SUSY 2024

Theory meets Experiment

Madrid, 10 – 14 June 2024

Modular tool for bayesian analysis with Machine Learning in DM Direct Detection

Andres Daniel Perez

Instituto de Física Teórica UAM-CSIC

In collaboration with David Cerdeño and Martín de los Rios

arXiv: 2406.XXXXX

Instituto de Física Teórica UAM-CSIC

Universidad Autónoma de Madrid

Outline

Reconstruct the DM parameters with DD experiments

Combine different data in a fast and simple way using Machine Learning.

Outline

Reconstruct the DM parameters with DD experiments

Combine different data in a fast and simple way using Machine Learning.

- Bayesian analysis with ML analysis to obtain posteriors
- Data sample generation:
  - DM-nucleon interaction with NR-EFT
  - Simulate the expected signal
- Parameter space that can be reconstructed

Bayesian Analysis

Bayesian Analysis

Bayes theorem:

Posterior probability of the parameters Θ of interest given the data X

Probability of the data X given the parameters Θ (likelihood)

Prior probability of the parameters Θ

$$P(\boldsymbol{\Theta}|\mathbf{X}) = \frac{P(\mathbf{X}|\boldsymbol{\Theta})P(\boldsymbol{\Theta})}{P(\mathbf{X})}$$

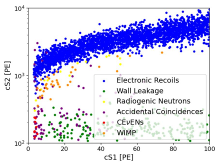
Probability of the data X also call evidence

Parameters

e.g. $(\sigma_i^{SI}, \theta_i, m_{DM})$

Data

Number of events or



Bayesian Analysis

Bayes theorem:

Traditional methods: need to assume a likelihood function

Probability of the data X given the parameters Θ (likelihood)

Posterior probability of the parameters Θ of interest given the data X

Prior probability of the parameters Θ

$$P(\boldsymbol{\Theta}|\mathbf{X}) = \frac{P(\mathbf{X}|\boldsymbol{\Theta})P(\boldsymbol{\Theta})}{P(\mathbf{X})}$$

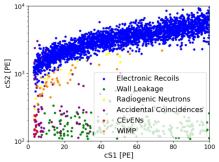
Probability of the data X also call evidence

Parameters

e.g. $(\sigma_i^{SI}, \theta_i, m_{DM})$

Data

Number of events or



Conditional probabilities

$$P(\mathbf{X}|\mathbf{\Theta}) = \frac{P(\mathbf{X}, \mathbf{\Theta})}{P(\mathbf{\Theta})}$$

$$P(\mathbf{\Theta}|\mathbf{X}) = \frac{P(\mathbf{X}, \mathbf{\Theta})}{P(\mathbf{X})}$$

Bayesian Analysis

Likelihood-to-evidence ratio:

$$r(\mathbf{X}, \mathbf{\Theta}) := \frac{P(\mathbf{X}|\mathbf{\Theta})}{P(\mathbf{X})} = \frac{P(\mathbf{\Theta}|\mathbf{X})}{P(\mathbf{\Theta})} = \frac{P(\mathbf{X}, \mathbf{\Theta})}{P(\mathbf{\Theta})P(\mathbf{X})}$$

Conditional probabilities

$$P(\mathbf{X}|\mathbf{\Theta}) = \frac{P(\mathbf{X}, \mathbf{\Theta})}{P(\mathbf{\Theta})}$$

$$P(\mathbf{\Theta}|\mathbf{X}) = \frac{P(\mathbf{X}, \mathbf{\Theta})}{P(\mathbf{X})}$$

Bayesian Analysis

Likelihood-to-evidence ratio:

$$r(\mathbf{X}, \mathbf{\Theta}) := \frac{P(\mathbf{X}|\mathbf{\Theta})}{P(\mathbf{X})} = \frac{P(\mathbf{\Theta}|\mathbf{X})}{P(\mathbf{\Theta})} = \frac{P(\mathbf{X}, \mathbf{\Theta})}{P(\mathbf{\Theta})P(\mathbf{X})}$$

compute the posterior →

$$P(\mathbf{\Theta}|\mathbf{X}) = r(\mathbf{X}, \mathbf{\Theta})\, P(\mathbf{\Theta})$$

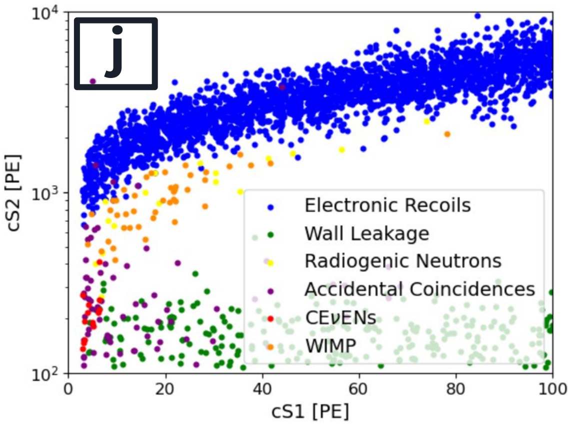
Bayesian Analysis with SWYFT

SWYFT ⟶ Sampling-based inference tool that estimates *likelihood-to-evidence ratio* with ML algorithms to obtain marginal and joint posteriors

Parameters

$$(\sigma^j, \theta^j, m^j_{DM})$$

Data

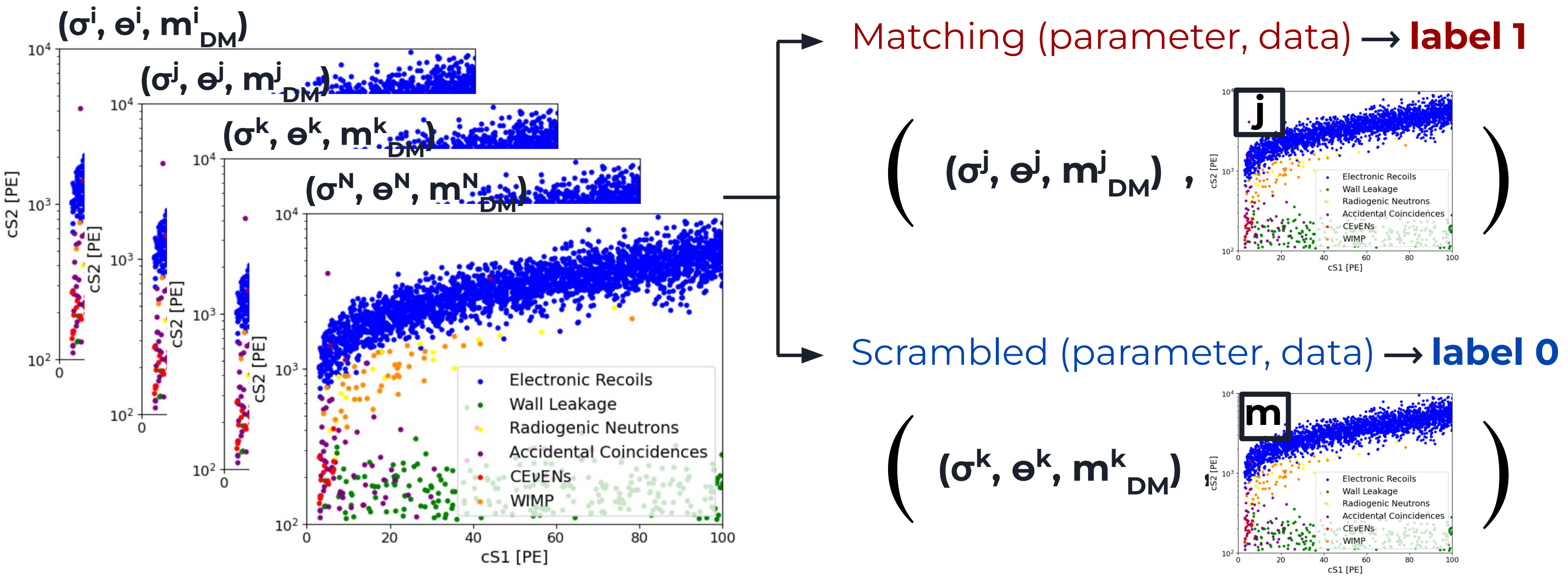


You can consider uncertainties:

- DM halo model,
- Nuclear response functions,
- Detector response …

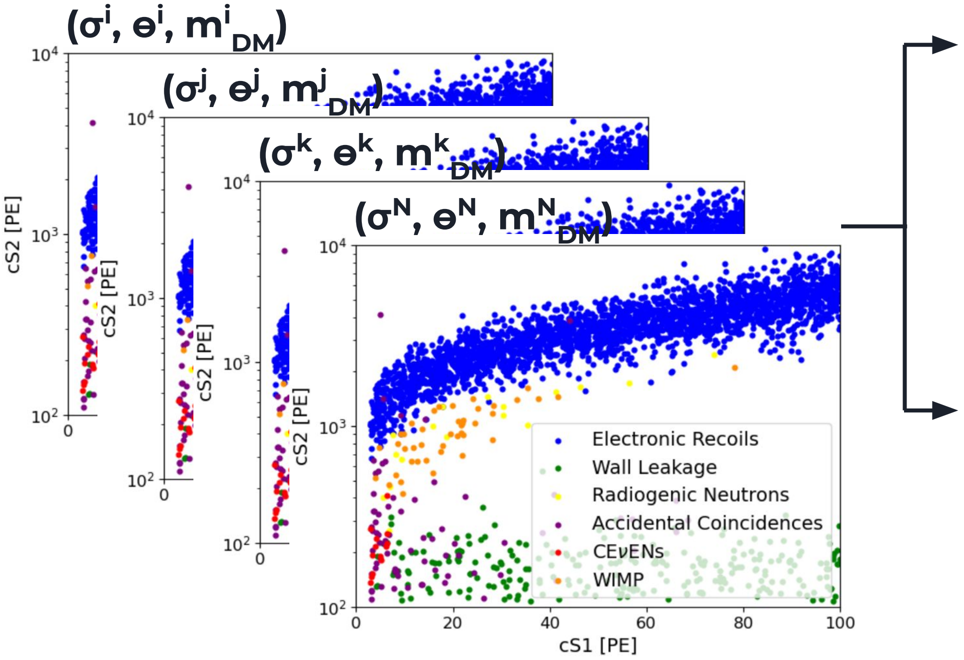
Bayesian Analysis with SWYFT

SWYFT ⟶ Sampling-based inference tool that estimates *likelihood-to-evidence ratio* with ML algorithms to obtain marginal and joint posteriors



Bayesian Analysis with SWYFT

SWYFT ⟶ Sampling-based inference tool that estimates *likelihood-to-evidence ratio* with ML algorithms to obtain marginal and joint posteriors



Matching (parameter, data) ⟶ **label 1**

Sampled from $P(\mathbf{X}, \boldsymbol{\Theta})$

Scrambled (parameter, data) ⟶ **label 0**

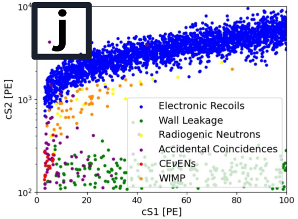
Sampled from $P(\boldsymbol{\Theta})P(\mathbf{X})$

Bayesian Analysis with SWYFT

SWYFT ⟶ Sampling-based inference tool that estimates *likelihood-to-evidence ratio* with ML algorithms to obtain marginal and joint posteriors

Matching (parameter, data) ⟶ **label 1**

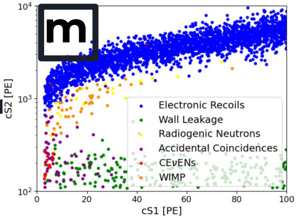
$$\left((\sigma^j, \theta^j, m^j_{DM}) \, , \right.$$



$$\left. \right)$$

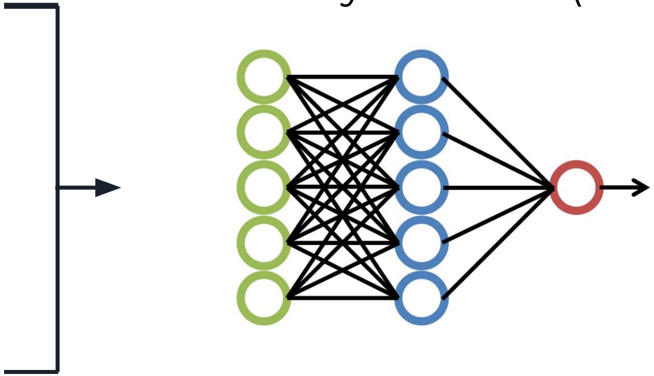
Scrambled (parameter, data) ⟶ **label 0**

$$\left((\sigma^k, \theta^k, m^k_{DM}) \, , \right.$$



$$\left. \right)$$

Binary classifier (DNN, CNN, ...)



