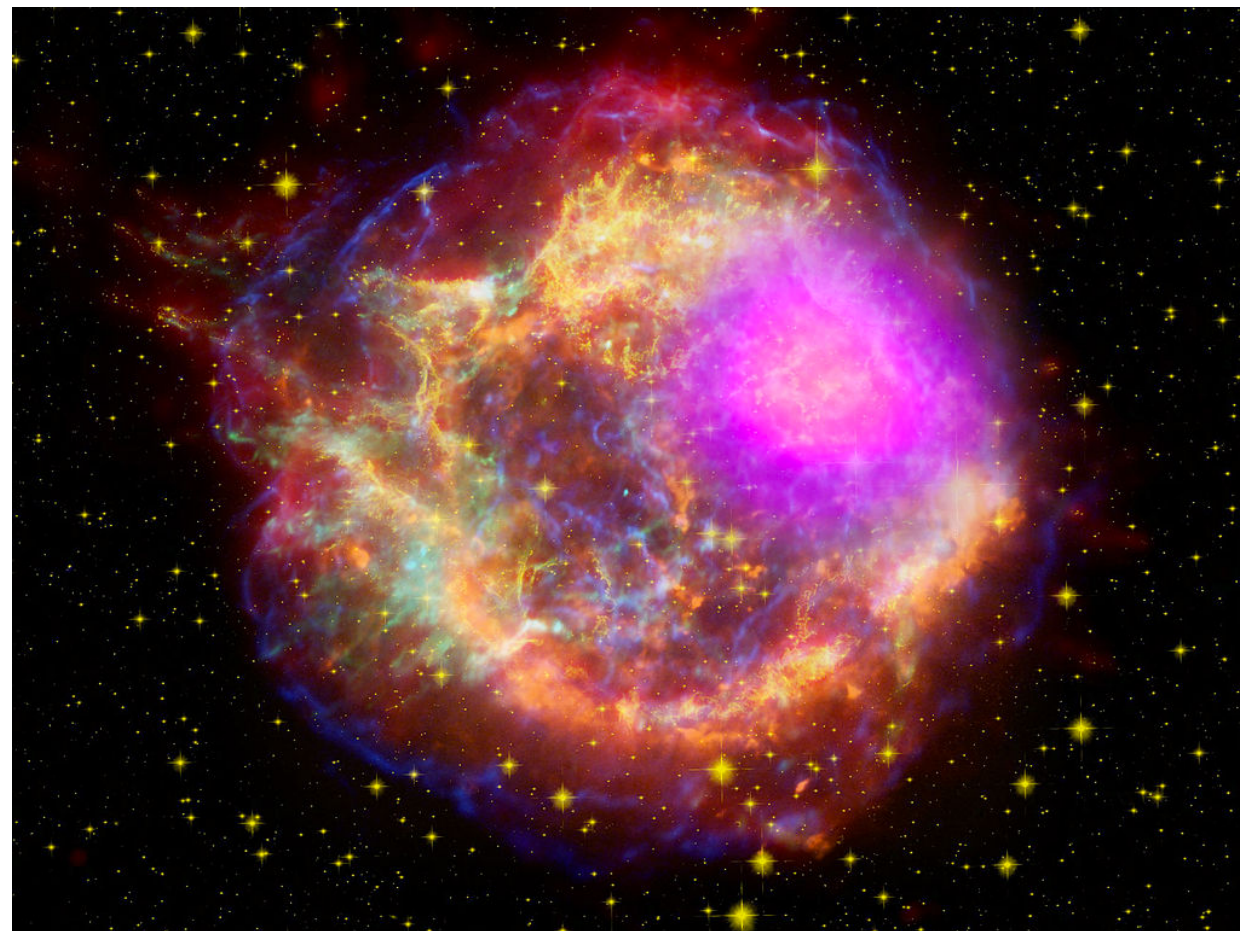
Uncertainties are fully characterized if sampling is done properly.



