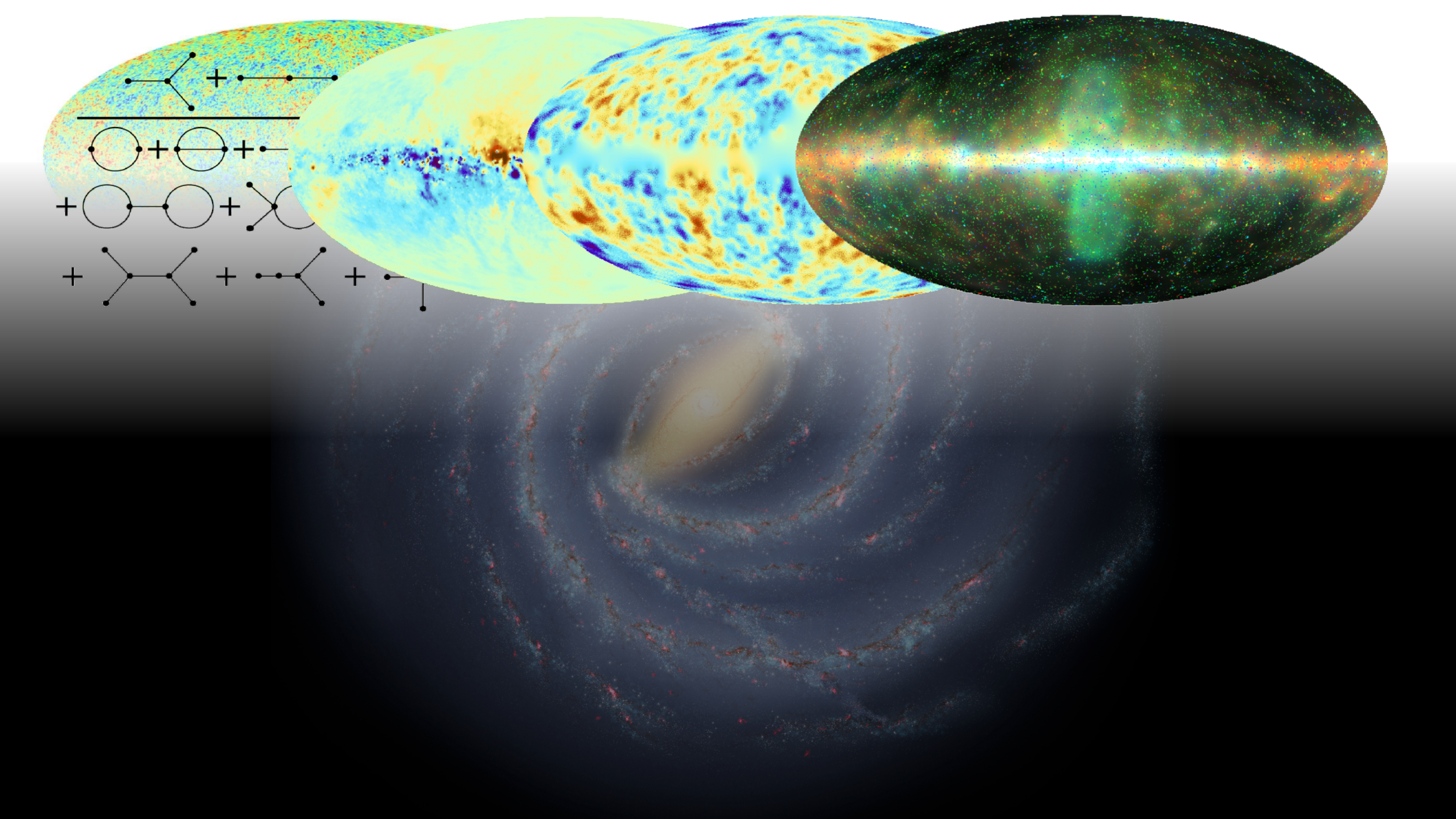
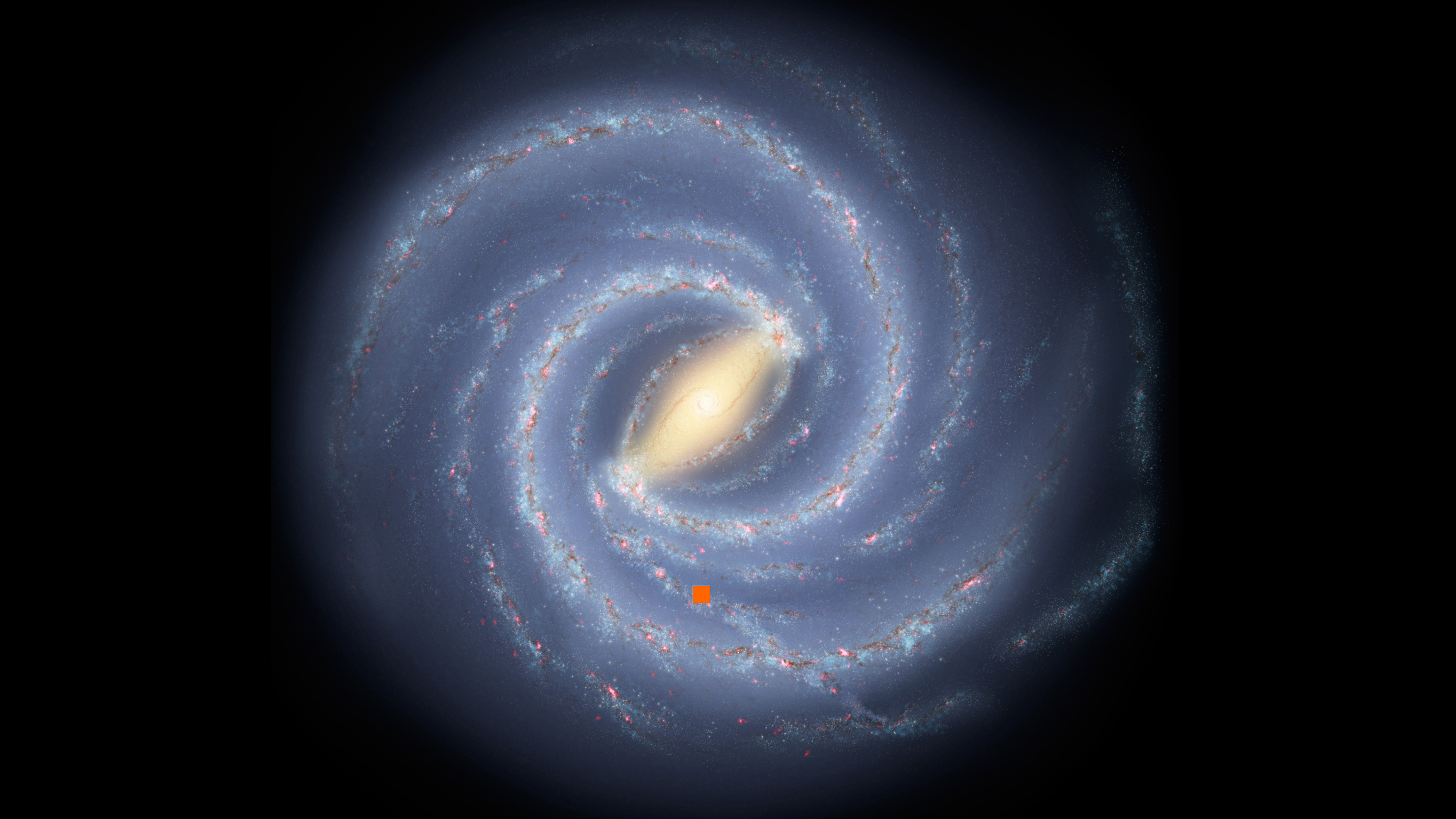
Signal Reconstruction with Python

Numerical Information Field Theory – a NIFTy tutorial

Philipp Arras, Torsten Enßlin,
Jakob Knollmüller
MPI for Astrophysics

IFT Team: Philipp Arras, Michael Bell, Vanessa Böhm, Sebastian Dorn, Martin Dupont, Mona Frommert, Philipp Frank, Mahsa Ghaempanah, Maksim Greiner, Philipp Haim, Sebastian Hutschenreuter, Henrik Junklewitz, Francisco-Shu Kitaura, Jakob Knollmüller, Christoph Lienhard, Reimar Leike, Ancla Müller, Johannes Oberpriller, Niels Oppermann, Natalia Porquerese, Daniel Pumpe, Tiago Ramalho, Martin Reinecke, Julia Stadler, Marco Selig, Theo Steininger, Valentina Vacca, Cornelius Weig, Margret Westerkamp, & many more





Galactic Tomography

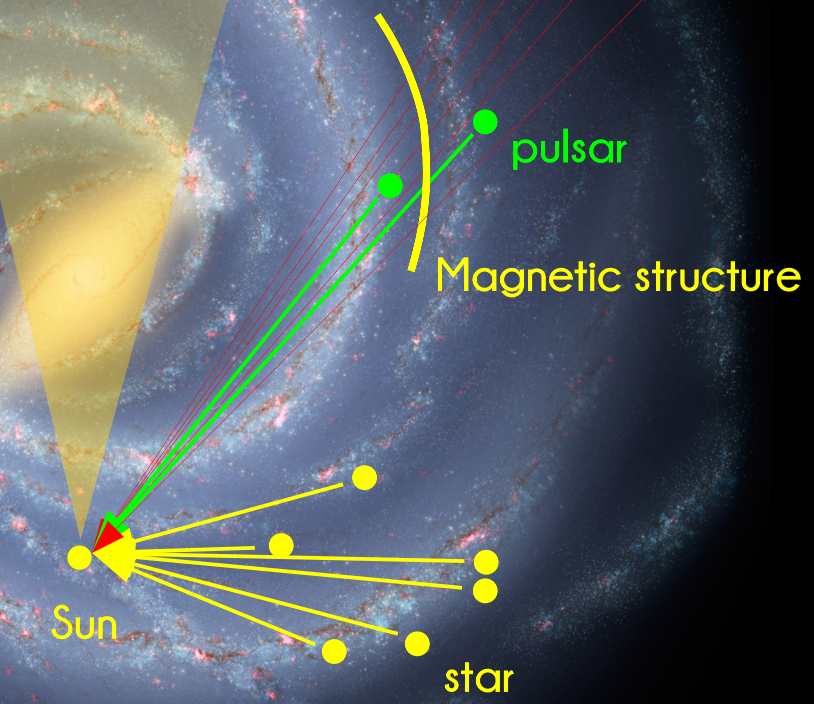
Pulsars:
Dispersion Measure → electron density
Rotation Measure → magnetic field x el. dens.
Scintillation Measure → el. dens. x turbulence

Extragalactic sources:
Rotation Measure → magnetic field x el. dens.
Ultra High Energy Cosmic Rays → mag. fields

Stars:
Dust reddening → dust density & properties
Positions → stellar density & radiation field
Kinematics → gravitational potential

Emission Processes:
Dust emission → dust density & radiation field
Synchrotron → relativistic el. x mag. Fields
Bremsstrahlung → thermal, rel. el. x gas density
Inverse Compton → rel. el. x radiation field
Hadronic interactions → rel. nuclei x gas density
Lines (21 cm, CO, ...) → gas density & kinematics

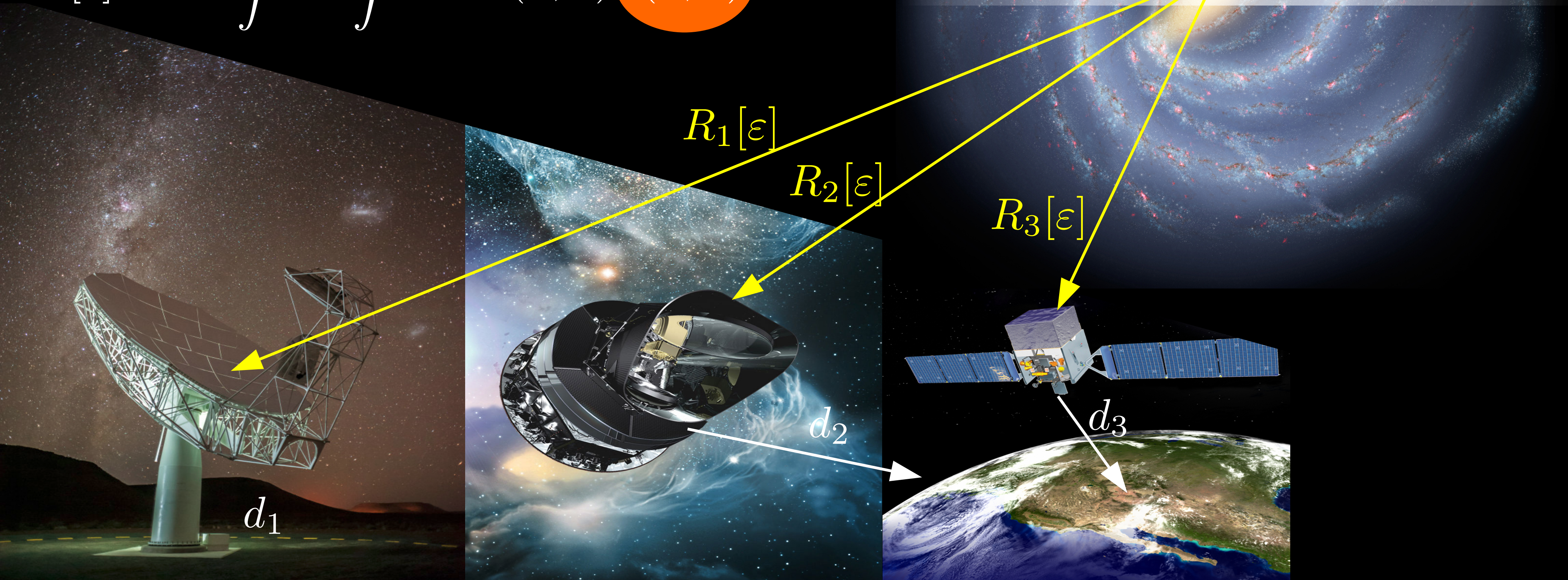
Other information sources:
Correlation structures (auto- & cross-correlations)
Approximate symmetries
Physical laws
Empirical laws, ...



Data Fusion

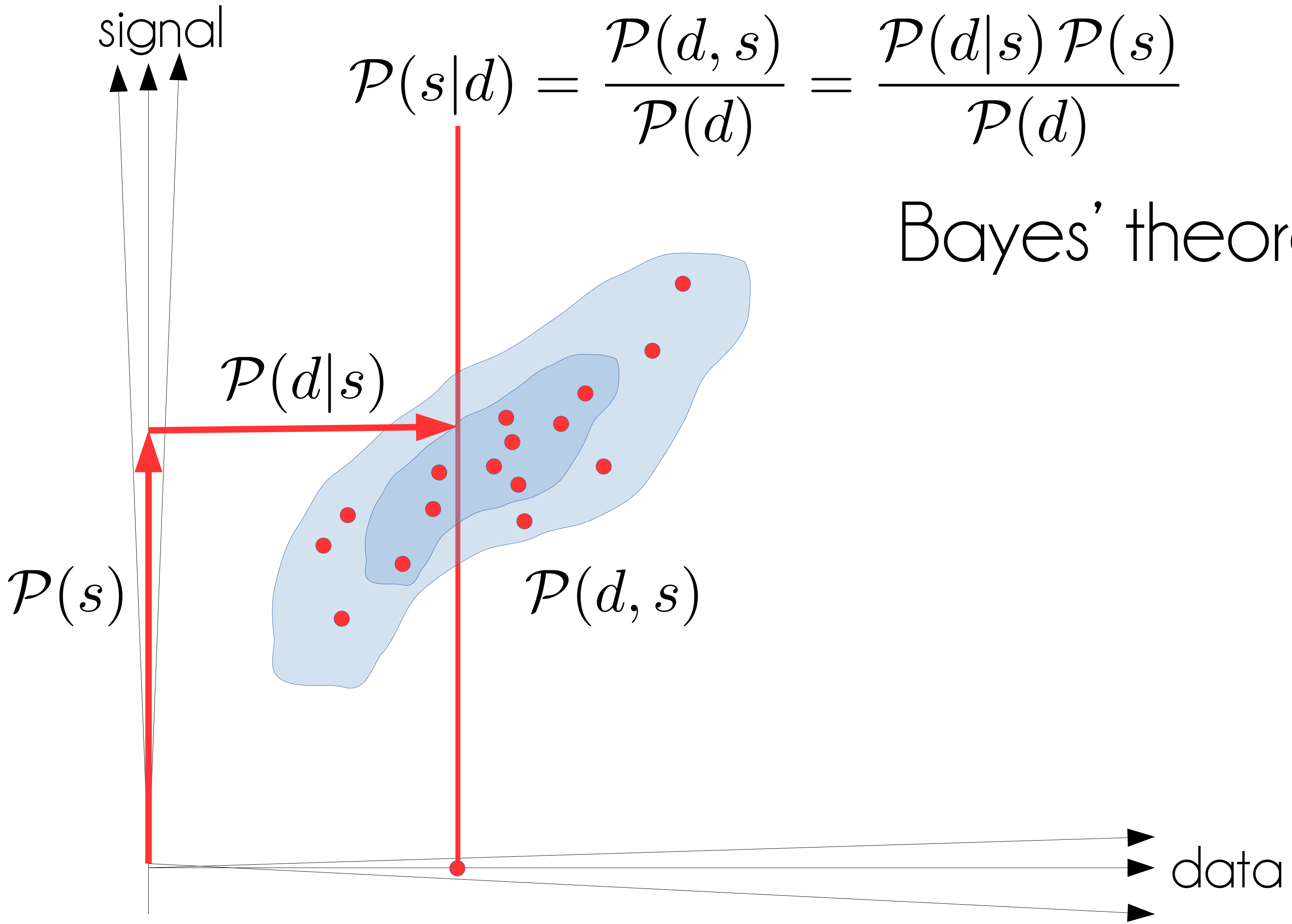
$$d_i = R_i[\varepsilon] + n_i$$

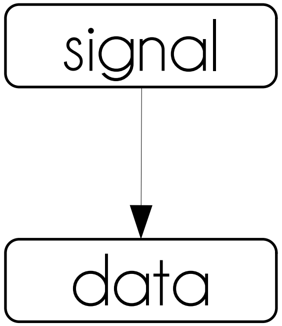
$$R_i[\varepsilon] = \int \mathrm{d}x \int \mathrm{d}\nu \, R_i(x,\nu)\, \varepsilon(x,\nu)$$



$$\mathcal{P}(s|d) = \frac{\mathcal{P}(d,s)}{\mathcal{P}(d)} = \frac{\mathcal{P}(d|s)\,\mathcal{P}(s)}{\mathcal{P}(d)}$$

Bayes' theorem





Information theory

$$\mathcal{P}(s|d) = \frac{\mathcal{P}(d,s)}{\mathcal{P}(d)} = \frac{e^{-\mathcal{H}(d,s)}}{\mathcal{Z}(d)}$$

$$\mathcal{H}(d,s) = -\log \mathcal{P}(d,s)$$

Information

$$\mathcal{Z}(d) = \mathcal{P}(d) = \int \mathcal{D}s\, \mathcal{P}(d,s)$$

$$\mathcal{P}(d,s) = \mathcal{P}(d|s)\, \mathcal{P}(s)$$

$$\mathcal{H}(d,s) = \mathcal{H}(d|s) + \mathcal{H}(s)$$

is additive

metric regularization

Information theory

$$
\begin{aligned}
\mathcal{P}(s|d) &= \frac{\mathcal{P}(d,s)}{\mathcal{P}(d)} = \frac{e^{-\mathcal{H}(d,s)}}{\mathcal{Z}(d)} \\
\mathcal{H}(d,s) &= -\log \mathcal{P}(d,s) \\
\mathcal{Z}(d) &= \mathcal{P}(d) \\
&= \int \mathcal{D}s\, \mathcal{P}(d,s) \\
\mathcal{P}(d,s) &= \mathcal{P}(d|s)\, \mathcal{P}(s) \\
\mathcal{H}(d_1, d_2, s) &= \mathcal{H}(d_1|s) + \mathcal{H}(d_2|s) + \mathcal{H}(s)
\end{aligned}
$$

Information

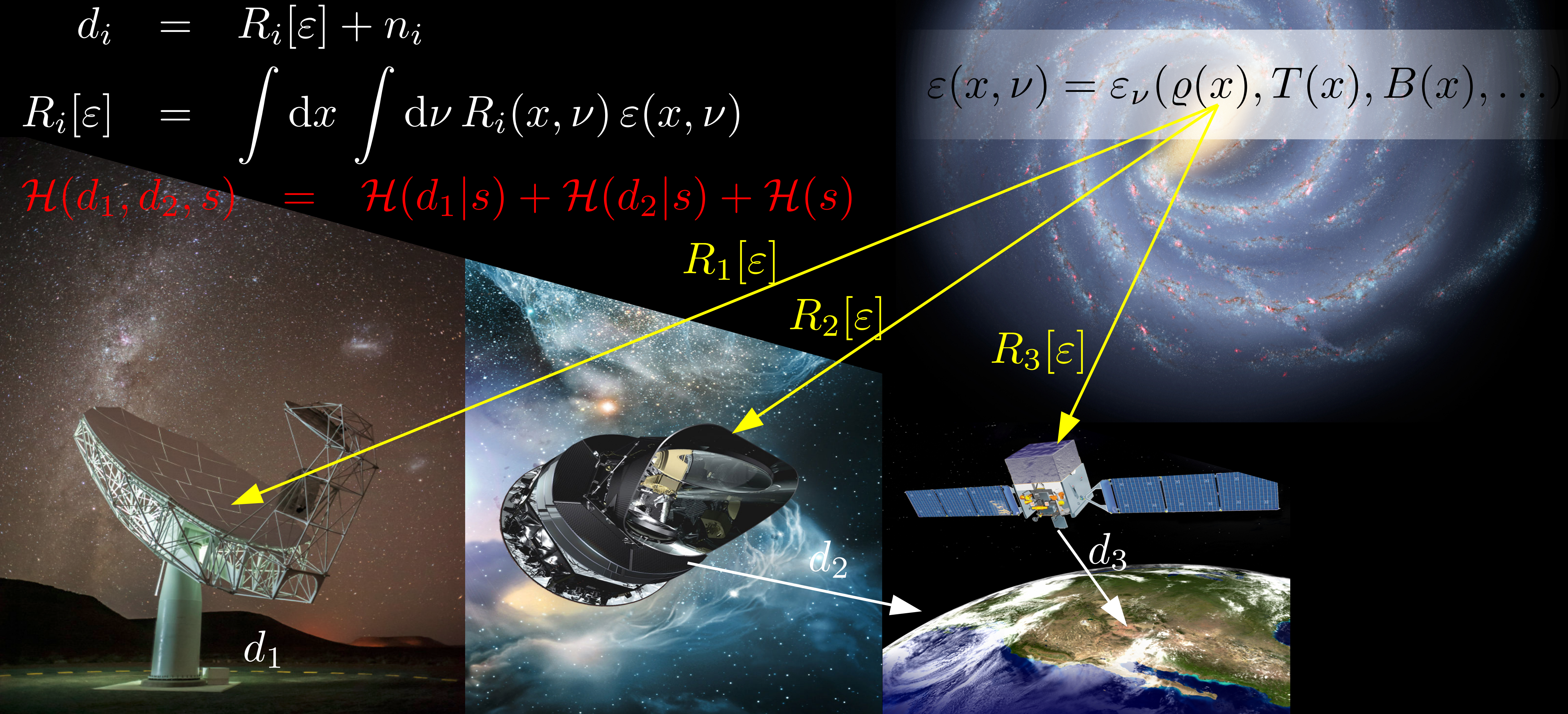
is additive

Data Fusion

$$d_i = R_i[\varepsilon] + n_i$$

$$R_i[\varepsilon] = \int \mathrm{d}x \int \mathrm{d}\nu \, R_i(x,\nu)\,\varepsilon(x,\nu)$$

$$\mathcal{H}(d_1, d_2, s) = \mathcal{H}(d_1|s) + \mathcal{H}(d_2|s) + \mathcal{H}(s)$$



Probability & Information

$$\mathcal{P}(s) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{s^2}{2\sigma^2}}$$

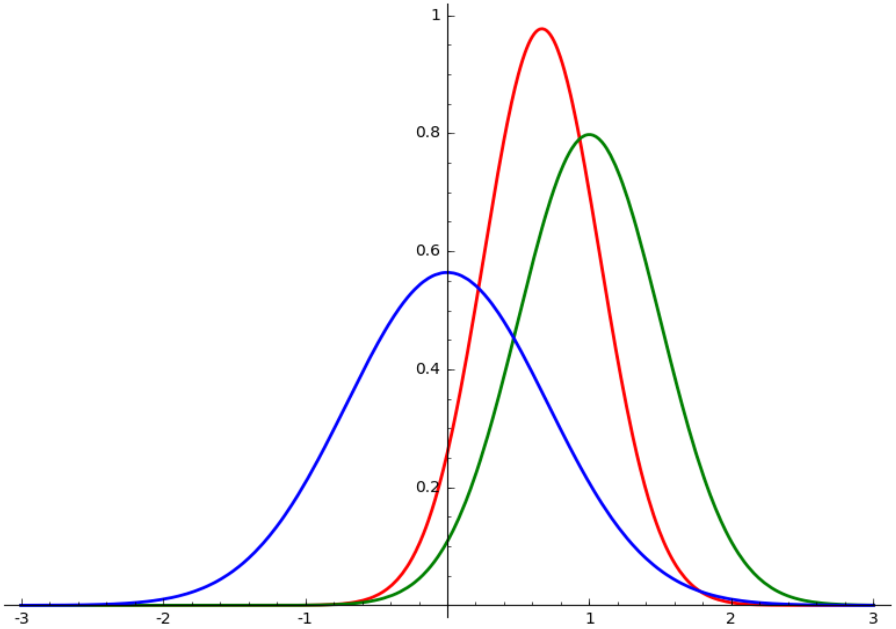
$$\mathcal{P}(d|s) \propto e^{-\frac{(s-d)^2}{2\sigma'^2}}$$

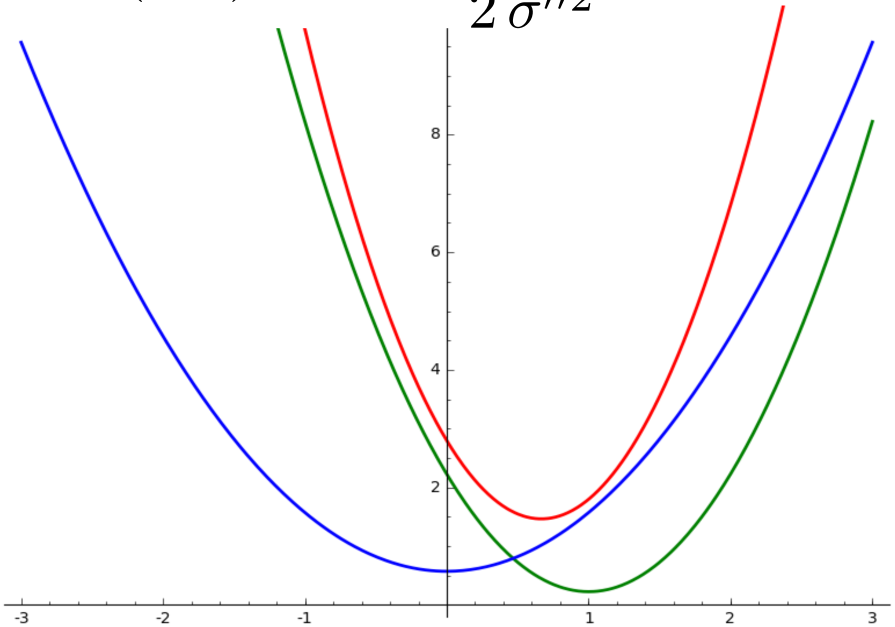
$$\mathcal{P}(s|d) \propto e^{-\frac{(s-m)^2}{2\sigma''^2}}$$

$$\mathcal{H}(s) \widehat{=} \frac{s^2}{2\,\sigma^2}$$

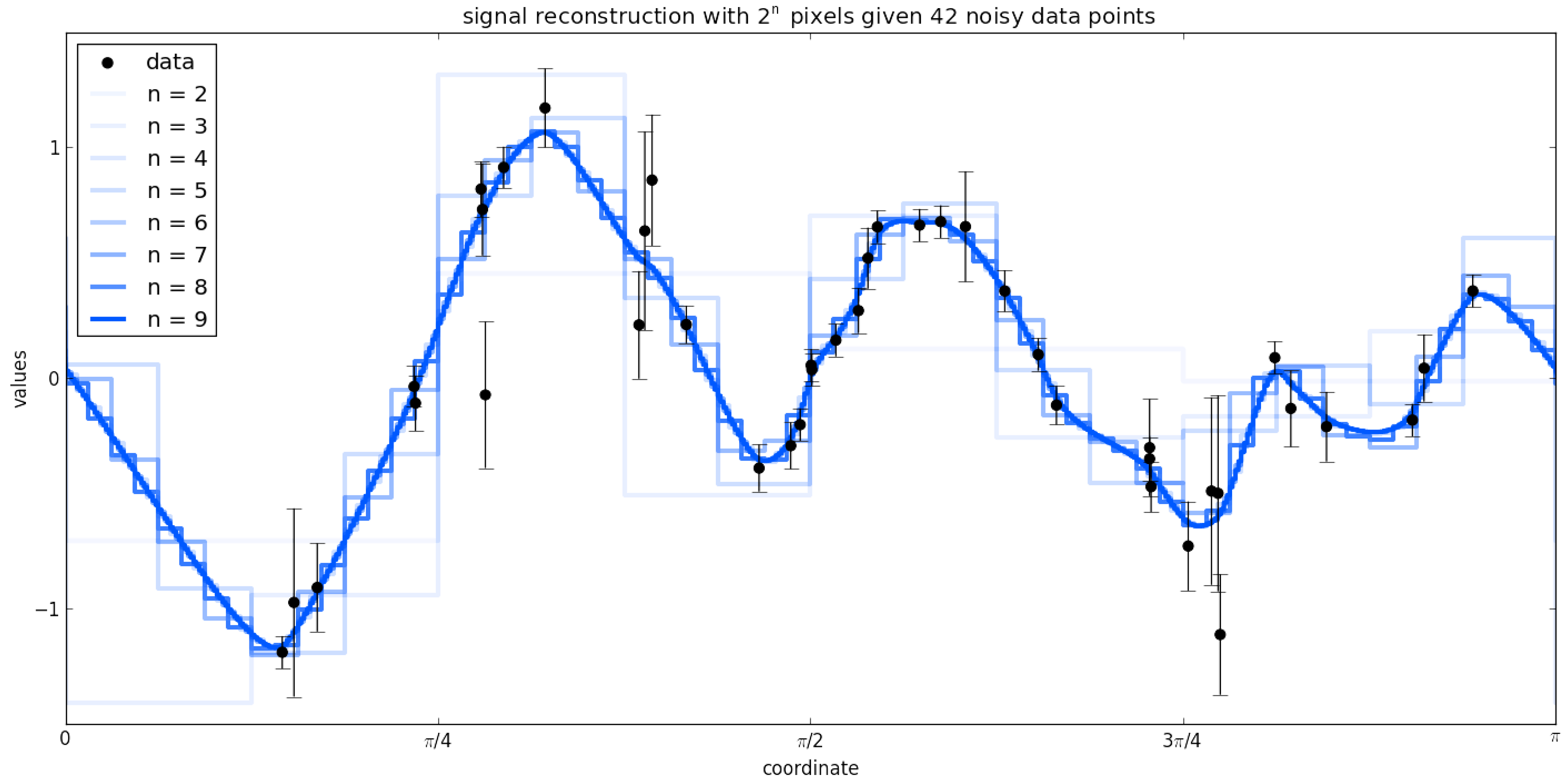
$$\mathcal{H}(d|s) \widehat{=} \frac{(s-d)^2}{2\,\sigma'^2}$$

$$\mathcal{H}(d,s) \widehat{=} \frac{(s-m)^2}{2\,\sigma''^2}$$





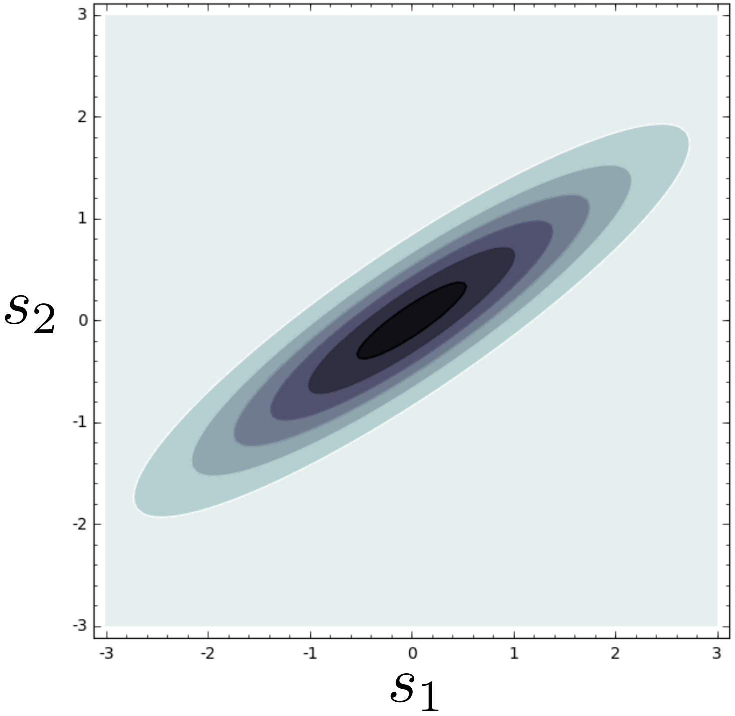

Correlations



signal reconstruction with 2ⁿ pixels given 42 noisy data points
data
n = 2
n = 3
n = 4
n = 5
n = 6
n = 7
n = 8
n = 9
1
0
−1
values
0
π/4
π/2
3π/4
π
coordinate

