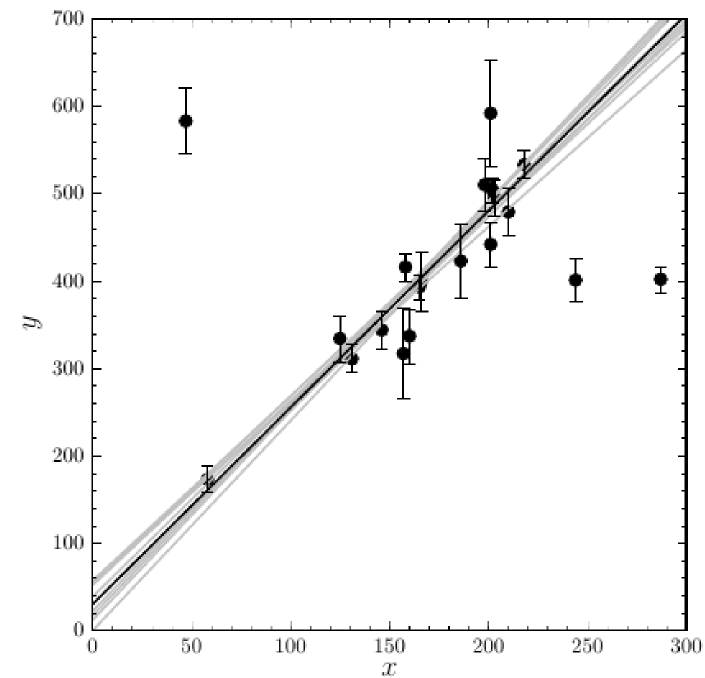
PHYS 7810: Solar Physics with DKIST

Lecture 18: Model Fitting

Ivan Milic *ivan.milic@colorado.edu*



Previous lectures

- Were observations and modeling
- We are very rarely going to perform observations and report them as such (not impossible, and nothing to scoff at, though)
- Often we want to:

1) Use a theoretical model to reproduce/justify and thus understand what we have seen. (Remember H alpha example from previous class)

2) Fit a model to the data with the aim of inferring some parameters, that, hopefully, allow us to draw some quantitative conclusions

Some examples

- Fit an ellipse to the trajectory of the observed star to find the location and the mass of the Black Hole
- Fit a parabola to the distance modulus vs redshift function to infer / detect acceleration of the universe
- Fit a straight line to the T^2 vs l, dependency to infer gravitational acceleration using simple pendulum
- Fit a cosmological model to CMB map / power spectrum to find cosmological parameters
- Fit a line formation model to the observed Stokes spectrum to infer (~~measure~~) magnetic field, velocity, temperature

Let me tell you a story about little me...

- When I was a 15 years old kid, I was attending, fanatically, this “boarding school for nerds” close to my hometown
- It is a institution for high school kids who have a keen interest in science, where you are taught scientific process
- One of the first exercises involved model fitting
- I remember using these magnificent programs back than called “Origin” and “Table Curve” and thinking:
- *“How come the program itself cannot figure which function to use to fit the data?”*

I was obviously missing a point!

- Fitting is not it’s own purpose!
- If you see some data looking like a straight line or a parabola, does not mean that you should immediately whip out your scipy.optimize.minimize package
- (Sure, there are, so-to-speak, non-inferential applications of fitting, but we are not talking about that here)
- If you are fitting **a model** to **the data** , you need a model, you need the measurements, you need errors (uncertainties), and a few more things, that we going to talk about today...

To understand all this, one article is enough and one article only:

- https://arxiv.org/pdf/1008.4686.pdf

Data analysis recipes: Fitting a model to data*

David W. Hogg
Center for Cosmology and Particle Physics, Department of Physics, New York University
Max-Planck-Institut für Astronomie, Heidelberg

Jo Bovy
Center for Cosmology and Particle Physics, Department of Physics, New York University

Dustin Lang
Department of Computer Science, University of Toronto
Princeton University Observatory

We will start from the simplest possible example

- We are measuring intensity from one pixel of our image few times (in counts). We have a strong reason to assume that the “original” (call it, “true”) number of counts is constant in time.
- We measure 20 times and get the following results:
- [10099.45461478 10033.91038118 9949.99580719 9929.26995655
- 10009.59103032 10023.19828581 10048.77589944 9878.03777698
- 9970.63765657 9898.44337474 9949.03521708 9861.05450482
- 10104.43740336 9871.37116346 9999.58484226 10070.42870939
- 10042.07927595 9922.21971703 9950.06443439 10033.89015678]

We want to figure out the true value

- We can't know for sure
- We can only estimate, pay attention now:

  The most probable value of the true value given the observations we have. (And the prior information about the true value).

- In this case, these 20 measurements (random variables) are our observations
- (Unknown) constant value is our model.

