自動微分可能なスペクトル第一原理フィットツールの開発と高温下の分子線幅測定計画

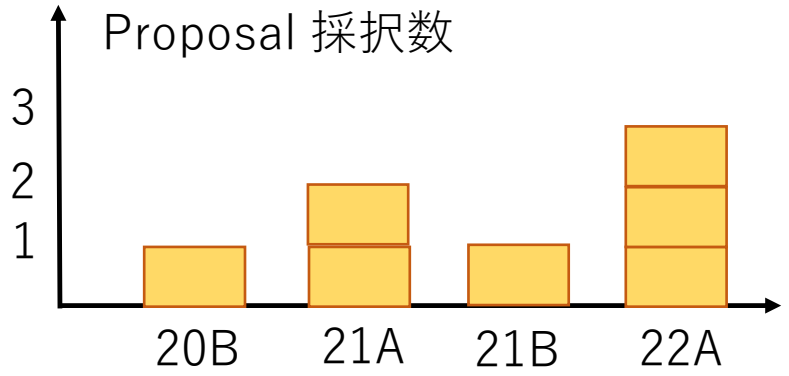
Hajime Kawahara, Yui Kawashima, Kento Masuda, Ian Crossfield, Erwan Pannier, van den Bekerom, Takayuki Kotani, Kazuo Yoshioka, Tako Ishikawa, Stevanus Nugroho, and REACH collaboration

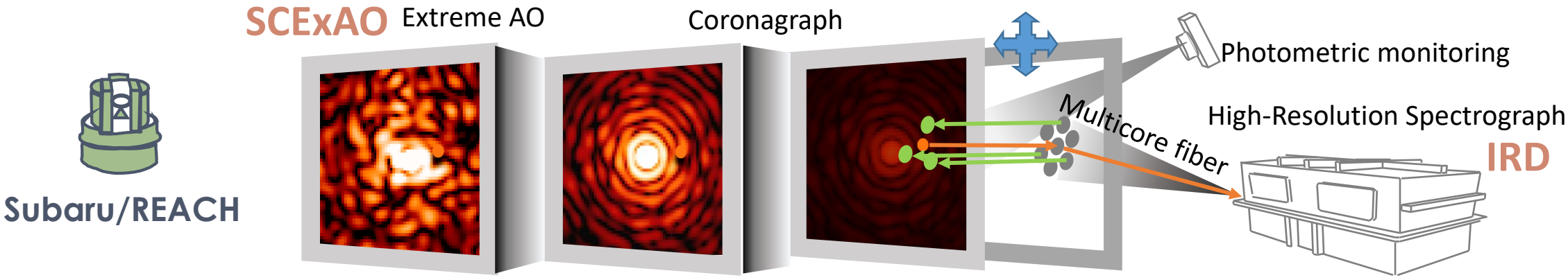
Kawahara+ ApJS in press 2105.14782

REACH: High-Dispersion Spectrum for High-Contrast Instruments

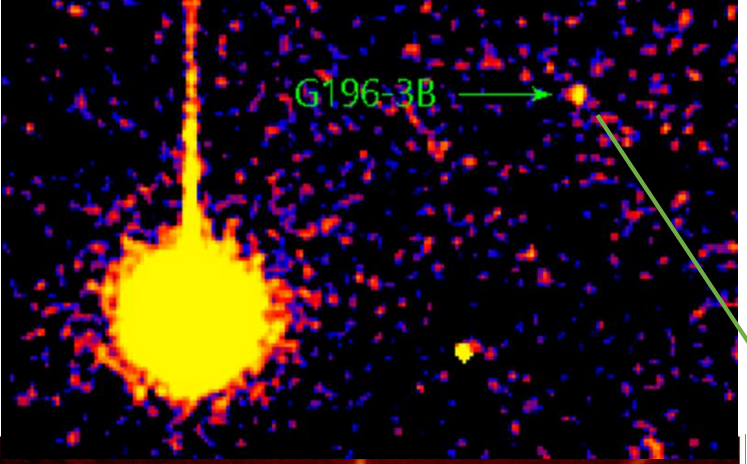
High-Dispersion Coronagraphy on Subaru

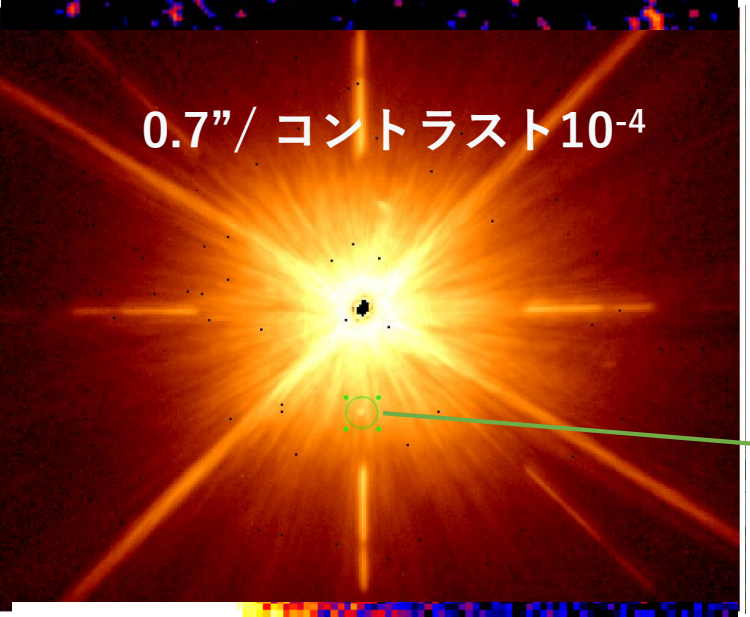
Kawahara, Murakami, Matsuo, and **Kotani** (2014) ApJS
Late 2014: Started to develop such an instrument
with SCExAO team (**Guyon, Lozi, Jovanovic** etc)
and IRD (**Kotani**, **Vievard** etc) on Subaru
Late 2020: available for science use (open use)

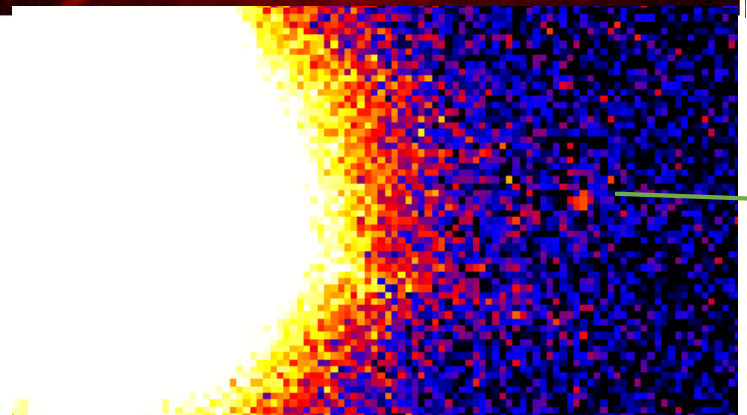




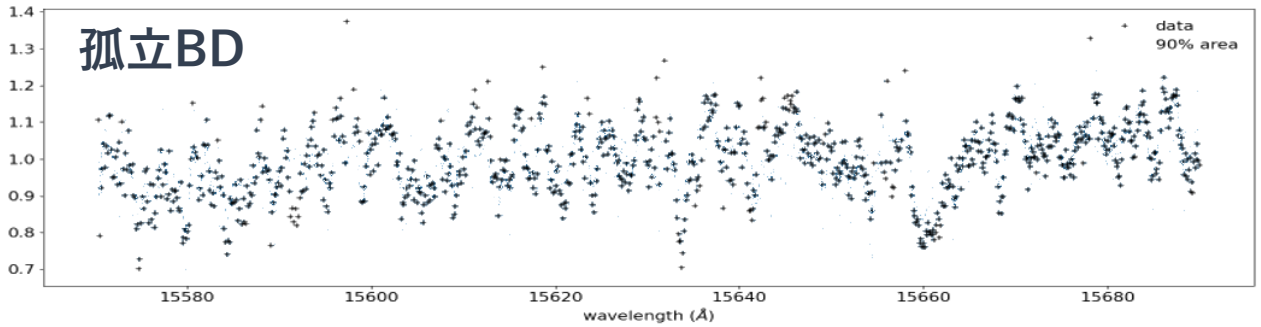
REACH (Subaru):	y, J, H	R=100,000 (SCExAO + IRD)
KPIC (Keck):	K, L, M	R=30,000 (NIRSPEC)
HiRISE (VLT):	Under construction H, K, …	R=100,000 (SPHERE + CRIRES+)


G196-3B

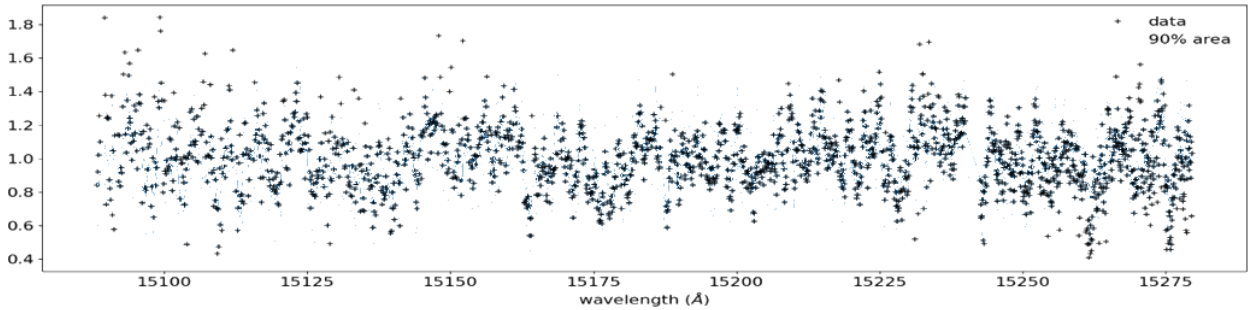

0.7”/ コントラスト10-4



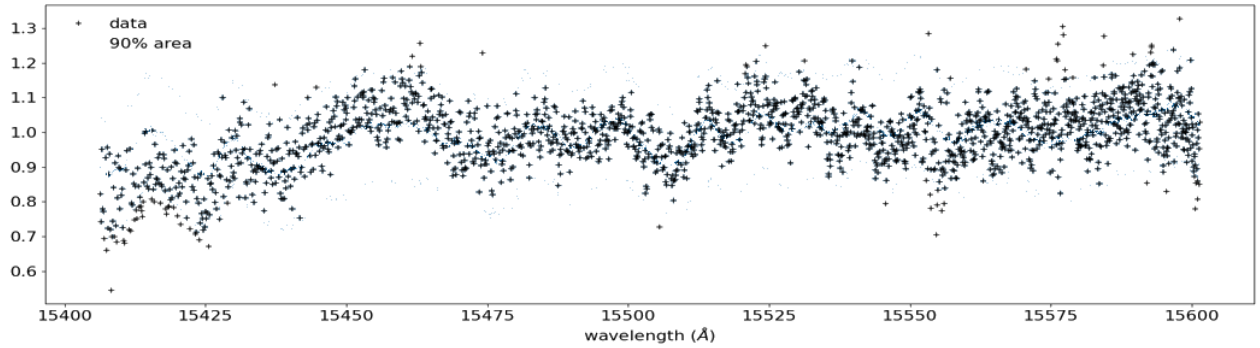
L1


孤立BD
data
90% area
wavelength (Å)

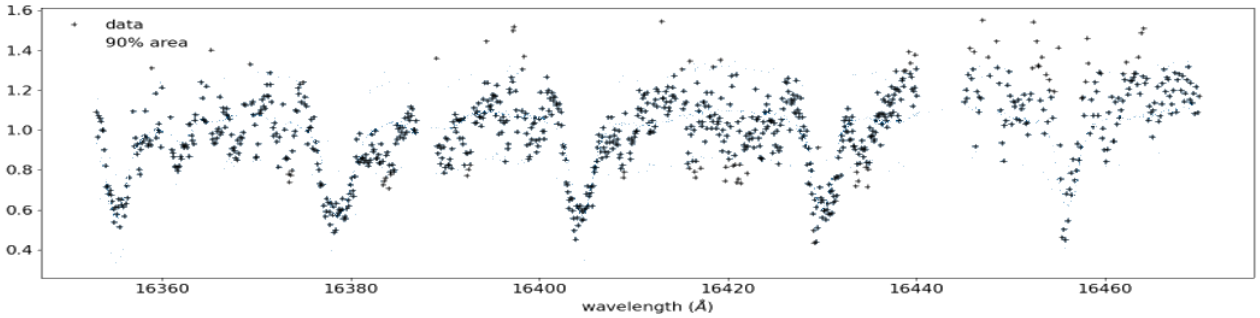
L3


data
90% area
wavelength (Å)

