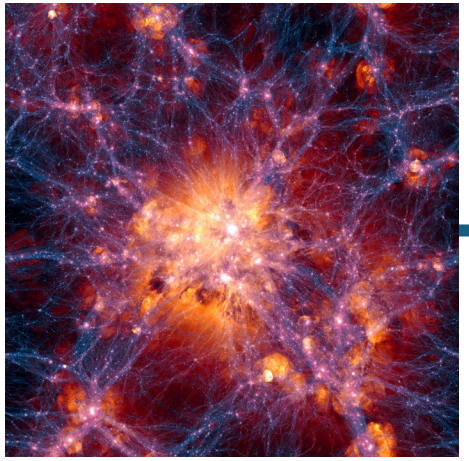
Cosmological Information from Clustering Statistics with Mul-guisin

Sumi Kim
Cristiano Giovanni Sabiu
Inkyu Park
Young Ju

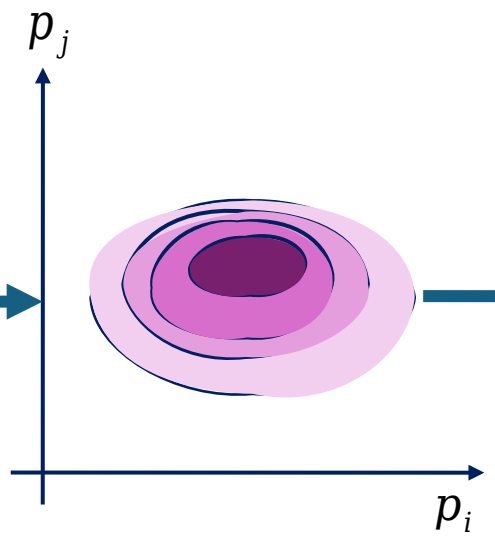
About Cosmological Parameter



https://www.eso.org/public/images/eso1438b/

Summary Statistics

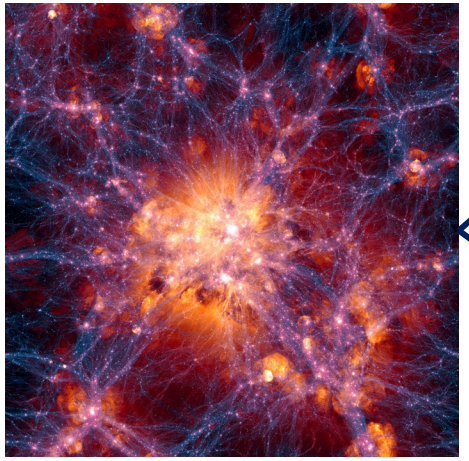
ex. Power spectrum, N – point correla-tion fucntion



