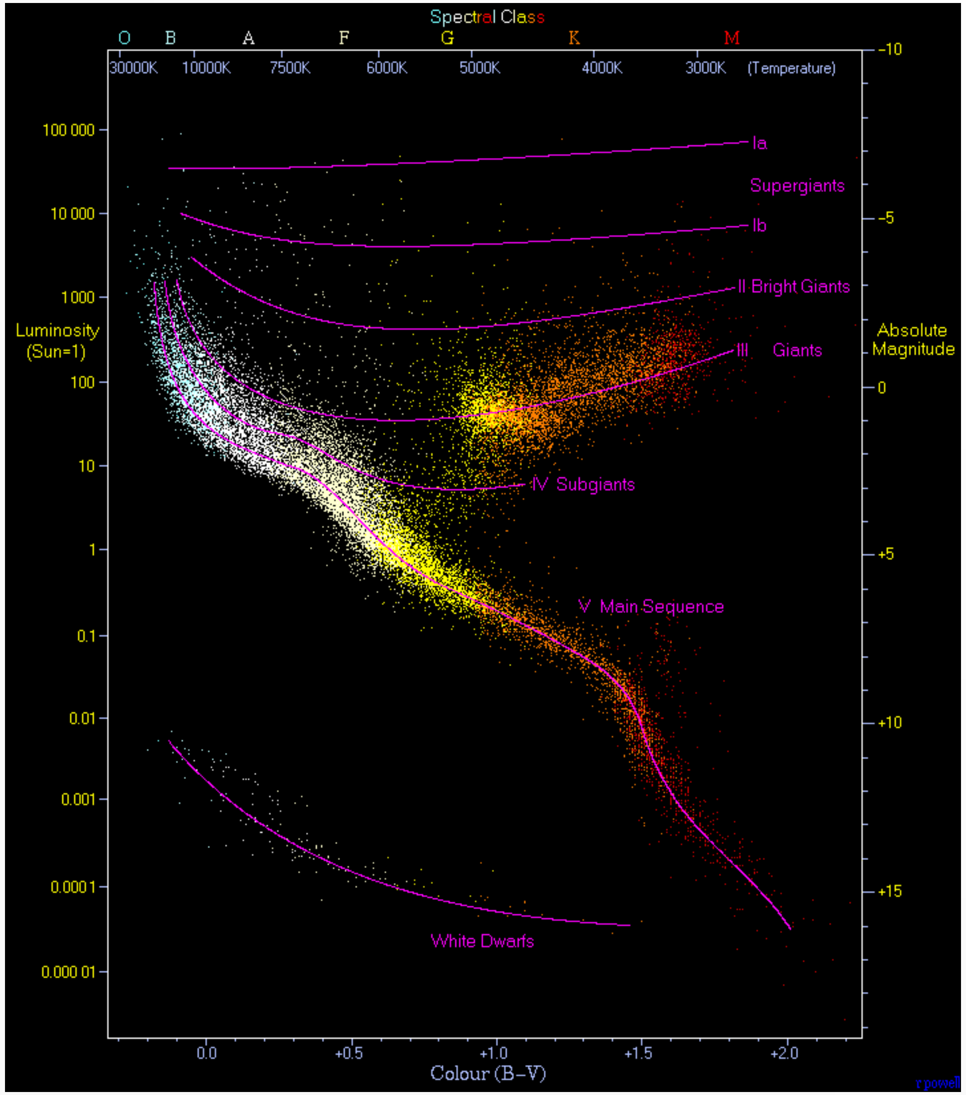
Development of an ultra-fast, likelihood-based, distance inference for the next generation of type Ia supernovae surveys