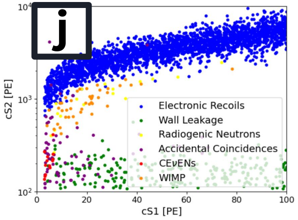
output ∈ [0,1]

Bayesian Analysis with SWYFT

SWYFT $\longrightarrow$ Sampling-based inference tool that estimates *likelihood-to-evidence ratio* with ML algorithms to obtain marginal and joint posteriors

Matching (parameter, data) $\longrightarrow$ **label 1**

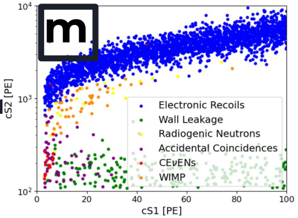
($(\sigma^j, \theta^j, m^j_{DM})$,



)

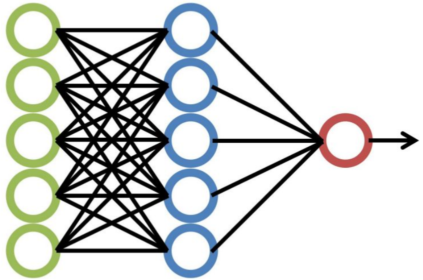
Scrambled (parameter, data) $\longrightarrow$ **label 0**

($(\sigma^k, \theta^k, m^k_{DM})$,



)

Binary classifier (DNN, CNN, ...)



output $\in [0,1]$

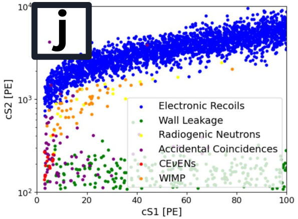
label 1

output $= P(k=1|\mathbf{X},\mathbf{\Theta}) \longrightarrow P(\mathbf{X},\mathbf{\Theta})$

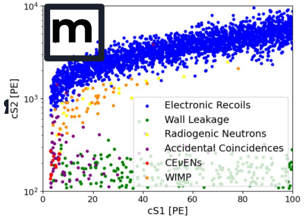
Bayesian Analysis with SWYFT

SWYFT ⟶ Sampling-based inference tool that estimates *likelihood-to-evidence ratio* with ML algorithms to obtain marginal and joint posteriors

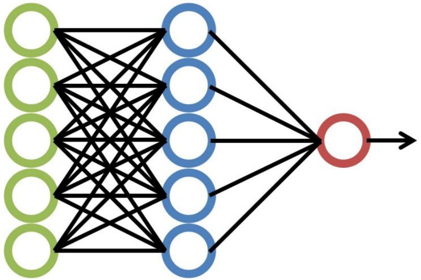
Matching (parameter, data) ⟶ **label 1**

$\Big((\sigma^j, \theta^j, m^j_{DM})$,  $\Big)$

Scrambled (parameter, data) ⟶ **label 0**

$\Big((\sigma^k, \theta^k, m^k_{DM})$,  $\Big)$

Binary classifier (DNN, CNN, ...)



output ∈ [0,1]

label 1

output $= P(k=1|\mathbf{X},\mathbf{\Theta}) \rightarrow P(\mathbf{X},\mathbf{\Theta})$

label 0

1 - output $= P(k=0|\mathbf{X},\mathbf{\Theta}) \rightarrow P(\mathbf{\Theta})P(\mathbf{X})$

Bayesian Analysis with SWYFT

SWYFT $\longrightarrow$ Sampling-based inference tool that estimates *likelihood-to-evidence ratio* with ML algorithms to obtain marginal and joint posteriors

$$r(\mathbf{X}, \mathbf{\Theta}) = \frac{\textbf{output}}{1 - \textbf{output}}$$

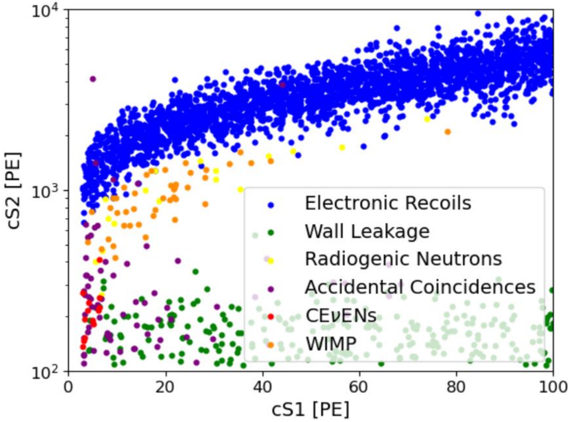
compute the posterior $\rightarrow$

$$P(\mathbf{\Theta}|\mathbf{X}) = r(\mathbf{X}, \mathbf{\Theta})\, P(\mathbf{\Theta})$$

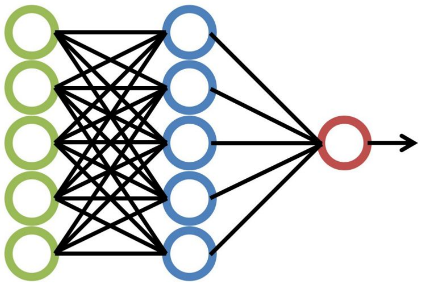
Bayesian Analysis with SWYFT

SWYFT ⟶ Sampling-based inference tool that estimates *likelihood-to-evidence ratio* with ML algorithms to obtain marginal and joint posteriors

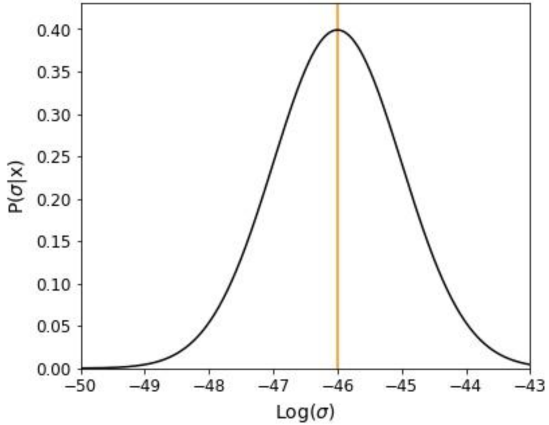
New data sample x^{new}



Trained binary classifier



Posterior $P(\sigma|x^{new})$



For another data sample ⟶ we do **not** need to train everything again, use the same classifier

Combined Bayesian Analysis

Posterior using an experimental dataset X_1 (e.g. images)

non-informative prior

$$P(\mathbf{\Theta}|\mathbf{X}_1) = \frac{P(\mathbf{X}_1|\mathbf{\Theta})P(\mathbf{\Theta})}{P(\mathbf{X}_1)} = r(\mathbf{X}_1, \mathbf{\Theta})P(\mathbf{\Theta})$$

Combined Bayesian Analysis

Posterior using an experimental dataset X_1 (e.g. images)

non-informative prior

$$P(\boldsymbol{\Theta}|\mathbf{X}_1) = \frac{P(\mathbf{X}_1|\boldsymbol{\Theta})P(\boldsymbol{\Theta})}{P(\mathbf{X}_1)} = r(\mathbf{X}_1, \boldsymbol{\Theta})P(\boldsymbol{\Theta})$$

Posterior using an experimental dataset X_2 (e.g. spectra)

use the previous posterior as prior

$$P(\boldsymbol{\Theta}|\mathbf{X}_2) = \frac{P(\mathbf{X}_2|\boldsymbol{\Theta})P(\boldsymbol{\Theta})}{P(\mathbf{X}_2)} = r(\mathbf{X}_2, \boldsymbol{\Theta})P(\boldsymbol{\Theta}) \longrightarrow r(\mathbf{X}_2, \boldsymbol{\Theta})P(\boldsymbol{\Theta}|\mathbf{X}_1)$$

Combined Bayesian Analysis

Posterior using an experimental dataset X_1 (e.g. images)

non-informative prior

$$P(\Theta|\mathbf{X}_1) = \frac{P(\mathbf{X}_1|\Theta)P(\Theta)}{P(\mathbf{X}_1)} = r(\mathbf{X}_1, \Theta)P(\Theta)$$

Posterior using an experimental dataset X_2 (e.g. spectra)

use the previous posterior as prior

$$P(\Theta|\mathbf{X}_2) = \frac{P(\mathbf{X}_2|\Theta)P(\Theta)}{P(\mathbf{X}_2)} = r(\mathbf{X}_2, \Theta)P(\Theta) \longrightarrow r(\mathbf{X}_2, \Theta)P(\Theta|\mathbf{X}_1)$$

Product of likelihood-to-evidence ratios

$$P(\Theta|\mathbf{X}_2) = r(\mathbf{X}_2, \Theta)\, r(\mathbf{X}_1, \Theta)\, P(\Theta)$$

Combined Bayesian Analysis

Product of likelihood-to-evidence ratios

$$P(\boldsymbol{\Theta}|\mathbf{X}_n) = \prod_{i=1}^{n} r(\mathbf{X}_i, \boldsymbol{\Theta}) P(\boldsymbol{\Theta})$$

each one obtained training SWYFT with only the X_i dataset

To include new data you do not need to re-train everything. Train with your new data and include $r(X_{new}, \Theta)$

To remove a dataset do not include $r(X_{old}, \Theta)$

Data sample generation

DM-nucleon effective field theory (NR-EFT)

For DM particles with spin up to ½ , the effective DM-nucleon scattering interaction Lagrangian

$$\mathcal{L}_{\mathrm{EFT}} = \sum_{\tau} \sum_{i} c_i^{\tau} \mathcal{O}_i \bar{\chi}\chi\bar{\tau}\tau$$

i=14 possible interactions

O_1: spin-independent (SI)
O_4: spin-dependent (SD)

usually shown assuming isoscalar interactions

$c^p = c^n$ $c^0 = 1$ and $c^1 = 0$

Change to polar coordinates:

$$c_i^0 = \frac{1}{2}(c_i^p + c_i^n) = A_i \sin(\theta_i)$$

$$c_i^1 = \frac{1}{2}(c_i^p - c_i^n) = A_i \cos(\theta_i)$$

$$\sigma_i \propto A_i^2$$

isospin basis
c^0: isoscalar
c^1: isovector

nucleon basis
c^p: proton
c^n: neutron

DM-nucleon effective field theory (NR-EFT)

For DM particles with spin up to ½ , the effective DM-nucleon scattering interaction Lagrangian

$$\mathcal{L}_{\rm EFT} = \sum_{\tau} \sum_{i} c_i^{\tau} \mathcal{O}_i \bar{\chi}\chi\bar{\tau}\tau$$

i=14 possible interactions

O_1: spin-independent (SI)
O_4: spin-dependent (SD)

Change to polar coordinates:

$$c_i^0 = \frac{1}{2}(c_i^p + c_i^n) = A_i \sin(\theta_i)$$

$$c_i^1 = \frac{1}{2}(c_i^p - c_i^n) = A_i \cos(\theta_i)$$

$$\sigma_i \propto A_i^2$$

isospin basis
c^0: isoscalar
c^1: isovector

nucleon basis
c^p: proton
c^n: neutron

For each operator
2 parameters:

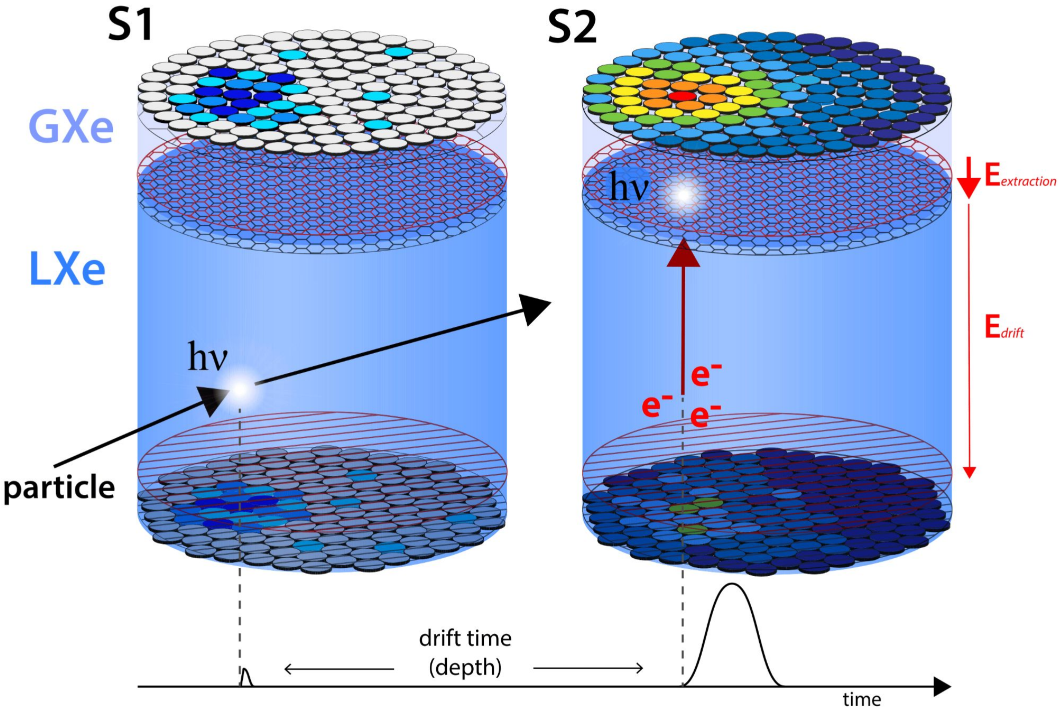
- amplitude (cross-section)
- phase

+ DM mass

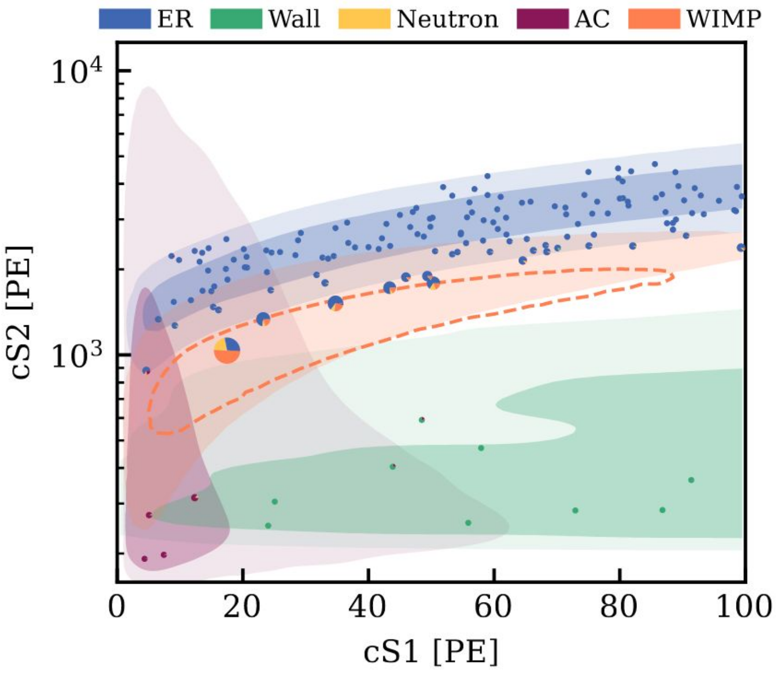
(σ_i, θ_i, m_{DM})

DM signal

XENONnT



Universe **2021**, *7*(8), 313



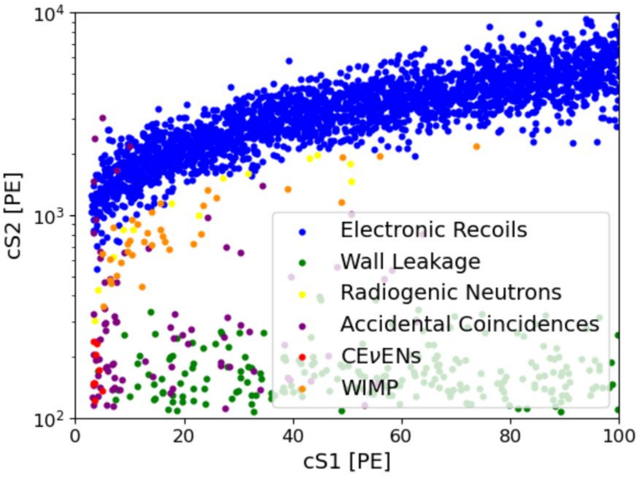
Phys. Rev. Lett. 131, 041003 (2023)