Cosmological Information

$\boldsymbol{\Omega_m, \sigma_8, w_0, n_s, h}$

About Cosmological Parameter



https://www.eso.org/public/images/eso1438b/

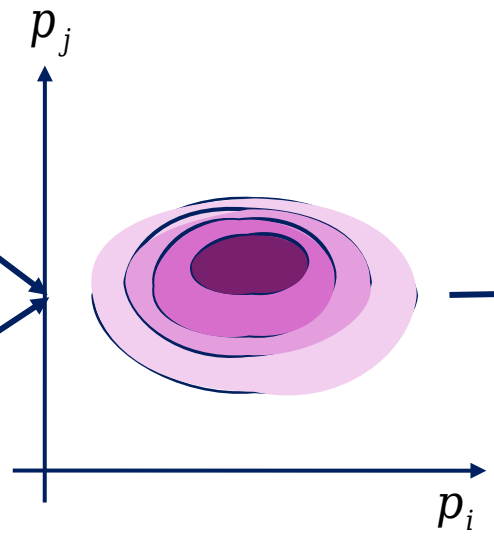
Npcf

2pcf

Isotropic
Ansiotropic

3pcf

Isotropic
Ansiotropic

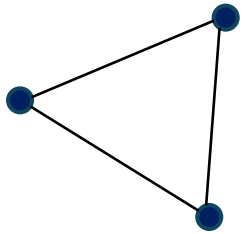


Cosmological Information

$\boldsymbol{\Omega_m, \sigma_8, w_0, n_s, h}$

