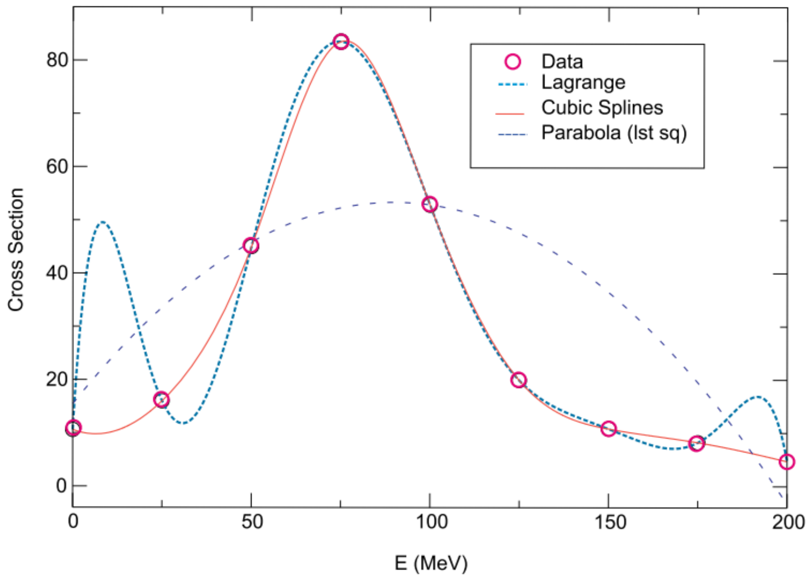
Interpolation vs regression

- Interpolation: find a continuous function that passes through all the data points ("join the dots")
  - Typical use: change sampling, resolution
  - Formula of the fitting function is not interesting
- Fitting, regression: find the best parameters of a known model function, that is closest to data points
  - Typical use: find parameters of a physical model
  - Sometimes the model is just approximation



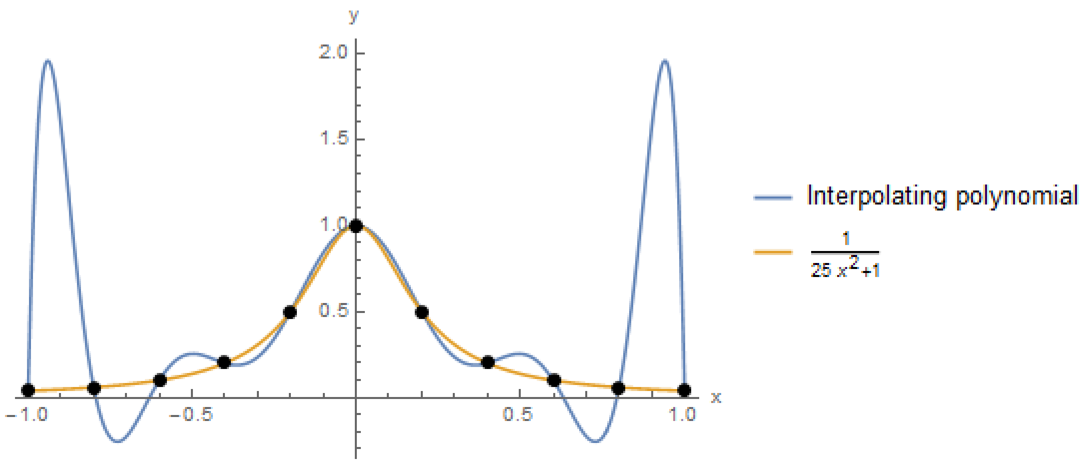
Interpolation

- Lagrange's formula
  - Polynomial goes through **all** points

Through any two points there is a unique line. Through any three points, a unique quadratic. Et cetera. The interpolating polynomial of degree $N-1$ through the N points $y_1 = f(x_1), y_2 = f(x_2), \ldots, y_N = f(x_N)$ is given explicitly by Lagrange's classical formula,

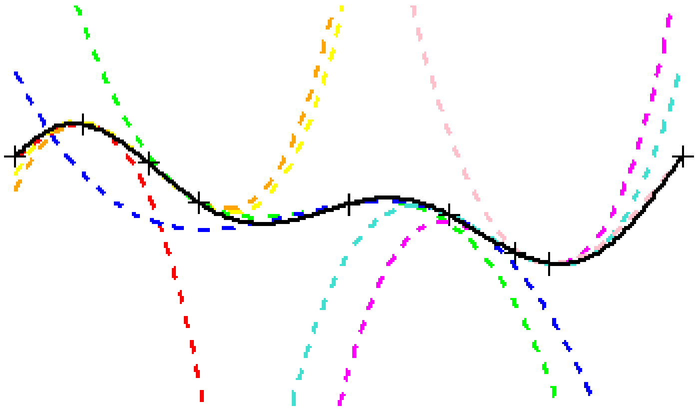
$$P(x) = \frac{(x-x_2)(x-x_3)\ldots(x-x_N)}{(x_1-x_2)(x_1-x_3)\ldots(x_1-x_N)}y_1 + \frac{(x-x_1)(x-x_3)\ldots(x-x_N)}{(x_2-x_1)(x_2-x_3)\ldots(x_2-x_N)}y_2$$
$$+\cdots+\frac{(x-x_1)(x-x_2)\ldots(x-x_{N-1})}{(x_N-x_1)(x_N-x_2)\ldots(x_N-x_{N-1})}y_N \qquad (3.1.1)$$

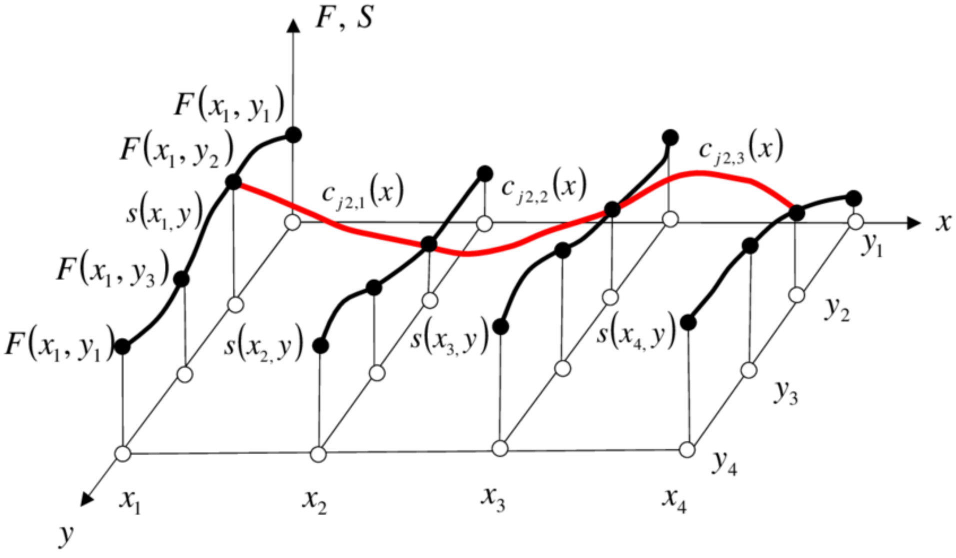
There are N terms, each a polynomial of degree $N-1$ and each constructed to be zero at all of the x_i except one, at which it is constructed to be y_i.