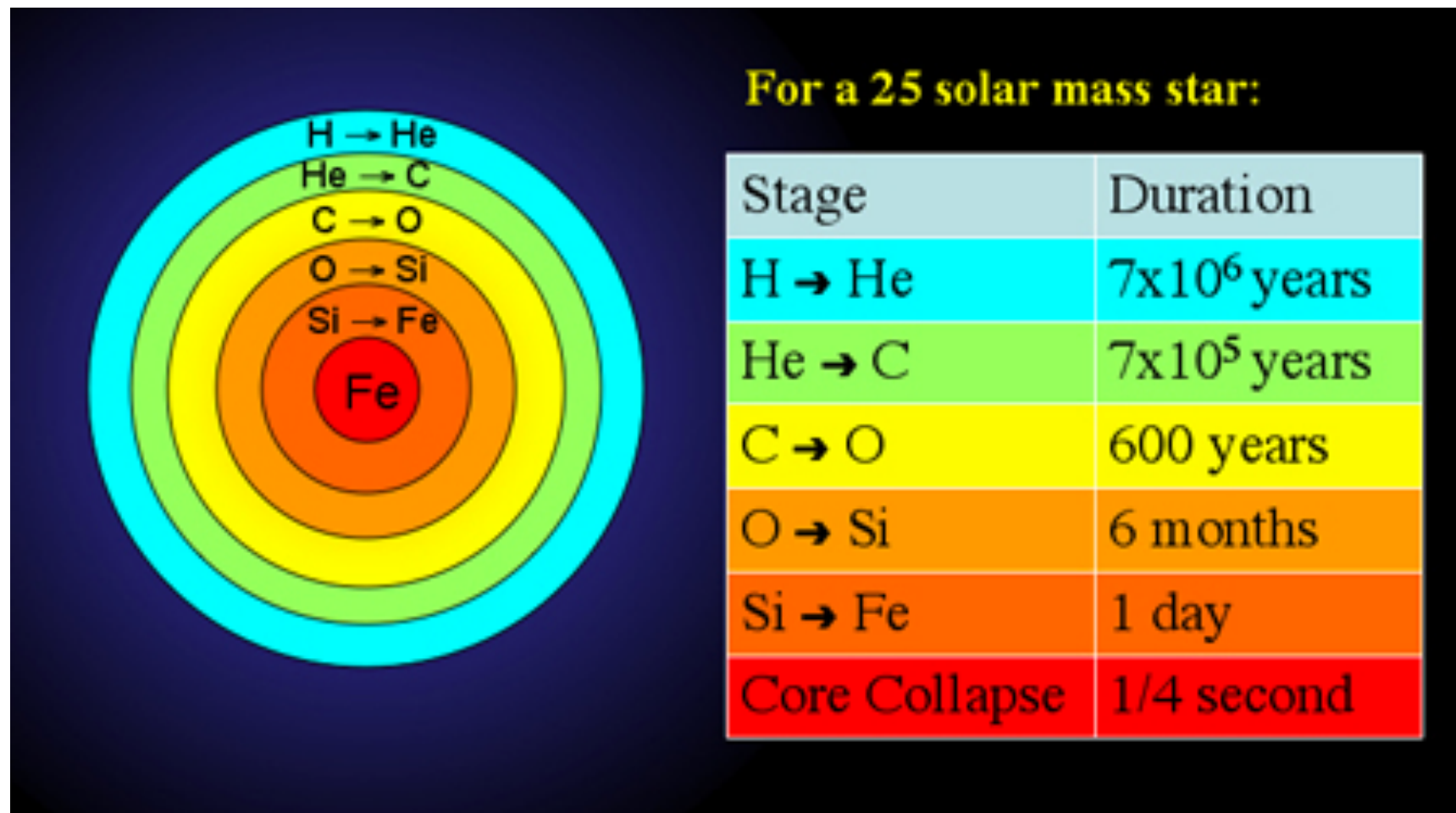


# Fitting X-ray images and spectra (w/ P. Slane)

- How do we infer physical information from the most extreme objects in the Universe?
- Let's take the example of one the most famous supernova remnant imaged by Chandra: Cassiopeia A