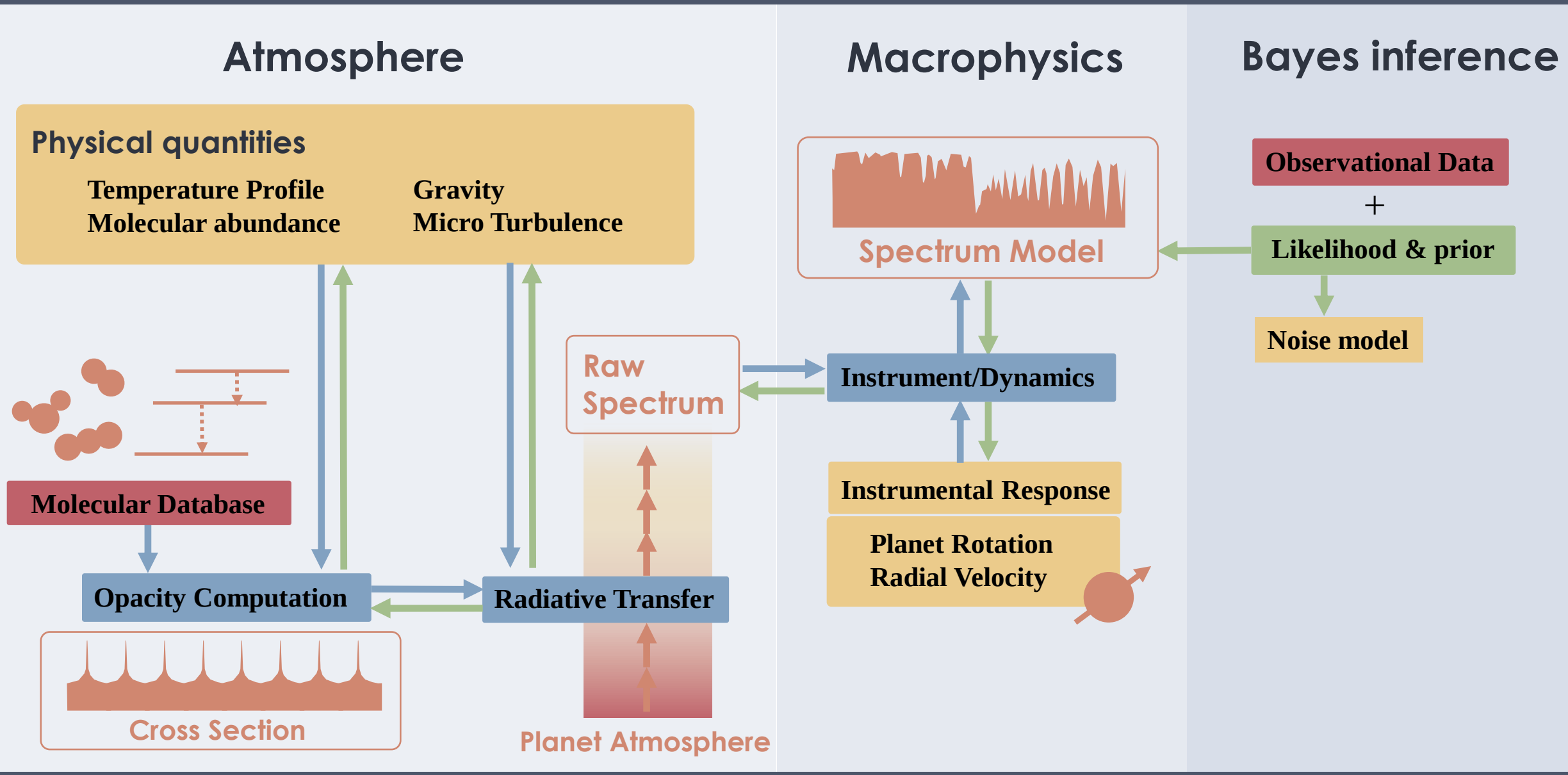
L4


data
90% area
wavelength (Å)

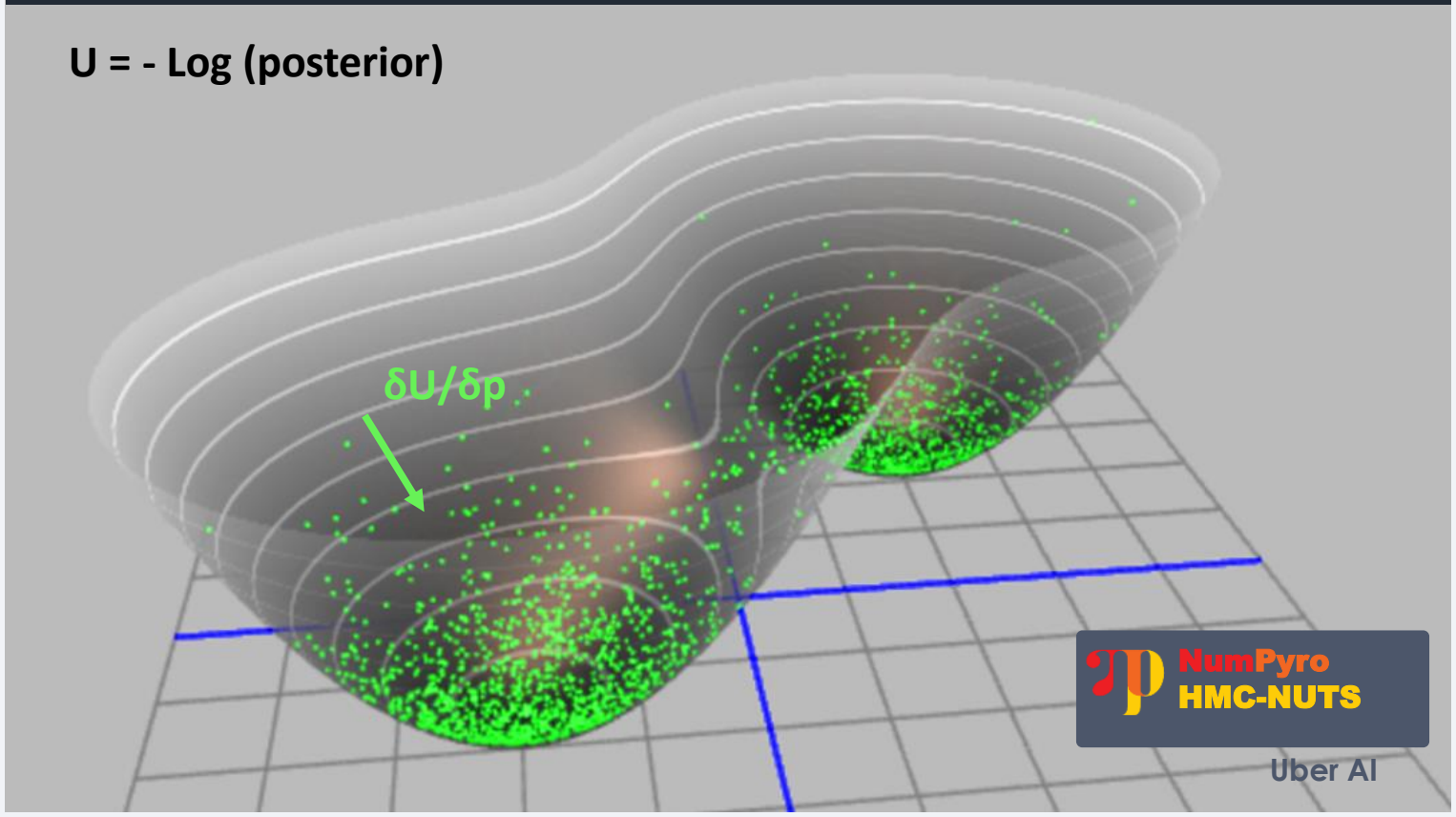
T4


data
90% area
wavelength (Å)

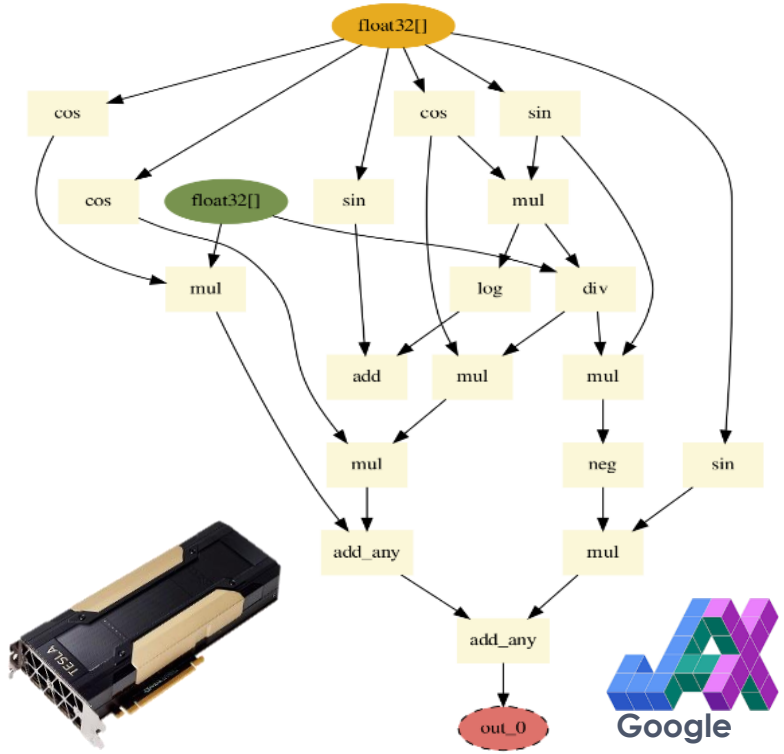
exojax auto-differentiable spectrum model



Gradient-based Efficient MCMC Sampling



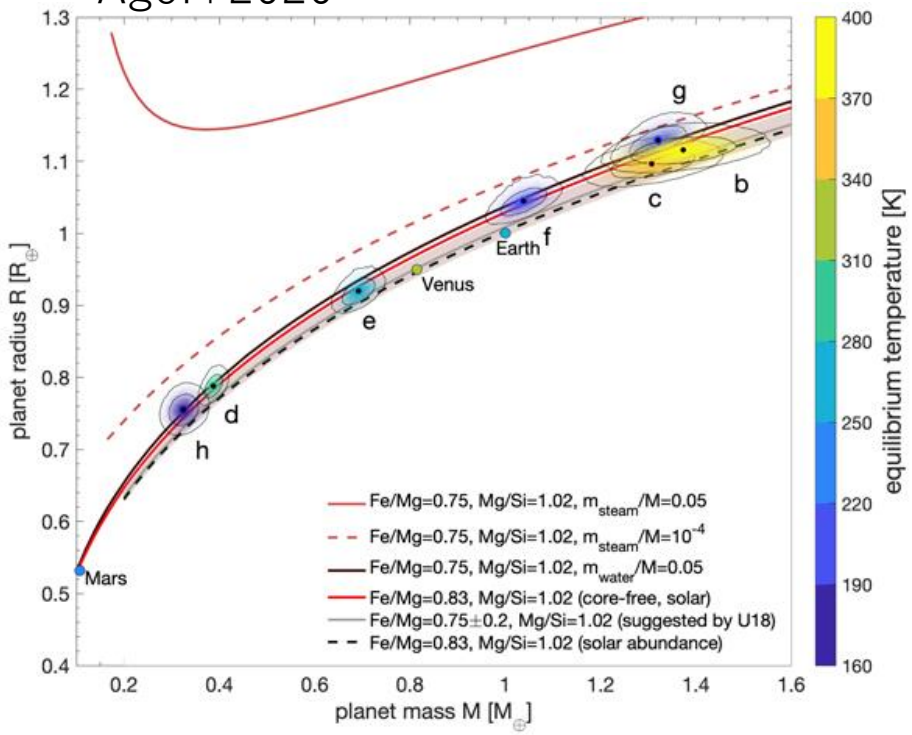
Auto-Gradient + GPU



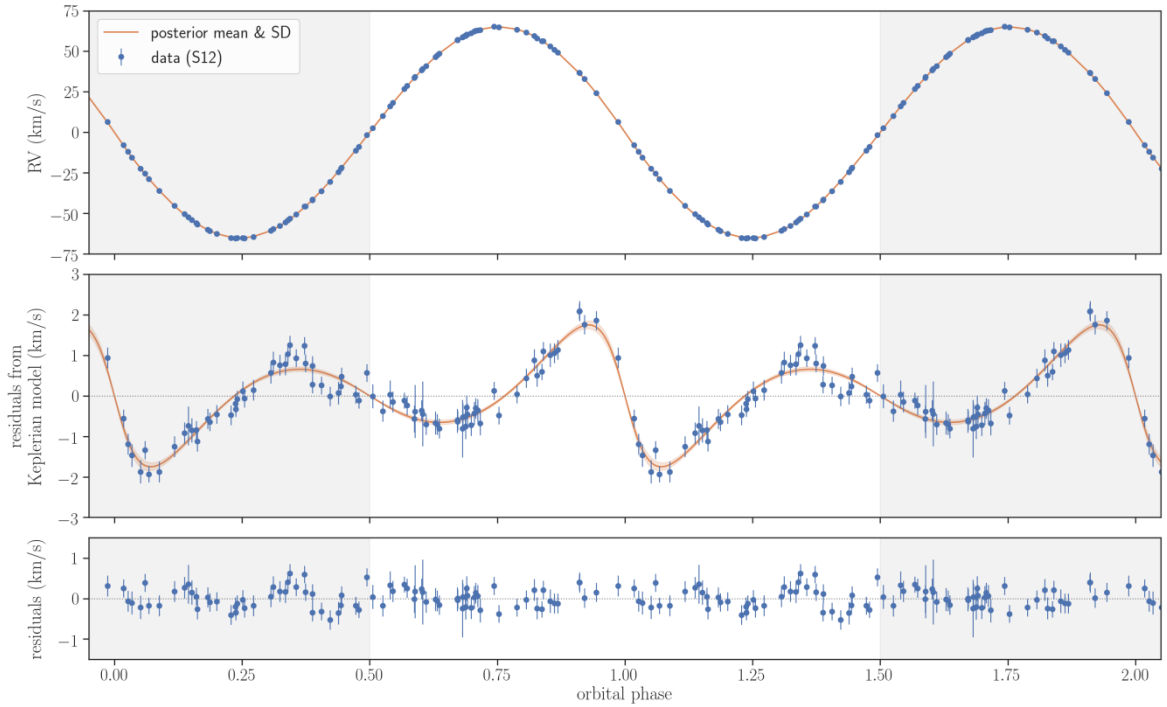
$$\partial_x (\log \cos x \sin x + \sin x)$$