Interpolation

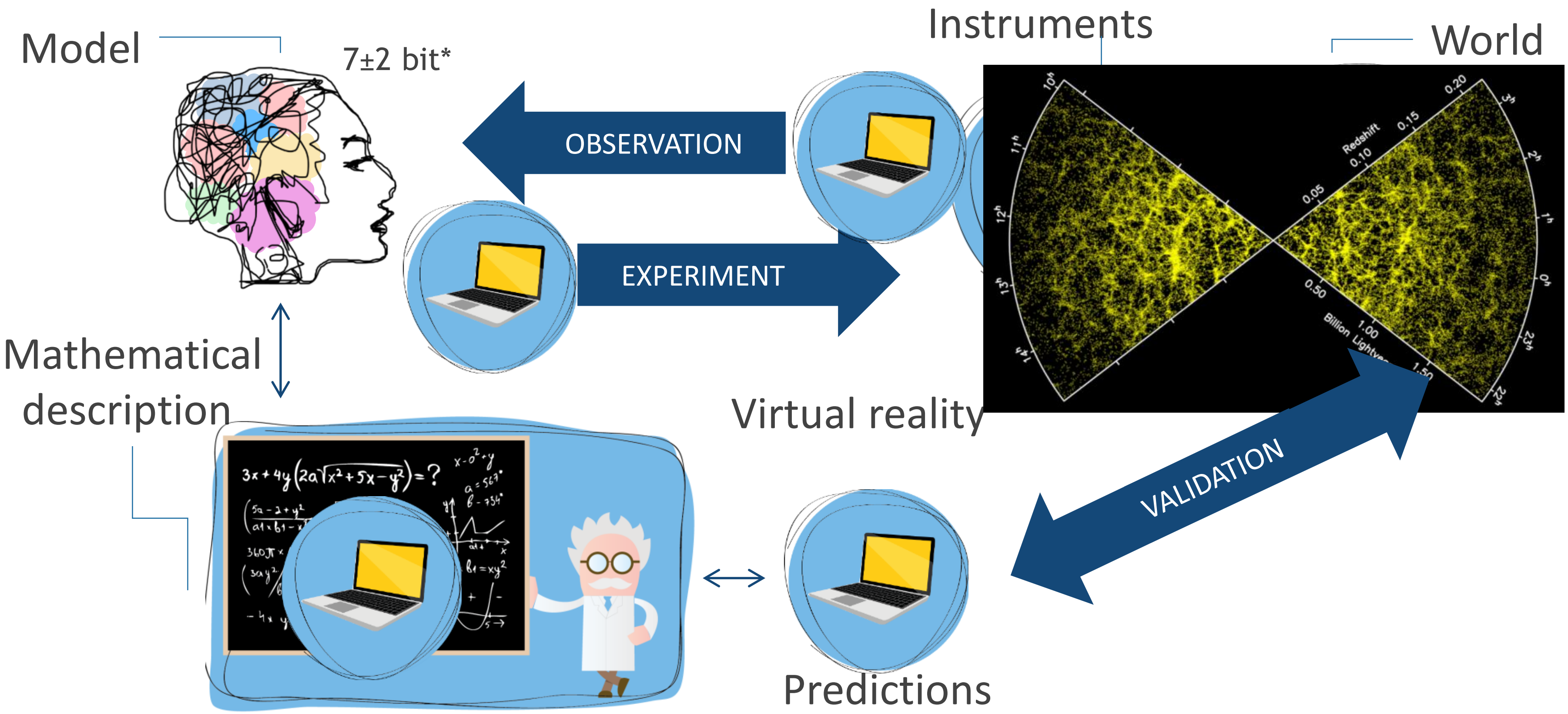
- Cubic spline:
  - Local: low order polynomial through **neighboring** points
  - Local interpolators join smoothly to each other
  - Bi-cubic spline for 2D