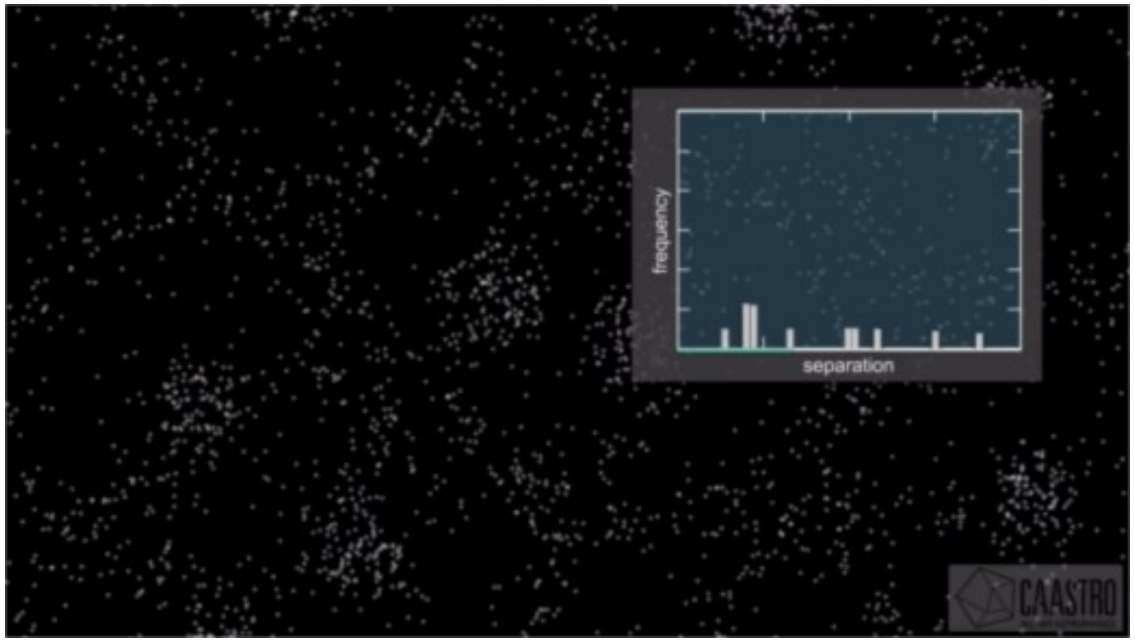
N – point Correlation Function

3pcf



2pcf



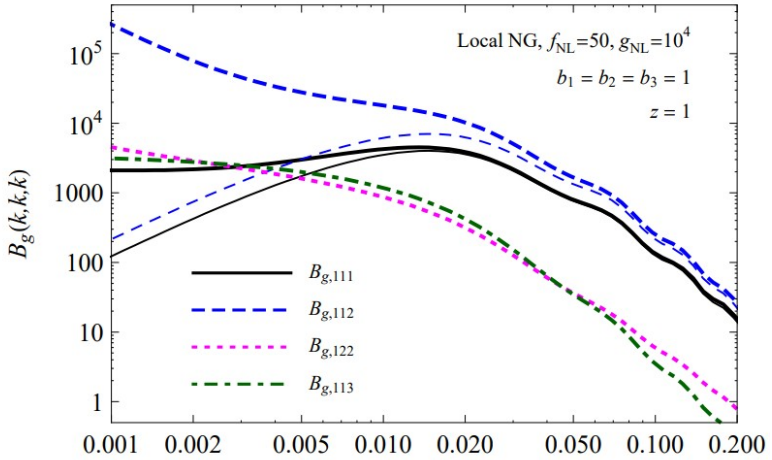


https://www.youtube.com/watch?v=jpXuYc-wzk4&t=1s

Reason why : universe is not Gaussian random

✓ Nonlinear structure formation

Detect the non – gaussianity of distribution can’t be captured by 2pcf

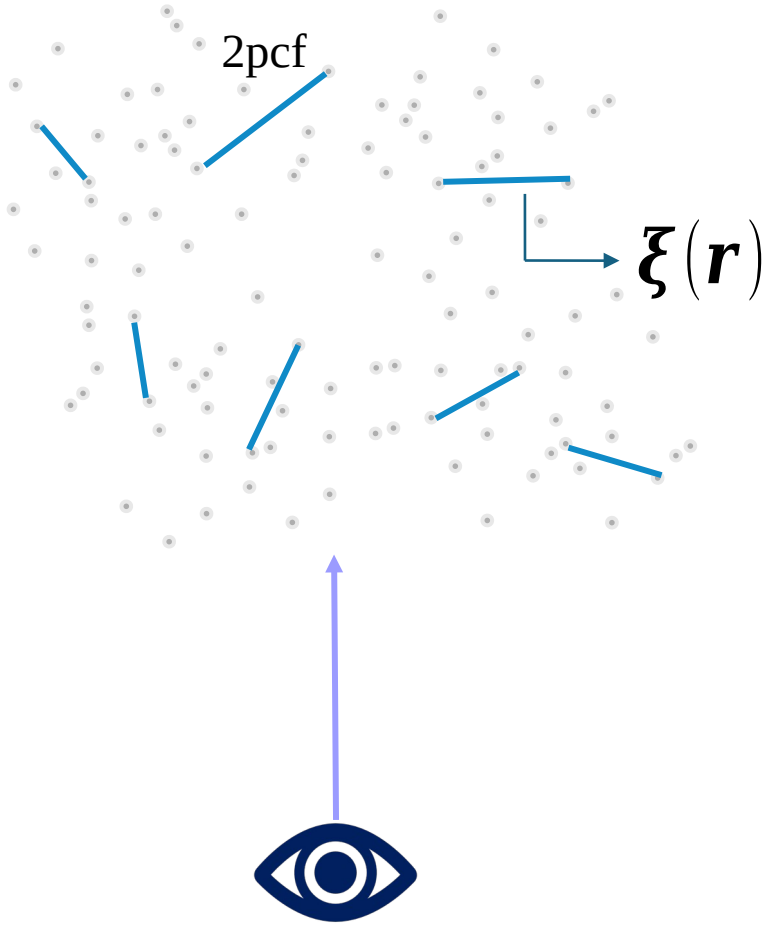