XENONnT 20ty simulator

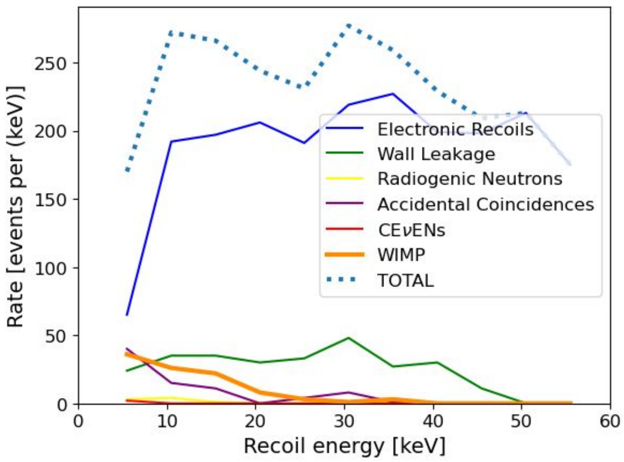
We specify background and signal characteristics

Data Representation:

cS1 vs cS2 plane



differential rate



number of events

	name	pseudo_exp_events
0	er	2459
1	radiogenics	17
2	ac	71
3	wall	246
4	WIMP	43
5	CEVNS-SM	13

WIMPs: differential rate compute with WimPyDD for a particular operator, (σ_i, θ_i, m_{DM}), and standard DM halo model

O1: $\sigma = 10^{-47} cm^2$
$\theta = \pi/2 \quad (c^p = c^n)$
$m = 50 GeV$

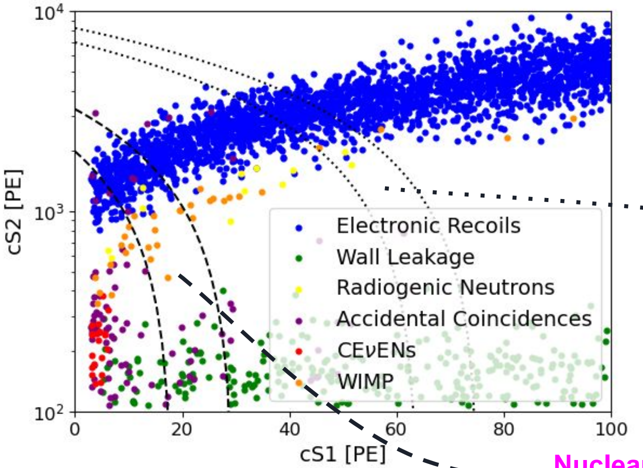
CEνENs: differential rate compute with SNuDD

Data Representation:

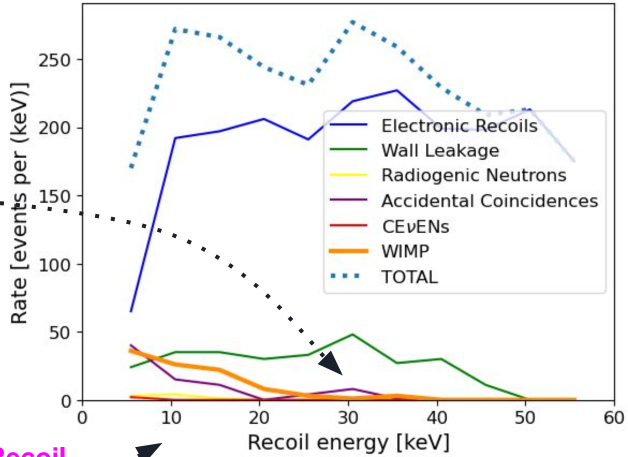
XENONnT 20ty simulator

We specify background and signal characteristics

cS1 vs cS2 plane



differential rate



Nuclear Recoil isoenergy curves

number of events

	name	pseudo_exp_events
0	er	2459
1	radiogenics	17
2	ac	71
3	wall	246
4	WIMP	43
5	CEVNS-SM	13

WIMPs: differential rate compute with WimPyDD for a particular operator, (σ_i, θ_i, m_{DM}), and standard DM halo model

O1: $\sigma=10^{-47}cm^2$
$\theta=\pi/2 \quad (c^p=c^n)$
$m=50GeV$

CEvENs: differential rate compute with SNuDD

Results

Posterior

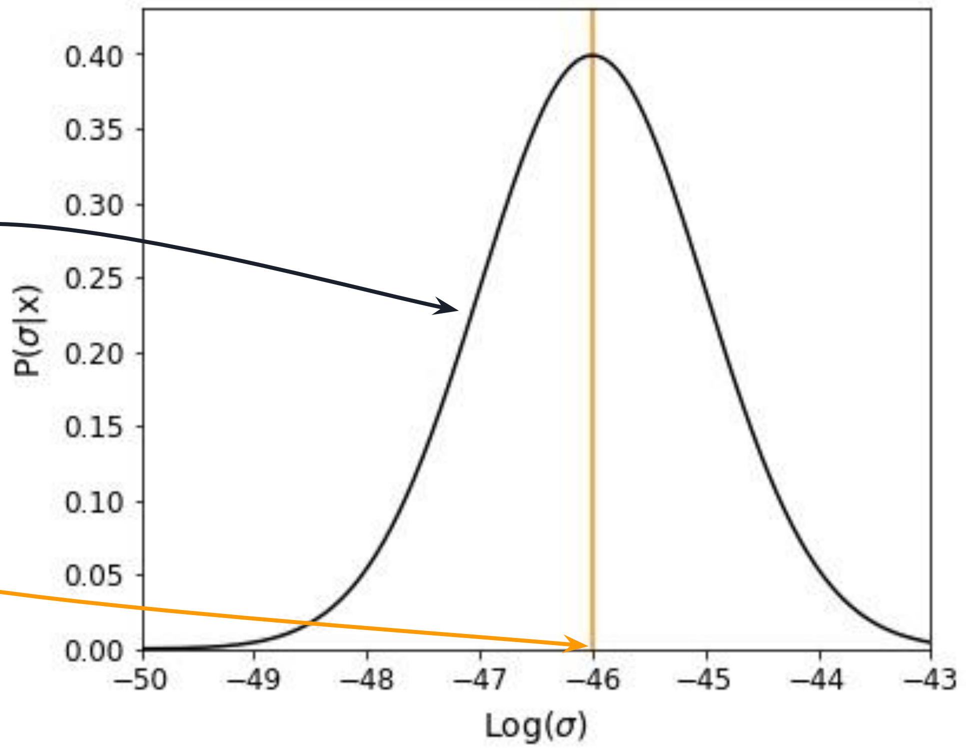
Compute the posterior for a new pseudo experiment

For example:

$P(\sigma|x)$

$\sigma^{true}=10^{-46}cm^2$

this is a gaussian as an example, not the actual posterior!



Posterior

Compute the posterior for a new pseudo experiment

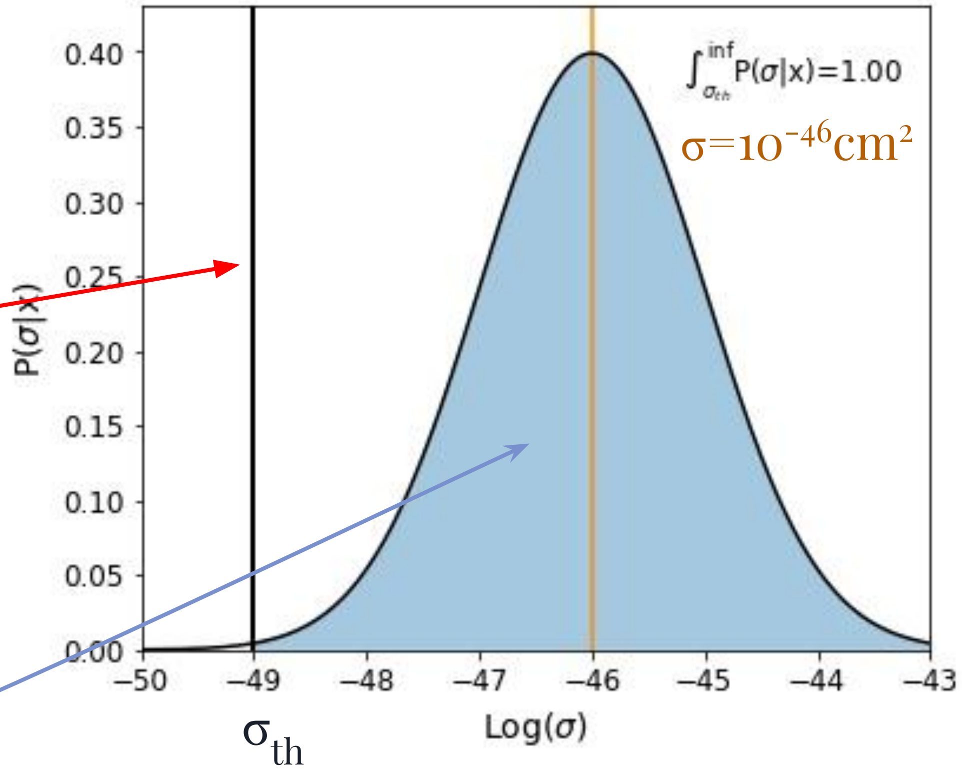
We define a σ_{th} threshold:

σ_{th}=$10^{-49}cm^2 \longrightarrow$ **NO SIGNAL!**

Then, we can ***reconstruct*** σ if:

$$\int_{\sigma_{th}}^{inf} P(\sigma|x) > 0.90$$

this is a gaussian as an example, not the actual posterior!



Posterior

Compute the posterior for a new pseudo experiment

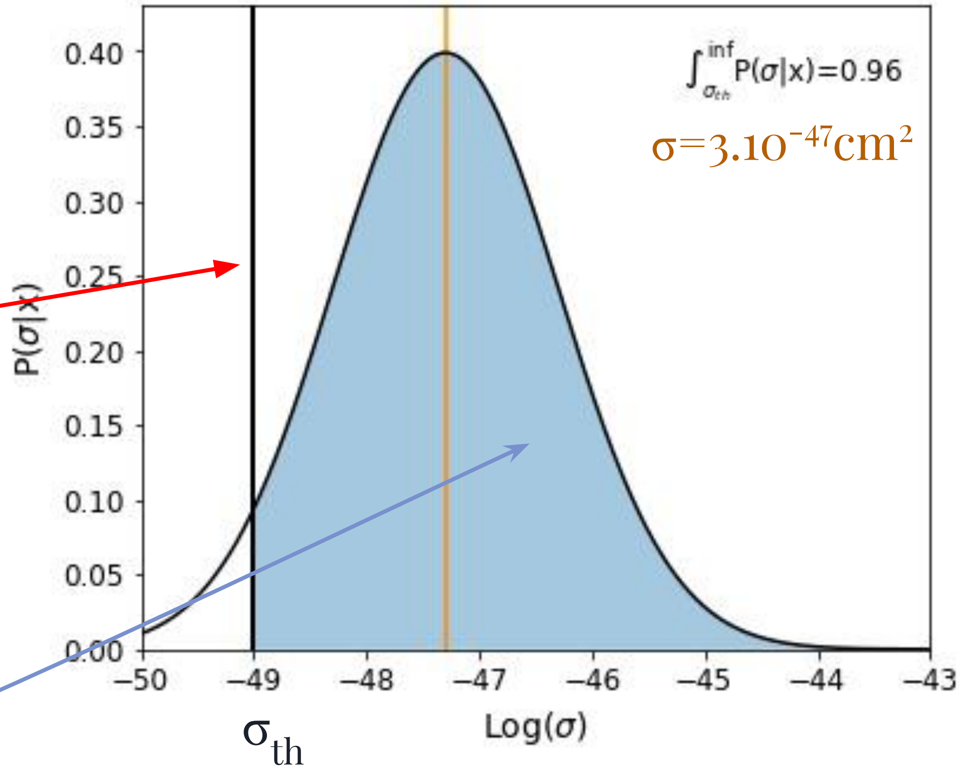
We define a σ_{th} threshold:

σ_{th}=10^{-49}cm^2 ⟶ **NO SIGNAL!**

Then, we can ***reconstruct*** σ if:

$$\int_{\sigma_{th}}^{inf} P(\sigma|x) > 0.90$$

this is a gaussian as an example, not the actual posterior!



Posterior

Compute the posterior for a new pseudo experiment

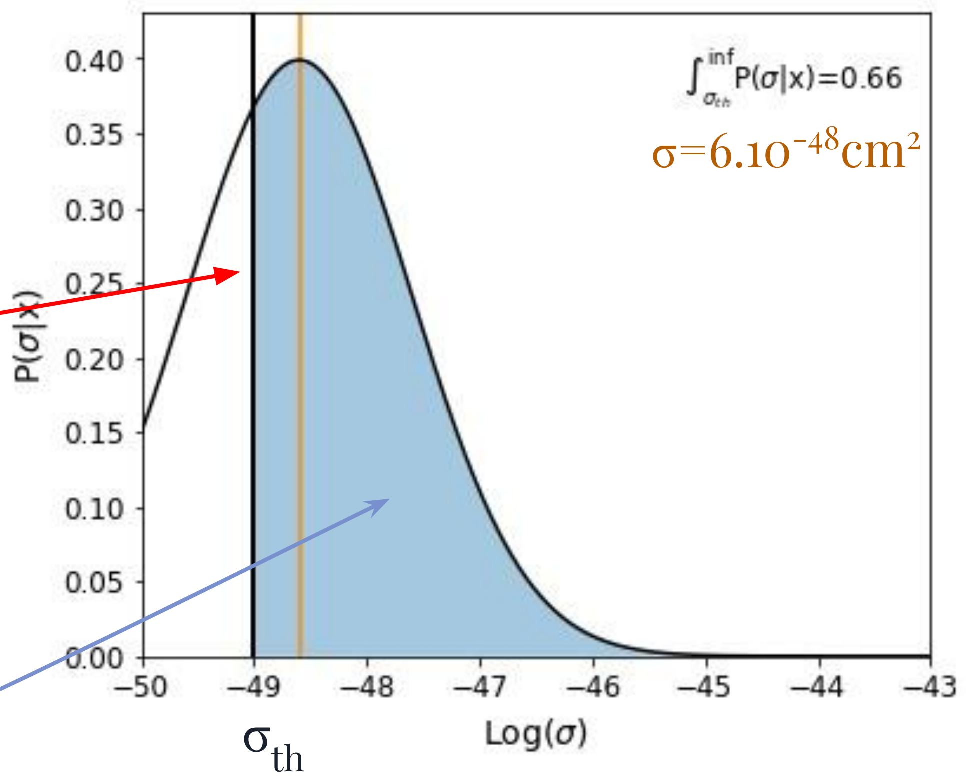
We define a σ_{th} threshold:

σ_{th}=10^{-49}cm^2 ⟶ **NO SIGNAL!**

Then, we can ***reconstruct*** σ if:

$$\int_{\sigma_{th}}^{inf} P(\sigma|x) > 0.90$$

this is a gaussian as an example, not the actual posterior!