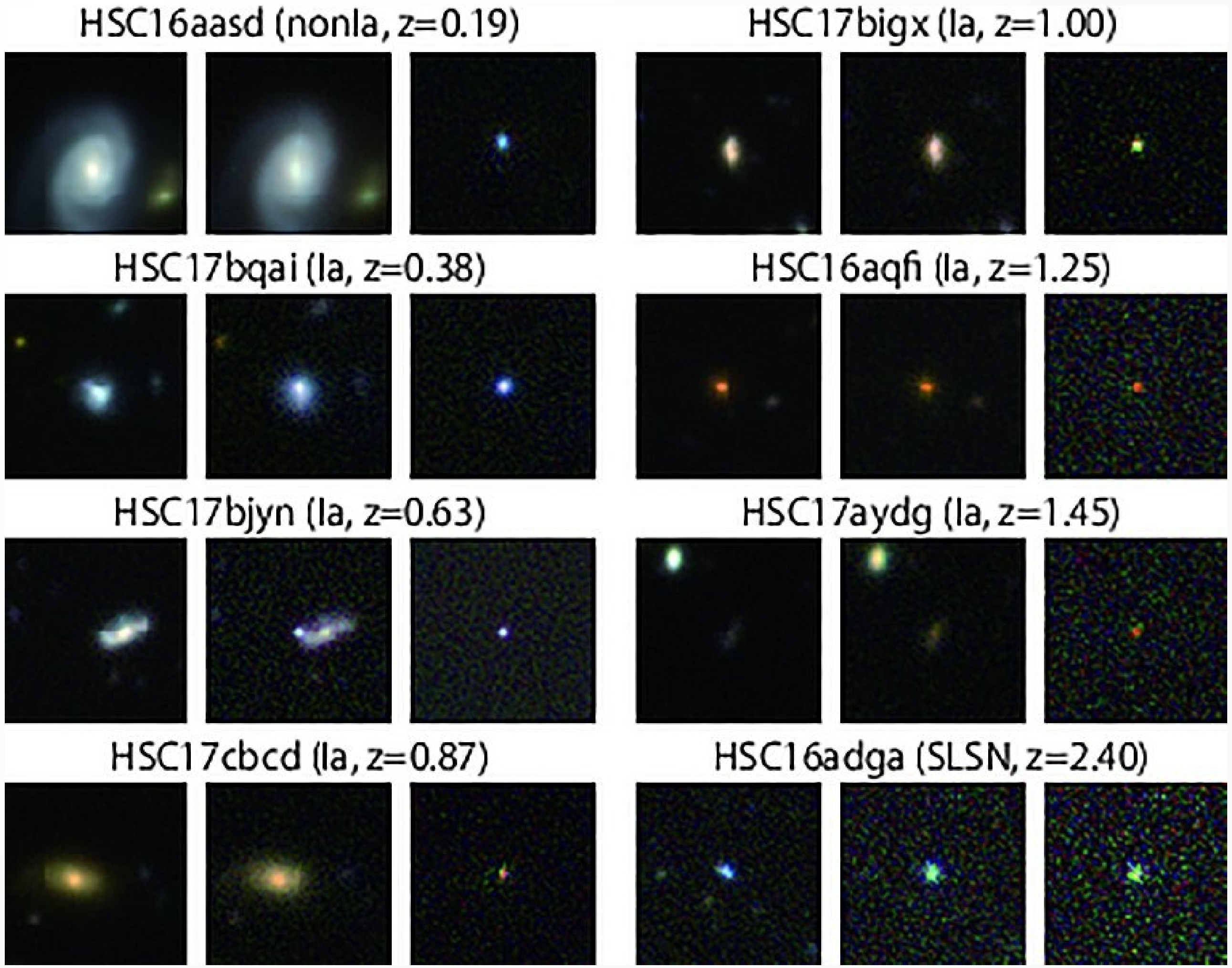
October, 24th 2023

Dylan KUHN, Marc BETOULE

Context

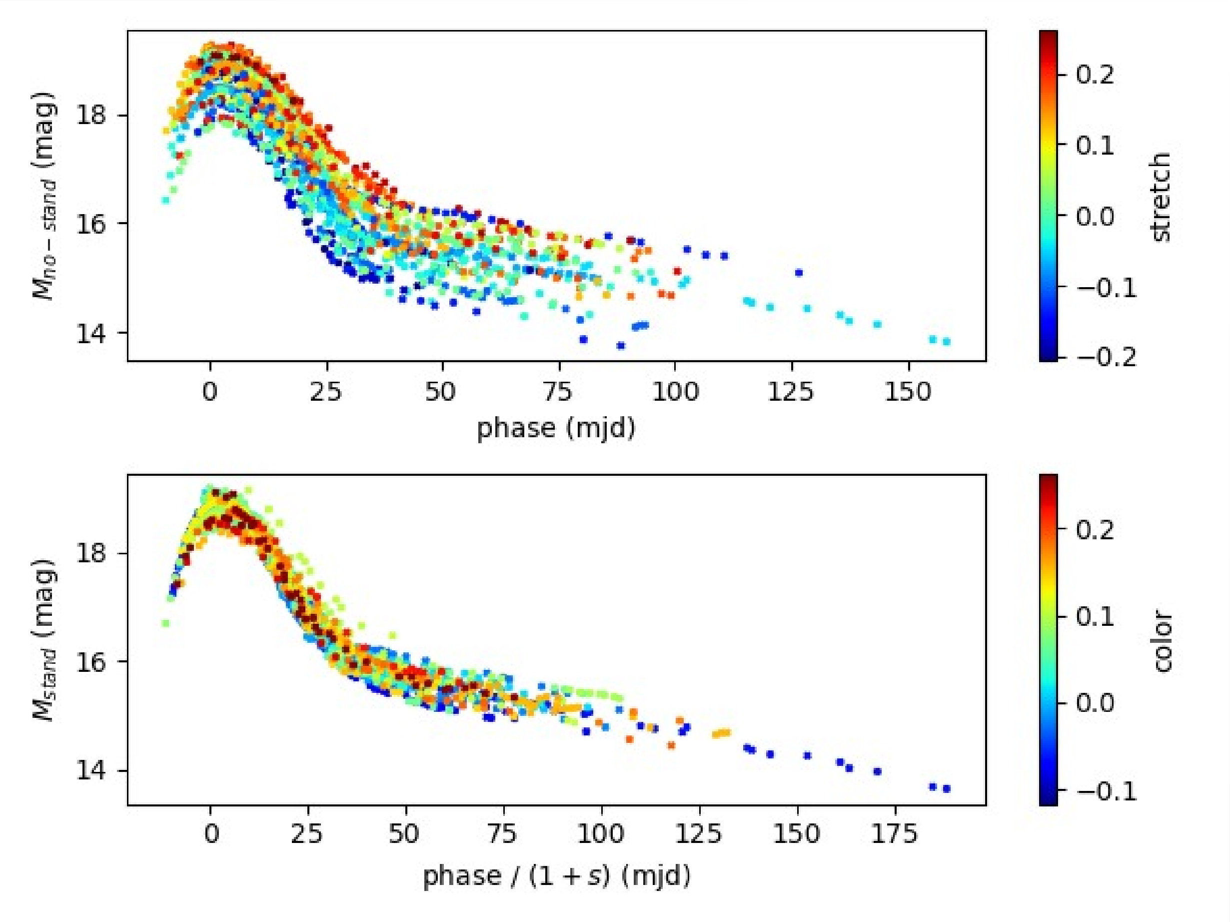
What is a type Ia supernova ?





Credits : N. Yasuda et al.

Why are SNeIa useful to estimate distances ?



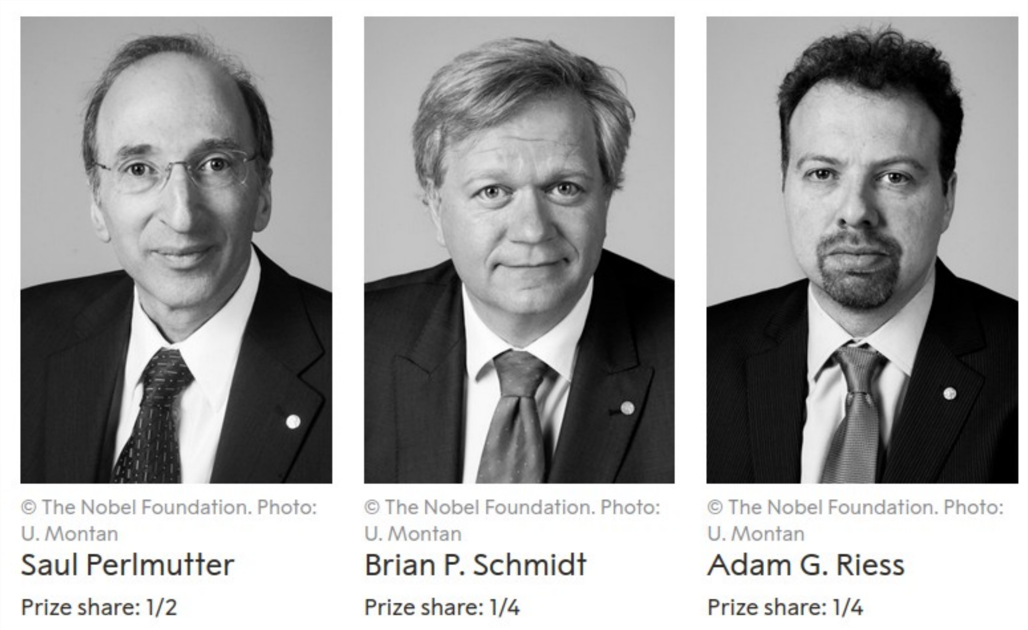
~ 45 % dispersion

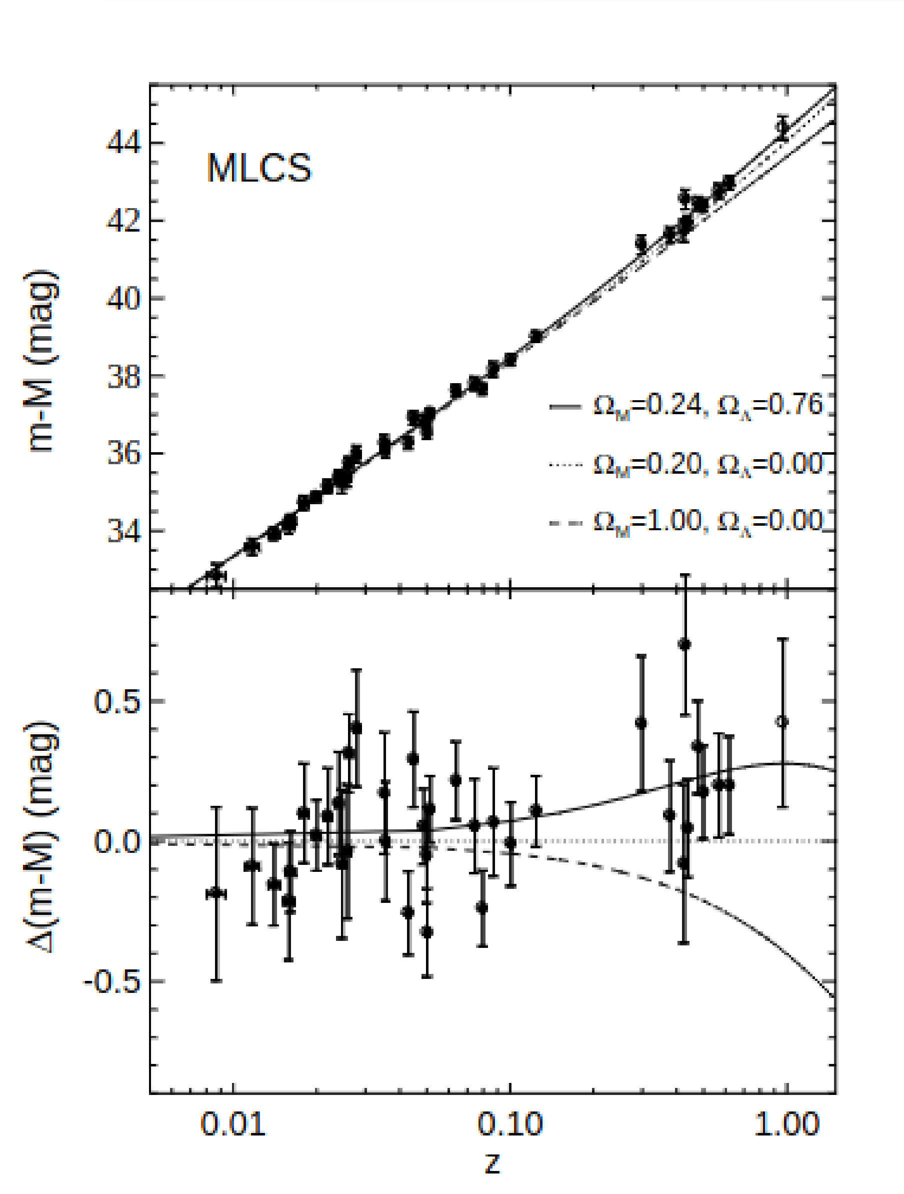
SNeIa → autosimilar events !
Brighter → bluer & slower