Least Squares and Maximum Likelihood

Modern data science



Initial values

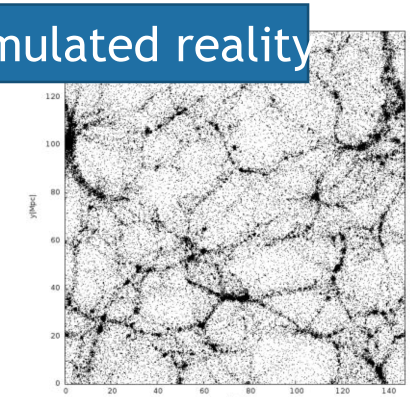
“laws”, equations

Simulated reality

Λ=0.7
Ω_m=0.3

$$F = G\frac{m_1 m_2}{r^2}$$

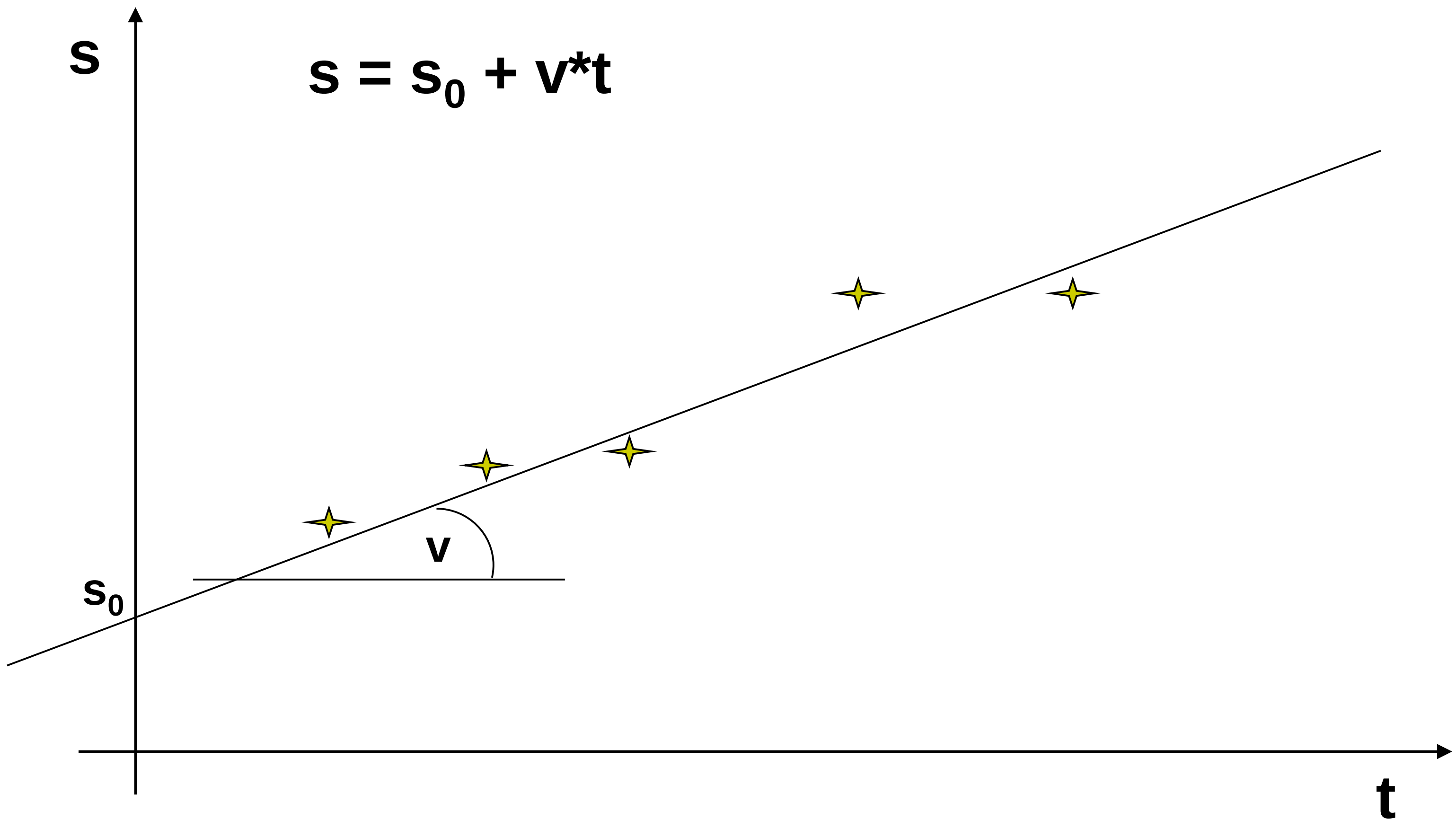
$$R_{\mu\nu} - \frac{1}{2} R\, g_{\mu\nu} + \Lambda g_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$$



Goal

- Find a MODEL for measurement/simulation data, that:
  - Fits some theory
  - Reproduce measurement data
  - Can be parametrized
- Parameters may have „physical” meaning
  - Displacement/time diagram – linear fit – slope: speed

Example: linear motion: speed



Errors

- Measurements contain **errors** (in each variables!)
  - noise, inaccuracy
  - Systematic errors
  - outliers
- Hence: the calculated parameters will have errors, too

Least square fit

Given: $(x_i, y_i);\ i=1..N$ measurement points. Fit some function :

$$y(x) = y(x; a_1, a_2, \ldots, a_M)$$

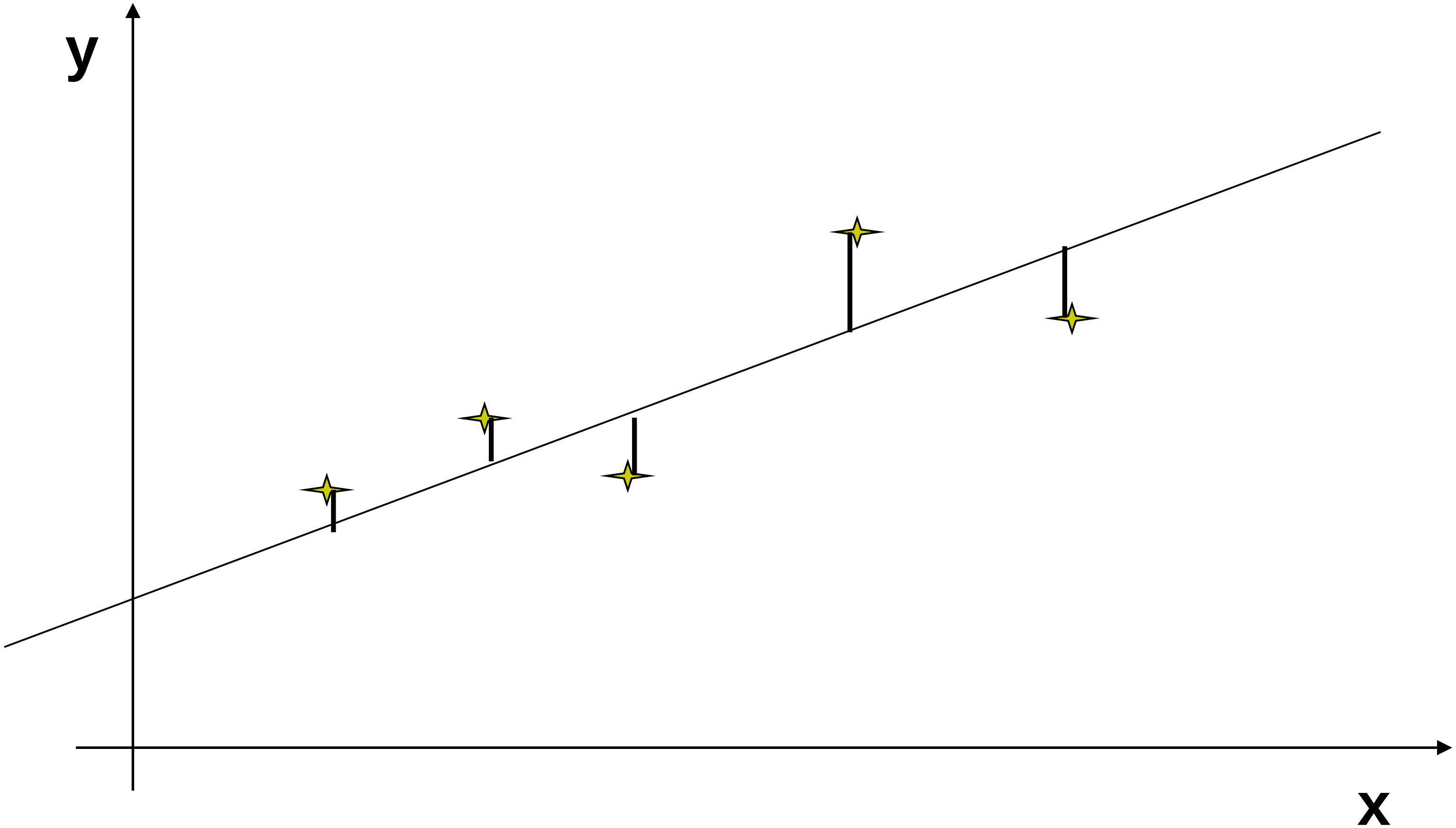
a_j parameters, $j=1..M$

Least square fit:

$$\min_{a_1 \ldots a_M} \left(\sum_{i=1}^{N} \left[y_i - y(x_i; a_1 \ldots a_M) \right]^2 \right)$$

Take derivatives by $a_1, a_2, \ldots, a_M$ -> set of linear equations (!! Also for some nonlin func.)