Correlations

$$\mathcal{P}(s) \qquad s = \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$



$$d = s_1 + n$$

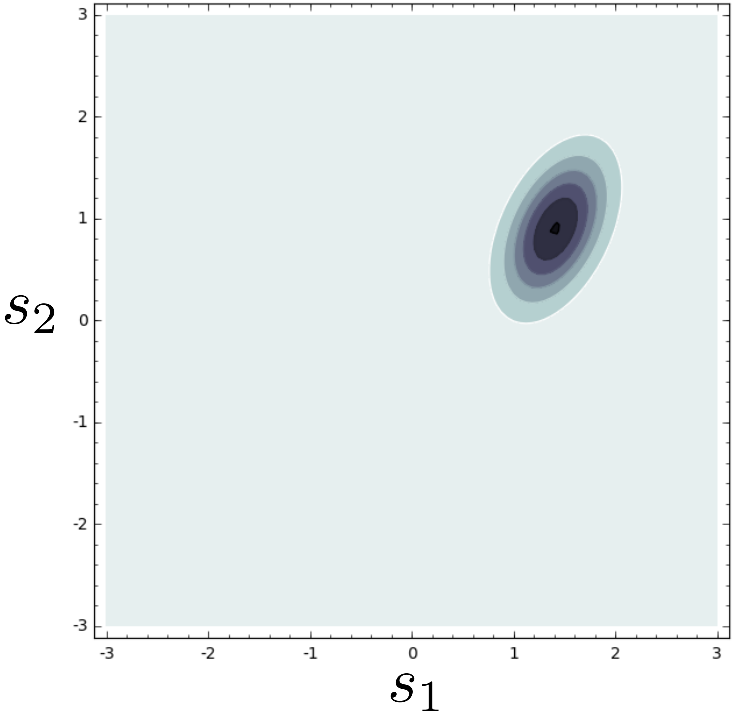
Correlations

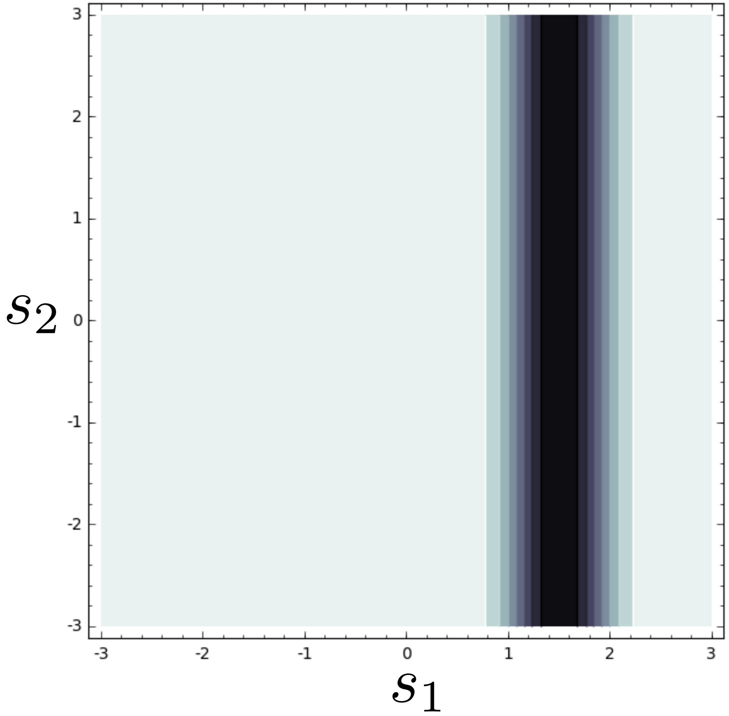
$\mathcal{P}(s|d)$

$$s = \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$

$\mathcal{P}(d|s)$

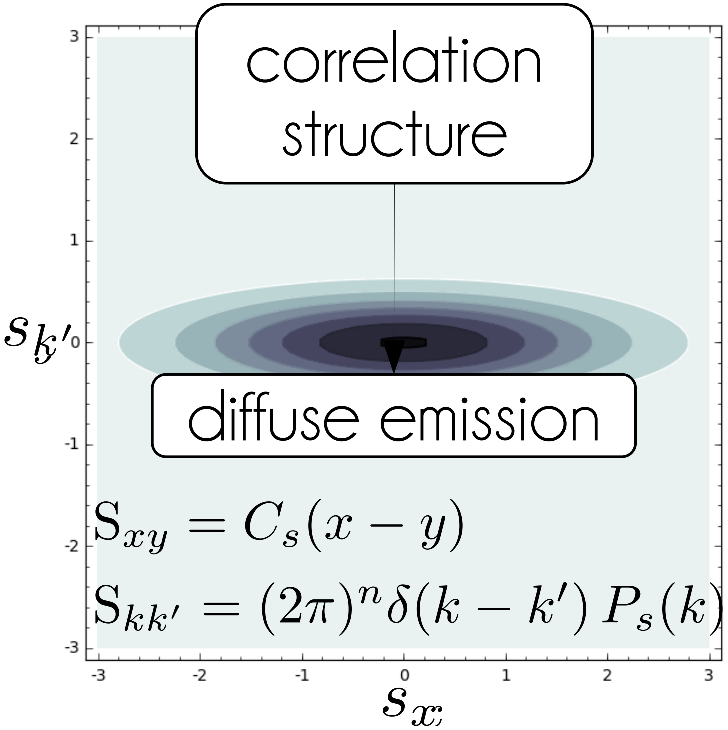
$d = s_1 + n$



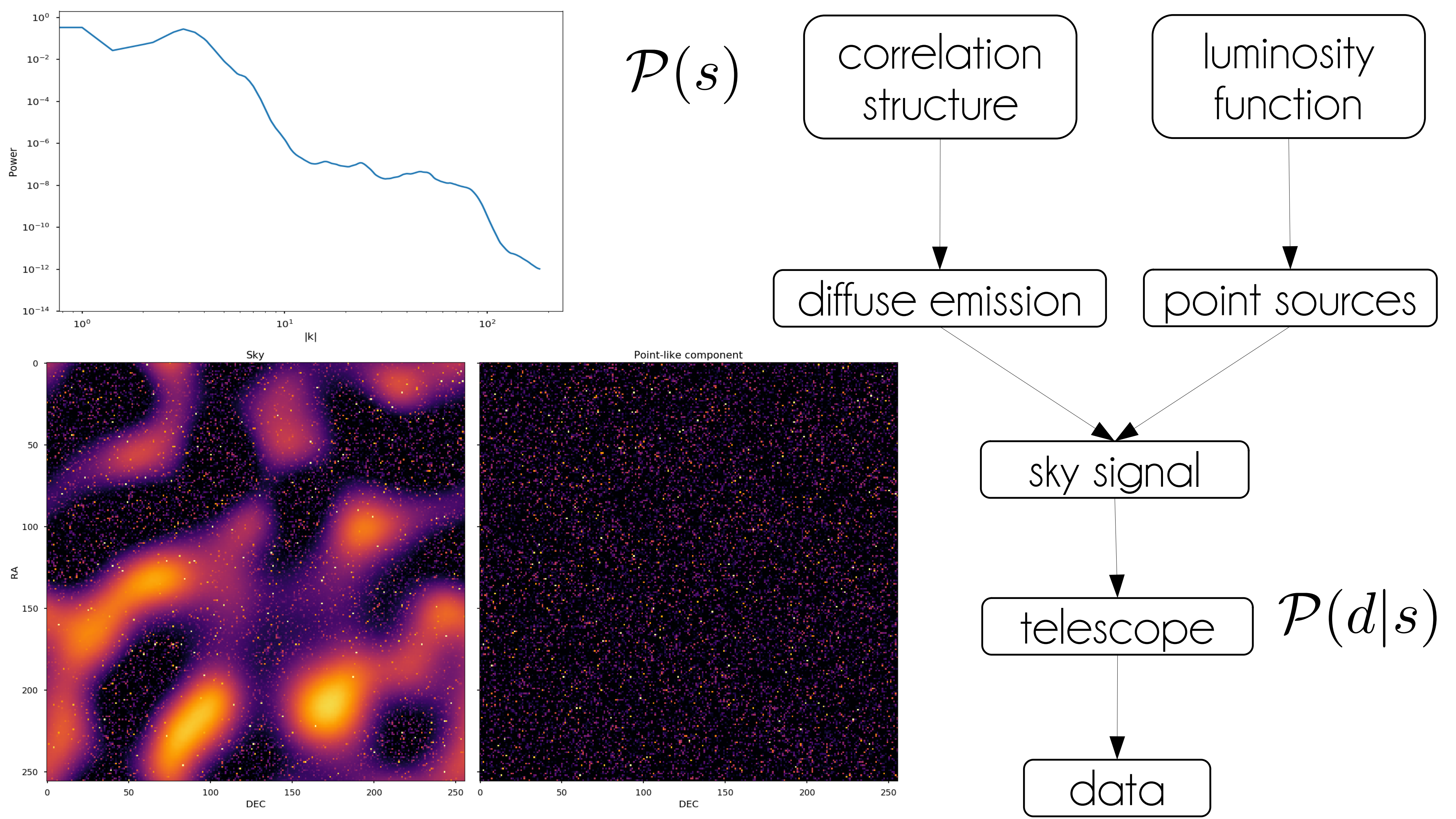


Correlations

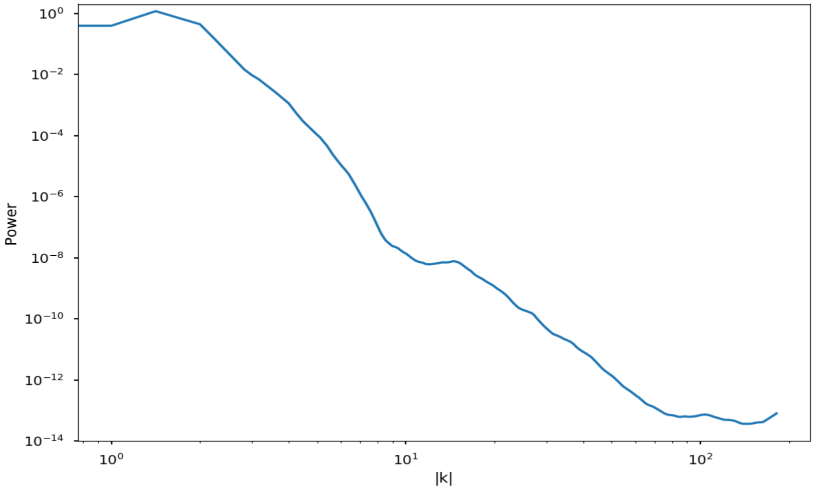
$$\mathcal{P}(s) \qquad s = \begin{pmatrix} s_1 \\ s_2 \end{pmatrix}$$

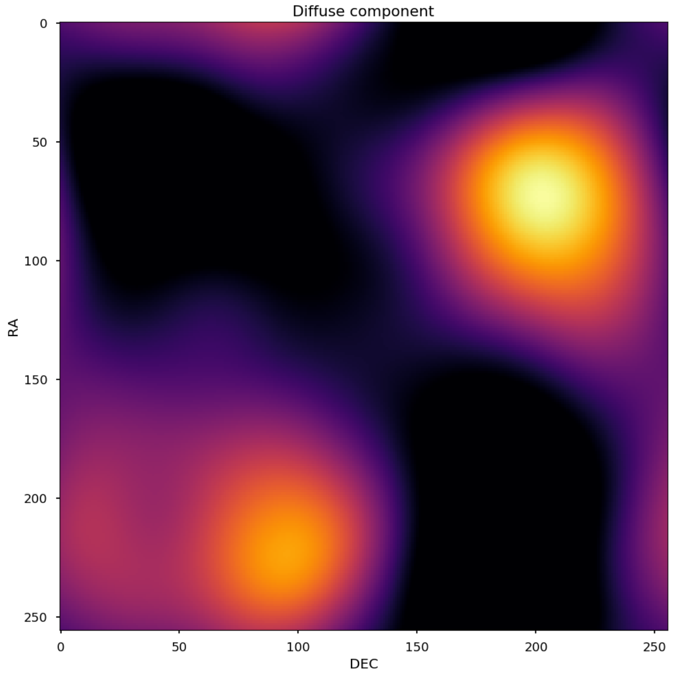


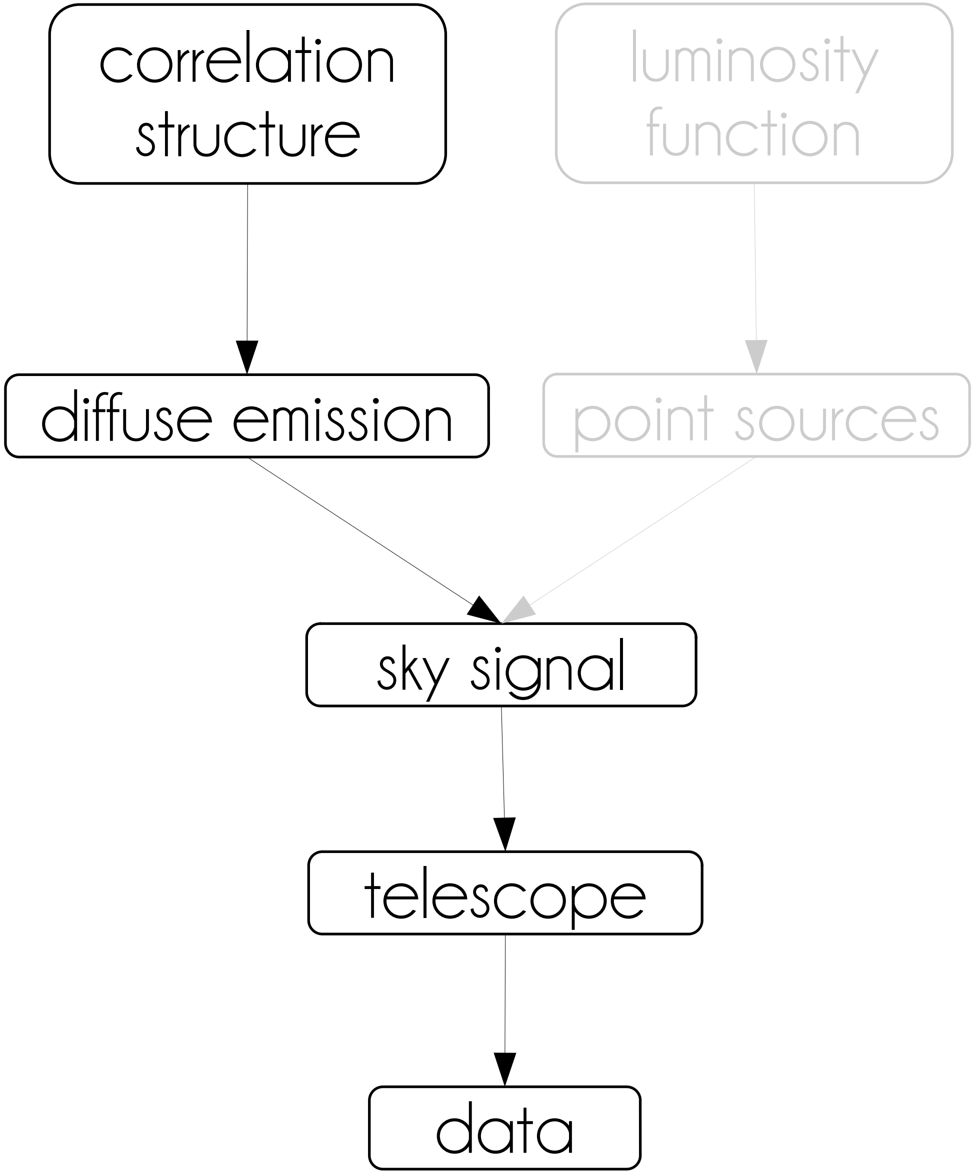
$$\begin{aligned}
\mathcal{P}(s) &= \mathcal{G}(s, S) \\
&= \frac{1}{\sqrt{|2\pi S|}} \exp\left(-\frac{1}{2} s^\dagger S^{-1} s\right) \\
S &= \begin{pmatrix} \langle s_1 s_1 \rangle & \langle s_1 s_2 \rangle \\ \langle s_2 s_1 \rangle & \langle s_2 s_2 \rangle \end{pmatrix} \quad \text{2-dim.} \\
S_{ij} &= \langle s_i s_j \rangle \quad n\text{-dim.} \\
S_{xy} &= \langle s_x s_y \rangle, \; x \in \mathbb{R}^n \quad \infty\text{-dim.}
\end{aligned}$$



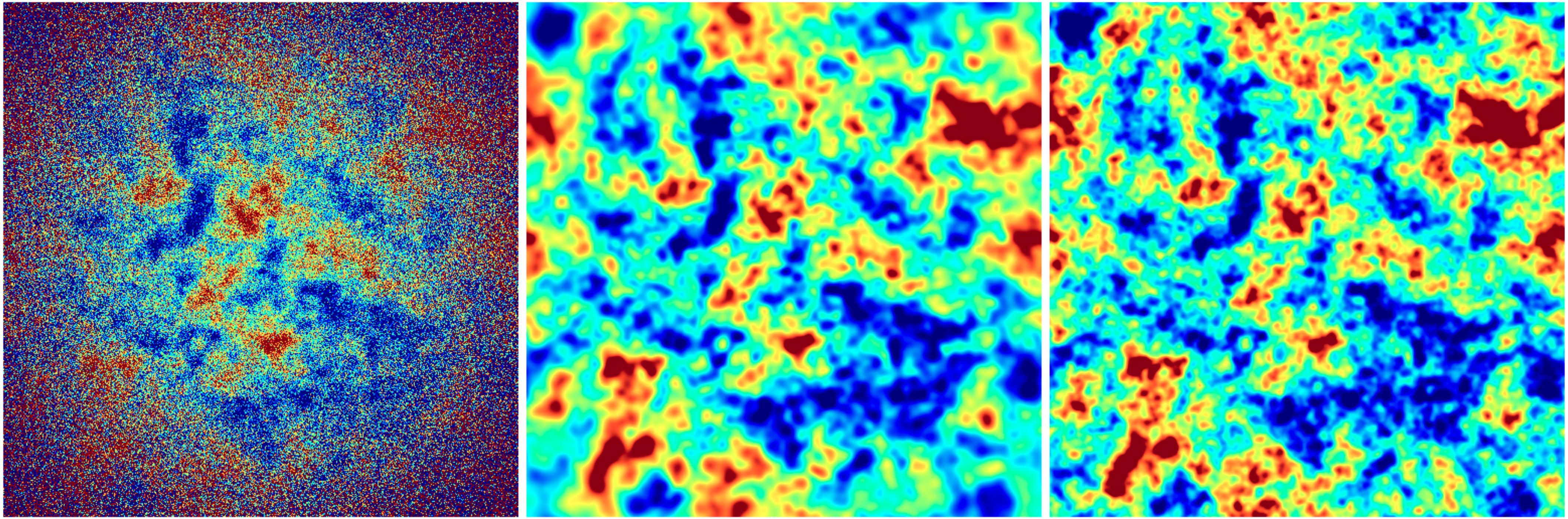
Power
|k|
$\mathcal{P}(s)$
Sky
Point-like component
RA
DEC
DEC
correlation structure
luminosity function
diffuse emission
point sources
sky signal
telescope
$\mathcal{P}(d|s)$
data


Power
|k|
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-14}
10^{1}
10^{2}


Diffuse component
RA
DEC
0
50
100
150
200
250


correlation structure
luminosity function
diffuse emission
point sources
sky signal
telescope
data

Wiener Filter



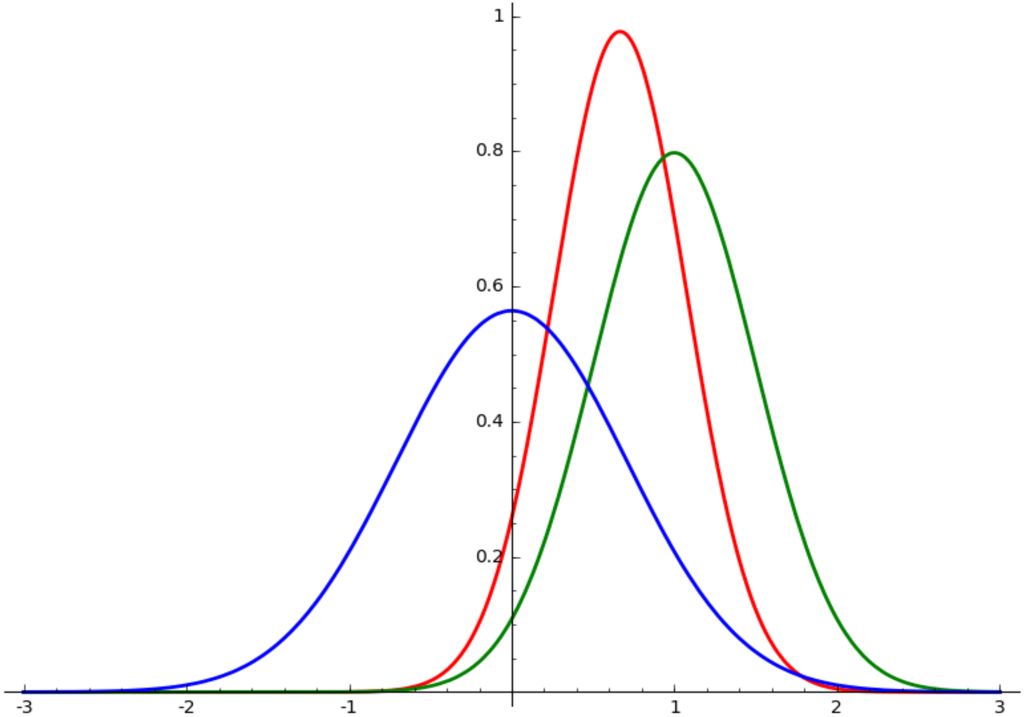
Noisy data | Wiener filtered | True signal

https://en.wikipedia.org/wiki/Generalized_Wiener_filter

$$d = R\,s + n$$ data

$$\mathcal{P}(d, s|R, S, N) = \mathcal{G}(s, S)\,\mathcal{G}(d - R\,s, N)$$ prior & likelihood

$$\mathcal{P}(s|d, R, S, N) = \mathcal{G}(s - m, D)$$ posterior



$$d = R\,s + n$$ data

$$\mathcal{P}(d, s|R, S, N) = \mathcal{G}(s, S)\,\mathcal{G}(d - R\,s, N)$$ prior & likelihood

$$\mathcal{P}(s|d, R, S, N) = \mathcal{G}(s - m, D)$$ posterior

$$\mathcal{H}(d, s|R, S, N) \,\widehat{=}\, \frac{1}{2}\,s^\dagger S^{-1} s + \frac{1}{2}\,(d - R\,s)^\dagger N^{-1}(d - R\,s)$$

$$\widehat{=}\, \frac{1}{2}\,[s^\dagger \underbrace{\left(S^{-1} + R^\dagger N^{-1} R\right)}_{=D^{-1}} s + s\,\underbrace{R^\dagger N^{-1} d}_{=j} + \underbrace{d^\dagger N^{-1} R}_{=j^\dagger}\; s]$$

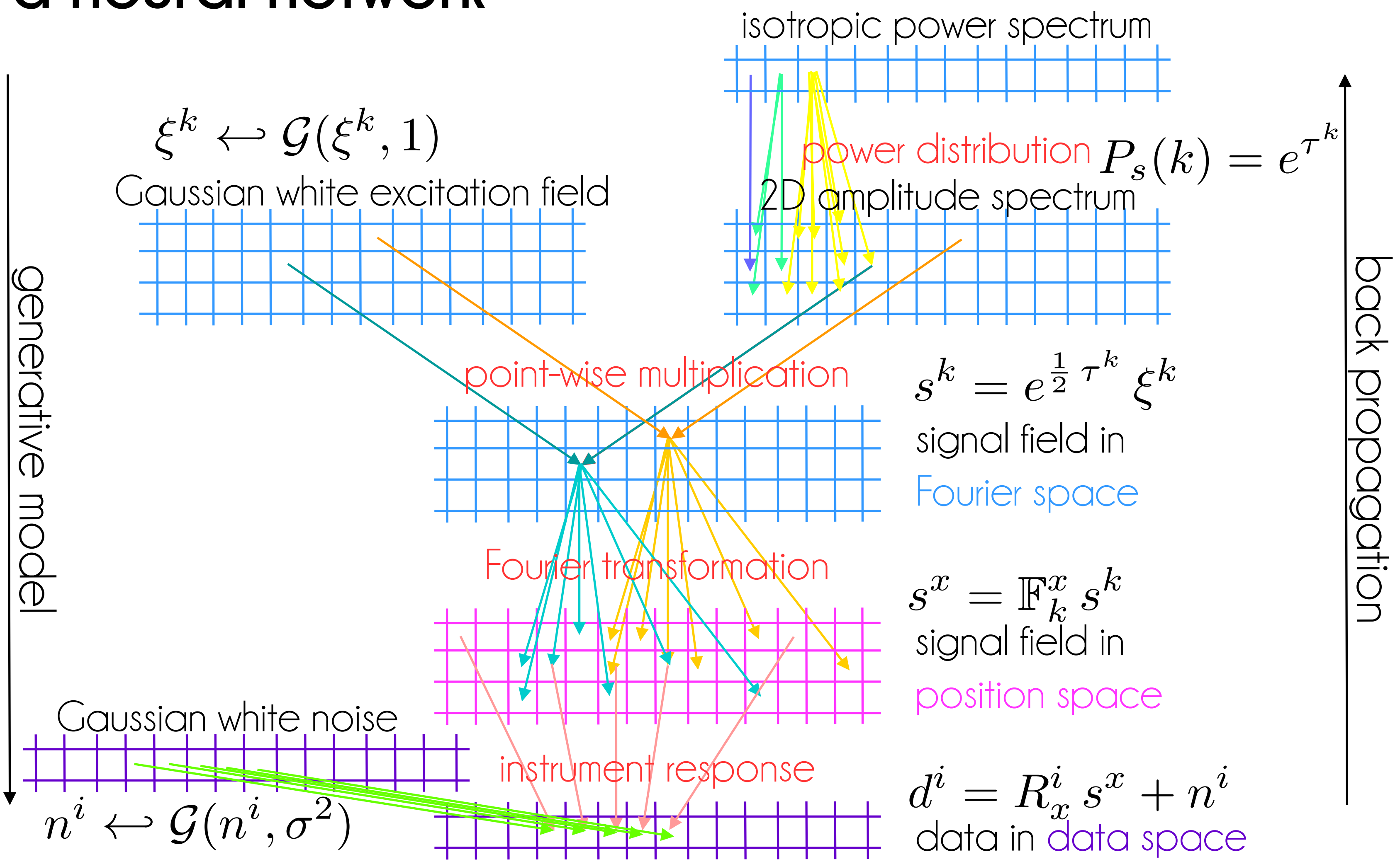
$$= \frac{1}{2}\,[s^\dagger D^{-1} s + s^\dagger j + j^\dagger s]$$

$$= \frac{1}{2}\,[s^\dagger D^{-1} s + s^\dagger D^{-1}\,\underbrace{D\,j}_{=m} + j^\dagger D\,D^{-1} s]$$

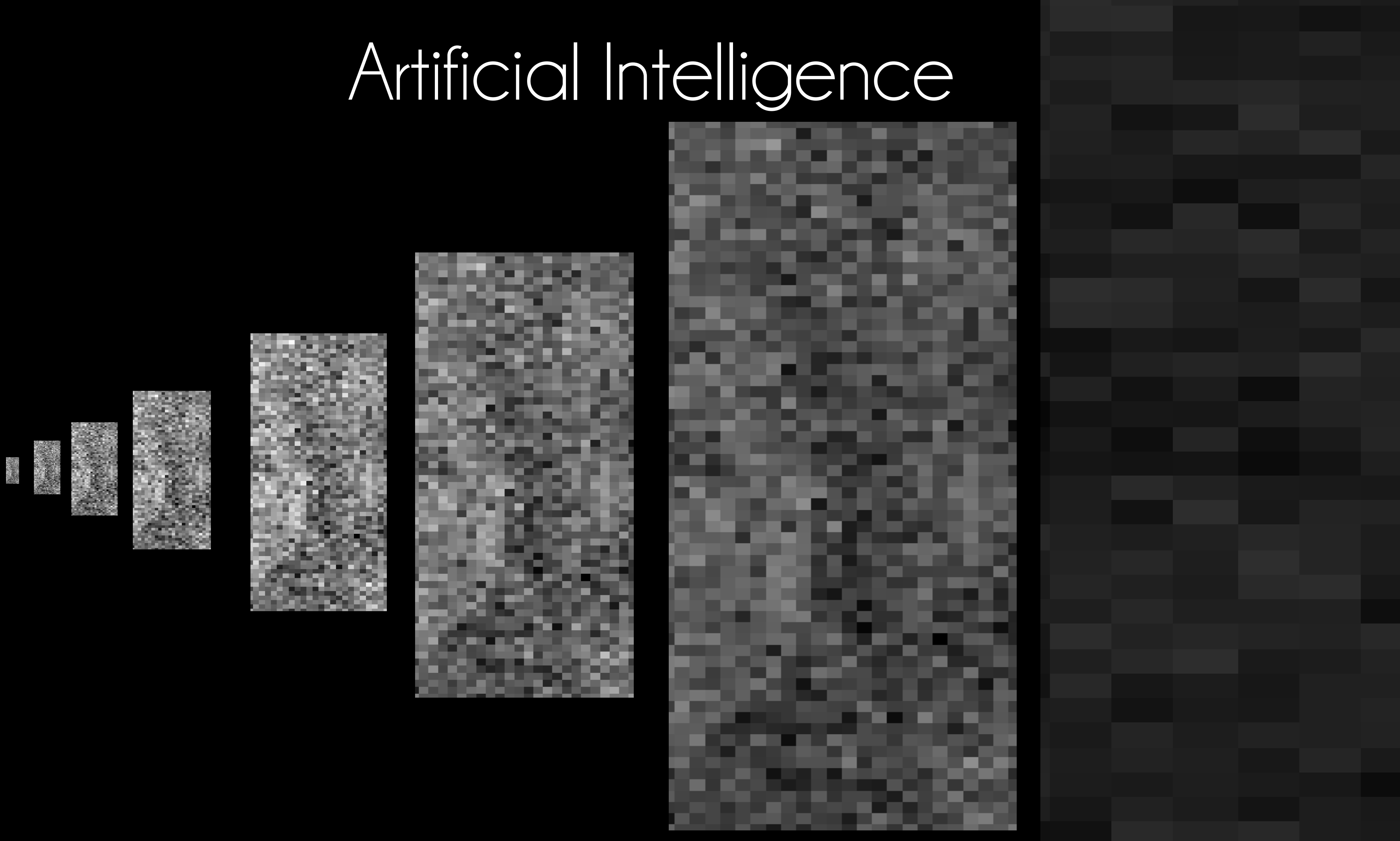
$$\widehat{=}\, \frac{1}{2}\,[(s - m)^\dagger D^{-1}(s - m)]$$

$$
\begin{aligned}
d &= R\,s + n & \text{data}\\
\mathcal{P}(d, s|R, S, N) &= \mathcal{G}(s, S)\,\mathcal{G}(d - R\,s, N) & \text{prior \& likelihood}\\
\mathcal{P}(s|d, R, S, N) &= \mathcal{G}(s - m, D) & \text{posterior}\\
m &= D\,j & \text{posterior mean}\\
j &= R^\dagger N^{-1} d & \text{information source}\\
D &= \left(S^{-1} + R^\dagger N^{-1} R\right)^{-1} & \text{information propagator}
\end{aligned}
$$

IFT as a neural network



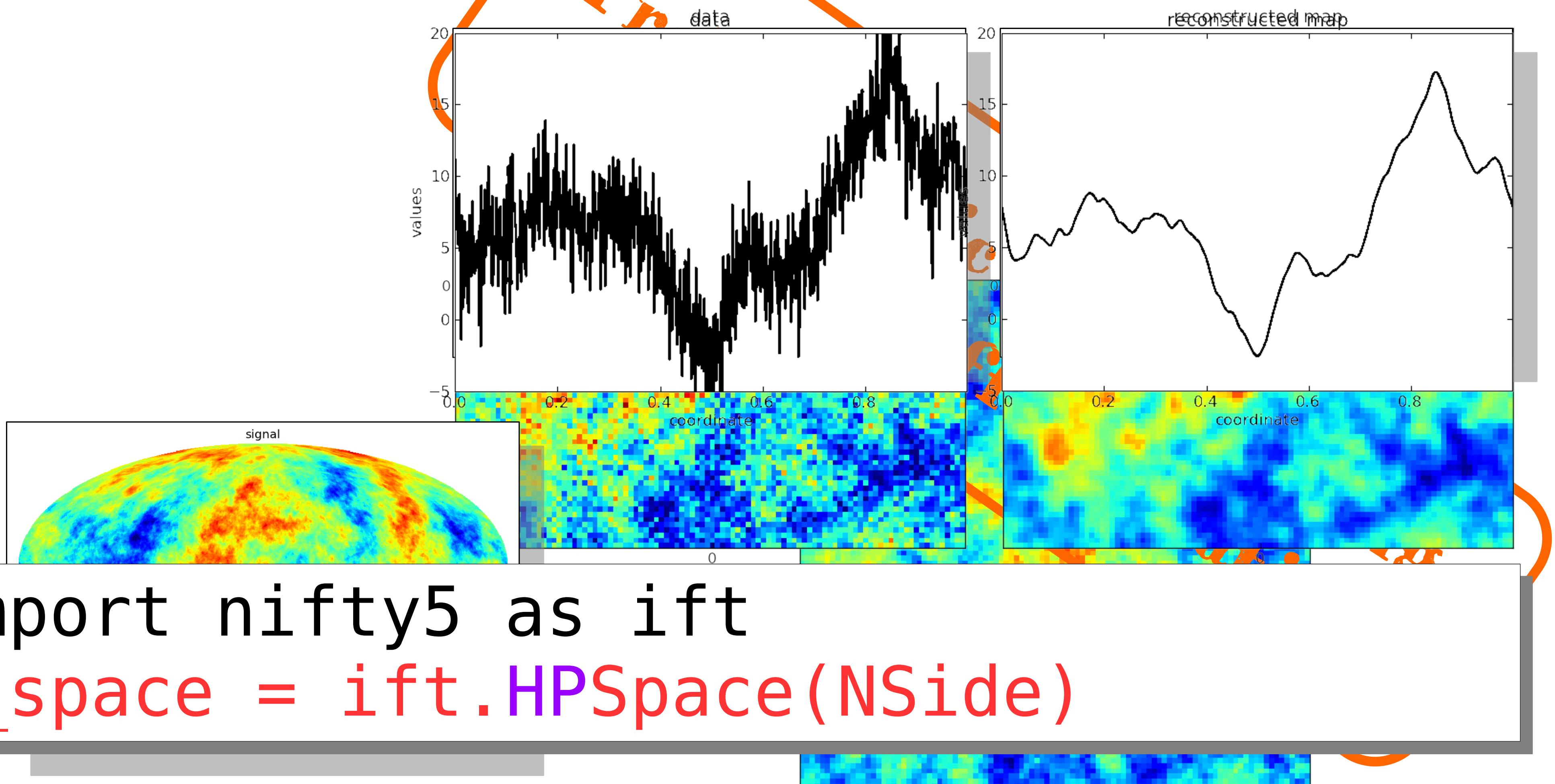
Artificial Intelligence



NIFTy5

NIFTy – Numerical Information Field Theory

NIFTy [1], [2], "Numerical Information Field Theory" is a versatile library designed to enable the development of signal inference algorithms that are independent of the underlying grids (spatial, spectral, temporal, …) and their resolutions. Its object-oriented framework is written in Python

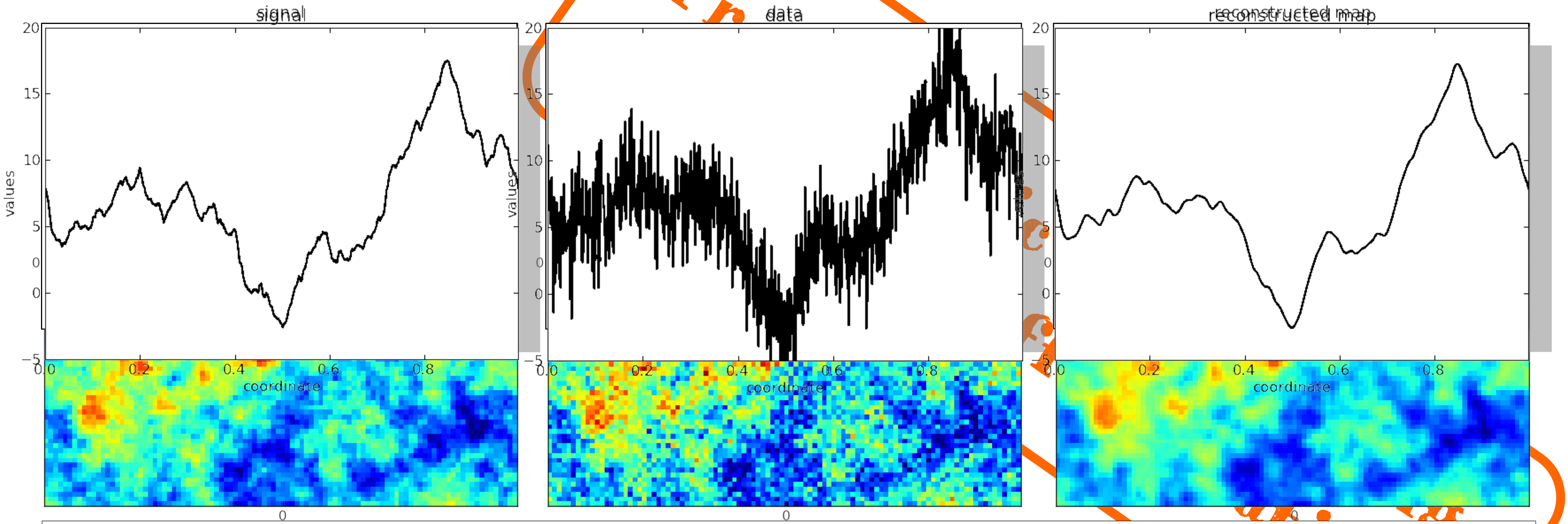



```
import nifty5 as ift
s_space = ift.HPSpace(NSide)
```

NIFTy5

NIFTy – Numerical Information Field Theory

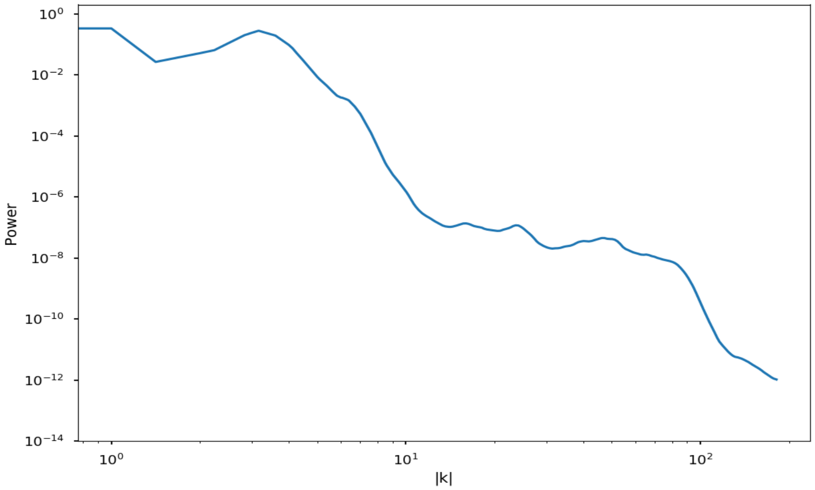
NIFTy [1], [2], "Numerical Information Field Theory" is a versatile library designed to enable the development of signal inference algorithms that are independent of the underlying grids (spatial, spectral, temporal, …) and their resolutions. Its object-oriented framework is written in Python

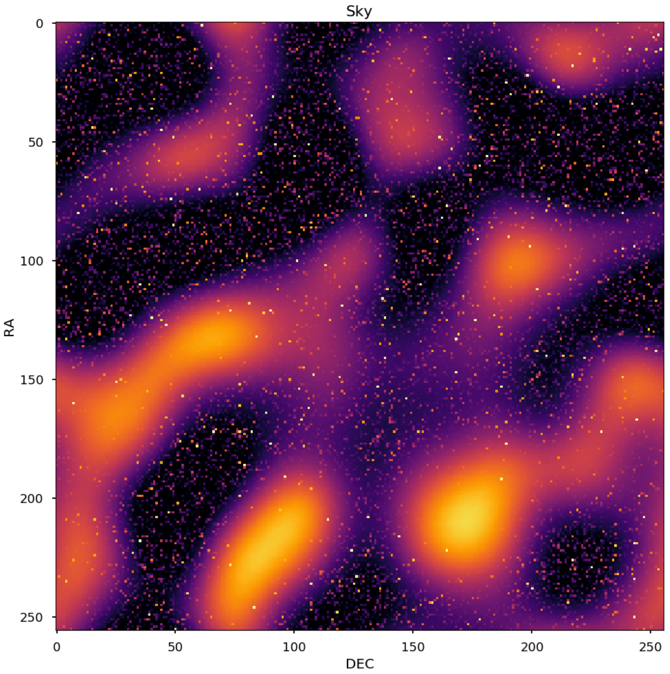


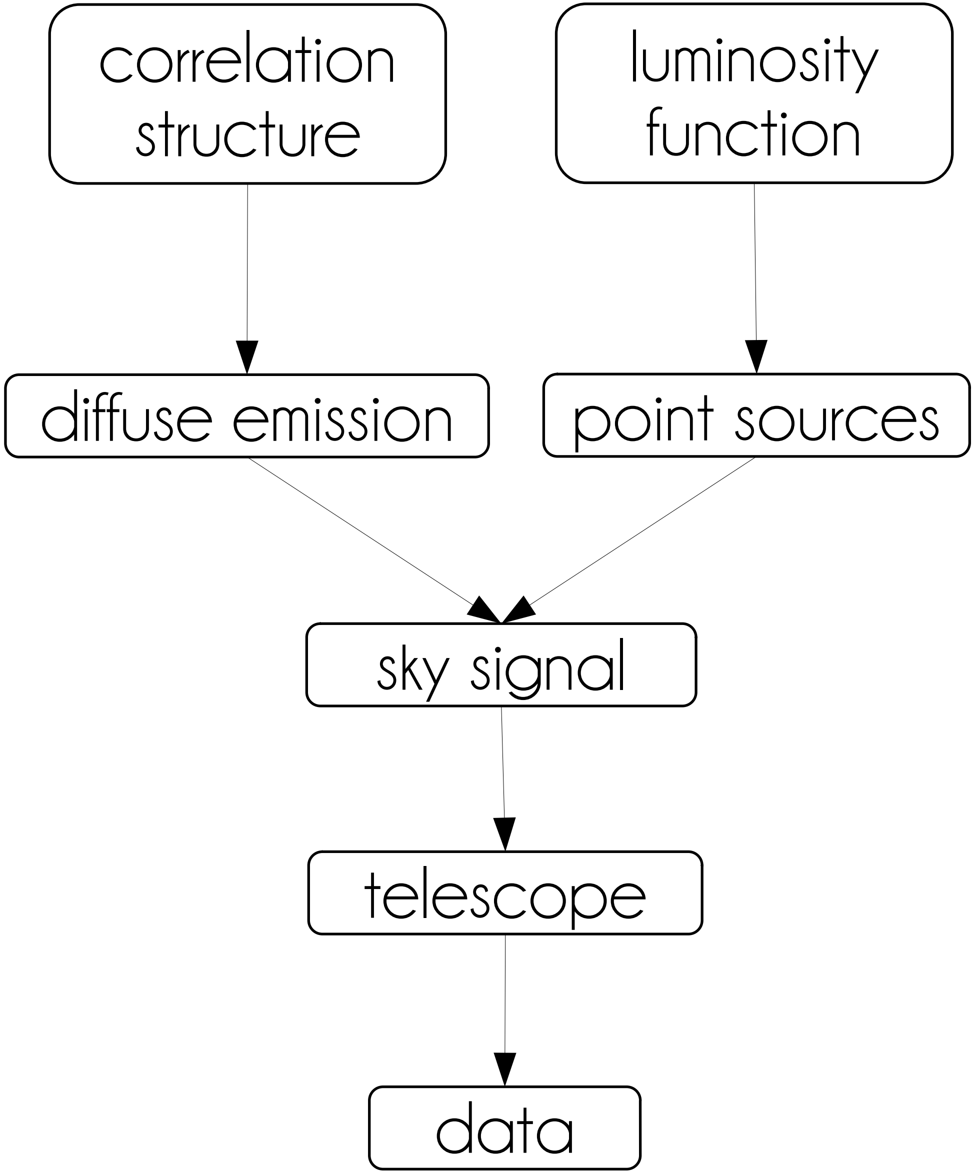

```
import nifty5 as ift
s_space = ift.HPSpace(NSide)
```