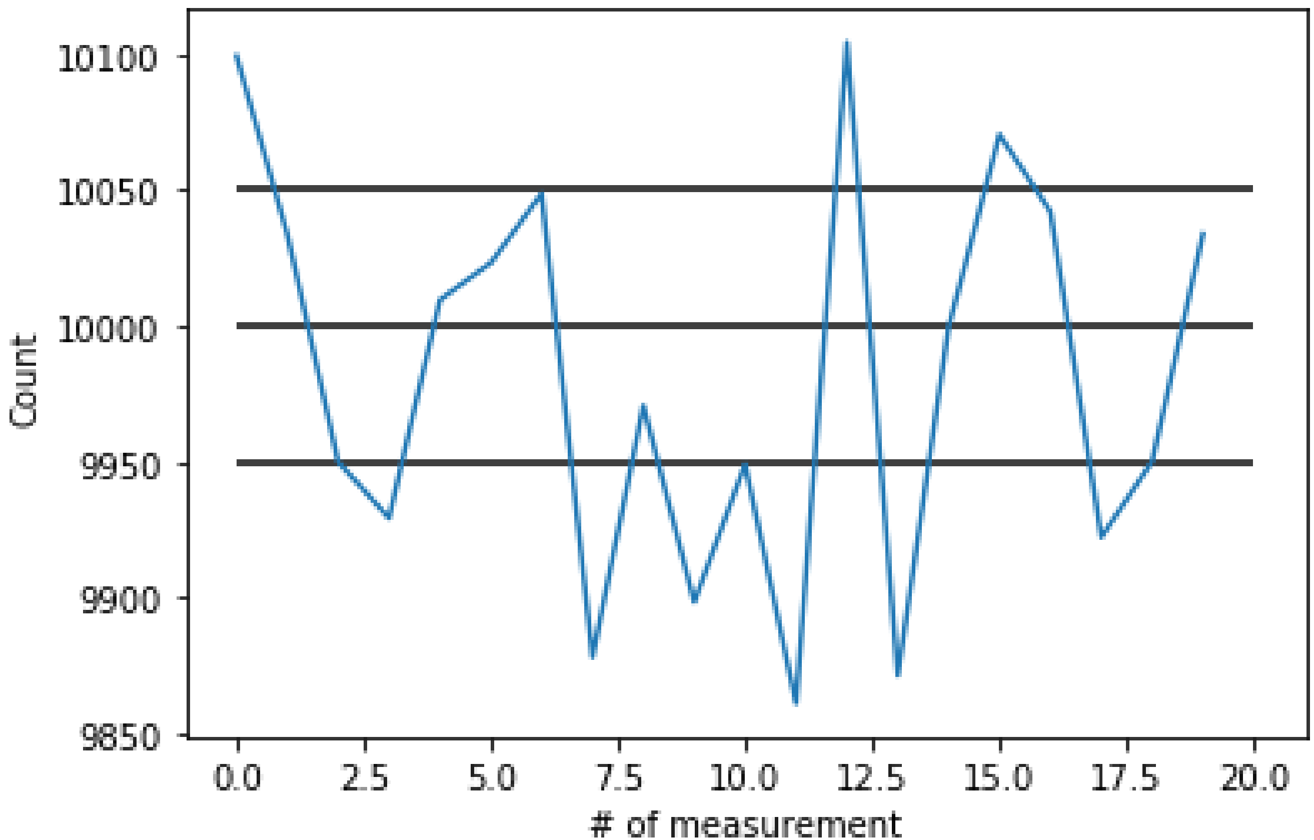
We want to figure out the true value

- We can't know for sure
- We can only estimate, pay attention now:

  The most probable value of the true value given the observations we have. (And the prior information about the true value).

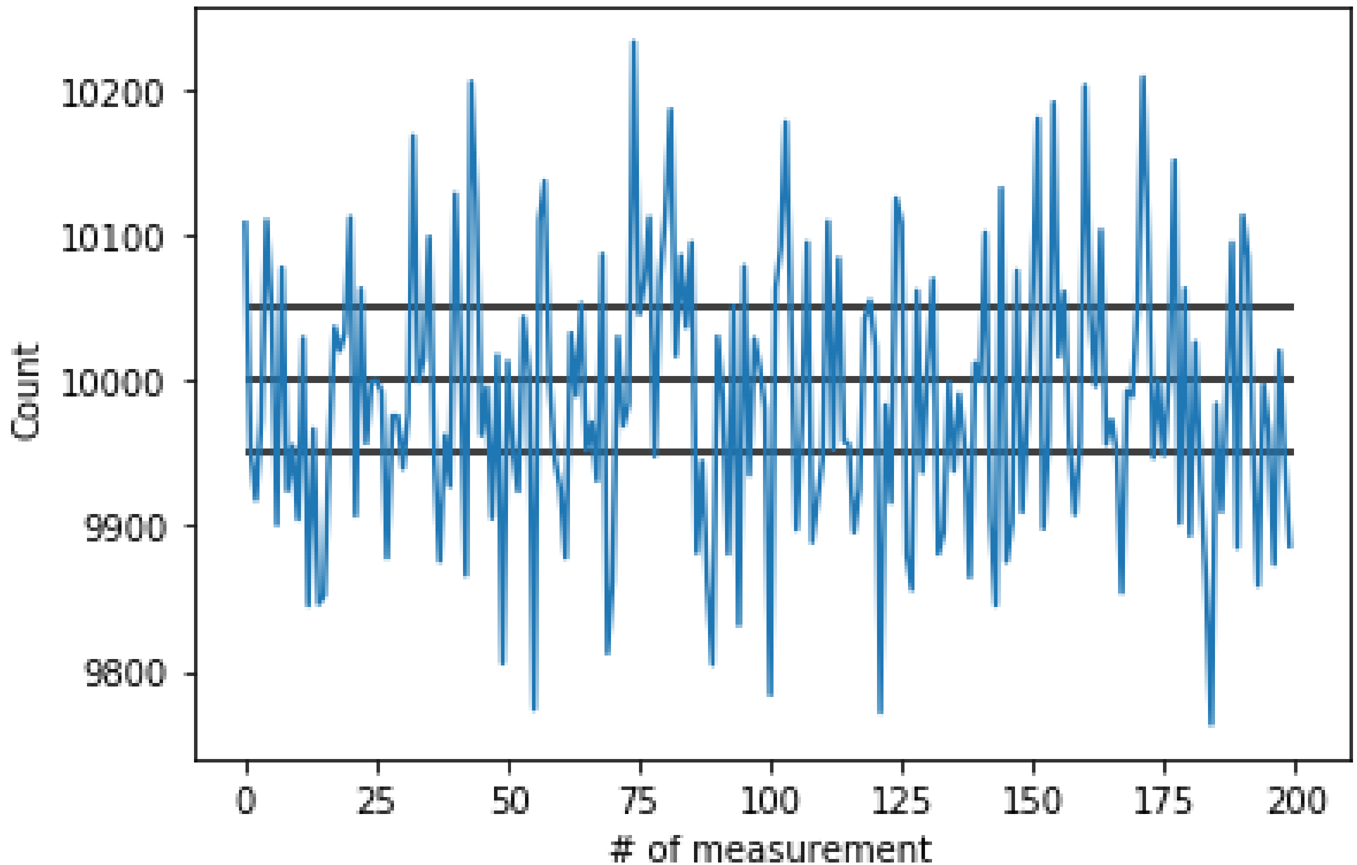
- In this case, these 20 measurements (random variables) are our observations
- (Unknown) constant value is our model.
- **And so are the measurement uncertainties.**

Let’s plot this



Which line is the closest to the “true” value?

Is it a bit easier now?



Ok, what is our model here?

$$y_i = y_{\text{true}} + \epsilon_i$$

Measured value

"True" value – a constant

Uncertainty – random!

Because of the uncertainty, our measured values are also random!

Our uncertainty (noise) is, most of the time, Gaussian:

$$p(\epsilon_i) = \frac{1}{\sqrt{\pi}\sigma_i} e^{-\epsilon_i^2/\sigma_i^2}$$

So, we see that probability of getting a certain measurement is:

$$p(y_i|y_{\text{true}}) = \frac{1}{\sqrt{\pi}\sigma_i} e^{-(y_i - y_{\text{true}})^2/\sigma_i^2}$$

And the whole set:

$$p(\mathbf{y}|y_{\text{true}}) = \prod_i p_i$$

What are we looking for

- We want to find the y_{true} that maximizes:

$$p(\mathbf{y}|y_{\text{true}}) = \prod_i p_i$$

What are we looking for

- We want to find the y_{true} that maximizes:

$$p(\mathbf{y}|y_{\text{true}}) = \prod_i p_i$$

Or do we? Let's read what this means:

Probability of getting the set of measurements, given the true value y_{true} is ...