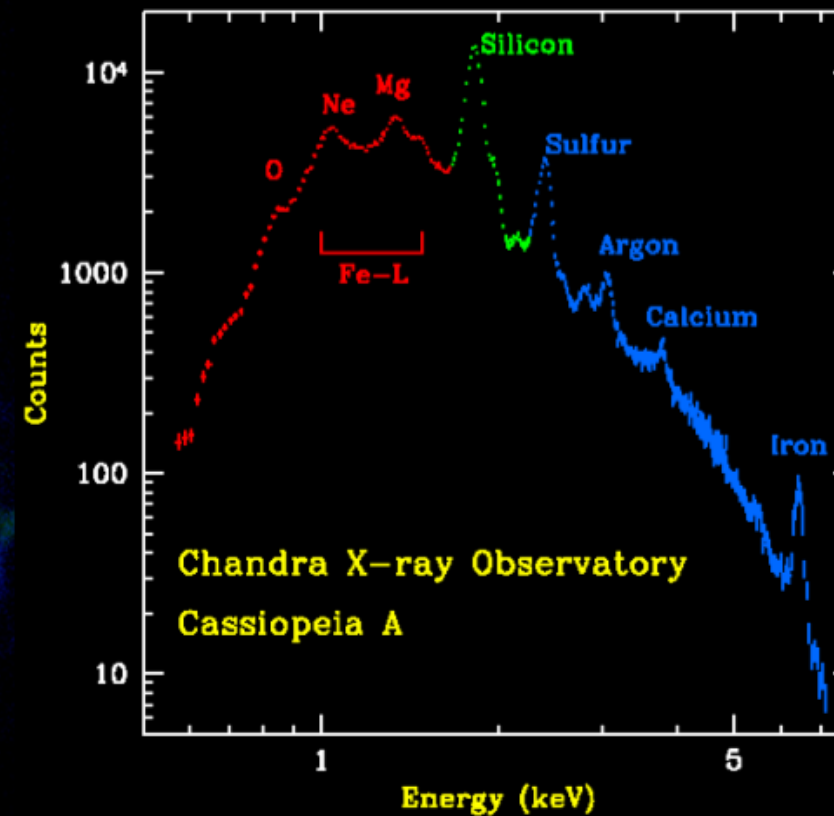
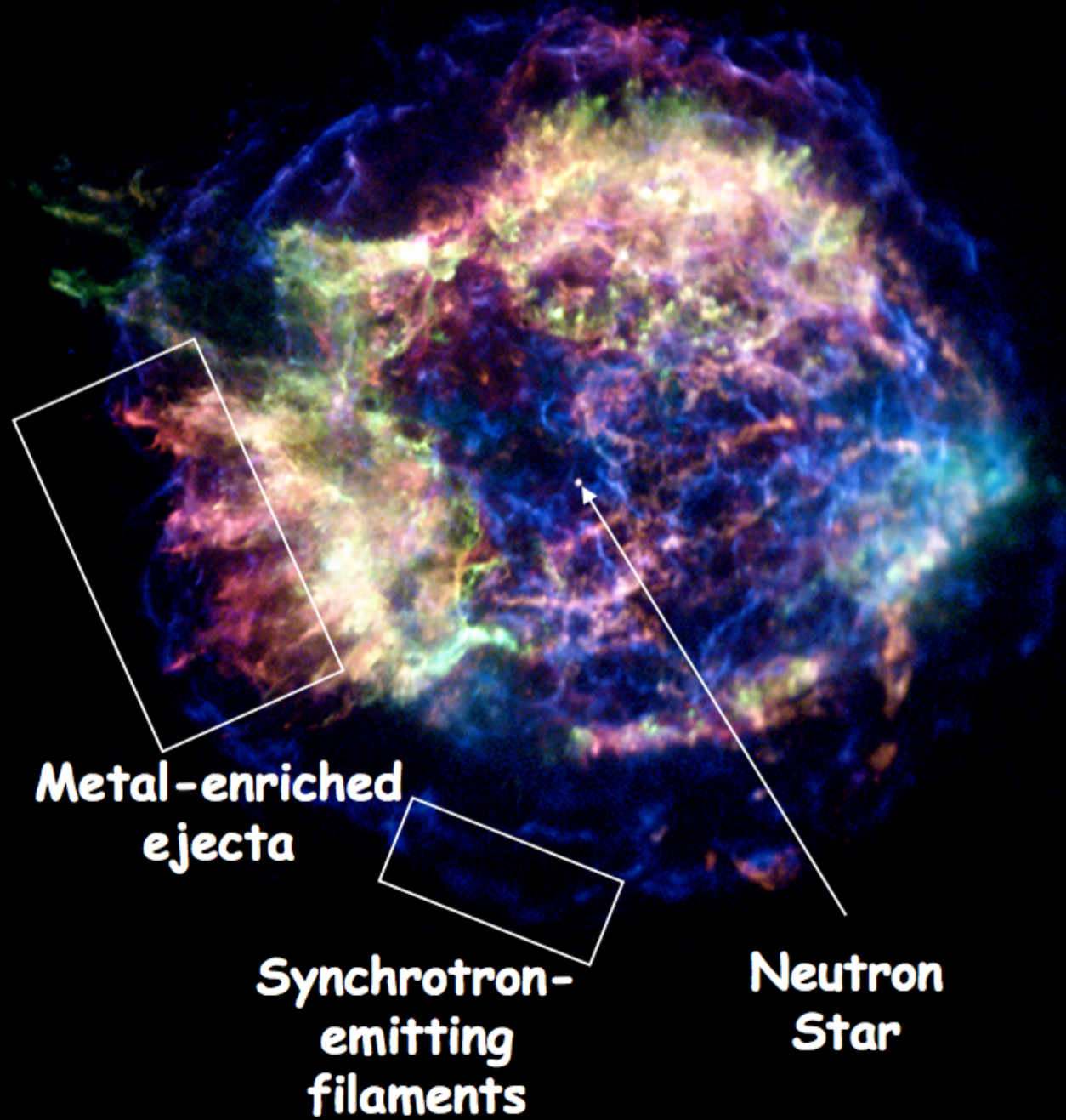
# Core-collapse SN