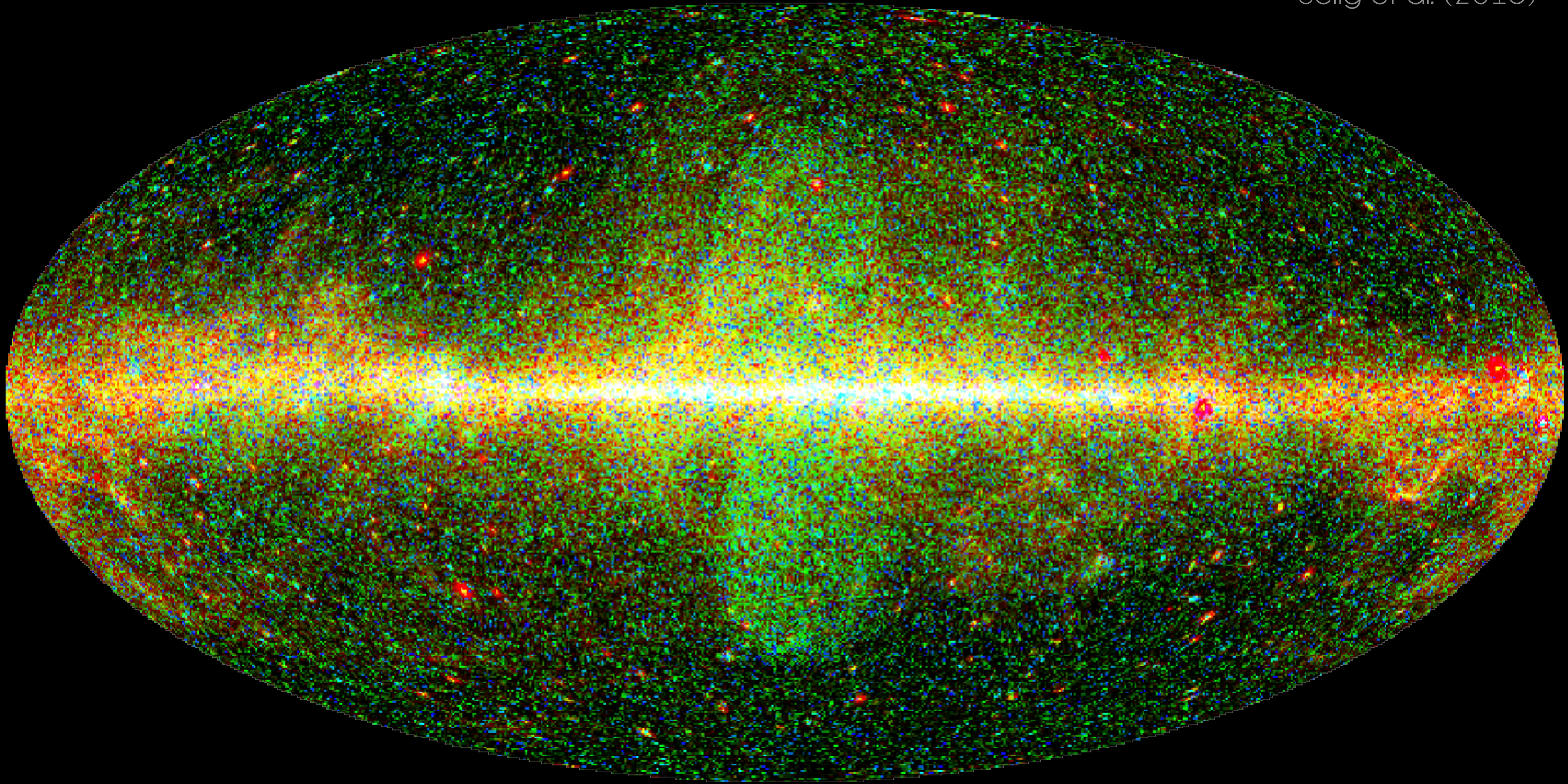
NIFTy tutorial part 1

linear reconstructions

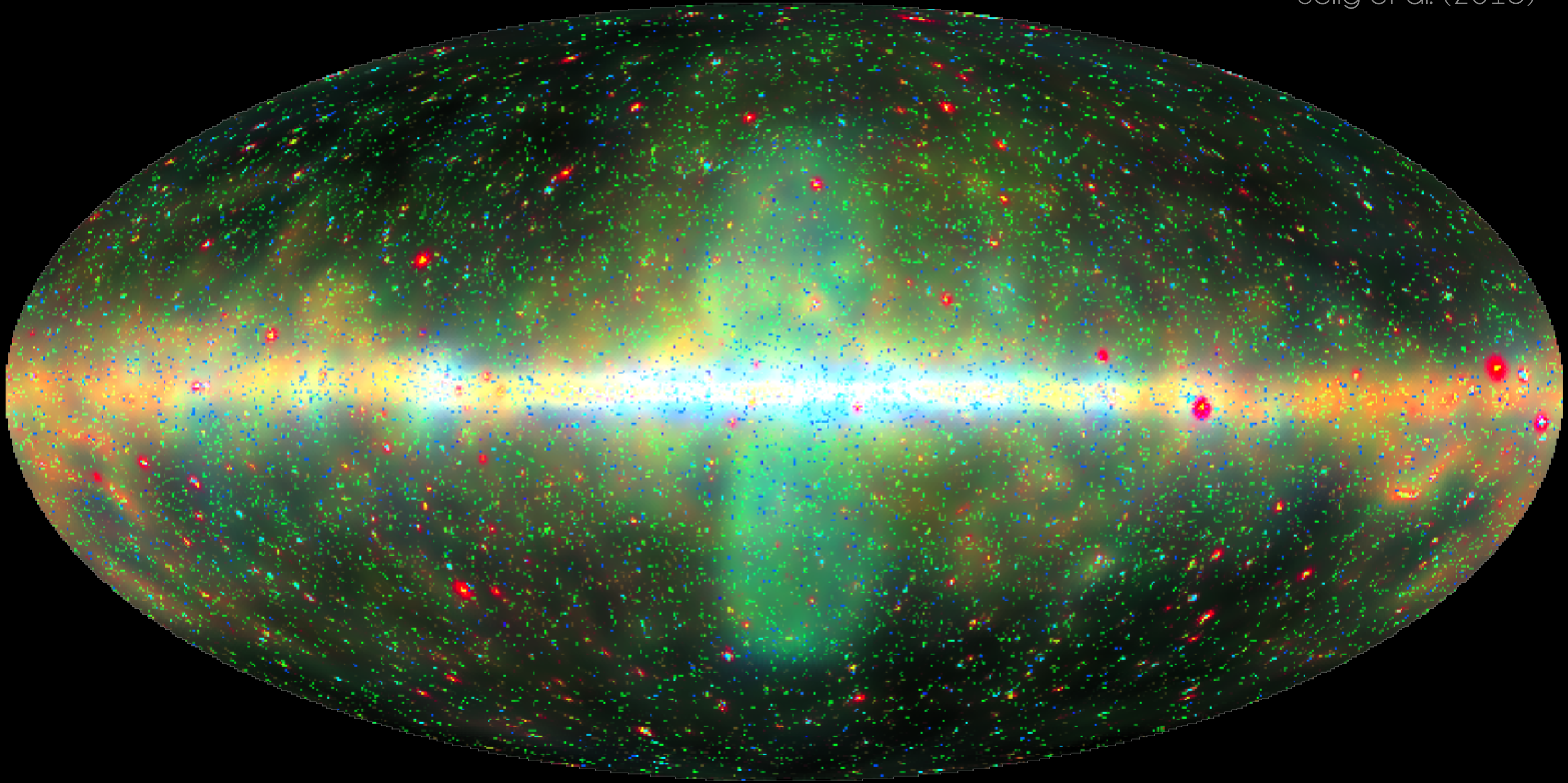

Power
|k|
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-14}
10^{0}
10^{1}
10^{2}


Sky
RA
DEC
0
50
100
150
200
250

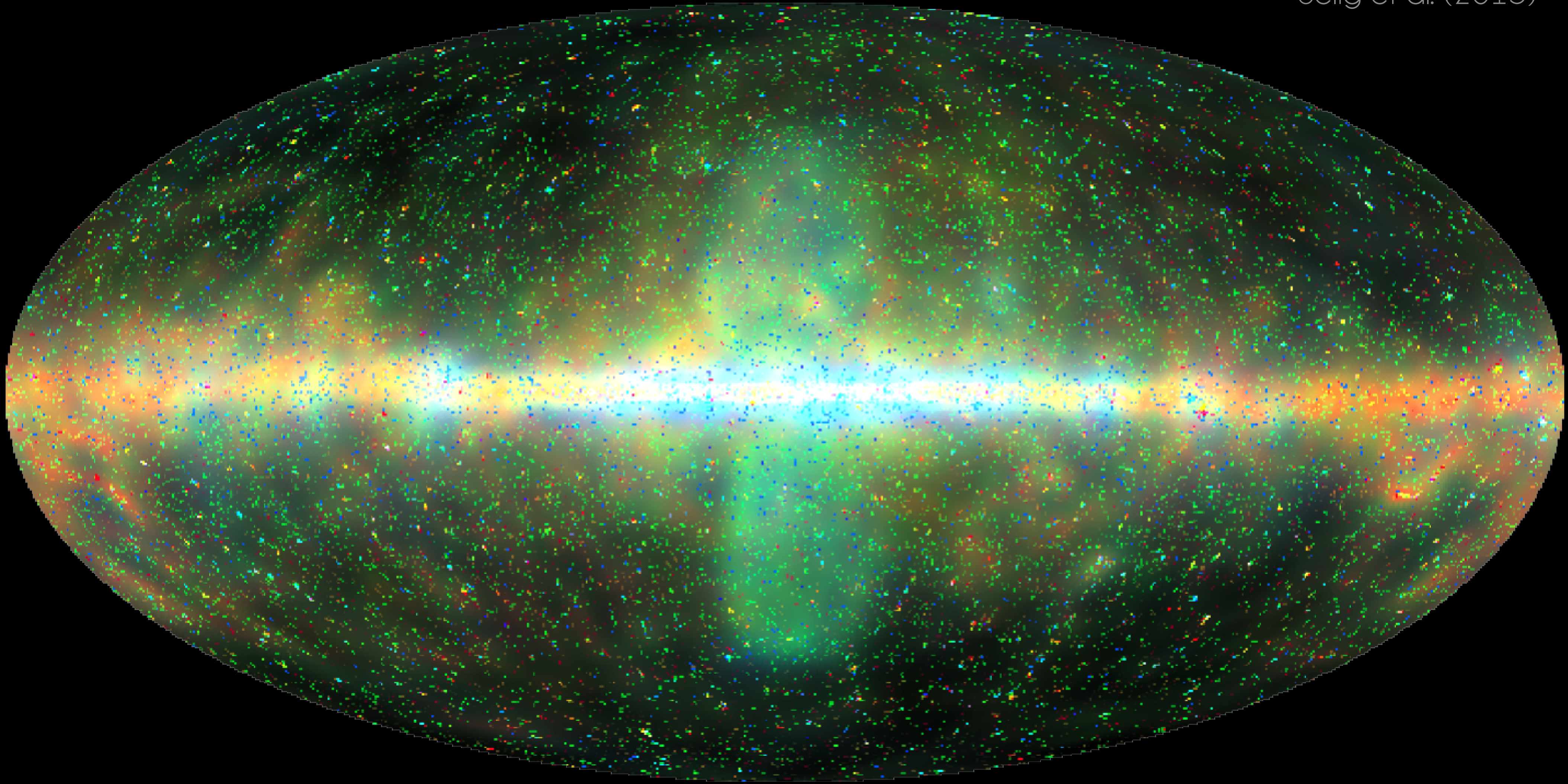

correlation structure
luminosity function
diffuse emission
point sources
sky signal
telescope
data



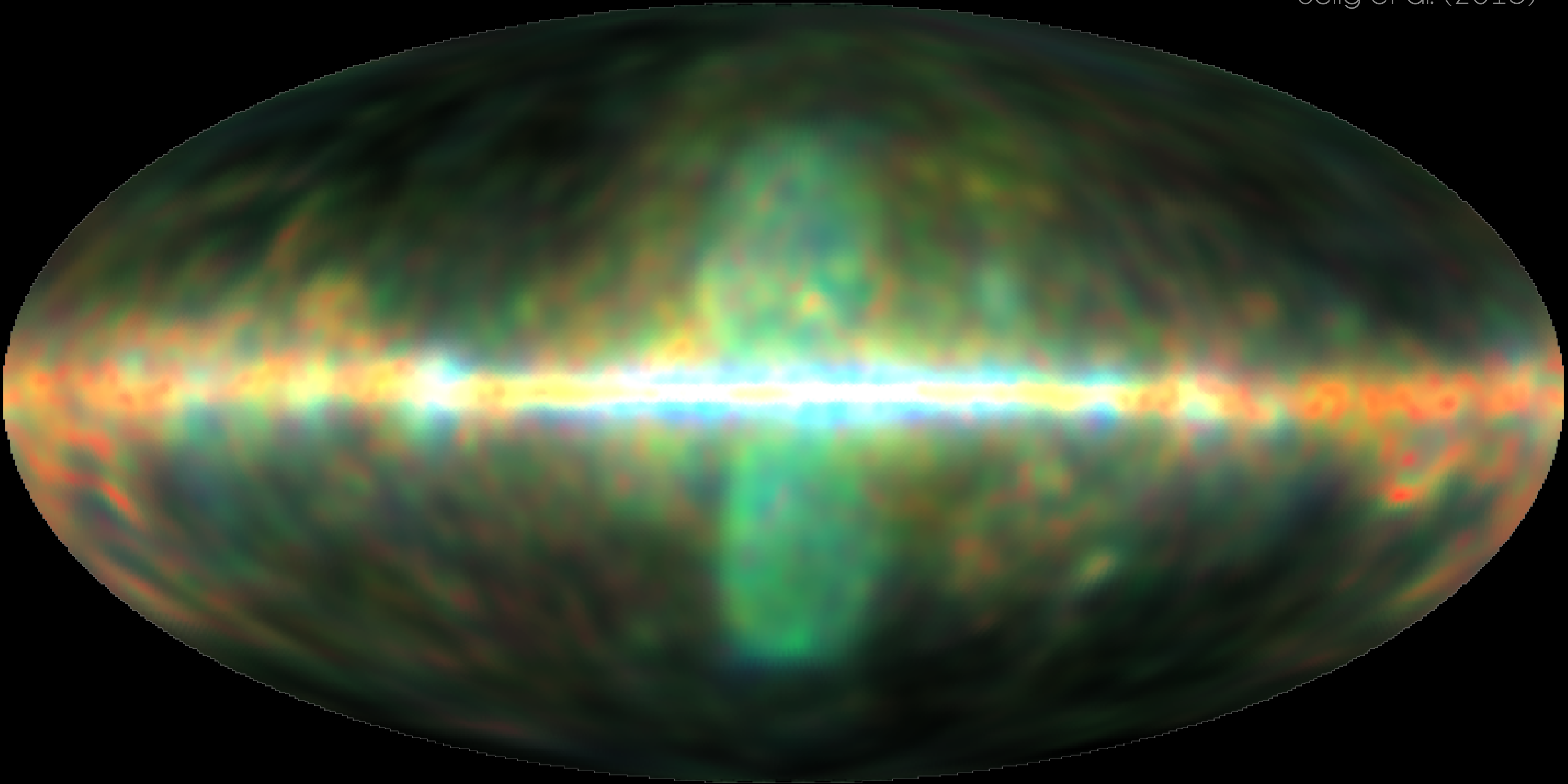
Selig et al. (2015)



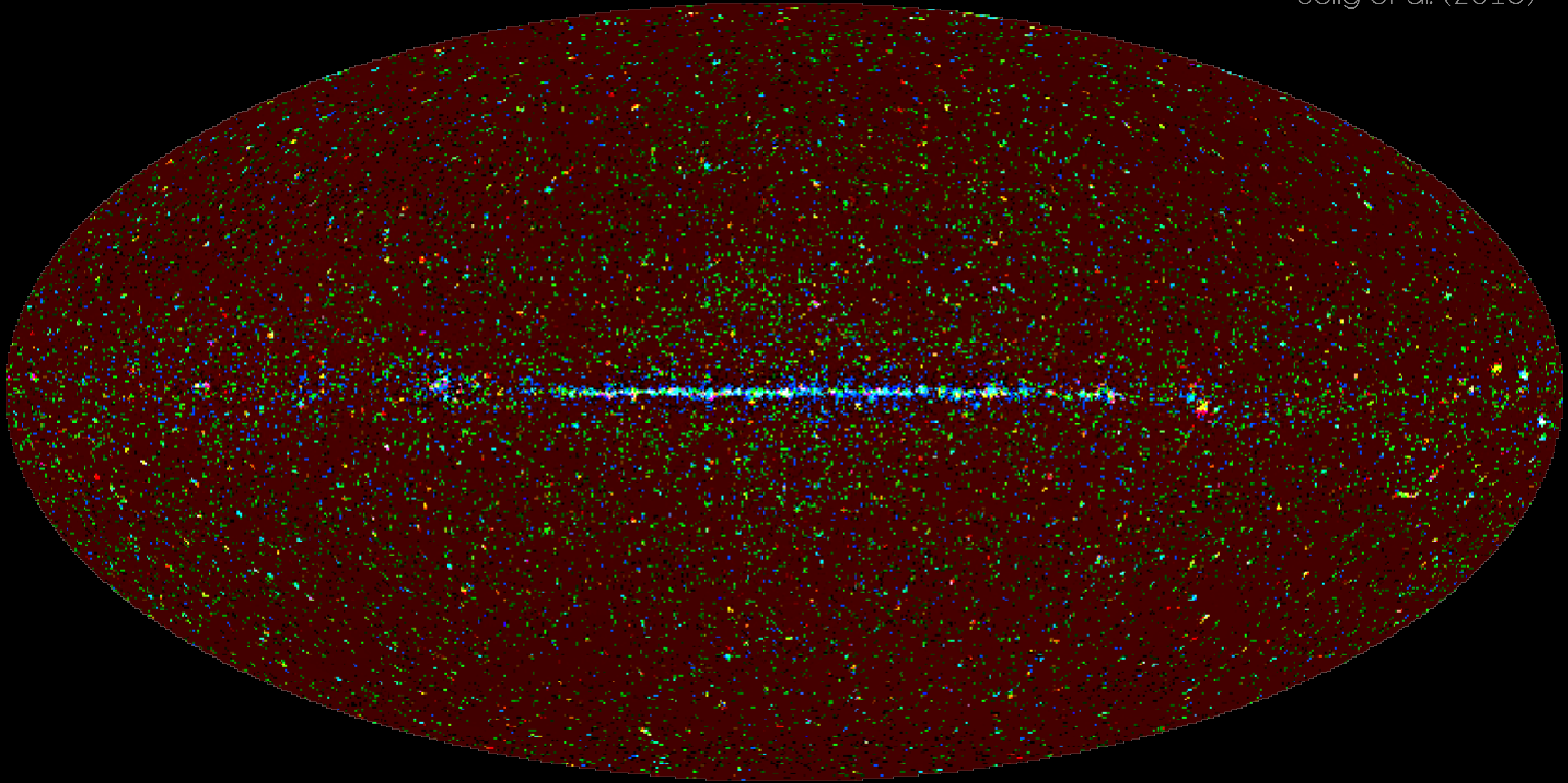
Selig et al. (2015)



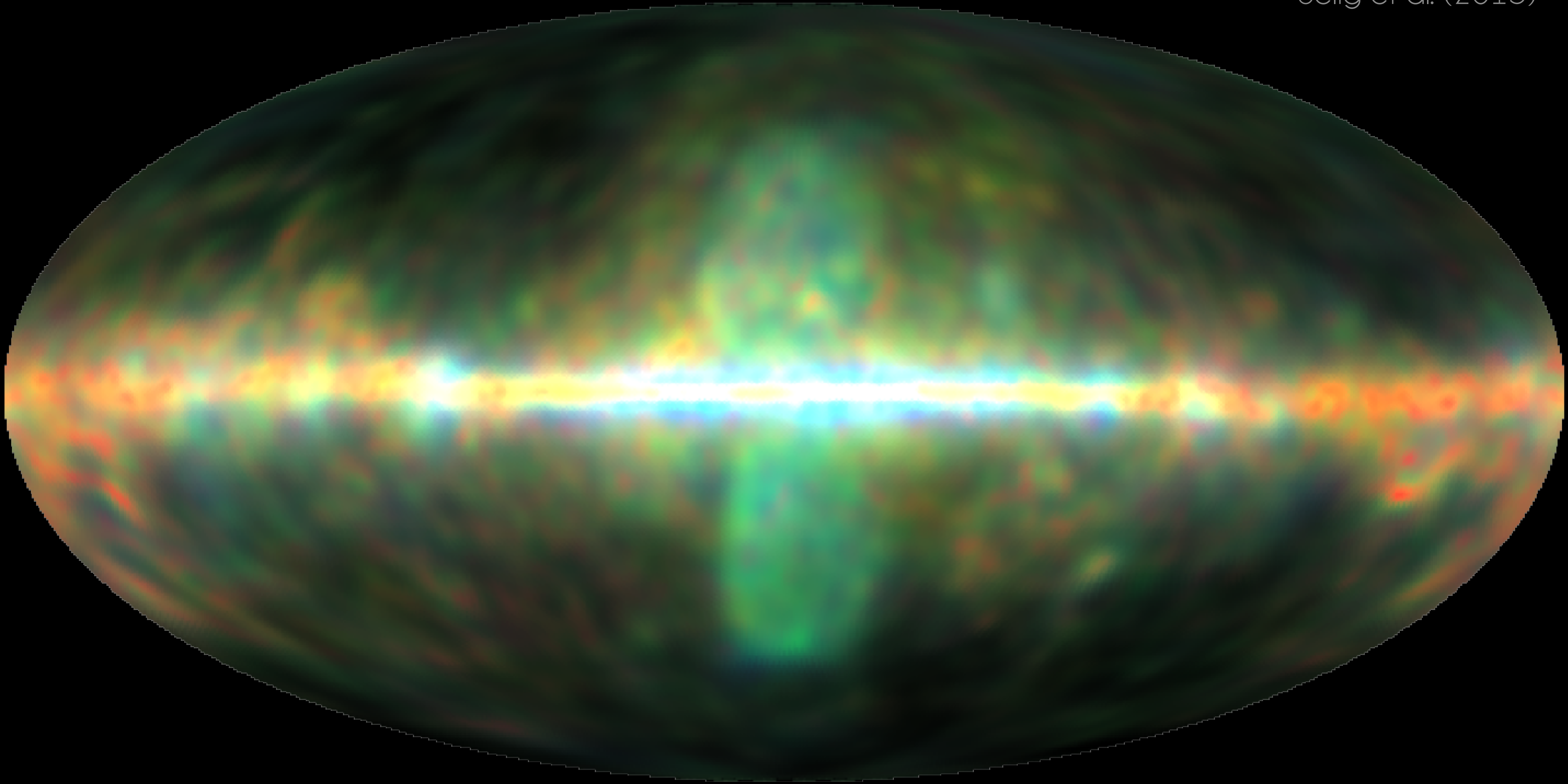
Selig et al. (2015)



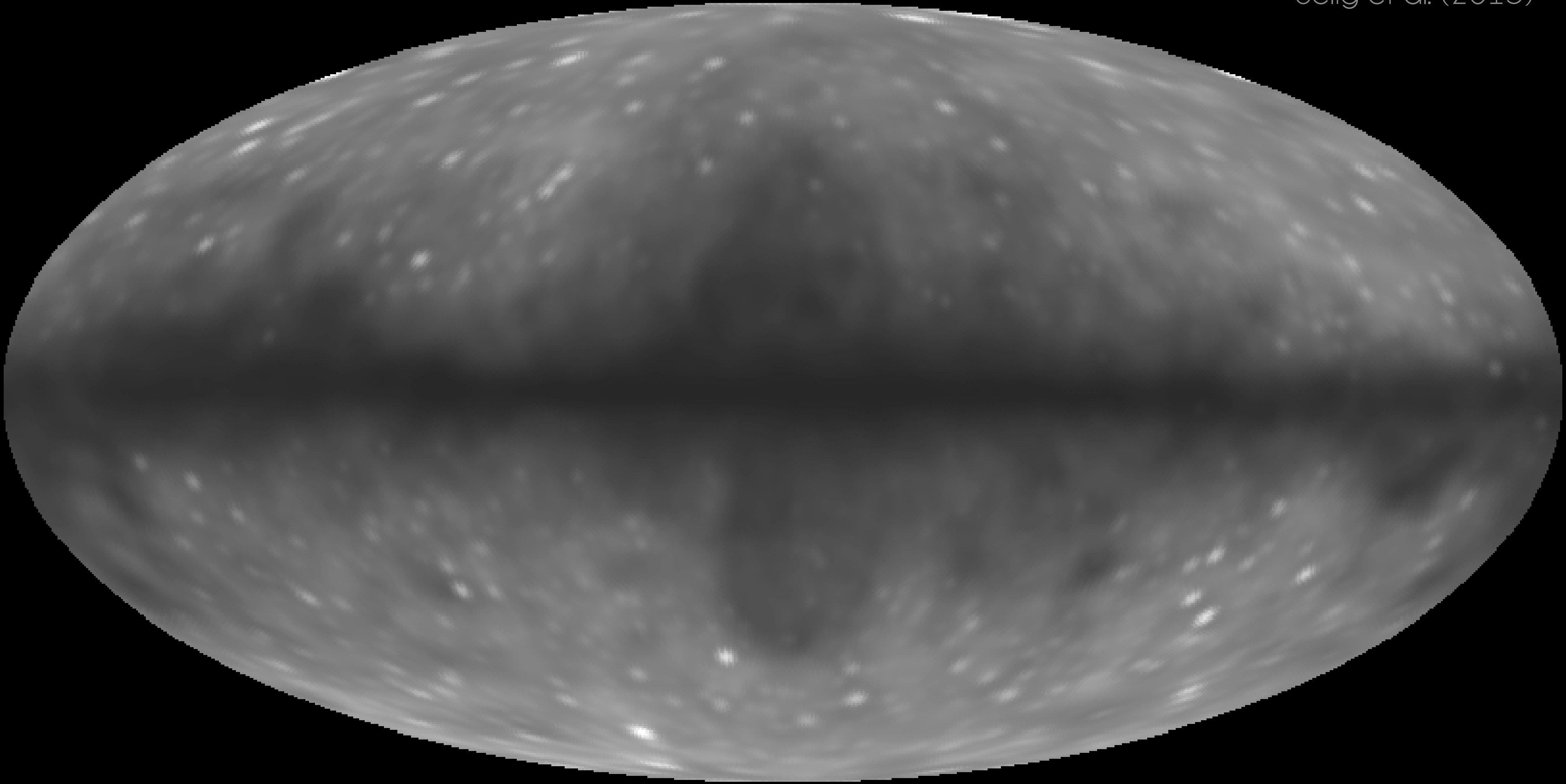
Selig et al. (2015)



Selig et al. (2015)



Selig et al. (2015)


Selig et al. (2015)

$\mathcal{P}(d|s)$ Data model

known → $d = R\,e^{s} + n$

unknown → $\lambda = R\,e^{s}$

$\mathcal{P}(s) = \mathcal{G}(s,\, S)$ unknown

$$\mathcal{P}(d|\lambda) = \prod_i \frac{\lambda_i^{d_i}}{d_i!} e^{-\lambda_i}$$

Information

$$\mathcal{H}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau}) = -\log \mathcal{P}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau})$$

$$= 1^\dagger \left[\log(d!) + \boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right] - \boldsymbol{d}^\dagger \log\left[\boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right]$$

$$+ \frac{1}{2}\boldsymbol{s}^\dagger \boldsymbol{S}^{-1}\boldsymbol{s} + \frac{1}{2}\log\left(\det\left[\boldsymbol{S}\right]\right)$$

$$+ (\boldsymbol{\alpha} - \mathbf{1})^\dagger \ldots$$

$$+ (\boldsymbol{\beta} - \mathbf{1})^\dagger \ldots$$

$$\boldsymbol{S} = \sum_k \mathrm{e}^{\tau_k} \ldots$$

- Convert into **generative model**
- Compress information into Gaussian via **Metric Gaussian Variational Inference**

Variational Bayes

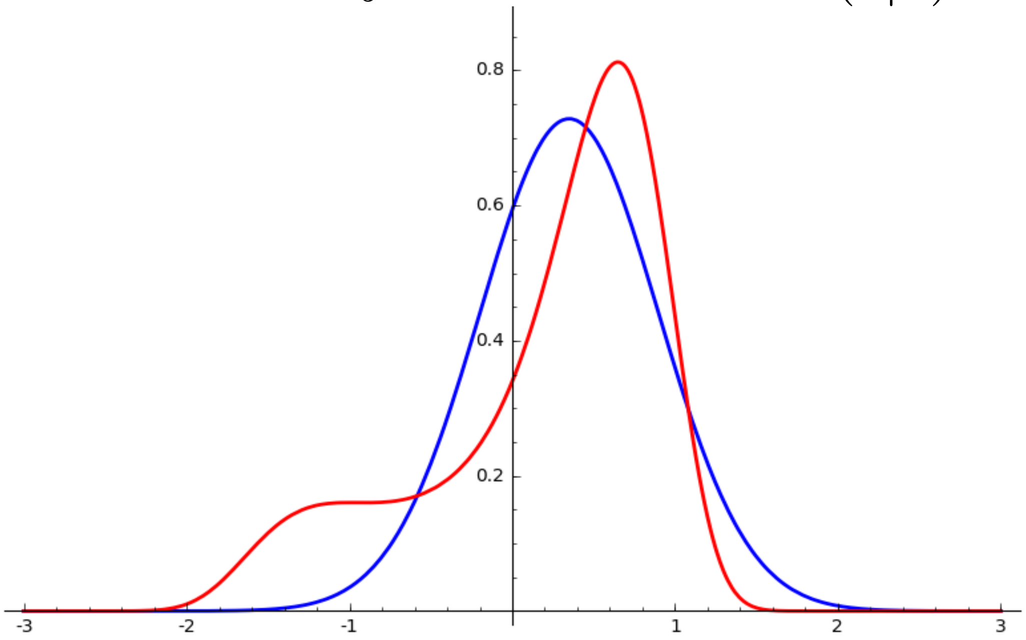
$\mathcal{P}(s|d)$

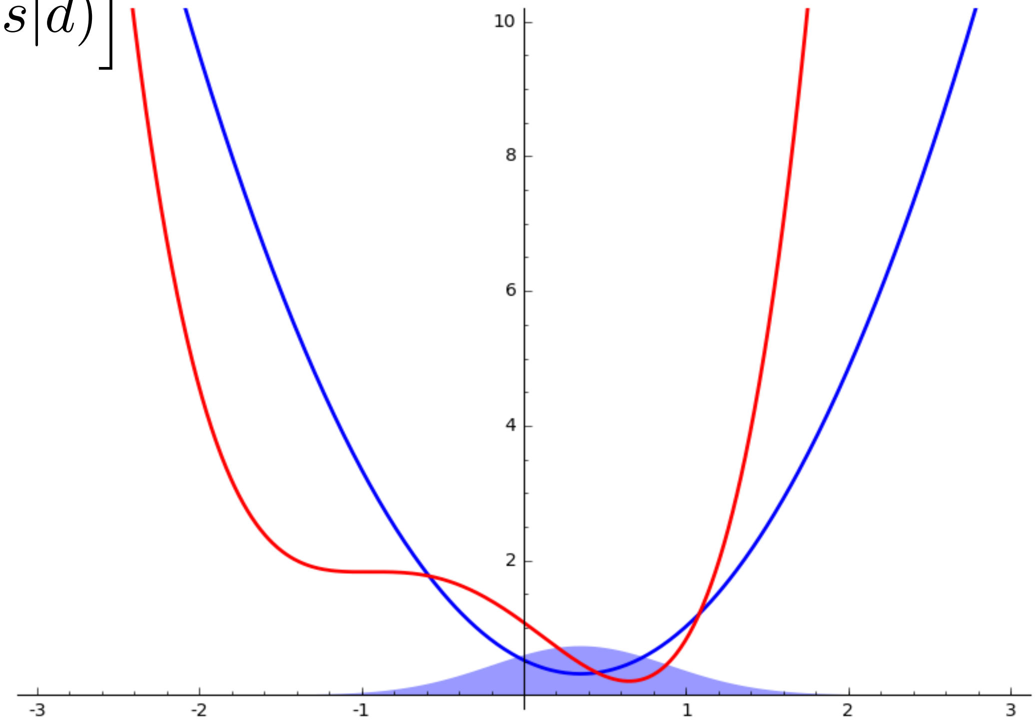
$\widetilde{\mathcal{P}}(s|d) = \mathcal{G}(s-m, D)$

$\mathcal{H}(s|d)$

$\widetilde{\mathcal{H}}(s|d) \widehat{=} \frac{1}{2}(s-m)^\dagger D^{-1}(s-m)$

$$\mathrm{KL}(\widetilde{\mathcal{P}}, \mathcal{P}) = \int \mathcal{D}s\, \widetilde{\mathcal{P}}(s|d) \left[\mathcal{H}(s|d) - \widetilde{\mathcal{H}}(s|d)\right]$$





Metric Gaussian Variational Bayes

$\mathcal{P}(s|d)$

$\widetilde{\mathcal{P}}(s|d) = \mathcal{G}(s-m, D)$

$\mathcal{H}(s|d)$

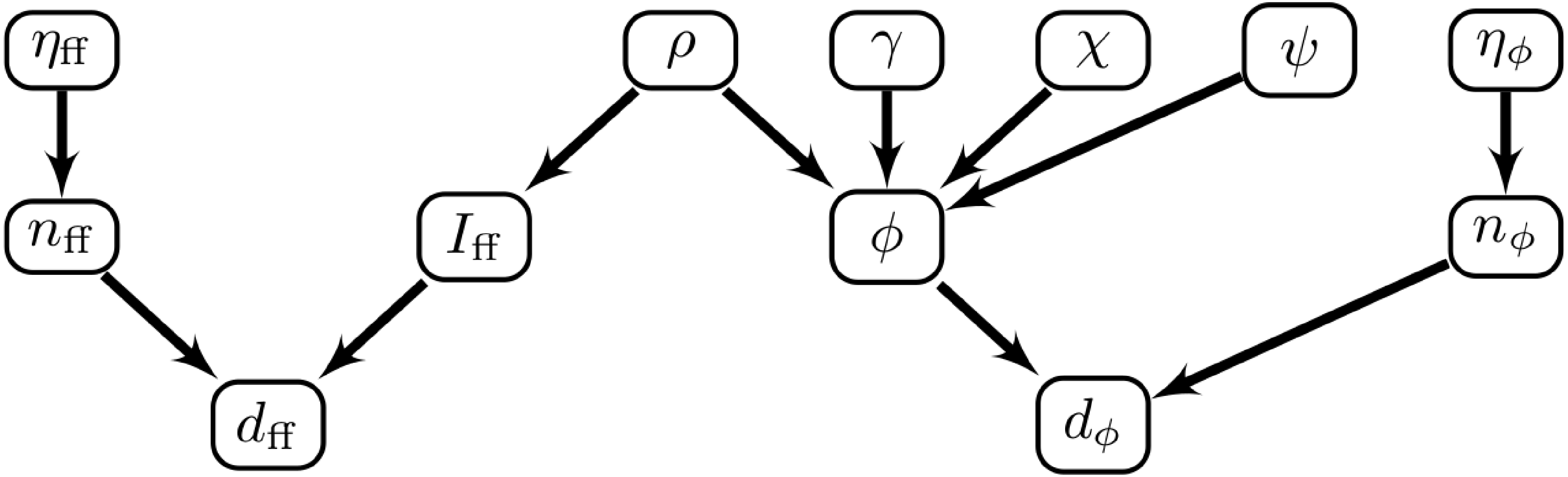
$\widetilde{\mathcal{H}}(s|d) \widehat{=} \frac{1}{2}(s-m)^\dagger D^{-1}(s-m)$