We do not want that!

- To illustrate that this is a wrong function to maximize, usually disease examples are used. We do not want that. Let's come up with a different example.

  "A pack of cashews was found missing from NSO. A print of Onitsuka tiger shoes was found next to the cupboard…"

The Bride wears the Tigers 100% of time

The Bride wears the Tigers 100% of time

Your lecturer wears the Tigers 30% of time

Who stole the Cashews!?!?

The Bride wears the Tigers 100% of time

Your lecturer wears the Tigers 30% of time

Who stole the Cashews!?!?

Let’s write down the probabilities in Asics notation

$$p(\text{Tigers}|\text{Ivan}) = 0.3$$

$$p(\text{Tigers}|\text{The bride}) = 1.0$$

But what we actually need is:

$$p(\text{The bride}|\text{Tigers}) = ?$$

$$p(\text{Ivan}|\text{Tigers}) = ?$$

How do we calculate this, what
do we need to do?

Ok let's abandon Asics notation and discuss Bayes theorem

Probability of the data given the model - **likelihood**

Probability of the model before the measurement - **prior**

$$p(M|D) = \frac{p(D|M)p(M)}{p(D)}$$

Probability of the model given the data – **posterior**

Probability of the data for all the models – normalizing factor

How do we use Bayes theorem?

- We can use it to compare probabilities of the two discrete events (who stole the Cashews?)
- We can use it to find the most probable values of the parameters (i.e. to infer a quantity)
- We can use it to compare different models (e.g. linear vs quadratic)
- We can do many things
- Let’s use it to solve some of the problems we were facing

Missing cashews

1.0 ~ 0

$$p(\text{The bride}|\text{Tigers}) = \frac{\text{p}(\text{Tigers}|\text{The bride})\text{p}(\text{The bride})}{\text{const}}$$

~ 0

0.3 ~ 1.0

$$p(\text{Ivan}|\text{Tigers}) = \frac{\text{p}(\text{Tigers}|\text{Ivan})\text{p}(\text{Ivan})}{\text{const}}$$

~ 1

Our measurement problem

$$p(y_{\text{true}}|\mathbf{y}) = \frac{p(\mathbf{y}|y_{\text{true}})p(y_{\text{true}})}{p(\mathbf{y})}$$

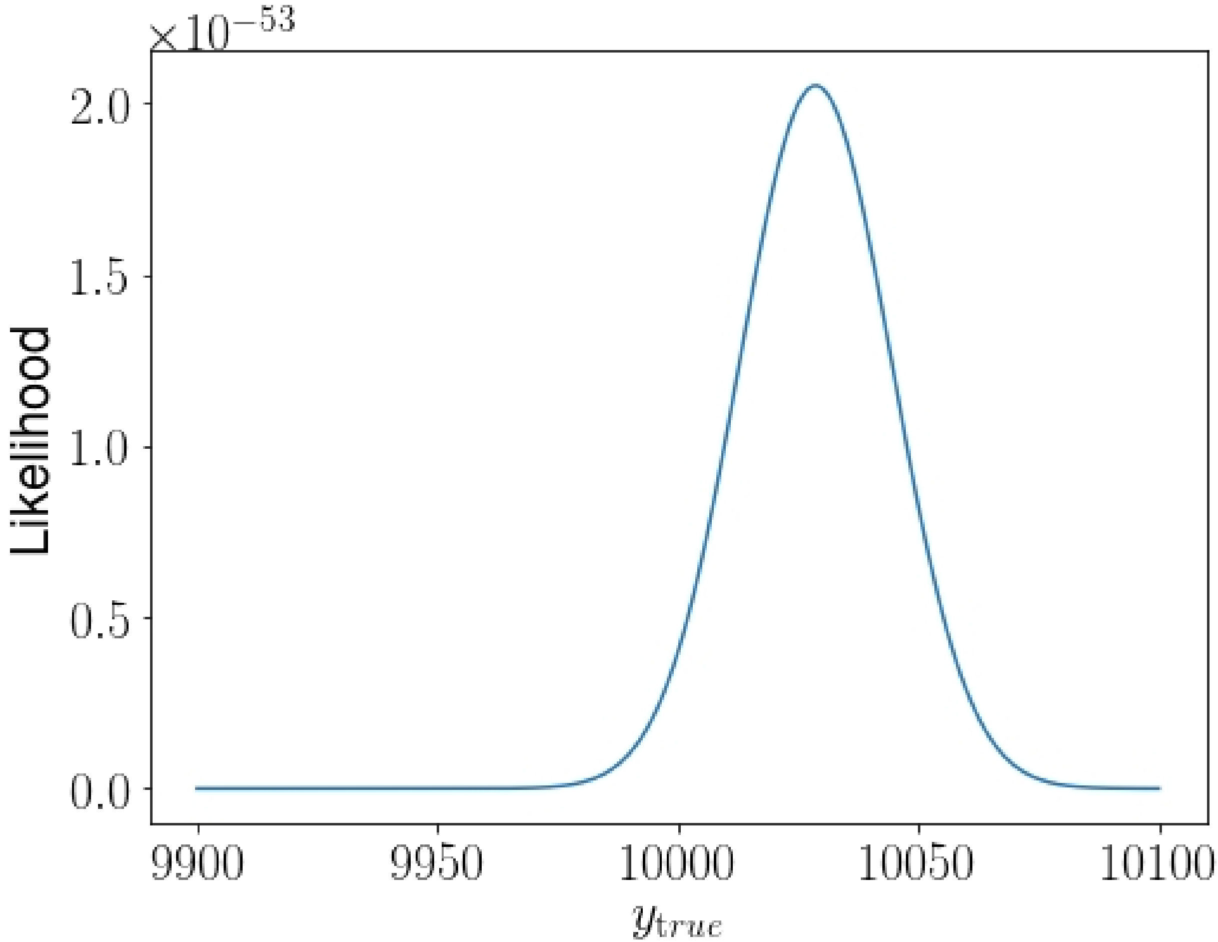
- p(**y**) is just a normalizing factor, we can neglect it now
- p(y_true) is interesting, let’s assume that we know nothing about it an all values are equally probable (so called “uniform” prior)
- However, some values are impossible due to their physical meaning
- If prior is uniform enough, posterior and likelihood have the same maximum in the y_true space.