~ 15 % dispersion

= « intrinsic dispersion »

Discovery of the accelerated expansion of the Universe (1998)





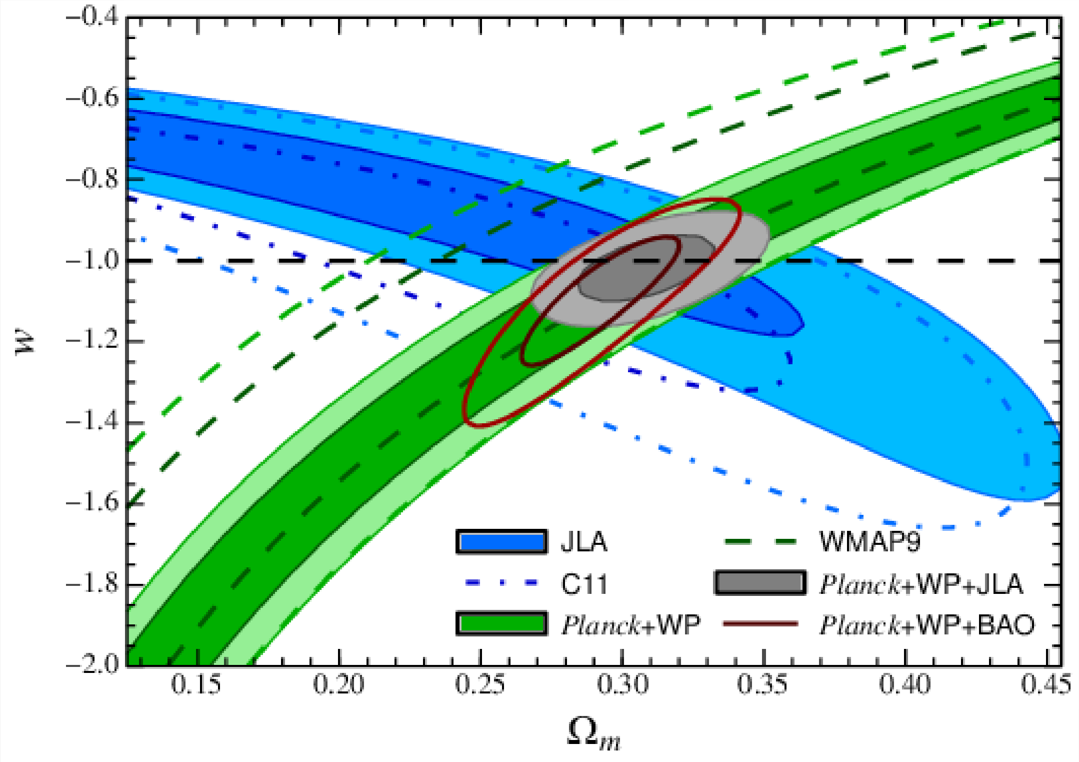
Nobel Prize 2011 :
The Universe is expanding and the expansion is accelerating

Credits : A. G. Riess et al., 1998

Measure of the Dark Energy equation of state w

In the flat-wCDM model, (luminosity) distance writes :

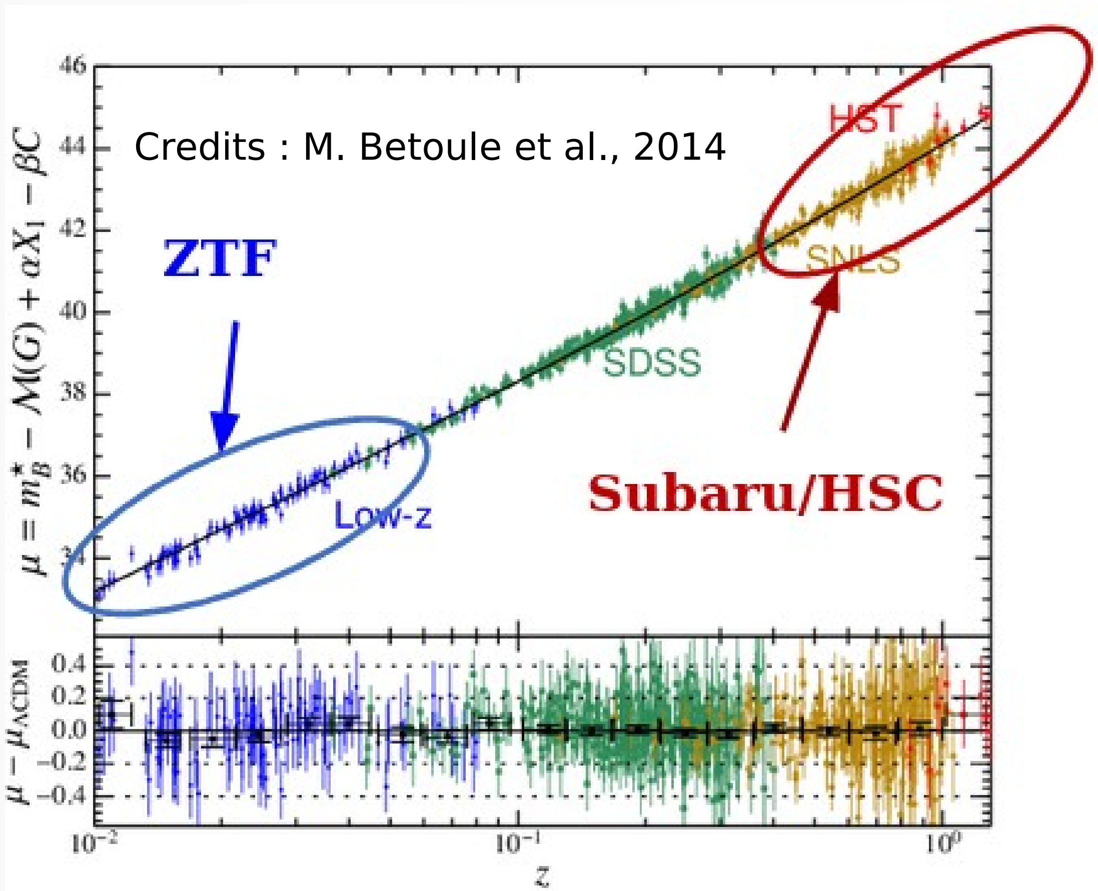
$$d_L = \frac{c}{H_0}(1+z)\int_0^z \frac{dz'}{\sqrt{\Omega_m(1+z')^3 + (1-\Omega_m)(1+z')^{3(1+w)}}}$$