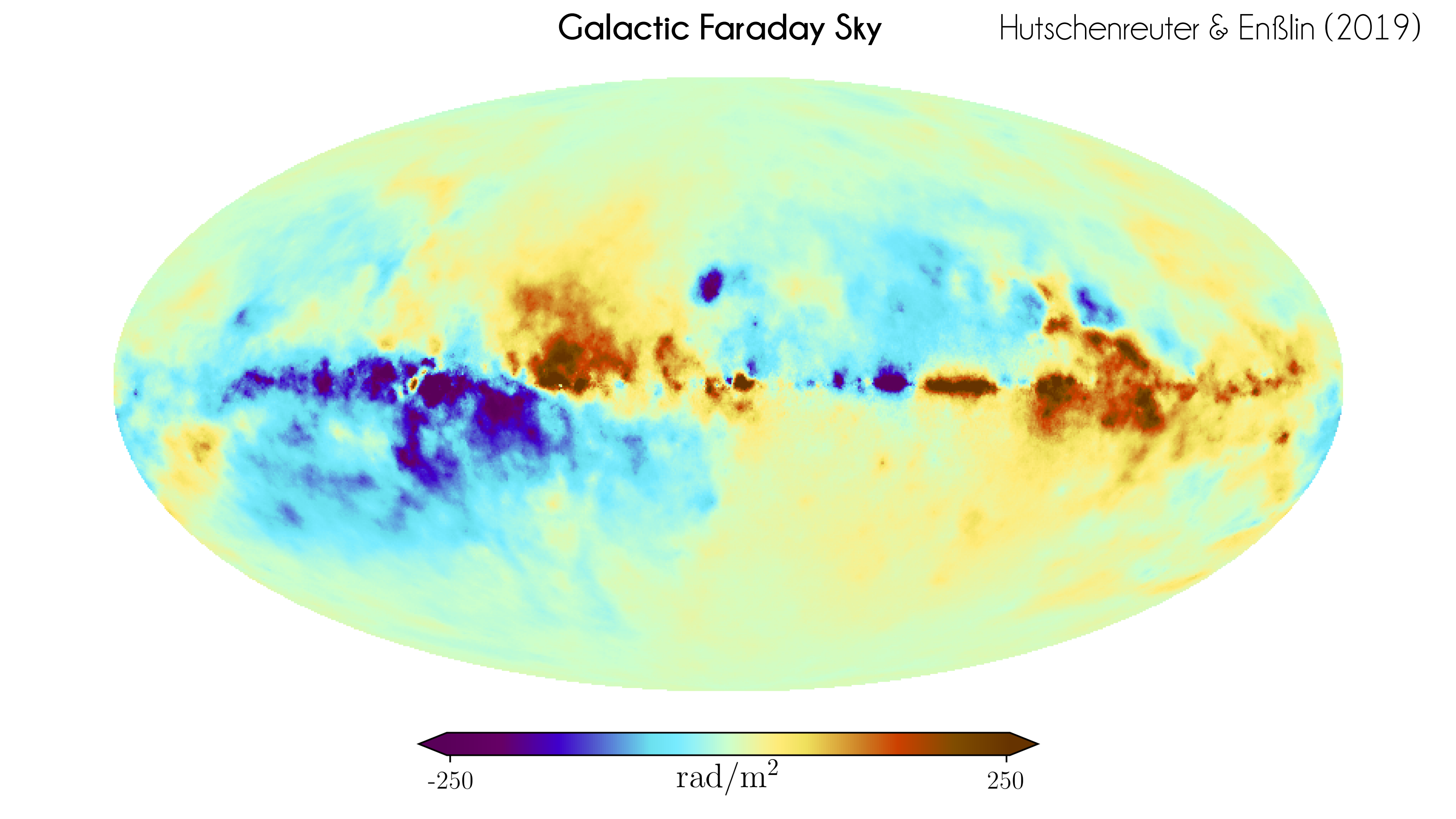
Knollmüller & Enßlin (2019)

$$\mathrm{KL}(\widetilde{\mathcal{P}}, \mathcal{P}) = \int \mathcal{D}s\, \widetilde{\mathcal{P}}(s|d) \left[\mathcal{H}(s|d) - \widetilde{\mathcal{H}}(s|d)\right]$$

$$D \approx \left\langle \frac{\partial \mathcal{H}(d,s)}{\partial s} \frac{\partial \mathcal{H}(d,s)}{\partial s}^\dagger \right\rangle_{(d|s=m)}^{-1}$$

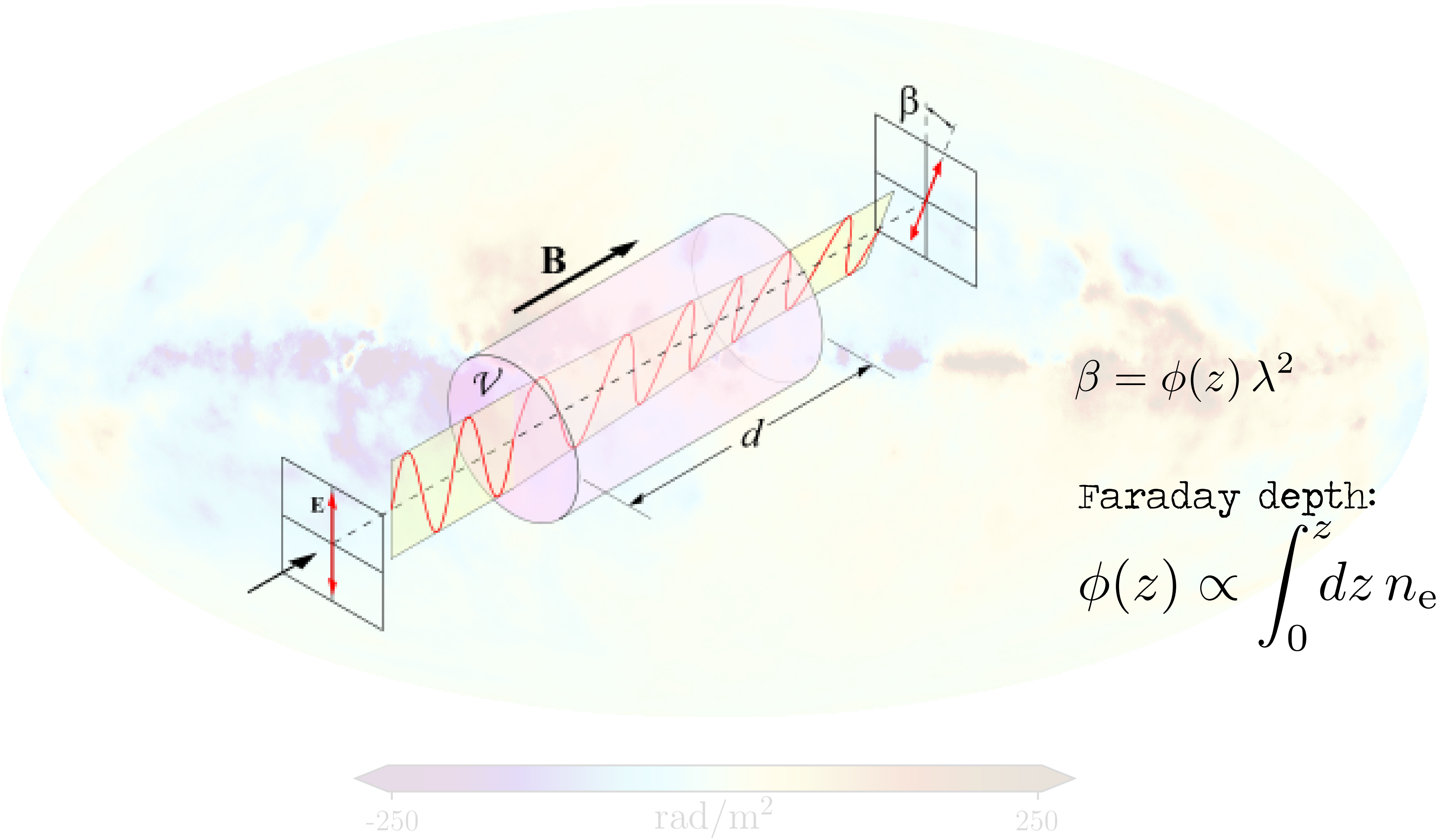
Hierarchical Bayesian Model





Galactic Faraday Sky
Hutschenreuter & Enßlin (2019)
-250
rad/m²
250

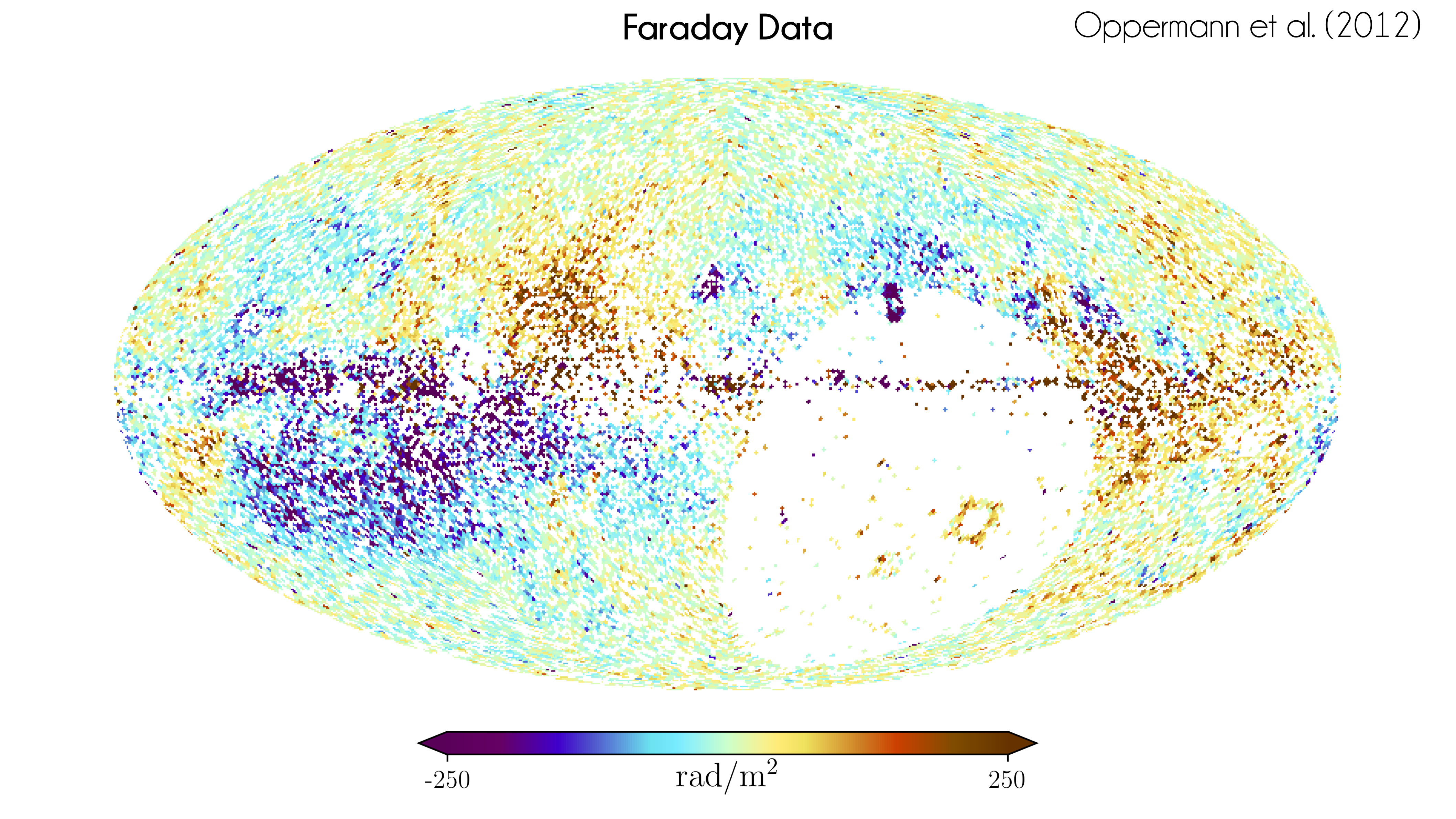
Faraday Effect



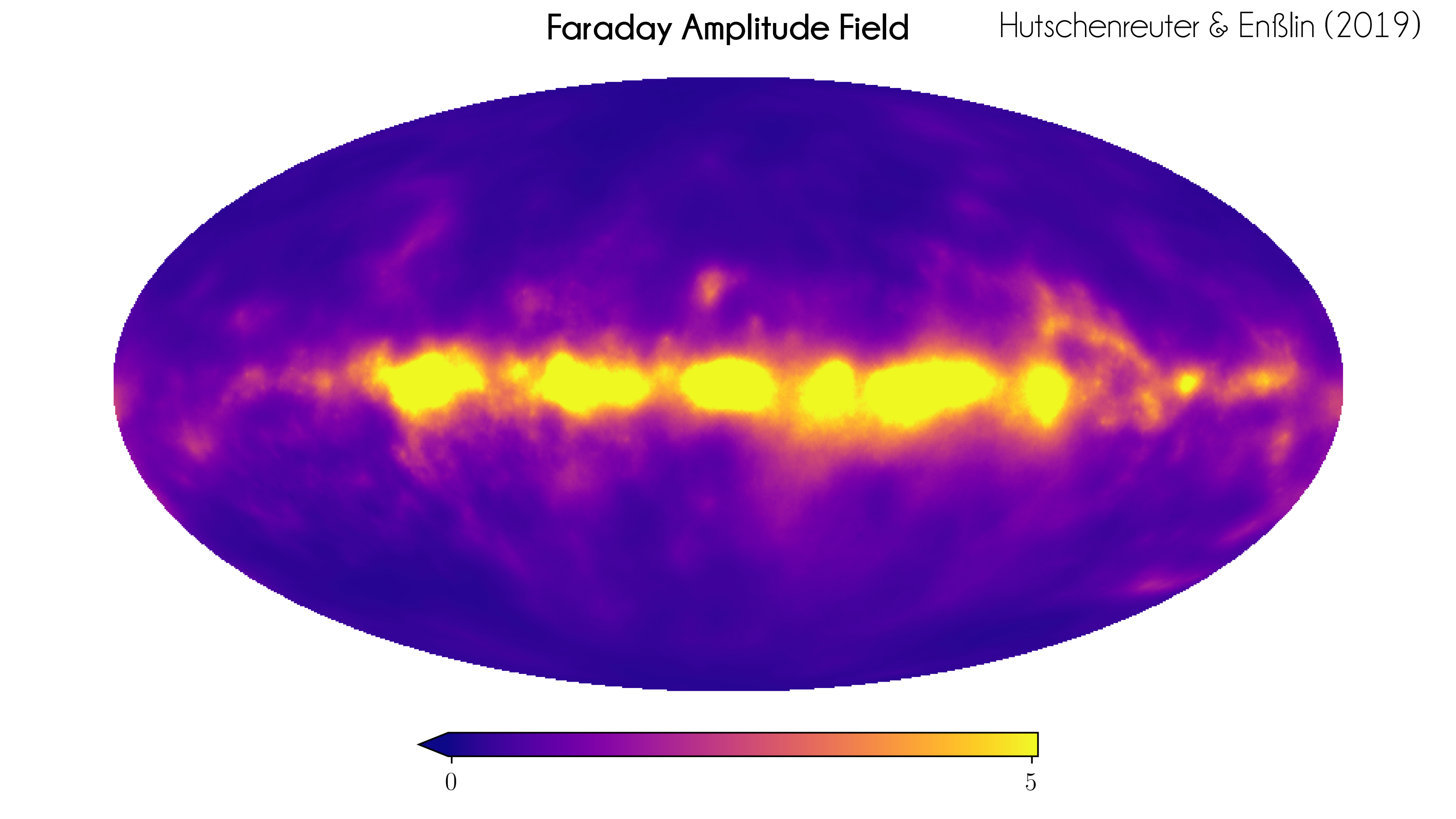
$$\beta = \phi(z)\,\lambda^2$$

Faraday depth:

$$\phi(z) \propto \int_0^z dz\, n_\mathrm{e}\, B_z$$



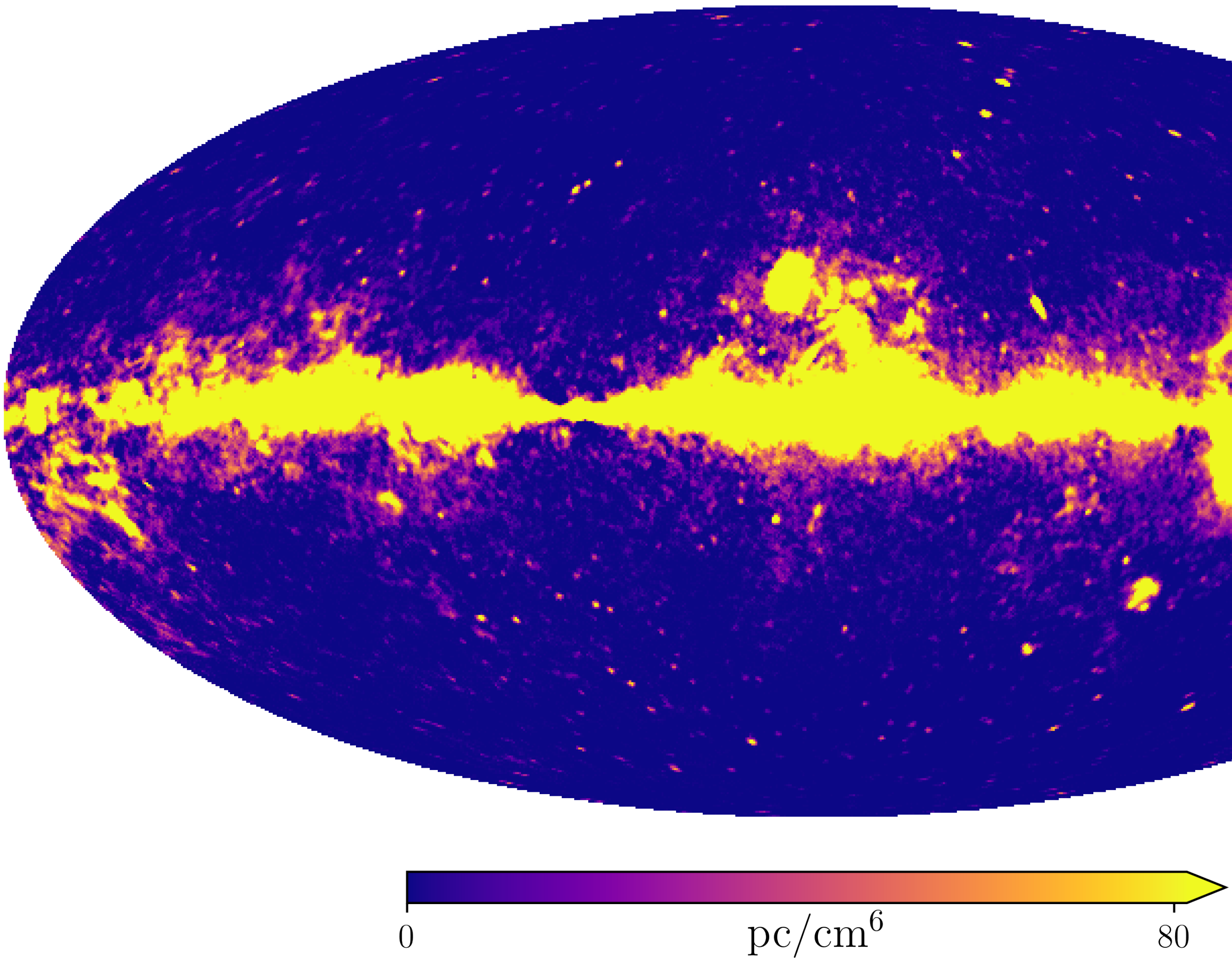
Faraday Data
Oppermann et al. (2012)
-250
rad/m^2
250



Faraday Amplitude Field
Hutschenreuter & Enßlin (2019)
0
5

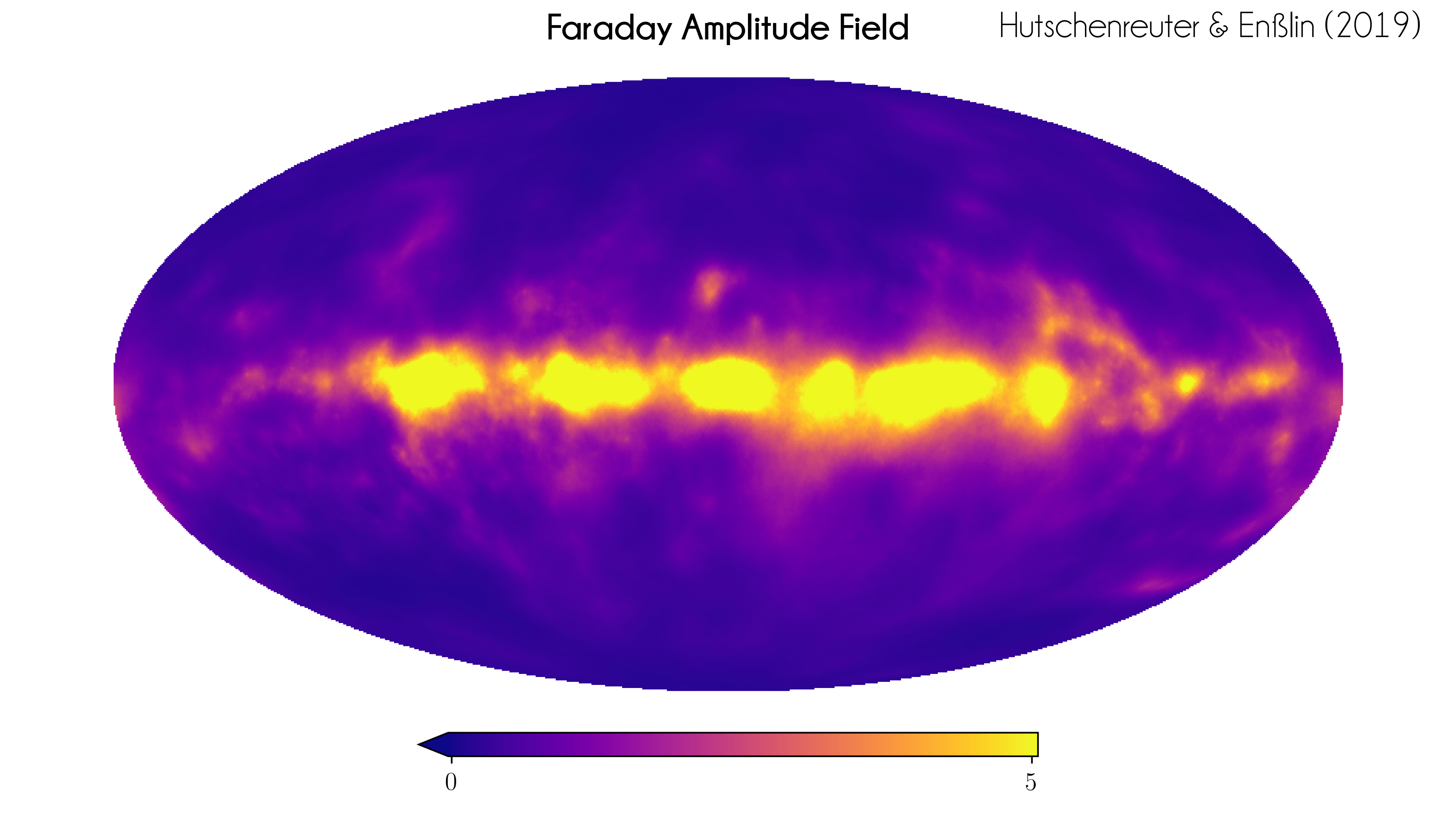
Planck Free-Free Emission

Hutschenreuter & Enßlin (2019)



Faraday depth:

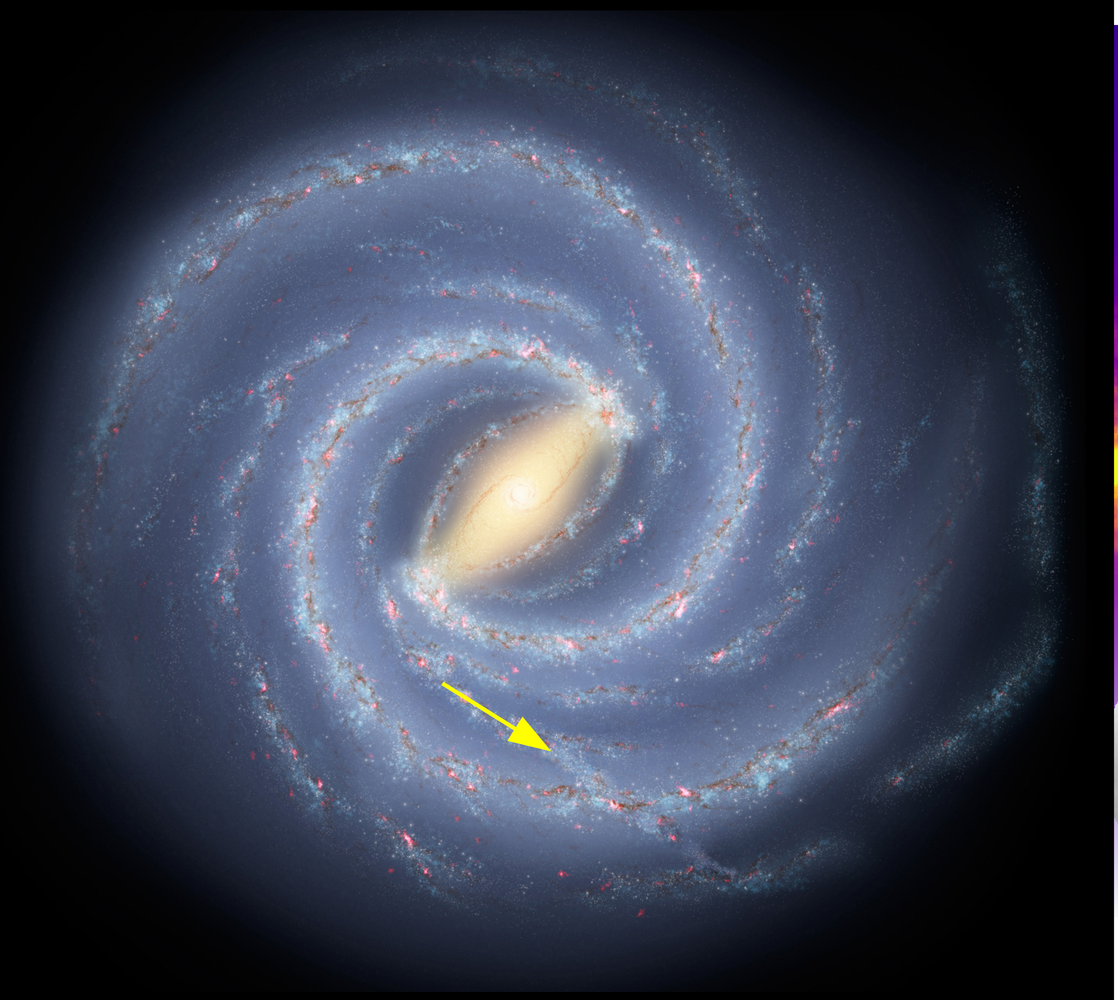
$$\phi(z) \propto \int_0^z dz\, n_e B_z$$

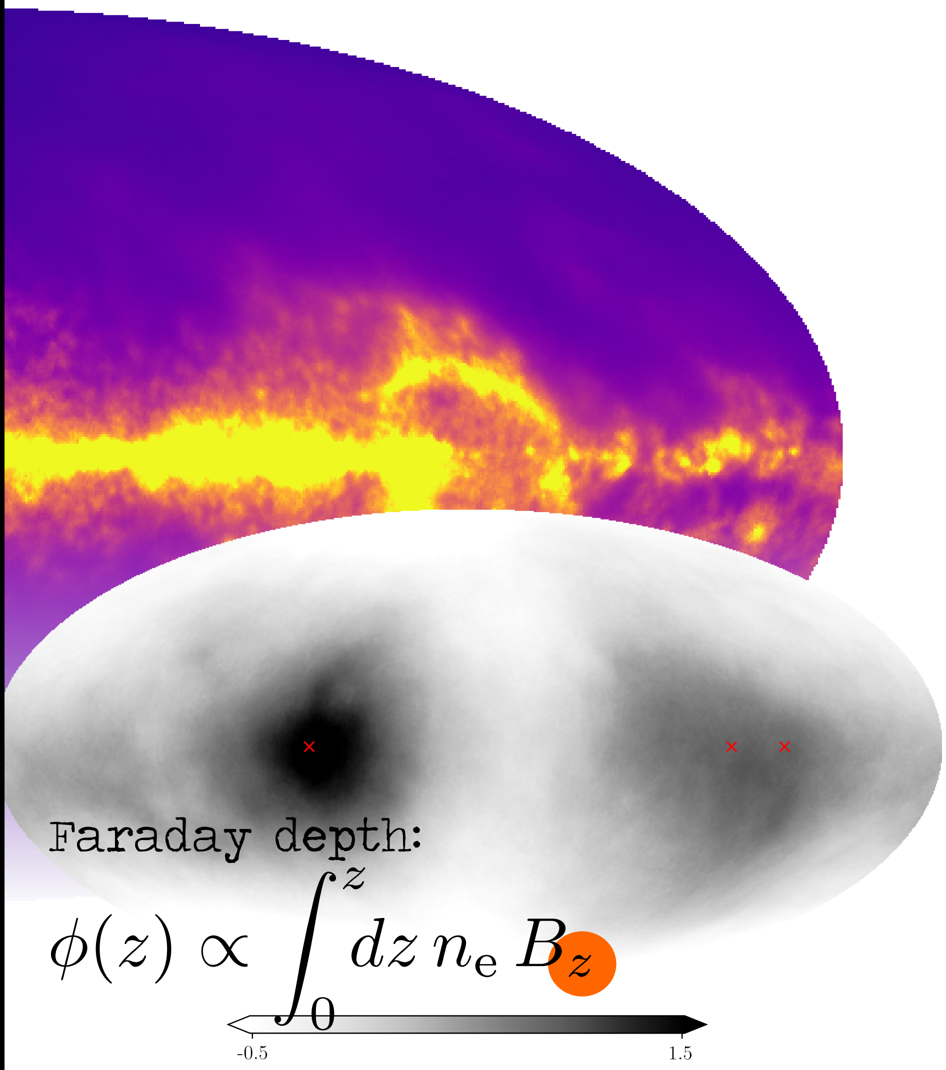


Faraday Amplitude Field
Hutschenreuter & Enßlin (2019)
0
5

tude Field

Hutschenreuter & Enßlin (2019)



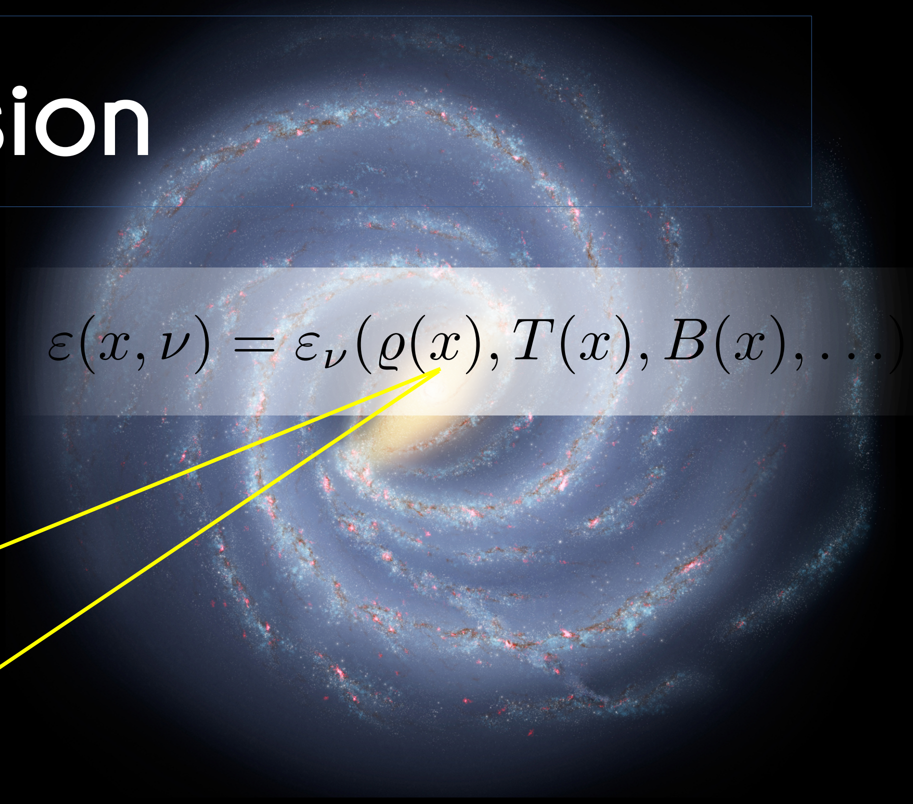


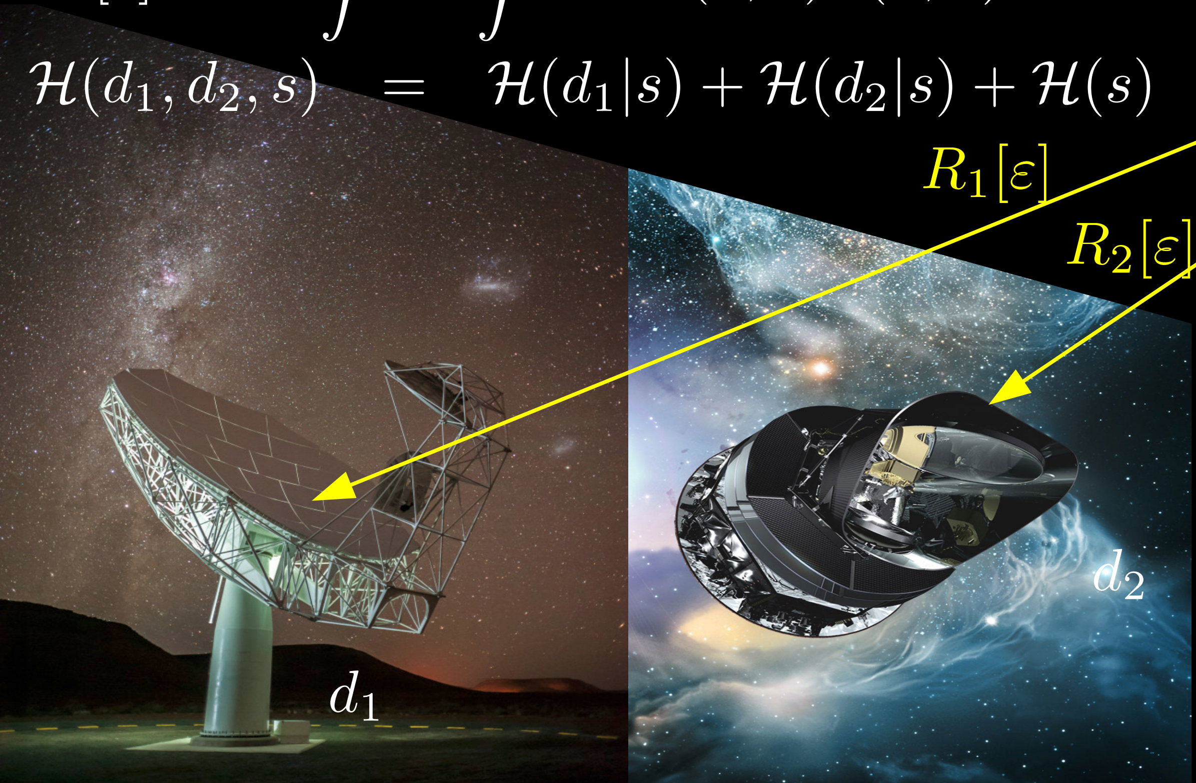
Faraday depth:

$$\phi(z) \propto \int_0^z dz\, n_\mathrm{e}\, B_z$$

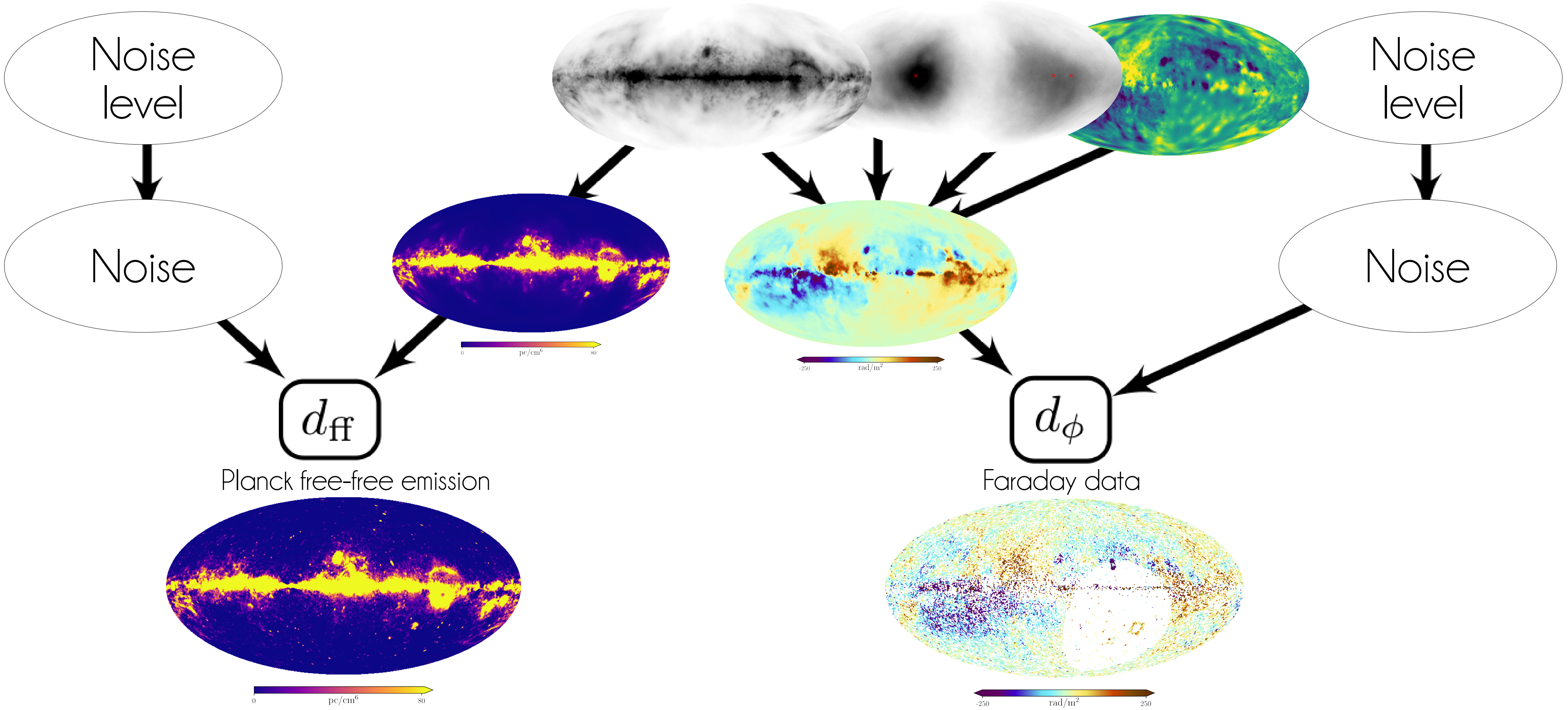
Data Fusion

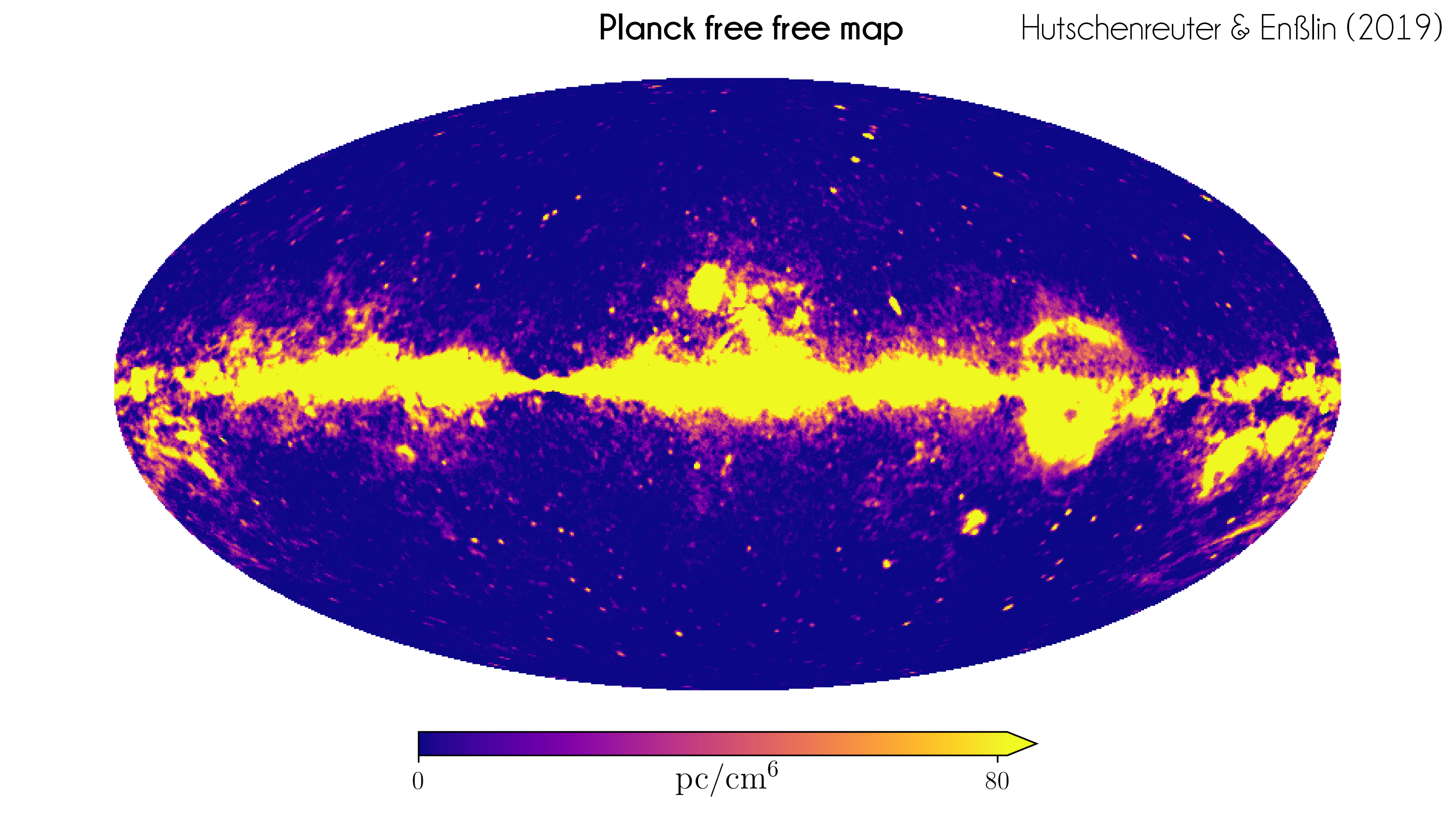
$$
\begin{aligned}
d_i &= R_i[\varepsilon] + n_i \\
R_i[\varepsilon] &= \int \mathrm{d}x \int \mathrm{d}\nu \, R_i(x,\nu)\,\varepsilon(x,\nu) \\
\mathcal{H}(d_1, d_2, s) &= \mathcal{H}(d_1|s) + \mathcal{H}(d_2|s) + \mathcal{H}(s)
\end{aligned}
$$