Next step – fitting

- Ok, cool – now we know we want to find the maximum likelihood
- What are we actually doing? We are looking for the maximum of the likelihood function in 1D space where y_true lives.
- Keep in mind, no matter how many measurements you have, you are searching for the maximum in the model space!
- That is why most fitting problems are actually optimization problems

$$p(\mathbf{y}|y_{\text{true}}) = \prod_i \frac{1}{\sqrt{\pi}\sigma_i} e^{-(y_i - y_{\text{true}})^2/\sigma_i^2}$$

So, let's take a grid of values in a reasonable range and see



What we did here was "sampling" - we probed a set of possible values and sketched the probability distribution

Maximizing likelihood – minimizing chi-squared

- From the maximum likelihood we immediately get the minimum chi-squared

$$\mathcal{L}(y_{\text{true}}) = p(\mathbf{y}|y_{\text{true}}) = \prod_i \frac{1}{\sqrt{\pi}\sigma_i} e^{-(y_i - y_{\text{true}})^2/\sigma_i^2}$$

$$log\mathcal{L} = \text{const} - \sum_{\text{i}} \frac{(\text{y}_\text{i} - \text{y}_{\text{true}})^2}{\sigma_\text{i}^2}$$

Or, more generally:

Value that model parameters predict

$$\chi^2(\mathbf{M}) = \sum_i \frac{(y_i - f(x_i, \mathbf{M}))^2}{\sigma_i^2}$$

Model parameters

Some things to know

- Chi-squared minimization is strictly proper when our priors are uniform and noise is Gaussian
- That is often the case, mostly because we don't know better
- Can you think of some situations when priors are not uniform and noise is not gaussian?
- Chi-squared is also used for model assessment – should be unity (but...)

$$\chi^2_{\mathrm{reduced}} = \frac{\chi^2}{N_{\mathrm{measurements}} - N_{\mathrm{parameters}}}$$

Numerical methods

- Minimizing chi-squared is a numerical problem, usually solved by some sort of numerical minimization
- You will most likely want to use your favorite python minimization / curve fitting tool to do this.
- E.g. scipy.optimize.minimize will do a good job
- It is a good practice to code your own sometimes
- If you use very specialized models you might have to
- There is also “sampling” - we will go back to this soon!

Linear models

- What is a linear model? Can you give me some examples?

Linear models

- What is a linear model? Can you give me some examples?
- That is correct, linear models are the models that are **linear in the parameters** , the relationship between x and y does not have be linear.

Linear: $$y = ax^2 + bx + c + de^x$$

Non-linear: $$y = ax^2 + \sqrt{a}x$$

Linear models

- Linear model fitting is literally solving a linear system of equations:

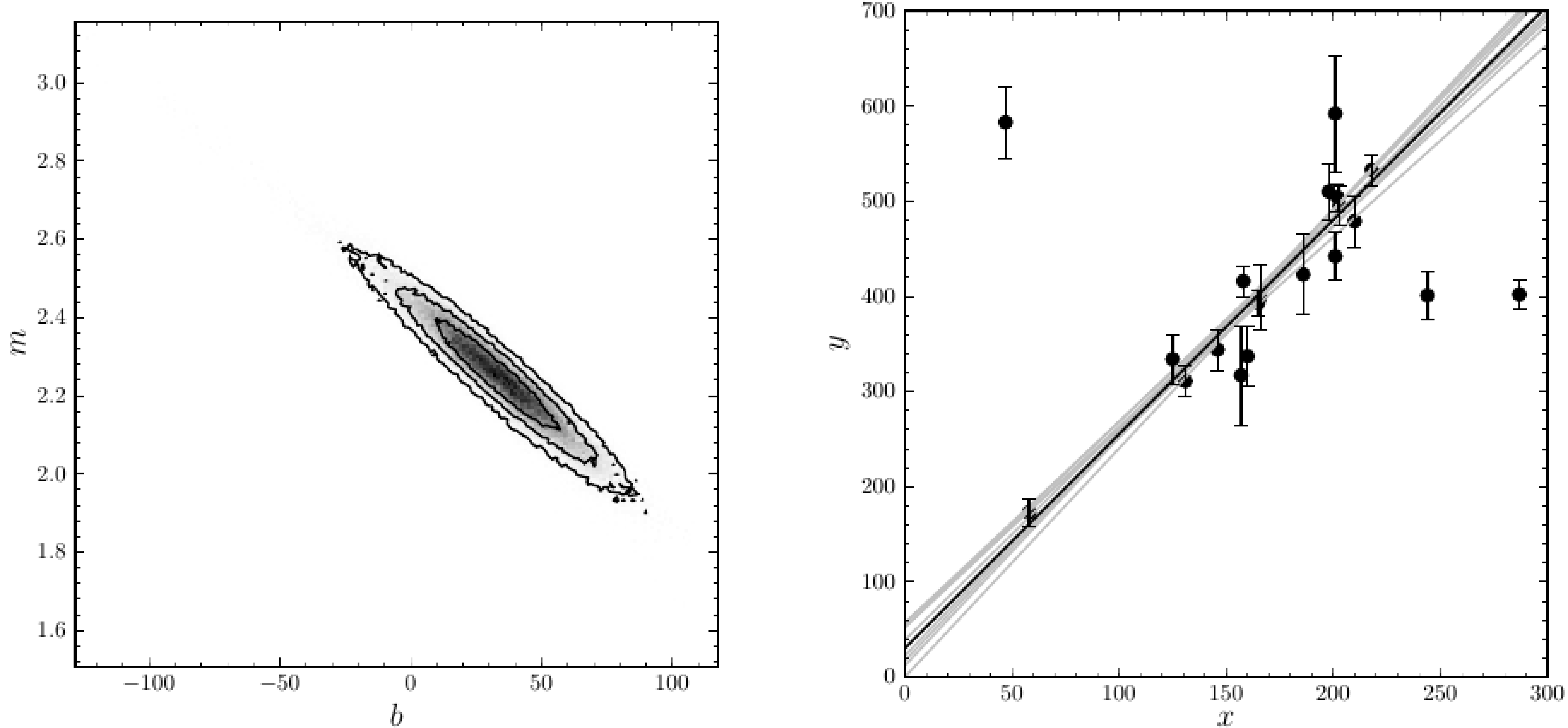
$$y_1 = kx_1 + m$$
$$y_2 = kx_2 + m$$
$$\vdots$$
$$y_n = kx_n + m$$

- Solving this linear system using a pseudo-inverse guarantees chi-squared minimization (max likelihood)