D^5/4 (HMC) D^2 (random MCMC)

Agol+2020

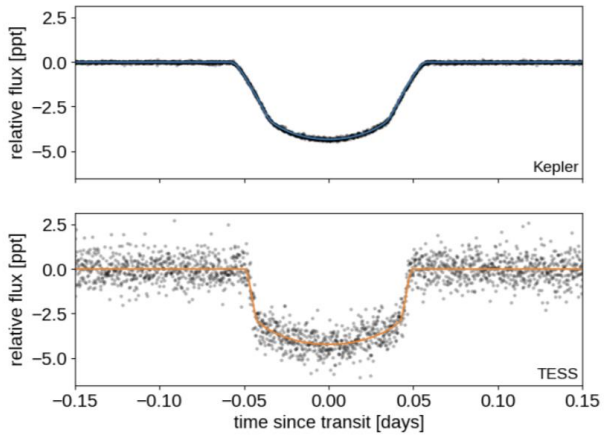


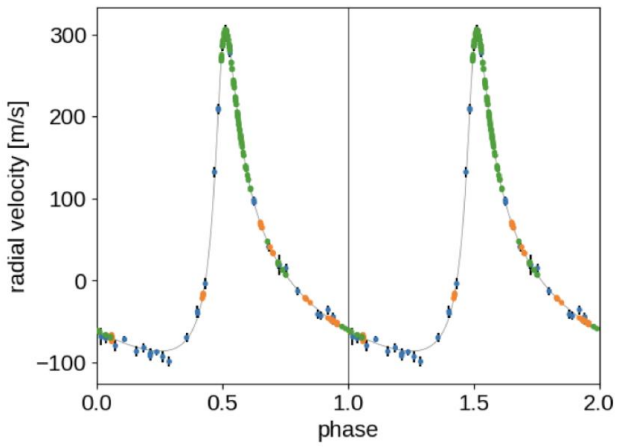
Masuda Hirano 2021



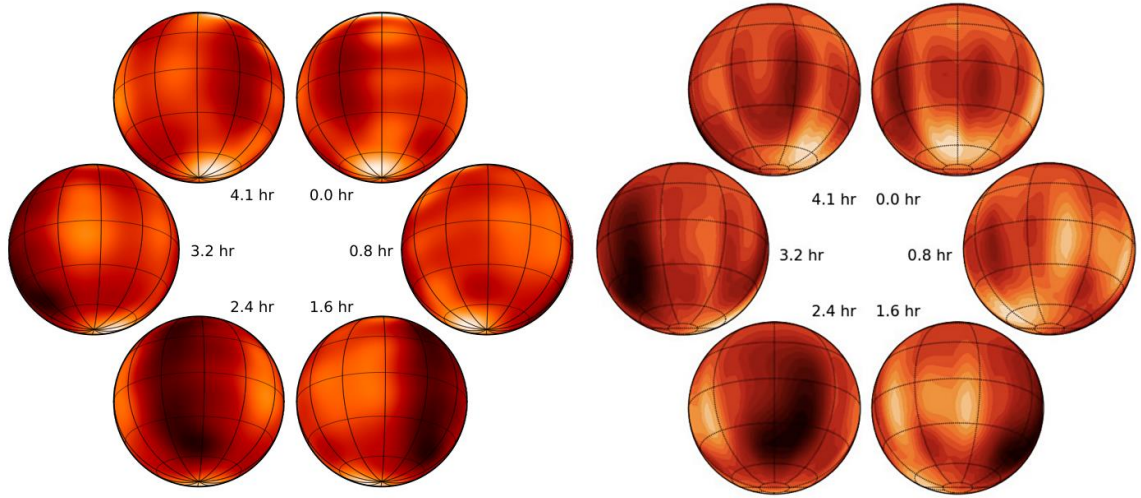
Luger et al. (2021)

DFM+2021





Crossfield et al. (2014)



$$f(x) = |x| + \sin x$$

```
1 import jax.numpy as jnp
```

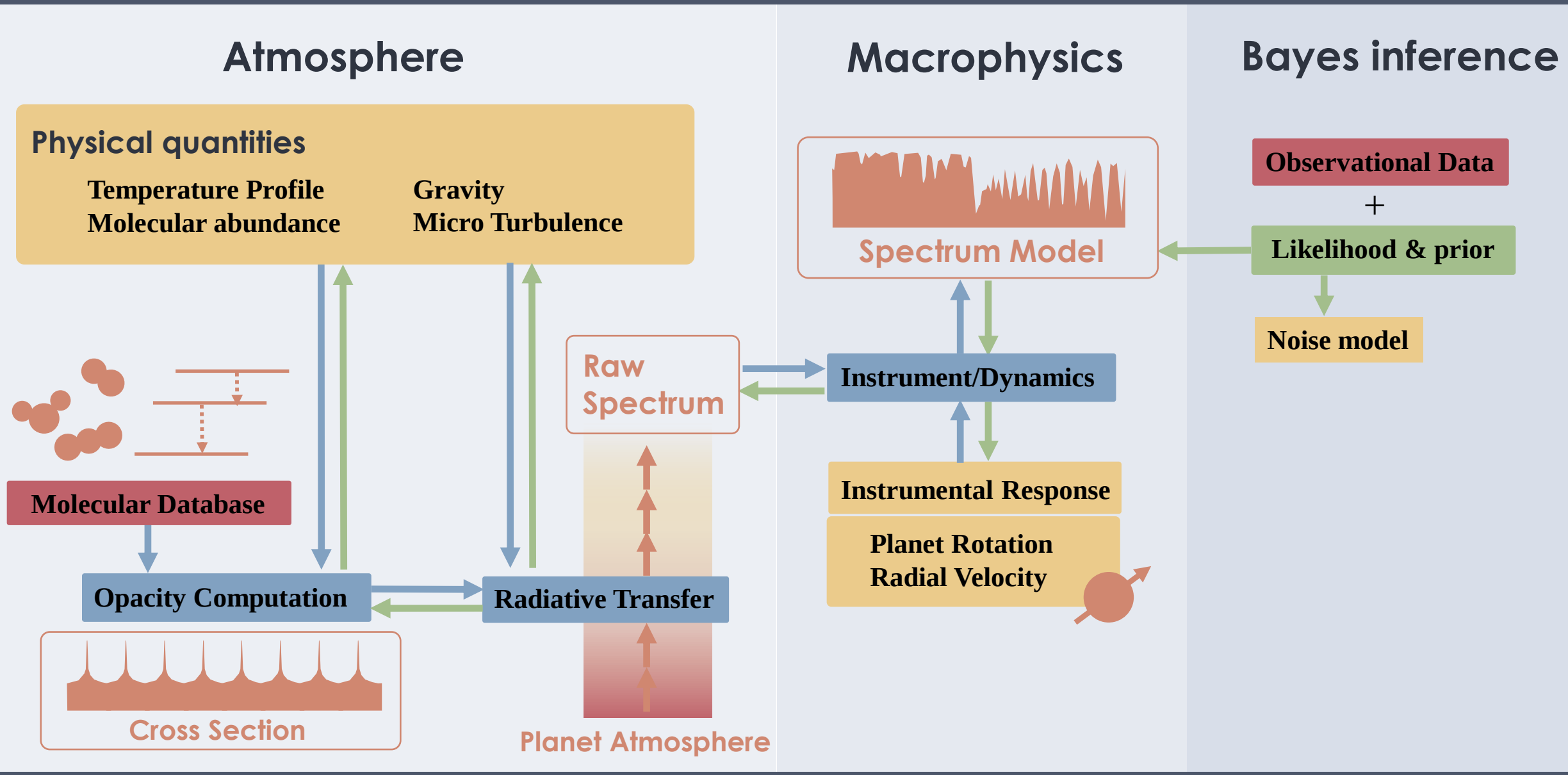
```
1 def f(x):
2     return jnp.abs(x) + jnp.sin(x)
```

```
1 from jax import grad
```

```
1 df=grad(f)
```

```
1 df(1.)    # → DeviceArray(1.5403023, dtype=float32)
```

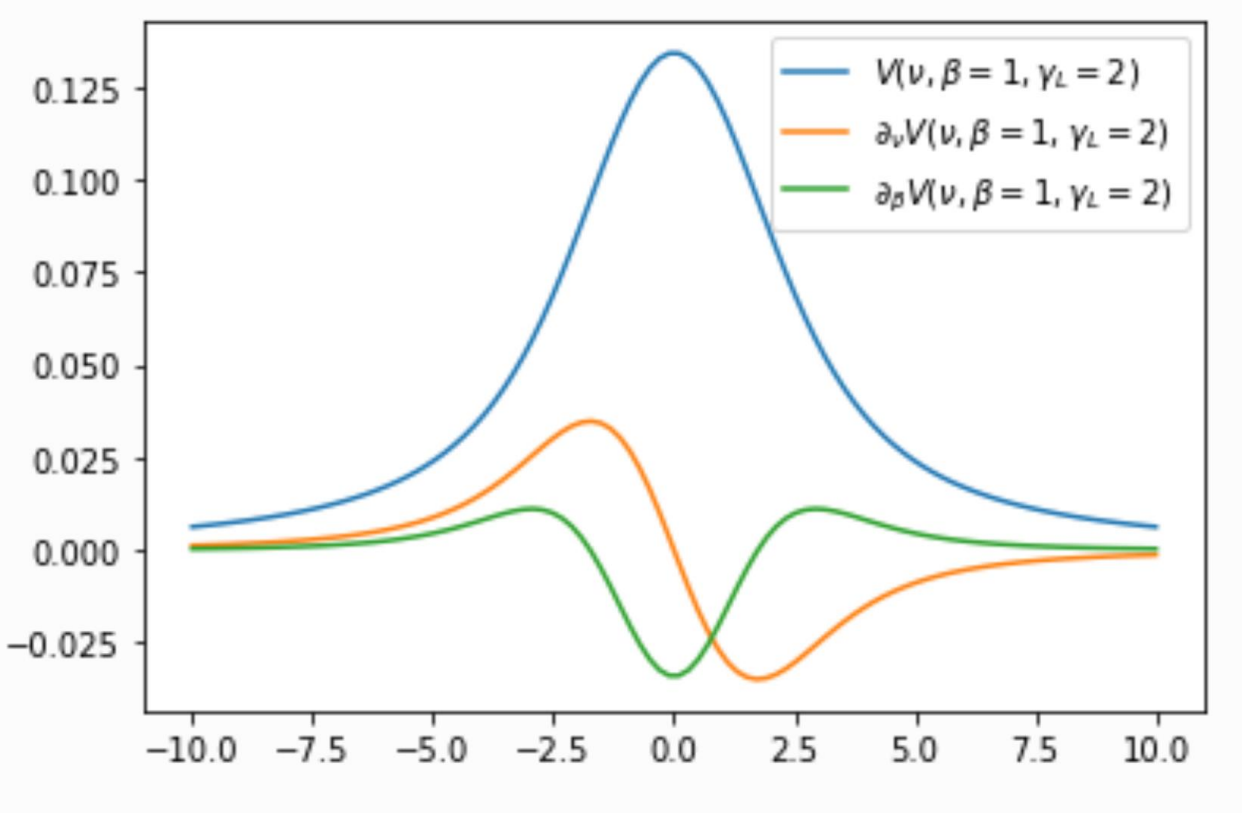
exojax auto-differentiable spectrum model



Differentiable Voigt Profile

```
from exojax.spec import voigtone
from jax import grad, vmap

dvoigt_nu=vmap(grad(voigtone,argnums=0),(0,None,None),0) #derivative by nu
dvoigt_beta=vmap(grad(voigtone,argnums=1),(0,None,None),0) #derivative by beta
```

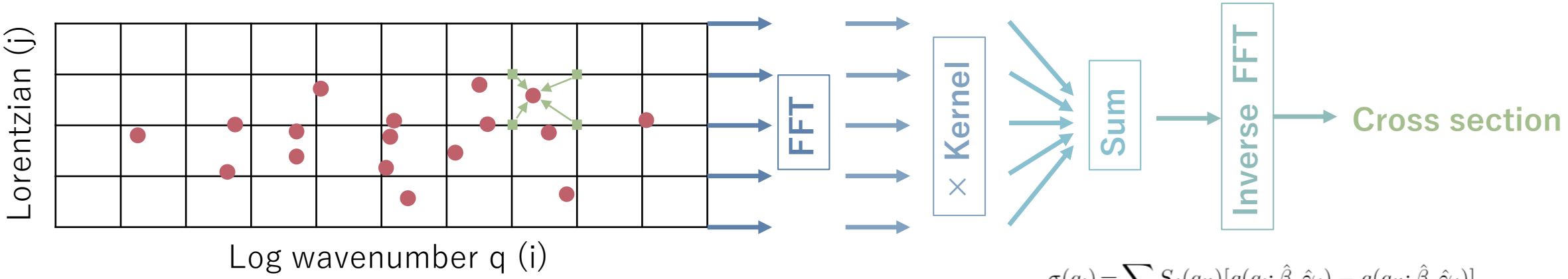


Algorithm 916: Zaghloul & Ali (2011)

