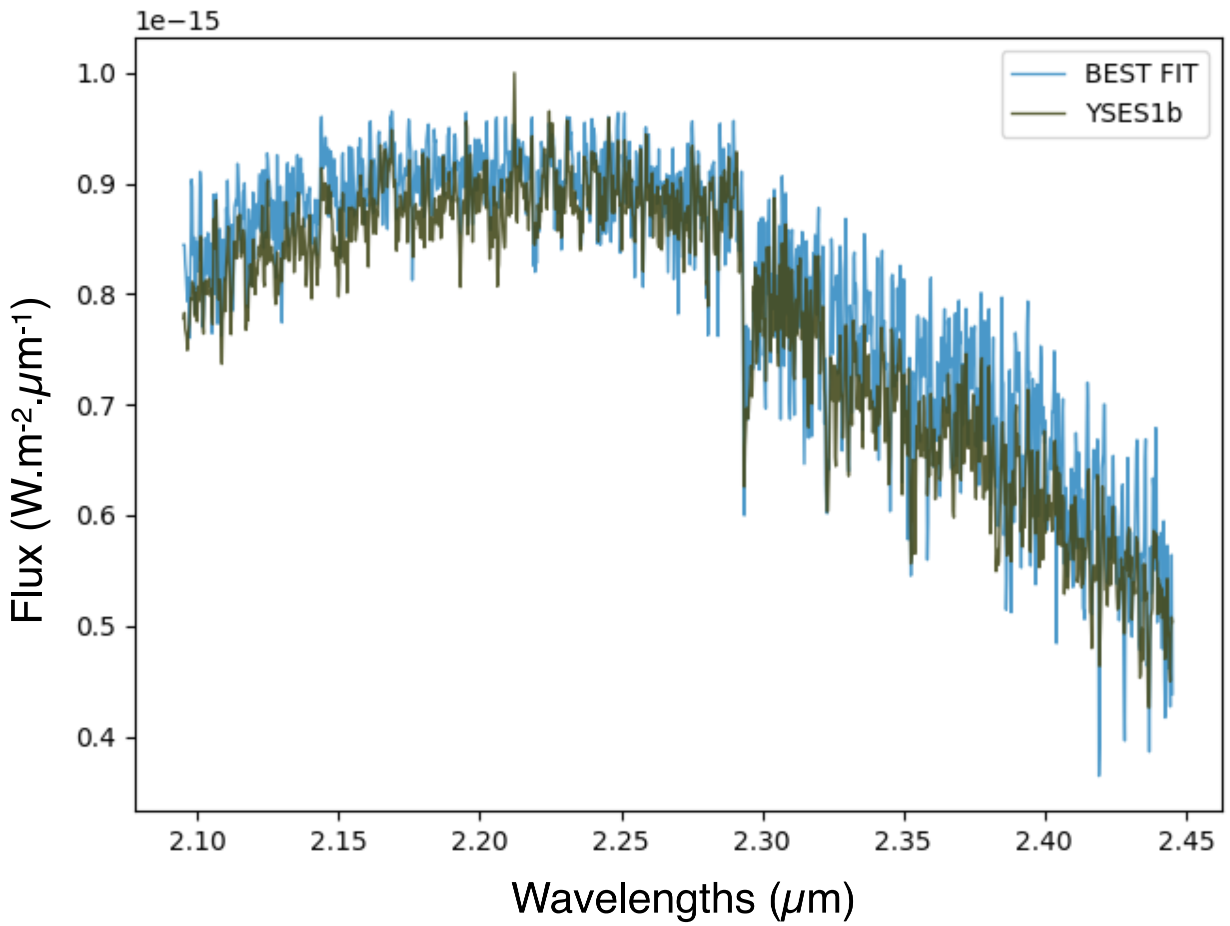
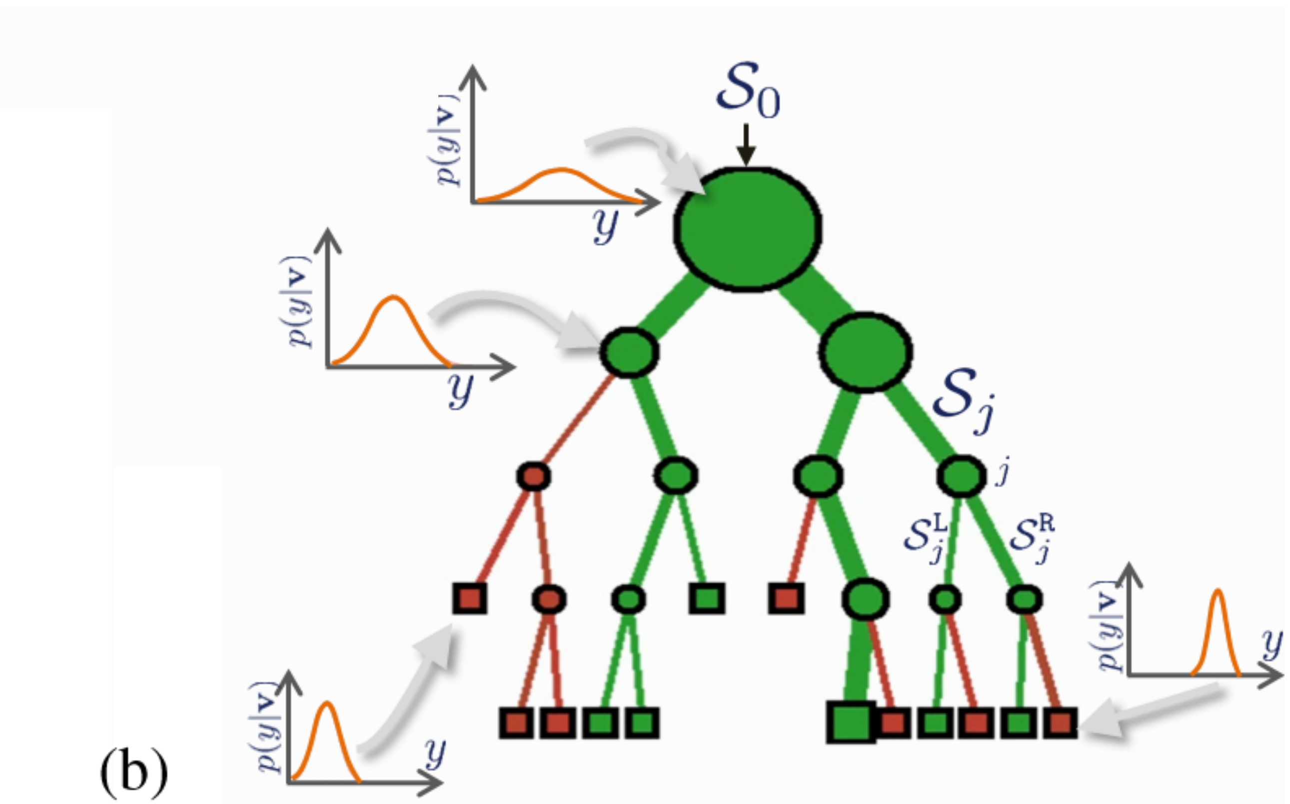
Inversion of Exoplanet Spectra @ COBREX from Bayesian Inference to Machine Learning

Mickaël Bonnefoy (IPAG-CNRS)

Dev. Team: S. Petrus (U. Valparaiso), P. Palma-Bifani (OCA-Lagrange), P. KOMBA-BETAMBO (LESIA-CNRS), M. Bonnefoy (IPAG-CNRS), G. Chauvin (OCA-Lagrange), A.-M. Lagrange (LESIA-CNRS)





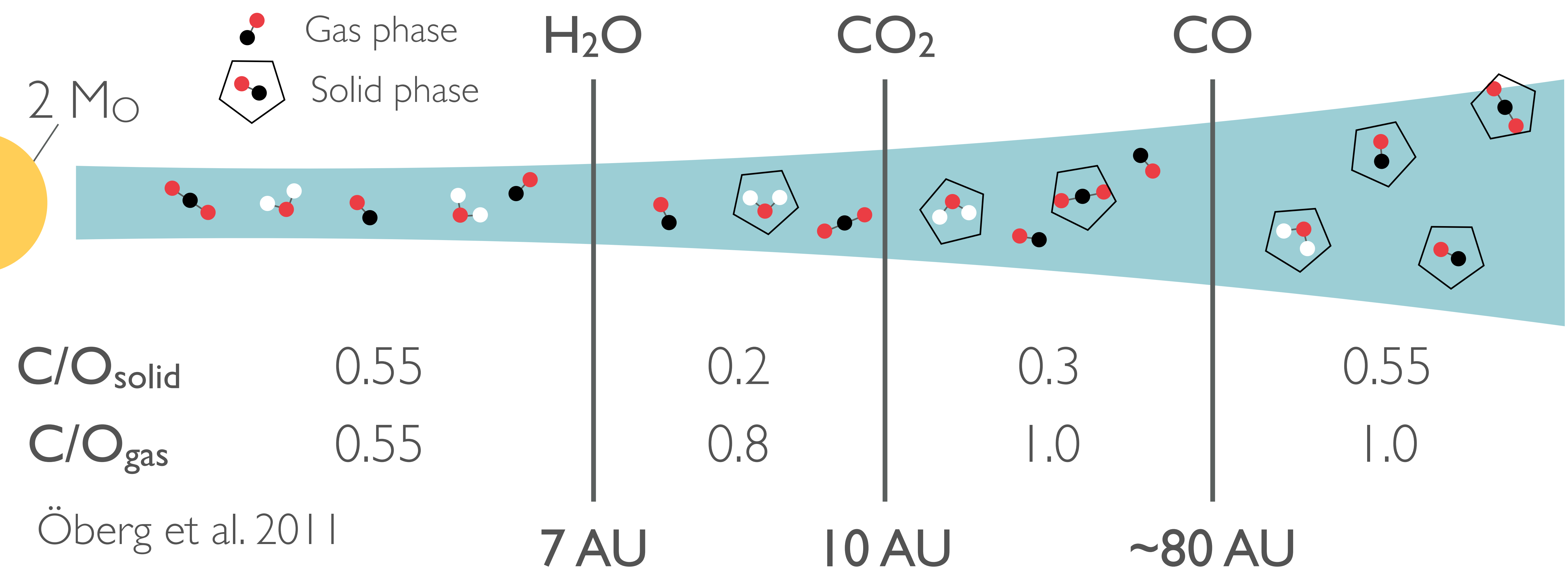
COBREX meeting · Saint Aygulf · Oct. 4, 2022

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (COBREX; grant agreement n° 885593)

Motivation

Formation tracers

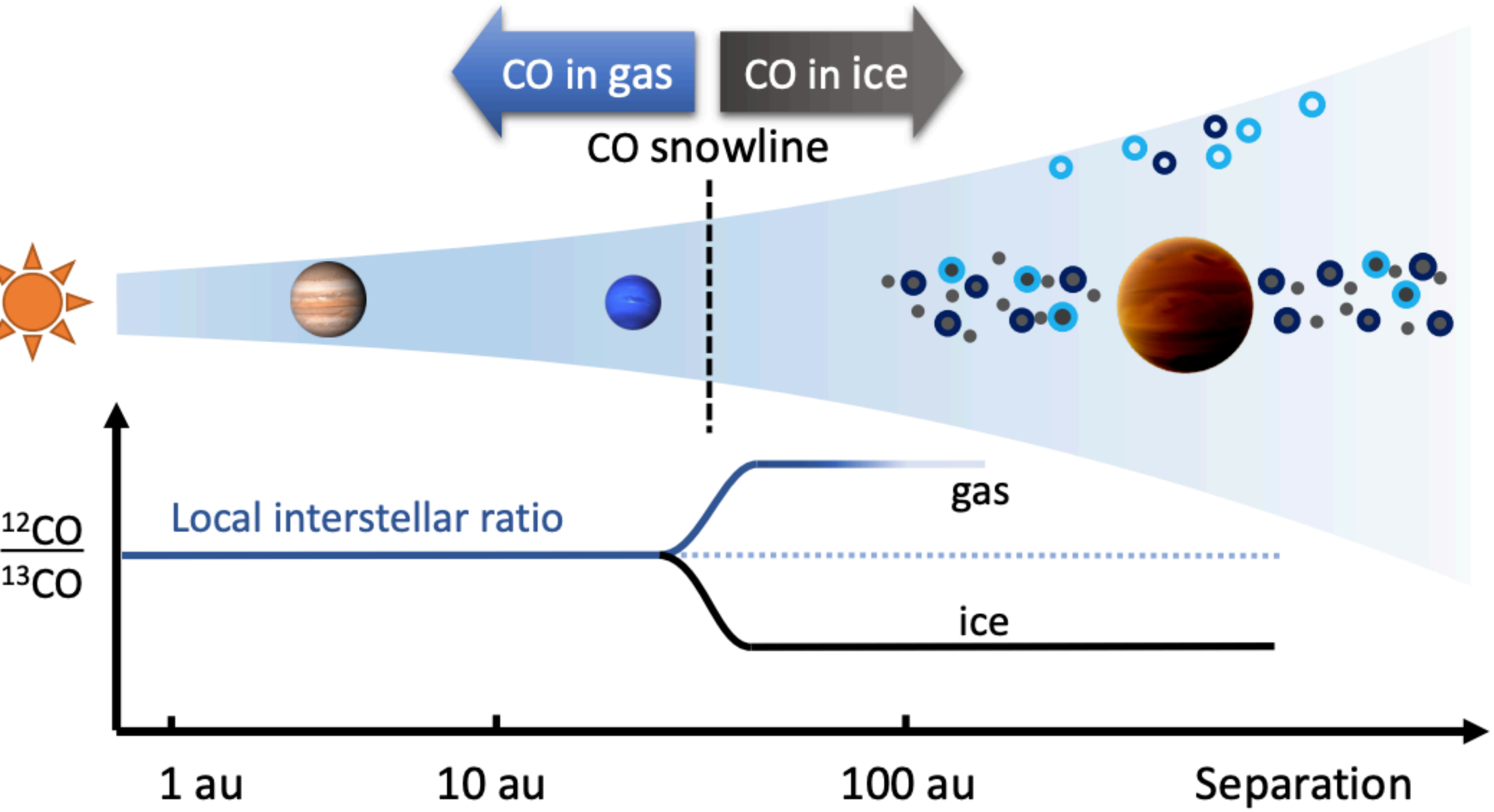
The C/O ratio: formation location & accretion of solids [?]