But, if you can afford it – it is still better to sample

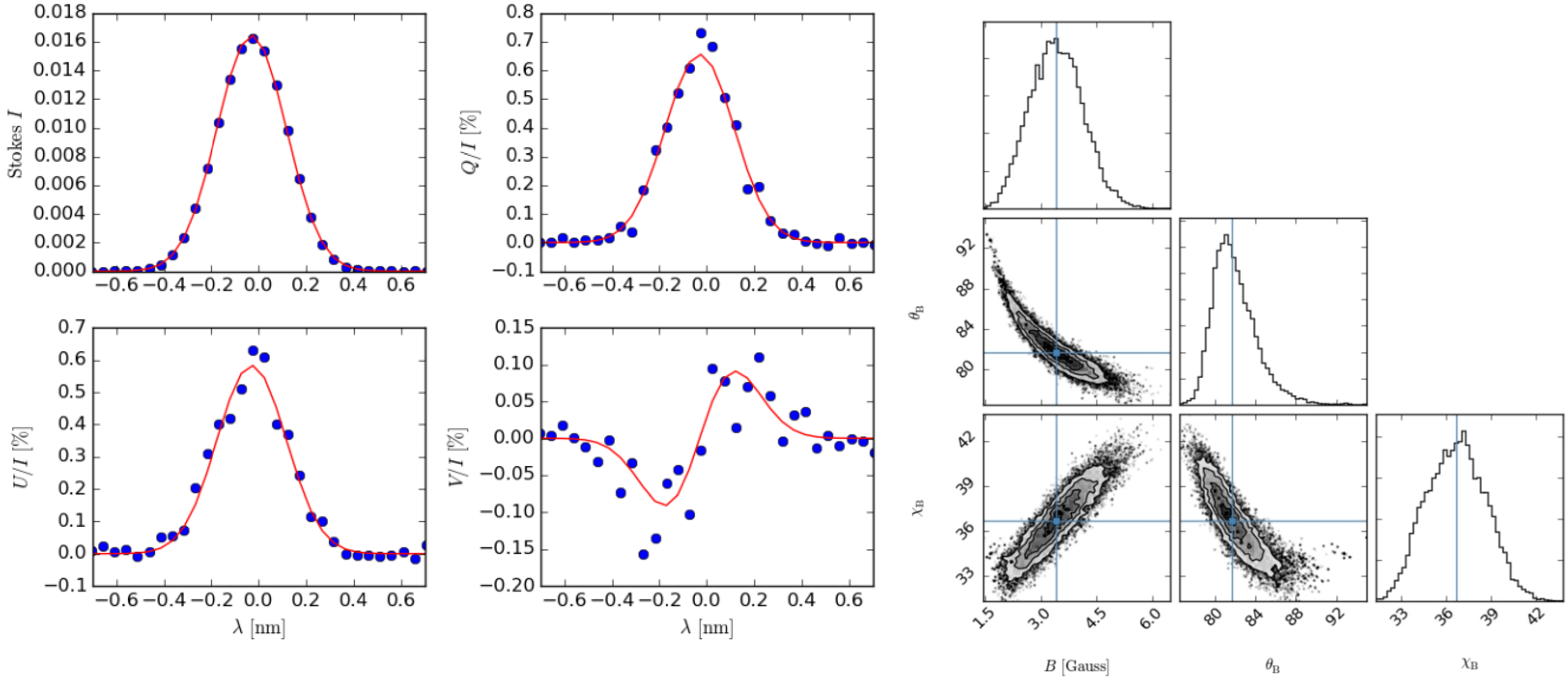
- Sampling, that is : probing your parameter space gives you insight in the full shape of your chi-squared surface
- This way you can better explore degeneracies (correlations), estimate uncertainties, spot multiple minima, etc.
- Uncertainties are essential
- They allow us to asses the strength of our conclusions, and to compare different datasets, results, etc.