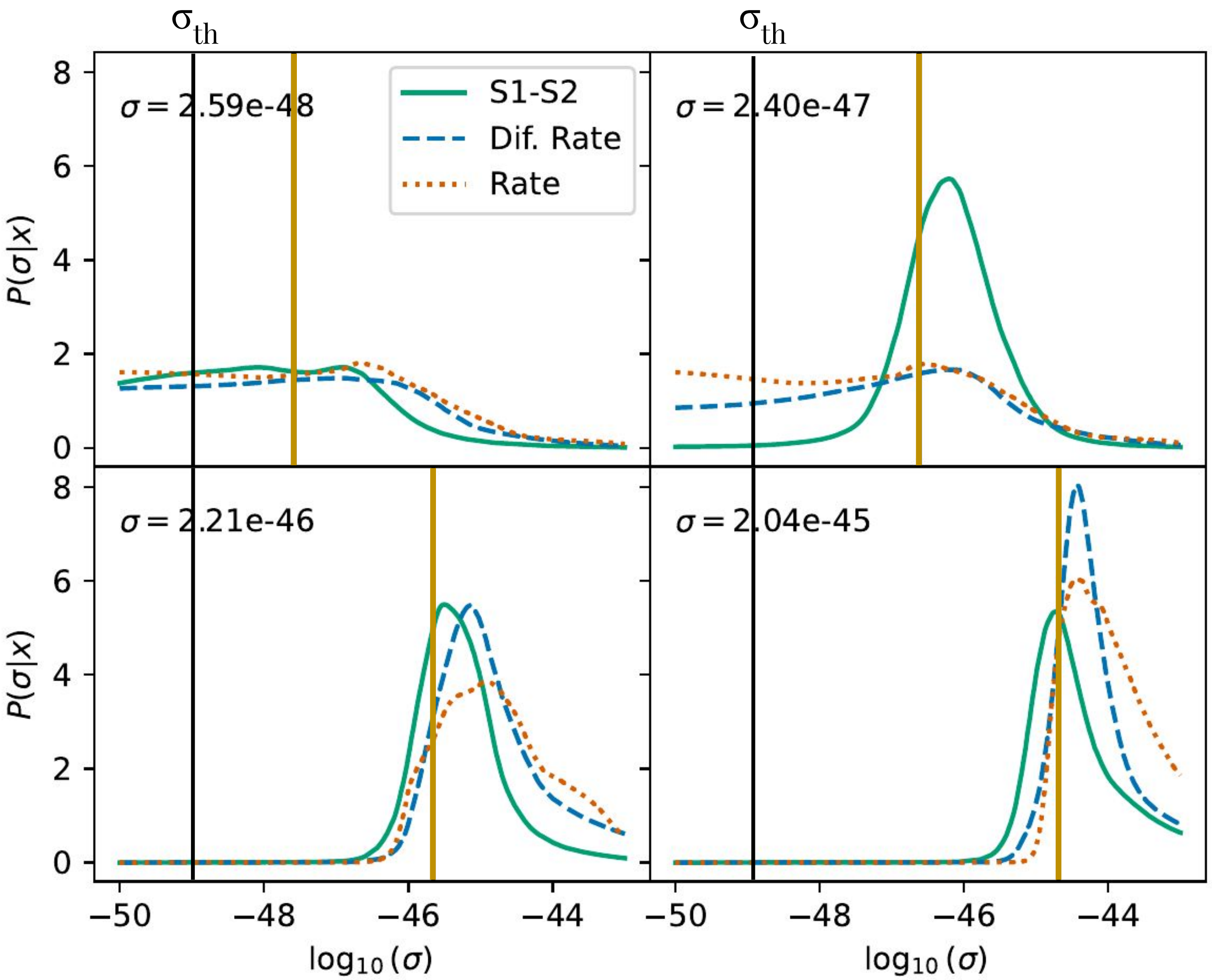
Results

Data:

cS1 vs cS2 plane
differential rate
total number of events

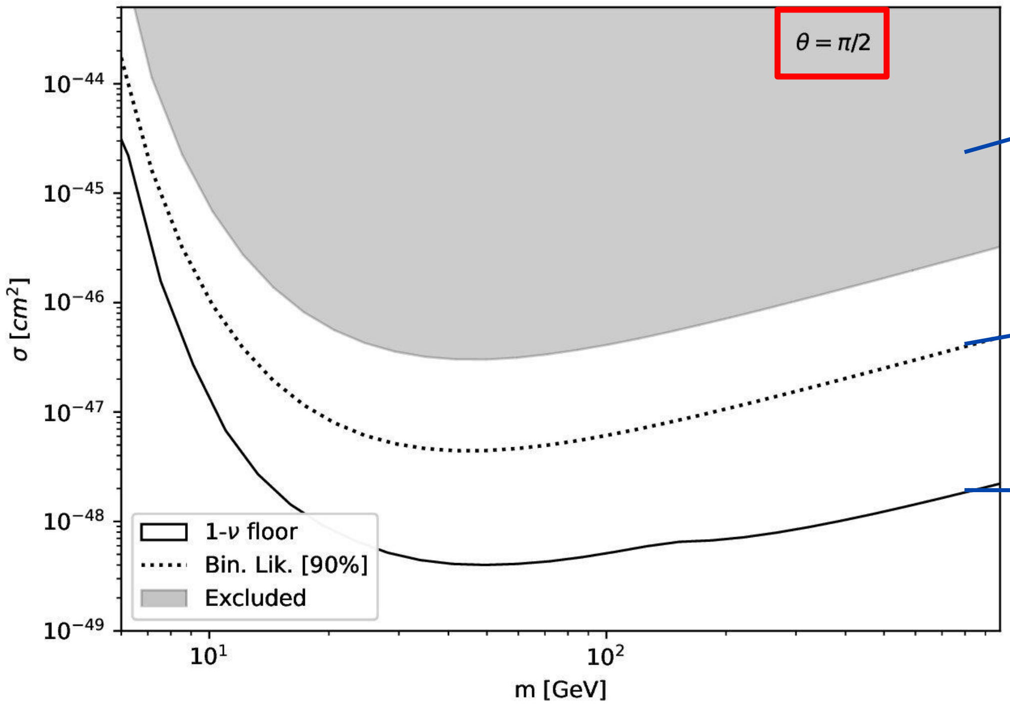
Posteriors for:

O_1 (SI)
m_{DM}≃100GeV ⟶ **fixed**
θ=π/2 ⟶ **fixed**



Results: parameter reconstruction

For O_1 with fixed θ=π/2



Exclusion:

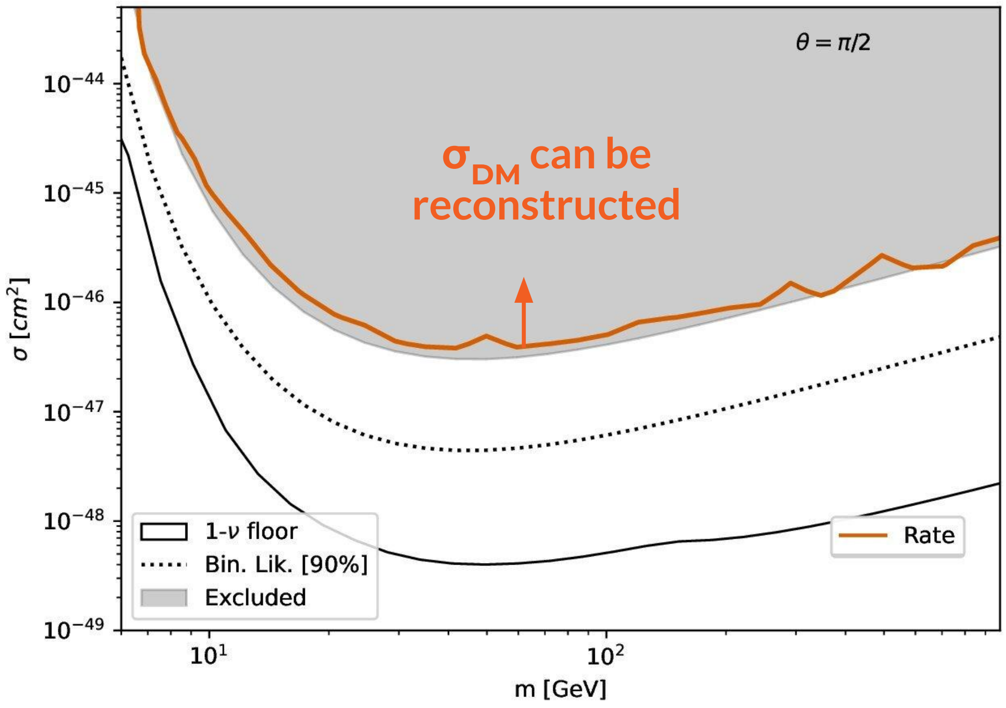
XENONnT 1 ton year expected **exclusion** limit (90% C.L.)

XENONnT 20 ton year proyected expected **exclusion** limit (90% C.L.)

1ν neutrino floor

Results: parameter reconstruction

For O_1 with fixed θ=π/2



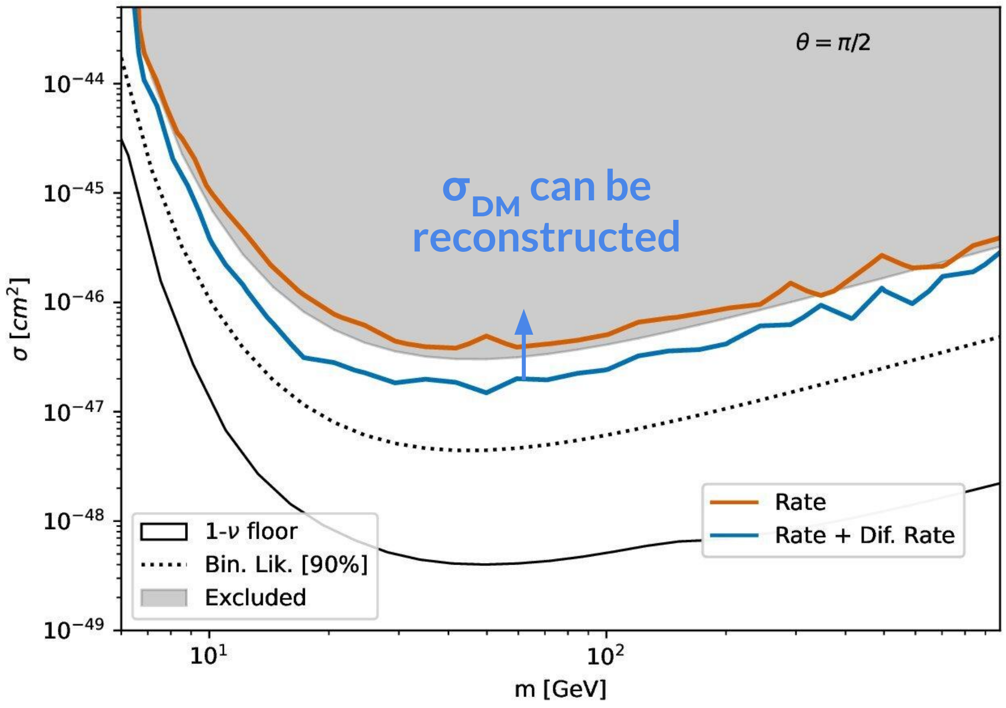
Reconstruction of the cross-section σ_{DM}

ML (SWYFT):

— **rate**

Results: parameter reconstruction

For O_1 with fixed θ=π/2



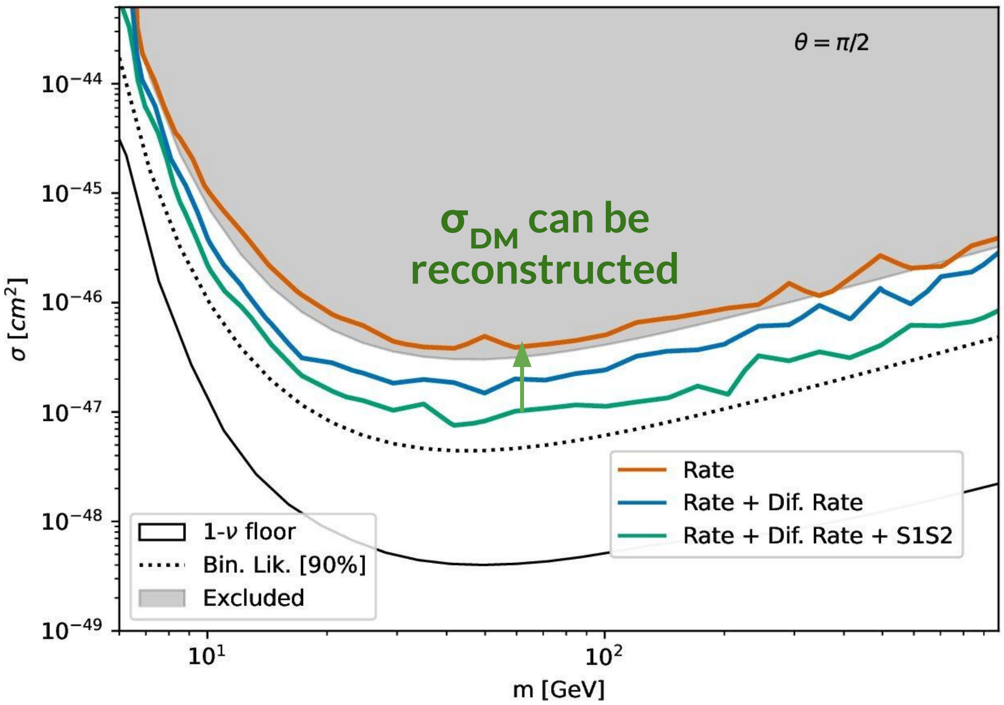
Reconstruction of the cross-section σ_{DM}

ML (SWYFT):

- **rate**
- **rate + diff. rate**

Results: parameter reconstruction

For O_1 with fixed θ=π/2



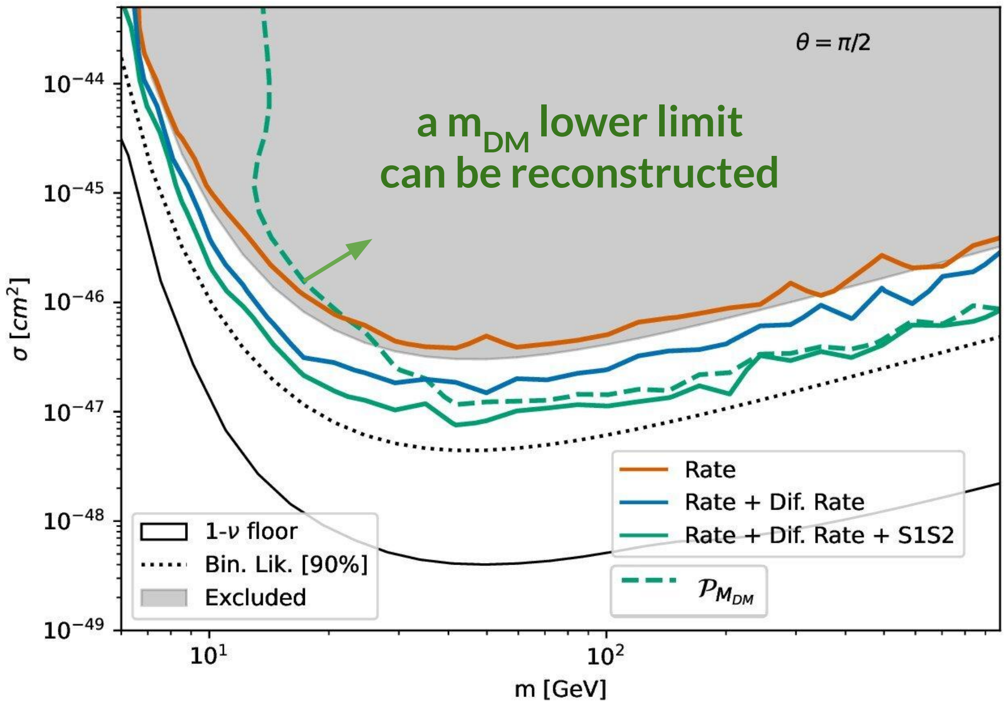
Reconstruction of the cross-section σ_{DM}