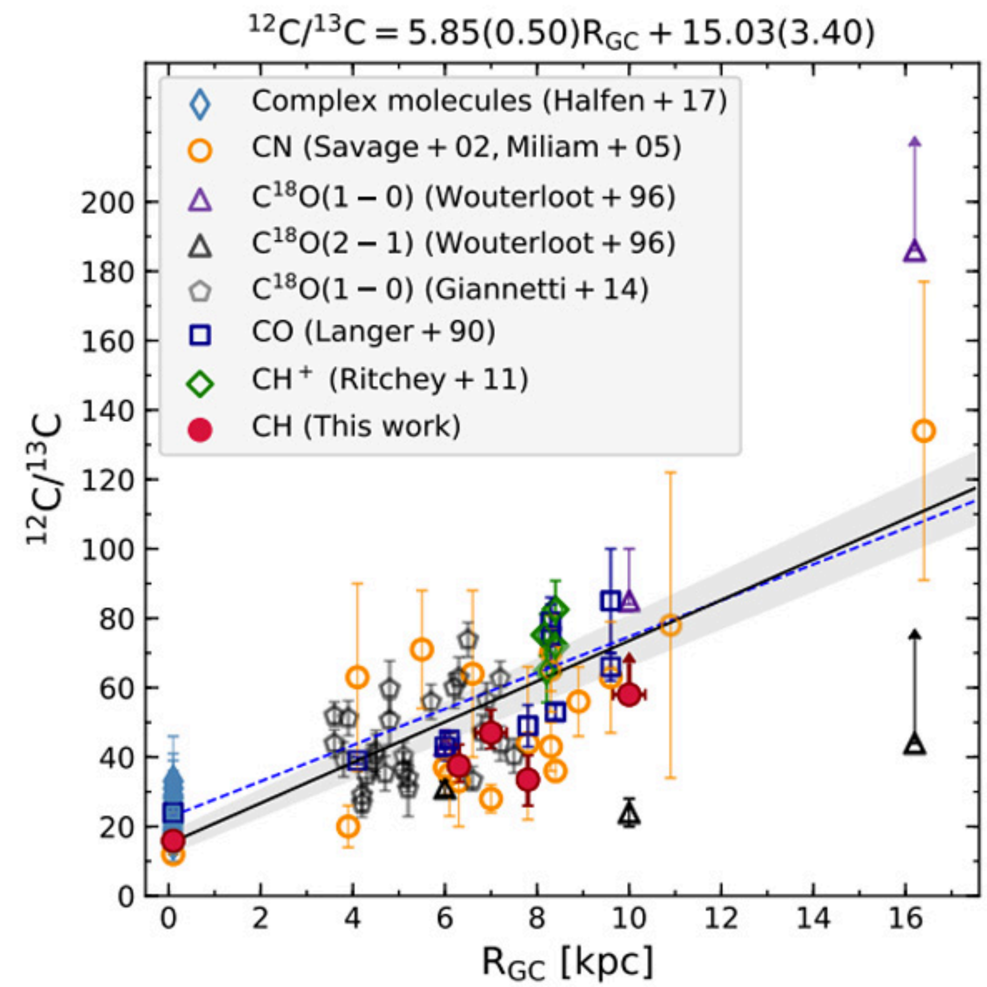


Motivation

Formation tracers

The $^{12}CO/^{13}CO$ ratio: thermal processing of solids @ formation [?]



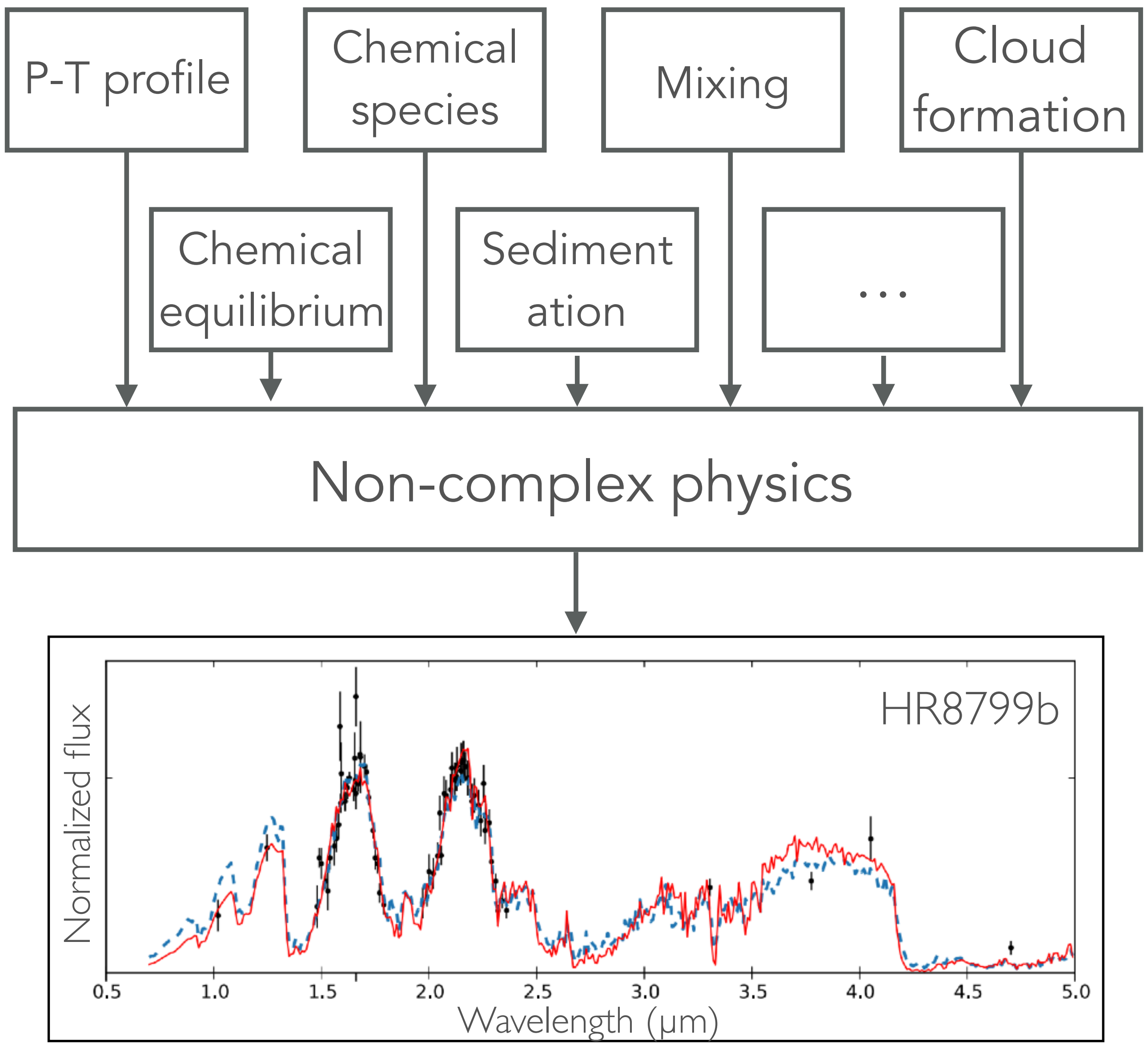


Cowing et al. 2020

Motivation

Formation tracers

RETRIEVAL



HELIOS-R (Lavie et al. 2017)

The data drive the fit

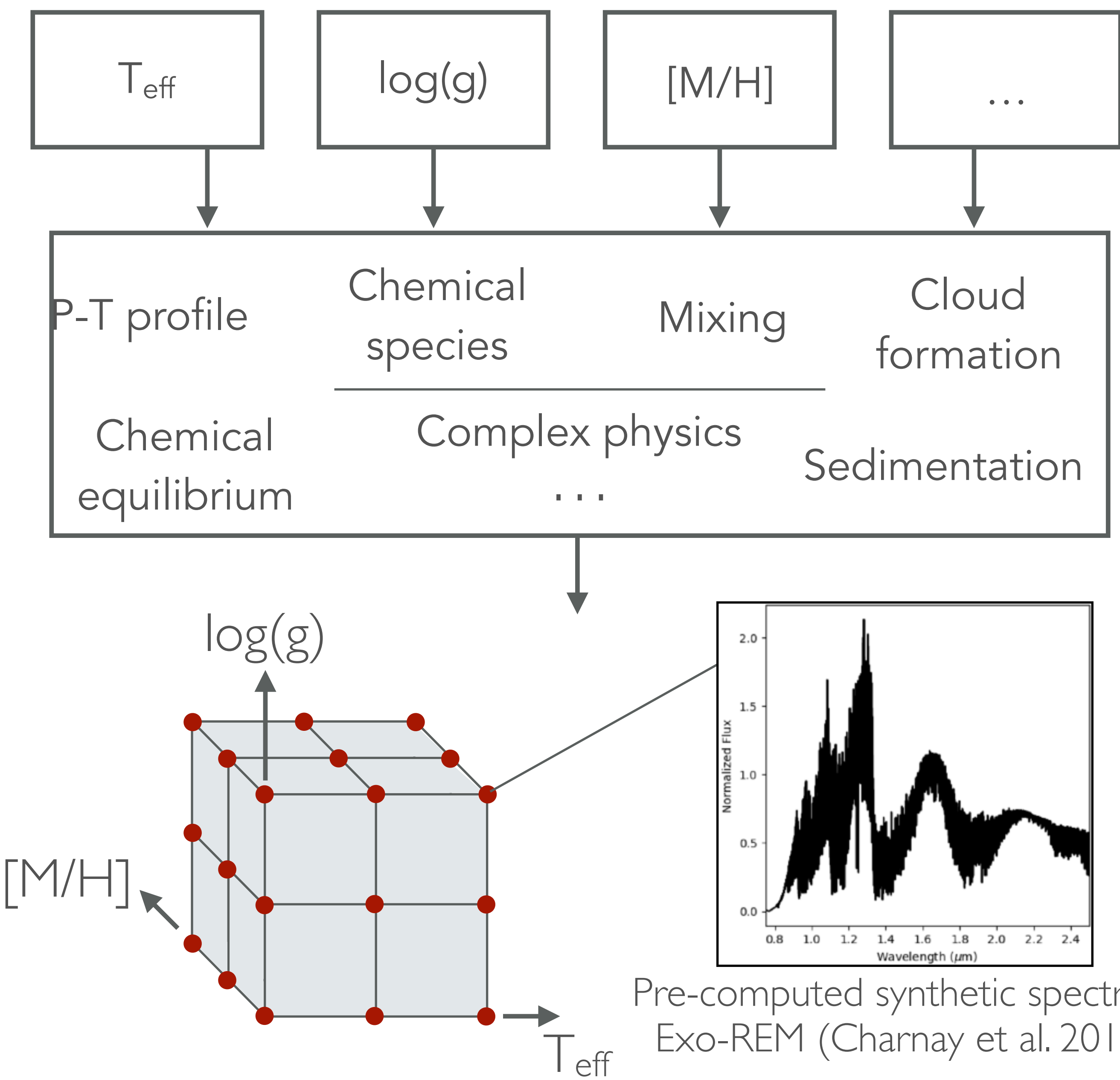
<u>Several tens of free-parameters</u>

Retrieval models: **<u>complex enough to reproduce the clouds</u>** ?

<u>Several days</u> of computing time for data with thousands points

Motivation

FORWARD MODELING



Pre-computed synthetic spectrum
Exo-REM (Charnay et al. 2018)

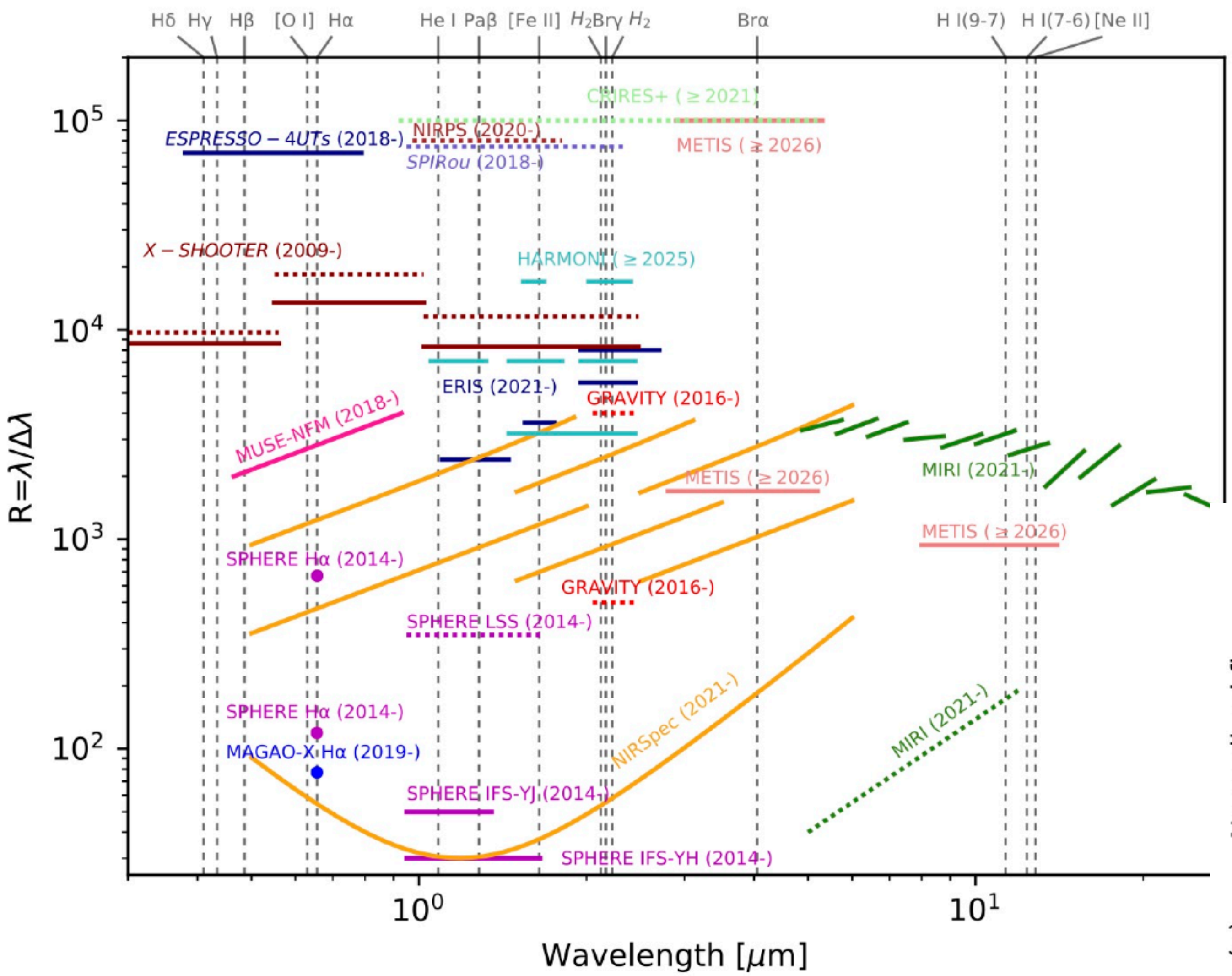
Models drive the fit

Fundamental parameters of the atmosphere (< 5 today)