Just before core-collapse, the interior of a massive star looks a little like an onion, with shells of successively lighter elements burning around an iron core. These burning stages become shorter and shorter as lighter elements are fused into heavier elements.



# Exercise: Spectra of Cas A and its CCO



## ACIS-S Observation:

3-color image in soft/medium/hard bands (ds9)

Spectra of discrete regions from the SNR(specextract)

Spectral fitting (xspec/sherpa, NEI models w/ variable abundances; power law model; blackbody model)

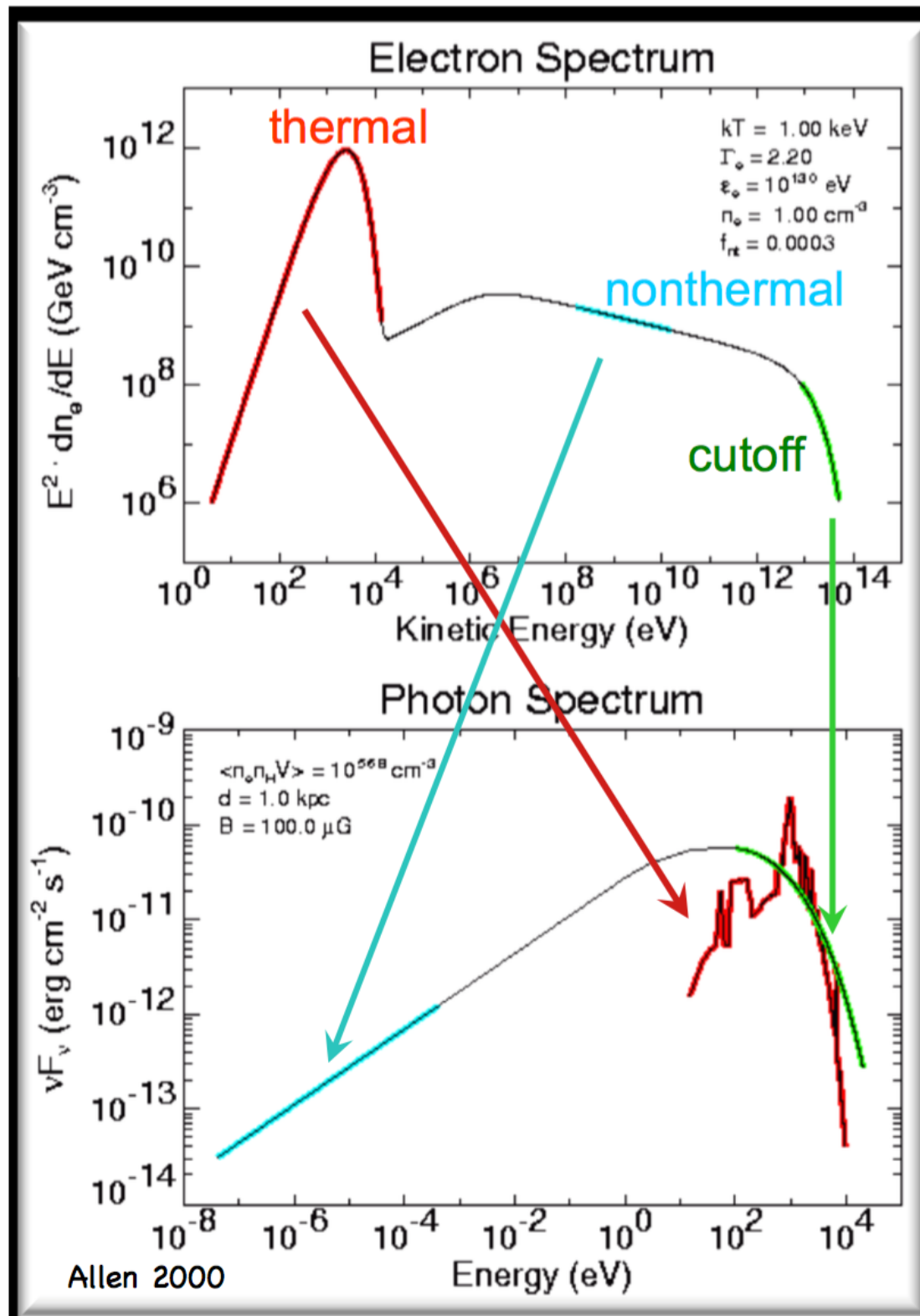
- Complex ejecta distribution
- Nonthermal filaments
- Neutron star in interior