ML (SWYFT):

- rate
- rate + diff. rate
- rate + diff. rate + cS1-cS2

Results: parameter reconstruction

For O_1 with fixed $\theta=\pi/2$



Reconstruction of the cross-section σ_{DM}

ML (SWYFT):

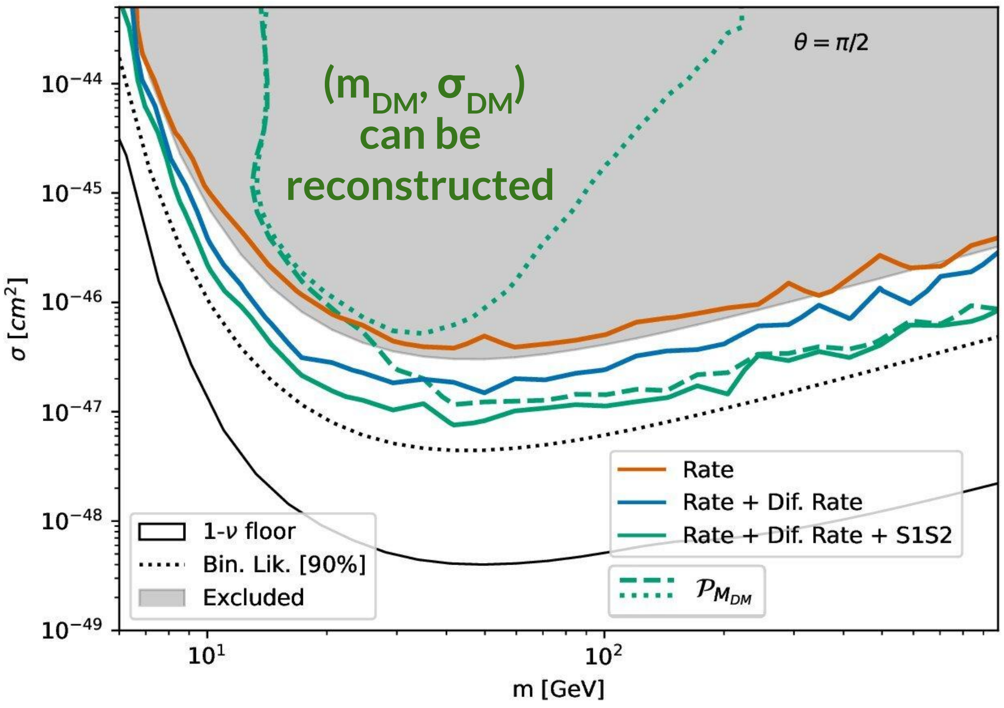
- rate
- rate + diff. rate
- rate + diff. rate + cS1-cS2

Reconstruction of the mass m_{DM}

- rate + diff. rate + cS1-cS2

Results: parameter reconstruction

For O_1 with fixed $\theta = \pi/2$



Reconstruction of the cross-section σ_{DM}

ML (SWYFT):

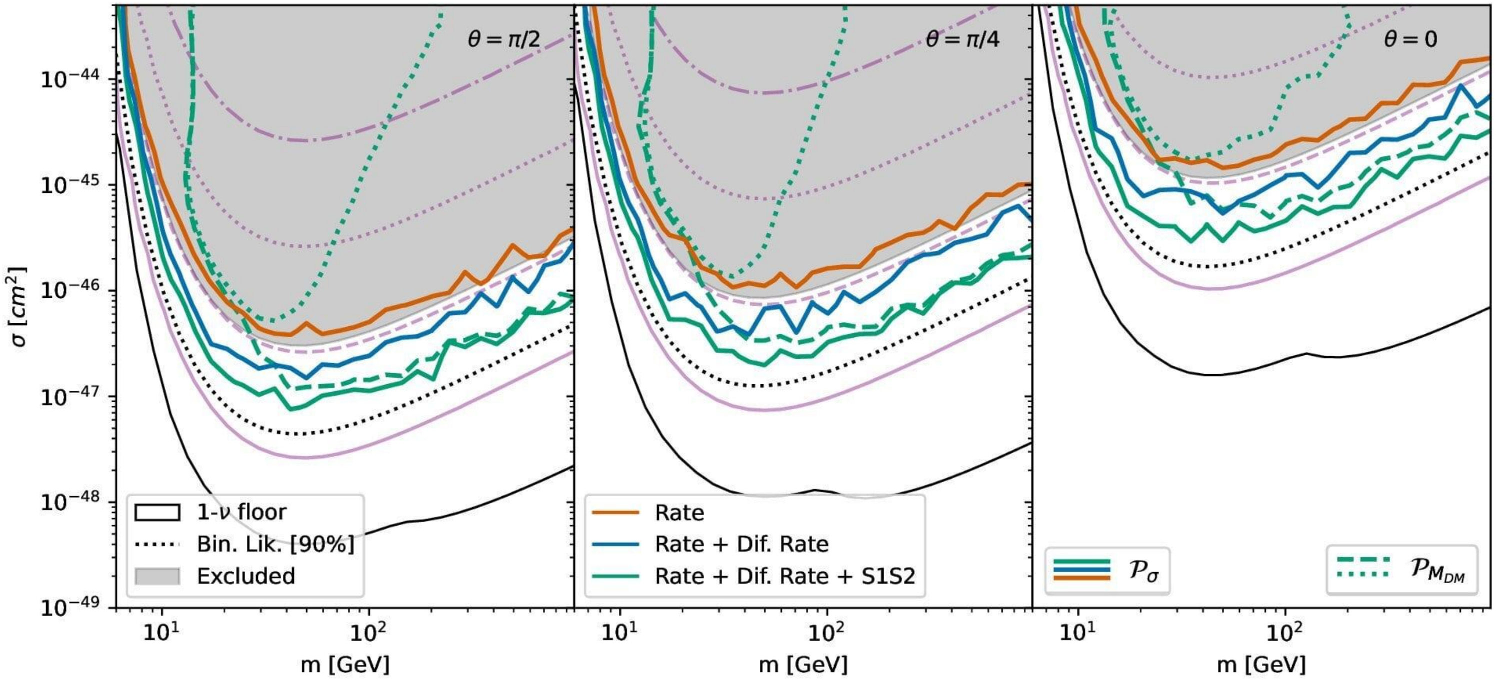
- rate
- rate + diff. rate
- rate + diff. rate + cS1-cS2

Reconstruction of the mass m_{DM}

- rate + diff. rate + cS1-cS2

Results: parameter reconstruction

O_1 operator
XENONnT 20ty

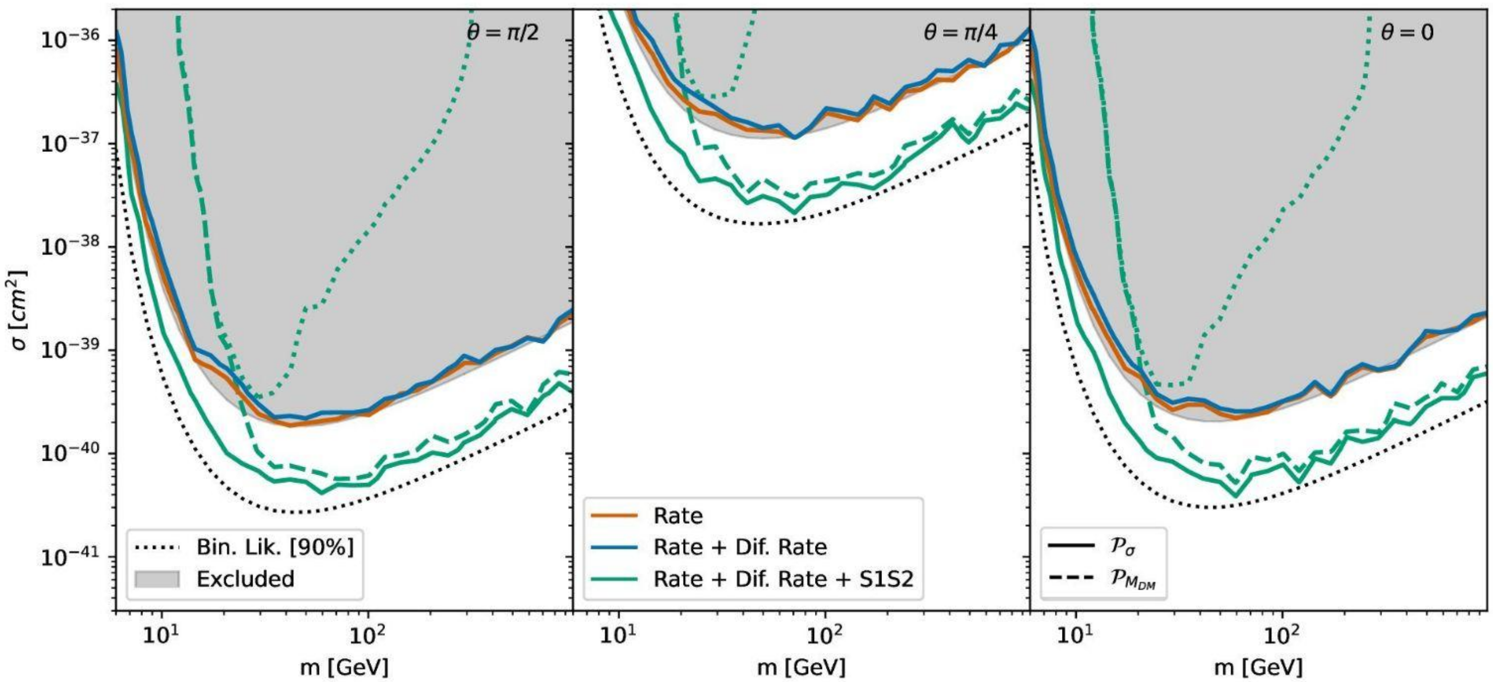


This panel is the usually shown **SI** parameter space

Data: **total number of events**
+ differential rate
+ cS1 vs cS2 plane

Results: parameter reconstruction

O_4 operator
XENONnT 20ty



Data: **total number of events**
+ differential rate
+ cS1 vs cS2 plane

Python package

CADDENA ⟶ https://github.com/Martindelosrios/CADDENA

In [6]:
```
# Now let's load the dataset with XENON nT simulations.
#  This data was create with .....

ref = files("CADDENA") / "dataset/"
DATA_PATH = str(ref)
with h5py.File(DATA_PATH + "/testset.h5", "r") as data:
  x_norm_rate  = data['x_norm_rate'][()]
  x_norm_drate = data['x_norm_drate'][()]
  x_norm_s1s2  = data['x_norm_s1s2'][()]
  pars_norm    = data['pars_norm'][()]
```

Load the dataset

In [7]:
```
# Let's pick some random synthetic observation from the simulated dataset.

i = np.random.randint(len(pars_norm))
print('nobs = ' + str(i))

pars_true   = pars_norm[i,:]
x_obs_rate  = x_norm_rate[i,:]
x_obs_drate = x_norm_drate[i,:]
x_obs_s1s2  = x_norm_s1s2[i,:].reshape(1,96,96)
```

Pick one as the ‘observation'

Python package

CADDENA → https://github.com/Martindelosrios/CADDENA

In [11]:
```
models.XENONnT_O1_rate.load_weights()
models.XENONnT_O1_drate.load_weights()
models.XENONnT_O1_s1s2.load_weights()
```

Load the already trained ML algorithms

In [13]:
```
# Let's sample the parameters from a flat prior

pars_prior = np.random.uniform(low = 0, high = 1, size = (100_000, 3))
```

Set a prior

Python package

CADDENA → https://github.com/Martindelosrios/CADDENA

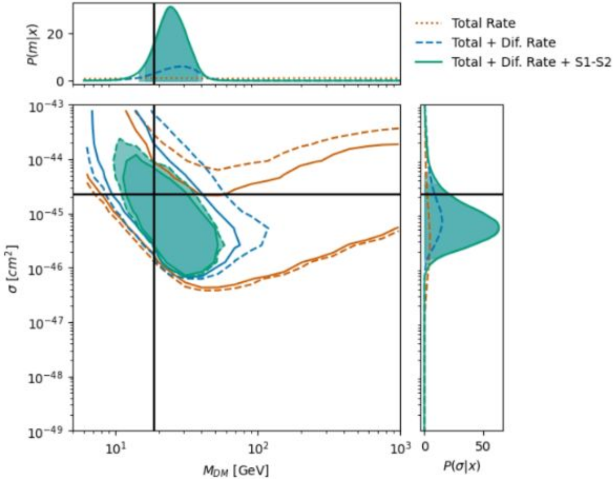
In [11]:

```
# Let's analyze the each type of data with the corresponding model

# The first argument is a list with the observation that will be analyzed.
# The second argument is the sample of paramters that will be paired with the observations.
# The third argument is a list with the models that will be used.

logratios1D, logratios2D = caddena.ratio_estimation([x_obs_rate, x_obs_drate, x_obs_s1s2],
                                                    pars_prior,
                                                    [models.XENONnT_01_rate, models.XENONnT_01_drate,
                                                     models.XENONnT_01_s1s2])
```

Compute the posteriors



Plot the posterior distributions

Conclusions

- We developed a Bayesian analysis to explore the reach of direct detection experiments that can be applied to any DM model (NR-EFT)

- The ML implementation (SWYFT) is fundamental:
  - **fast** estimation of posteriors,
  - simple way to **include-combine-remove** data.

- O_1 (SI) and O_4 (SD) presented here as examples

- We computed the parameter space where $\mathbf{m_{DM}}$ and $\boldsymbol{\sigma}$ that can be **reconstructed**,
  using: total number of events - differential rate - full cS1,cS2 space.