Auto-consistent models

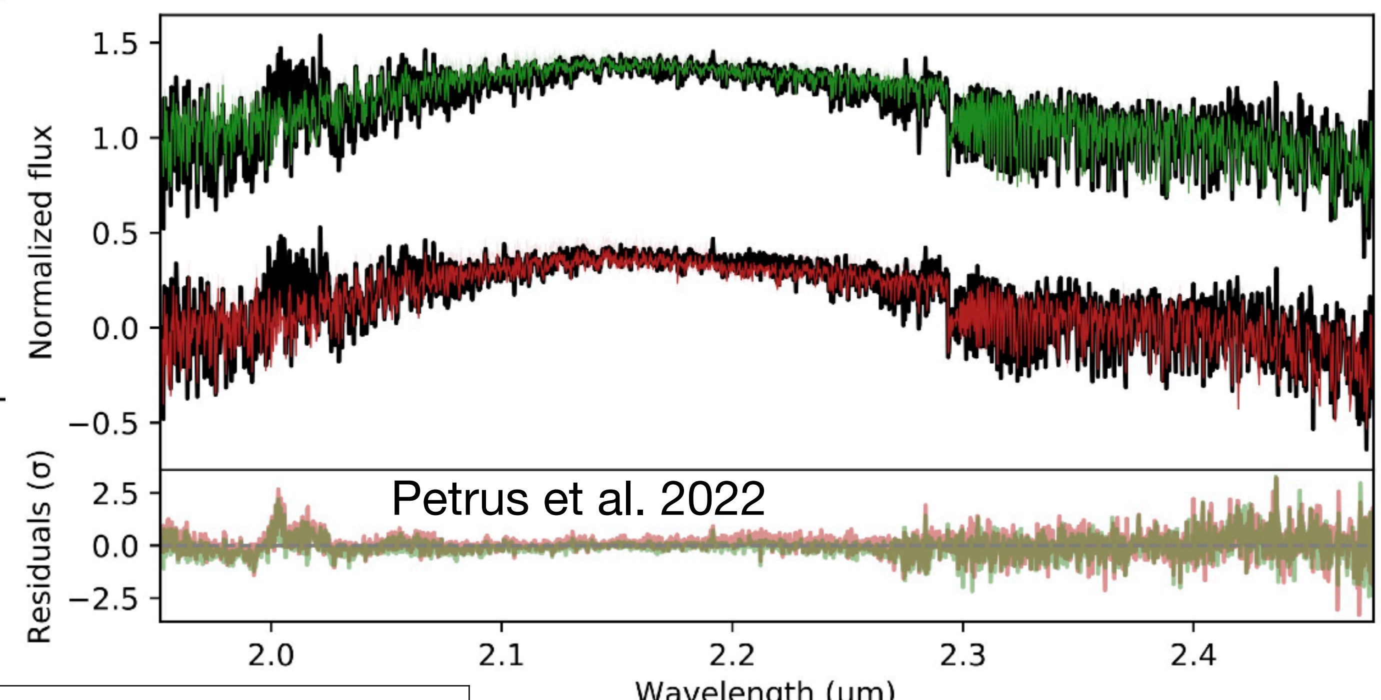
Several hours of computing time for data with thousands points

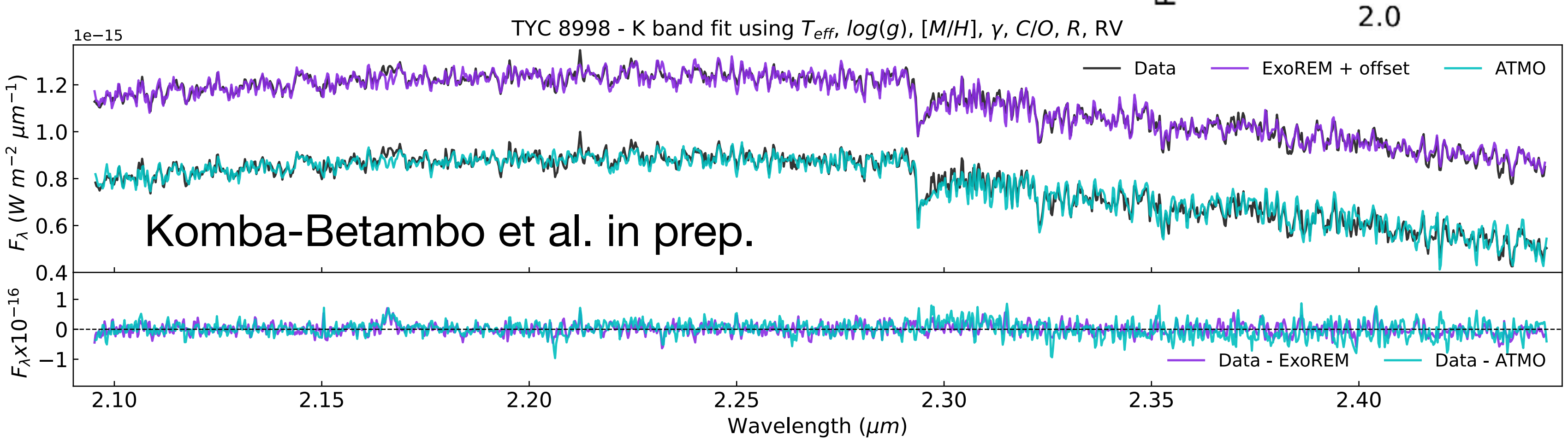
Motivation



An *embarrassment of riches*

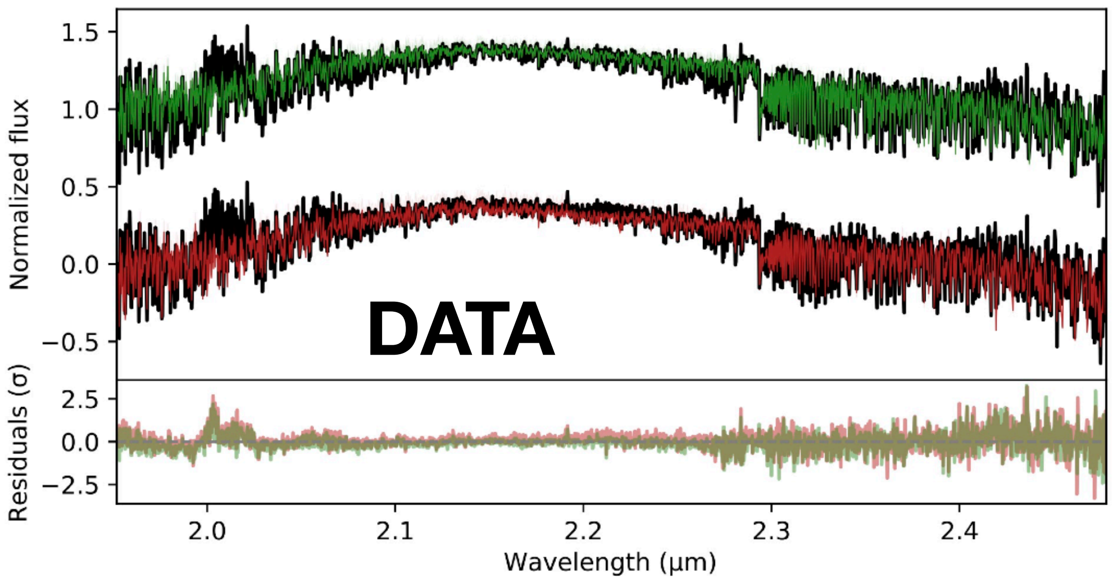
- 10^3 to 10^6 datapoints
- possibility of vast spectral libraries



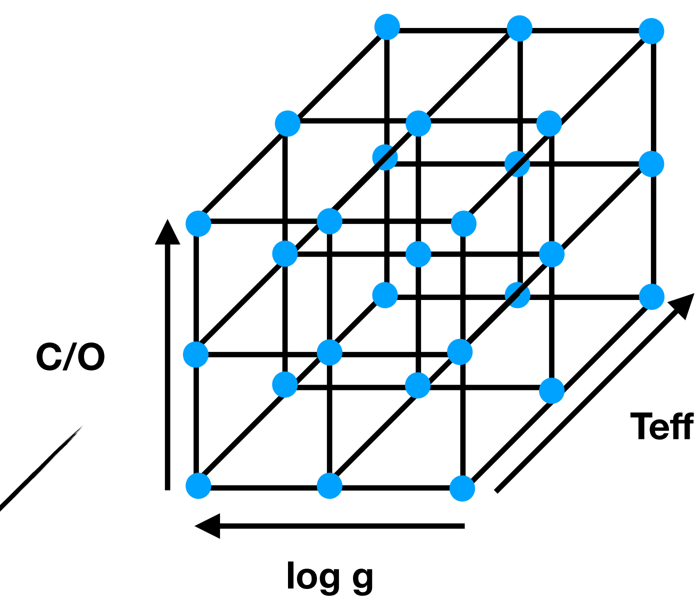


Motivation

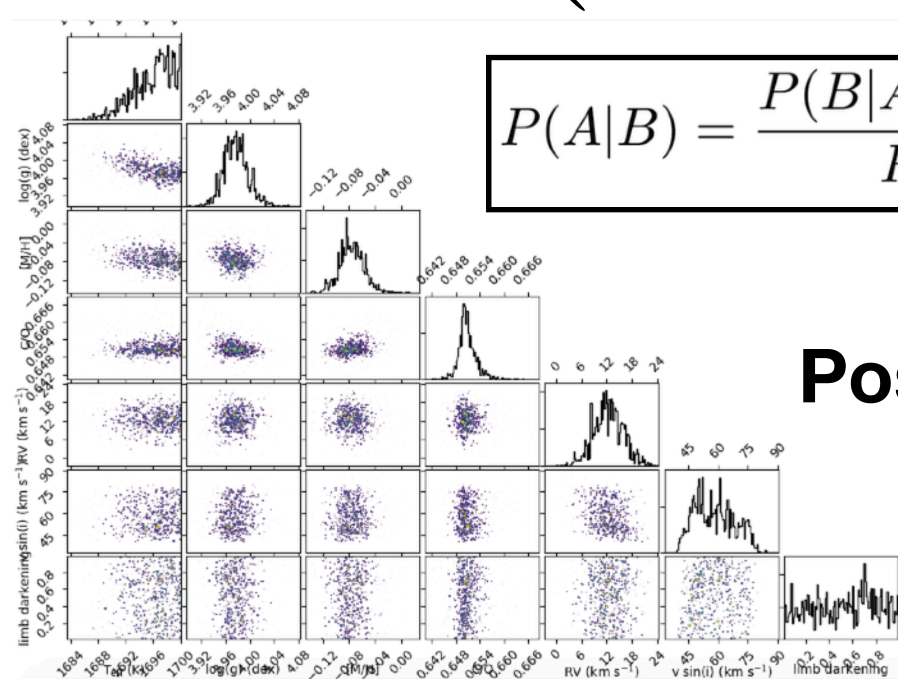
Bayesian Framework: posteriors, model selection



Model spectra



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



Posterior distributions

…but **time-consuming** (computation time explodes with dimensionality)

Motivation

Bayesian Framework: posteriors, model selection

Prior

Probability to have θ a priori.

(Information that we have)

Likelihood

Probability that θ reproduce correctly $\mathcal{D}$

(Comparison $\mathcal{M}(\theta)$ to $\mathcal{D}$)

$$\frac{\Pr(\theta \mid \mathcal{M}) \times \Pr(\mathcal{D} \mid \theta, \mathcal{M})}{\Pr(\mathcal{D} \mid \mathcal{M})} = \Pr(\theta \mid \mathcal{D}, \mathcal{M})$$

$\mathcal{D}$: Data

$\mathcal{M}$: Model

θ : Parameters

Evidence

Probability that $\mathcal{D}$ give information related to $\mathcal{M}$

(Usefull for model to model comparison)

Posterior

Probability to have θ a posteriori.

(Increased information that we have)

Motivation

Bayesian Framework: posteriors, model selection

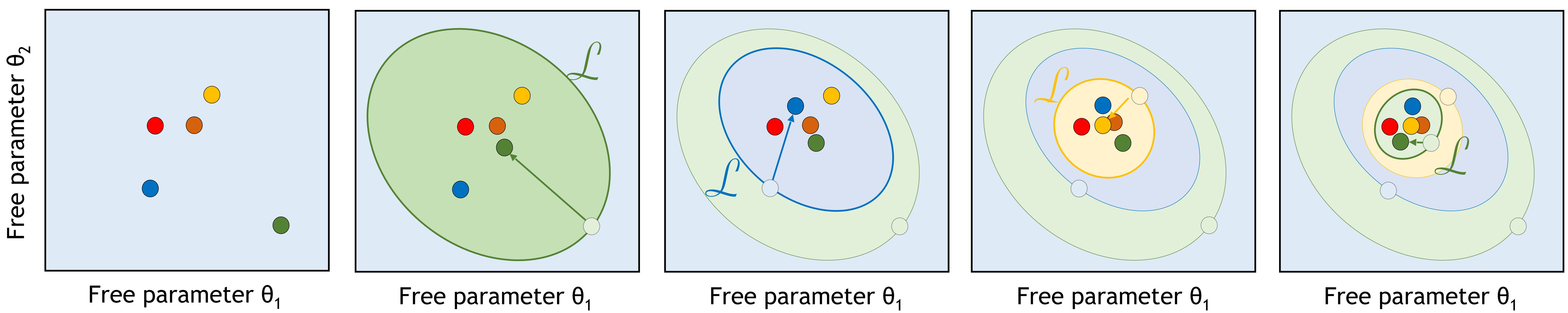
The nested sampling algorithm (alternative to MCMC)