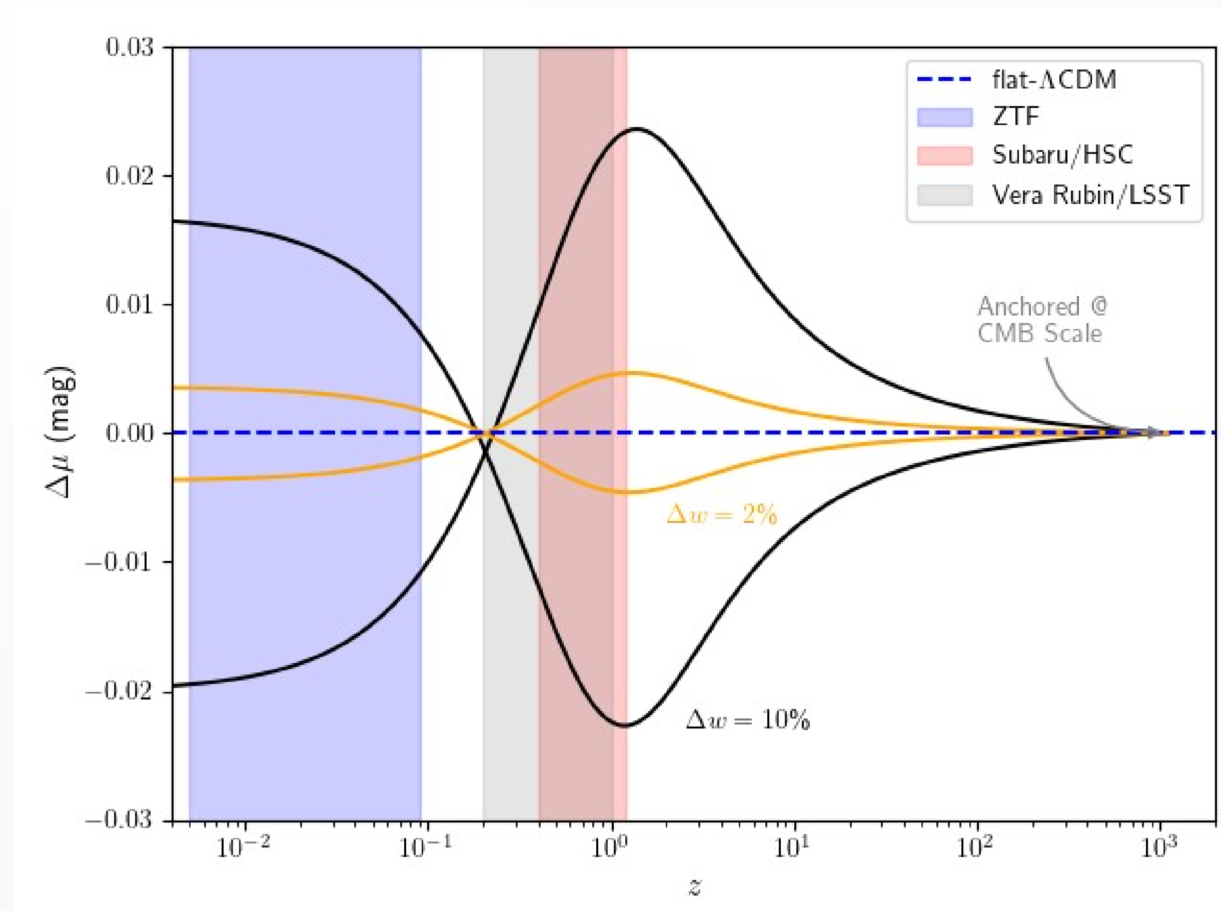
Important uncertainty on w
→ goal : decrease them to the % level
→ need to work on statistic and systematic uncertainties sources

Credits : M. Betoule et al., 2014

Why and where should we increase statistics in the Hubble Diagram ?



ZTF → + ~3000 SN at low redshift
Subaru → + ~400 SN at high redshift

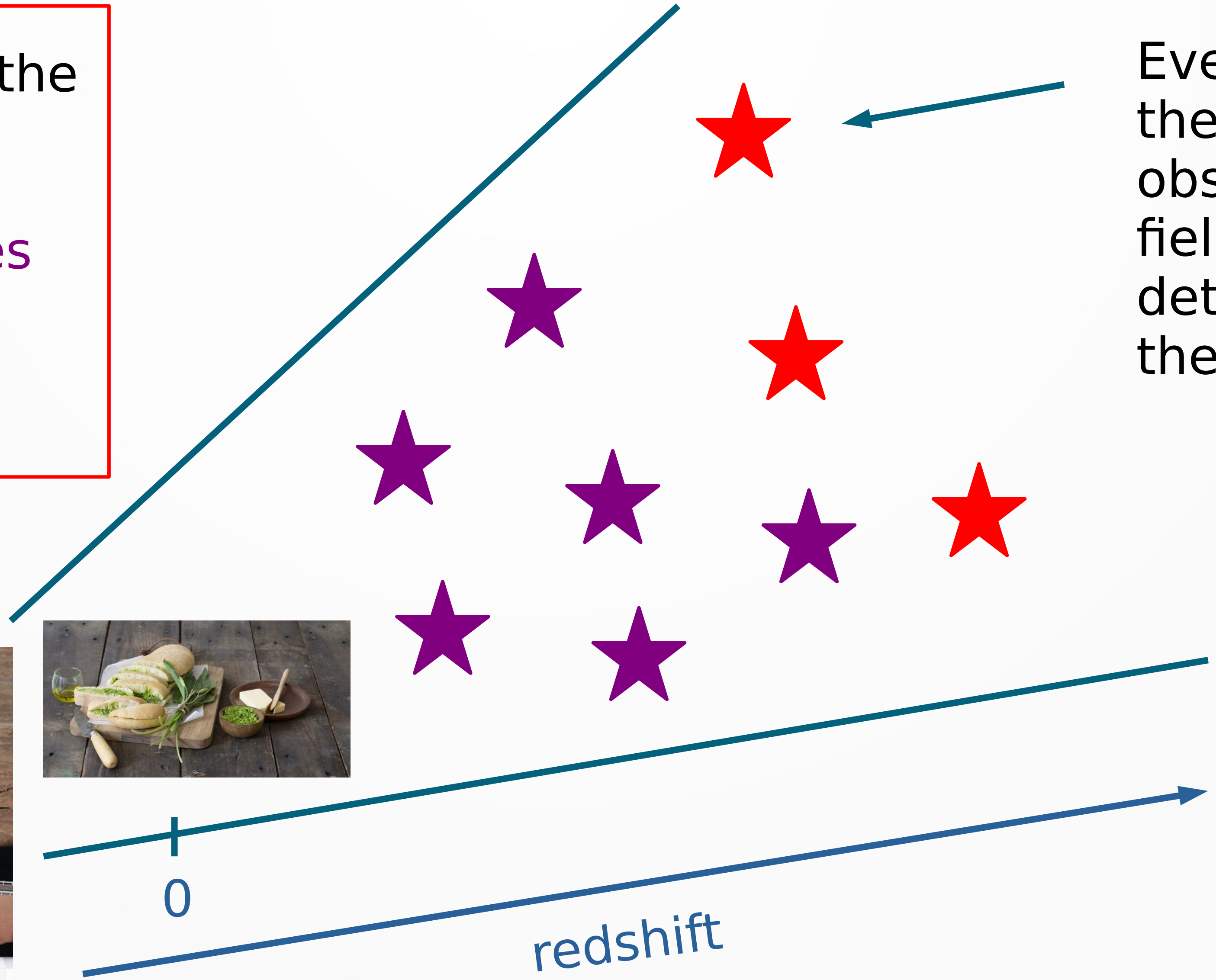


3 mmag variation on μ → 2 % on w
SNe Ia → Δμ = 0.15 mag
O(5000) SNe Ia → 3 mmag sur μ