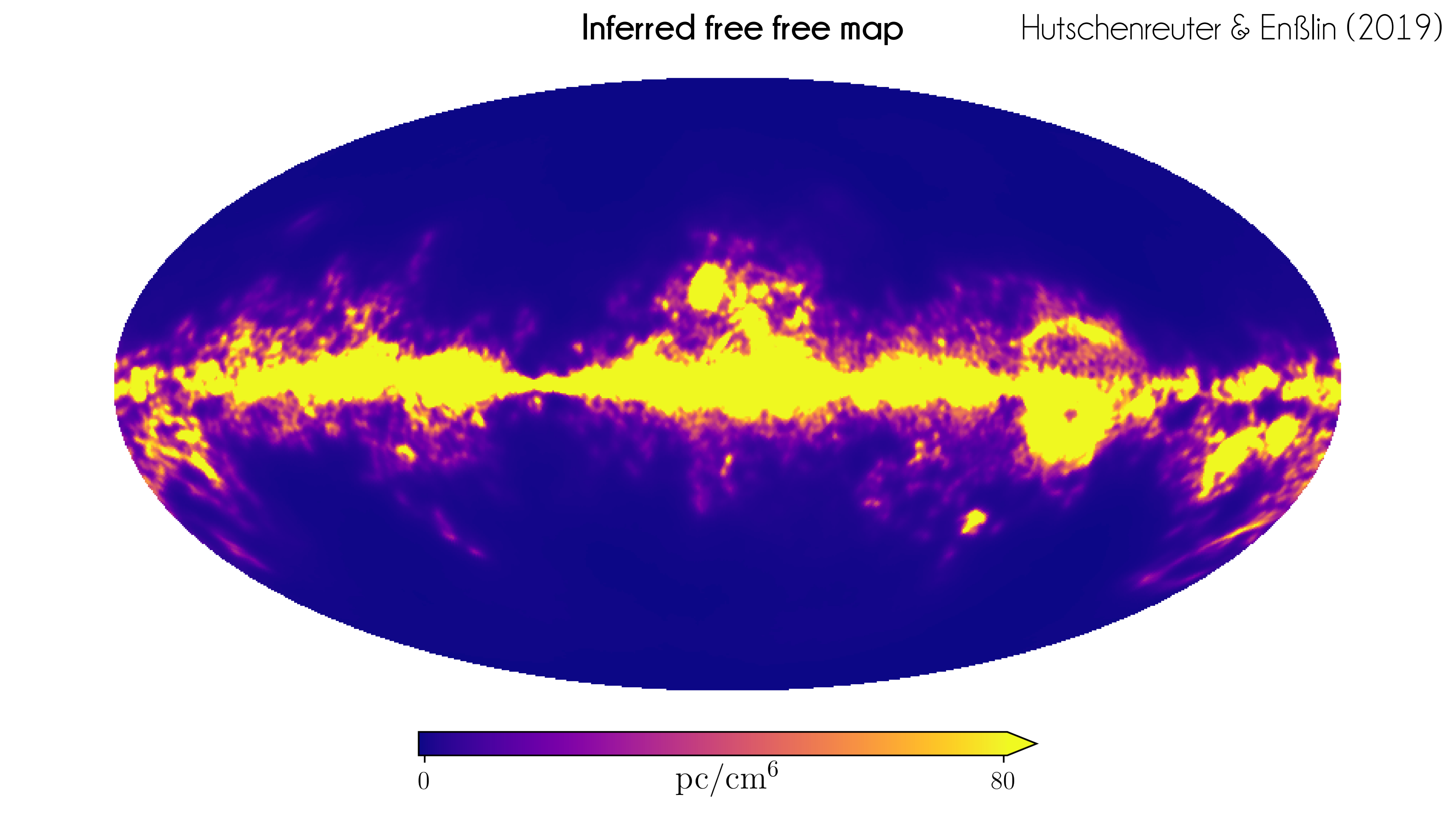


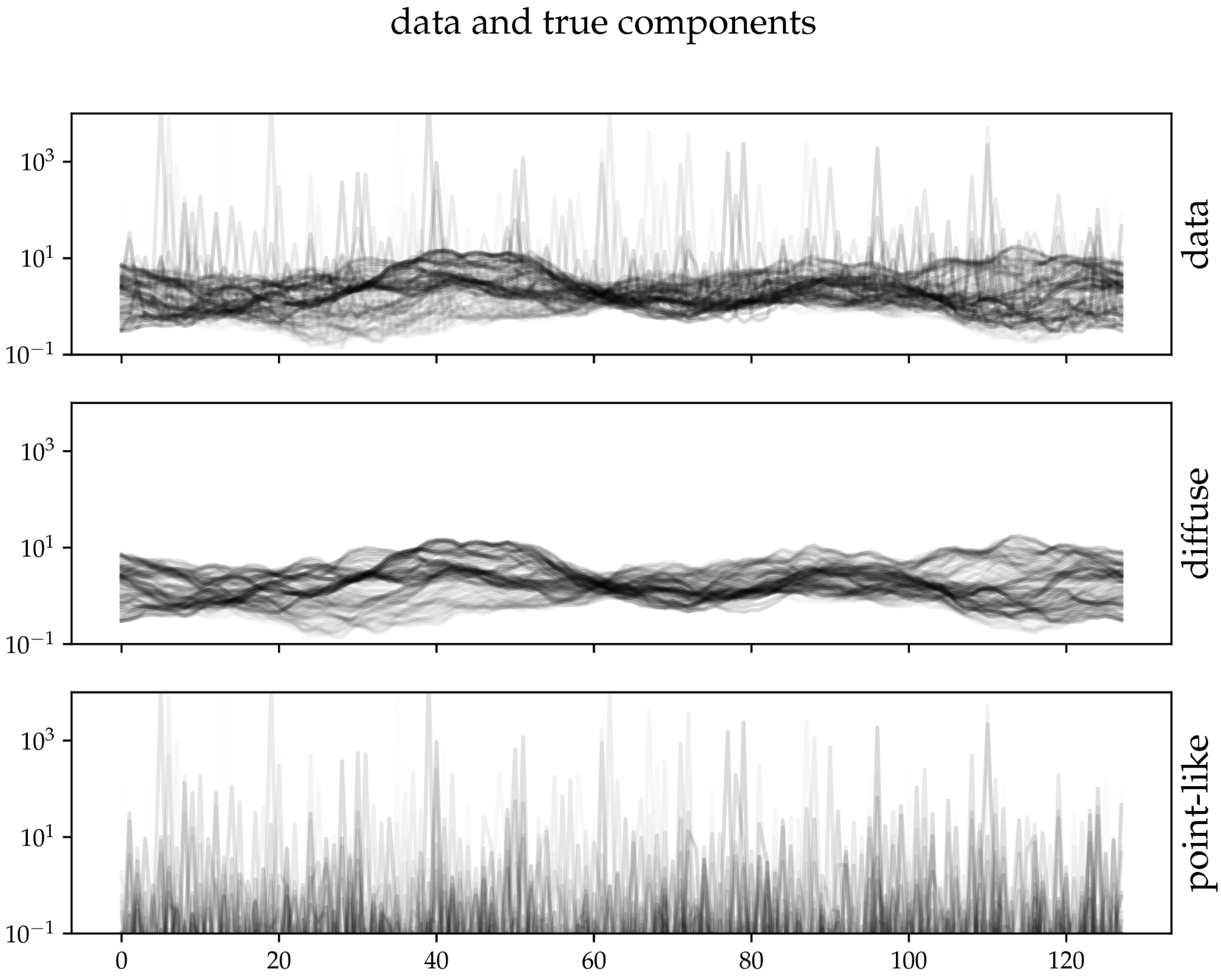
Hierarchical Bayesian Model



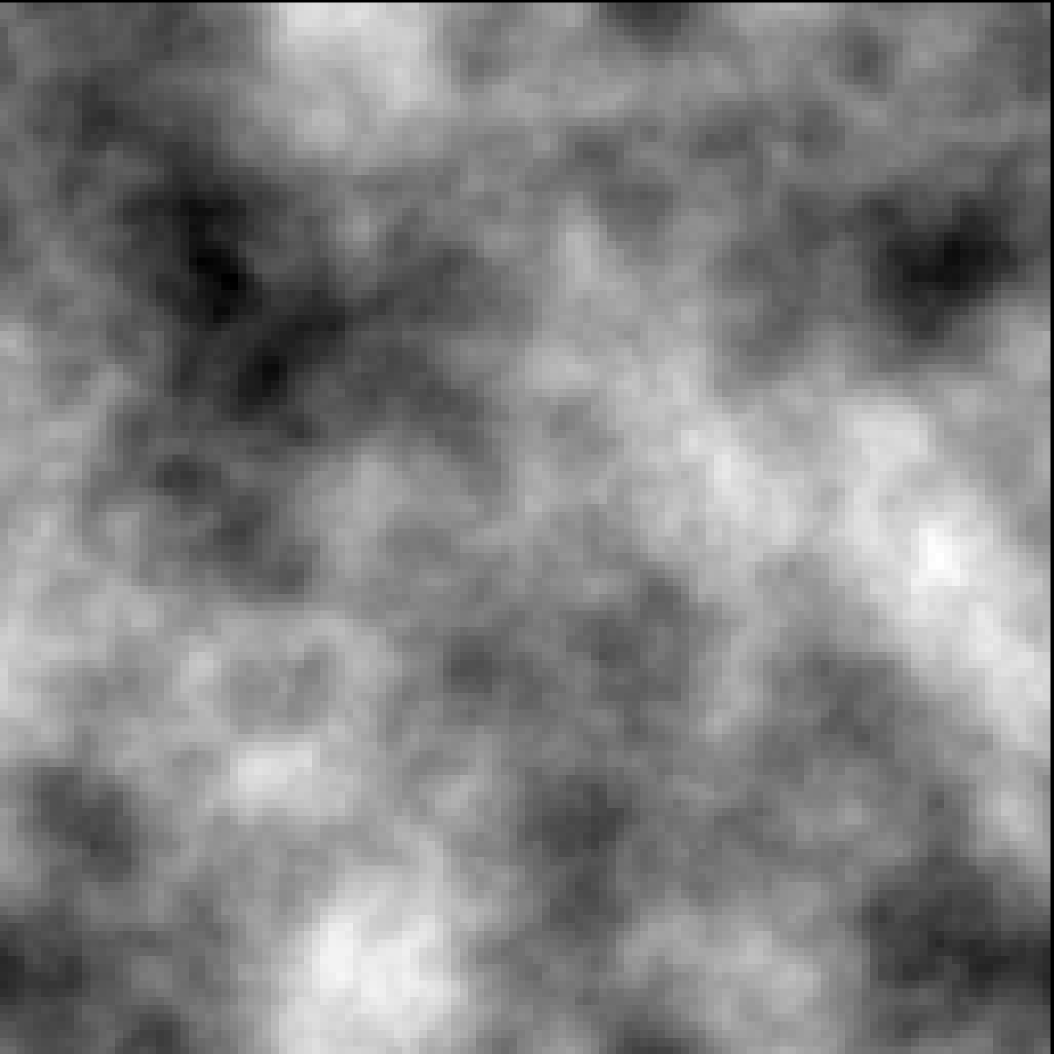


Planck free free map
Hutschenreuter & Enßlin (2019)
0
pc/cm^6
80

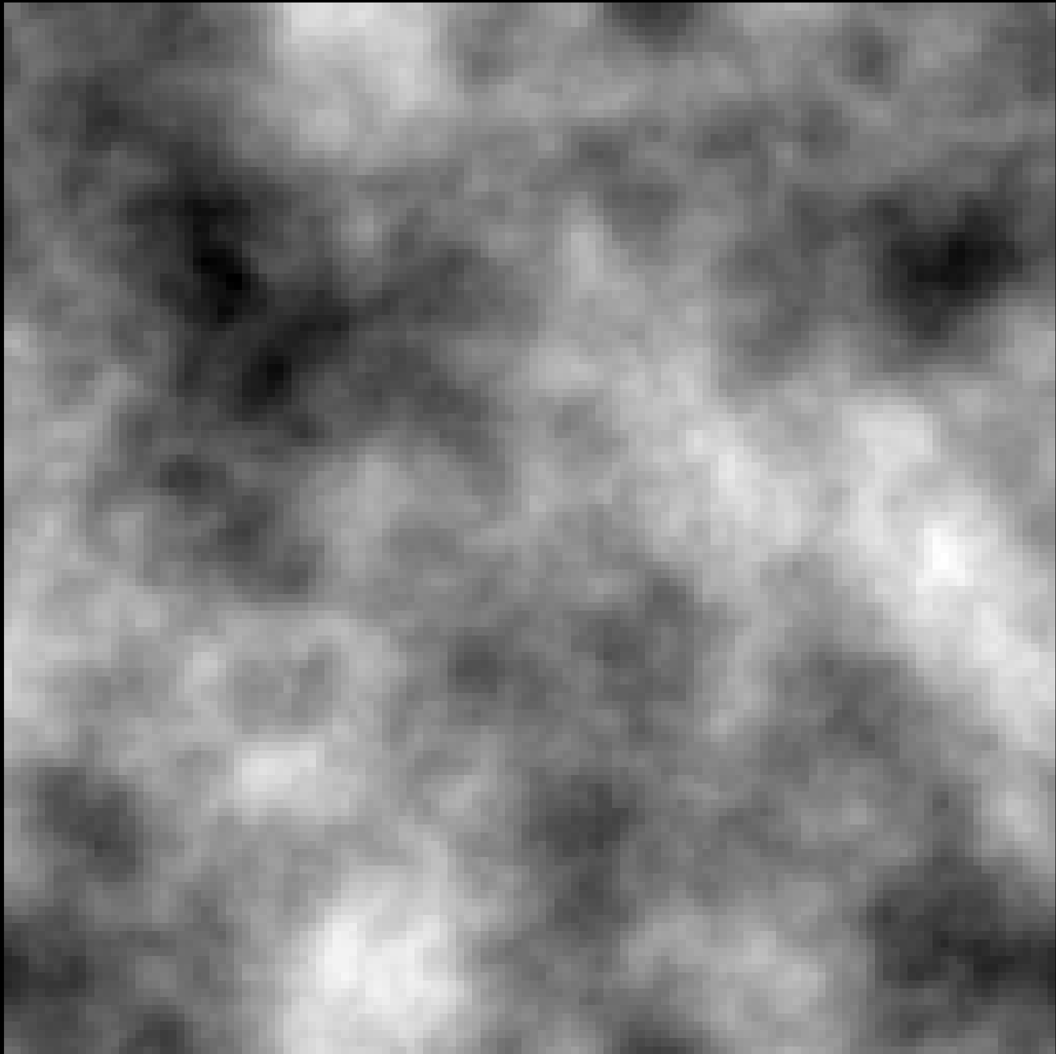

Inferred free free map
Hutschenreuter & Enßlin (2019)
0
pc/cm^6
80


data and true components
data
diffuse
point-like
10^3
10^1
10^-1
0
20
40
60
80
100
120

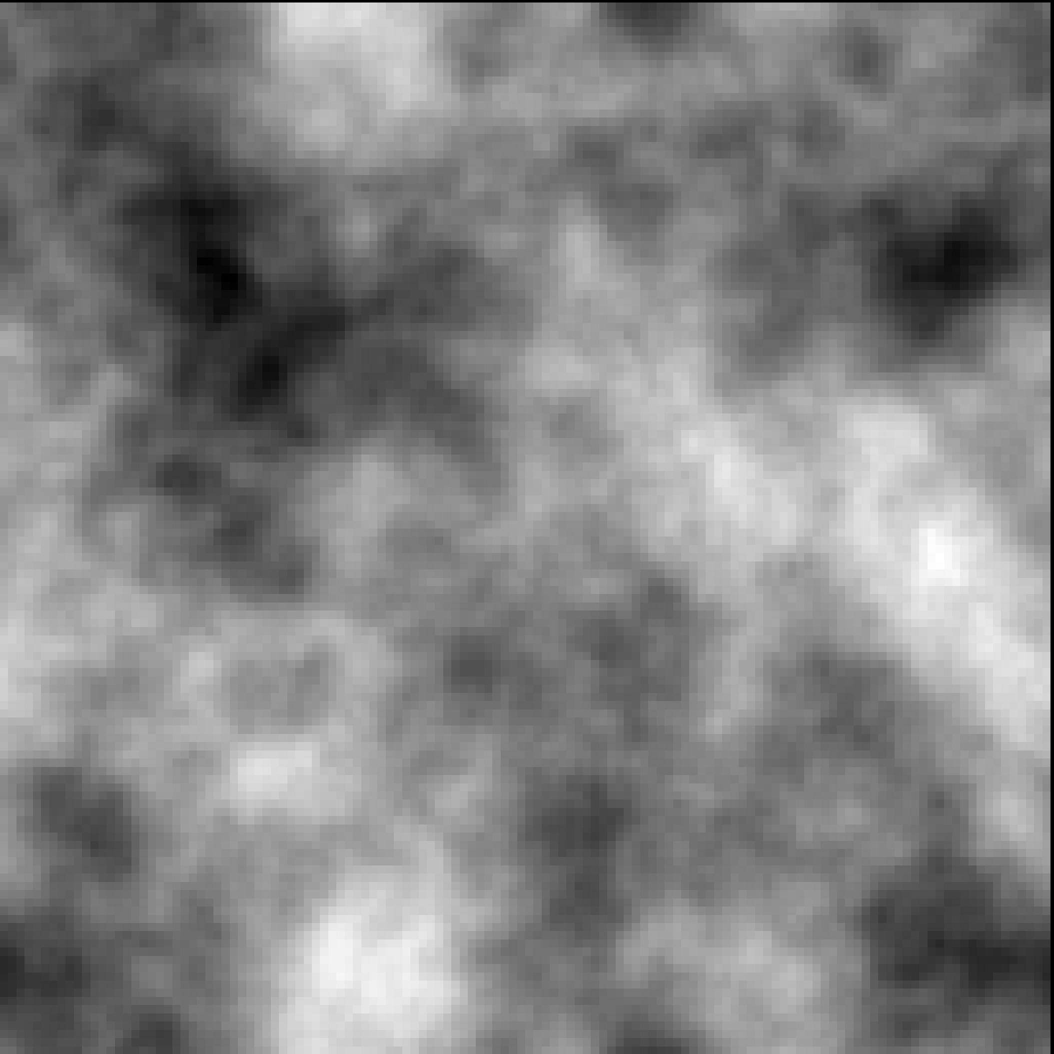
ground truth / starblade



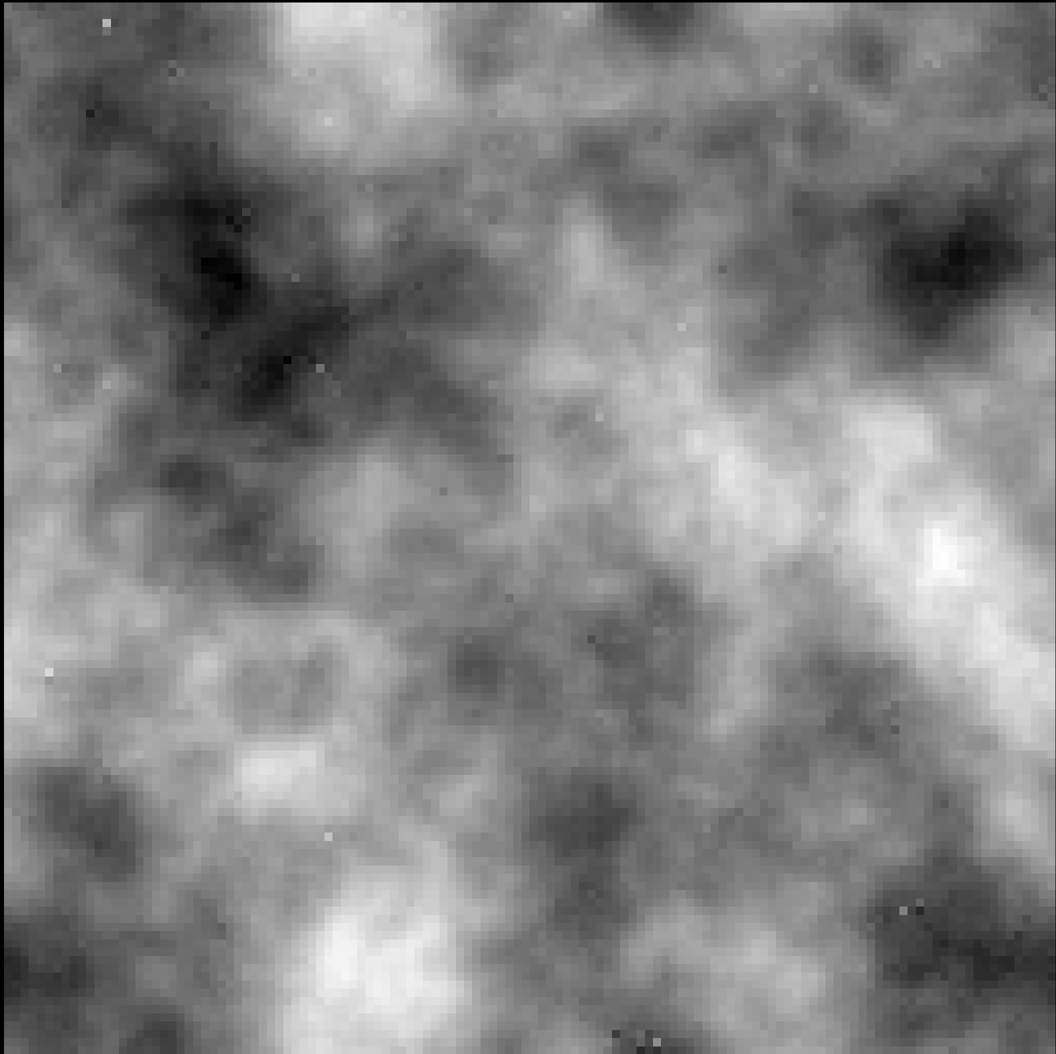
ground truth / autoencoder

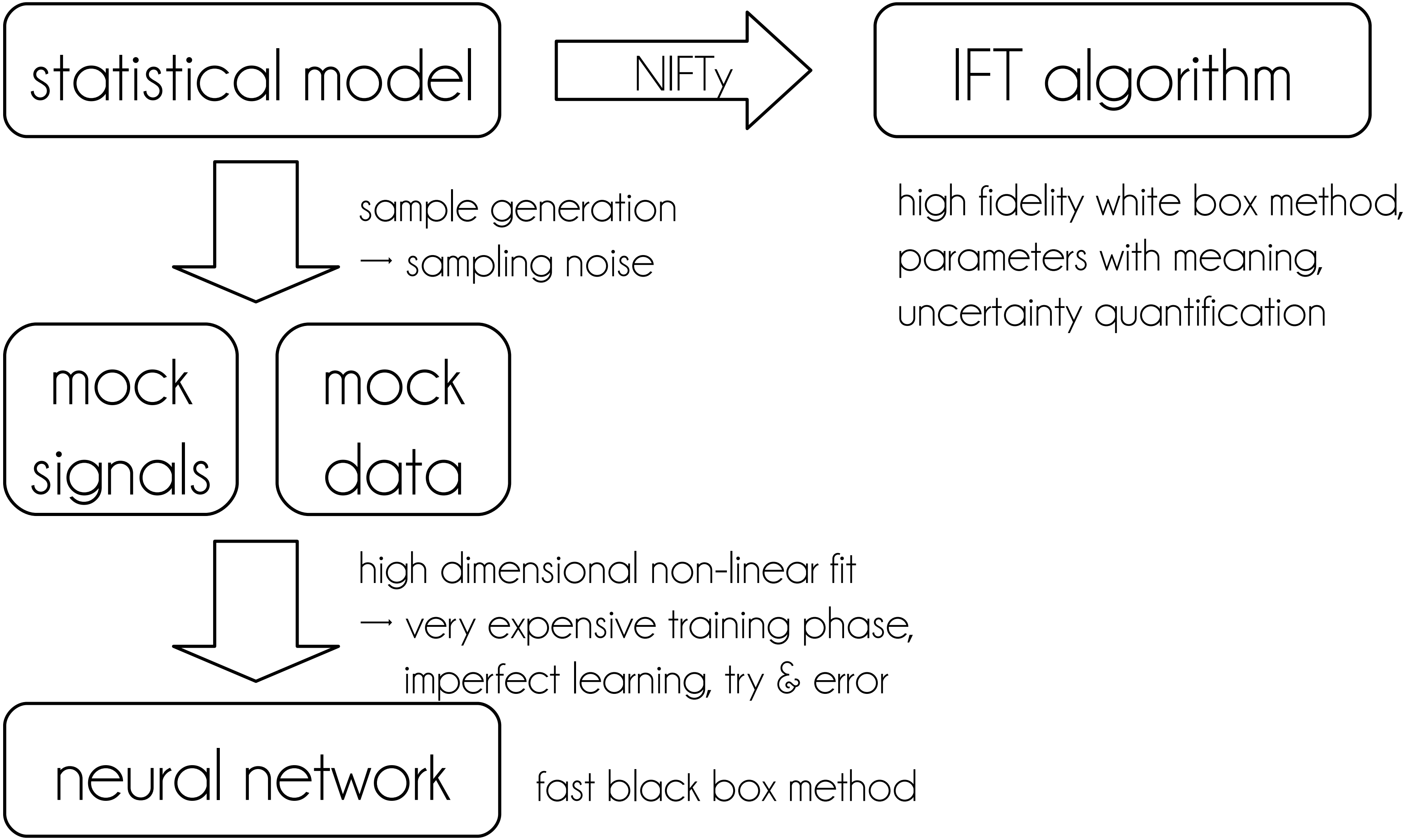


ground truth / starblade



ground truth / autoencoder




statistical model
NIFTy
IFT algorithm
sample generation
→ sampling noise
high fidelity white box method,
parameters with meaning,
uncertainty quantification
mock signals
mock data
high dimensional non-linear fit
→ very expensive training phase,
imperfect learning, try & error
neural network
fast black box method

NIFTy tutorial part 2

nonlinear reconstructions

NIFTy5

5.0

Search docs

NIFTy – Numerical Information Field Theory

NIFTy [1], [2], "**Numerical Information Field Theory**" is a versatile library designed to enable the development of signal inference algorithms that are independent of the underlying grids (spatial, spectral, temporal, …) and their resolutions. Its object-oriented framework is written in Python, although it accesses libraries written in C++ and C for efficiency.

NIFTy offers a toolkit that abstracts discretized representations of continuous spaces, fields in these spaces, and operators acting on these fields into classes. This allows for an abstract formulation and programming of inference algorithms, including those derived within information field theory. NIFTy's interface is designed to resemble IFT formulae in the sense that the user implements algorithms in NIFTy independent of the topology of the underlying spaces and the discretization scheme. Thus, the user can develop algorithms on subsets of problems and on spaces where the detailed performance of the algorithm can be properly evaluated and then easily generalize them to other, more complex spaces and the full problem, respectively.

The set of spaces on which NIFTy operates comprises point sets, *n*-dimensional regular grids, spherical spaces, their harmonic counterparts, and product spaces constructed as combinations of those. NIFTy takes care of numerical subtleties like the normalization of operations on fields and the numerical representation of model components, allowing the user to focus on formulating the abstract inference procedures and process-specific model properties.

References

[1] Selig et al., "NIFTY - Numerical Information Field Theory. A versatile PYTHON library for signal inference ", 2013, Astronmy and Astrophysics 554, 26; [DOI], [arXiv:1301.4499]

[2] Steininger et al., "NIFTy 3 - Numerical Information Field Theory - A Python framework for multicomponent signal inference on HPC clusters", 2017, accepted by Annalen der Physik; [arXiv:1708.01073]

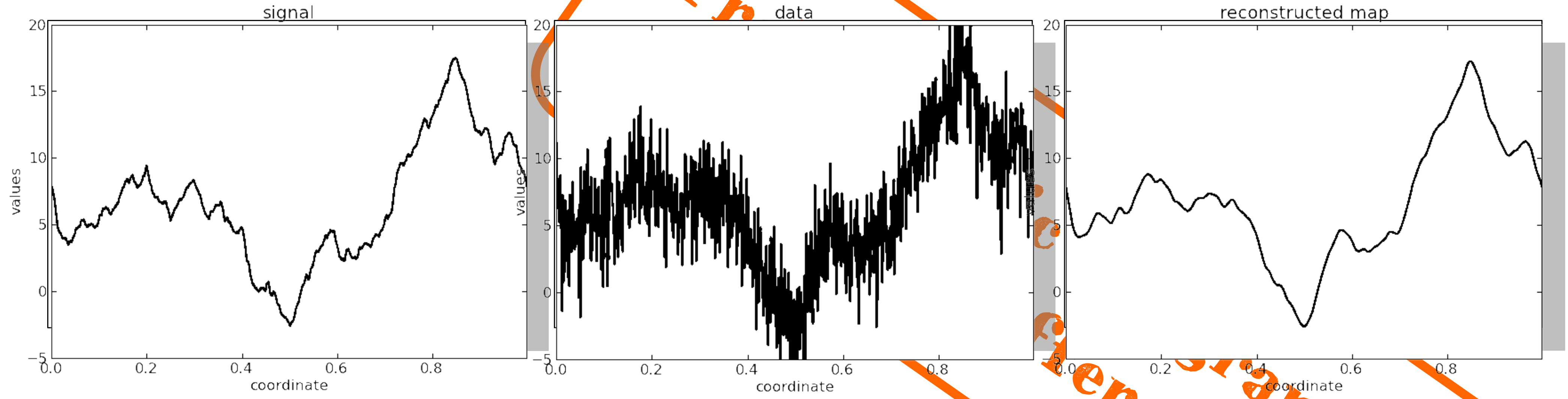
Contents

Probabilistic programming with auto-differentiation

NIFTy5

NIFTy – Numerical Information Field Theory

NIFTy [1], [2], “Numerical Information Field Theory” is a versatile library designed to enable the development of signal inference algorithms that are independent of the underlying grids (spatial, spectral, temporal, …) and their resolutions. Its object-oriented framework is written in Pytho

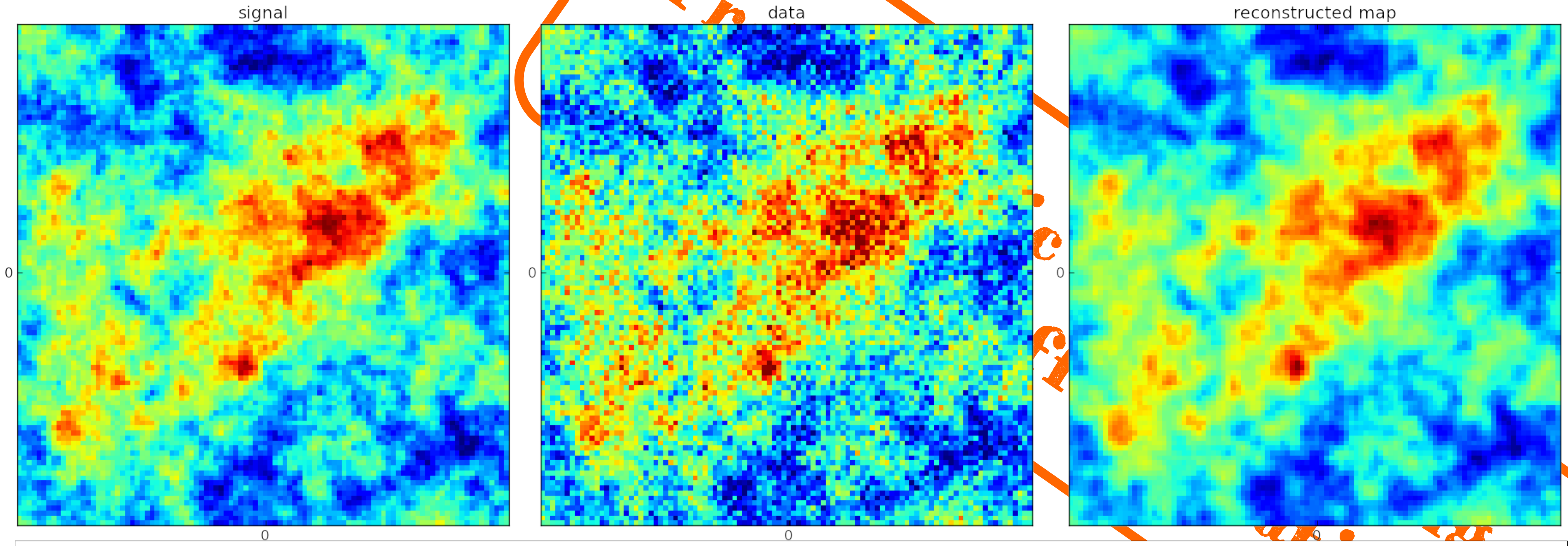



```
import nifty5 as ift
s_space = ift.RGSpace([N])
```

NIFTy5

NIFTy – Numerical Information Field Theory

NIFTy [1], [2], "Numerical Information Field Theory" is a versatile library designed to enable the development of signal inference algorithms that are independent of the underlying grids (spatial, spectral, temporal, …) and their resolutions. Its object-oriented framework is written in Python

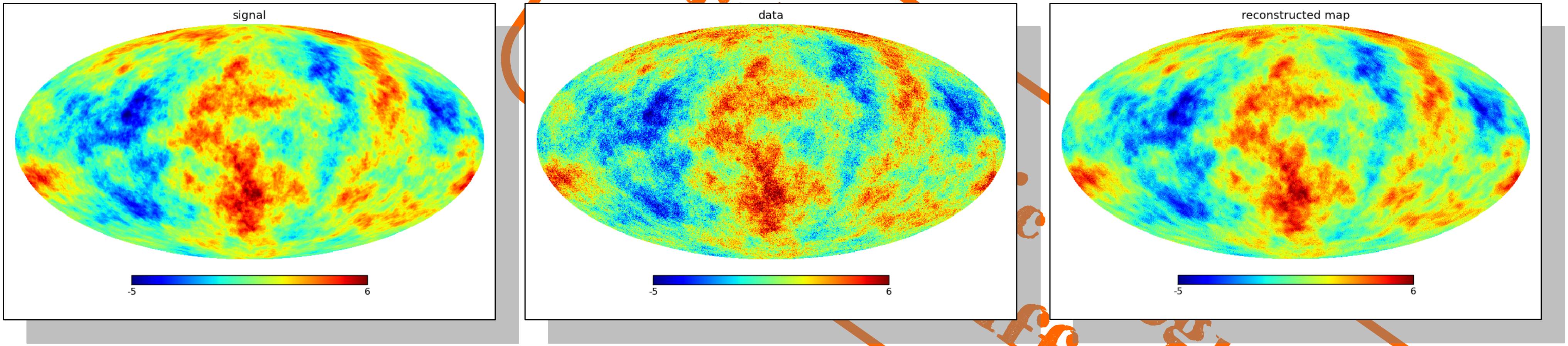



```
import nifty5 as ift
s_space = ift.RGSpace([N,N])
```

NIFTy5

NIFTy – Numerical Information Field Theory

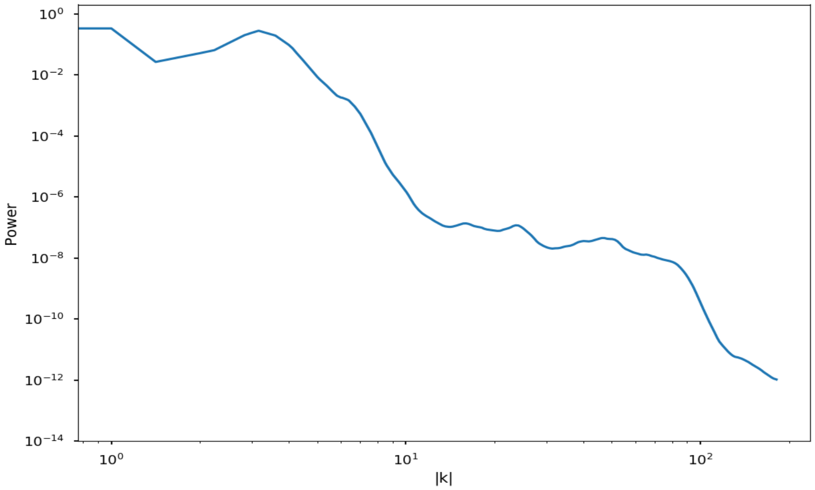
NIFTy [1], [2], “Numerical Information Field Theory” is a versatile library designed to enable the development of signal inference algorithms that are independent of the underlying grids (spatial, spectral, temporal, …) and their resolutions. Its object-oriented framework is written in Python