Generate a sample of living points

Identify the point with $\mathcal{L}_{min}$ and define iso-$\mathcal{L}$

Delete this point and re-generate it in the iso-$\mathcal{L}$

Converge to $\mathcal{L}_{max}$



Iterative process

Motivation

Bayesian Framework: posteriors, model selection

Pros 👍

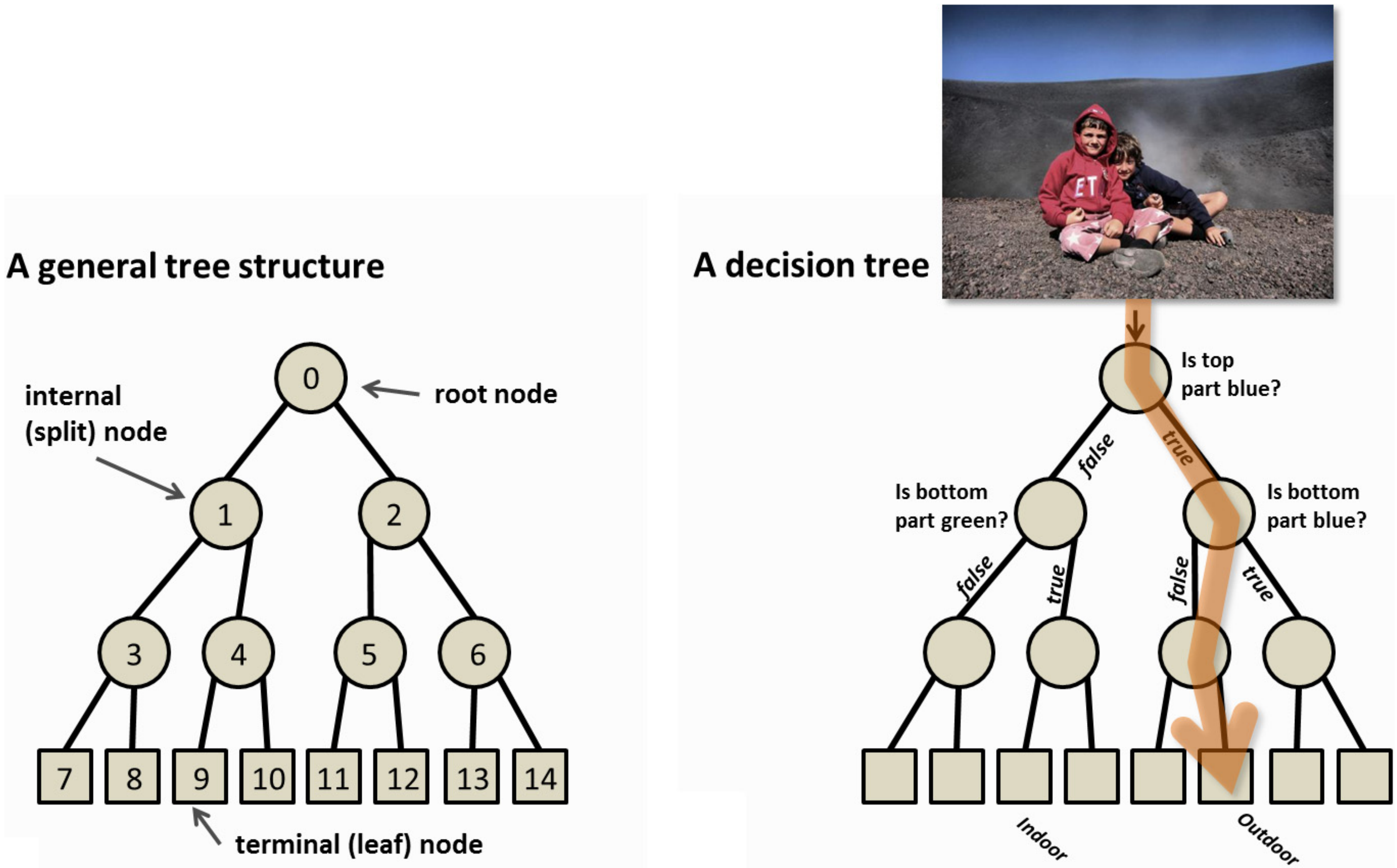
- Accurate estimate of posterior distributions
- Allows to input prior information (flat, log, normal)
- Can account for correlated noise (covariance) and penalties

Cons 👎

- Time consuming
- Non replicable inversion
- Do not relate data to constraints on free parameters

The random forests method

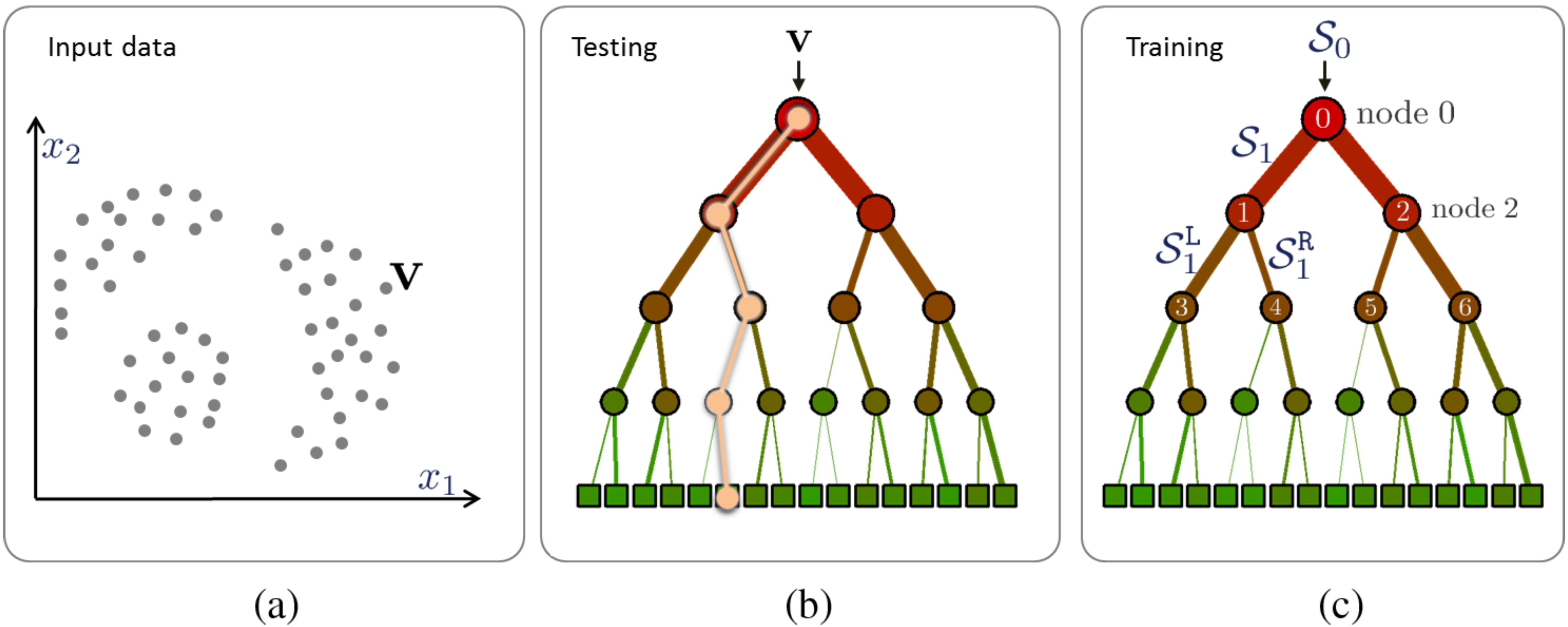
A versatile machine learning technique for regression & classification



(a) (b)

The random forests method

A versatile machine learning technique for regression & classification



The random forests method

A versatile machine learning technique for regression & classification

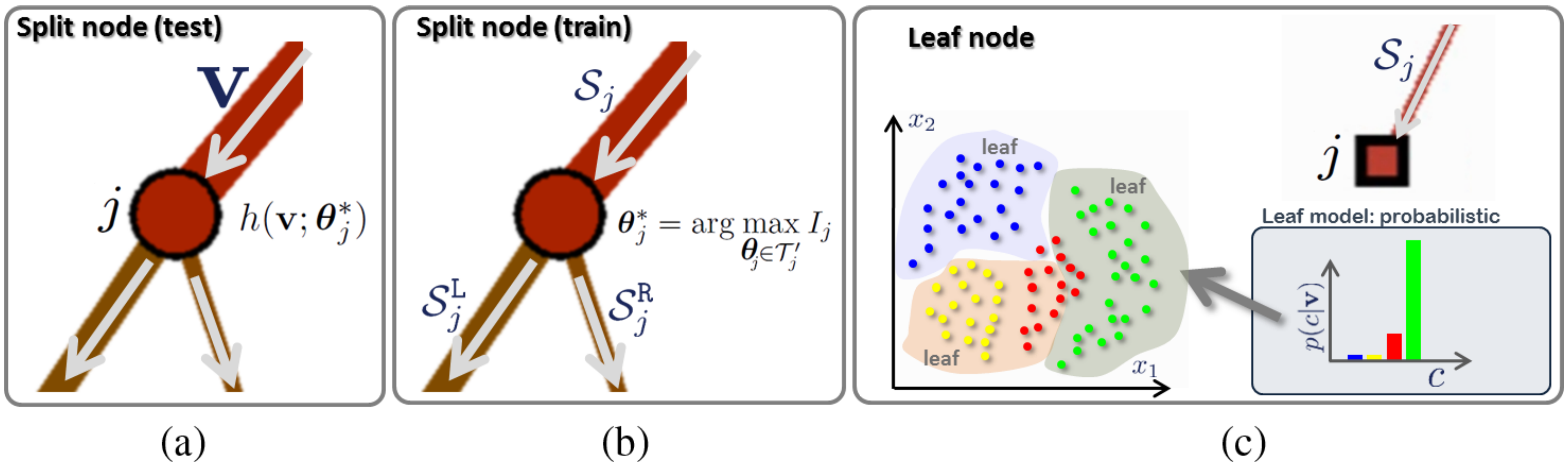
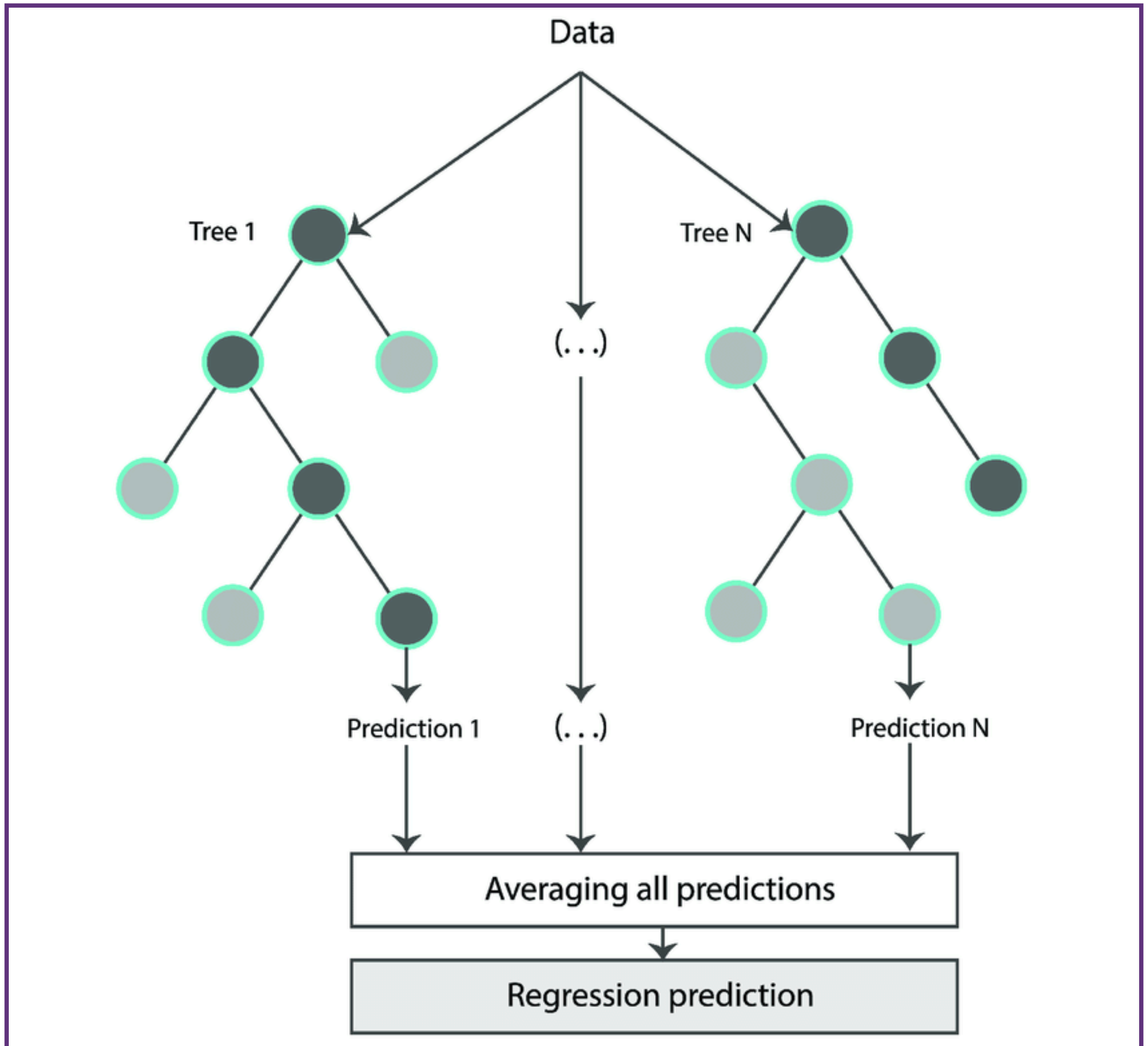


Fig. 2.3 Split and leaf nodes. (a) Split node (testing). A split node is associated with a weak learner (or split function, or test function). (b) Split node (training). Training the parameters $\boldsymbol{\theta}_j$ of node j involves optimizing a chosen objective function (maximizing the information gain I_j in this example). (c) A leaf node is associated with a predictor model. For example, in classification we may wish to estimate the conditional $p(c|\mathbf{v})$ with $c \in \{c_k\}$ indicating a class index.

The random forests method

A versatile machine learning technique for regression & classification



N trees : bagging (Mont-Carlo)

$$y = f(x)$$

Dependent variable
Atmospheric parameters

Feature vector
Spectrum

The random forests method

Original Framework

Application (code) : scikit-learn

Application (exoplanet atmospheres) : Marquez-Neila et al. 2018
Oreshenko et al. 2019 **HELA code**

1 - projection of spectrum errors on grids (monte-carlo)

1.5 - reinterpolation of grids (finer mesh)

2 - training on grid (.fit)

3 - regression (.predict)