Least squares



Why not least absolute values, fourth powers, etc?

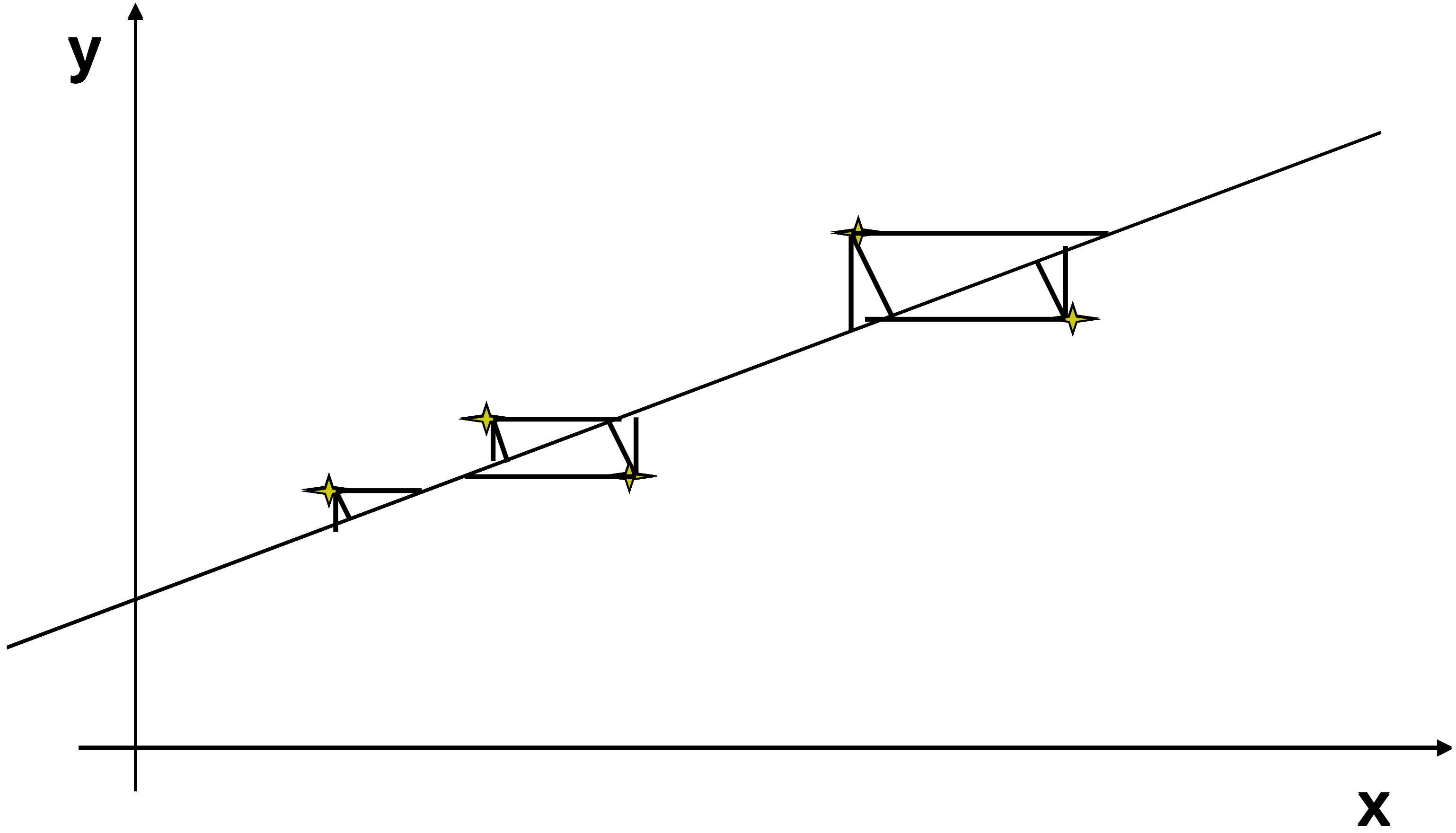
$$\min_{a_1 \ldots a_M} \left(\sum_{i=1}^{N} \left[y_i - y(x_i; a_1 \ldots a_M) \right]^2 \right)$$

$$\min_{a_1 \ldots a_M} \left(\sum_{i=1}^{N} \left| y_i - y(x_i; a_1 \ldots a_M) \right| \right)$$

$$\min_{a_1 \ldots a_M} \left(\sum_{i=1}^{N} \left[y_i - y(x_i; a_1 \ldots a_M) \right]^4 \right)$$

$$\min_{a_1 \ldots a_M} \left(\sum_{i=1}^{N} \sqrt[8]{\left| y_i - y(x_i; a_1 \ldots a_M) \right|} \right)$$

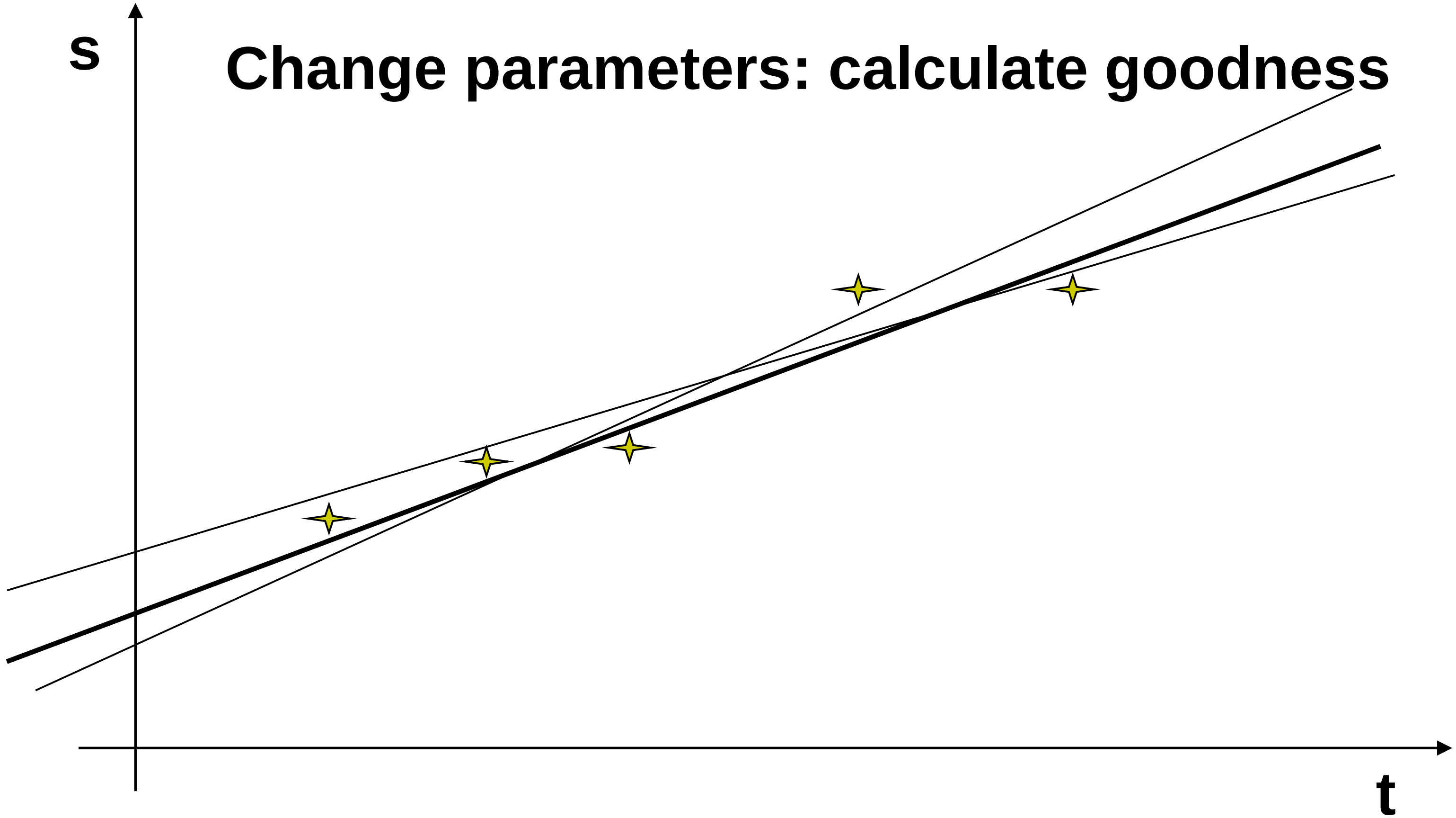
Least distances?