(Liguori et al.,

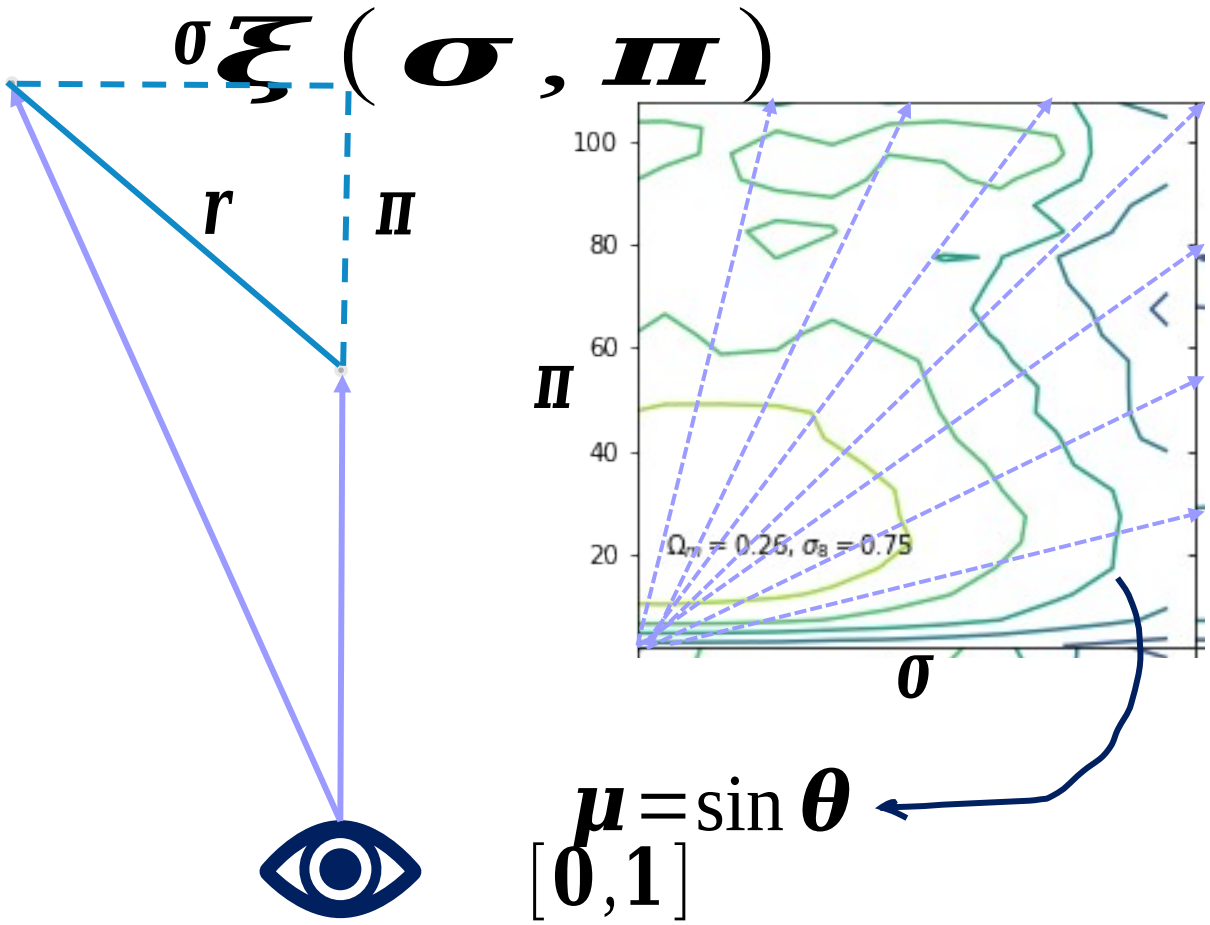
Isotropic / Anisotropic

ex. 2pcf

Isotropic



Anisotropic



About This Work

1	2	3	4
Npcf Result	Regressor	Estimation	Contour
Calculate 2pcf 3pcf MGS	Put the result into machine learning regres-sor	Cosmological parameter esti-mation	Compare the contours of each parameters and Npcfs