Malmquist bias & truncated SNeIa surveys

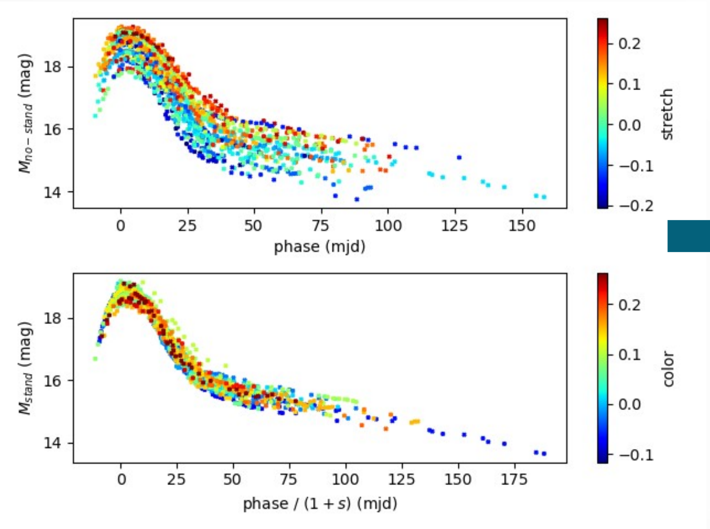
What do we call « Malmquist bias » ?

Effect : decreases the apparent mean magnitude of the population → biases the estimation of distances at high redshifts !



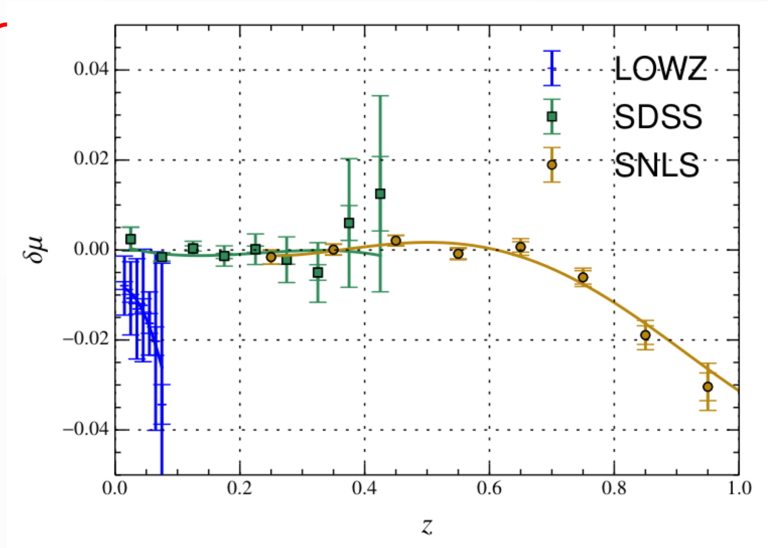
Even though the SN is in the observation field, it is not detected by the telescope

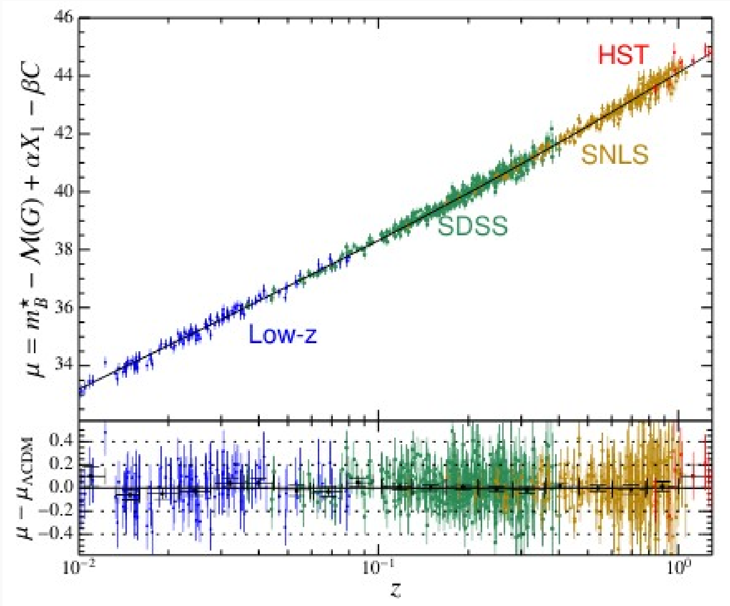
Estimation of the Malmquist bias : state of the art



SNeIa light-curves

$$\begin{pmatrix} m_B \\ x_1 \\ c \end{pmatrix} + \mathrm{Cov}(m_B, x_1, c)$$





Determination of Malmquist bias & correction of distances

My current work : how to do the last part in a single step ?

Credits : Marc Betoule et al., 2014

Illustration of the bias on a toy model

- Toy model :

$$m_i^* = M^* + \mu_i + \epsilon_i \text{ with } \epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

Information compressed in binned distance moduli

- The truncation is modelled as follows :

$$m_i = m_i^* \text{ if } m_i^* \leqslant m_{lim}$$

$$m_i \text{ is unobserved otherwise}$$

Useful relations :

$$\mu = \Xi \xi$$

$$\Phi(z) = \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{z}{\sqrt{2}}\right)\right)$$

- The associated negative log-likelihood function is :

$$\Gamma = \sum_i 2\ln(\sigma) + \frac{1}{\sigma^2} r^\dagger r + \ln\left(\Phi\left(\frac{m_{lim} - M^* - \mu_i}{\sigma}\right)\right)$$

Related to the instrument

Intrinsic dispersion

Issues when dealing with real supernovae

- Type Ia supernovae are standardized :

$$m_i = M^* + \mu_i \longrightarrow m_i = M^* + \mu_i + \underbrace{\alpha_1 Y_1 + ... + \alpha_n Y_n}_{\alpha x_1 + \beta c}$$

- Additional noise :

$$\eta \sim \mathcal{N}(0, \underbrace{\mathrm{Cov}(m, Y_1, ..., Y_n)}_{= C})$$

Model with latent parameters

- Selection function depends also on the weather :

→ introduction of fluctuations on the limit magnitude σ_d

(See Madeleine's talk for more information about standardization)

EDRIS

French for ‘Distance Estimator for Incomplete Supernovae Surveys’

Model used for the EDRIS analysis

$$\begin{pmatrix} {m^*}_i \\ {Y^*_1}_i \\ {Y^*_2}_i \\ \vdots \\ {Y^*_n}_i \end{pmatrix} = \begin{pmatrix} \mu_i(z,\theta) \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \cdots & \alpha_n \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} {X^*_1}_i \\ {X^*_2}_i \\ {X^*_3}_i \\ \vdots \\ {X^*_n}_i \end{pmatrix} + \begin{pmatrix} \epsilon_i \\ 0 \\ \vdots \\ 0 \end{pmatrix} \text{ with } \epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

LCs parameters

Distances

Standardization coefficients

Latent parameters

Noise depending on the intrinsic dispersion

$$\left.\begin{array}{l} Y_i = Y_i^* + \eta_i \text{ with } \eta \sim \mathcal{N}(0, C) \text{ if } m_i \leqslant m_{lim} + \kappa_i \text{ with } \kappa_i \sim \mathcal{N}(0, \sigma_d^2) \\ \qquad\qquad Y_i \text{ is unobserved otherwise} \end{array}\right\} \text{Truncation}$$

Negative log-likelihood function

$$\Gamma = -\ln(|W|) + r^\dagger W r + \sum_i \left[2\ln\left(\Phi\left(\frac{m_{lim} - \mu_i - \alpha_1 X^*_{1\,i} - \cdots - \alpha_n X^*_{ni}}{\sqrt{\sigma_d^2 + \sigma^2}}\right)\right) - 2\ln\left(\Phi\left(\frac{m_{lim} - m_i}{\sqrt{\sigma_d^2 + f(C_i)}}\right)\right)\right]$$

allows to estimate the intrinsic dispersion

classic chi2 term

takes into account the truncation effects

Parameters :