Why least squares?

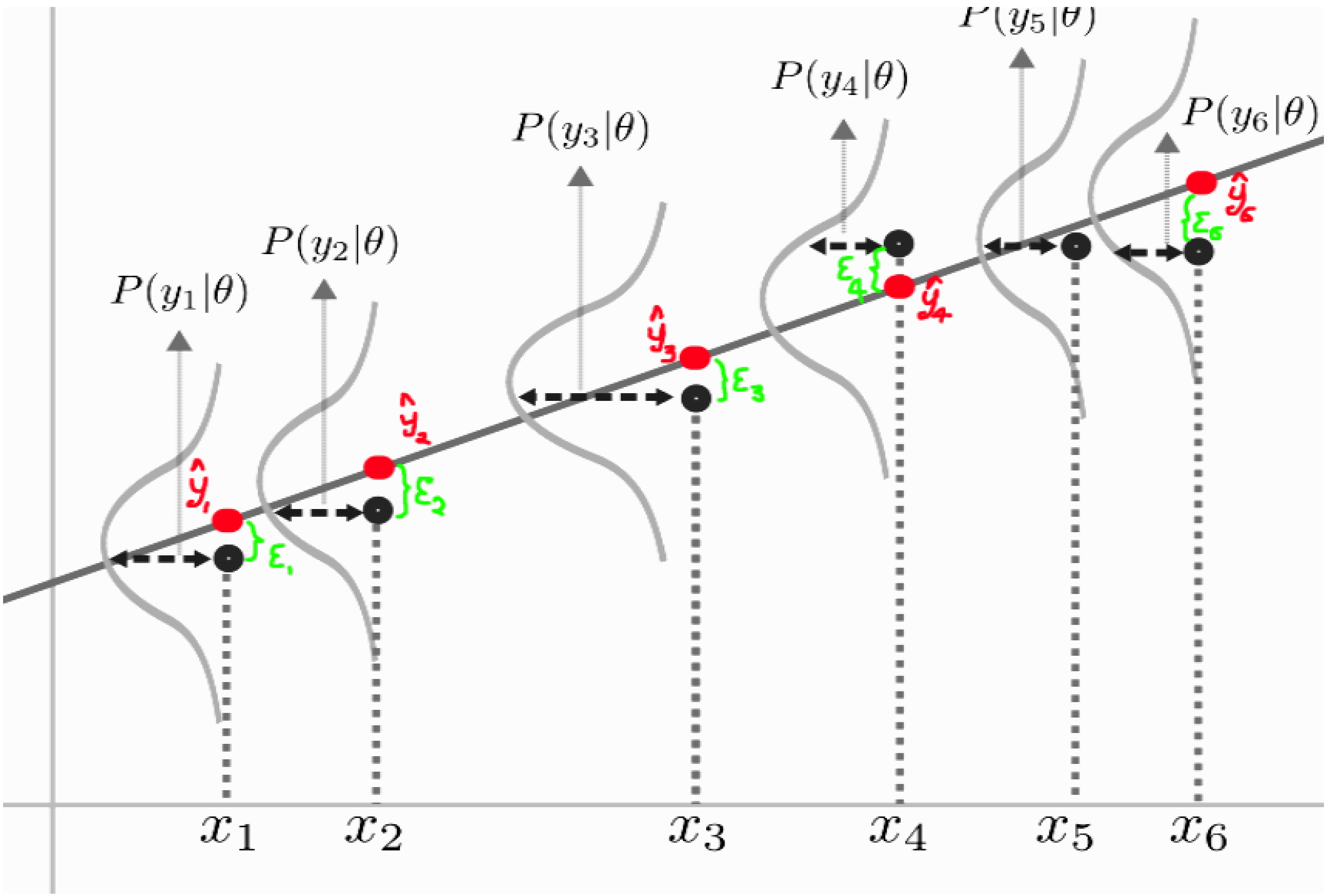
- Maximum likelihood estimation
- Least quare gives best parameters, if :
  - Errors only in y_i
  - Errors are normal (Gauss) distributed
  - Errors are independent