Modified Discrete Integral Transform (MODIT)

With van den Bekeroma and Pannier

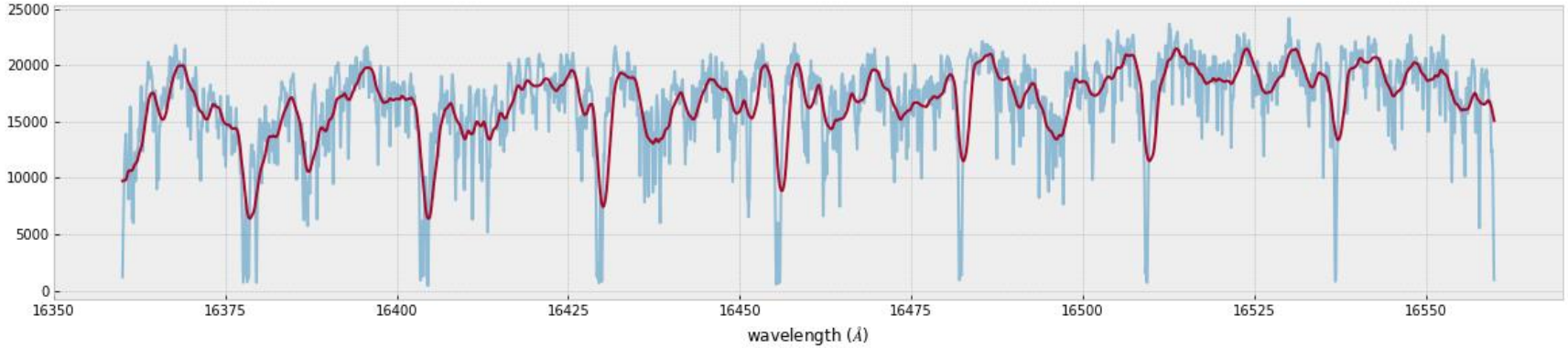
In log wavenumber, a Doppler width is common

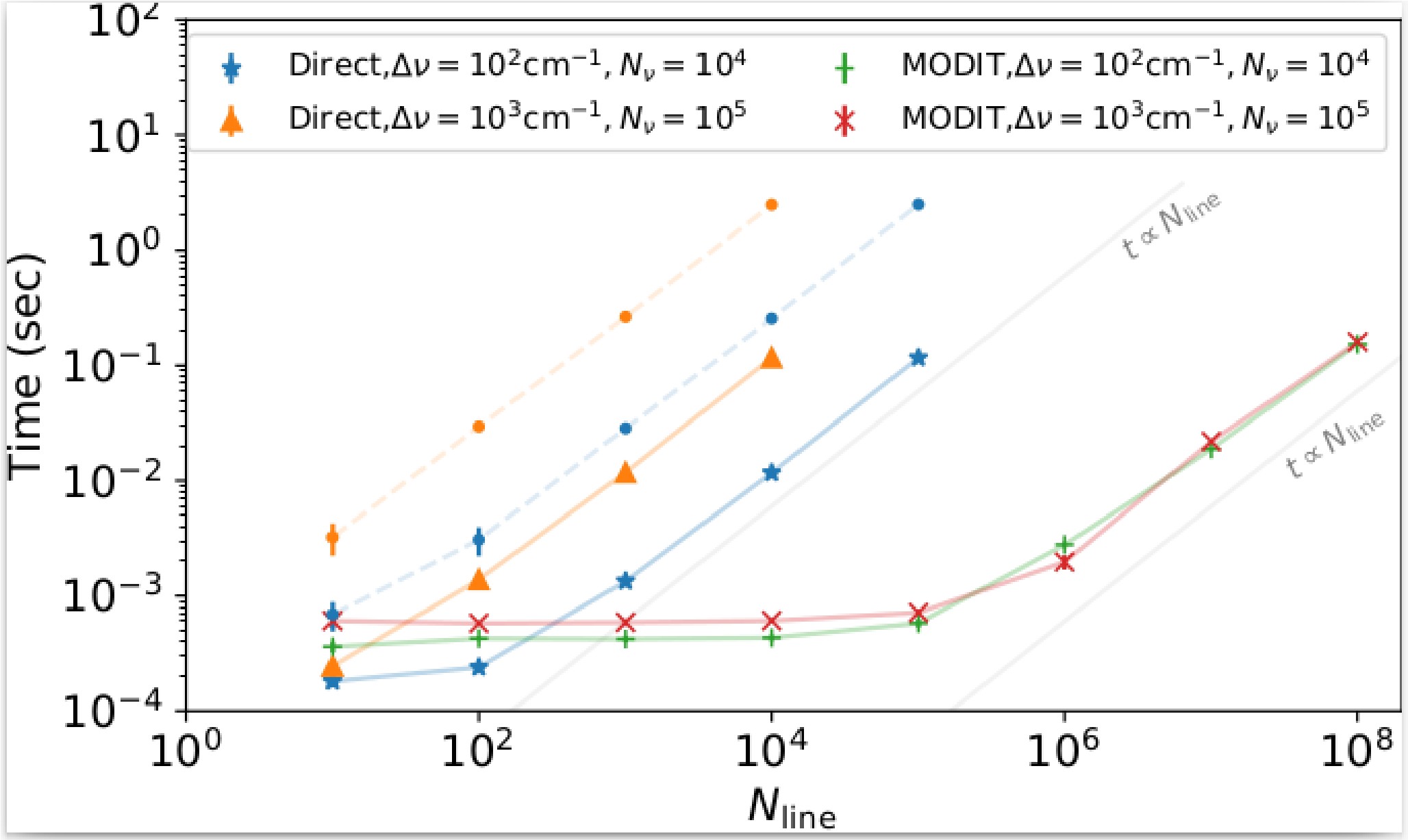


$$\sigma(q_i) = \sum_{i',j} S_j(q_{i'})[g(q_i;\hat{\beta},\hat{\gamma}_j) - g(q_{i'};\hat{\beta},\hat{\gamma}_j)]$$

$$= \mathcal{FT}^{-1}\left[\sum_j \tilde{S}_j(k)\tilde{g}(k;\hat{\beta},\hat{\gamma}_j)\right]$$

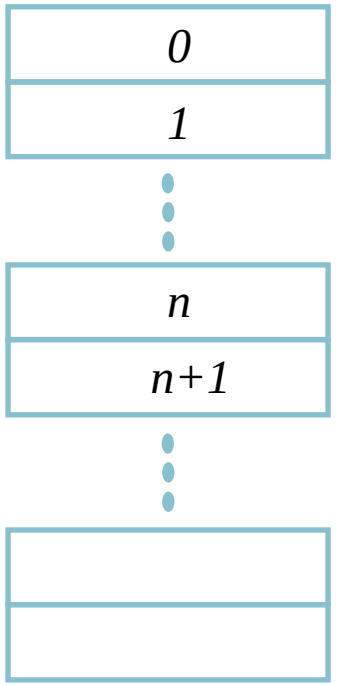
Methane spectrum from scratch (exomol) using my laptop




Direct,$\Delta\nu = 10^2 \mathrm{cm}^{-1}, N_\nu = 10^4$
Direct,$\Delta\nu = 10^3 \mathrm{cm}^{-1}, N_\nu = 10^5$
MODIT,$\Delta\nu = 10^2 \mathrm{cm}^{-1}, N_\nu = 10^4$
MODIT,$\Delta\nu = 10^3 \mathrm{cm}^{-1}, N_\nu = 10^5$
$t \propto N_{\mathrm{line}}$
$t \propto N_{\mathrm{line}}$
Time (sec)
N_{line}
10^2
10^1
10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^0
10^2
10^4
10^6
10^8

Differentiable Radiative Transfer

$$F_0 = \text{sum}(Q * \text{cumprod}(T, \text{axis}=0), \text{axis}=0)$$



$$F_0 = \mathcal{T}_0(\mathcal{T}_1(\mathcal{T}_2(\cdots \mathcal{T}_{N-2}(\mathcal{T}_{N-1}F_B + \mathcal{Q}_{N-1}) + \mathcal{Q}_{N-2}) + \cdots + \mathcal{Q}_2) + \mathcal{Q}_1) + \mathcal{Q}_0 = \boldsymbol{\phi} \cdot \boldsymbol{q}$$

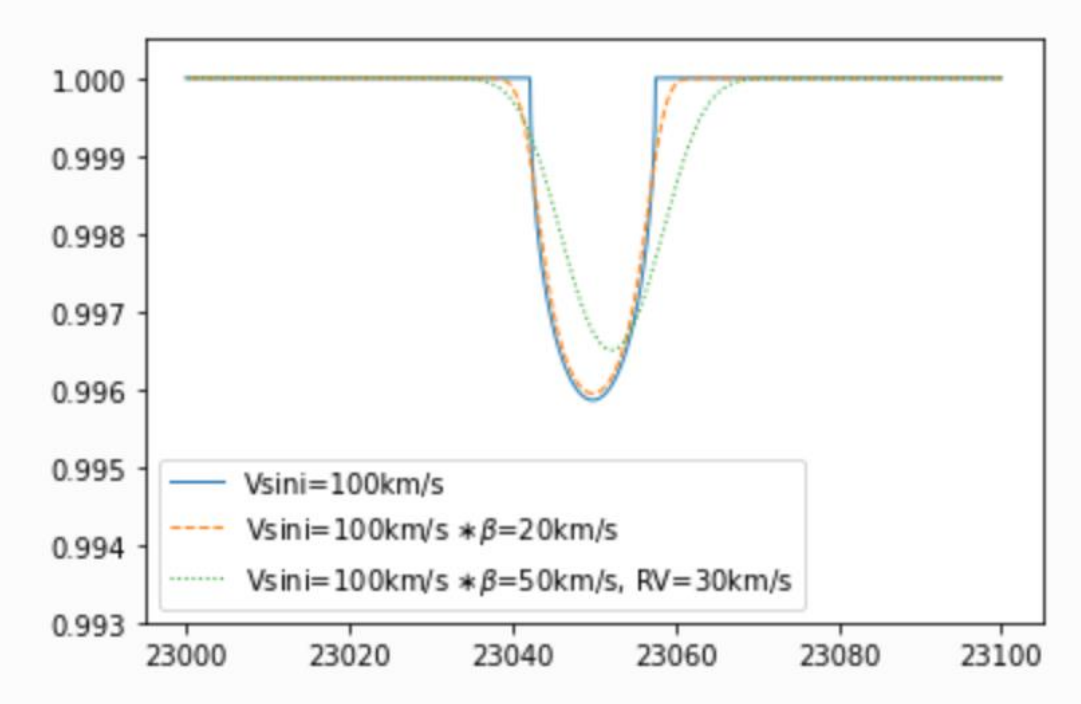
No scattering

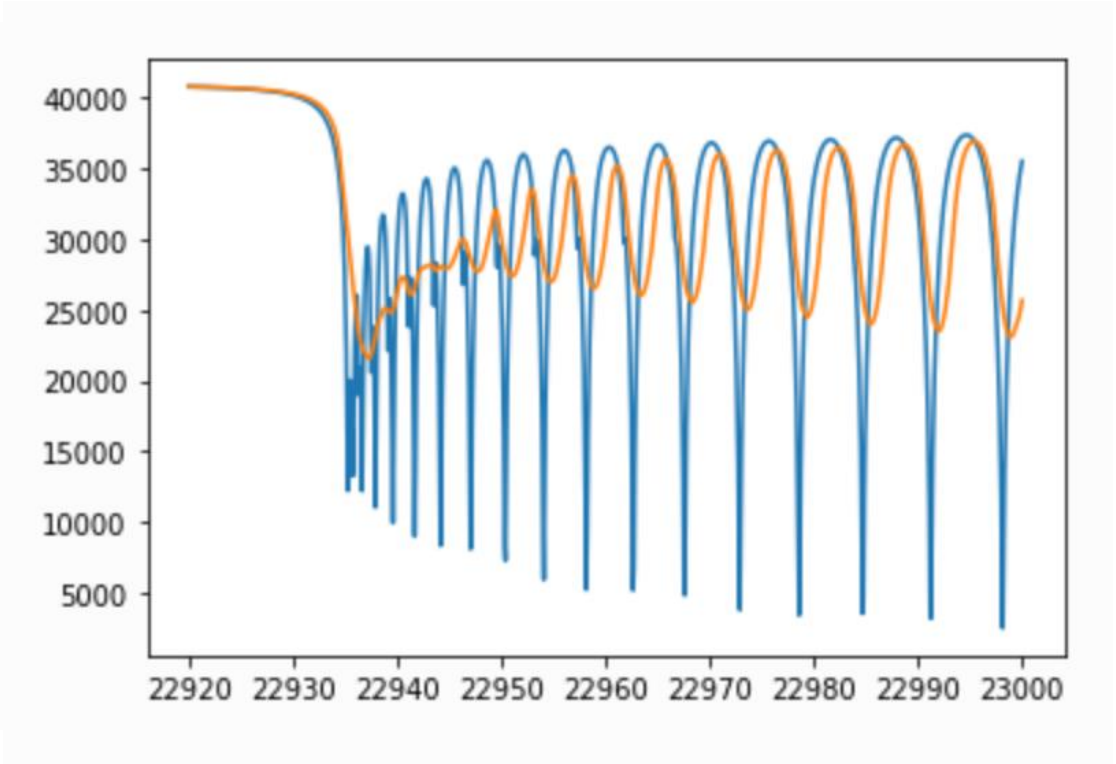
$$F_n = \mathcal{T}_n F_{n+1} + \overbrace{(1 - \mathcal{T}_n)\,\mathcal{S}_n}^{\mathcal{Q}_n},$$

$$\begin{aligned} \boldsymbol{\phi} &= (1, \mathcal{T}_0, \mathcal{T}_0\mathcal{T}_1, \cdots, \mathcal{T}_0\mathcal{T}_1 \cdots \mathcal{T}_{N-2}, \mathcal{T}_0\mathcal{T}_1 \cdots \mathcal{T}_{N-1})^\top \\ \boldsymbol{q} &\equiv (\mathcal{Q}_0, \mathcal{Q}_1, \cdots, \mathcal{Q}_N)^\top \end{aligned}$$