ξ : distance moduli at the nodes of the spline

$\begin{pmatrix}\alpha_1 & \cdots & \alpha_n\end{pmatrix}$: standardization coefficients

X^* : latent parameters

σ : intrinsic dispersion

m_{lim} : limit magnitude

σ_d : fluctuations of the limit magnitude

Useful relations :

$$\mu = \Xi\xi$$

$$\Phi(z) = \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{z}{\sqrt{2}}\right)\right)$$

Fast computation of the likelihood

- Inversion of the covariance matrix by the Schur complement technique :

$$W = \begin{pmatrix} C^{mm} + \sigma^2 I_N & C_1 \\ C_1^\dagger & C_2 \end{pmatrix}^{-1} \Rightarrow W = \begin{pmatrix} S^{-1} & -S^{-1} C_1 C_2^{-1} \\ -C_2^{-1} C_1^\dagger S^{-1} & C_2^{-1} + C_2^{-1} C_1^\dagger S^{-1} C_1 C_2^{-1} \end{pmatrix}$$

$$\text{with } S = C^{mm} + \sigma^2 I_N - C_1 C_2^{-1} C_1^\dagger$$

- By writing :

$$C^{mm} - C_1 C_2^{-1} C_1^\dagger = Q \Lambda Q^\dagger$$

we obtain :

$$S^{-1} = Q(\Lambda + \sigma^2 I_N)^{-1} Q^\dagger$$

Fast computation of the likelihood

- We rewrite r to match the structure of W : $r = (r_1, r_2)$
- The chi2 term writes as follows :

$$r^\dagger W r = r_1^\dagger S^{-1} r_1 - 2 r_1 S^{-1} C_1 C_2^{-1} r_2 + r_2^\dagger C_2^{-1} + r_2^\dagger C_2^{-1} C_1^\dagger S^{-1} C_1 C_2^{-1} r_2$$

- The determinant of W writes as follows :

$$-\ln(|W|) = \ln(|C_2|) + \ln(|S|) = \ln(|C_2|) + \sum_i \ln(\Lambda_i + \sigma^2)$$

Only matrix-to-vector products

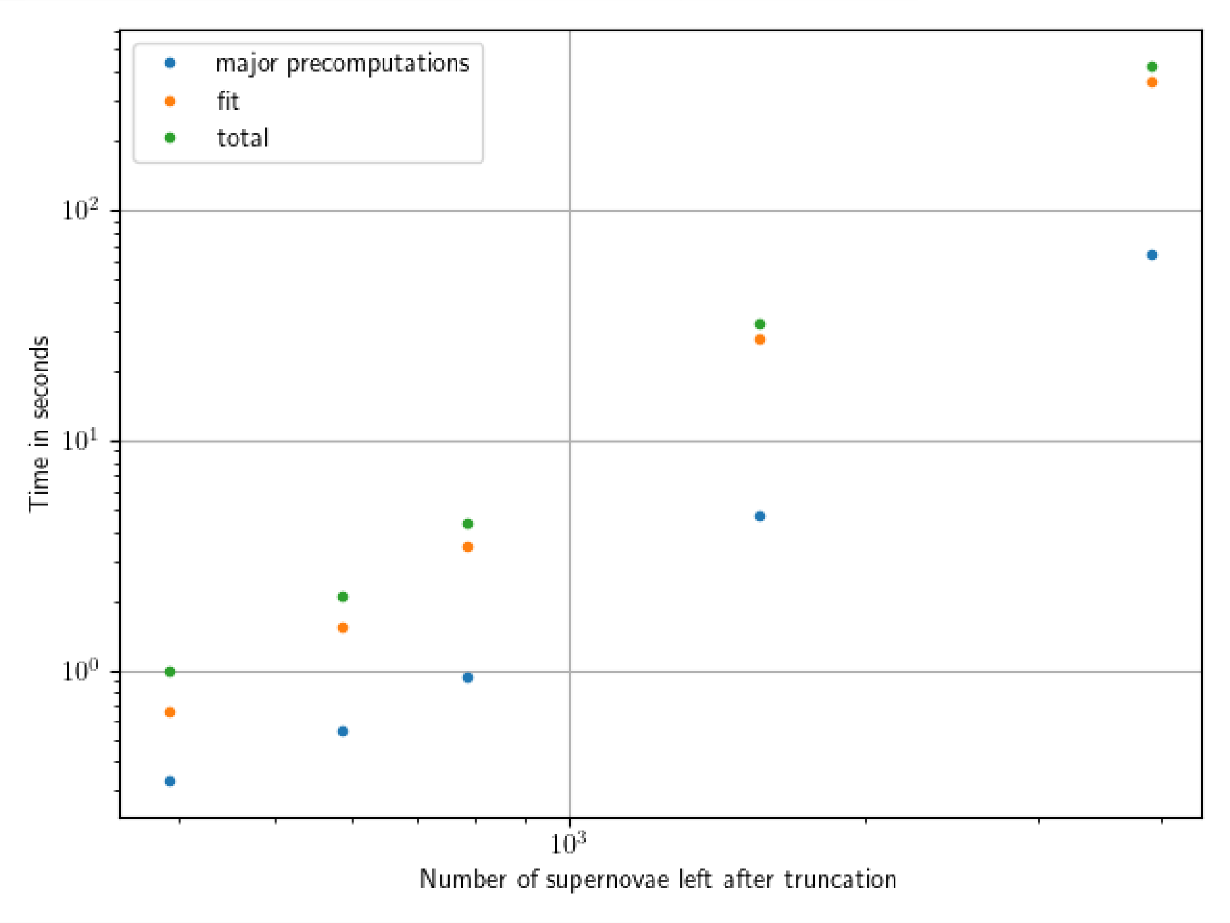
- At the end of the day, for one iteration we only need to compute :

$$\left(r \quad (\Lambda + \sigma^2 I_N)^{-1} \quad \textstyle\sum_i \ln(\Lambda_i + \sigma^2)\right)$$

Computation in $O(N^2)$

Time scaling of the log-likelihood minimization

Scaling for a hard-coded Gauss-Newton algorithm



Fit steps :

→ blue dots : major precomputations (see maths)