Cost function

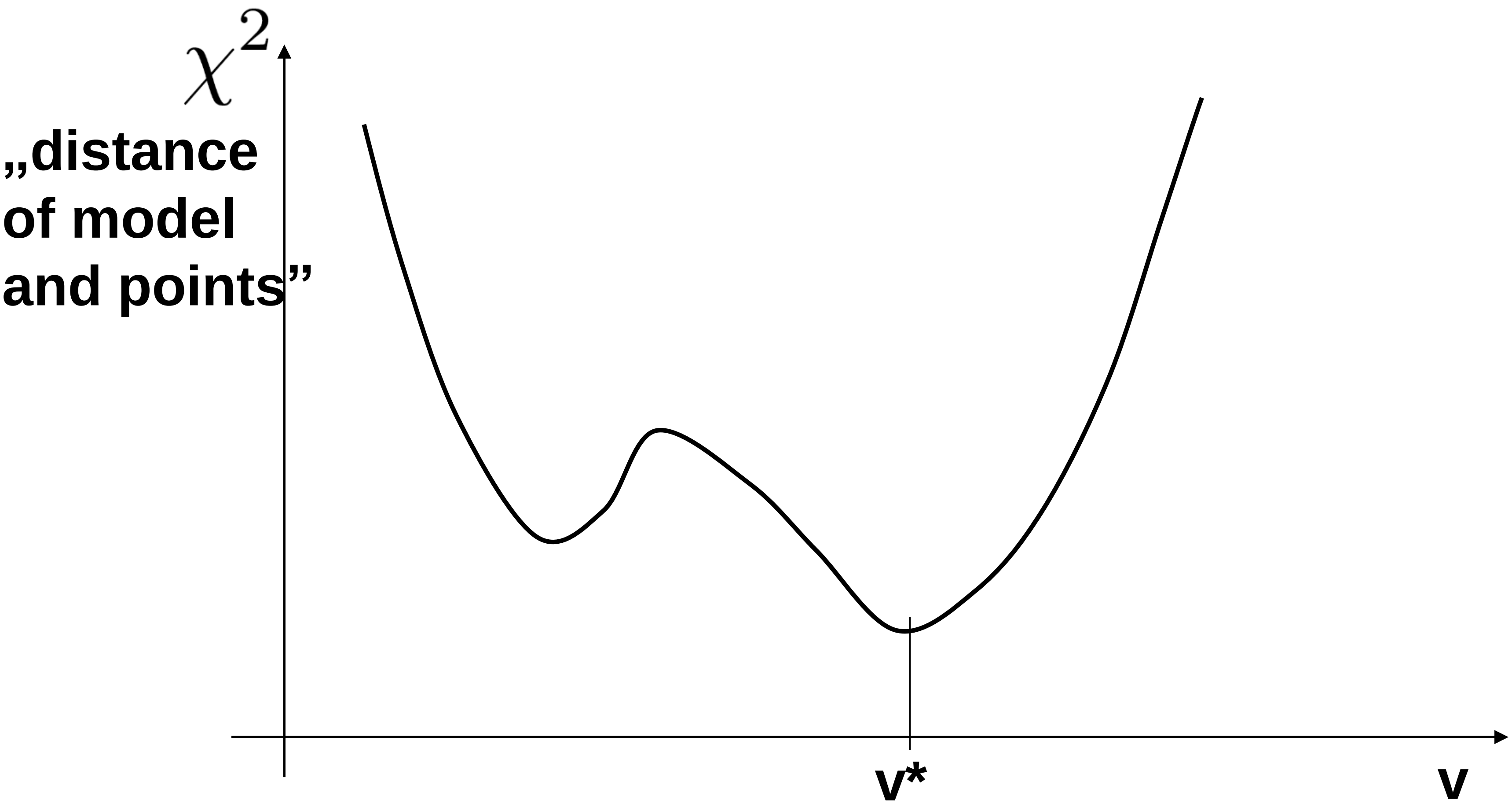


Find optimal parameters,
with largest goodness / smallest „cost/energy”

Maximum likelihood explained



Cost function



Finding minimum: in general leads to a nonlinear optimization problem

In special case: linear set of equations

Maximum likelihood

$$P \propto \prod_{i=1}^{N} \left\{ \exp\left[-\frac{1}{2} \left(\frac{y_i - y(x_i)}{\sigma} \right)^2 \right] \Delta y \right\}$$

Find parameters where P is maximal.

The log() func. is monotonuous, so find maximum of log(P), or minimum of –log(P)

`-log(P)=` $$\left[\sum_{i=1}^{N} \frac{[y_i - y(x_i)]^2}{2\sigma^2} \right] - N \log \Delta y$$

This is Least Squares!

Note

- Regression of a **nonlinear function** with **linear parameters** still leads to linear problem (Numerical Recipes 15.4)

$$y(x) = a_1 + a_2 x + a_3 x^2 + \cdots + a_M x^{M-1}$$

The general form of this kind of model is

$$y(x) = \sum_{k=1}^{M} a_k X_k(x)$$

$$\chi^2 = \sum_{i=1}^{N} \left[\frac{y_i - \sum_{k=1}^{M} a_k X_k(x_i)}{\sigma_i} \right]^2$$

The minimum of (15.4.3) occurs where the derivative of χ^2 with respect to all M parameters a_k vanishes. Specializing equation (15.1.7) to the case of the model (15.4.2), this condition yields the M equations

$$0 = \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left[y_i - \sum_{j=1}^{M} a_j X_j(x_i) \right] X_k(x_i) \qquad k = 1, \ldots, M \tag{15.4.6}$$

Interchanging the order of summations, we can write (15.4.6) as the matrix equation