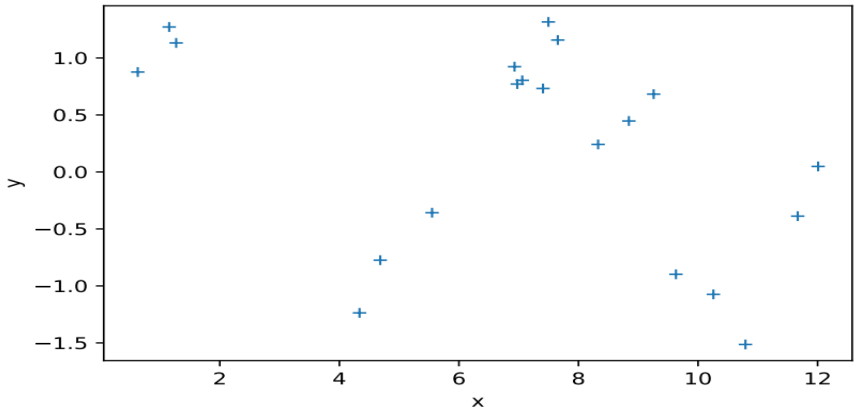
$$\begin{aligned} \mathcal{T}_n &= 2E_3(\Delta\tau_n) \\ &= (1 - \Delta\tau_n)\exp(-\Delta\tau_n) + (\Delta\tau_n)^2 E_1(\Delta\tau_n) \end{aligned}$$

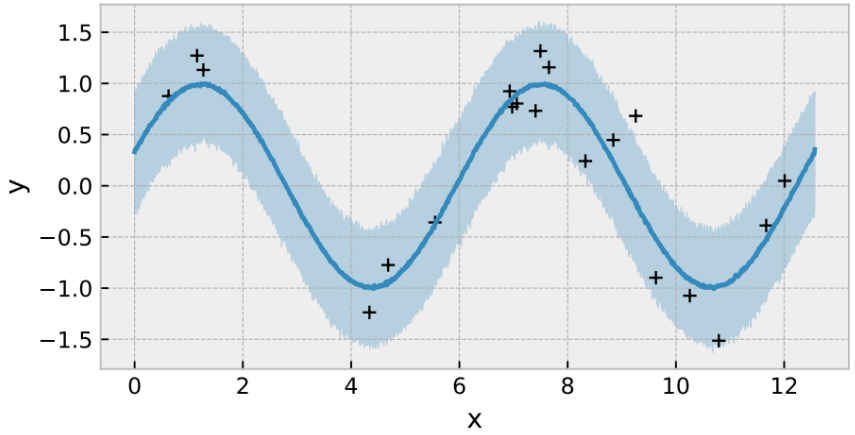
Auto-differentiable Rotation Kernel and Instrumental Profile





簡単NumPyro






```python
import jax.numpy as jnp
import numpyro
import numpyro.distributions as dist

def model(x,y):
    phase = numpyro.sample('phase', dist.Uniform(-1.0*jnp.pi, 1.0*jnp.pi))
    sigma = numpyro.sample('sigma', dist.Exponential(1.))
    mu=jnp.sin(x+phase)
    numpyro.sample('y', dist.Normal(mu, sigma), obs=y)
```

この例からわかるように、確率分布は dist で指定されたものを、numpyro の sample 内で定義することにより確率変数として扱われる。最後のデータ y の部分は式 (4.1,4.2) が

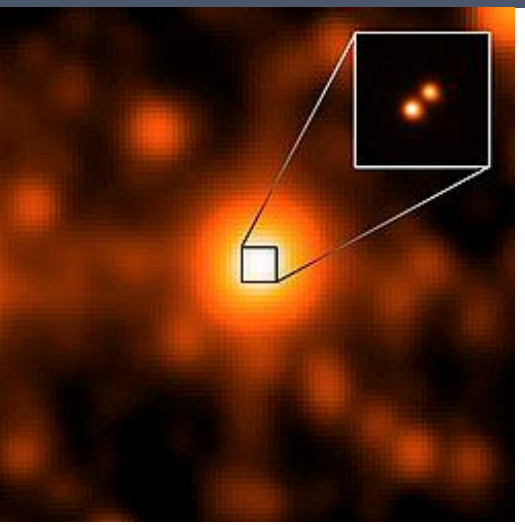
$$y \sim \mathcal{N}(\sin(x+\phi), \sigma) \tag{4.5}$$

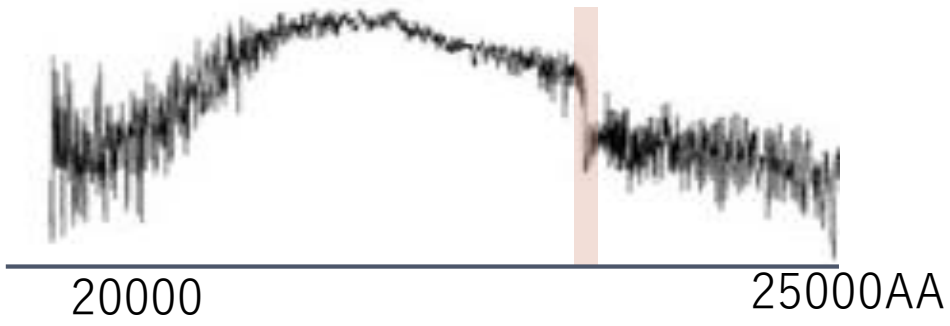
と書き直されることよりわかる。さてこれを numpyro の HMC-NUTS にかける。

```python
from jax import random
from numpyro.infer import MCMC, NUTS
rng_key = random.PRNGKey(0)
rng_key, rng_key_ = random.split(rng_key)
num_warmup, num_samples = 1000, 2000
```

```python
kernel = NUTS(model)
mcmc = MCMC(kernel, num_warmup=num_warmup, num_samples=num_samples)
mcmc.run(rng_key_, x=x, y=y)
mcmc.print_summary()
```

Demonstration using Luhman 16A as observed by CRIRES



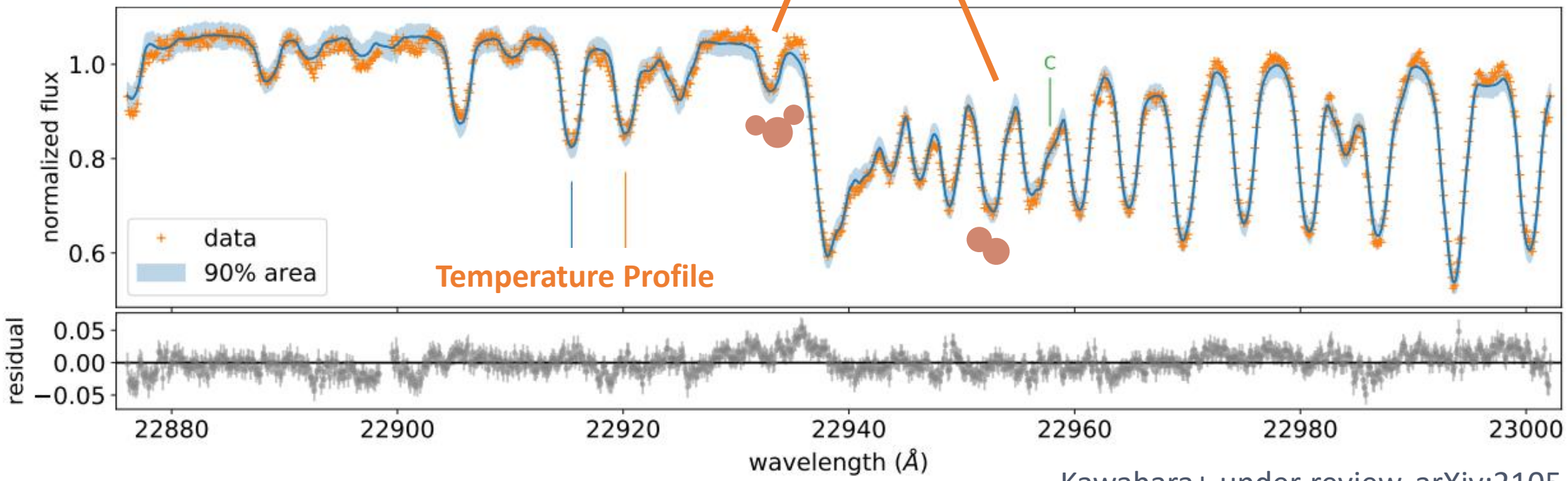


CO+H2O+CIA

Rotation

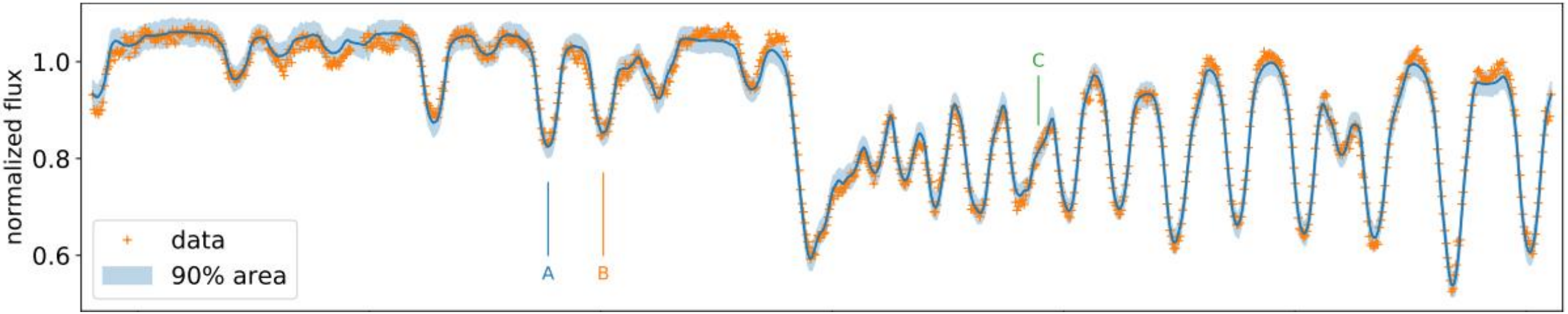
Carbon/Oxygen Ratio

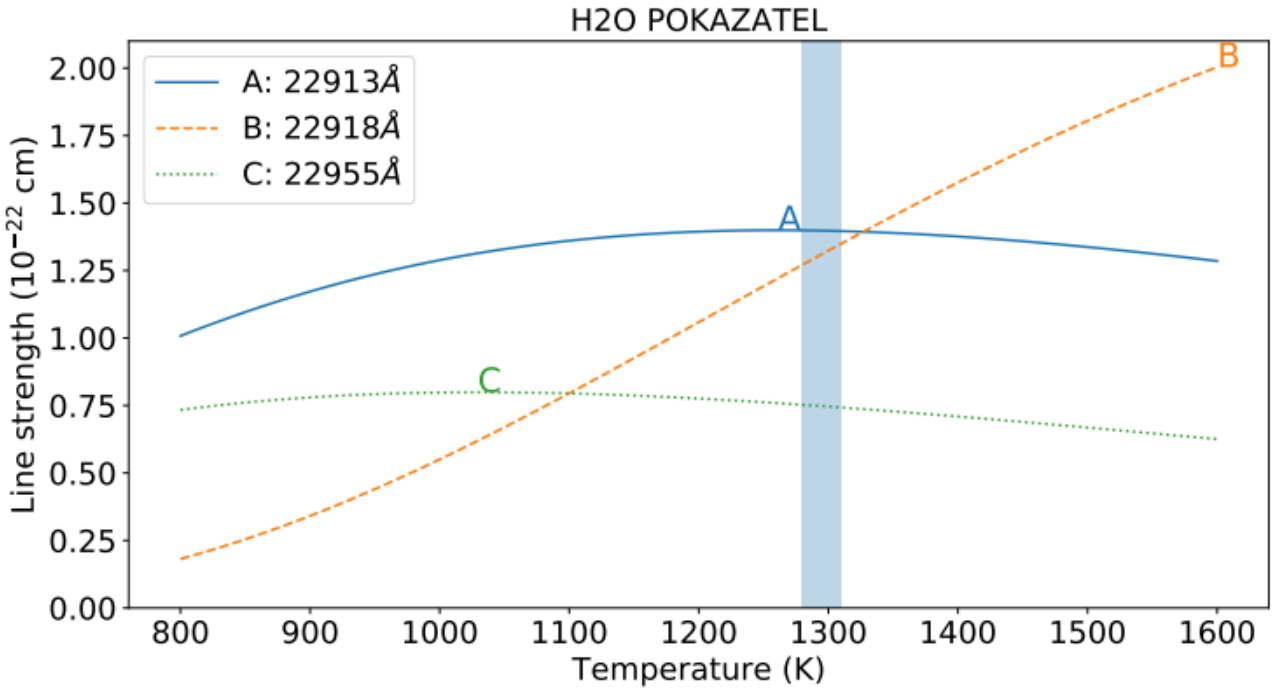
CRIRES data (Crossfield+2014)