Example results obtained by sampling



Hogg et al. “Fitting a model to the data” , 2010 arxiv e-prints

Example results obtained by sampling

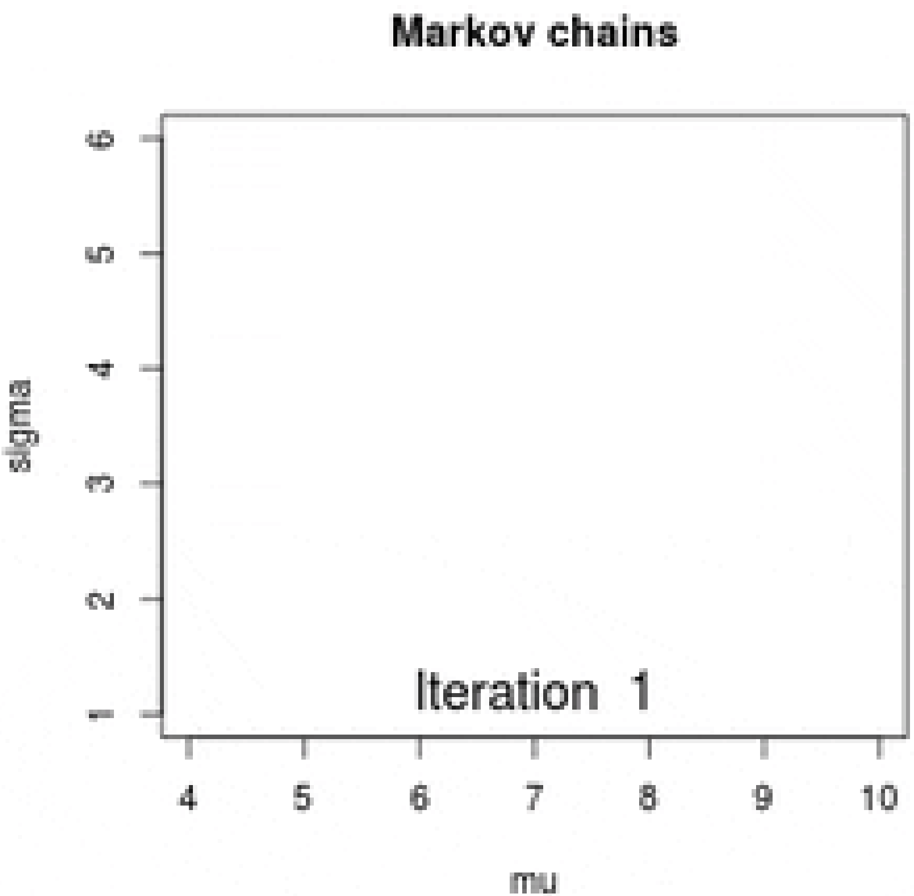


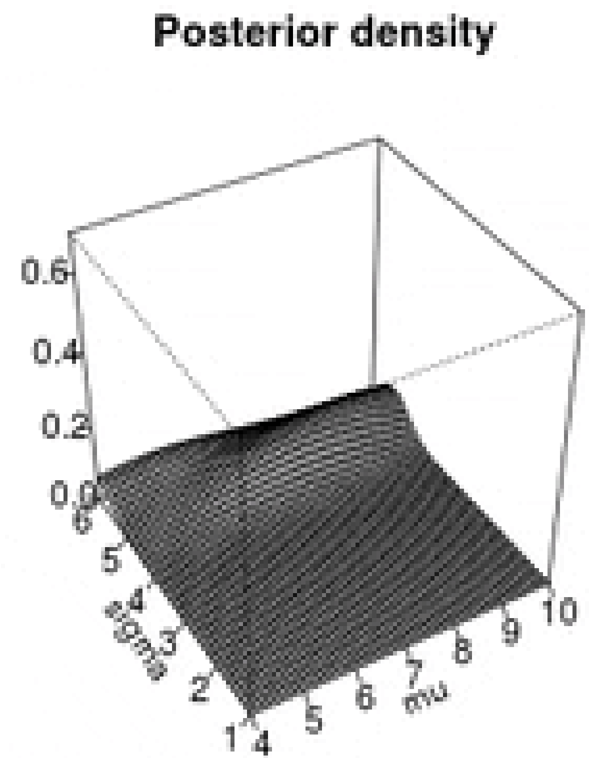
Milic et al. (2014) – a non-linear model

How do we get these? How do we sample?

- MCMC (Hammer) – Marko Chain Monte Carlo
- Codes that travel in a clever way through the phase space (space of parameters)
- The “walker” will visit points with high probability more often
- The plots that we saw are density plots of the walkers
- Easy to code (at least in the basic form)
- Works for all linear and non-linear functions
- Takes a lot of time (we need a lot of points for good statistics)

How does MCMC work?





What does it mean to be Bayesian

- Being Bayesian means being objective – you might be a Bayesian without knowing it!
- It means taking care of priors
- It means looking at the shape of your posterior
- It also means **marginalizing over nuisance parameters**
- What is this?

Nuisance parameters

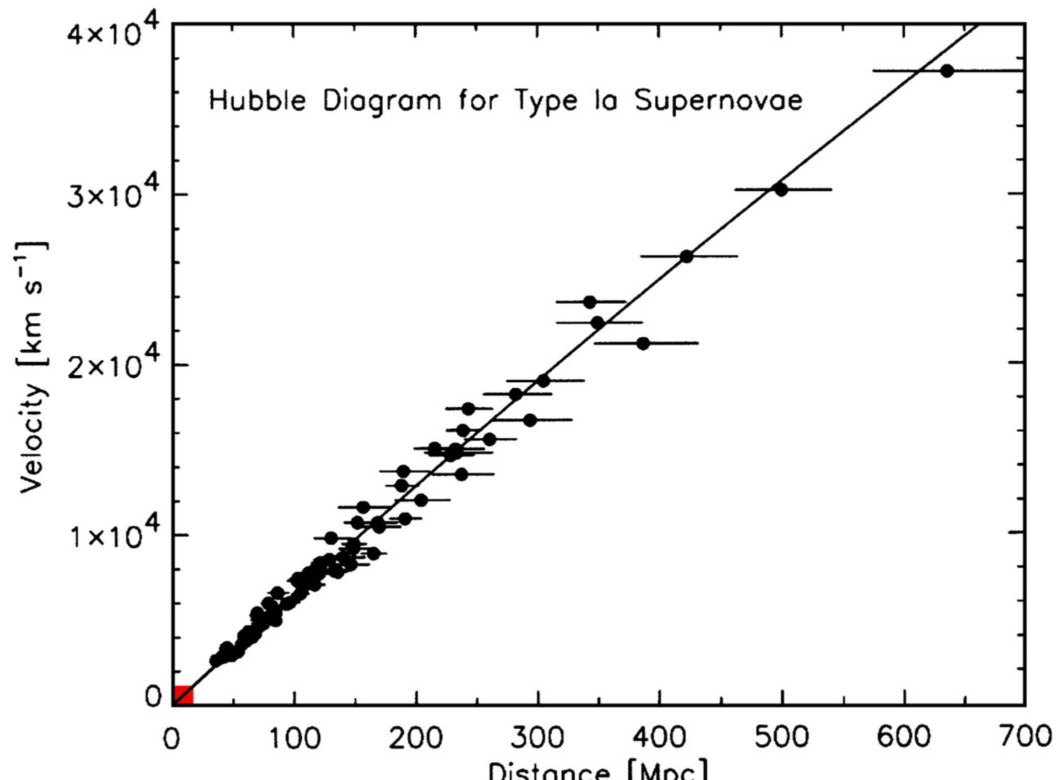
- Parameters that are needed for the fit, but are not important for answering our scientific question.

- Example: I am fitting v(d) dependency to determine Hubble's constant

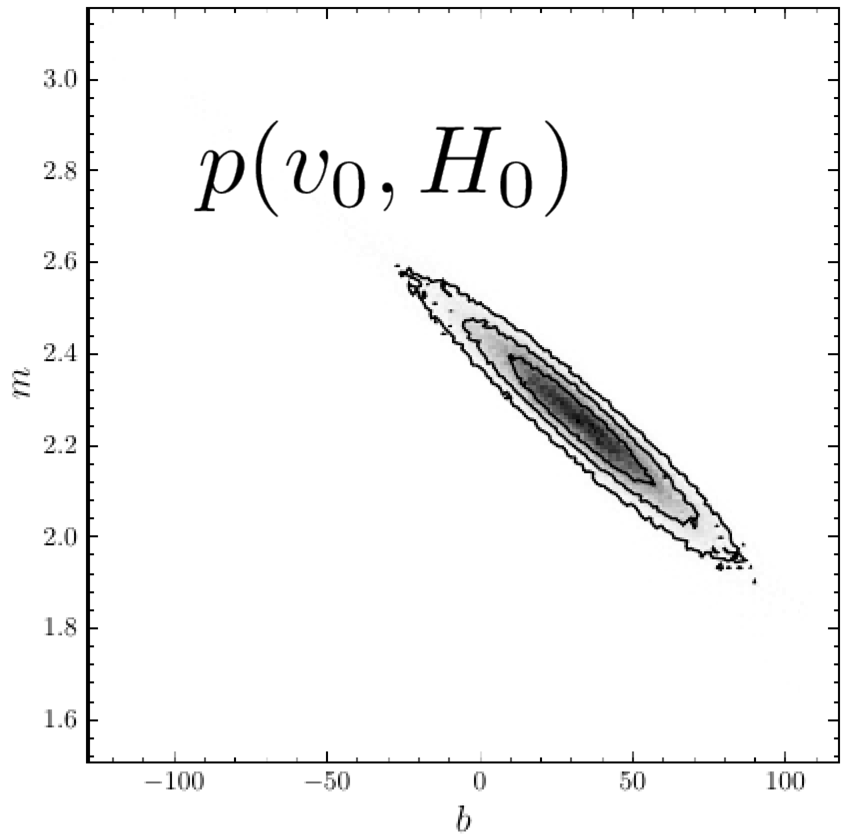
Slope – what we are really interested into