$$\sum_{j=1}^{M} \alpha_{kj} a_j = \beta_k \tag{15.4.7}$$

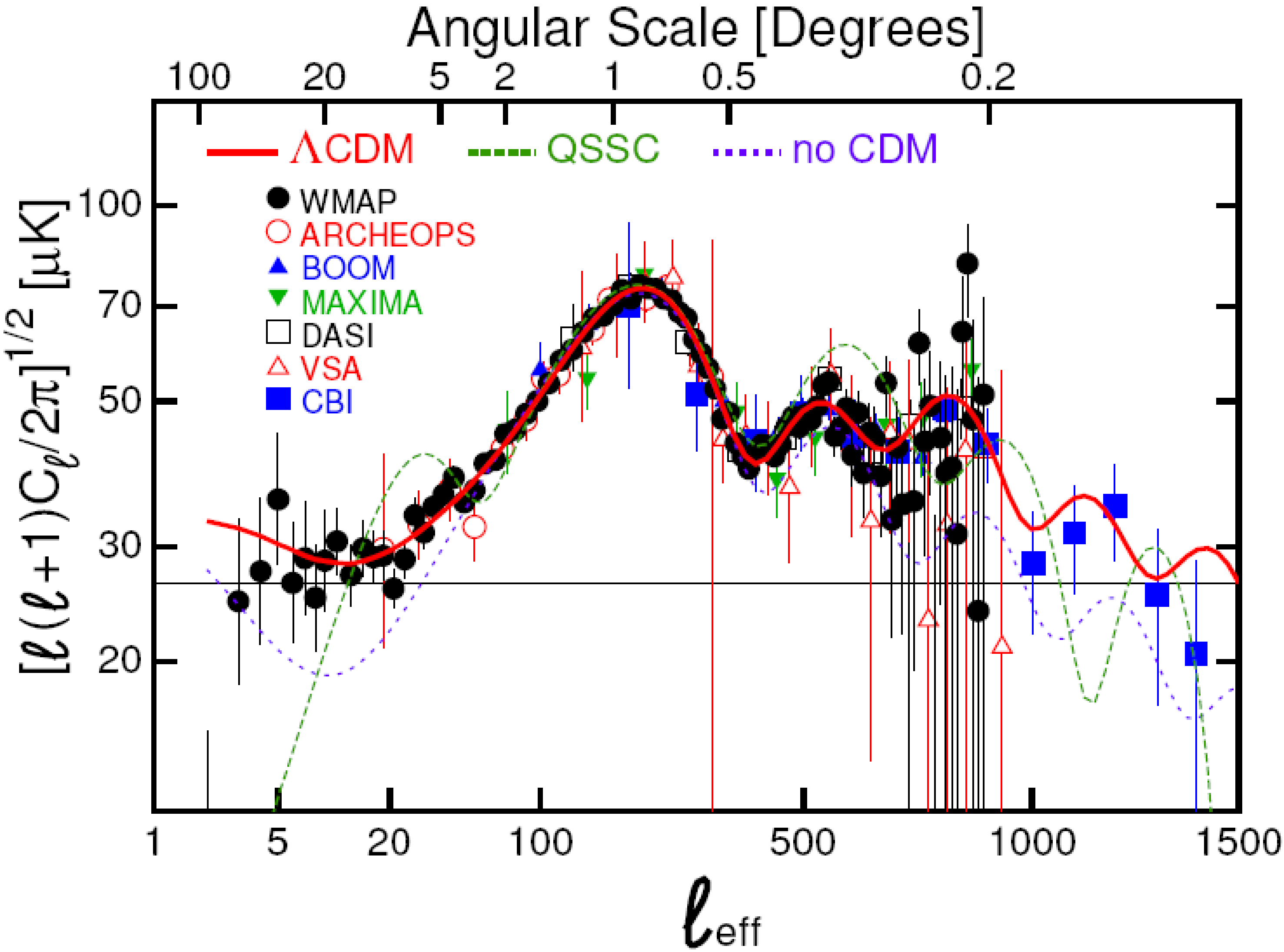
where

$$\alpha_{kj} = \sum_{i=1}^{N} \frac{X_j(x_i) X_k(x_i)}{\sigma_i^2} \qquad \text{or equivalently} \qquad [\alpha] = \mathbf{A}^T \cdot \mathbf{A} \tag{15.4.8}$$

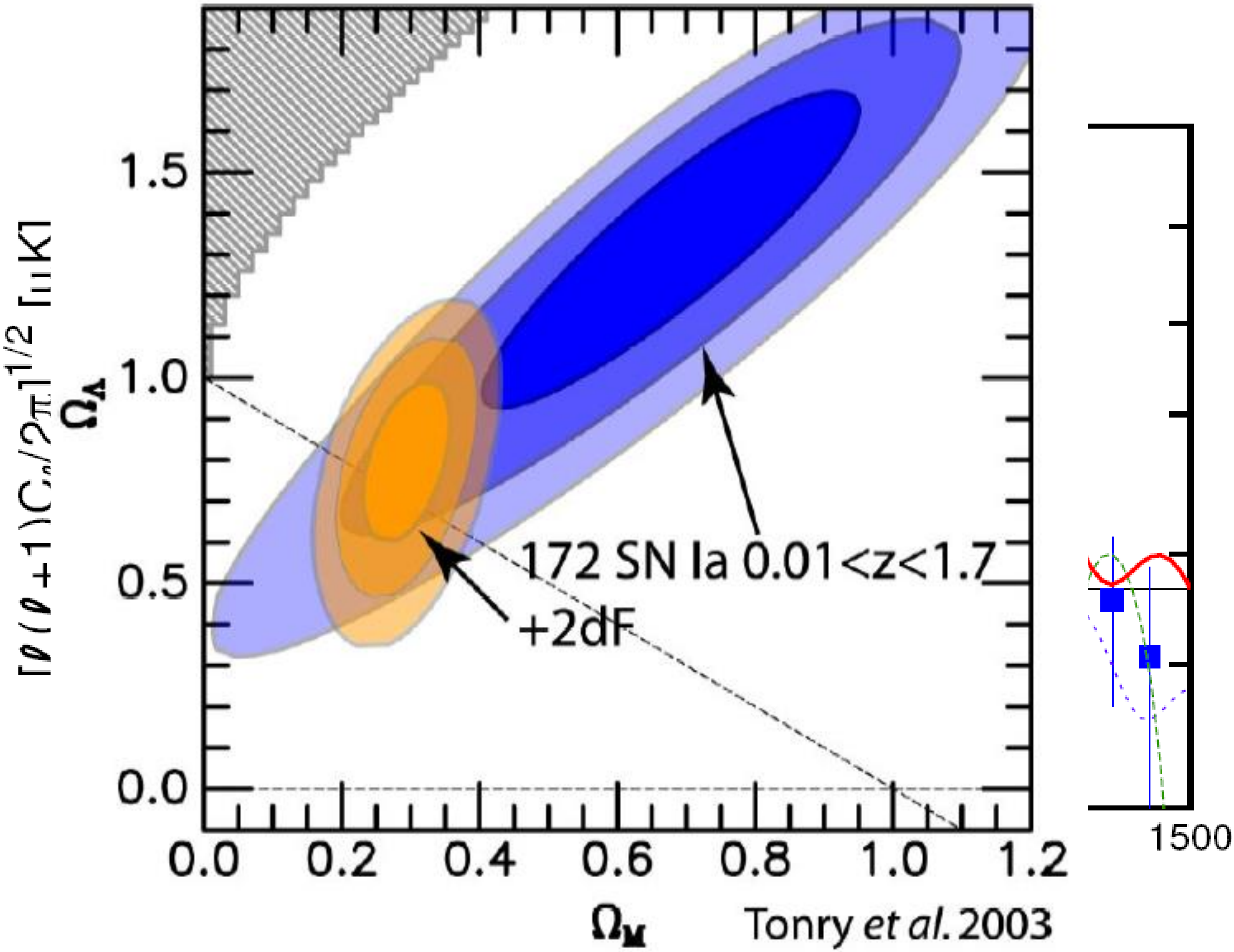
an $M \times M$ matrix, and

$$\beta_k = \sum_{i=1}^{N} \frac{y_i X_k(x_i)}{\sigma_i^2} \qquad \text{or equivalently} \qquad [\beta] = \mathbf{A}^T \cdot \mathbf{b} \tag{15.4.9}$$