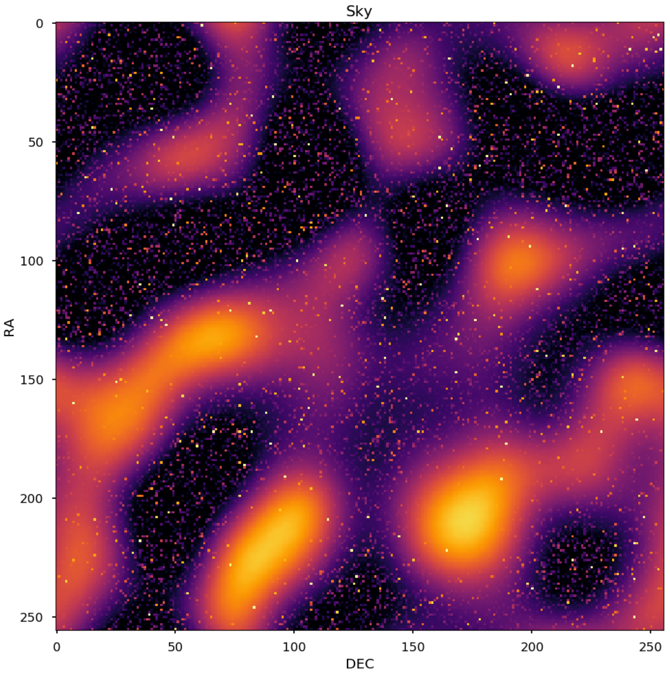


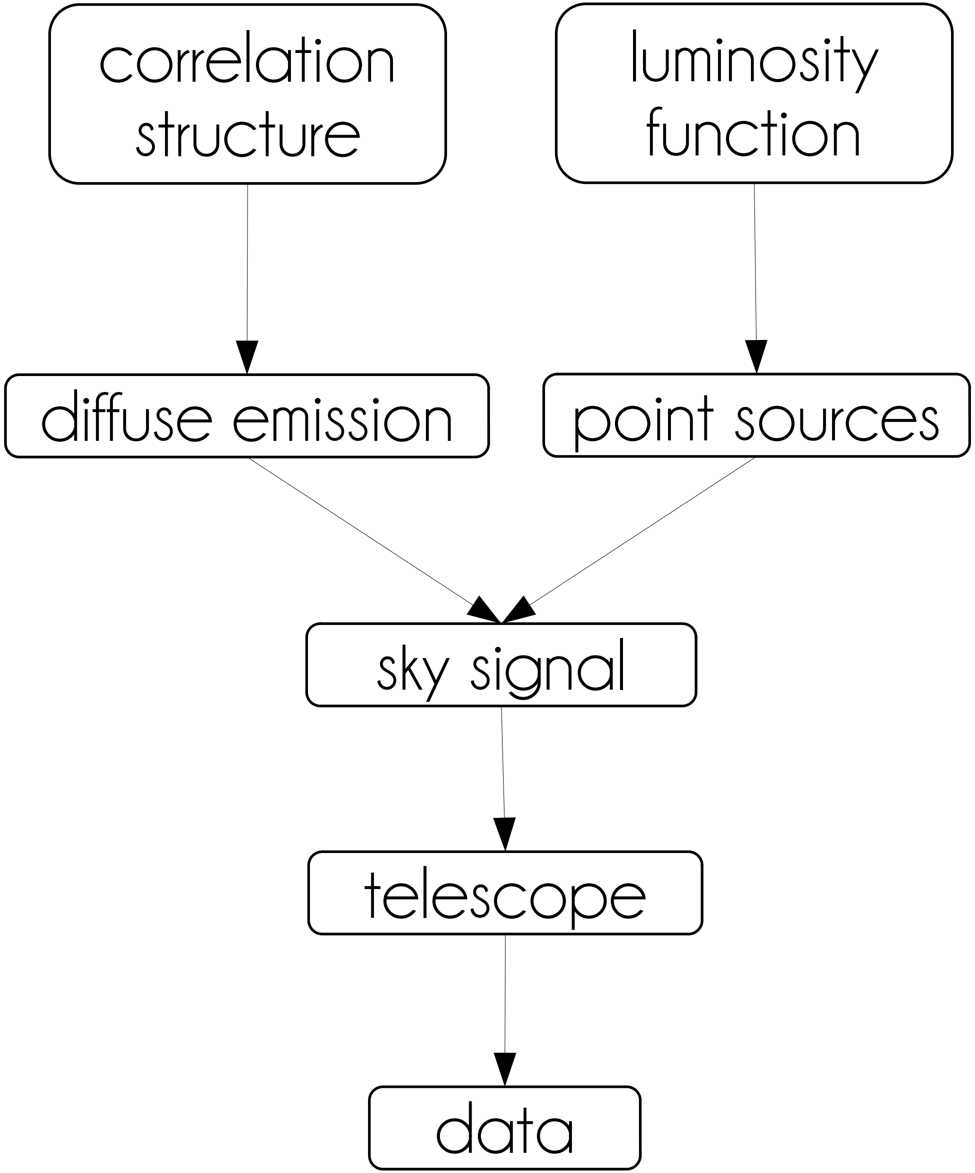

```
import nifty5 as ift
s_space = ift.HPSpace(NSide)
```

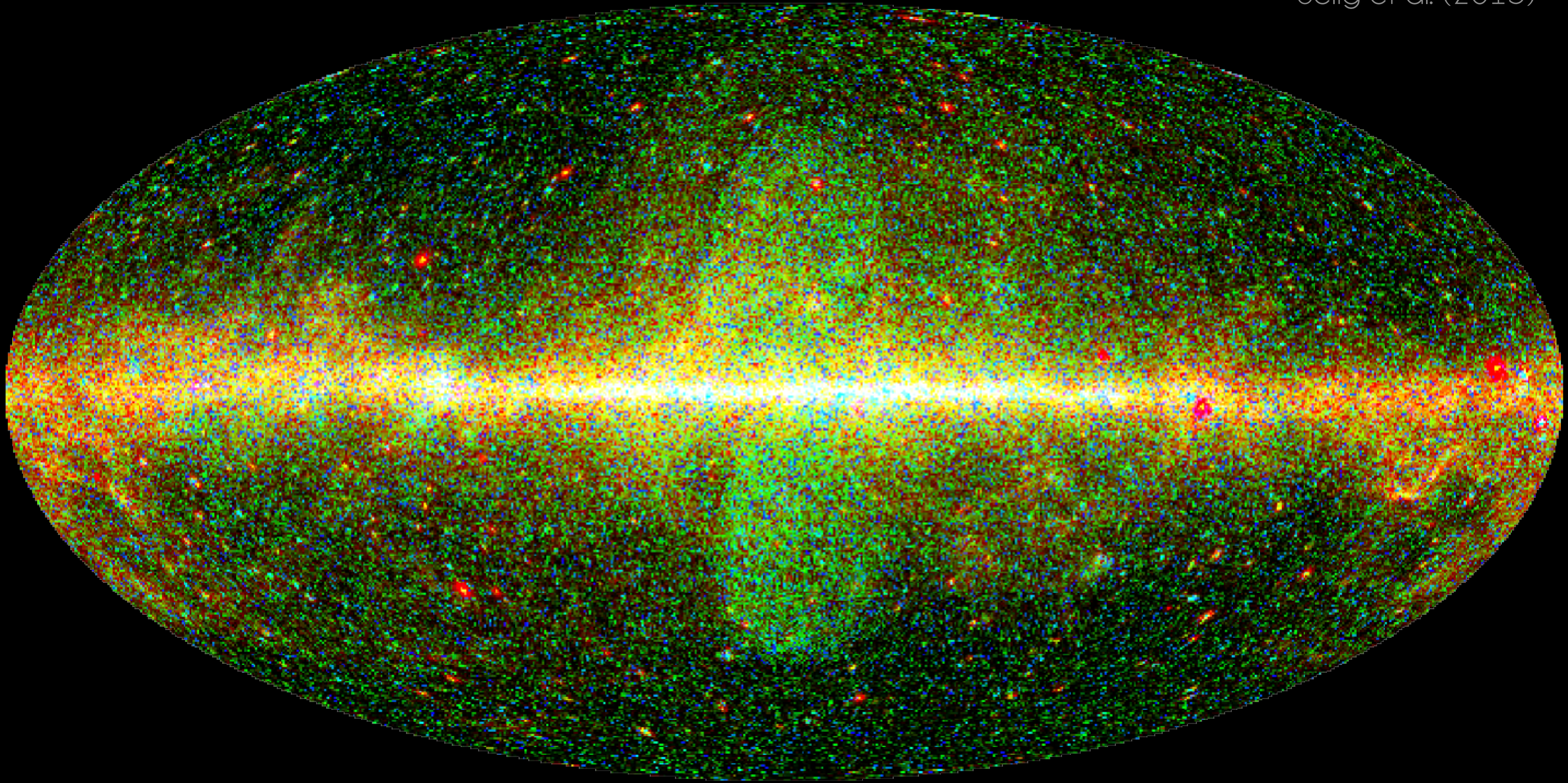
NIFTy tutorial part 1

linear reconstructions

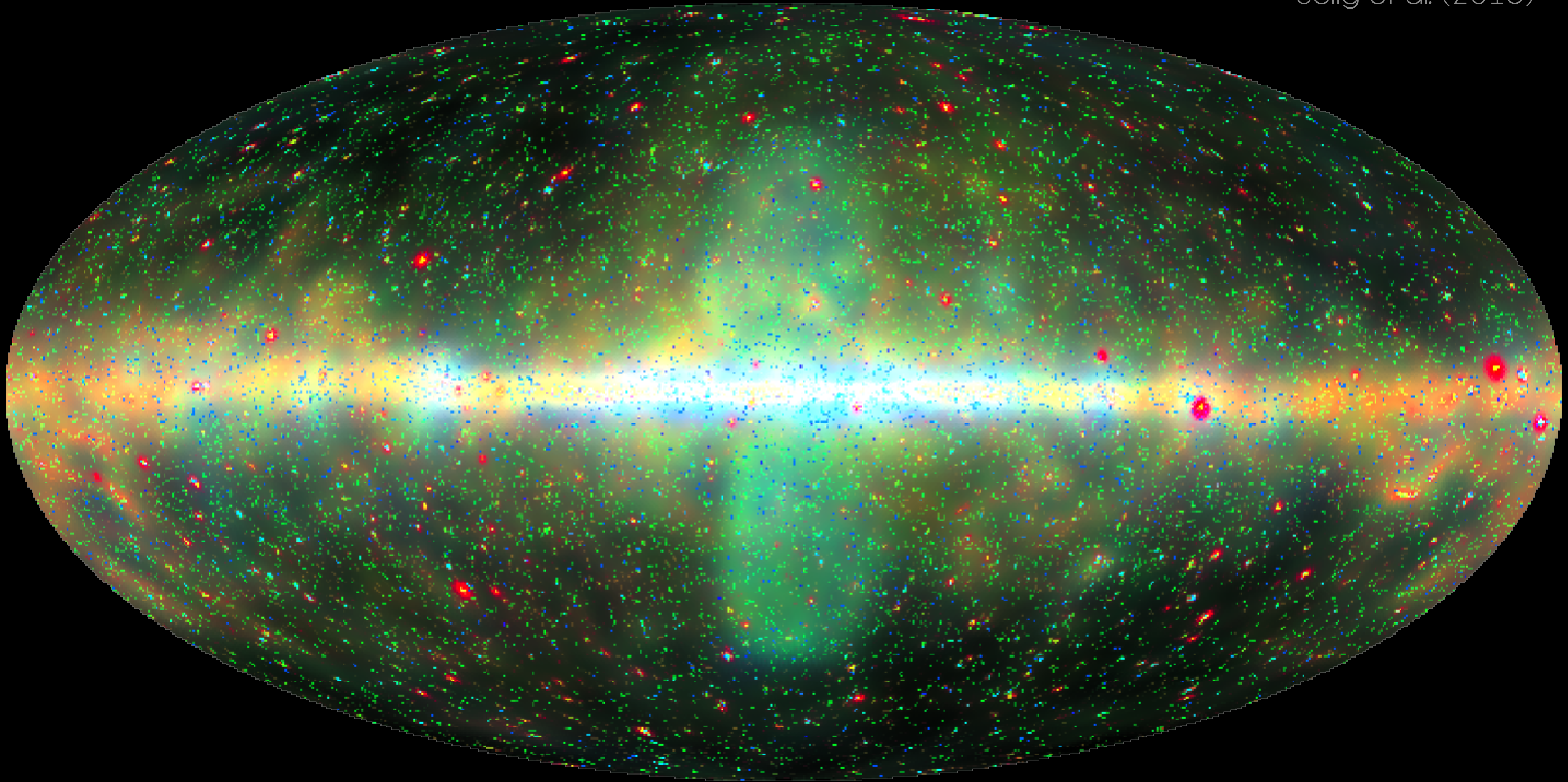

Power
|k|
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-14}
10^{0}
10^{1}
10^{2}


Sky
RA
DEC
0
50
100
150
200
250

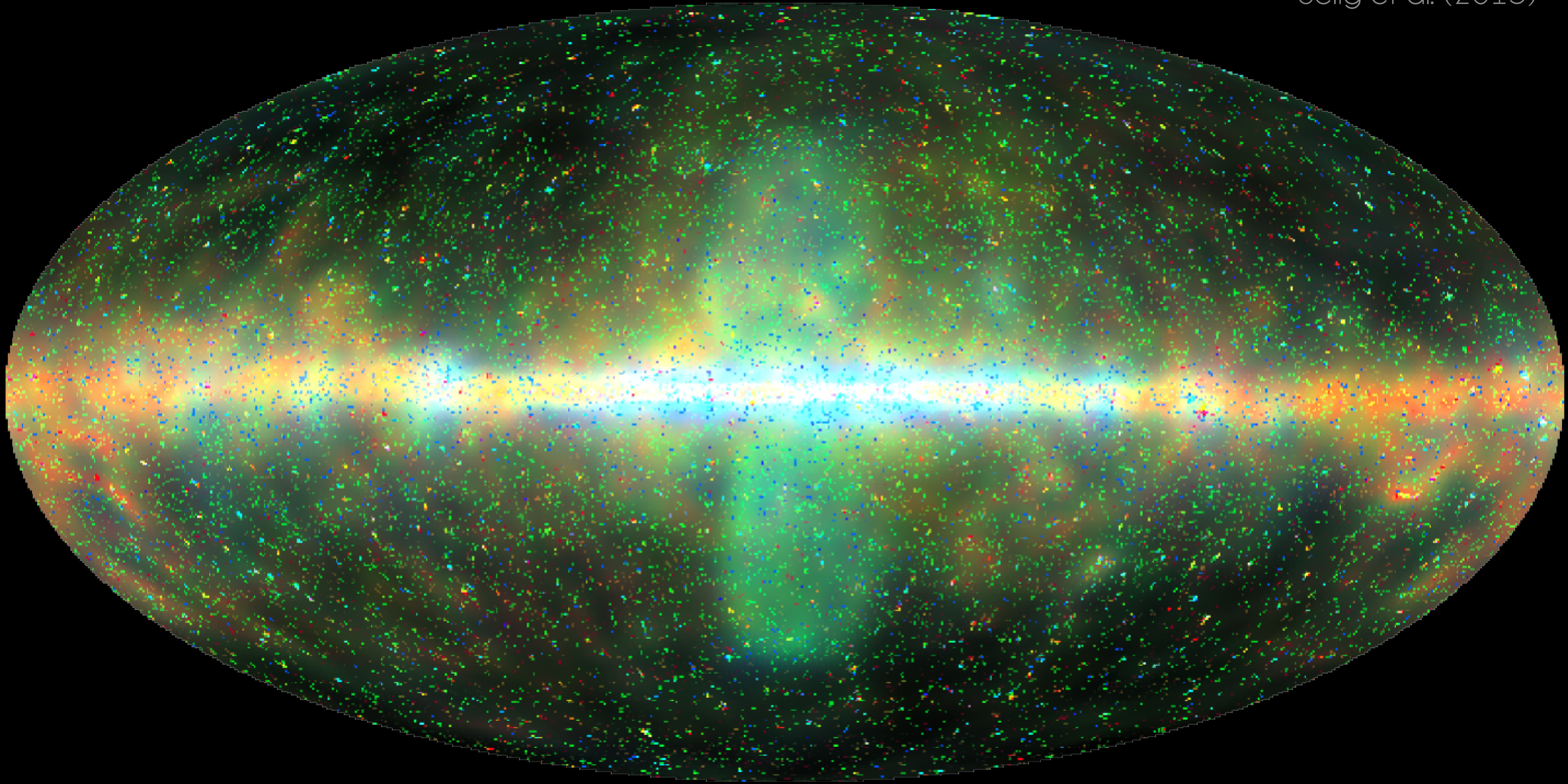

correlation structure
luminosity function
diffuse emission
point sources
sky signal
telescope
data



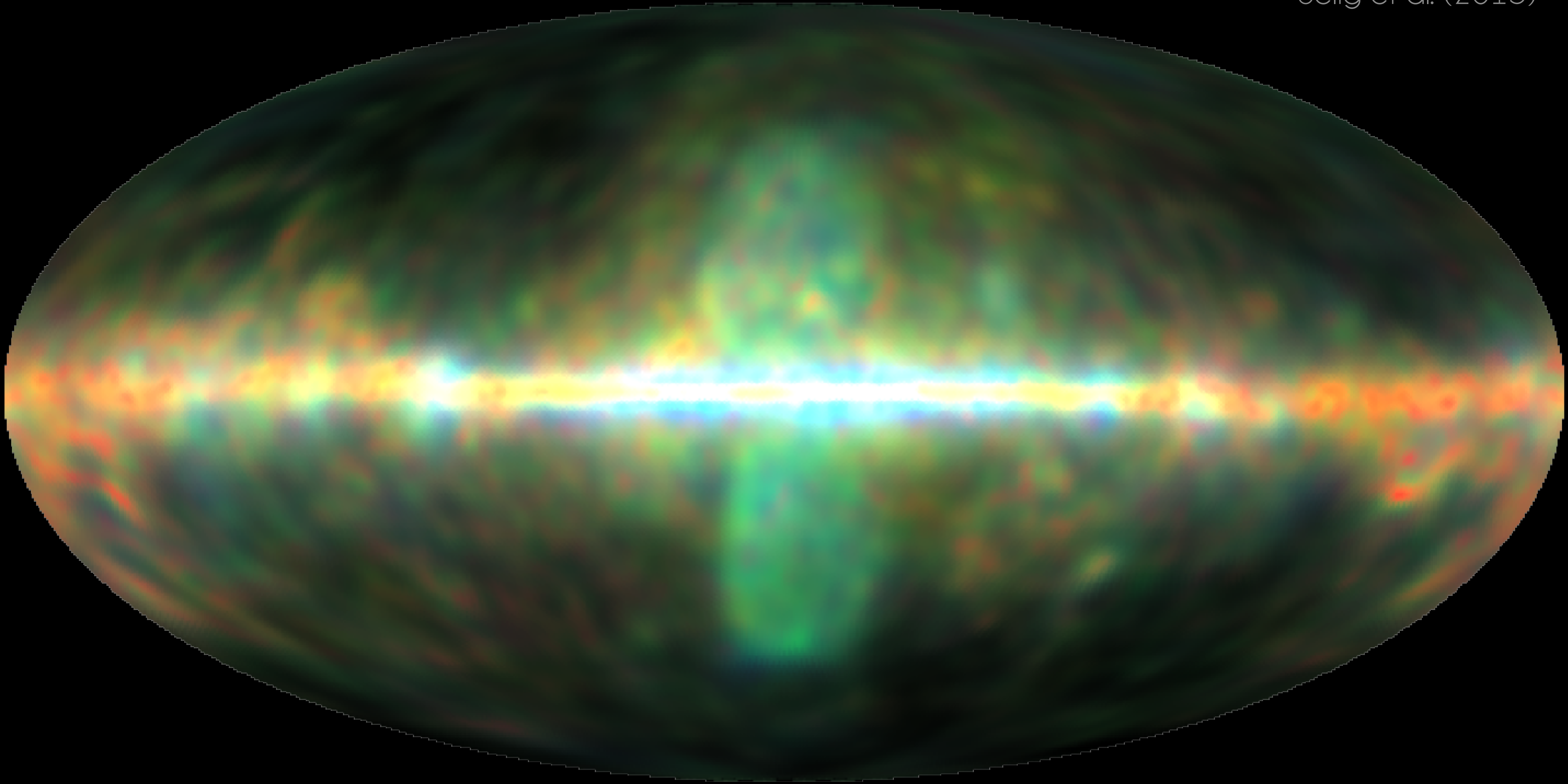
Selig et al. (2015)



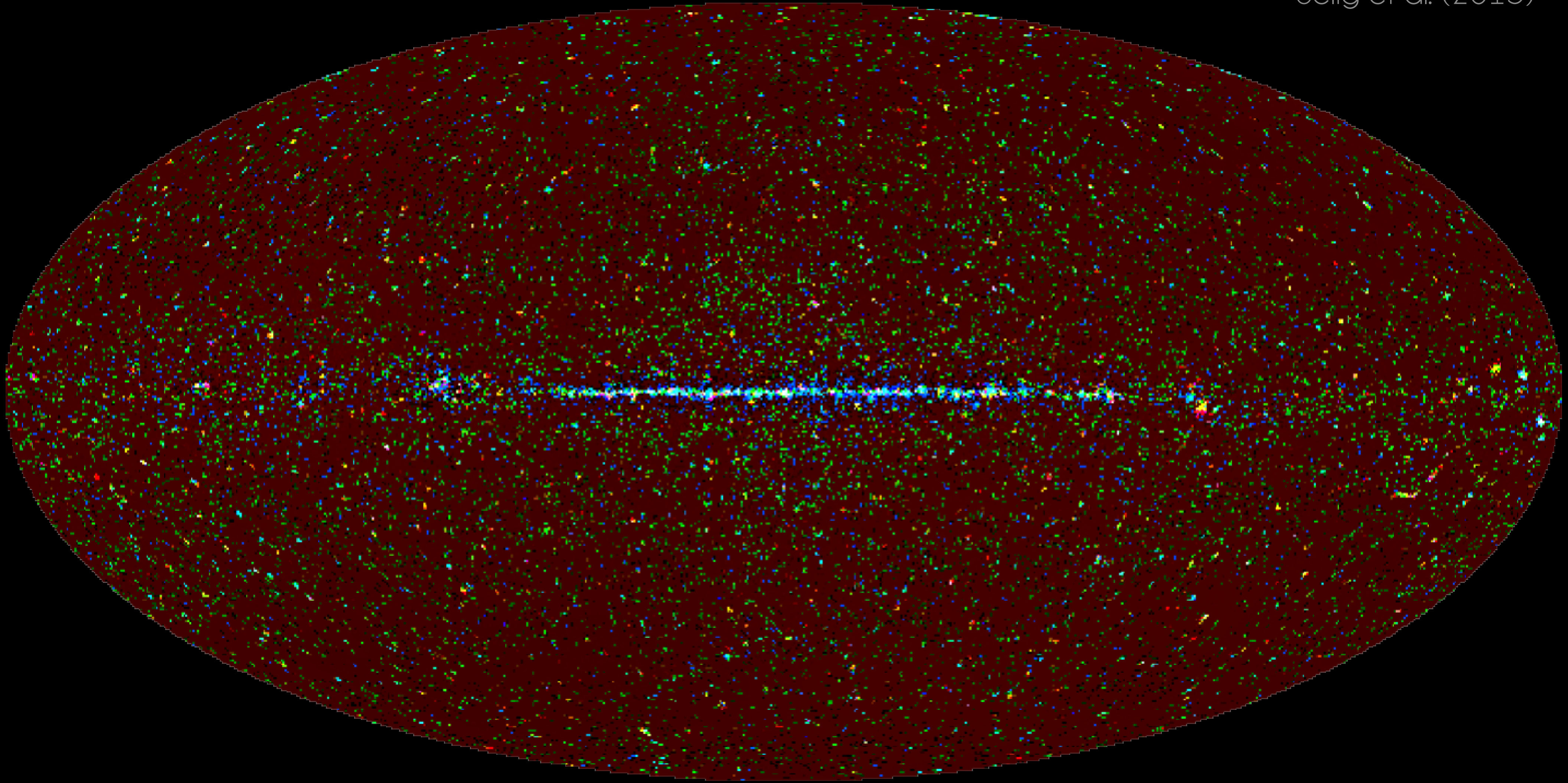
Selig et al. (2015)



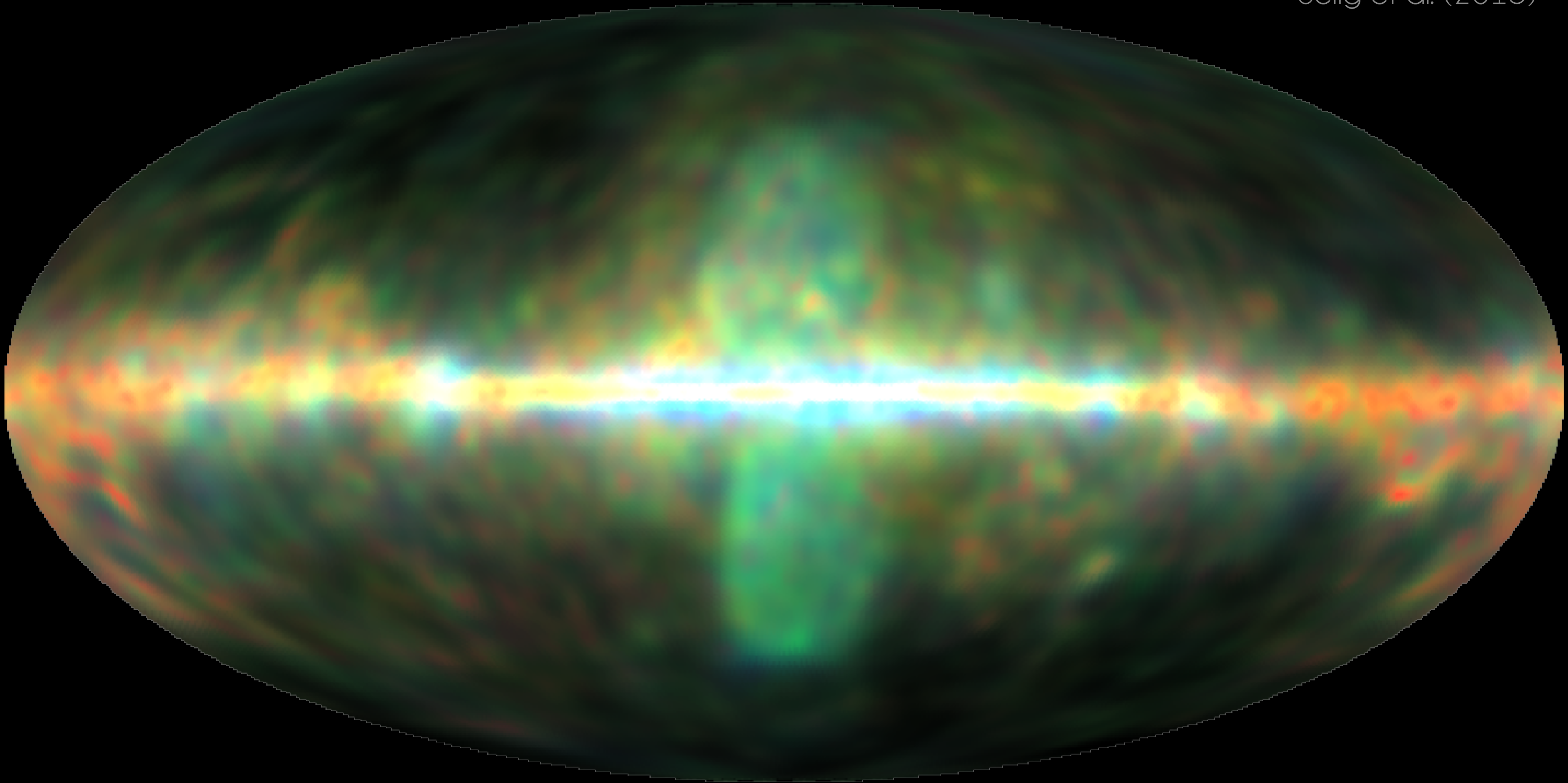
Selig et al. (2015)



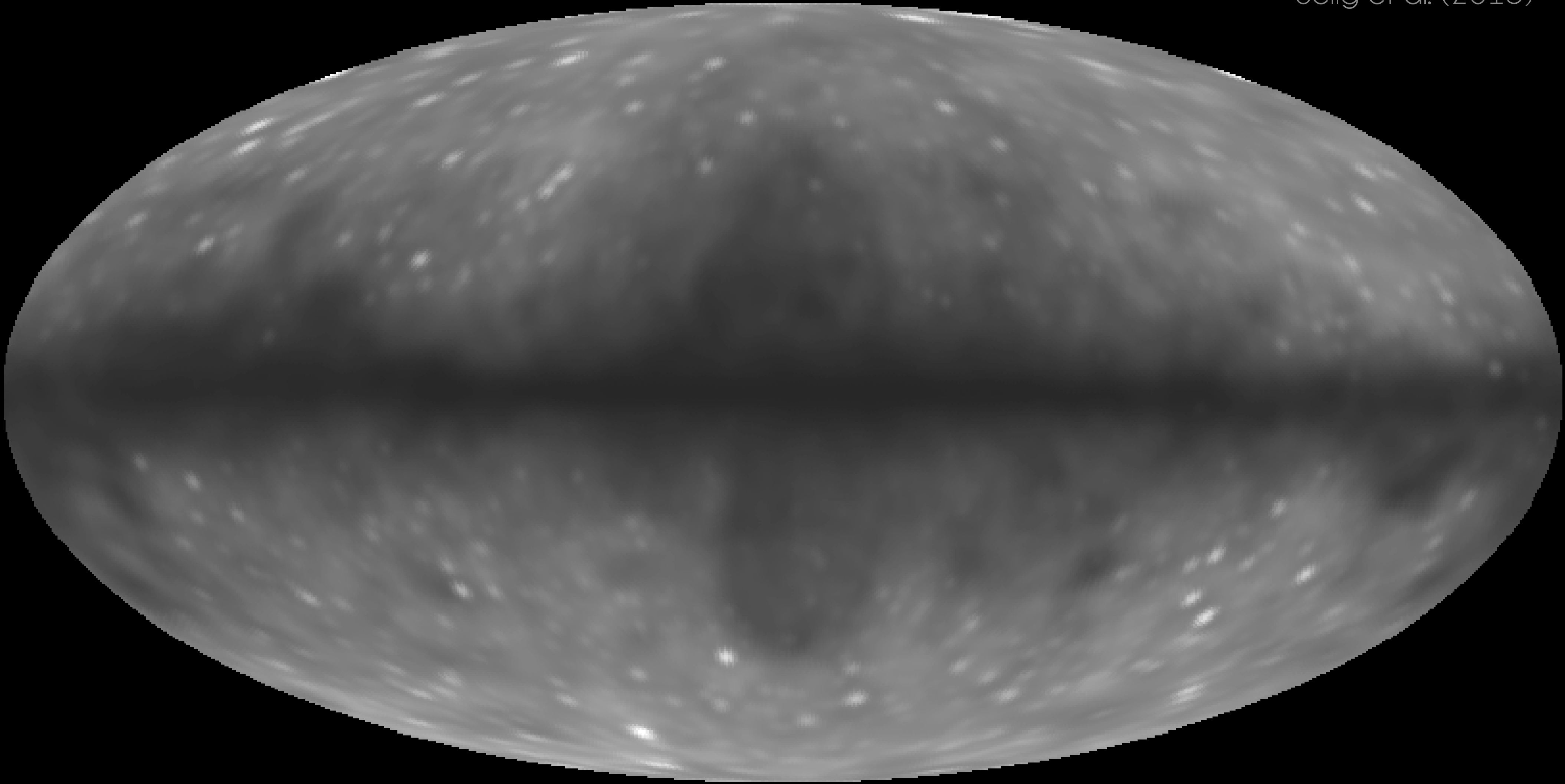
Selig et al. (2015)



Selig et al. (2015)



Selig et al. (2015)



Selig et al. (2015)

$\mathcal{P}(d|s)$ Data model

known → $d = R\,e^{s} + n$

unknown → $\lambda = R\,e^{s}$

$\mathcal{P}(s) = \mathcal{G}(s,\, S)$ unknown

$$\mathcal{P}(d|\lambda) = \prod_i \frac{\lambda_i^{d_i}}{d_i!}\, e^{-\lambda_i}$$

Information

$$
\begin{aligned}
\mathcal{H}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau}) &= -\log \mathcal{P}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau}) \\
&= 1^\dagger \left[\log(d!) + \boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right] - \boldsymbol{d}^\dagger \log\left[\boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right] \\
&\quad + \frac{1}{2}\boldsymbol{s}^\dagger \boldsymbol{S}^{-1}\boldsymbol{s} + \frac{1}{2}\log\left(\det\left[\boldsymbol{S}\right]\right) \\
&\quad + (\boldsymbol{\alpha} - \boldsymbol{1})^\dagger \boldsymbol{\tau} + \boldsymbol{q}^\dagger \mathrm{e}^{-\boldsymbol{\tau}} + \frac{1}{2}\boldsymbol{\tau}^\dagger \boldsymbol{T}\boldsymbol{\tau} \\
&\quad + (\boldsymbol{\beta} - \boldsymbol{1})^\dagger \boldsymbol{u} + \boldsymbol{\eta}^\dagger \mathrm{e}^{-\boldsymbol{u}}
\end{aligned}
$$

$$
\boldsymbol{S} = \sum_k \mathrm{e}^{\tau_k} \boldsymbol{S}_k
$$

Information

$$\mathcal{H}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau}) = -\log \mathcal{P}(\boldsymbol{d}, \boldsymbol{s}, \boldsymbol{\tau})$$

$$= 1^\dagger \left[\log(d!) + \boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right] - \boldsymbol{d}^\dagger \log\left[\boldsymbol{R}\left(\mathrm{e}^{\boldsymbol{s}} + \mathrm{e}^{\boldsymbol{u}}\right)\right]$$

$$+ \frac{1}{2}\boldsymbol{s}^\dagger \boldsymbol{S}^{-1}\boldsymbol{s} + \frac{1}{2}\log\left(\det\left[\boldsymbol{S}\right]\right)$$

$$+ (\boldsymbol{\alpha} - \mathbf{1})^\dagger \ldots$$

$$+ (\boldsymbol{\beta} - \mathbf{1})^\dagger \ldots$$

$$\boldsymbol{S} = \sum_k \mathrm{e}^{\tau_k} \ldots$$

- Convert into **generative model**
- Compress information into Gaussian via **Metric Gaussian Variational Inference**

Variational Bayes

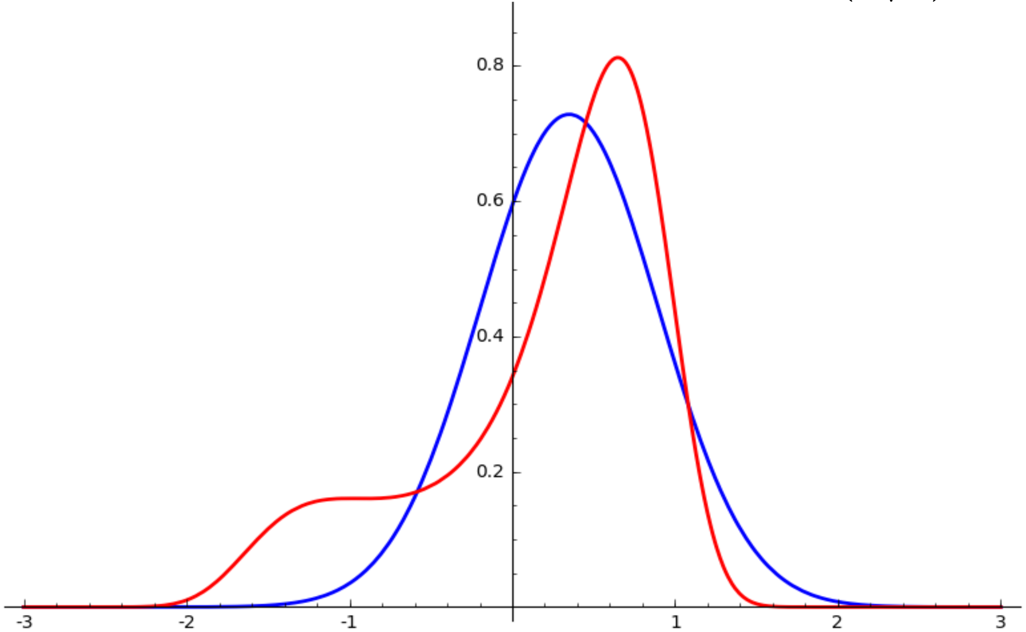
$\mathcal{P}(s|d)$

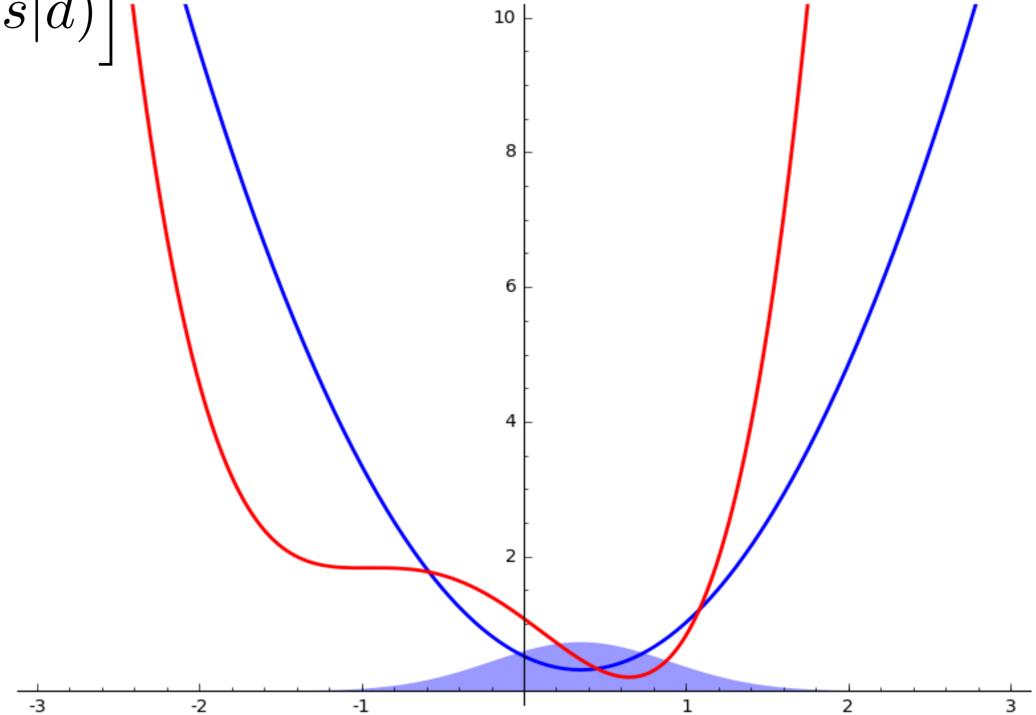
$\widetilde{\mathcal{P}}(s|d) = \mathcal{G}(s-m, D)$