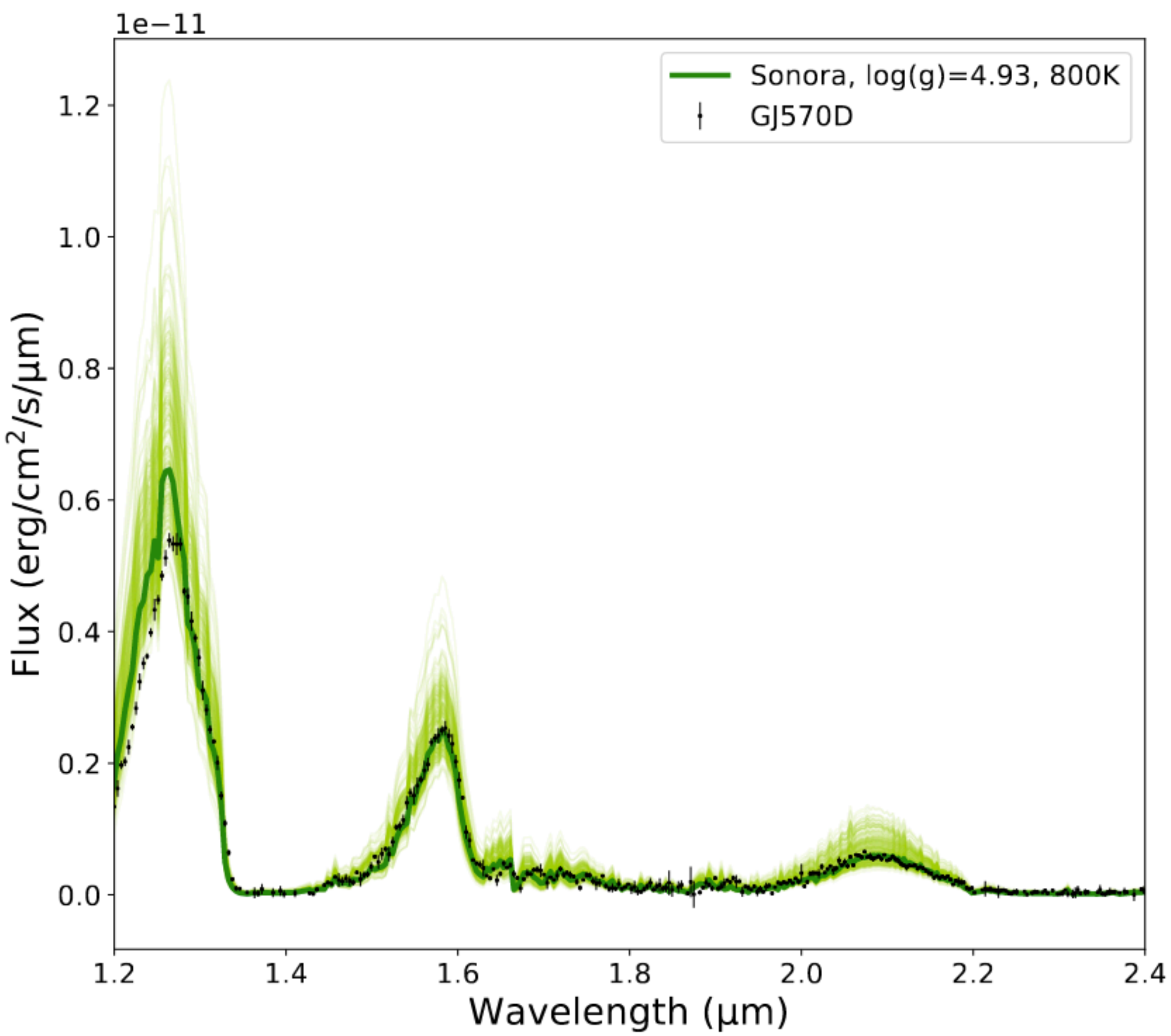
The random forests method

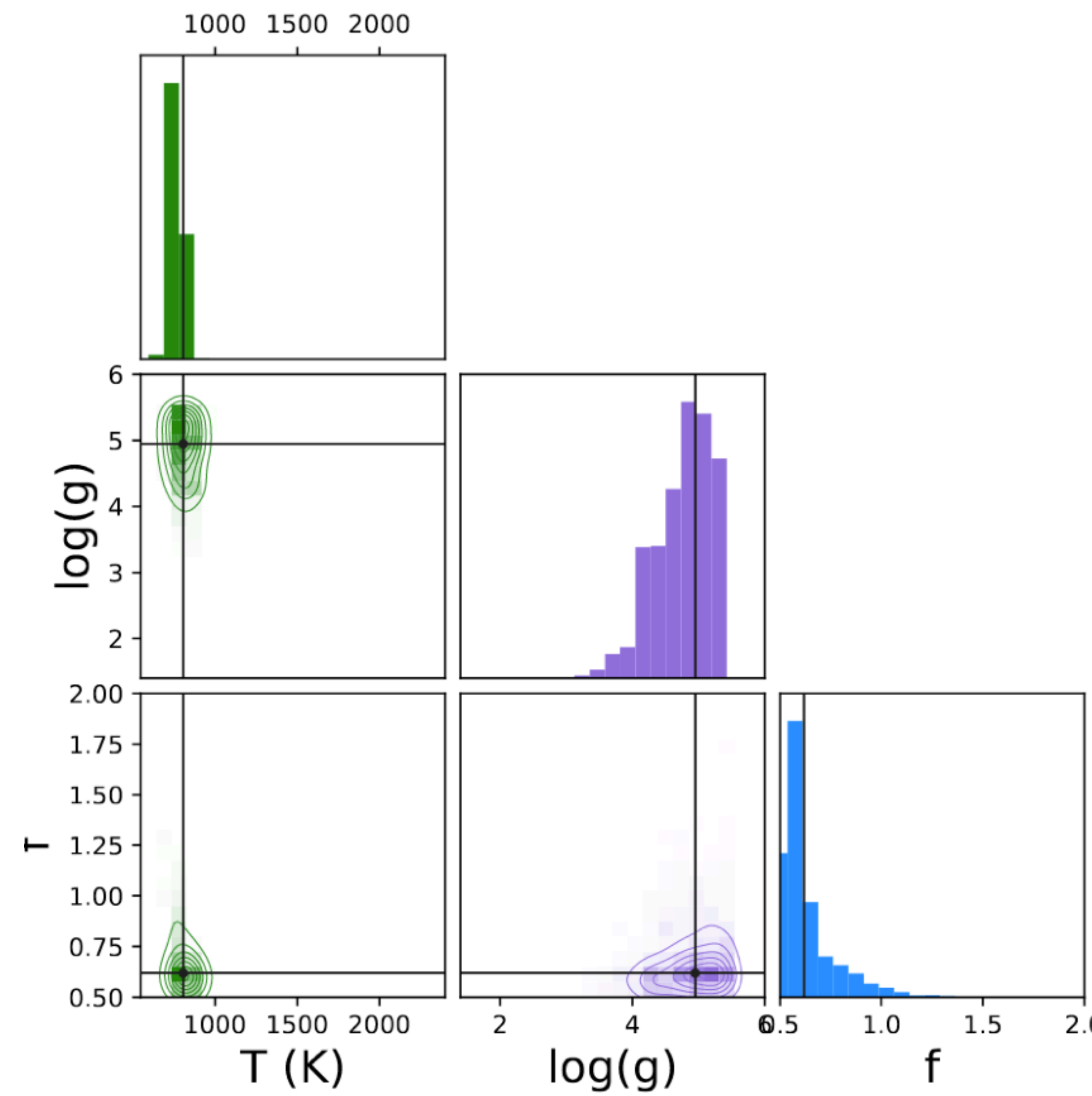
Original Framework

Application (code) : scikit-learn

Application (exoplanet atmospheres) : Marquez-Neila et al. 2018
Oreshenko et al. 2019

HELA code

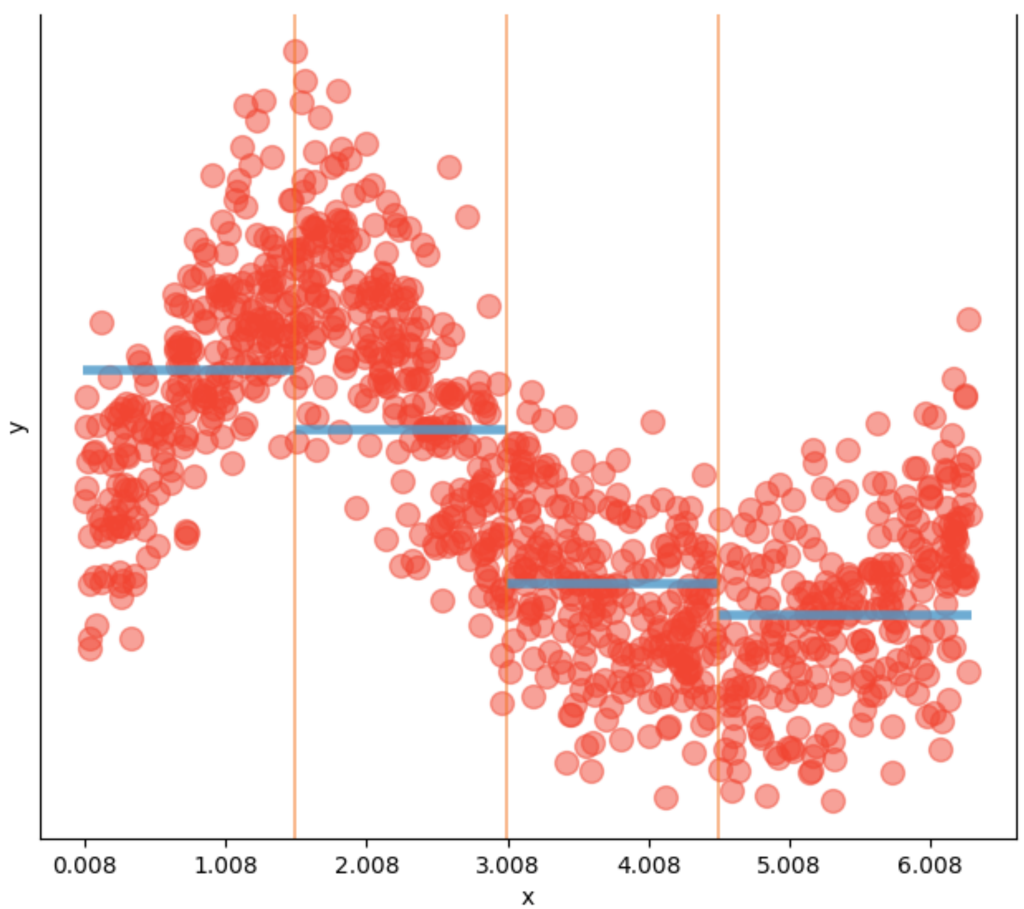


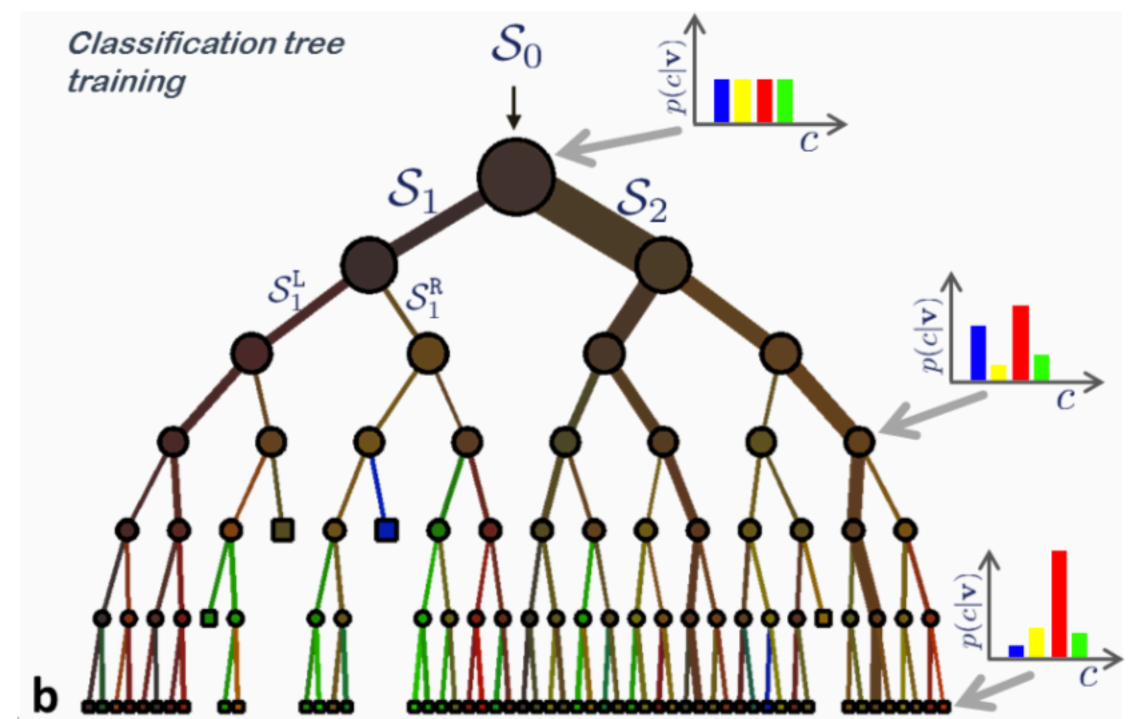


The random forests method

sklearn.ensemble.RandomForestRegressor

- Split function: mean-square error (variance reduction)
- Fit function: constant per partition



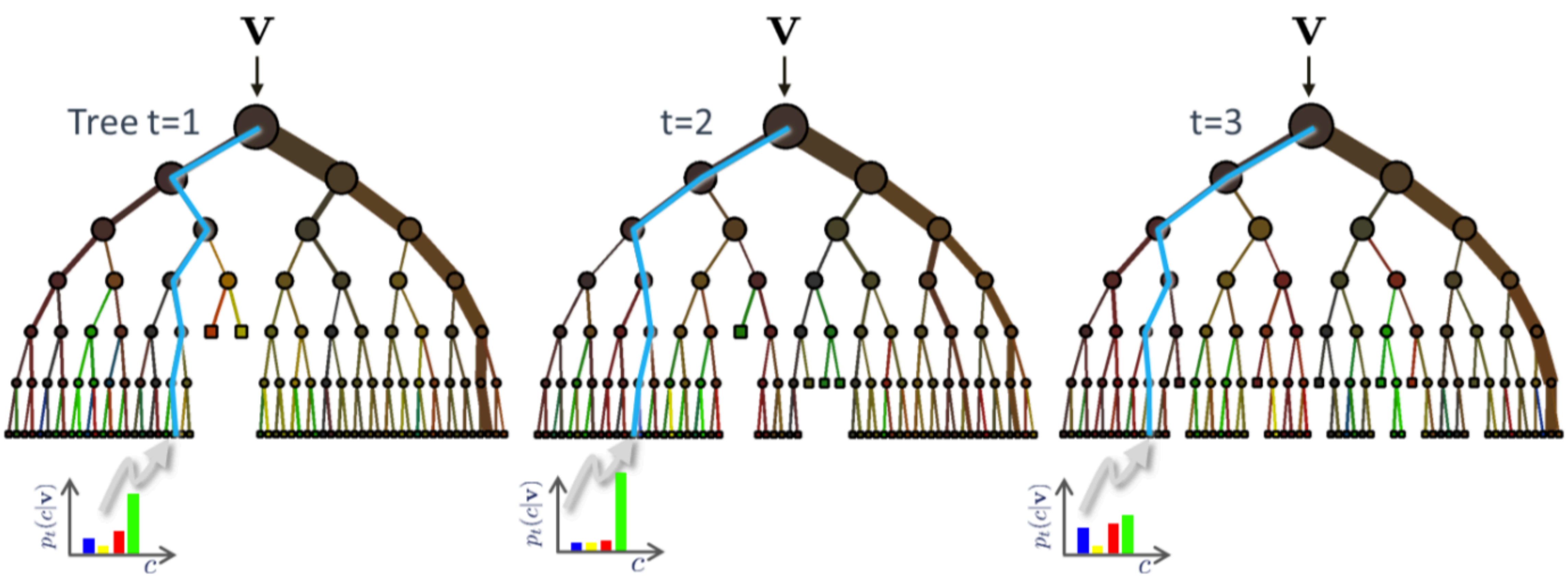


(b)

The random forests method

`sklearn.ensemble`.RandomForestRegressor

- Split function: mean-square error (variance reduction)
- Fit function: constant per partition



The random forests method

`sklearn.ensemble`.RandomForestRegressor

- Outputs

`predict`(*X*) [source]

Predict regression target for X.

The predicted regression target of an input sample is computed as the mean predicted regression targets of the trees in the forest.