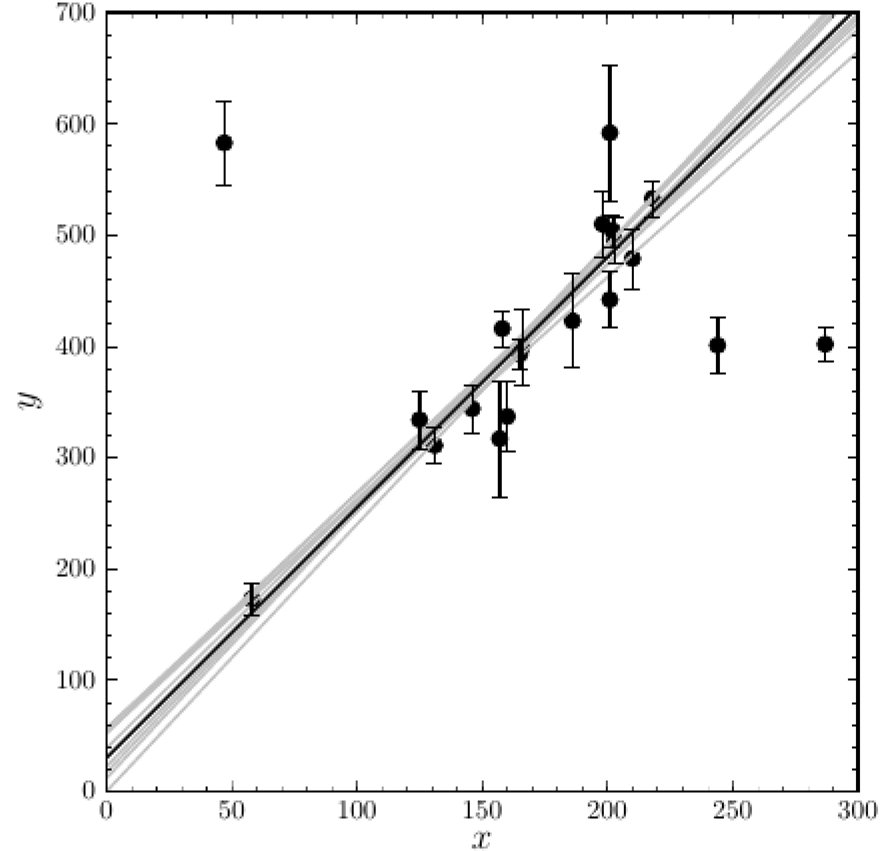
$$v = v_0 + H_0 d$$

Offset – can be there because of different reasons



After fitting I am going to get something like this





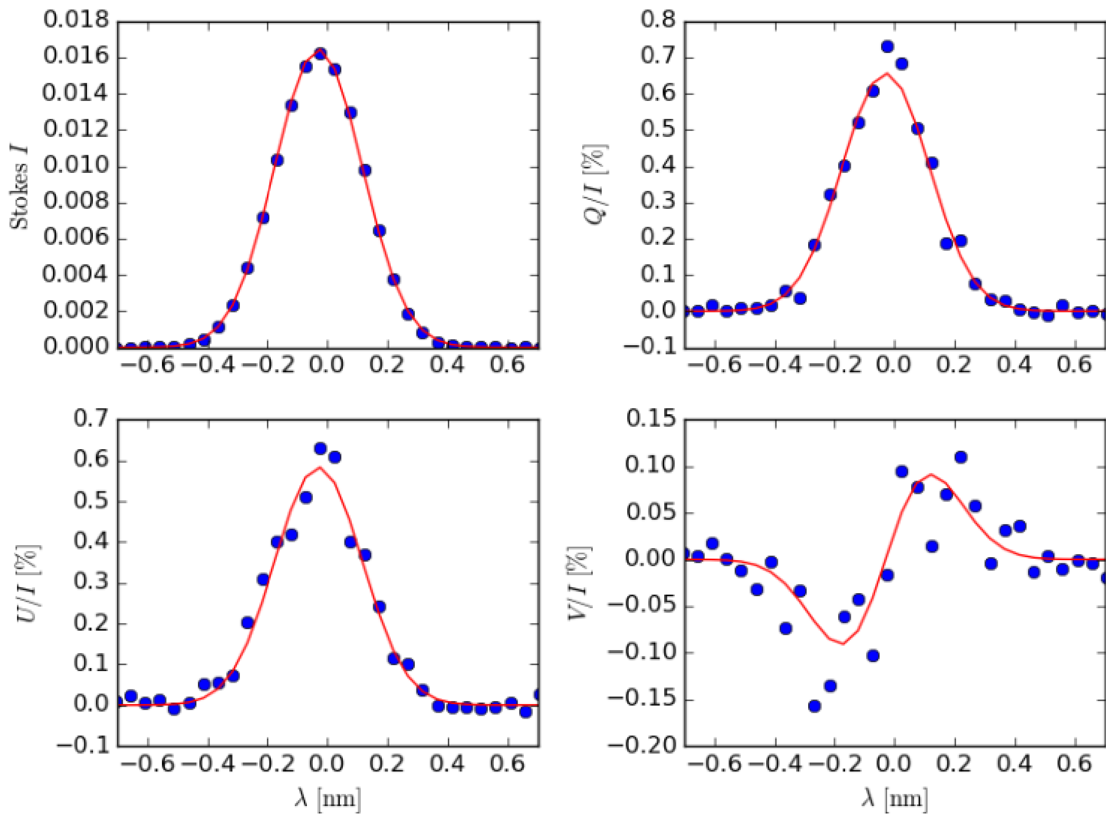
Hogg et al. “Fitting a model to the data” , 2010 arxiv e-prints