$\mathcal{H}(s|d)$

$\widetilde{\mathcal{H}}(s|d) \widehat{=} \frac{1}{2}(s-m)^\dagger D^{-1}(s-m)$

$$\mathrm{KL}(\widetilde{\mathcal{P}}, \mathcal{P}) = \int \mathcal{D}s\, \widetilde{\mathcal{P}}(s|d) \left[\mathcal{H}(s|d) - \widetilde{\mathcal{H}}(s|d)\right]$$





Metric Gaussian Variational Bayes

$\mathcal{P}(s|d)$

$\widetilde{\mathcal{P}}(s|d) = \mathcal{G}(s-m, D)$

$\mathcal{H}(s|d)$

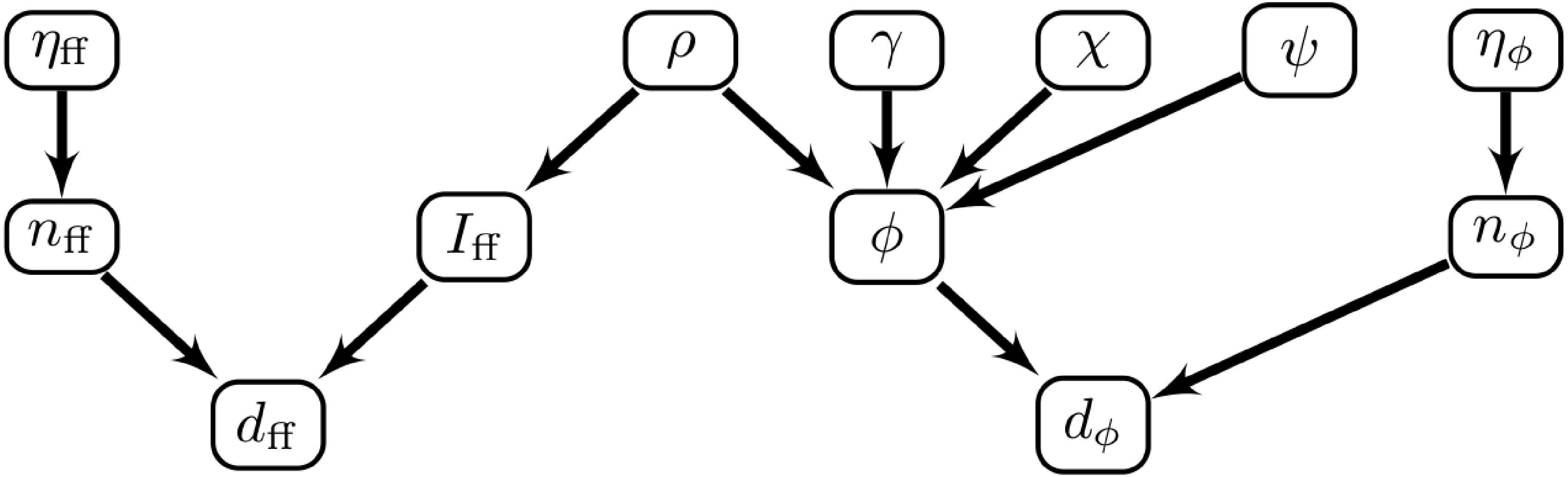
$\widetilde{\mathcal{H}}(s|d) \widehat{=} \frac{1}{2}(s-m)^\dagger D^{-1}(s-m)$

Knollmüller & Enßlin (2019)

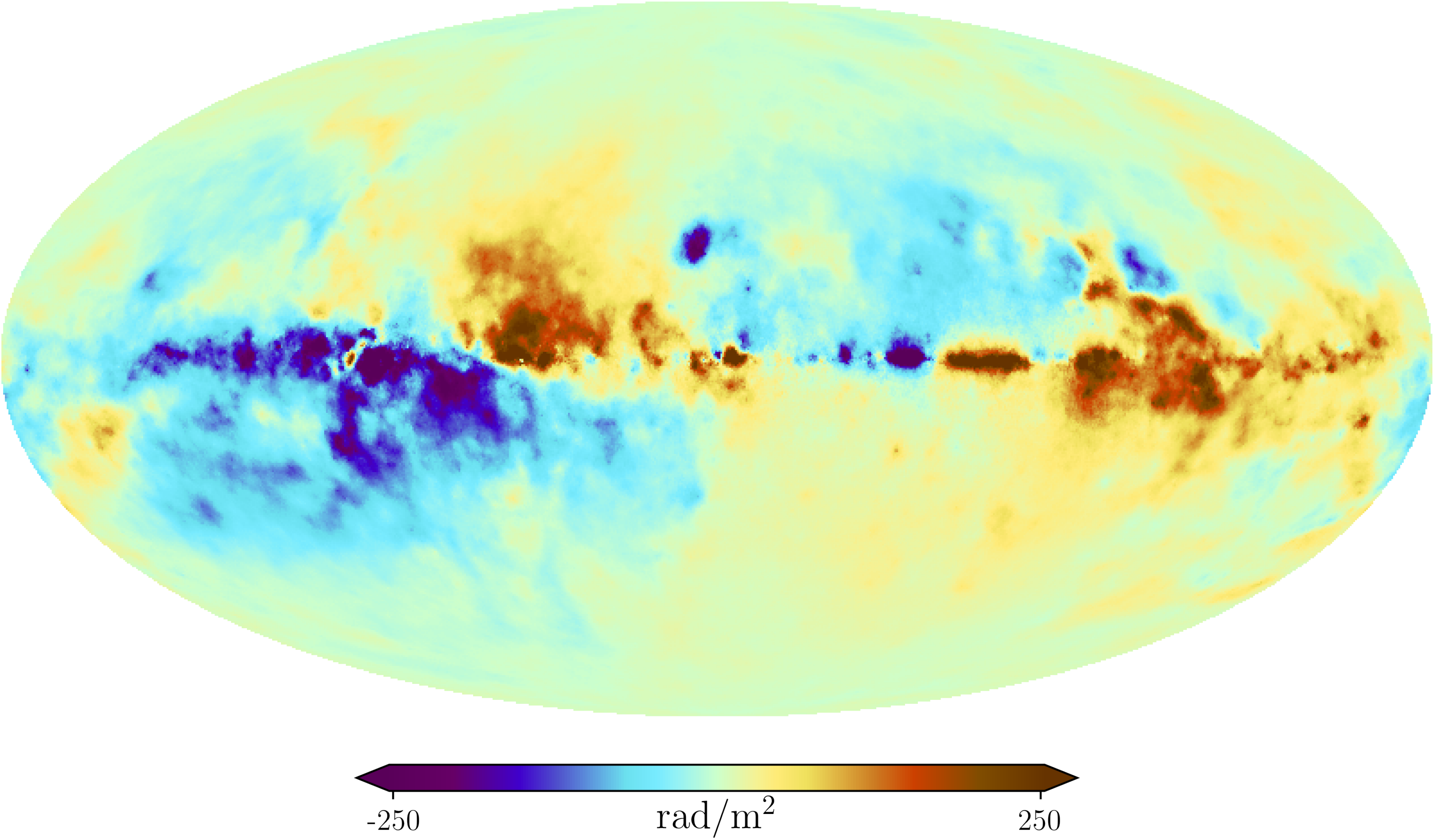
$$\mathrm{KL}(\widetilde{\mathcal{P}}, \mathcal{P}) = \int \mathcal{D}s\, \widetilde{\mathcal{P}}(s|d) \left[\mathcal{H}(s|d) - \widetilde{\mathcal{H}}(s|d)\right]$$

$$D \approx \left\langle \frac{\partial \mathcal{H}(d,s)}{\partial s} \frac{\partial \mathcal{H}(d,s)}{\partial s}^\dagger \right\rangle_{(d|s=m)}^{-1}$$

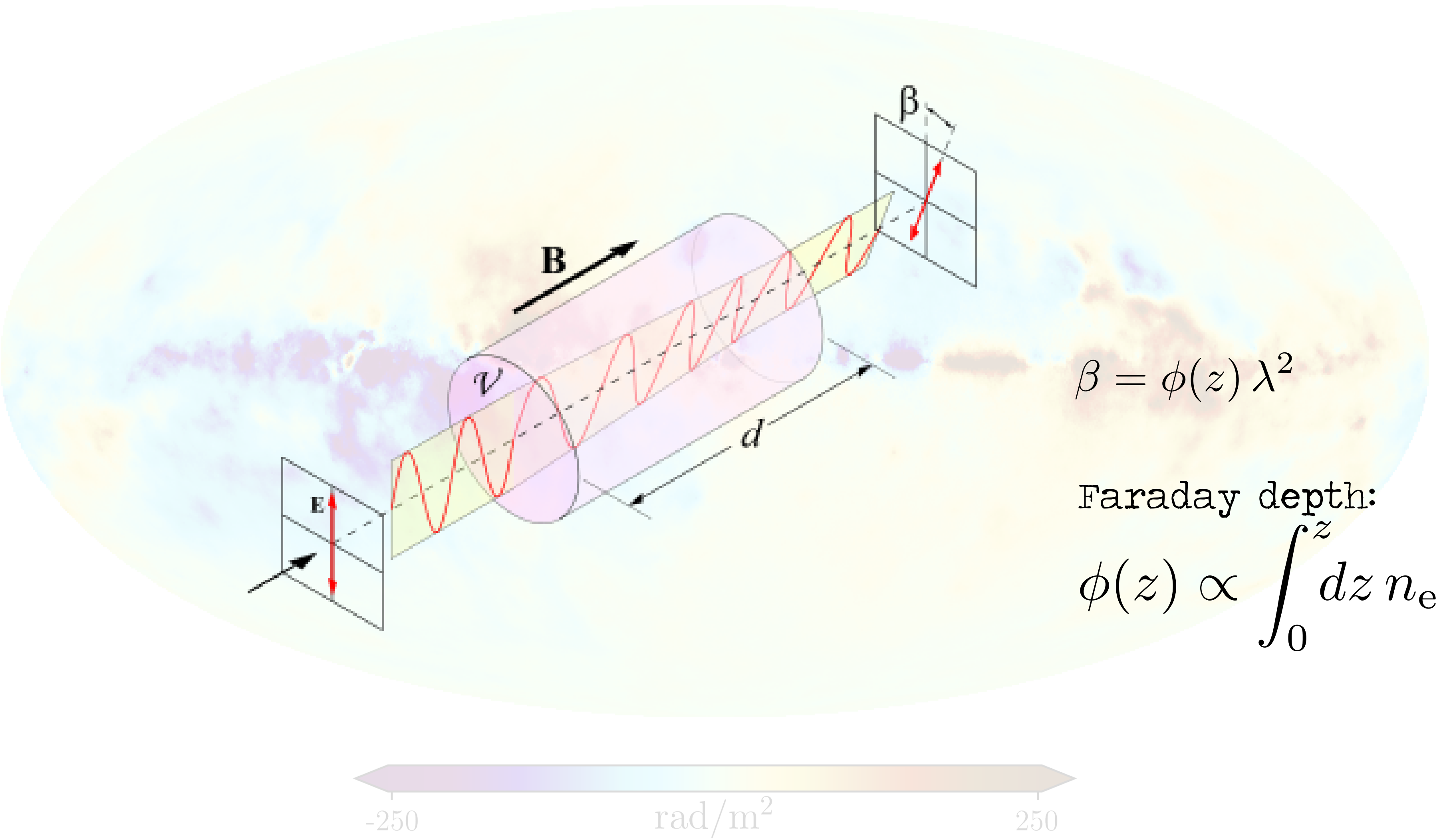
Hierarchical Bayesian Model



Galactic Faraday Sky Hutschenreuter & Enßlin (2019)



Faraday Effect



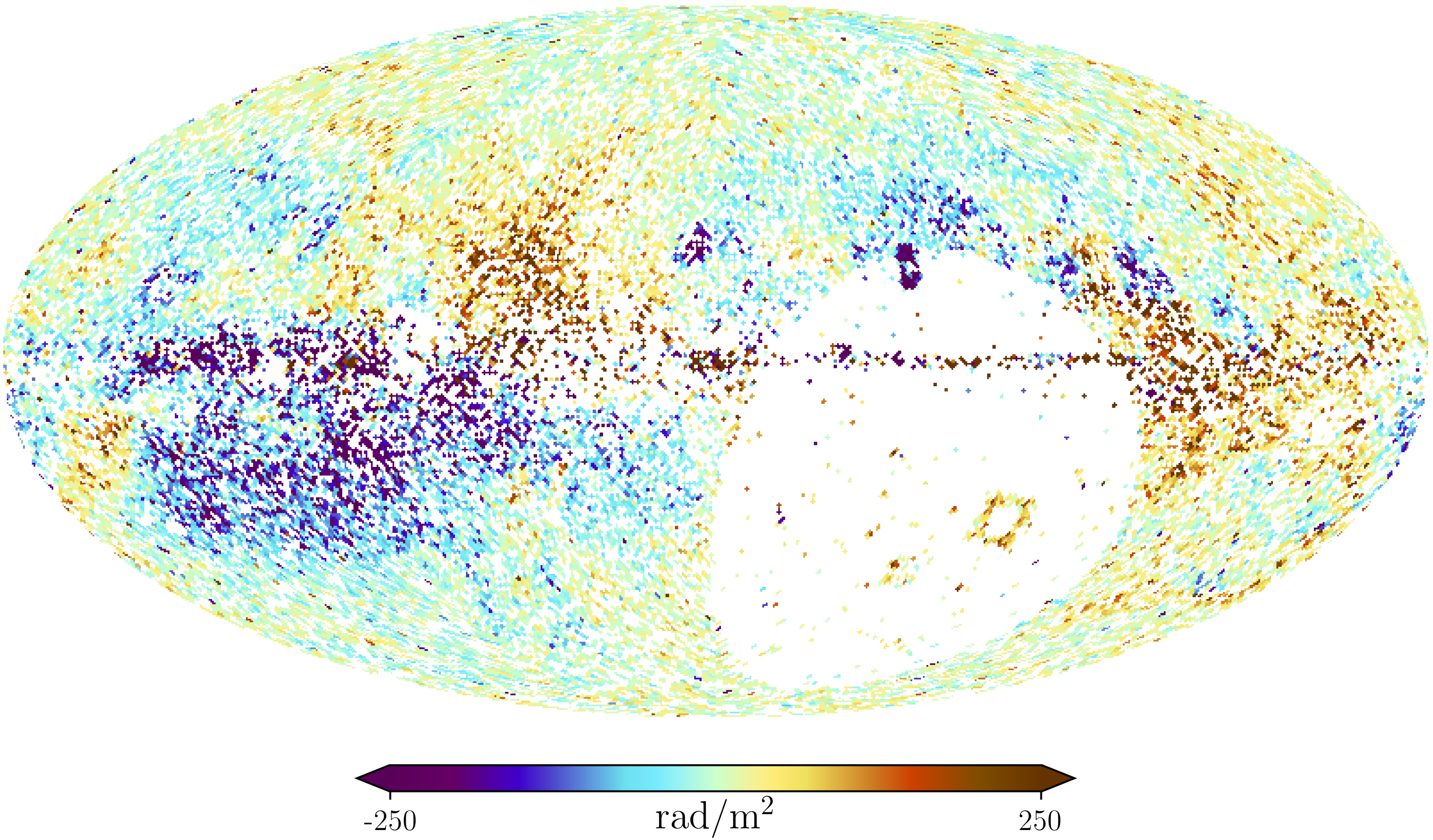
$$\beta = \phi(z)\,\lambda^2$$

Faraday depth:

$$\phi(z) \propto \int_0^z dz\, n_\mathrm{e}\, B_z$$

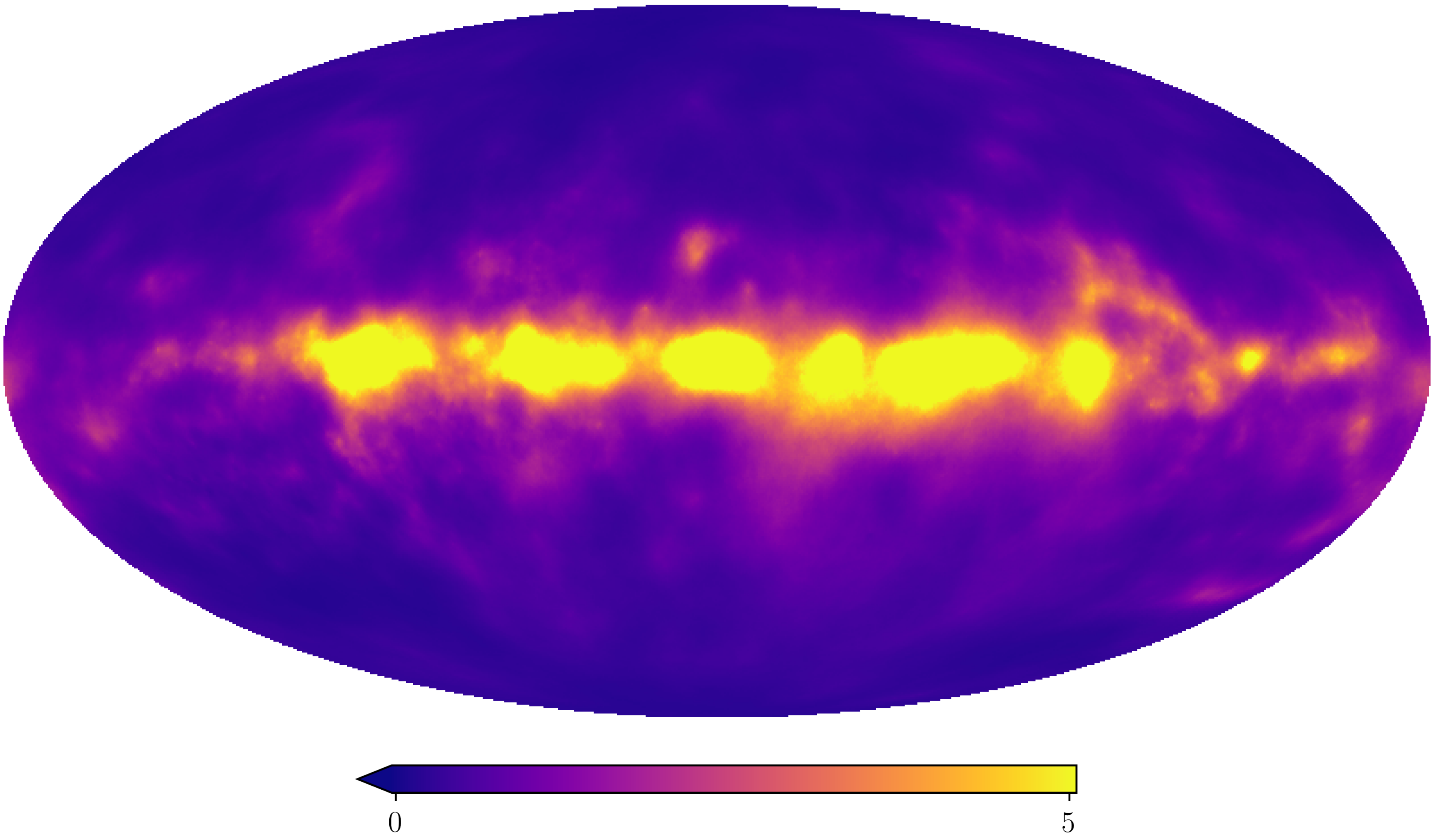
Faraday Data

Oppermann et al. (2012)



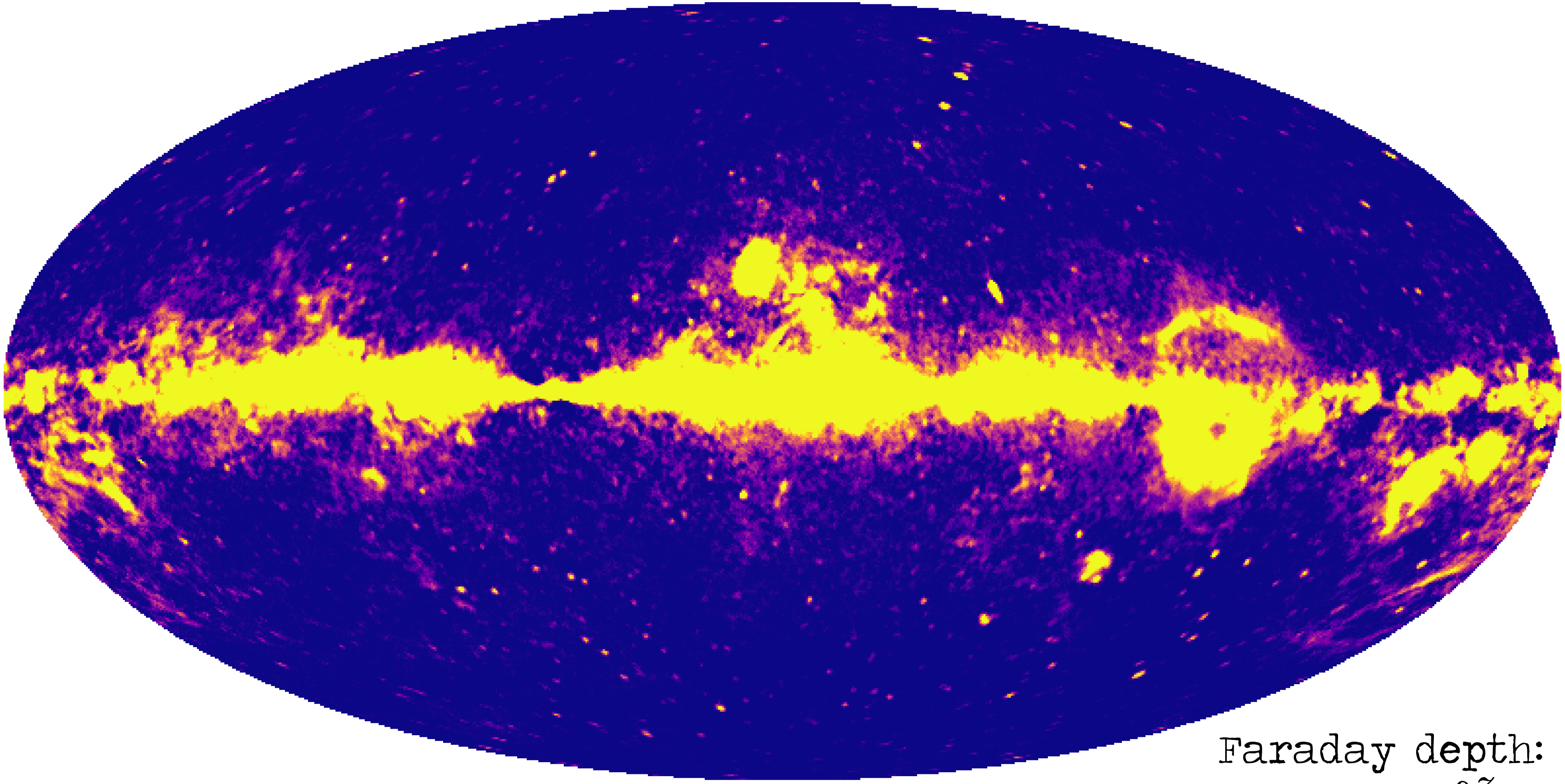
Faraday Amplitude Field

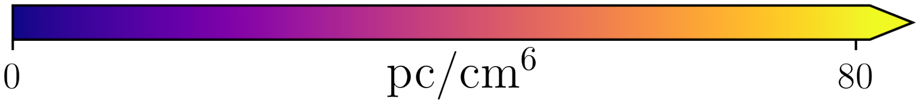
Hutschenreuter & Enßlin (2019)



Planck Free-Free Emission

Hutschenreuter & Enßlin (2019)



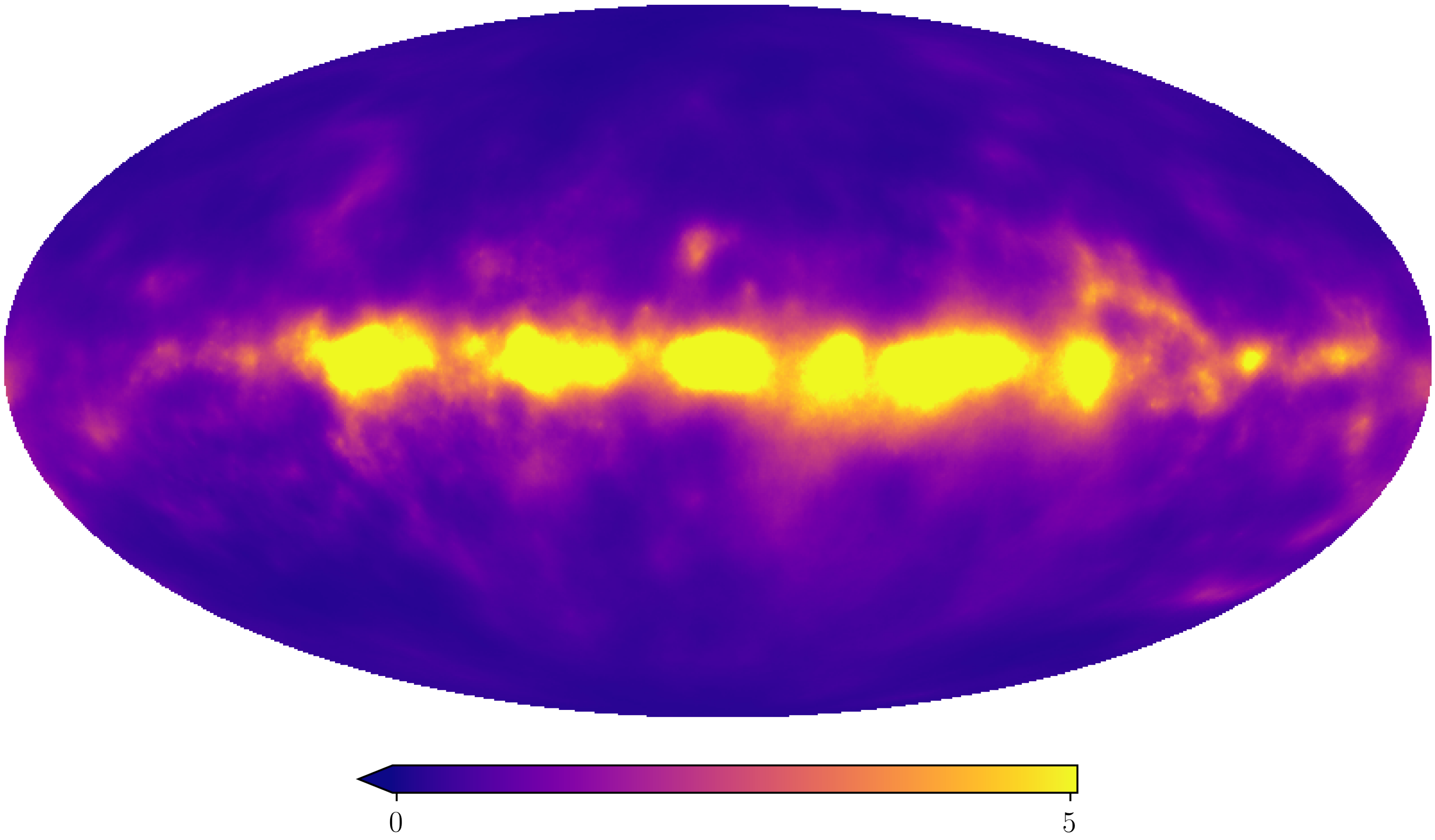


Faraday depth:

$$\phi(z) \propto \int_0^z dz\, n_{\mathrm{e}} B_z$$

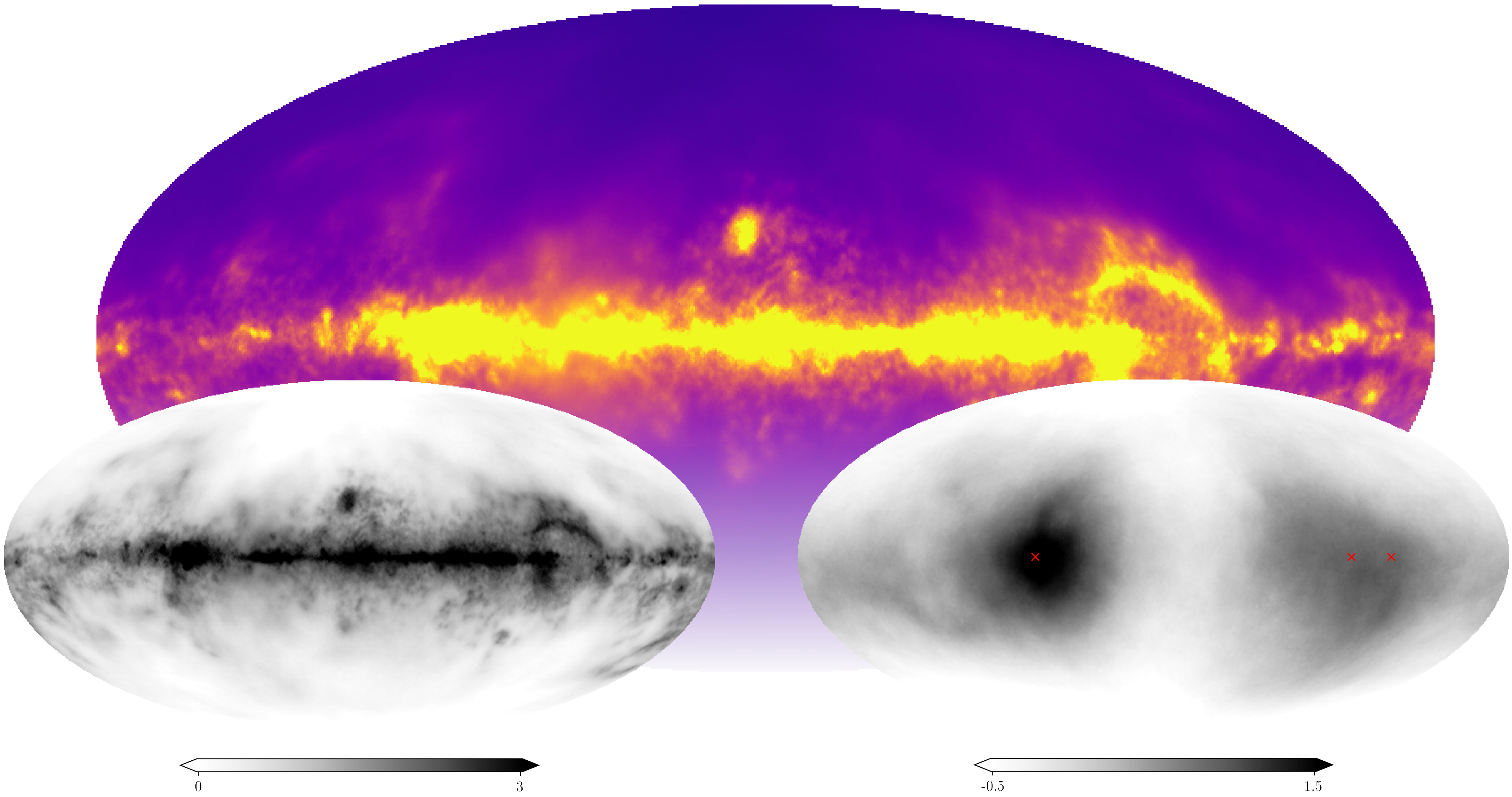
Faraday Amplitude Field

Hutschenreuter & Enßlin (2019)



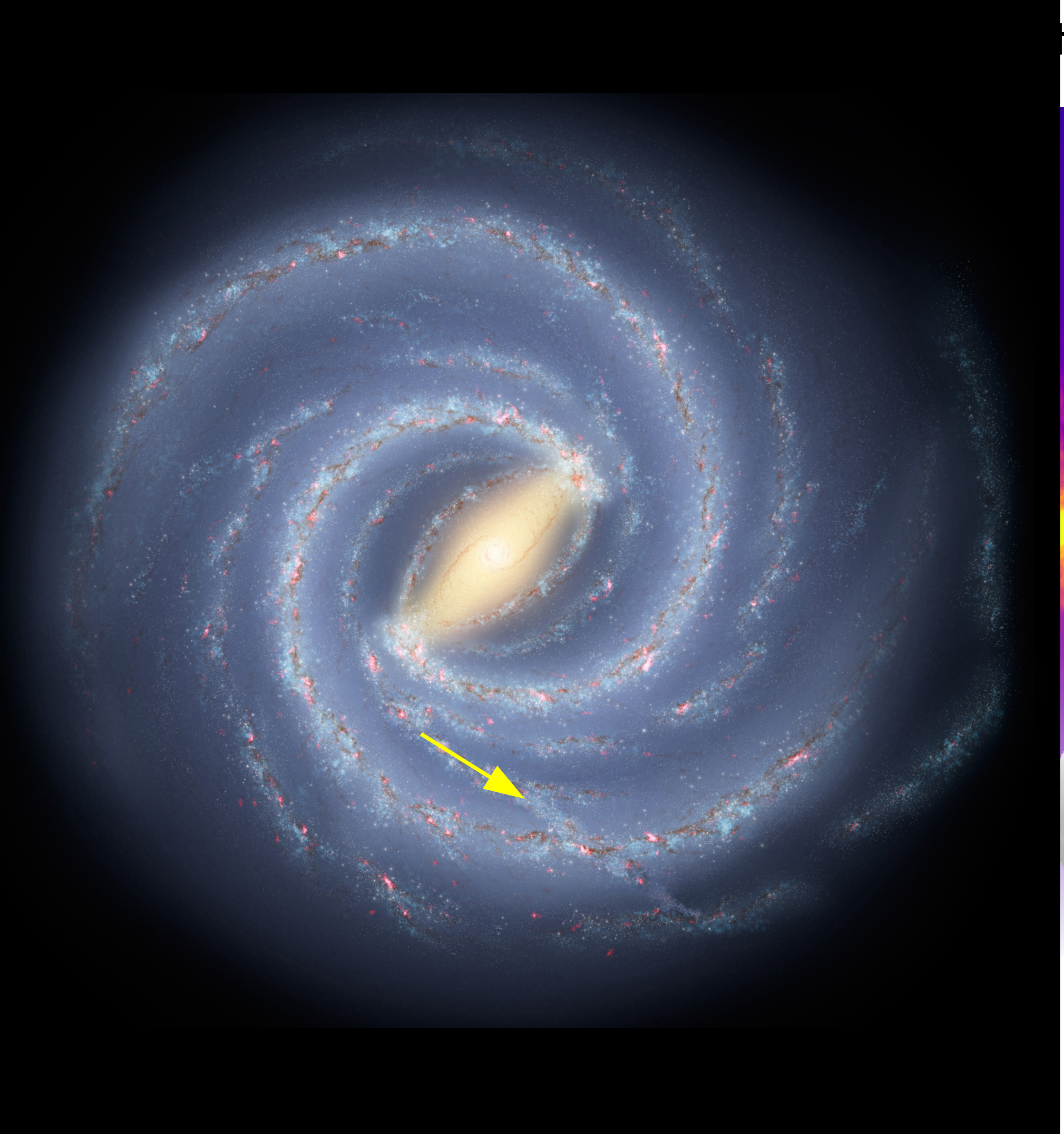
Faraday Amplitude Field

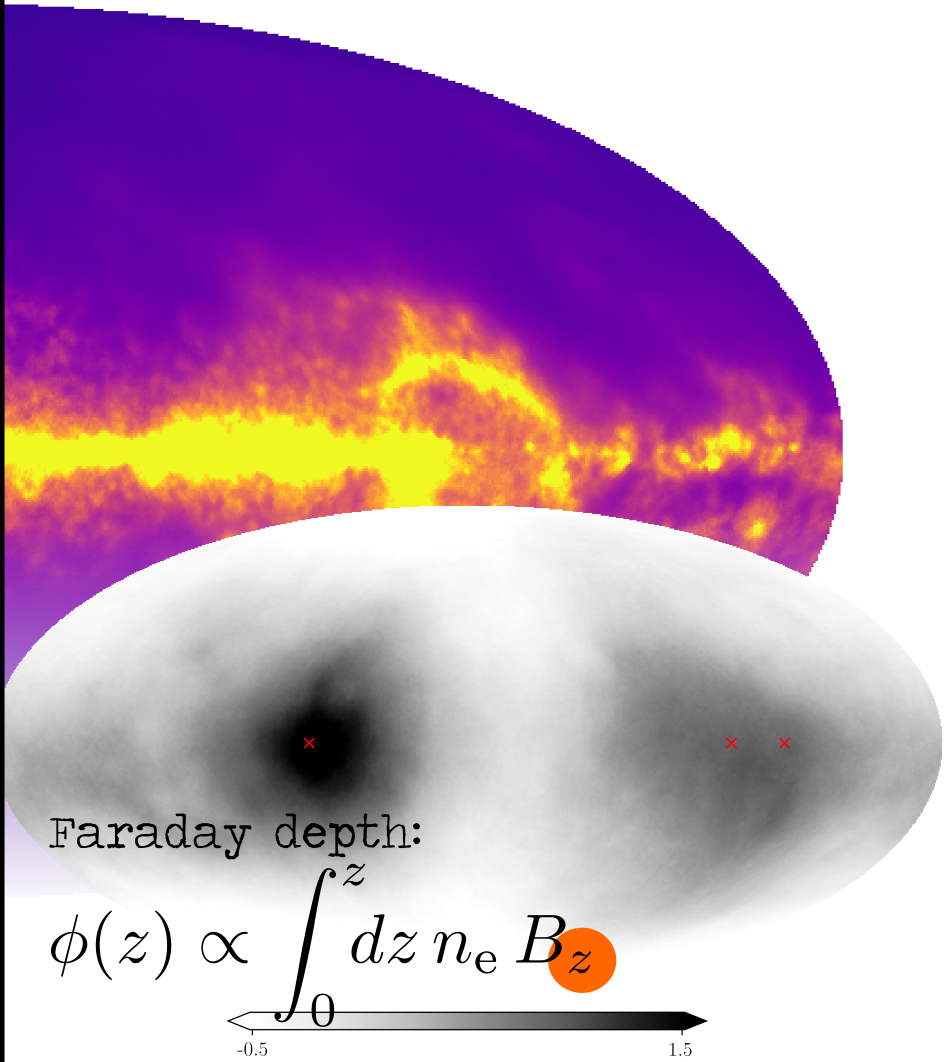
Hutschenreuter & Enßlin (2019)



tude Field

Hutschenreuter & Enßlin (2019)