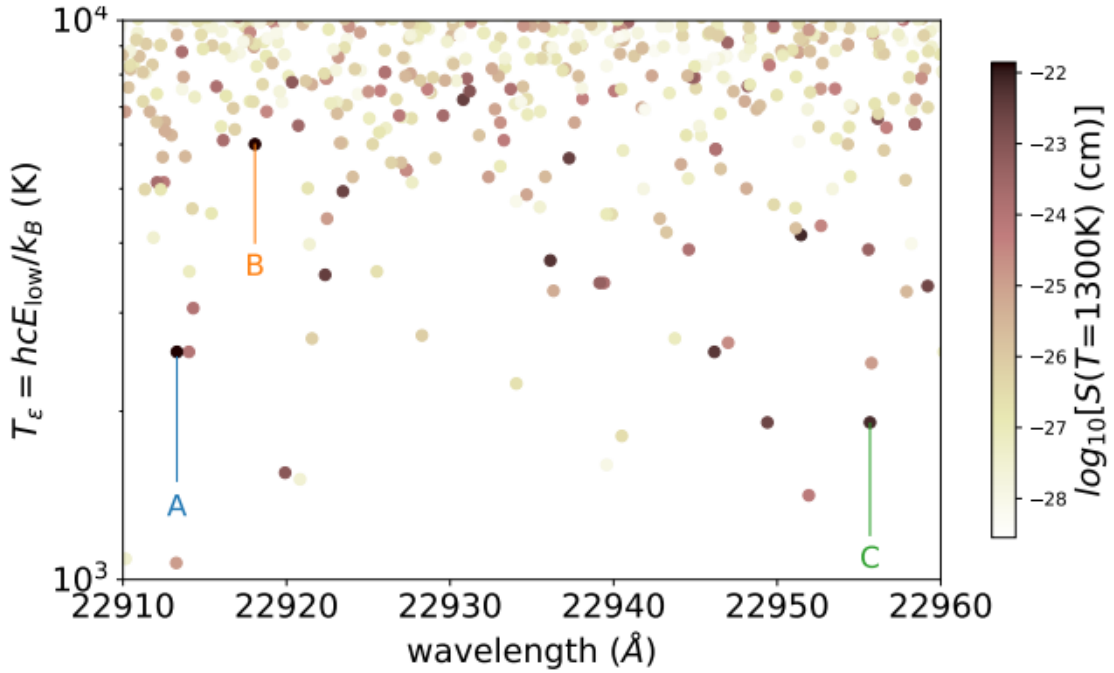


Kawahara+ under review, arXiv:2105.14782

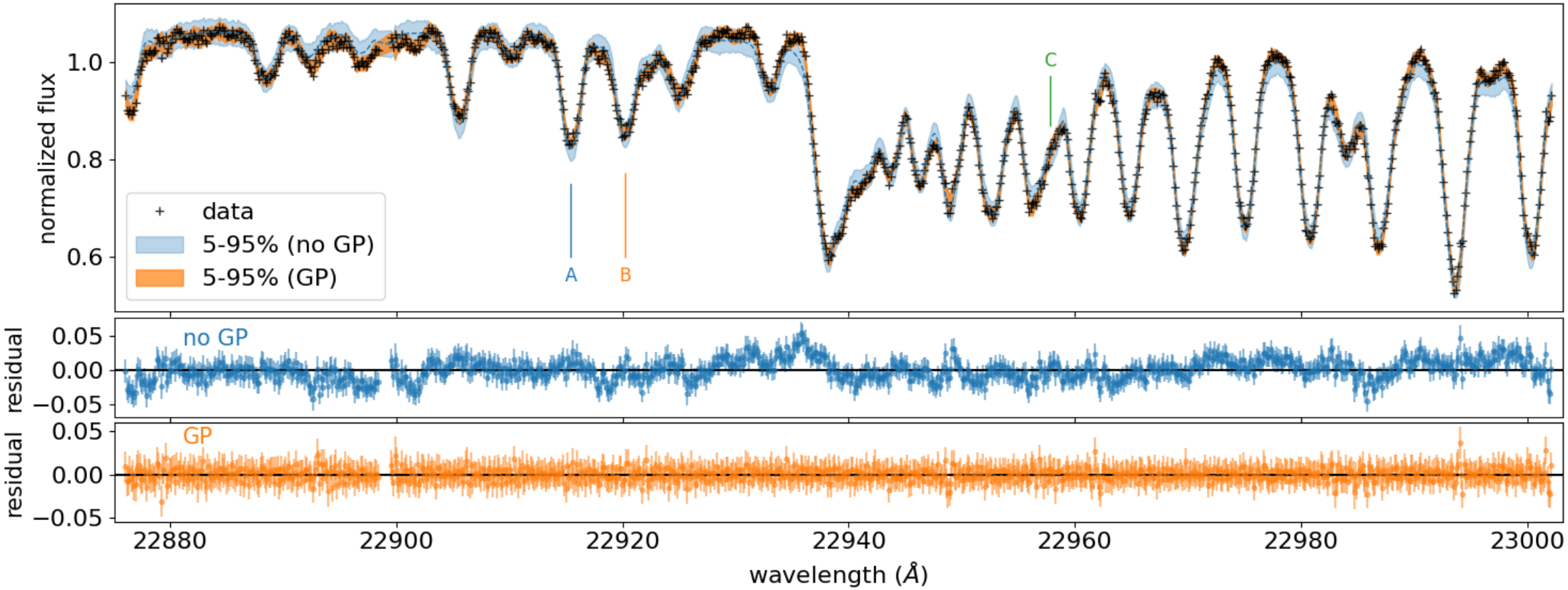
Line diagnostics of Temperature







GPもノイズモデルに簡単に入ります




```
numpyro.sample(tag, dist.MultivariateNormal(loc=mu, covariance_matrix=cov), obs=y)
```

GPも温度圧力モデルに簡単に入ります

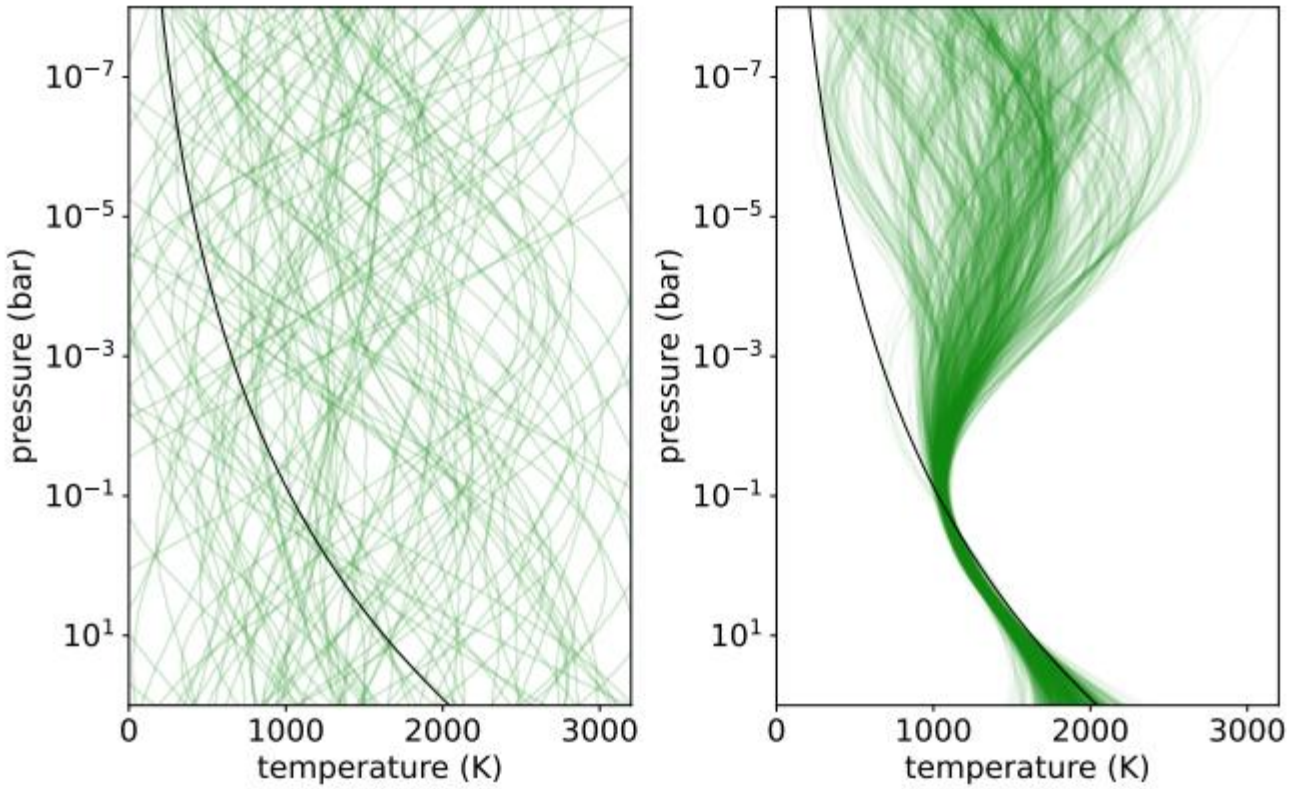
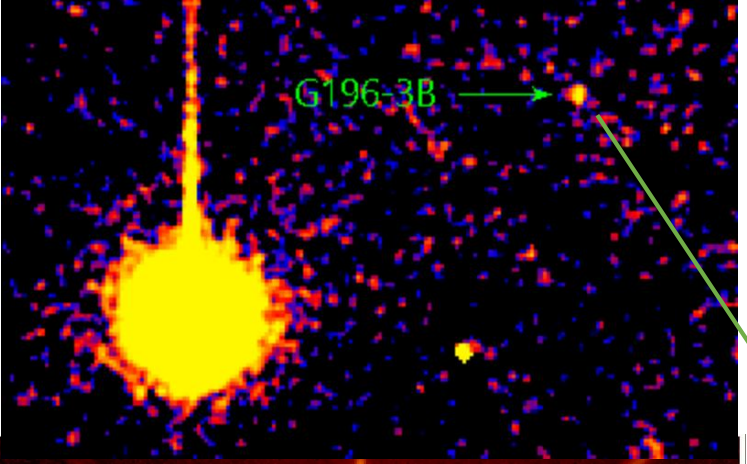
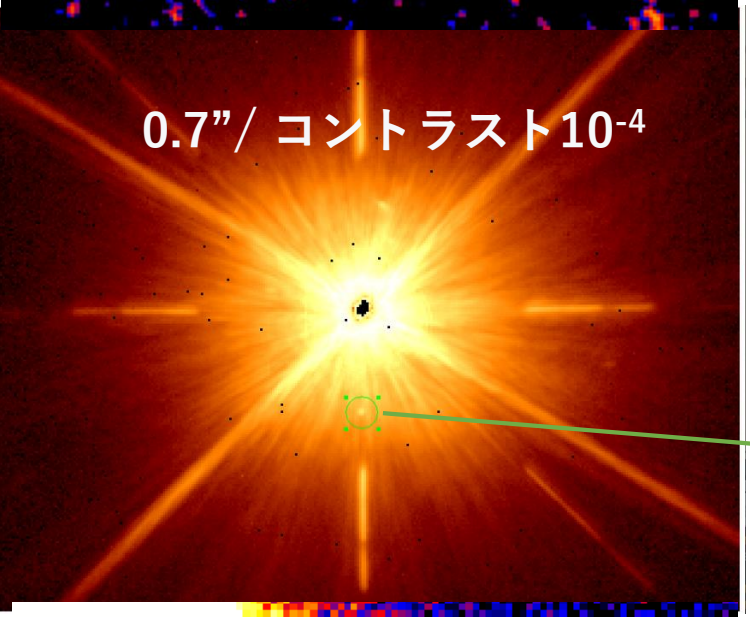
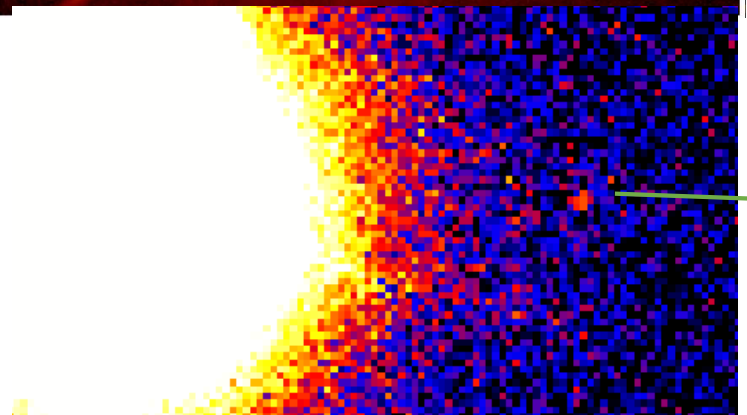