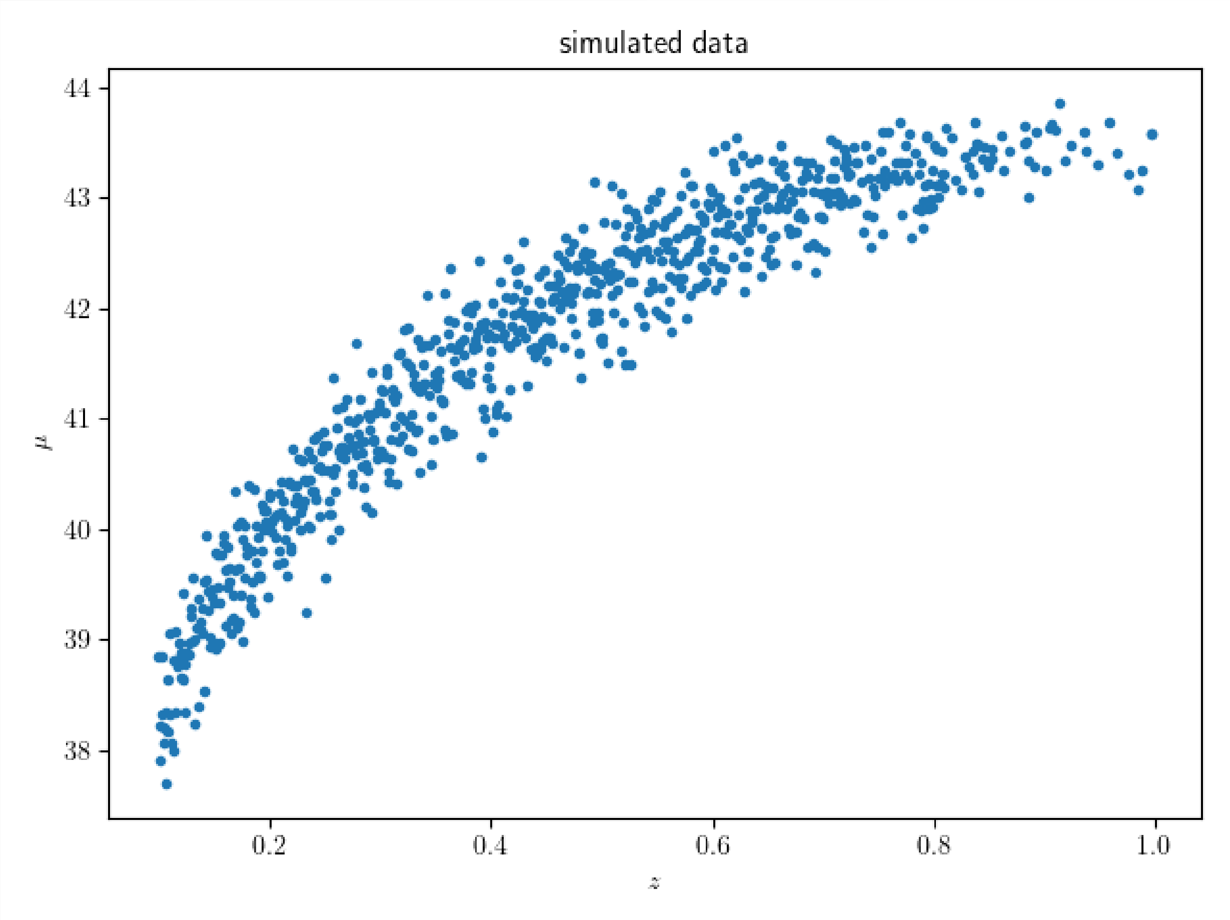
→ orange dots : minimization, dominated by the construction of the hessian matrix at each step (likelihood and gradient in $O(N^2)$)

→ Plot regularly updated with each new optimization

(When using hessian-free method + JAX/Optax Python librairies, ~ 15 s for O(5000) SNeIa… so very fast indeed !)

Characterization of the estimator with simulations

Example of simulated data



Parameters of the simulation :

$$N_{SN} = 1000$$
$$N_{bins} = 30$$
$$m_{lim} = 24.5$$
$$\sigma_d = 0.2$$
$$\sigma = 0.1$$
$$x_1 \sim \mathcal{N}(0,1)$$
$$c \sim \mathcal{N}(0,0.1)$$
$$\text{FLCDM} : H_0 = 70 \ \& \ \Omega_m = 0.3$$

Monte-Carlo simulation

- We compare the bias of two estimators with a Monte-Carlo simulation :

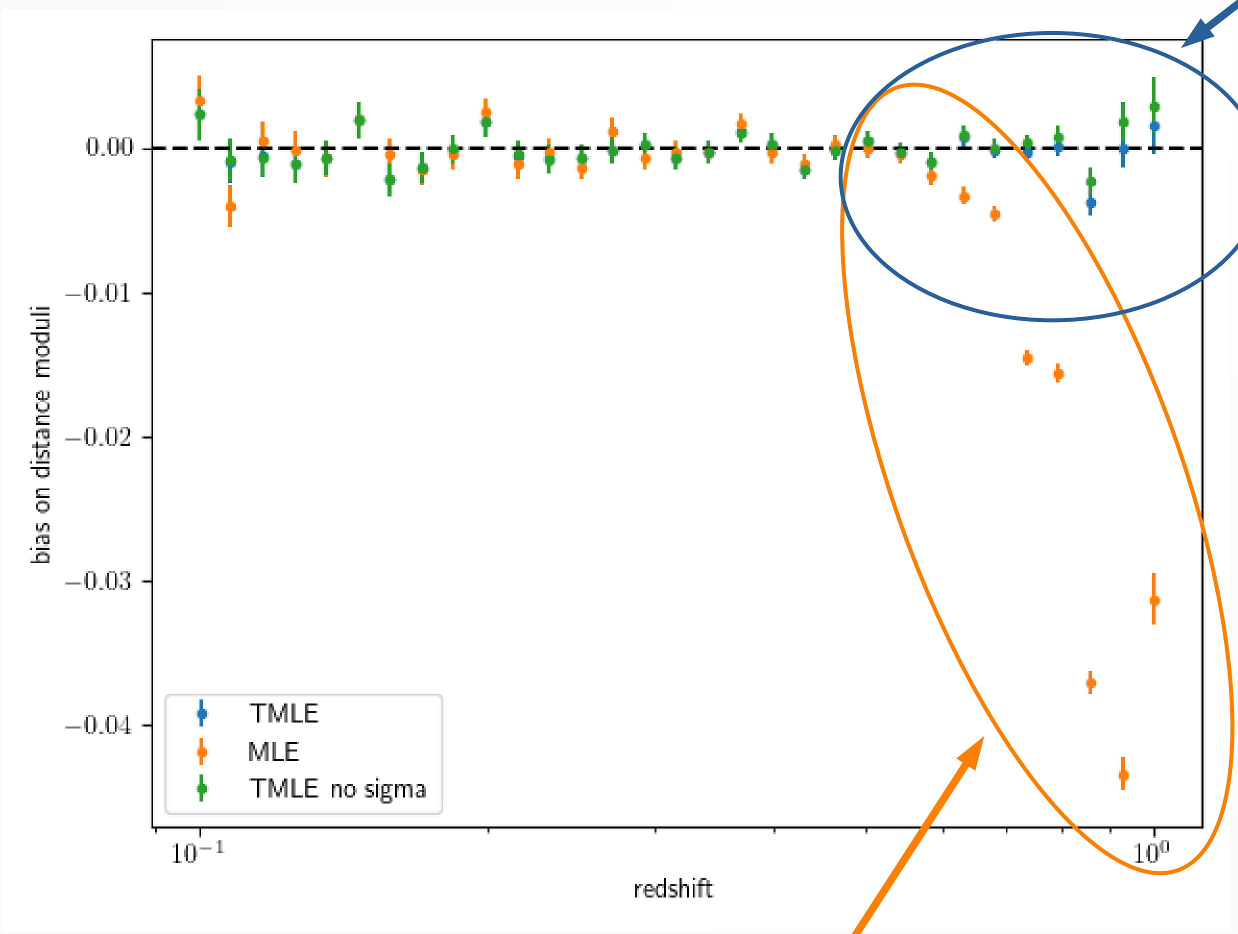
  → Estimator 1 (MLE) : classic maximum likelihood estimator associated with the following negative log-likelihood function :

$$\Gamma = -\ln(|W|) + r^{\dagger} W r$$

  → Estimator 2 (TMLE) : new maximum likelihood estimator with takes into account the truncation

Results when SN are very well measured

corrected Malmquist bias



Fitted parameters for the MLE :

$$\begin{pmatrix} \xi & \alpha_1 & \cdots & \alpha_n & \sigma & X^* \end{pmatrix}$$

Fitted parameters for the TMLE :