Hughes, Rakowski, Burrows, & Slane 2000, ApJ, 528, L109

Hwang, Holt, & Petre 2000, ApJ, 537, L119

Slide: P. Slane

# The spectrum of shocked electrons



## Shocked Electrons and their Spectra

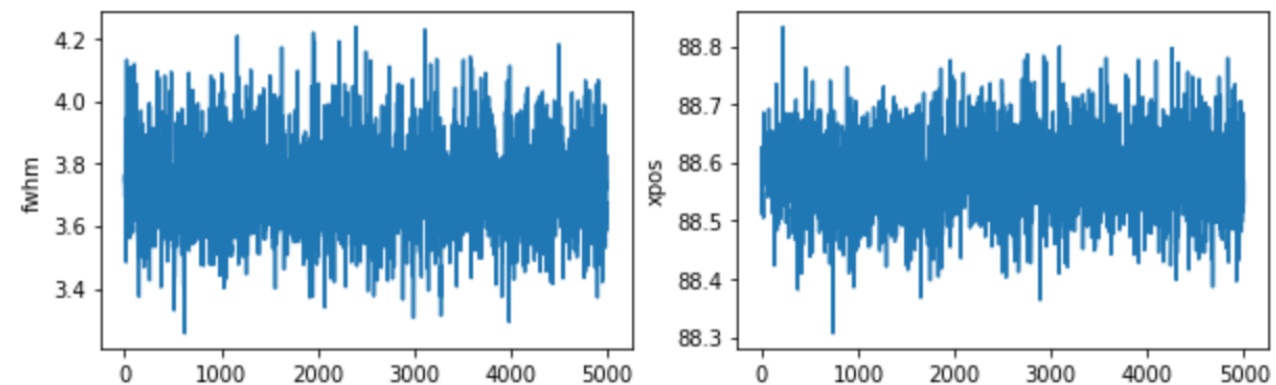
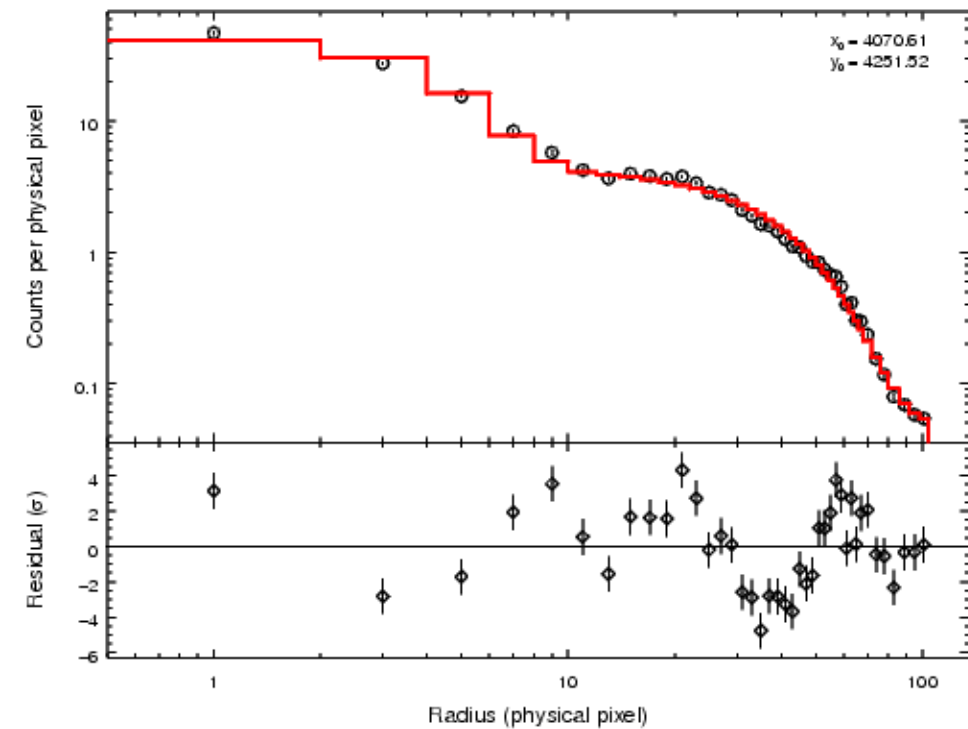
- **Thermal electrons produce line-dominated x-ray spectrum with bremsstrahlung continuum**
  - yields  $kT$ , ionization state, abundances
- **nonthermal electrons produce synchrotron radiation over broad energy range**
  - responsible for radio emission
- **high energy tail of nonthermal electrons yields x-ray synchrotron radiation**
  - rollover between radio and x-ray spectra gives **exponential cutoff** of electron spectrum, and **limits on energy of associated cosmic rays**
  - large contribution from this component **modifies dynamics** of thermal electrons





### **Sherpa lets you:**

- *fit 1-D data sets (simultaneously or individually), including: spectra, surface brightness profiles, light curves, general ASCII arrays;*
- *fit 2-D images/surfaces in the Poisson/Gaussian regime;*
- *access the internal data arrays;*
- *build complex model expressions;*
- *import and use your own models;*
- *choose appropriate statistics for modeling Poisson or Gaussian data;*
- *import new statistics, with priors if required by analysis;*