& covariance matrix

Data : MG - PICOLA

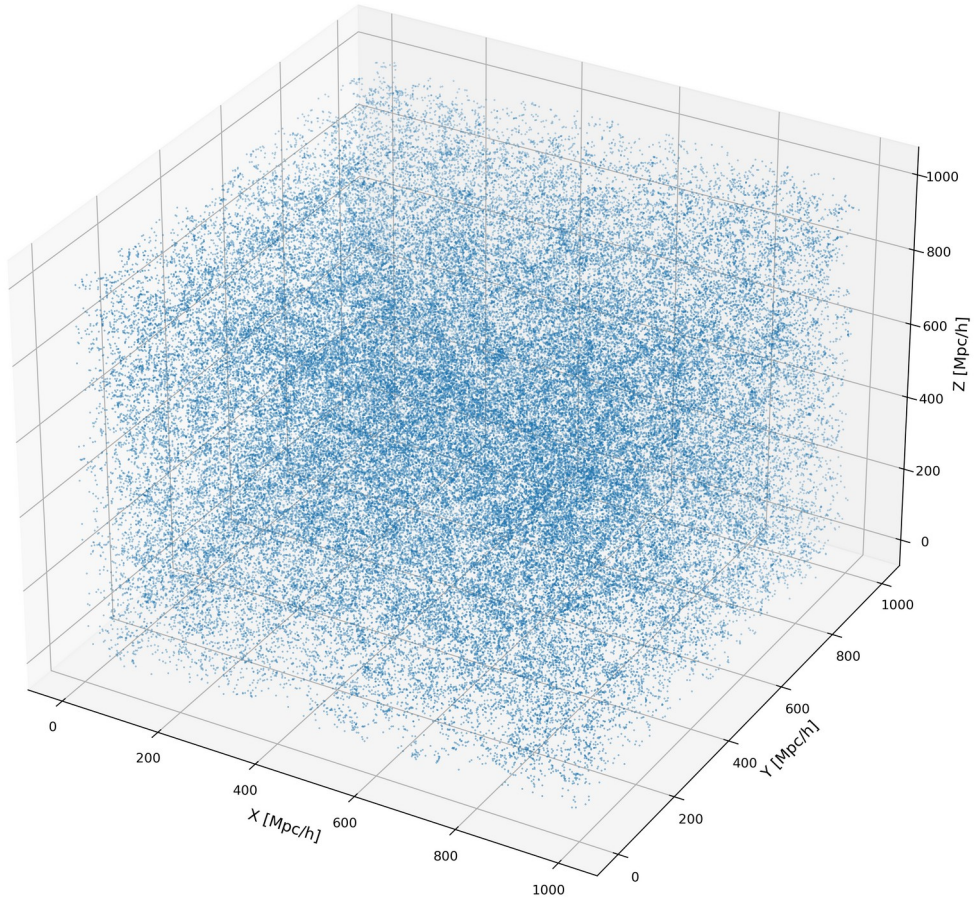
Data

MG – PICOLA – PUBLIC by HAWinther

1

Npcf Result

Calculate
2pcf
3pcf
MGS



Particle – Mesh N – body simulation

- Box size : 1 Gpc/h
- Run down to z = 0.700
- FoF halo – finder

1000 different cosmology
100 single cosmology

Data : Mulguisin

MGS

Mulguisin

1

Npcf Result

Calculate
2pcf
3pcf
MGS

Constructing a Galaxy Cluster Catalog in Illustri

Lael Shin,[1] Jubee Sohn,[1,2] Young Ju,[3,4] Inky

[1] *Department of Physics & Astronomy, Seoul National*
[2] *SNU Astronomy Research Center, Seoul National U*
[3] *Department of Physics, University of Seou*
[4] *Natural Science Research Institute, University o*

ABSTRAC

We present a new simulated galaxy cluster catalog b
the Mulguisin (MGS) algorithm to identify galaxy overc
the previous FoF cluster identification in two aspects; 1
subhalos instead of unobservable dark matter particles, a
galaxy overdensities hosted by massive galaxies. Our a
similar to the observed cluster catalogs using spectros
303 halos with $M_{200} > 10^{14} M_\odot$, including $\sim 10\%$ m
more systems because we separate some independent
into a single FoF algorithm. These independent MGS
spatial distribution and the phase-space diagram. Bec
that identifies local galaxy overdensities, we evaluate t
galaxies better than using the FoF cluster catalog. The
identifications of merging clusters by selecting systems v
highlight that the MGS cluster catalog is a useful tool fo
and for comparing with the observed cluster samples.

MulGuisin, a Topological Network Finder and its Performance on Galaxy Clustering

Young Ju,[a,b] Inkyu Park,[a,b] Cristiano G. Sabiu[a,b] and Sungwook E. Hong[c,d]

[a] Department of Physics, University of Seoul,
163 Seoulsiripdae-ro, Dongdaemun-gu, Seoul 02504, Republic of Korea
[b] Natural Science Research Institute, University of Seoul,
163 Seoulsiripdae-ro, Dongdaemun-gu, Seoul 02504, Republic of Korea
[c] Korea Astronomy and Space Science Institute,
776 Daedeok-daero, Yuseong-gu, Daejeon 34055, Republic of Korea
[d] Astronomy Campus, University of Science and Technology,
776 Daedeok-daero, Yuseong-gu, Daejeon 34055, Republic of Korea

E-mail: icpark@uos.ac.kr

Abstract. We introduce a new clustering algorithm, MulGuisin (MGS), that can identify distinct galaxy over-densities using topological information from the galaxy distribution. This algorithm was first introduced in an LHC experiment as a Jet Finder software, which looks for particles that clump together in close proximity. The algorithm preferentially considers particles with high energies and merges them only when they are closer than a certain distance to create a jet. MGS shares some similarities with the minimum spanning tree (MST) since it provides both clustering and network-based topology information. Also, similar to the density-based spatial clustering of applications with noise (DBSCAN), MGS uses the ranking or the local density of each particle to construct clustering. In this paper, we compare the performances of clustering algorithms using controlled data and some realistic simulation data as well as the SDSS observation data, and we demonstrate that our new algorithm finds networks most efficiently and defines galaxy networks in a way that most closely resembles human vision.