- **Next:**
  Apply to other NR-EFT operators → combine operators
  Different DD experiments → combine experiments

Thank you!

Back-up

Bayesian Analysis

Marginalization (traditional method):

one needs to assume a functional form for the likelihood

$$P(\mathbf{\Theta}_0|\mathbf{X}) = \int \frac{P(\mathbf{X}|\mathbf{\Theta})P(\mathbf{\Theta})\prod_j d\mathbf{\Theta}_{j\neq 0}}{P(\mathbf{X})}$$

If you want to study only one parameter Θ_0

The full posterior has to be computed

Bayesian Analysis with SWYFT

SWYFT $\longrightarrow$ Sampling-based inference tool that estimates *likelihood-to-evidence ratio* with ML algorithms to obtain marginal and joint posteriors

$$r(\mathbf{X}, \mathbf{\Theta}) = \frac{\textbf{output}}{1 - \textbf{output}}$$

compute the posterior $\rightarrow$

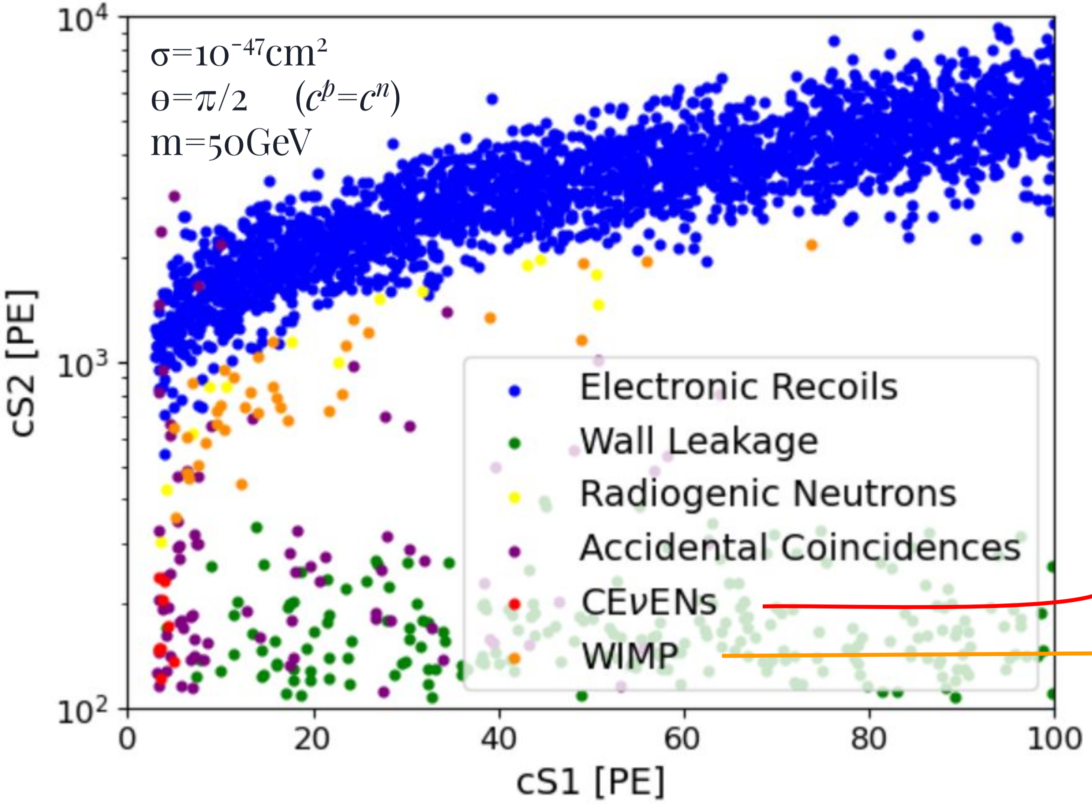
$$P(\mathbf{\Theta}|\mathbf{X}) = r(\mathbf{X}, \mathbf{\Theta})\, P(\mathbf{\Theta})$$

$$\begin{aligned} r(\mathbf{X}, \mathbf{\Theta}) &= \frac{P(\mathbf{X}, \mathbf{\Theta})}{P(\mathbf{\Theta})P(\mathbf{X})} = \frac{P(\mathbf{X}, \mathbf{\Theta}|k=1)}{P(\mathbf{X}, \mathbf{\Theta}|k=0)} \\ &= \frac{P(\mathbf{X}, \mathbf{\Theta}, k=1)}{P(\mathbf{X}, \mathbf{\Theta}, k=0)} = \frac{P(k=1|\mathbf{X}, \mathbf{\Theta})}{P(k=0|\mathbf{X}, \mathbf{\Theta})} \\ &= \frac{P(k=1|\mathbf{X}, \mathbf{\Theta})}{1 - P(k=1|\mathbf{X}, \mathbf{\Theta})} = \frac{\textbf{output}}{1 - \textbf{output}} \end{aligned}$$

DM signal

XENONnT 20ty

NR-EFT: O1



XENONnT simulator

We specify background and signal characteristics

differential rate compute with SNuDD

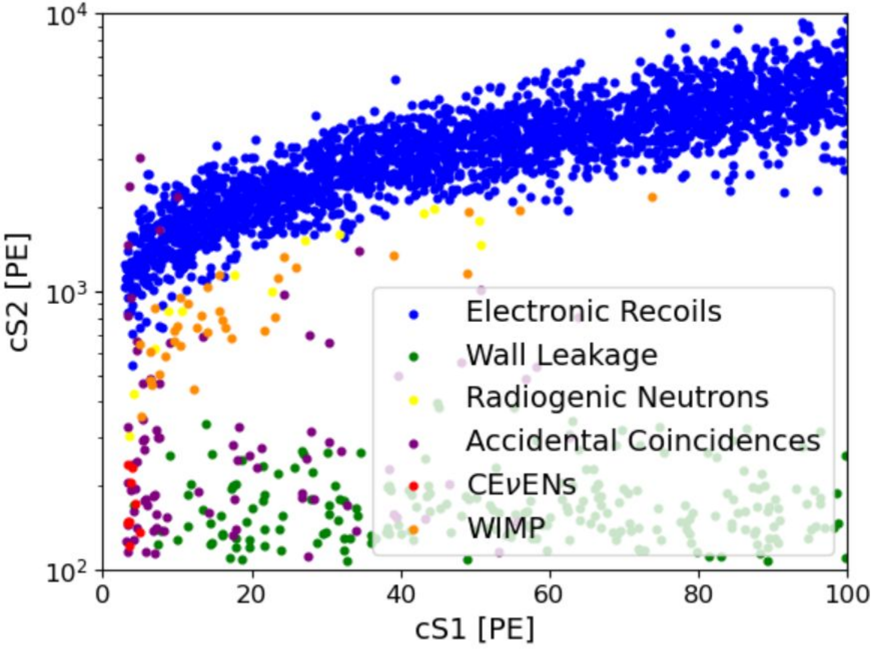
differential rate compute with WimPyDD for a particular operator, $(\sigma_i, \theta_i, m_{DM})$, and standard DM halo model

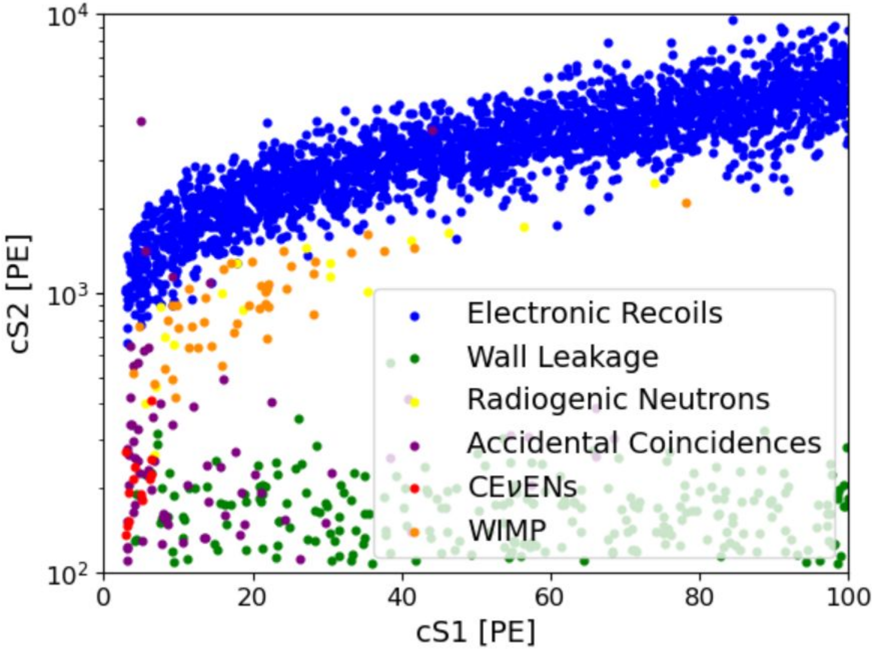
Data Representation: cS1 vs cS2 plane

XENONnT 20ty

NR-EFT: O1
$\sigma=10^{-47}cm^2$
$\theta=\pi/2 \quad (c^p=c^n)$
m=50GeV

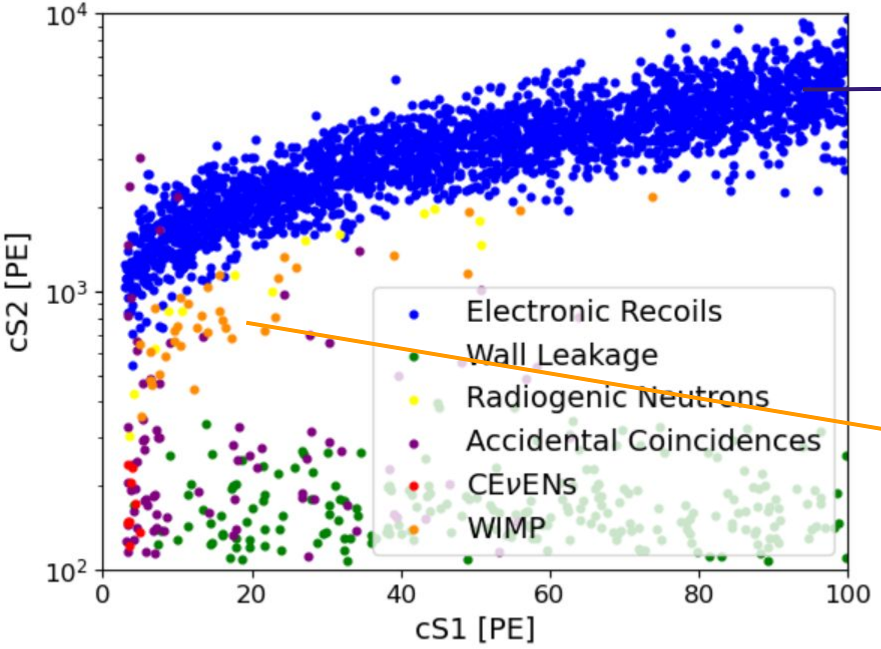
We generate a 10k pseudo experiments per operator varying σ, θ, and m_{DM}





Data Representation: number of events

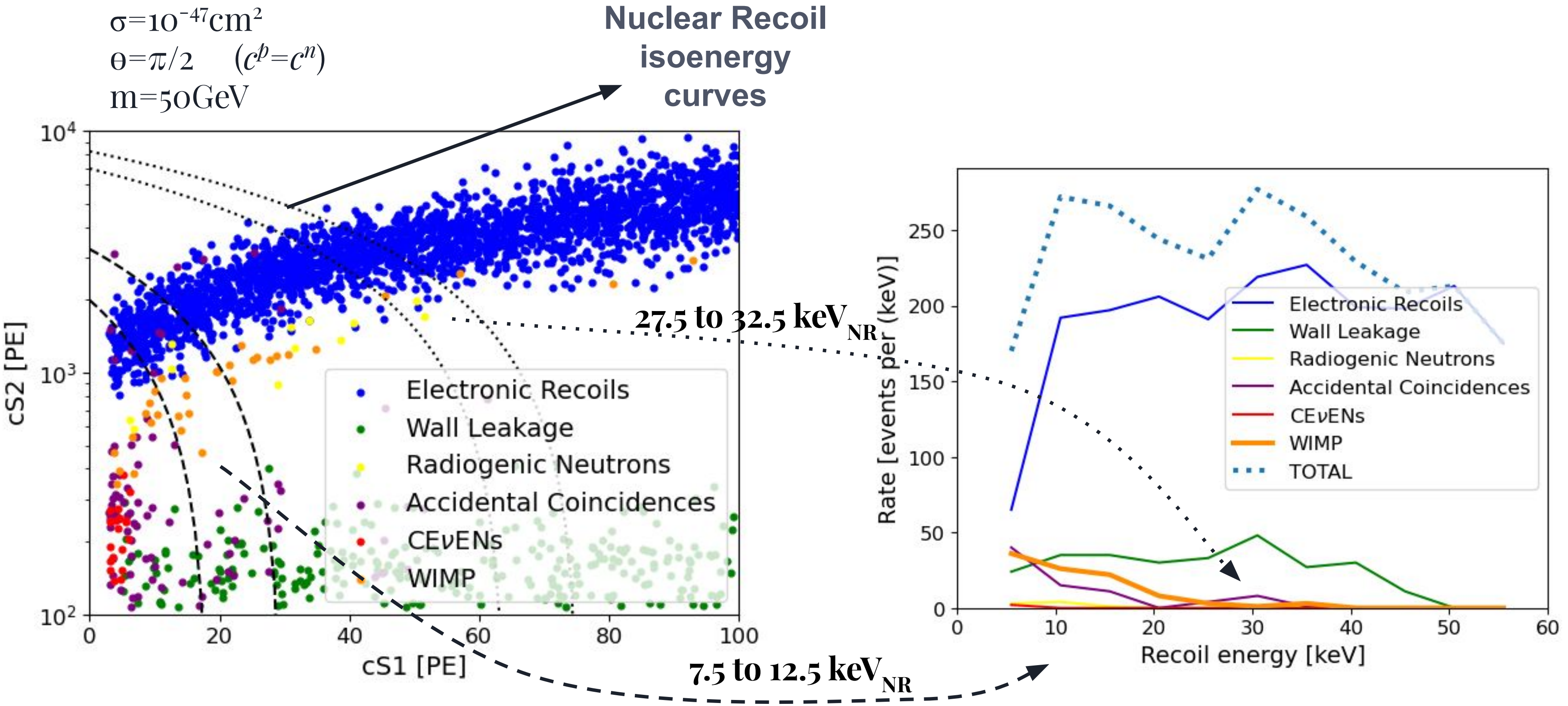
NR-EFT: O1
σ=10^{-47}cm^2
θ=π/2 ($c^p=c^n$)
m=50GeV



	name	pseudo_exp_events
0	er	2459
1	radiogenics	17
2	ac	71
3	wall	246
4	WIMP	43
5	CEVNS-SM	13