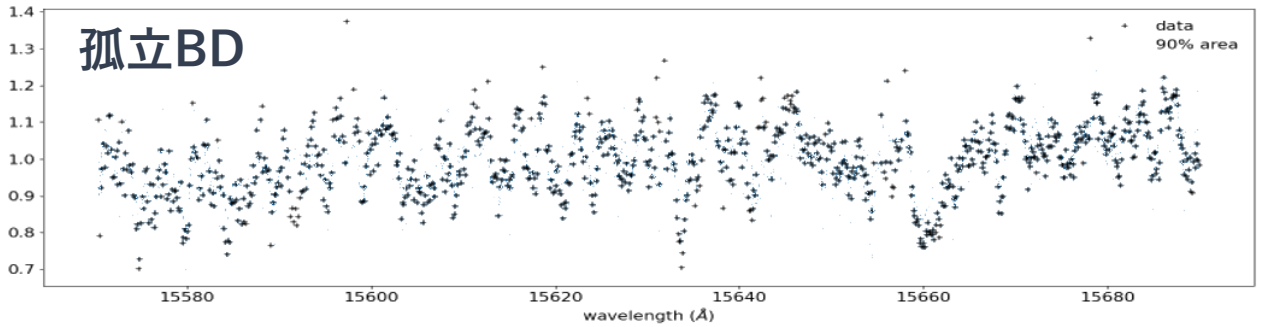
Figure 11. Prior (left) and posterior (right) sampling of the layer-by-layer temeprature model (green lines). The solid black lines indicate the best-fit power law model in Section 5.


G196-3B

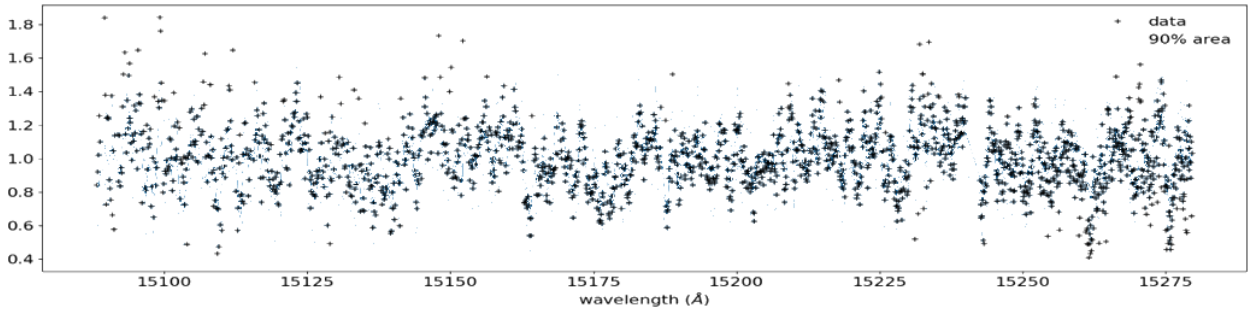

0.7”/ コントラスト10-4



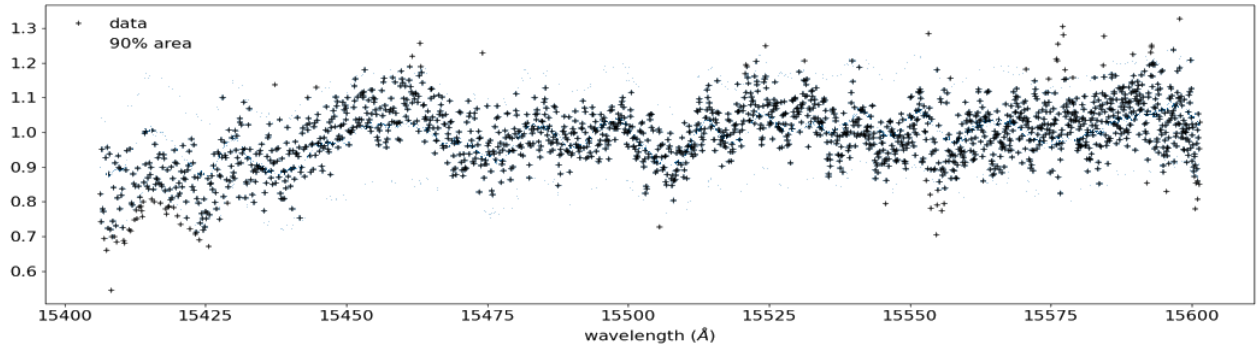
L1


孤立BD
data
90% area
wavelength (Å)

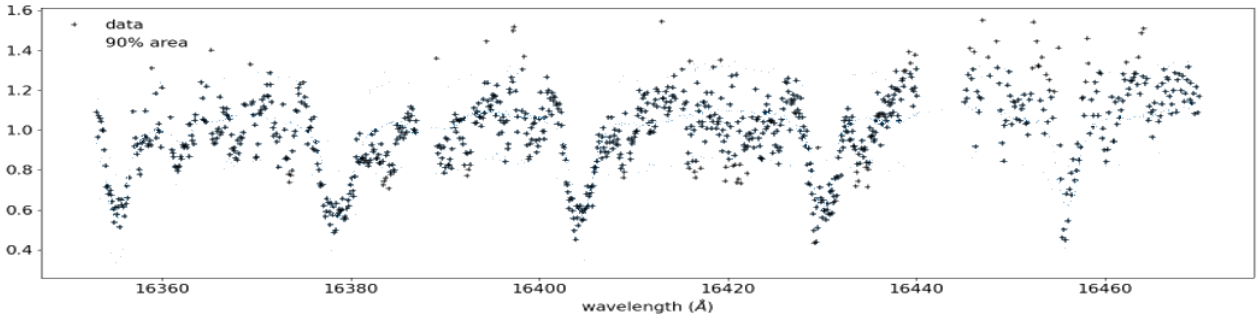
L3

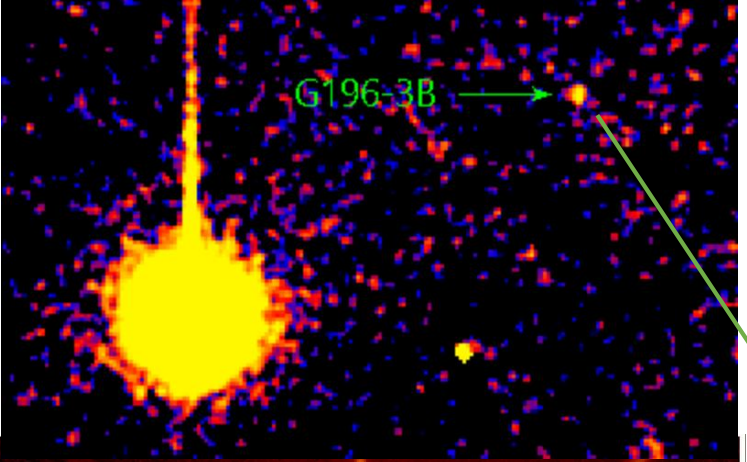

data
90% area
wavelength (Å)

L4

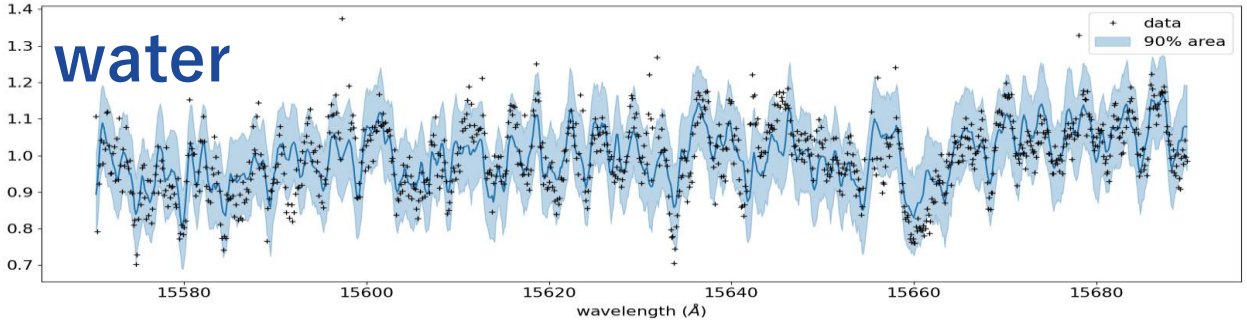

data
90% area
wavelength (Å)

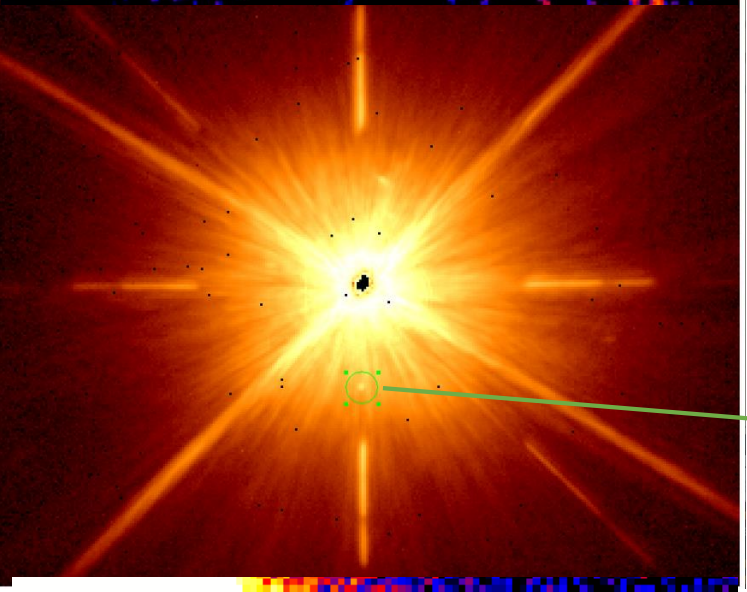
T4


data
90% area
wavelength (Å)