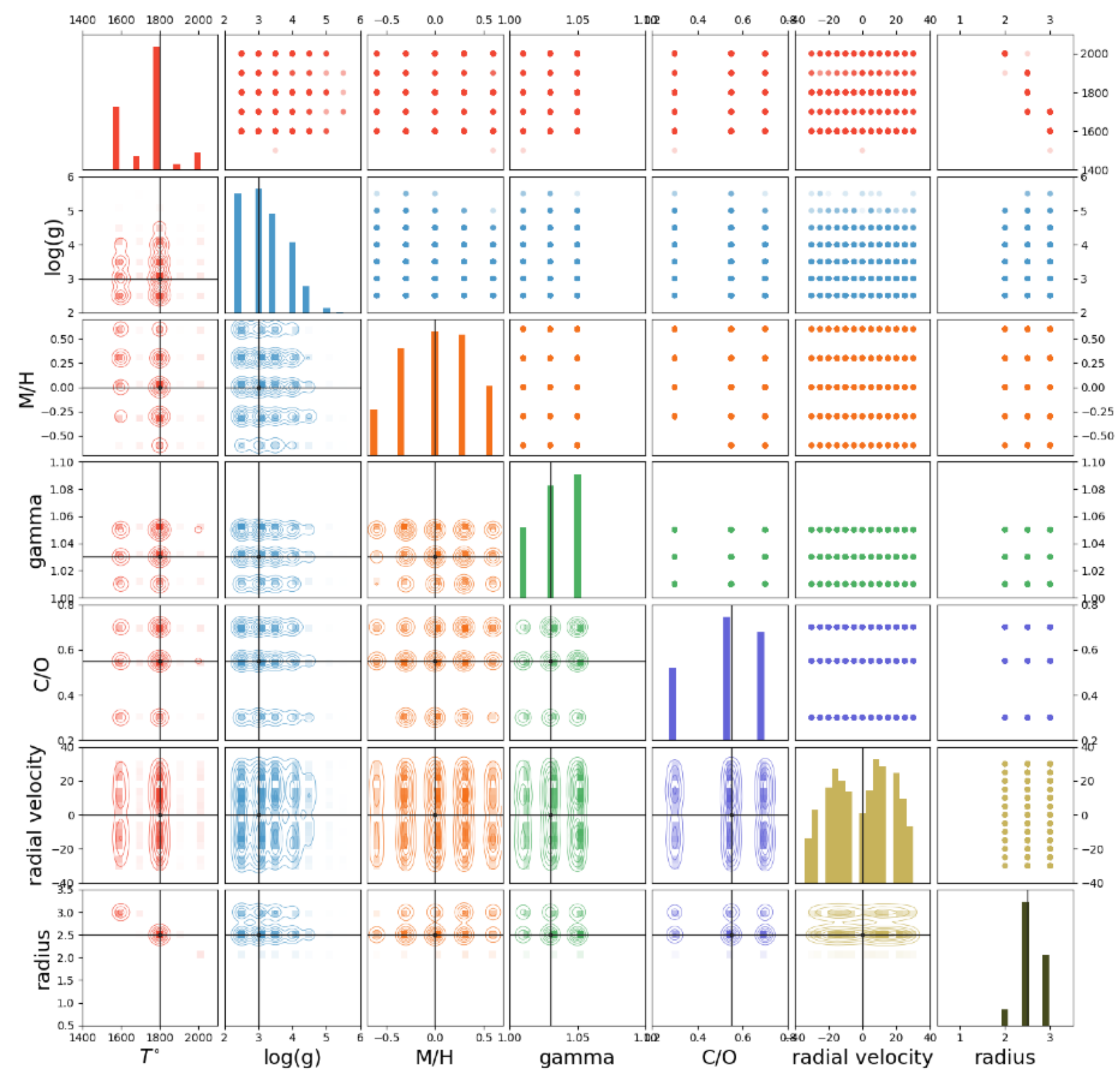
Parameters::	**X** : ***{array-like, sparse matrix} of shape (n_samples, n_features)*** The input samples. Internally, its dtype will be converted to `dtype=np.float32`. If a sparse matrix is provided, it will be converted into a sparse `csr_matrix`.
Returns::	**y** : ***ndarray of shape (n_samples,) or (n_samples, n_outputs)*** The predicted values.

The random forests method

sklearn.ensemble.RandomForestRegressor

- Outputs

The outputs are depends of the original grid mesh!

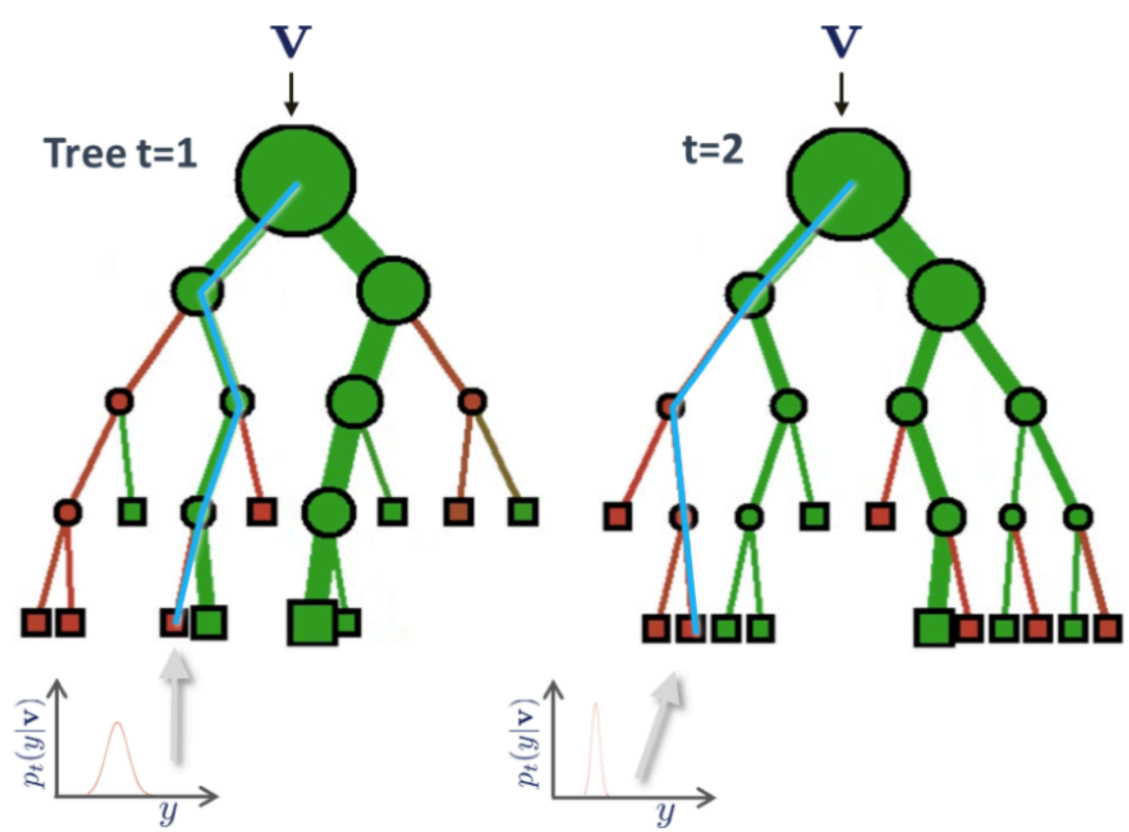


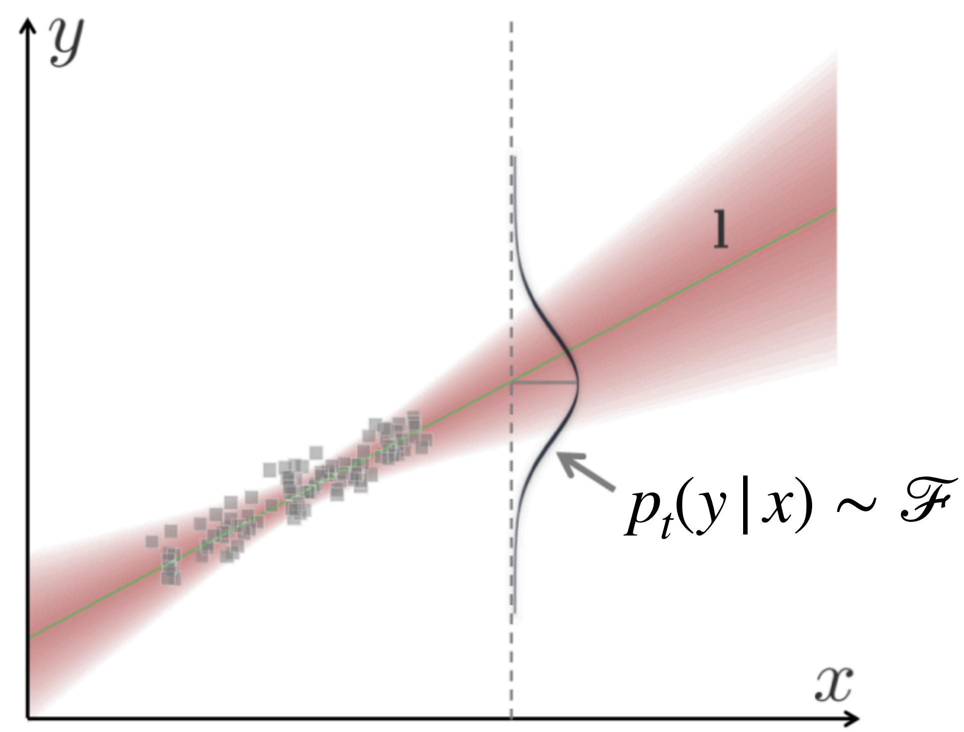
HELA code (Marquez-Neila et al. 2018)

The random forests method

Modified framework (Criminisi et al. 2012):