Faraday depth:

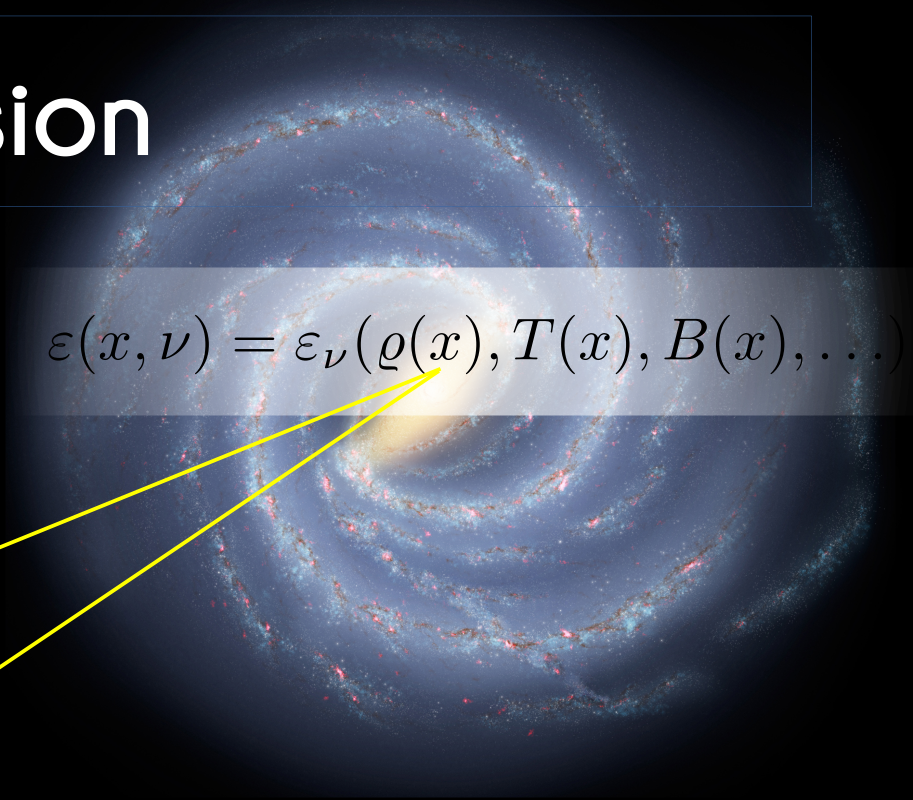
$$\phi(z) \propto \int_0^z dz\, n_\mathrm{e}\, B_z$$

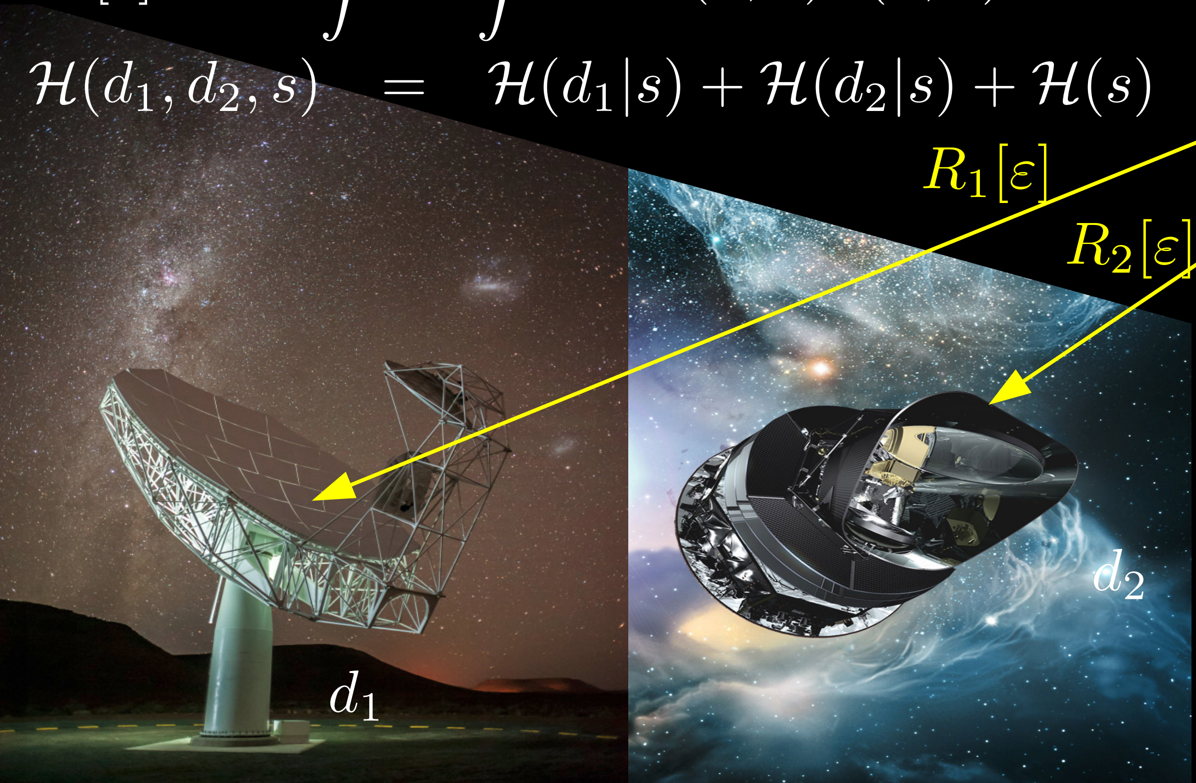
Data Fusion

$$d_i = R_i[\varepsilon] + n_i$$

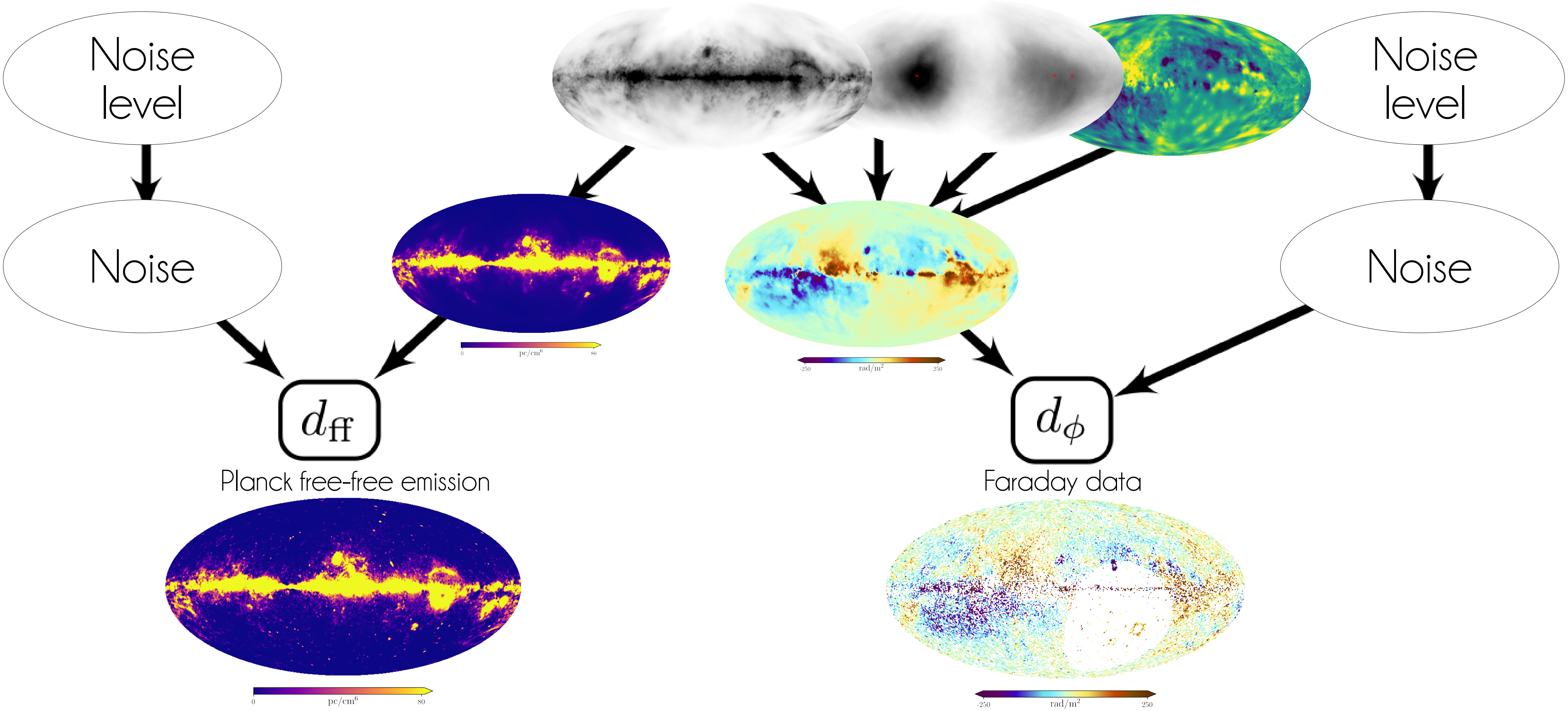
$$R_i[\varepsilon] = \int \mathrm{d}x \int \mathrm{d}\nu \, R_i(x,\nu)\, \varepsilon(x,\nu)$$

$$\mathcal{H}(d_1, d_2, s) = \mathcal{H}(d_1|s) + \mathcal{H}(d_2|s) + \mathcal{H}(s)$$



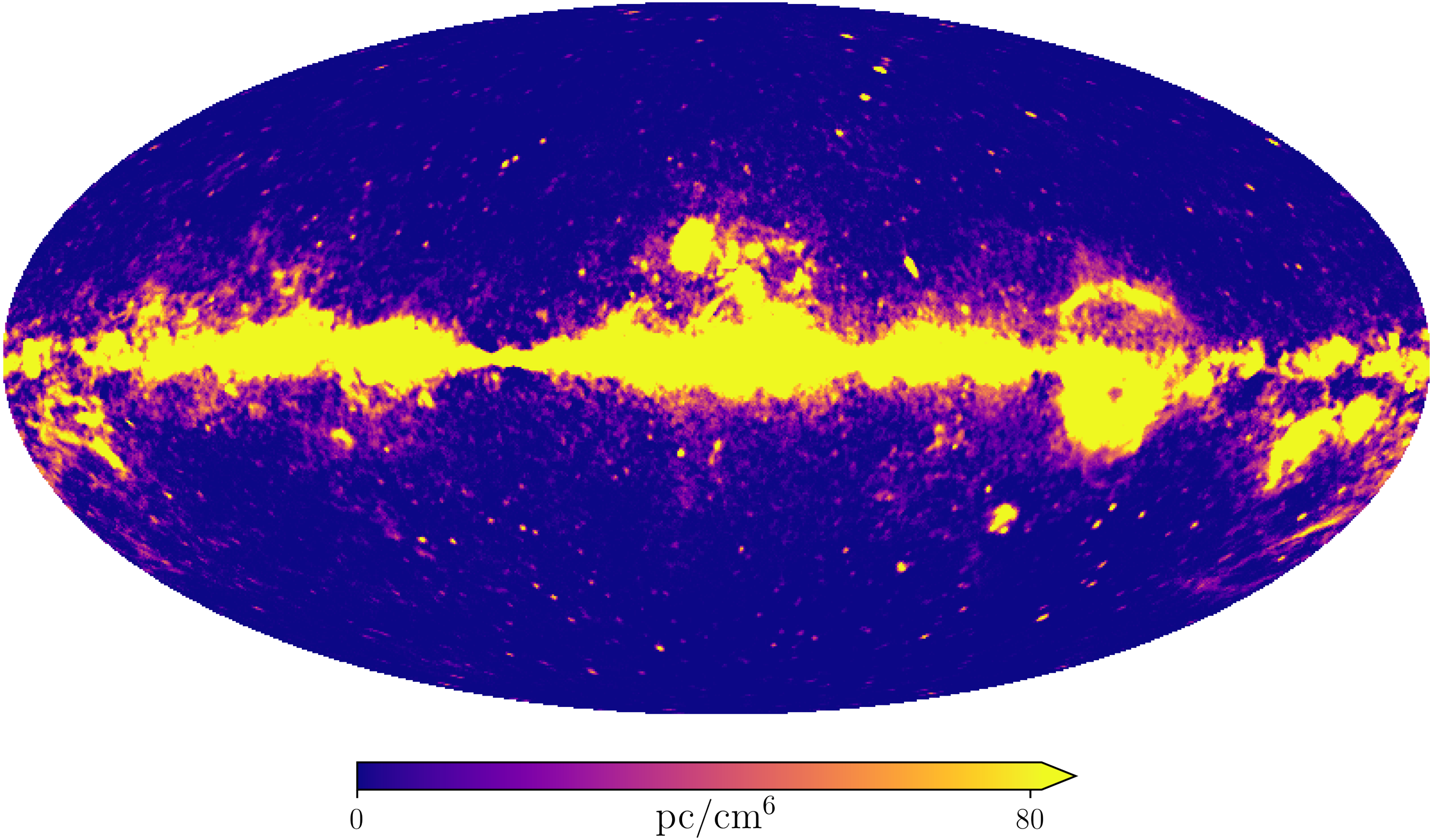


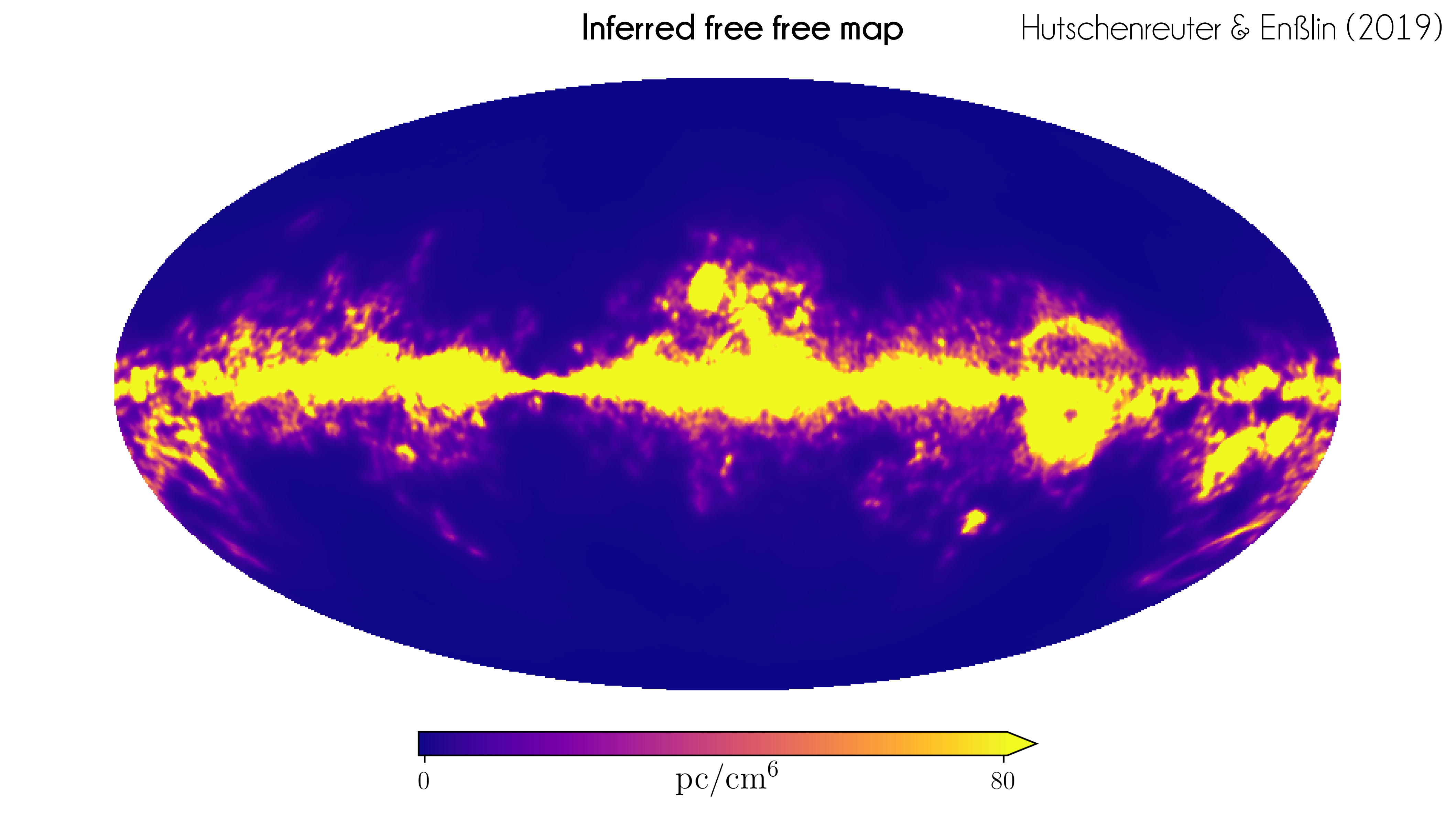
Hierarchical Bayesian Model



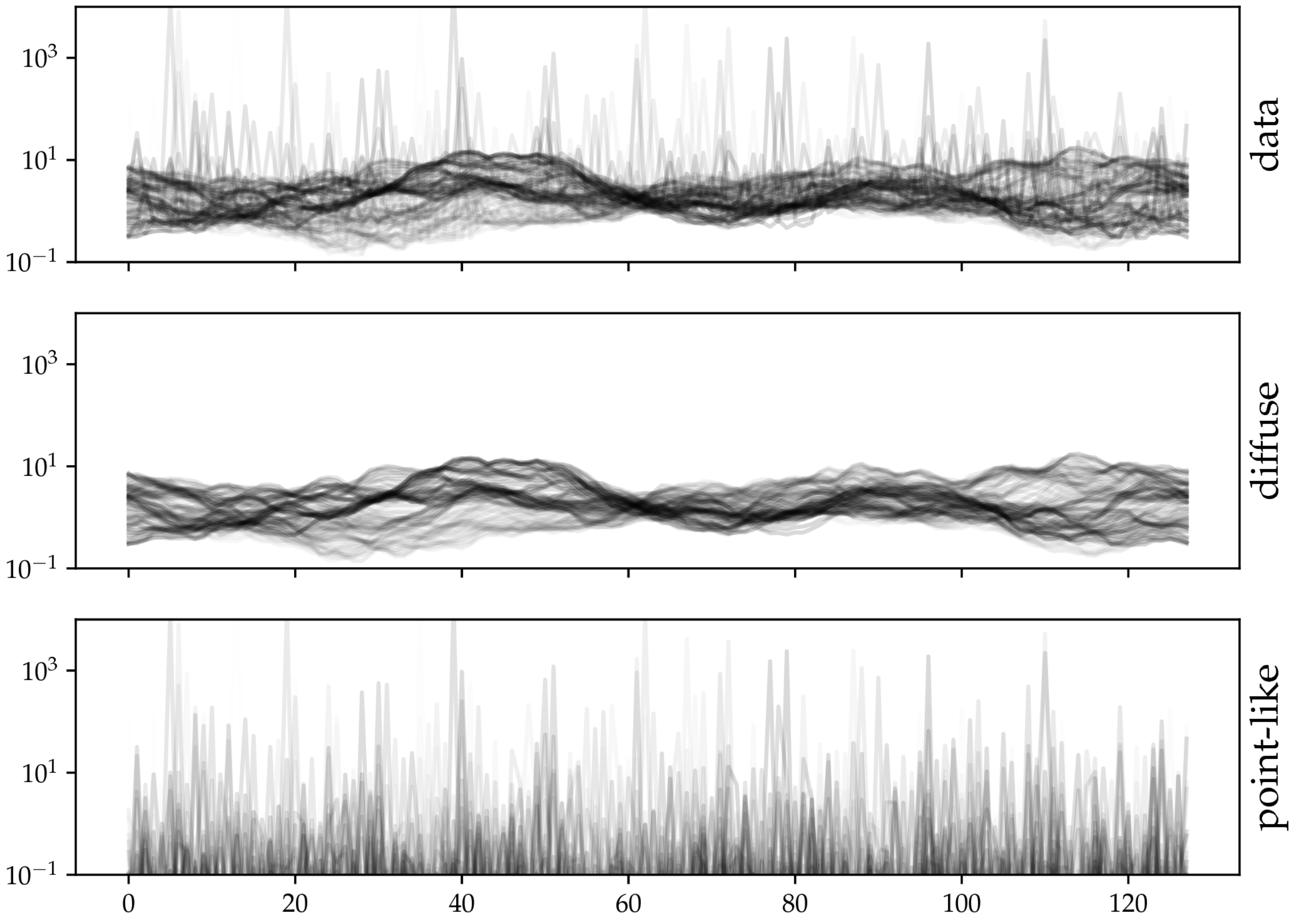
Planck free free map

Hutschenreuter & Enßlin (2019)

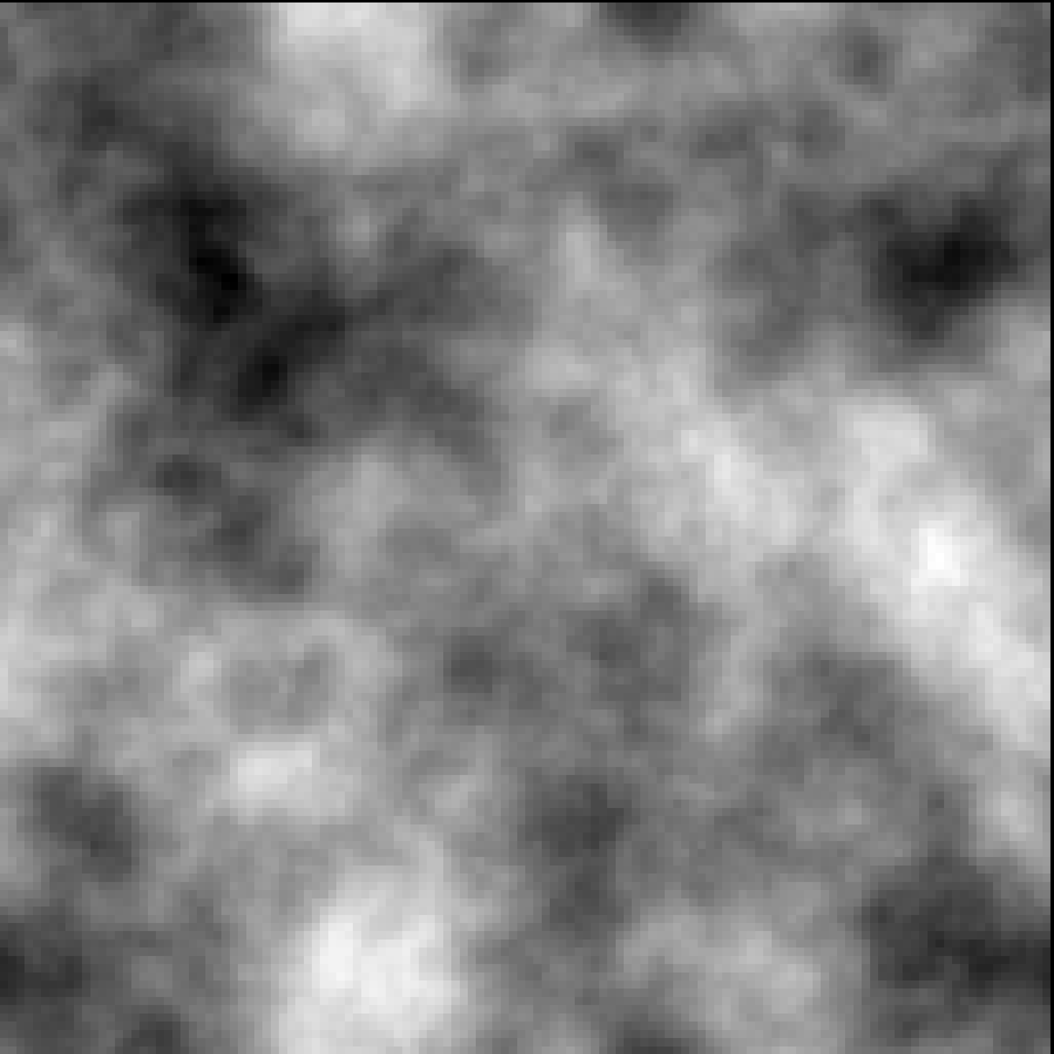



Inferred free free map
Hutschenreuter & Enßlin (2019)
0
pc/cm^6
80

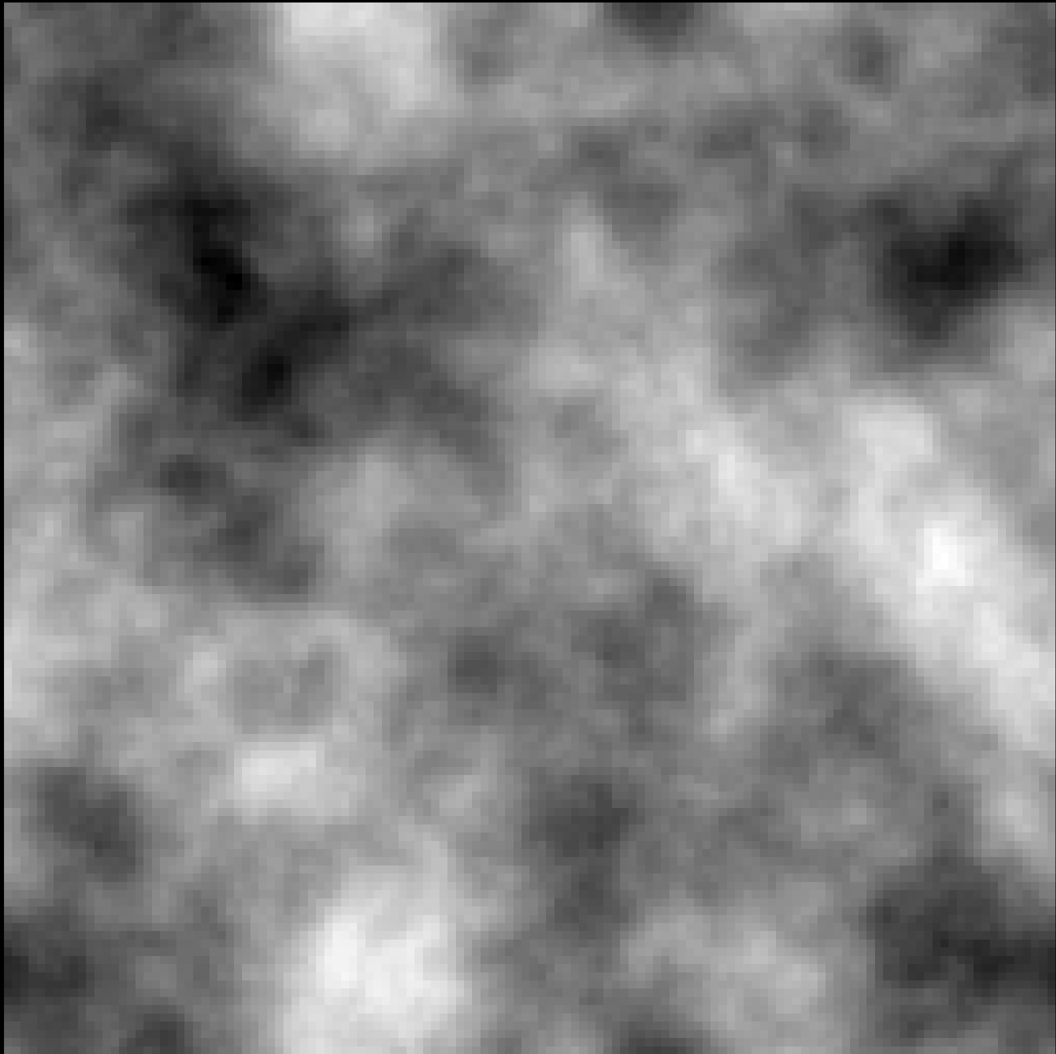
data and true components



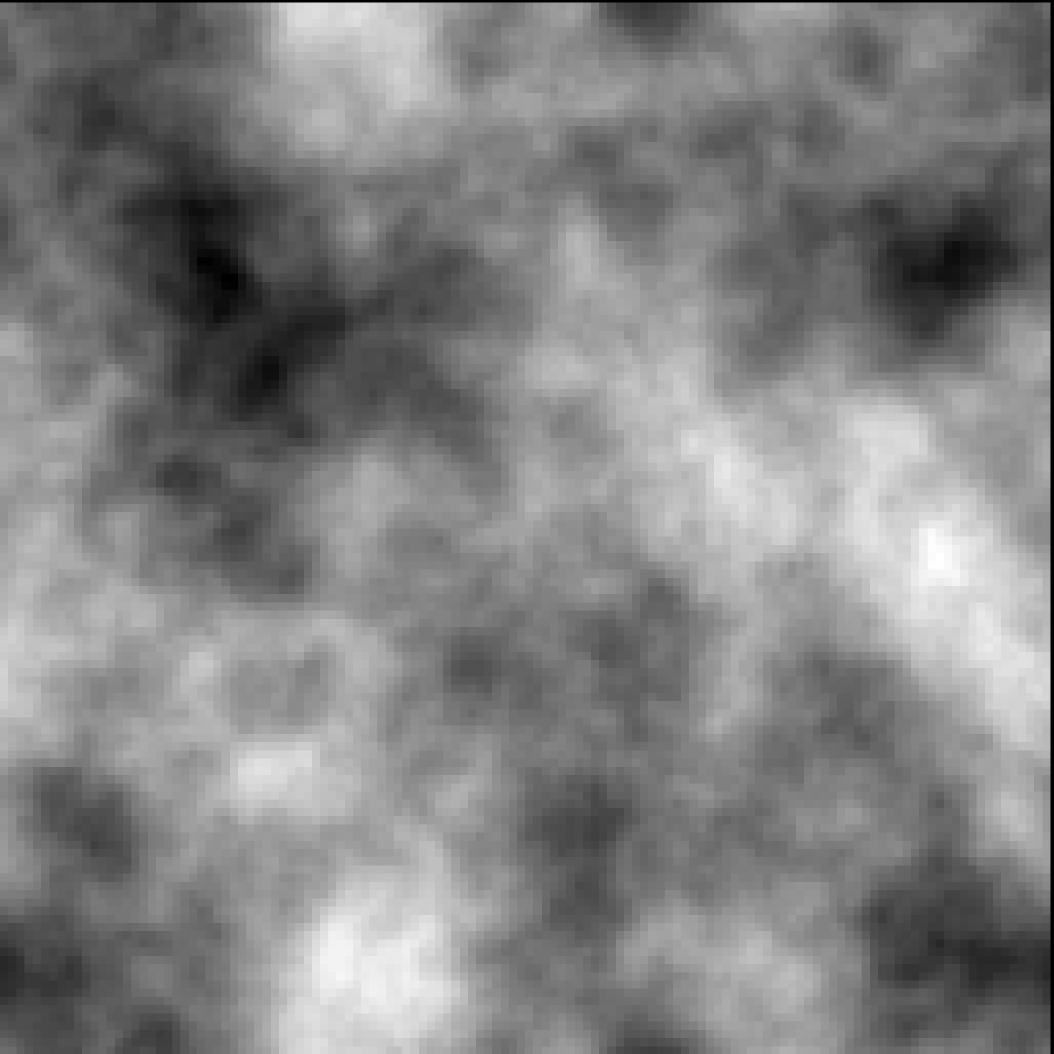
ground truth / starblade



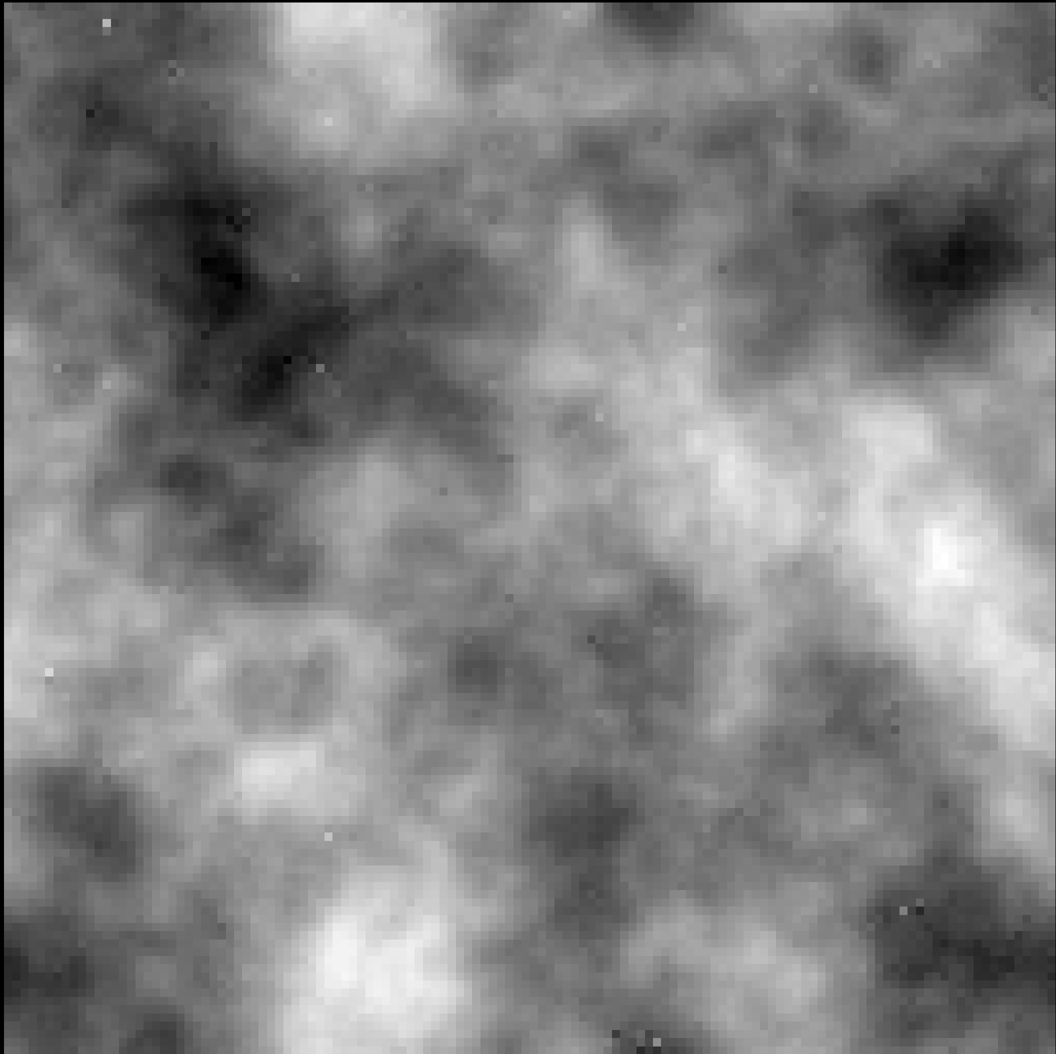
ground truth / autoencoder

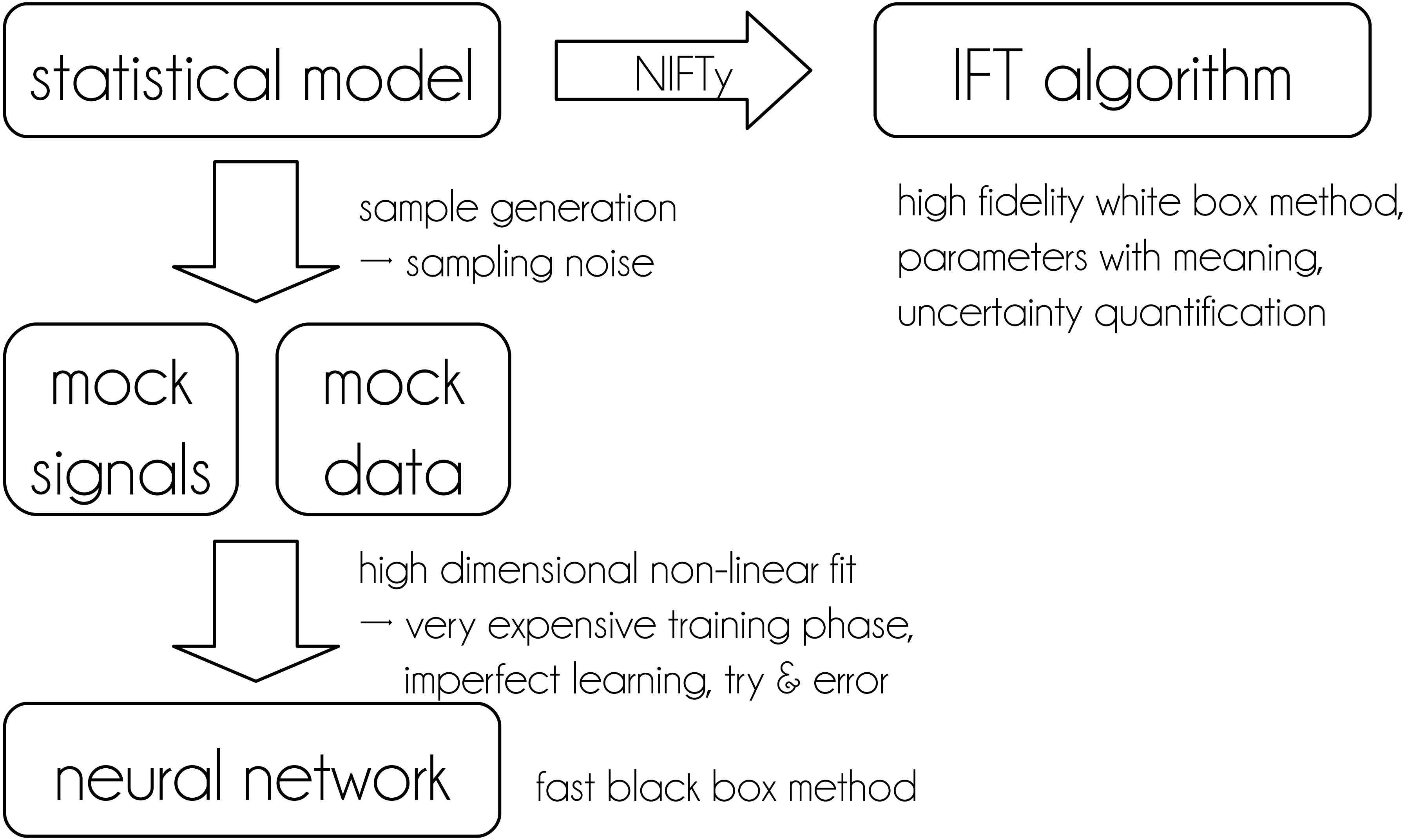


ground truth / starblade



ground truth / autoencoder




statistical model
NIFTy
IFT algorithm
sample generation
→ sampling noise
high fidelity white box method,
parameters with meaning,
uncertainty quantification
mock signals
mock data
high dimensional non-linear fit
→ very expensive training phase,
imperfect learning, try & error
neural network
fast black box method

NIFTy tutorial part 2

nonlinear reconstructions