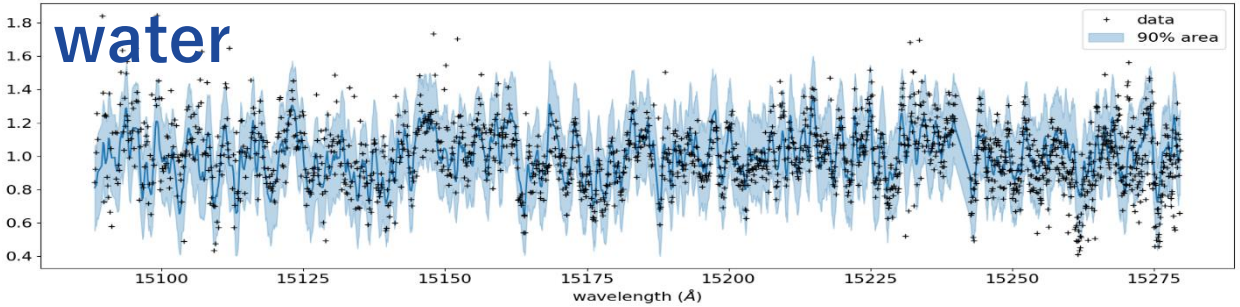


L1

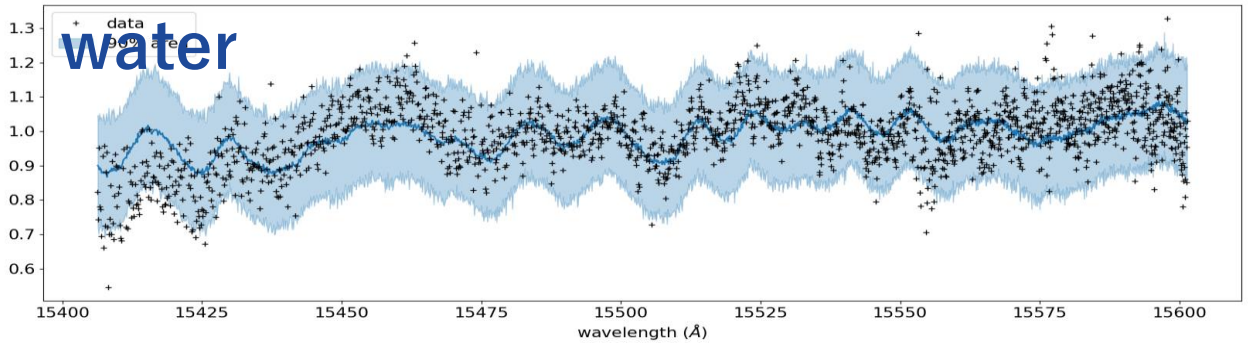


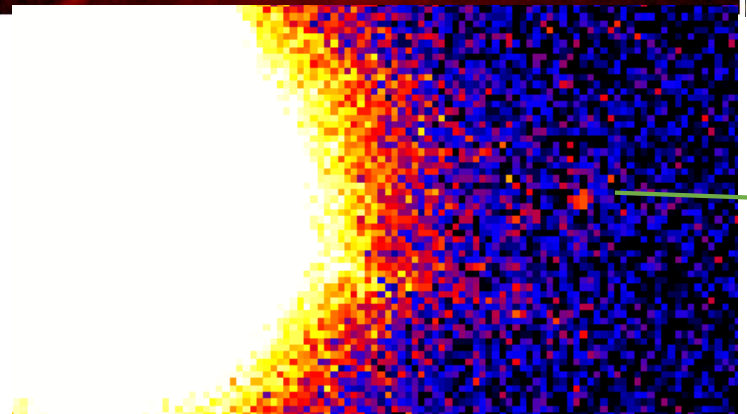


L3

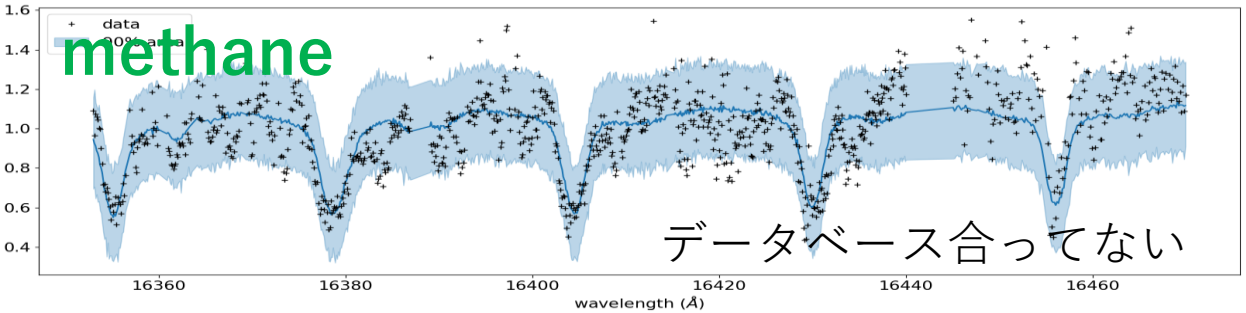


L4





T4



データベース合ってない

Gravity from line broadening

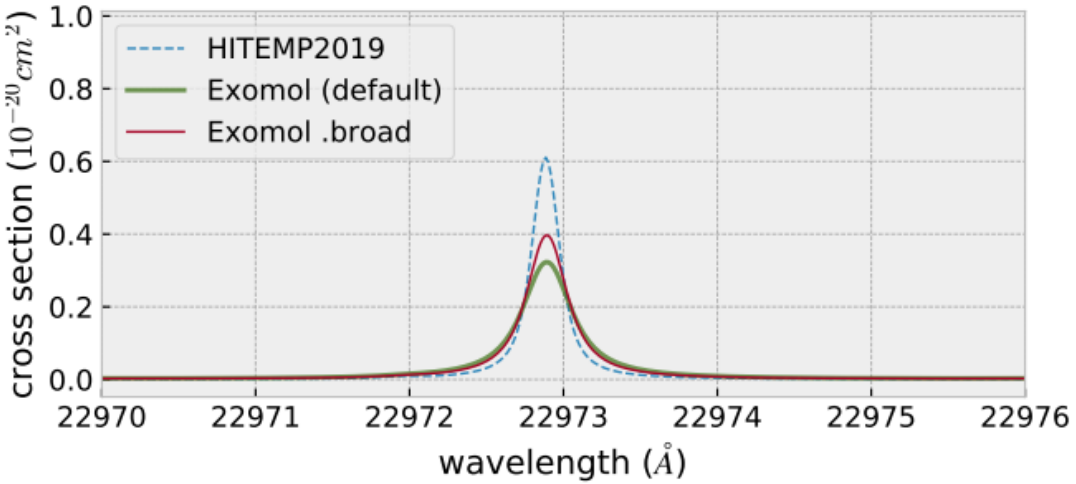


Figure 9. Comparison of the CO cross section between HITEMP2019 and ExoMol/Li2015 (default: constant default values of $\alpha_{\rm ref} = 0.07$ and $n_{\rm texp} = 0.5$ in the definition file, .broad: broadening parameters in the ".broad" file). We assume $T = 1300$ K, $P = 1$ bar ($P_{\rm air} = 0.99$ bar and $P_{\rm self} = 0.1$ bar for HITEMP2019).

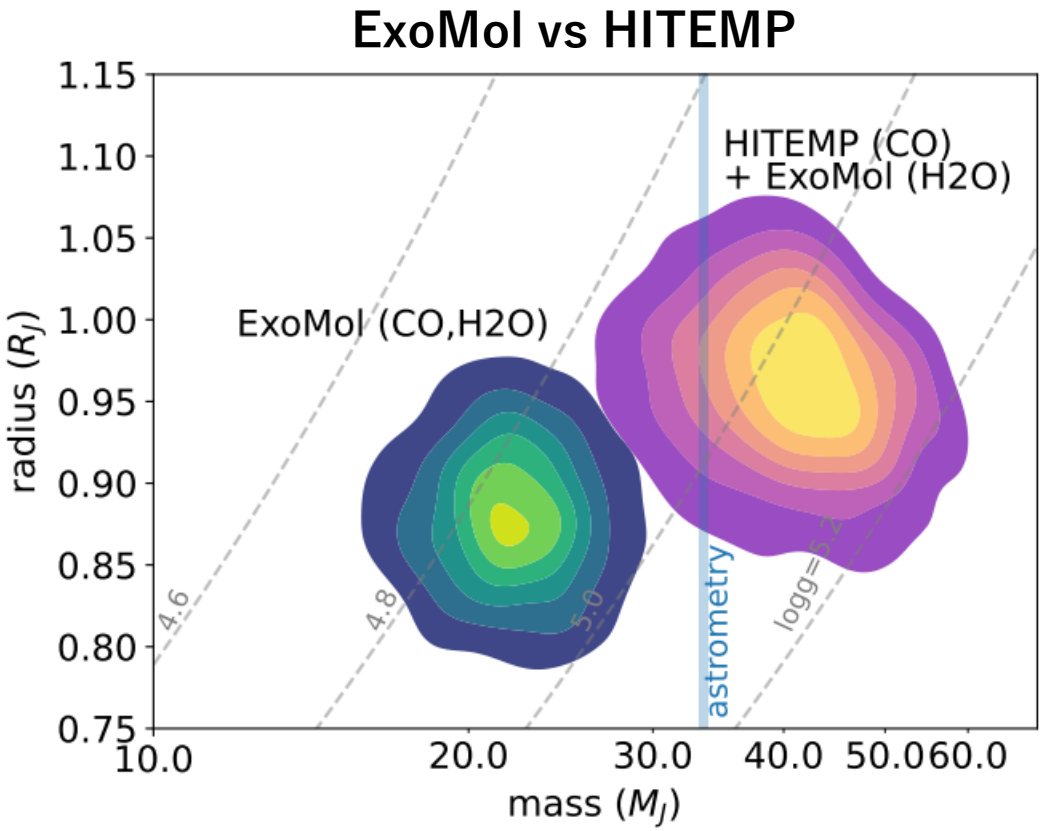
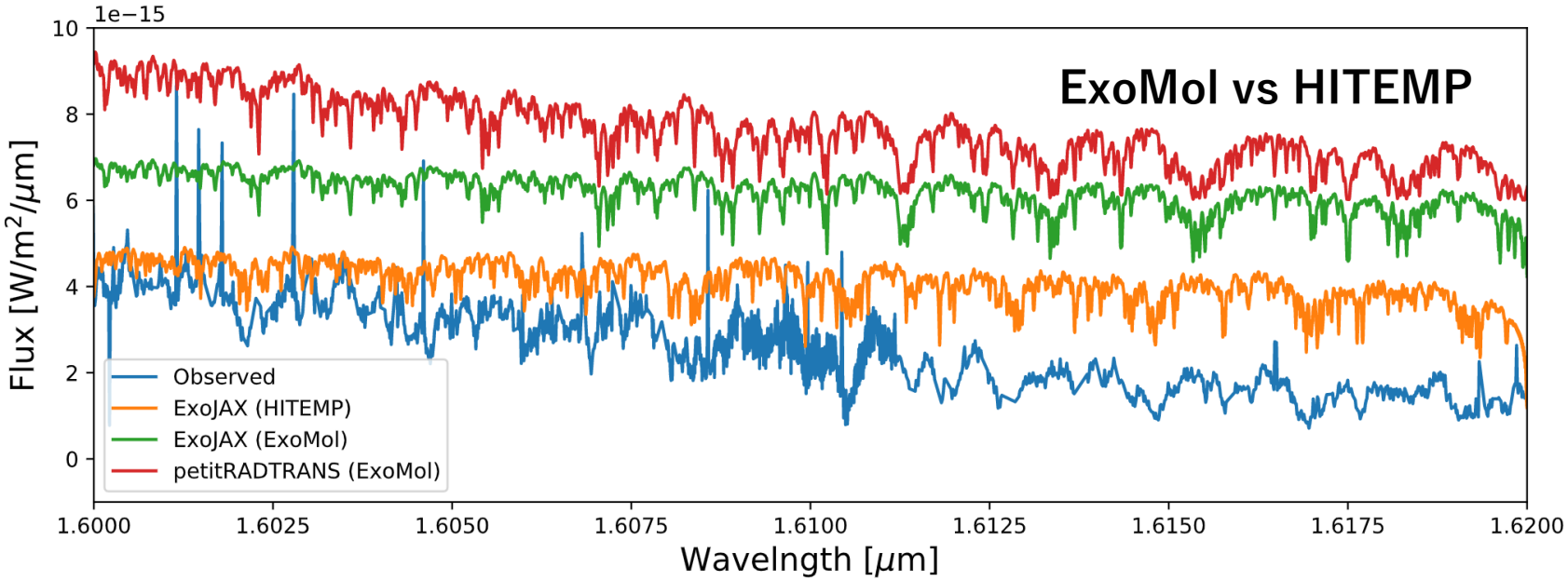


Figure 10. The mass and radius inferred using two different models for CO (ExoMol using the ".broad" file and HITEMP2019). The astrometric mass is shown in the blue shaded region (Lazorenko & Sahlmann 2018). The dashed lines show the iso–gravity lines.

○既存データベースの問題点

ExoMol　水素大気　but 実験での検証が不完全（とくに圧力幅）
HITEMP　地球大気

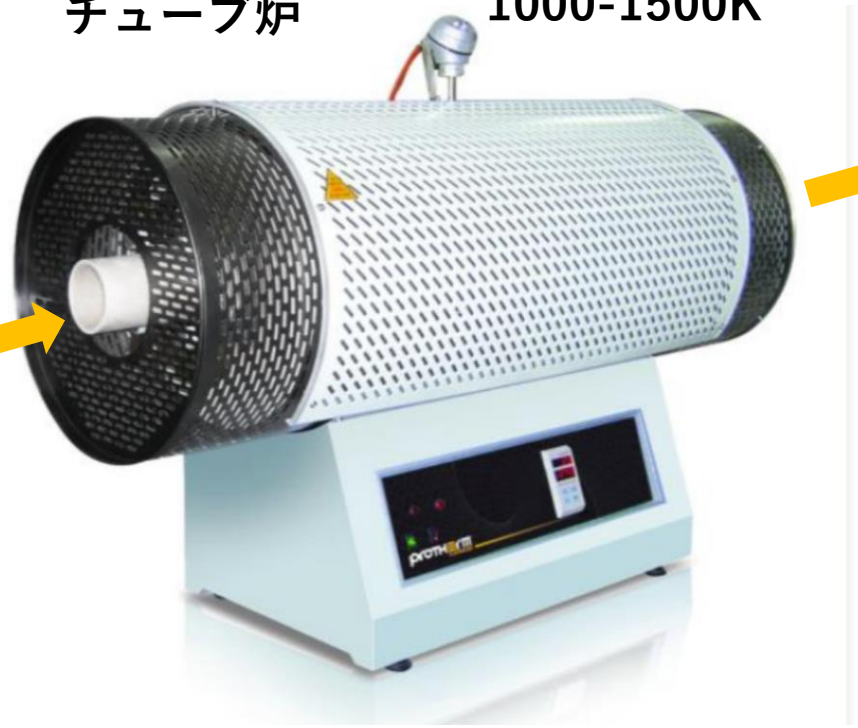


川島氏提供

○考えるのがめんどくさいので自分たちで調べたい

ガラス石英管

チューブ炉　**1000-1500K**



光

分光器

CO　(1%) + 水素
CH4 + 水素
etc を封入　→　分光したい

http://heattec.co.jp/roshinkan_sekiei.html

http://www.ogawaseiki.jpn.org/OSK481TVK205-645

写真はイメージです

○まとめ

分子データベースから計算できベイズ推定が可能な
系外惑星・褐色矮星スペクトルモデルをつくった

実際、観測スペクトルを良くフィットできる

基礎データがいまいちなところがあるので、惑星大気を模擬したガスをいれて分光実験を企画している（共同研究募集中）