Probabilistic linear regression post-partition

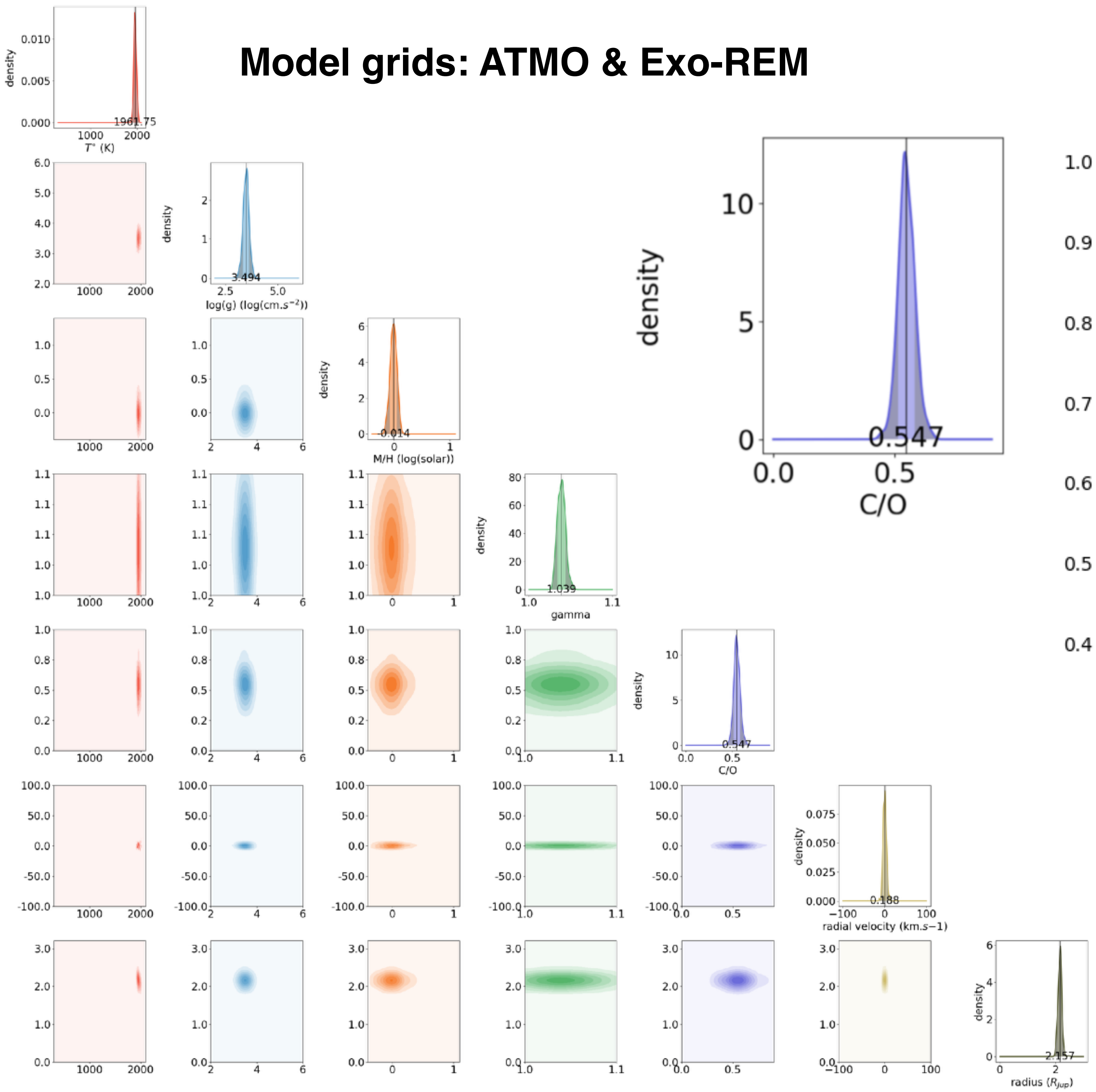


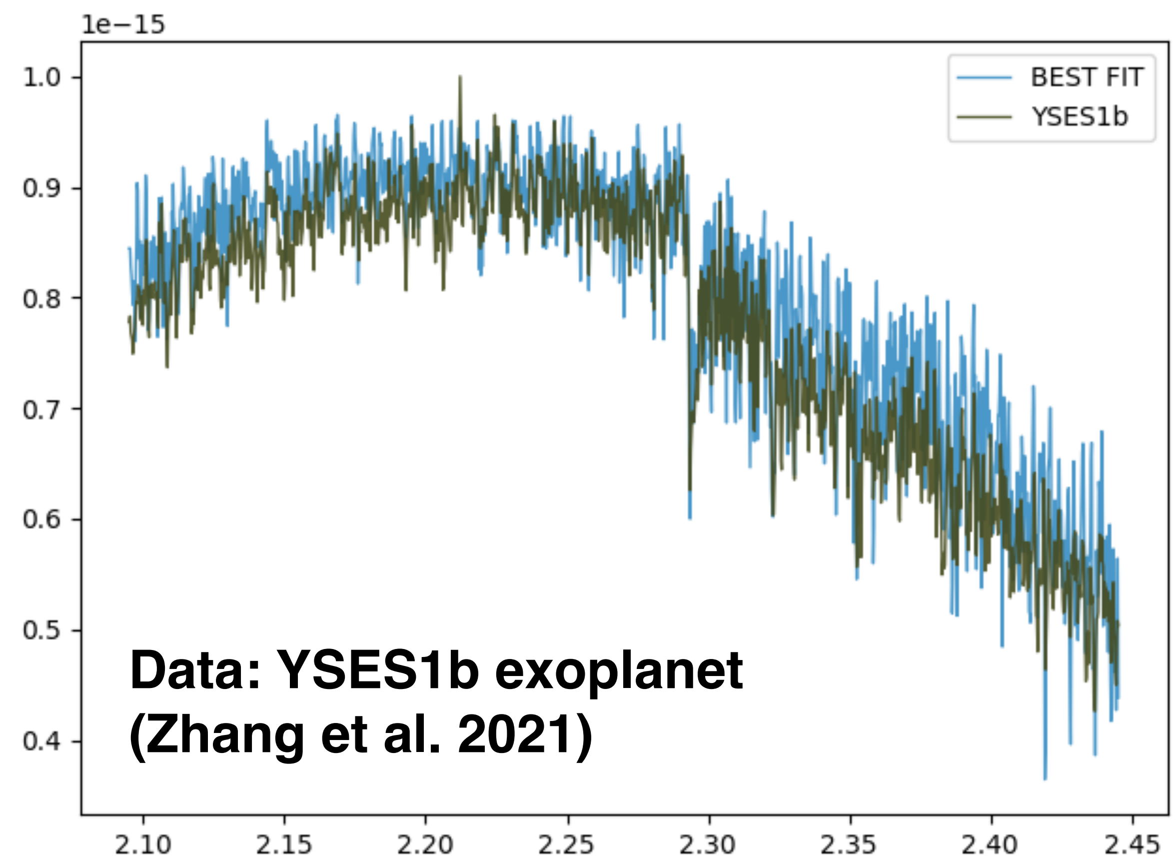


$$p(y|x) = \frac{1}{T}\sum_{t}^{T} p_t(y|x)$$

Application

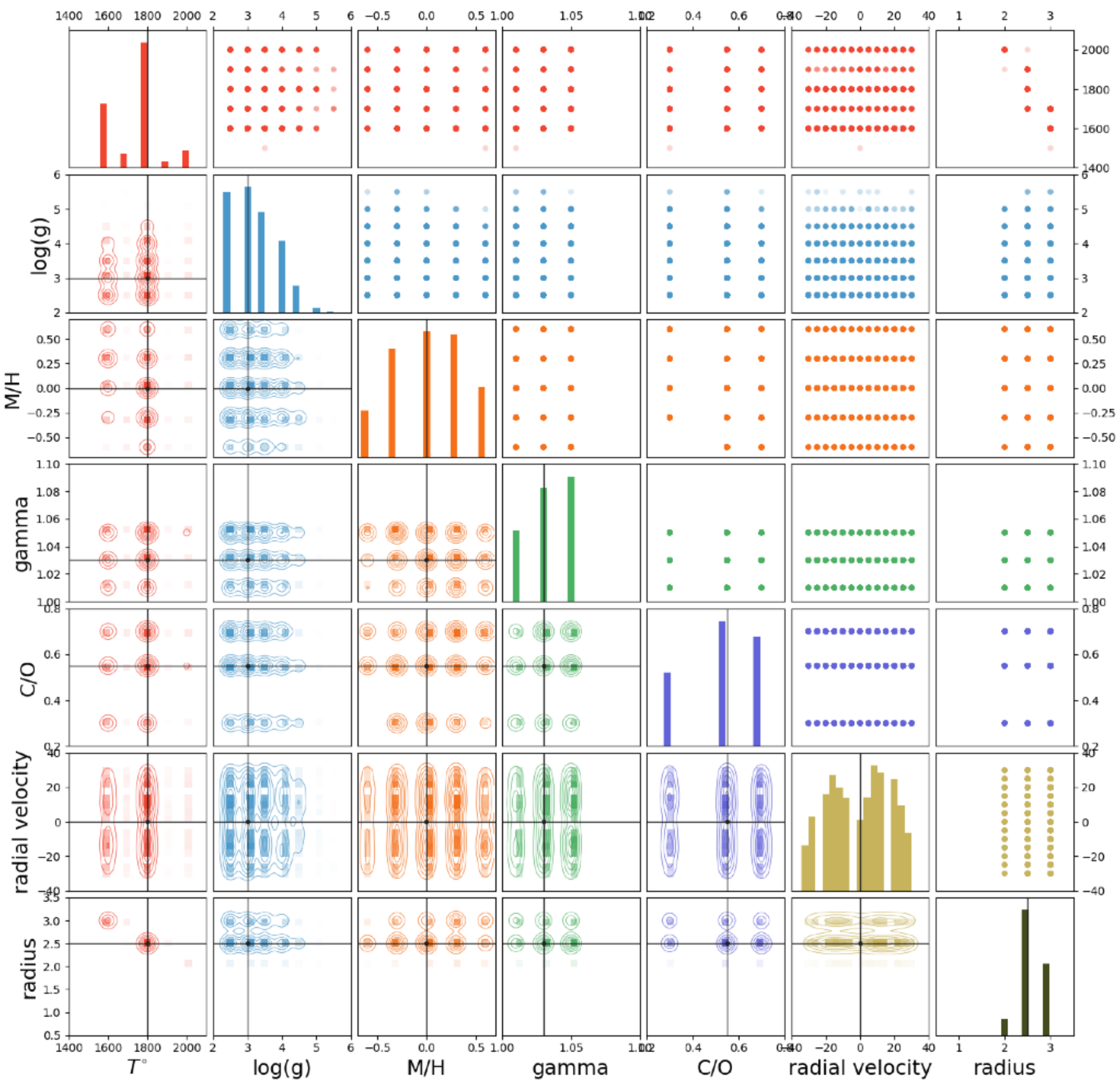
Model grids: ATMO & Exo-REM



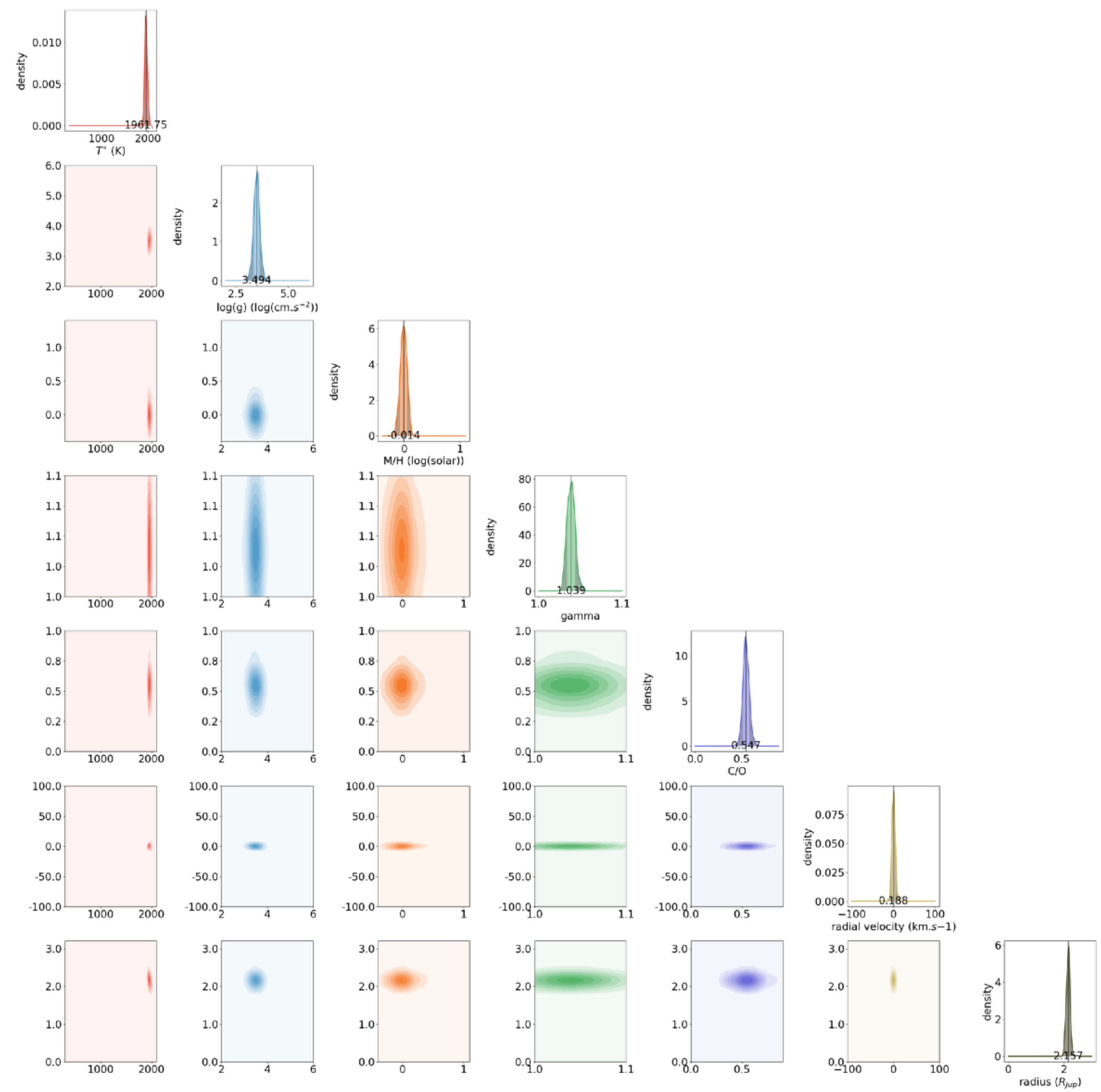


Data: YSES1b exoplanet (Zhang et al. 2021)

Pros & cons

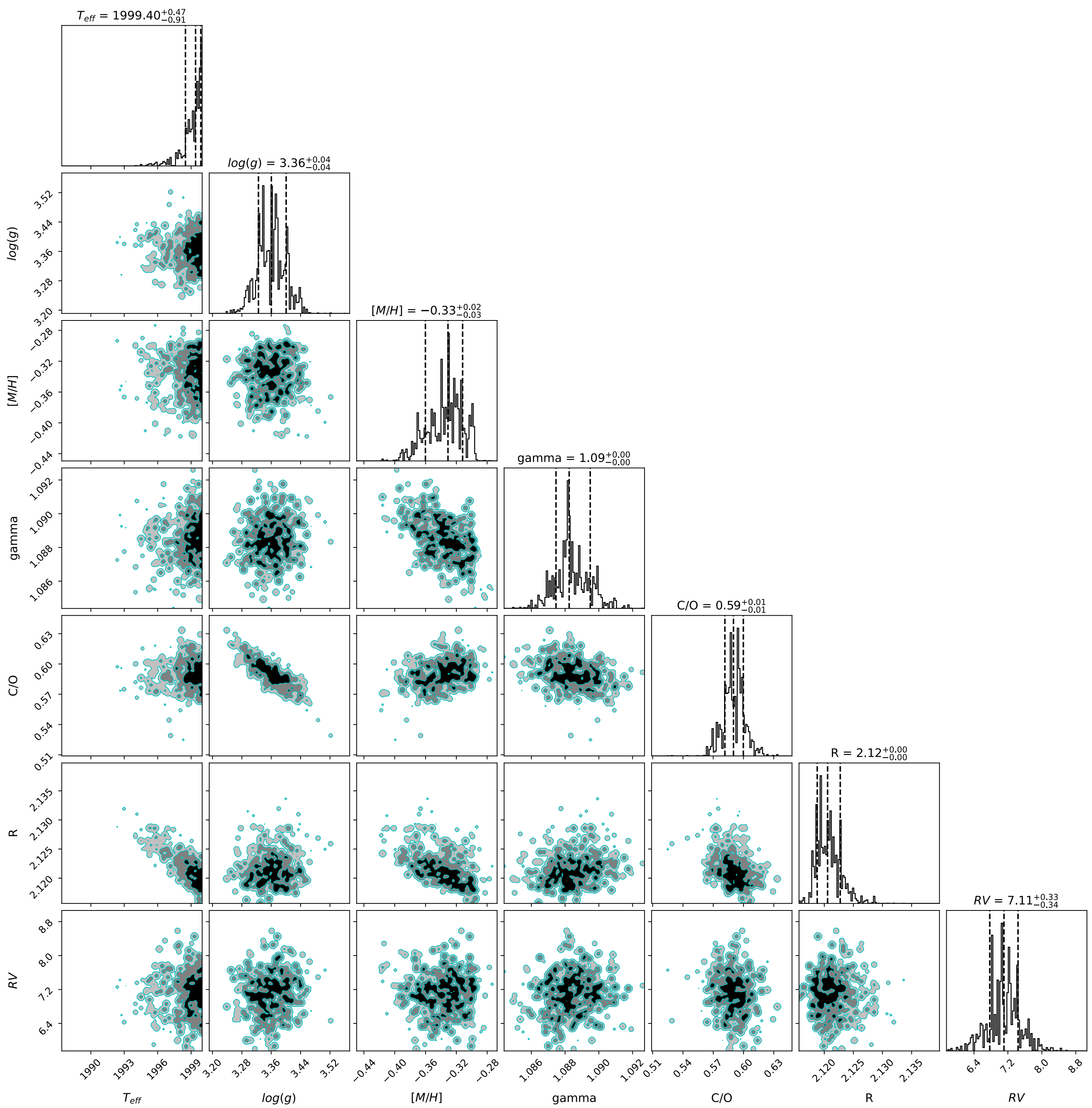


HELA code (Marquez-Neila et al. 2018)

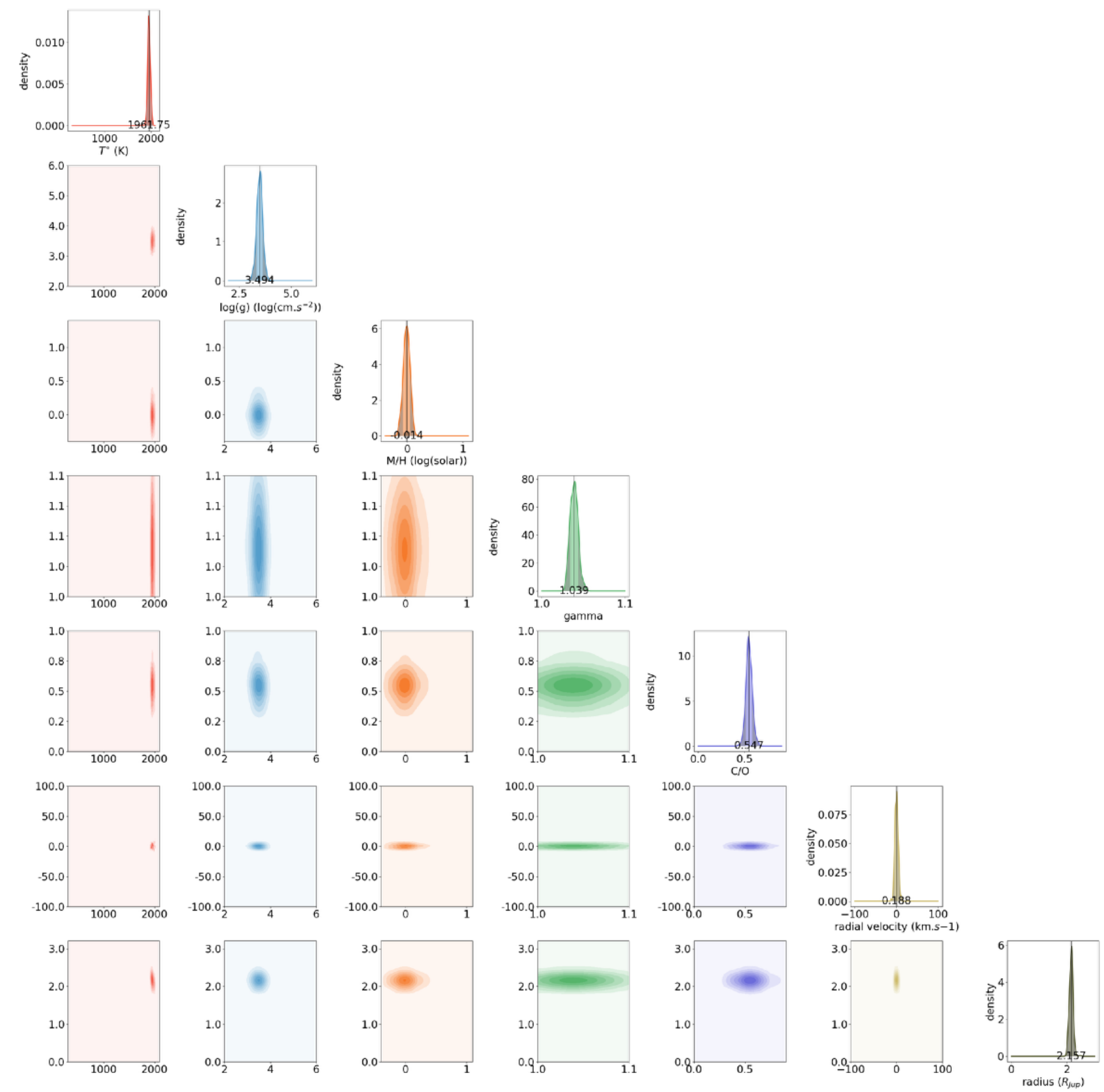


PCLRF (Komba-Betambo et al. 2022)