Data Representation: differential rate

NR-EFT: O1
$\sigma=10^{-47}\text{cm}^2$
$\theta=\pi/2 \quad (c^p=c^n)$
m=50GeV



Results

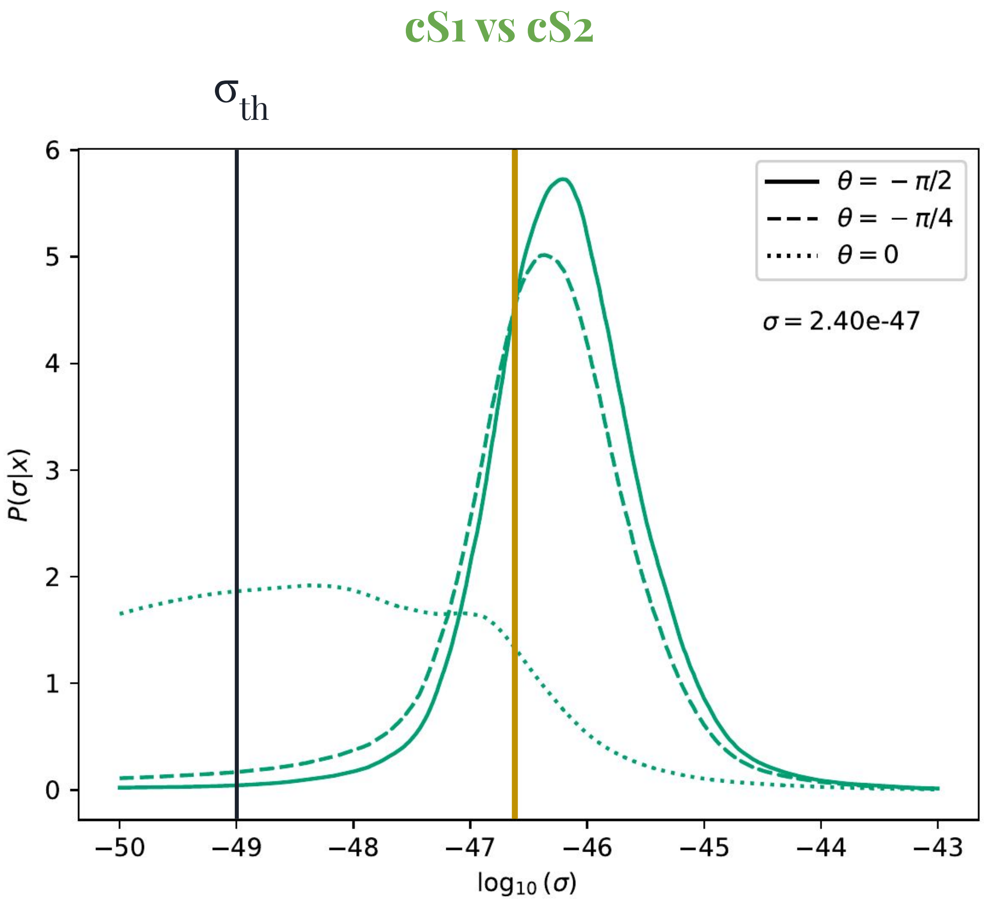
Data:

cS1 vs cS2 plane

Posteriors for:

O_1 (SI)
m_{DM}≃100GeV → **fixed**
σ=2.4 $10^{-47}cm^2$ → **fixed**

reconstruction
depends on θ



Comparison with Multinest

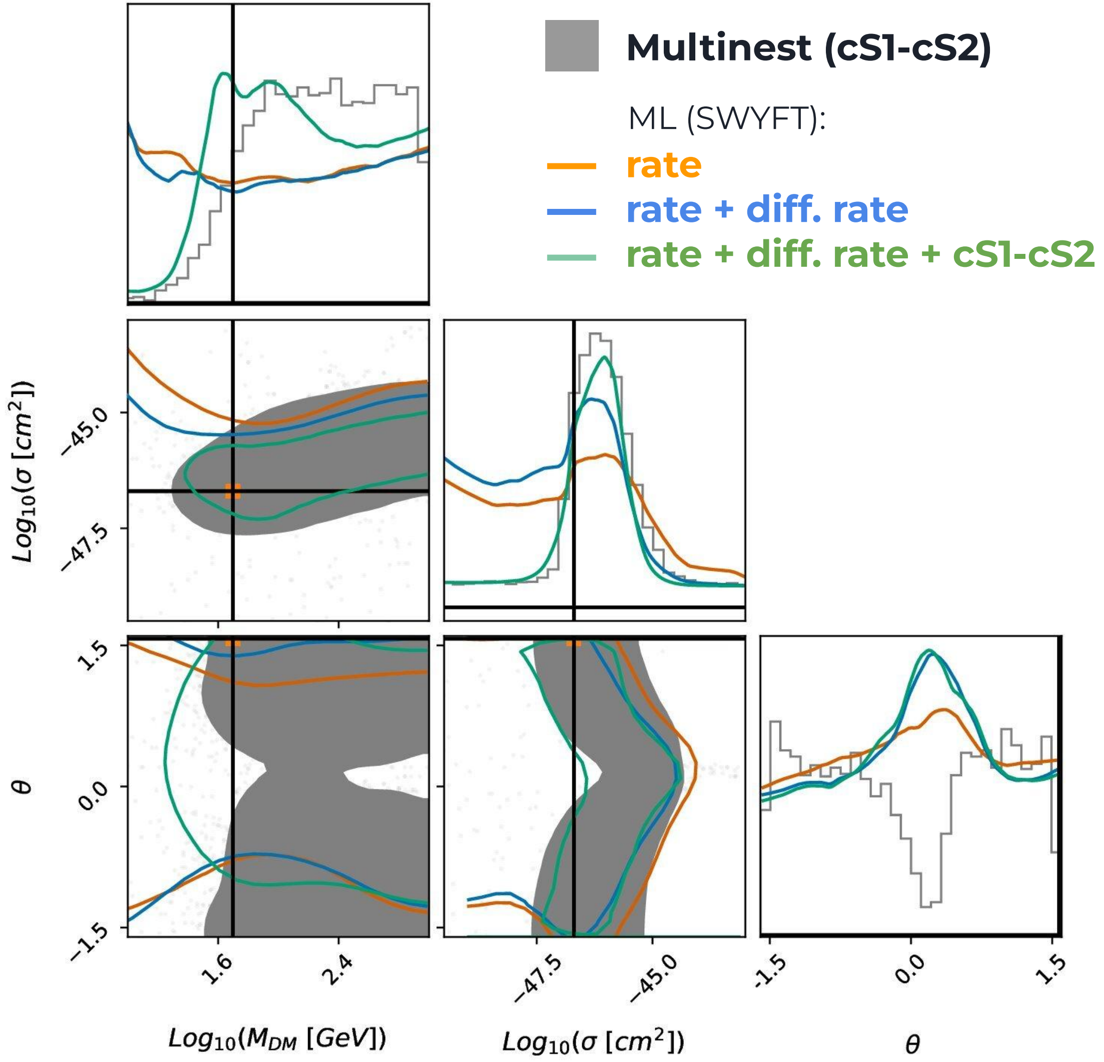
Posterior distributions for:

O_1 (SI)

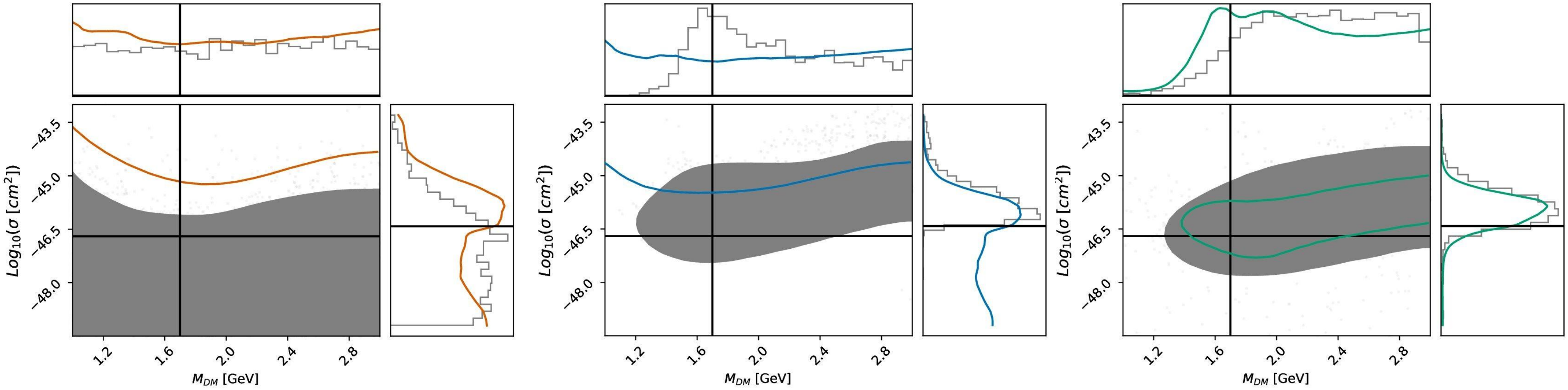
$m_{DM} \approx 50 GeV$ → **fixed**

$\sigma = 2.10^{-47} cm^2$ → **fixed**

$\theta = \pi/2$ → **fixed**



Results: parameter reconstruction



Posterior distributions for:

O_1 (SI)

m_{DM}≃50GeV → **fixed**

σ=2. $10^{-47}$$cm^2$ → **fixed**

θ=π/2 → **fixed**

Multinest (cS1-cS2)

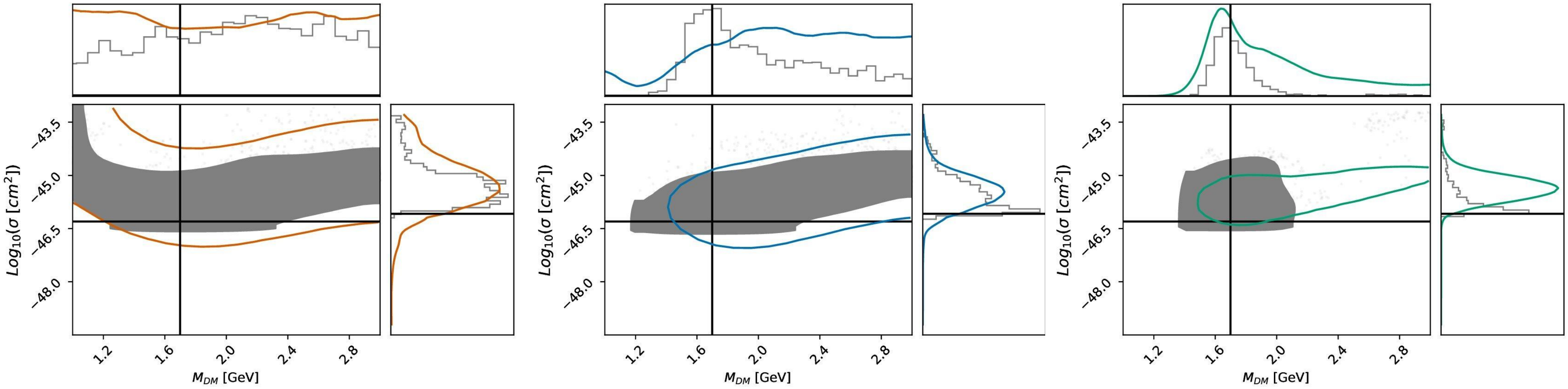
ML (SWYFT):

rate

rate + diff. rate

rate + diff. rate + cS1-cS2

Results: parameter reconstruction



Posterior distributions for:

O_1 (SI)
m_{DM}≃50GeV → **fixed**
$\sigma = 5.\ 10^{-47} cm^2$ → **fixed**
$\theta = \pi/2$ → **fixed**

Multinest (cS1-cS2)

ML (SWYFT):
rate
rate + diff. rate
rate + diff. rate + cS1-cS2

DM-nucleon non-relativistic effective field theory (NR-EFT)

For DM particles with spin up to ½ , the effective DM-nucleon scattering interaction Lagrangian

$$\mathcal{L}_{\text{EFT}} = \sum_{\tau} \sum_{i} c_i^{\tau} \mathcal{O}_i \overline{\chi} \chi \overline{\tau} \tau$$

i=14 possible interactions

$$i\frac{\vec{q}}{m_N}, \quad \vec{v}^{\perp}, \quad \vec{S}_{\chi}, \quad \vec{S}_N.$$

$$\vec{v} \equiv \vec{v}_{\chi,\text{in}} - \vec{v}_{N,\text{in}}$$

$$\vec{v}^{\perp} = \vec{v} + \frac{\vec{q}}{2\mu_N}$$

momentum transfer, spin operators, relative velocity

$\mathcal{O}_1 = 1_{\chi} 1_N$	$\mathcal{O}_9 = i\vec{S}_{\chi} \cdot (\vec{S}_N \times \frac{\vec{q}}{m_N})$
$\mathcal{O}_3 = i\vec{S}_N \cdot (\frac{\vec{q}}{m_N} \times \vec{v}^{\perp})$	$\mathcal{O}_{10} = i\vec{S}_N \cdot \frac{\vec{q}}{m_N}$
$\mathcal{O}_4 = \vec{S}_{\chi} \cdot \vec{S}_N$	$\mathcal{O}_{11} = i\vec{S}_{\chi} \cdot \frac{\vec{q}}{m_N}$
$\mathcal{O}_5 = i\vec{S}_{\chi} \cdot (\frac{\vec{q}}{m_N} \times \vec{v}^{\perp})$	$\mathcal{O}_{12} = \vec{S}_{\chi} \cdot (\vec{S}_N \times \vec{v}^{\perp})$
$\mathcal{O}_6 = (\vec{S}_{\chi} \cdot \frac{\vec{q}}{m_N})(\vec{S}_N \cdot \frac{\vec{q}}{m_N})$	$\mathcal{O}_{13} = i(\vec{S}_{\chi} \cdot \vec{v}^{\perp})(\vec{S}_N \cdot \frac{\vec{q}}{m_N})$
$\mathcal{O}_7 = \vec{S}_N \cdot \vec{v}^{\perp}$	$\mathcal{O}_{14} = i(\vec{S}_{\chi} \cdot \frac{\vec{q}}{m_N})(\vec{S}_N \cdot \vec{v}^{\perp})$
$\mathcal{O}_8 = \vec{S}_{\chi} \cdot \vec{v}^{\perp}$	$\mathcal{O}_{15} = -(\vec{S}_{\chi} \cdot \frac{\vec{q}}{m_N})((\vec{S}_N \times \vec{v}^{\perp}) \cdot \frac{\vec{q}}{m_N})$