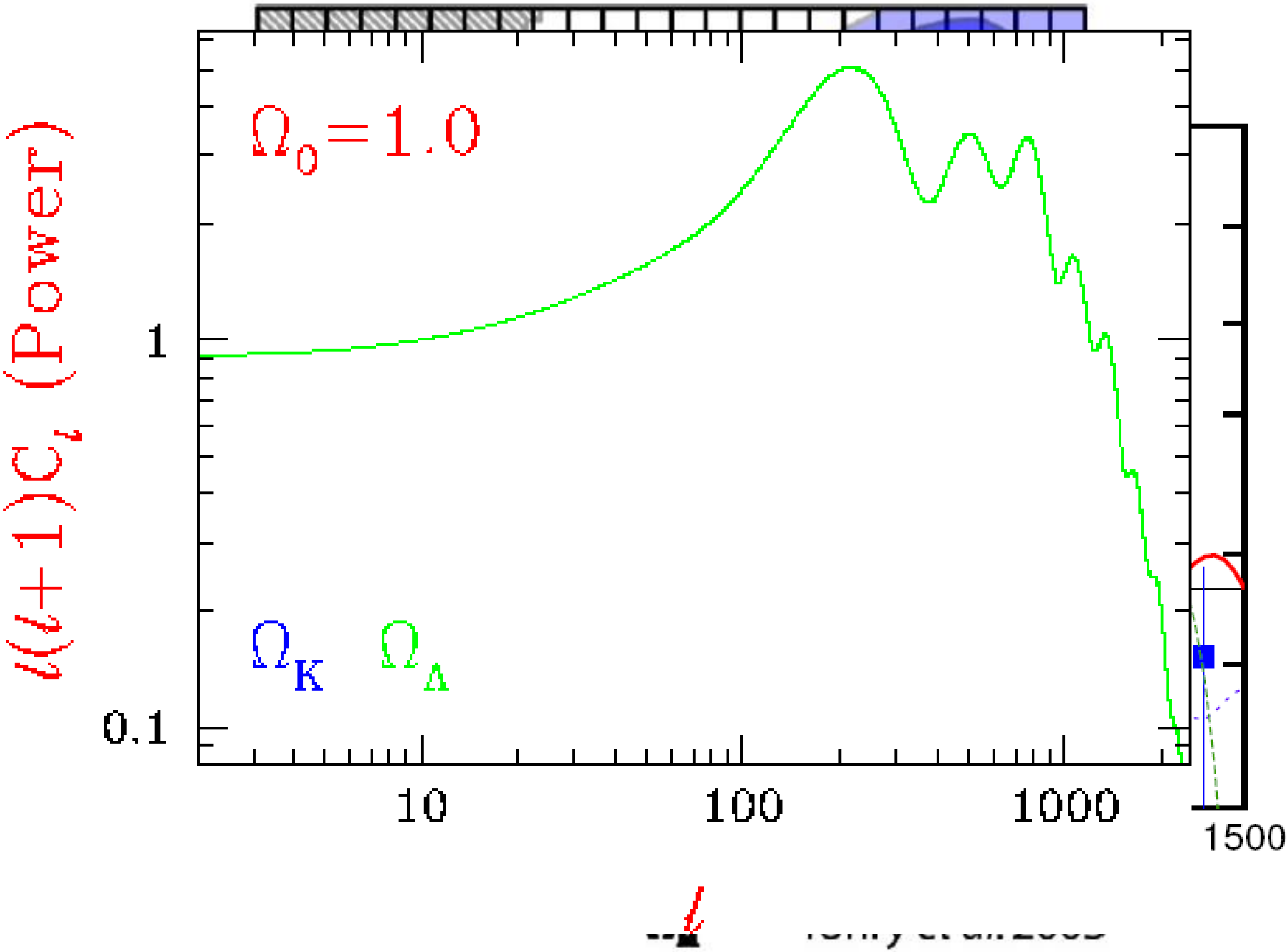
Complex example: cosmology



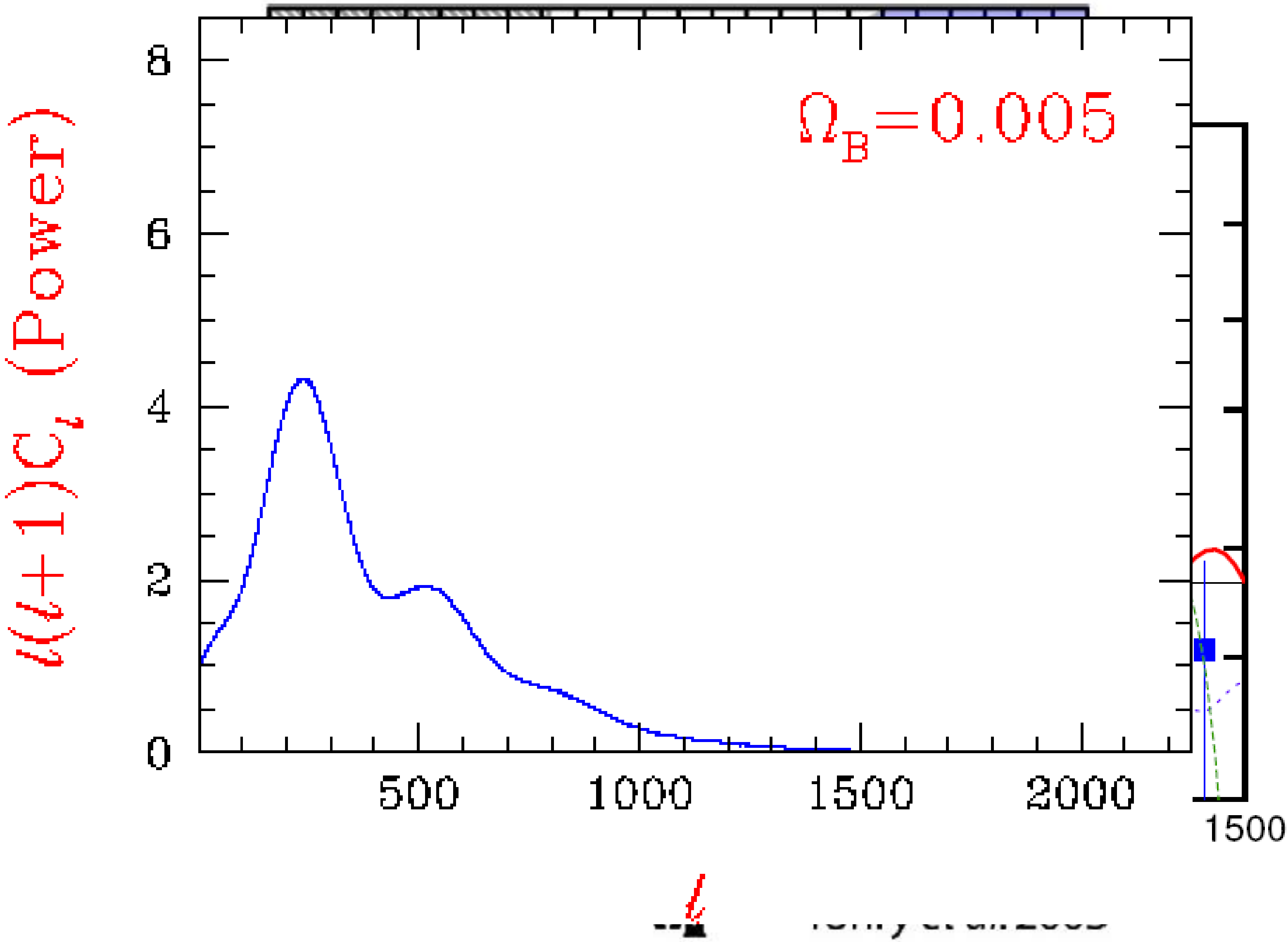
Complex example: cosmology



Complex example: cosmology



Complex example: cosmology



Complex example: cosmology