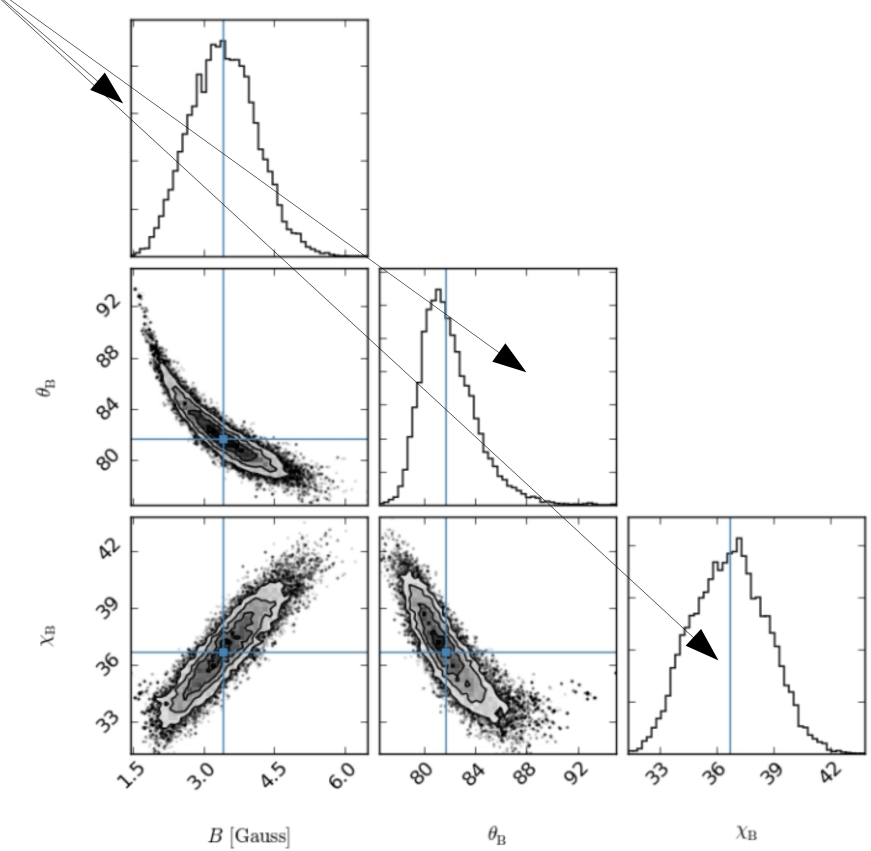
Now, to get my final results, I marginalize over the nuisance parameters

$$p(H_0) = \int_{-\infty}^{\infty} p(v_0, H_0) dv_0$$

Marginalizing results of MCMC chains

- Just ignore the axes that are nuisance parameters!





Milic et al. (2014) – a non-linear model

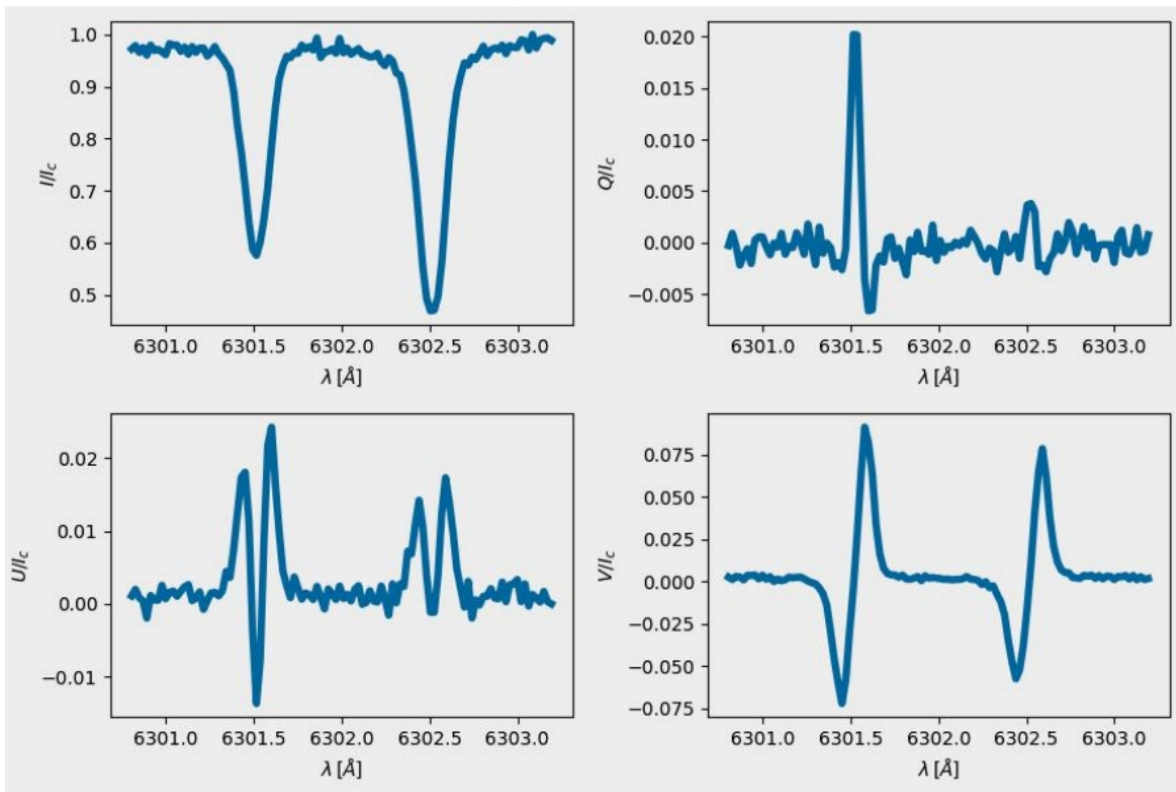
Summary

- Model fitting necessitates having a model, that (imho) should be motivated by the physics of your problem
- Sometimes it can be very simple (i.e. weak field approximation), sometimes it will be very complicated (full scale inversion)
- You have to maximize the posterior probability, that in case of uniform priors and Gaussian errors reduces to minimizing Chi-squared
- You can simply optimize to find **parameter values that minimize your chi-squared.**
- **But you can also "sample" and obtain full shape of posterior.**
- For next week: https://emcee.readthedocs.io/en/stable/

Solar physics examples – a linear model

- Weak field approximation predicts a relationship between Stokes I and V

$$V(\lambda_i) = 4.67 \times 10^{-13} \times (\frac{dI}{d\lambda})_i \times B \times g_L \times \lambda_0^2$$



Solar physics examples – a non linear model - “inversion”