- *visualize a parameter space with simulations or using 1-D/2-D cuts of the parameter space;*
- *calculate confidence levels on the best-fit model parameters;*
- *choose a robust optimization method for the fit: Levenberg-Marquardt, Nelder-Mead Simplex or Monte Carlo/Differential Evolution;*
- *perform Bayesian analysis with Poisson Likelihood and priors, using Metropolis or Metropolis-Hastings algorithm in the MCMC (Markov-Chain Monte Carlo);*
- *and use Python to create complex analysis and modeling functions, build the batch mode analysis or extend the provided functionality to meet the required needs.*



# Exercise

- We will now extract the spectrum of SNR Cas A and will practice our fitting and sampling skills.
- Use the same obsID as before - 12020 (Cassiopeia A), and the same regions we have defined
- Follow the recipe here to extract the spectra of these regions using CIAO: <http://cxc.harvard.edu/ciao/threads/extended/>
- Look at the differences in the spectra you extracted. What dominates the emission of the red area? What dominates the emission of the green are?
- Using Sherpa, use an absorbed blackbody to fit the spectrum of the central white dwarf. What is the temperature of the star? What is the confidence region for the temperature?
- Use `get_draws()` in Sherpa to obtain a MCMC chain. Visually inspect if the chain has converged, and plot the posteriors.