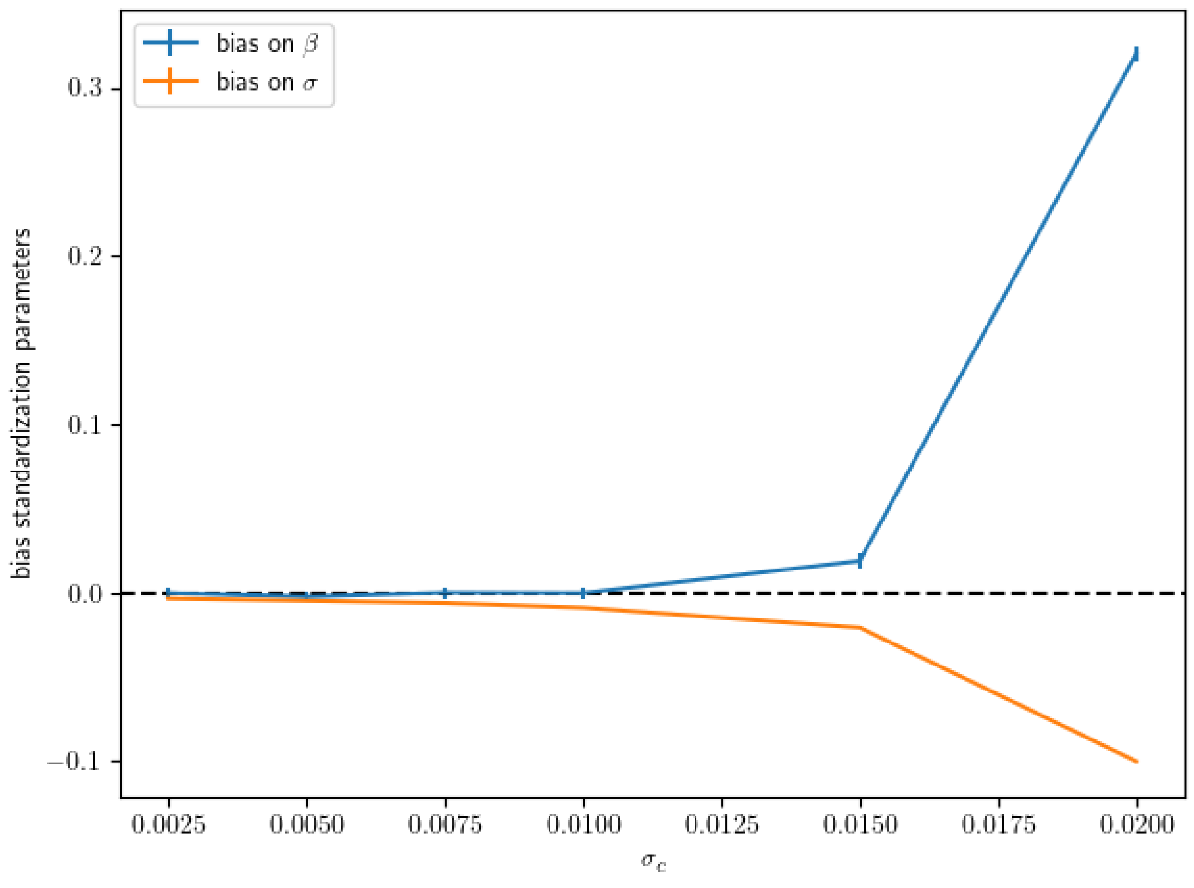
$$\begin{pmatrix} \xi & \alpha_1 & \cdots & \alpha_n & \sigma & X^* \end{pmatrix}$$

Fixed parameters for the TMLE :

$$\begin{pmatrix} m_{lim} & \sigma_d \end{pmatrix}$$

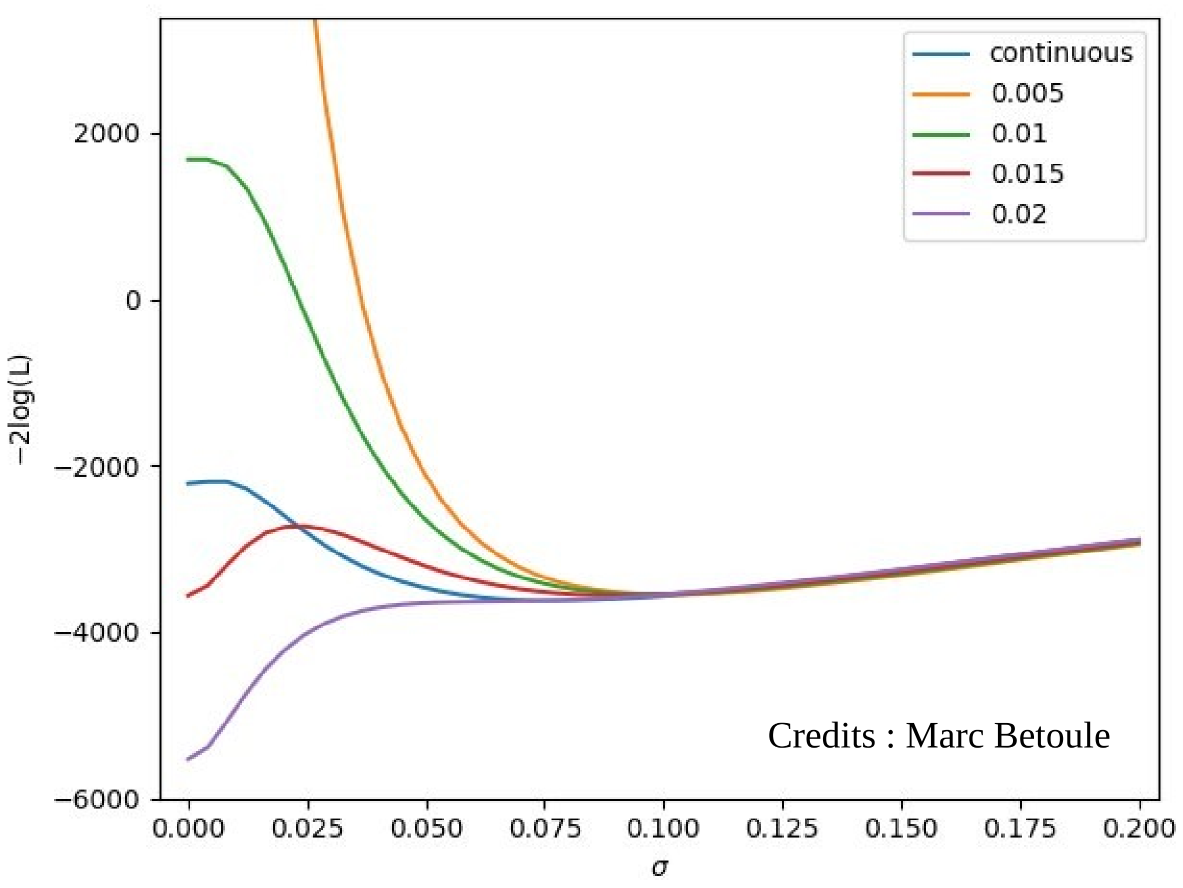
apparent Malmquist bias

Bias of the TMLE when SN are less well measured



The TMLE is biased when

$$\beta\sigma_c \sim \sigma$$



Similar effect when we increase the error on the stretch

Restricted Maximum Likelihood Estimator

- When using the MLE, denoting n the number of data and k the number of degrees of freedom

$$\mathbb{E}(\hat{\sigma}) = \frac{n-k}{n}\sigma$$

→ thus the variance estimator is biased

- Restricted Maximum Likelihood Estimator (ReMLE) allows to unbias the variance estimator by reducing the number of degrees of freedom
- Implemented on a simple toy model :

    → only color for standardization & no truncation

- Seems to work pretty well for now

Conclusions

Take-home message

- Fast distance estimator for truncated SNeIa surveys (EDRIS) : unbiases distances at high redshifts (main goal) but still has some weaknesses
- Open questions we are working on :

1°) Finding a solution to correct bias on standardization parameters and intrinsic dispersion :

Merge TMLE estimator with ReMLE estimator

2°) Estimating the parameters of the selection function :

Limit magnitude & fluctuations

3°) (Not included here) Implementing a simulation pipeline to study the behaviour of the estimator when we deviate from initial hypothesis :

Selection applied on observed magnitudes instead of reconstructed magnitudes, selection depending on the wavelenght/filter, training of light-curve fitter with a truncated dataset, spectroscopic selection, ...

Thanks for listening

Do not hesitate to ask questions