Pros & cons
YSES1b

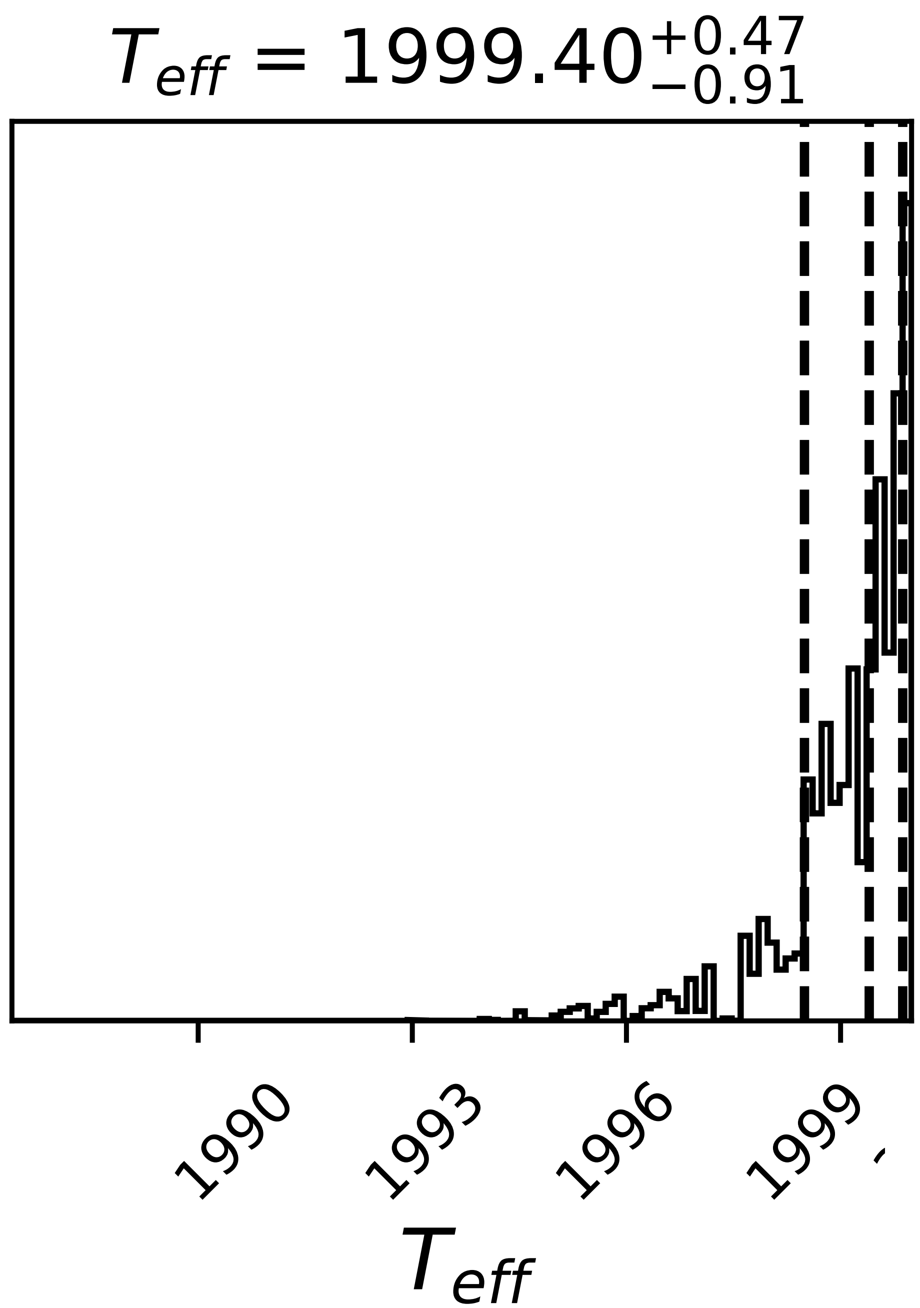


ForMoSa code (Petrus et al. 2020)

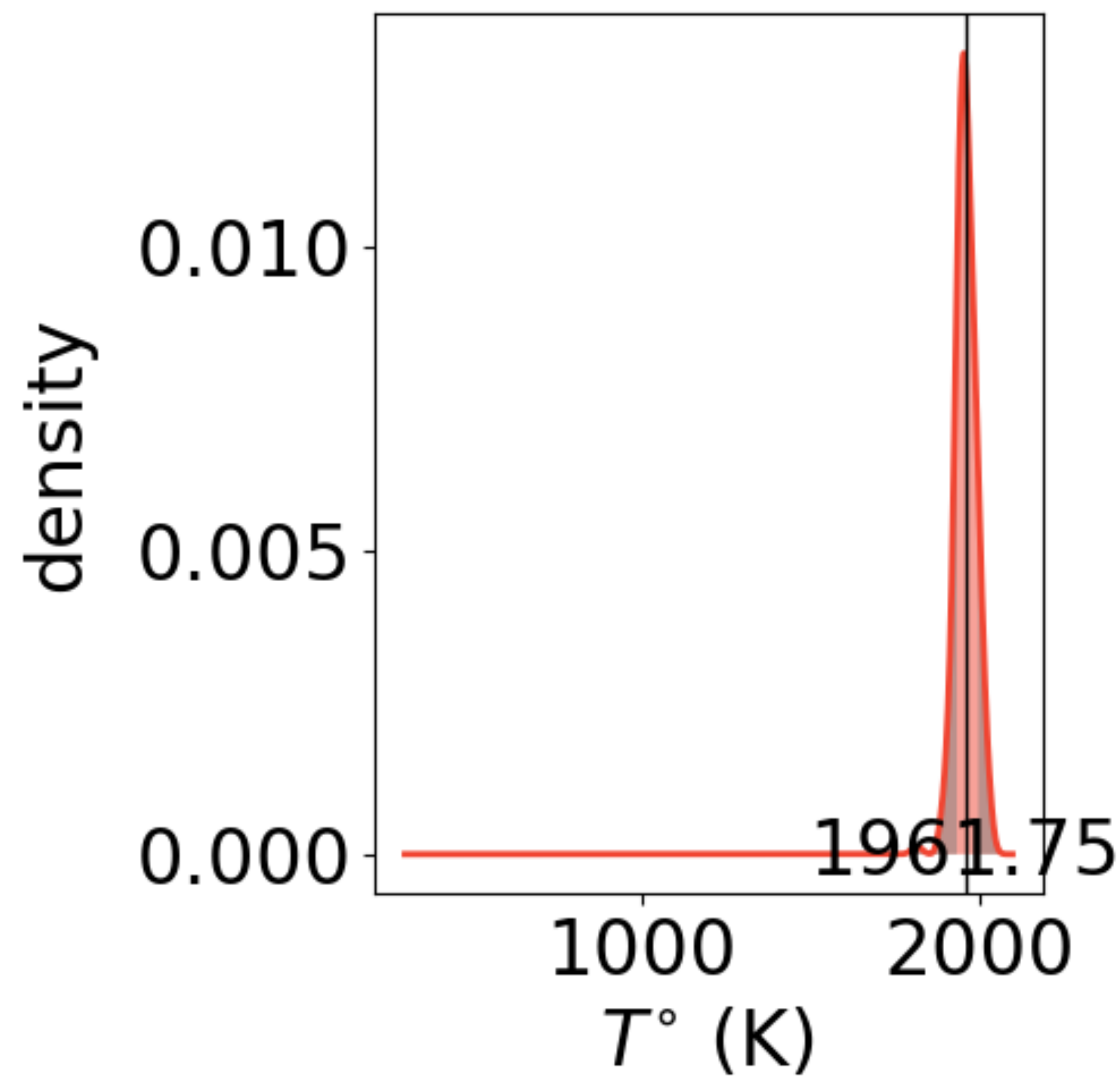


PCLRF (Komba-Betambo et al. 2022)

Pros & cons
YSES1b



ForMoSa code (Petrus et al. 2020)



PCLRF (Komba-Betambo et al. 2022)

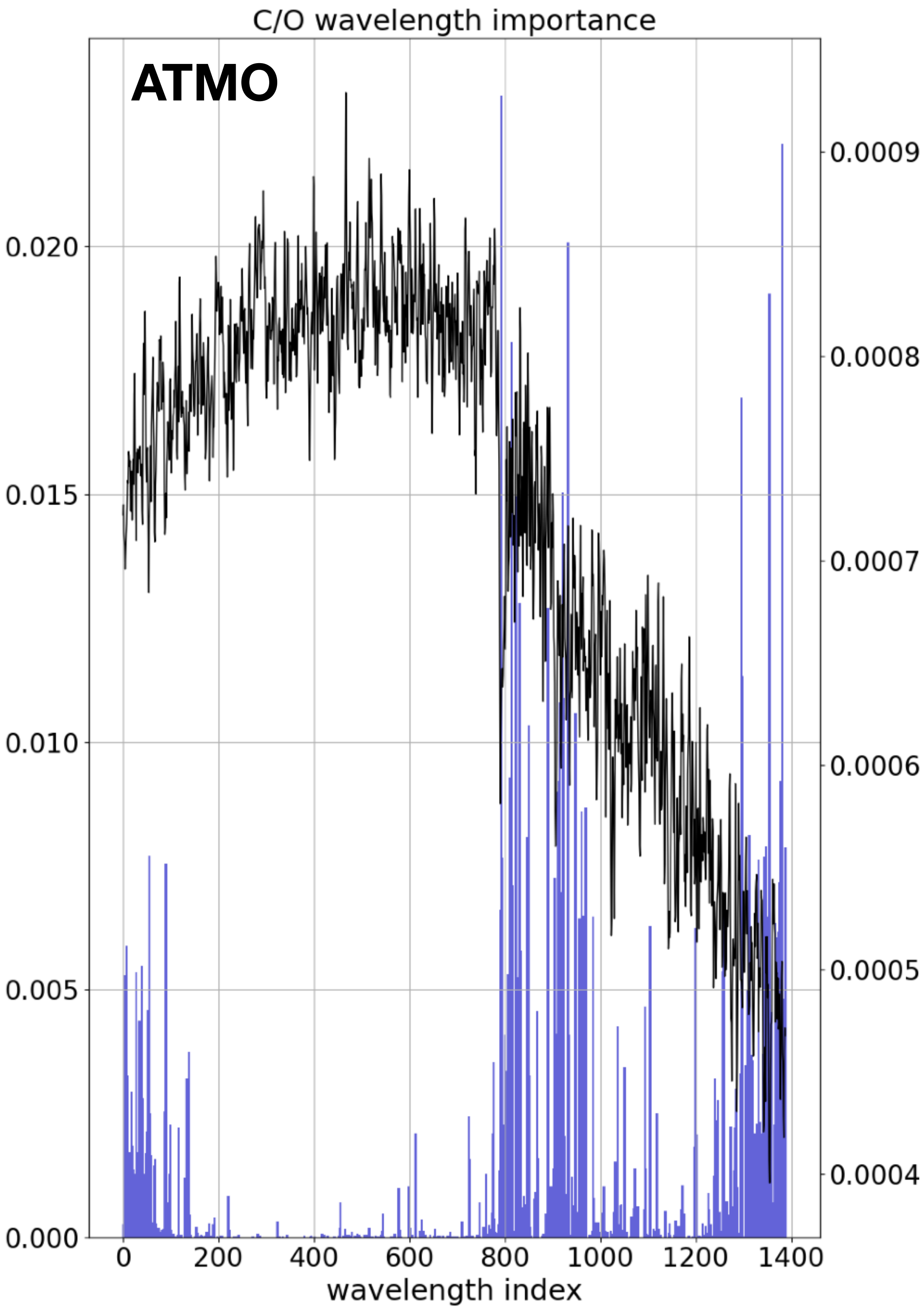
Pros & cons

AB Pic b

Table 1: Retrieved model parameters for AB Pic b using the Nested Sampling (ForMoSA-NS) and PCLRF approaches

Method	T_{eff} (K)	log g (dex)	M/H (dex)	γ	C/O	radial velocity ($km.s^{-1}$)	radius (R_{Jup})
ForMoSA-NS	2056^{+10}_{-11}	≤ 3.01	$-0.31^{+0.01}_{-0.04}$	1.08 ± 0.01	$0.69^{+0.01}_{-0.01}$	1.21 ± 0.01	1.21 ± 0.01
PCLRF	2121^{+40}_{-41}	$3.19^{+0.22}_{-0.22}$	$-0.11^{+0.1}_{-0.09}$	$1.03^{+0.006}_{-0.004}$	$0.62^{+0.04}_{-0.04}$	$11.4^{+4.4}_{-4.8}$	$1.08^{+0.046}_{-0.044}$
RF [1]	2200^{+0}_{-500}	$3^{+1}_{-0.5}$	$-0.3^{+0.6}_{-0.3}$	$1.03^{+0.02}_{-0.02}$	$0.55^{+0.15}_{-0.0}$	16^{+24}_{-24}	$1.0^{+0.5}_{-0.0}$

Pros & cons



Pros

- Computation time

  training = 10min
  regression = 34s
  (Comparison : Bayesian = 1 nuit)

  Application on massive datasets

- Feature importance plot

  Maximise S/N for key wavelengths

Cons

- We loose control on the model
- Treatment of uncertainties

Komba-Betambo et al. 2022 (in prep)