DM-nucleon non-relativistic effective field theory (NR-EFT)

For DM particles with spin up to ½ , the effective DM-nucleon scattering interaction Lagrangian

$$\mathcal{L}_{\rm EFT} = \sum_\tau \sum_i c_i^\tau \mathcal{O}_i \bar{\chi}\chi\bar{\tau}\tau$$

i=14 possible interactions

$$c_i^0 \mathbb{1}_{2\times 2} + c_i^1 \tau_3$$

isospin basis
c^0: isoscalar
c^1: isovector

$$c_i^0 = \frac{1}{2}(c_i^p + c_i^n)$$

$$c_i^1 = \frac{1}{2}(c_i^p - c_i^n)$$

nucleon basis
c^p: proton
c^n: neutron

O1: spin-independent (SI)
O4: spin-dependent (SD)

usually shown assuming isoscalar interactions

$c^p = c^n$ $c^0 = 1$ and $c^1 = 0$

DM-nucleon non-relativistic effective field theory (NR-EFT)

For DM particles with spin up to ½ , the effective DM-nucleon scattering interaction Lagrangian

$$\mathcal{L}_{\rm EFT} = \sum_{\tau} \sum_{i} c_i^{\tau} \mathcal{O}_i \overline{\chi} \chi \overline{\tau} \tau$$

i=14 possible interactions

Change to polar coordinates:

$$c_i^0 = \frac{1}{2}(c_i^p + c_i^n) \;= A_i \sin(\theta_i)$$

$$c_i^1 = \frac{1}{2}(c_i^p - c_i^n) \;= A_i \cos(\theta_i)$$

Natural choice for the EFT parameter space because the interaction cross section:

$$\sigma_i \propto A_i^2$$

For SI (O1)

$$\sigma_{\chi N}^{\rm SI} = \frac{A_1^2 \;\mu_{\chi N}^2}{\pi}$$

DM-nucleon reduced mass

DM-nucleon non-relativistic effective field theory (NR-EFT)

Contact interaction between a spin ½ DM and nucleon

$$\mathcal{L}_{\rm int}^{\rm SI}(\vec{x}) = c_1 \; \bar{\Psi}_\chi(\vec{x})\Psi_\chi(\vec{x}) \; \bar{\Psi}_N(\vec{x})\Psi_N(\vec{x})$$

$$U_\chi(p) = \sqrt{\frac{E+m}{2m}} \begin{pmatrix} \xi_\chi \\ \frac{\vec{\sigma}\cdot\vec{p}}{E+m_\chi}\xi_\chi \end{pmatrix} \sim \begin{pmatrix} \xi_\chi \\ \frac{\vec{\sigma}\cdot\vec{p}}{2m_\chi}\xi_\chi \end{pmatrix}$$

at low momenta.

Idem for the nucleon spinor

ξ Pauli spinors

at leading order in p/m $\quad c_1 \; 1_\chi 1_N \; \equiv \; c_1 \; \mathcal{O}_1$

DM-nucleon non-relativistic effective field theory (NR-EFT)

Another interaction

$$\mathcal{L}_{\rm int}^{\rm SD} = c_4 \; \bar{\chi}\gamma^\mu\gamma^5\chi\bar{N}\gamma_\mu\gamma^5 N$$

the dominant contribution in the non-relativistic limit comes from the spatial indices

$$\bar{\chi}\gamma^i\gamma^5\chi \sim \xi_\chi^\dagger\sigma^i\xi_\chi$$

Since $\hat{S}^i = \sigma^i/2$

$$-4c_4 \; \vec{S}_\chi \cdot \vec{S}_N \;\equiv\; -4c_4 \; \mathcal{O}_4$$

Data analysis to obtain posteriors

SWYFT → Sampling-based inference tool that estimates likelihood to evidence ratio with ML algorithms to obtain marginal and joint posteriors

	MCMC	**SWYFT**
Forward Model	x=f(parameters)	x=f(parameters)
Likelihood	L(x, f(parameters))	Data Driven
Samples	All parameters space > # samples	Only Interesting parameters
Amortization	NO	YES

Motivation

Bayes' Rule: determine a probability distribution over model parameters **θ** given an observation **x**

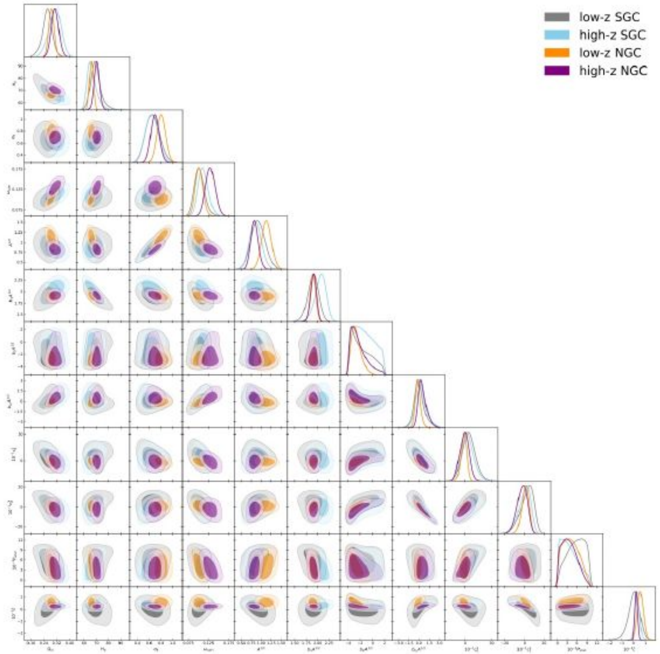
Likelihood of **x** given **θ**

$$p(\boldsymbol{\theta} \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid \boldsymbol{\theta})}{p(\boldsymbol{x})} p(\boldsymbol{\theta})$$

Posterior

Evidence of the data

Prior



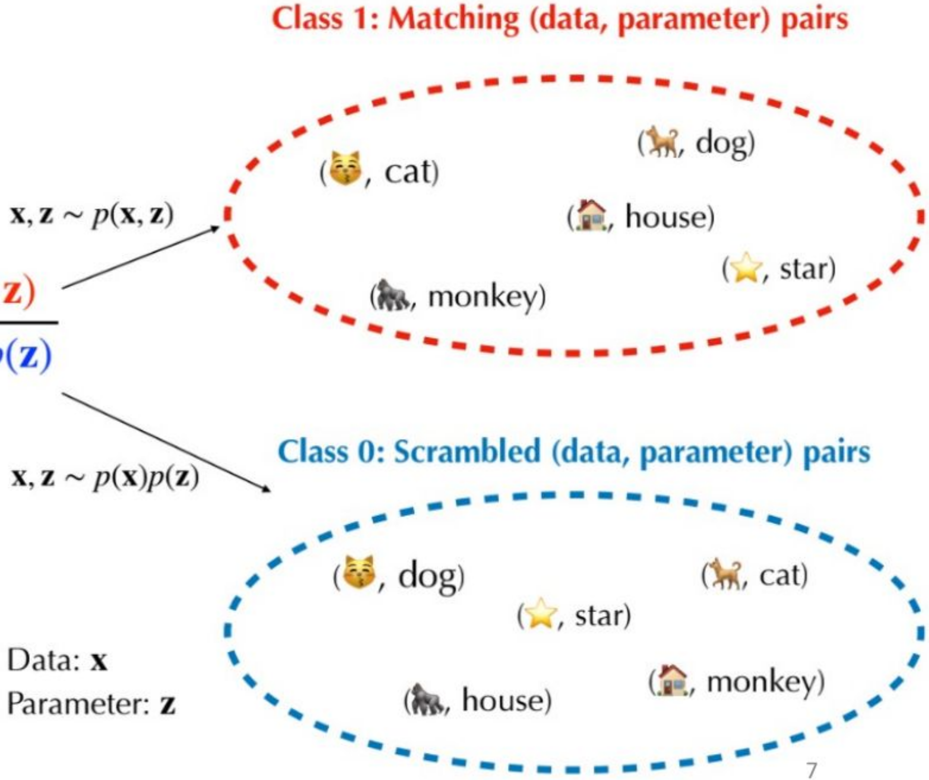
Samples tipically generated with *Markov Chain Monte Carlo* (MCMC) or *Nested sampling*

BOSS, Ivanov+ 1909.05277

Neural Ratio Estimation (NRE)

Approximate density ratios.

$$r(\mathbf{x};\mathbf{z}) \equiv \frac{p(\mathbf{x}\,|\,\mathbf{z})}{p(\mathbf{x})} = \frac{p(\mathbf{z}\,|\,\mathbf{x})}{p(\mathbf{z})} = \frac{p(\mathbf{x},\mathbf{z})}{p(\mathbf{x})p(\mathbf{z})}$$



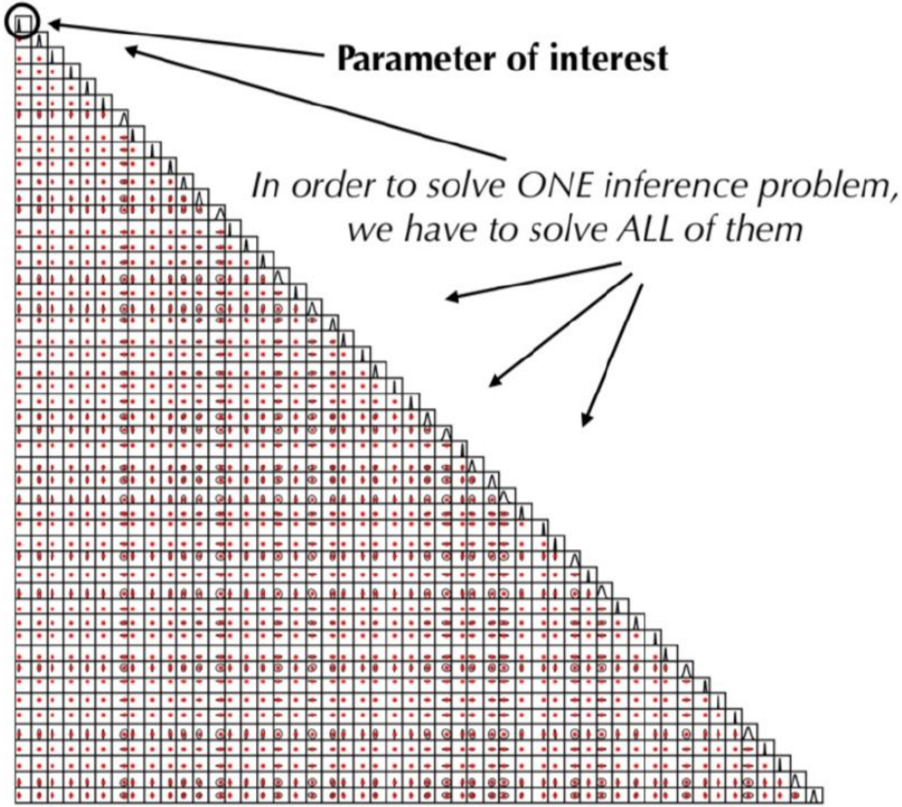
Strategy: *We estimate posteriors-to-prior ratio by training a binary classifier to discriminate between matching and scrambled (data, parameter) pairs.*

MARGINAL

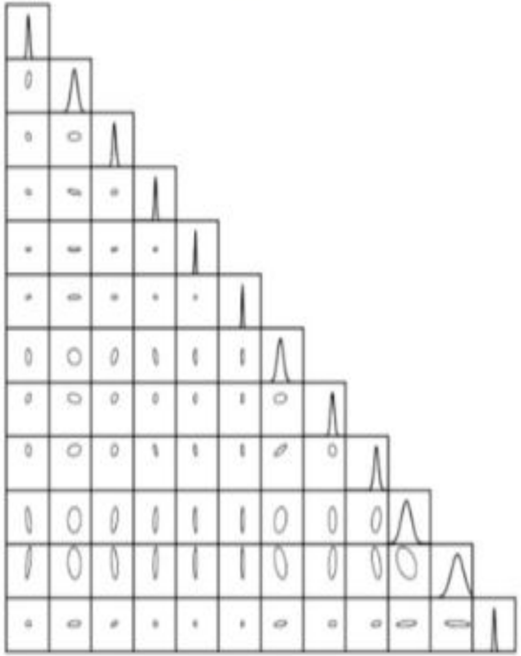
- MCMC or Nested sampling methods produce samples from the **posterior distribution**.

- Classical methods require sampling the **full joint posterior**, so that they are slow to converge.

- Novel approaches in the field of *simulation-based inference* (SBI) are starting to overcome these obstacles.

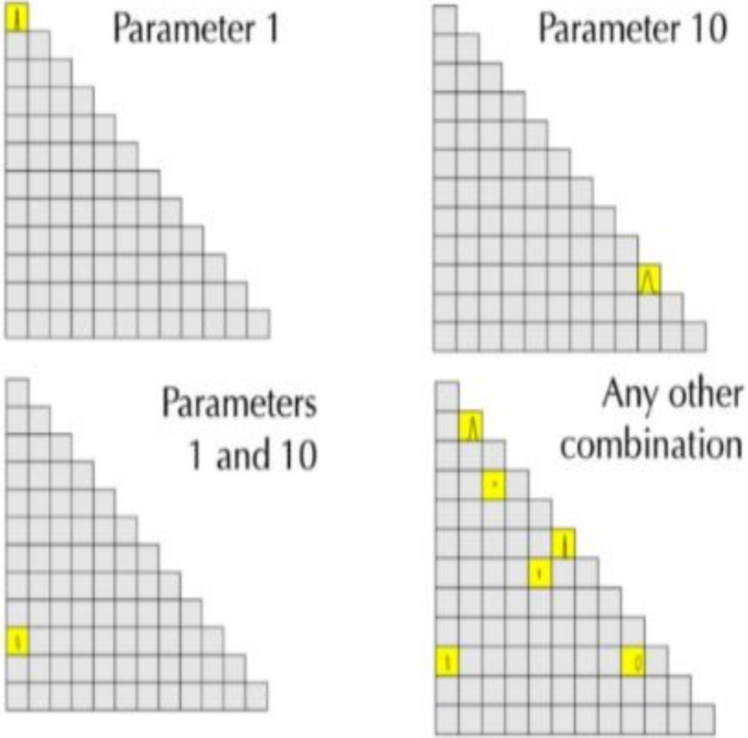


Instead of estimating all parameters…



50 parameters ~ 100 Million simulations

…we can choose what we care about



Depending on which parameter is scrambled

Results

Data:

entire cS1 vs cS2 plane

These are all the posteriors for

$m_{DM} \simeq 100GeV \longrightarrow$ fixed

$\theta = \pi/2 \longrightarrow$ fixed

red $\longrightarrow$ σ not reconstructed

~no signal,

~similar posteriors

black $\longrightarrow$ σ reconstructed