ArXiv ePrint: 2301.03278

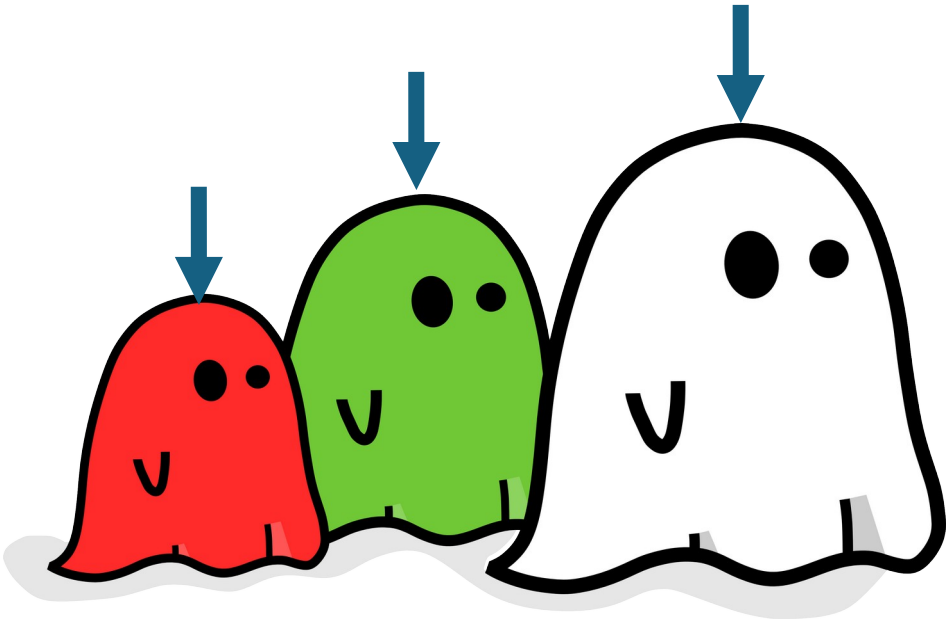
Data : Mulguisin

MGS

Mulguisin by Young Ju, Inkyu Park

1

Npcf Result
Calculate 2pcf 3pcf MGS

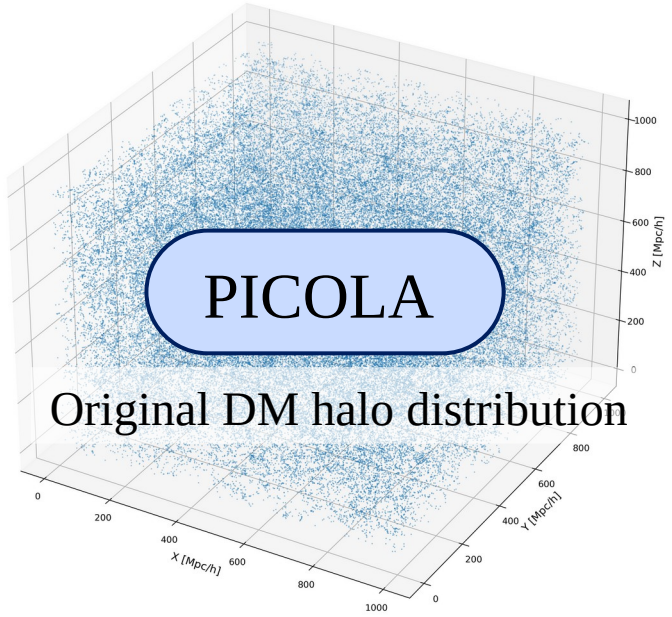


1. Find highest local number density, give network seed
2. Find second highest density, decide it becomes the network or separated one
3. Decision is made by linking length

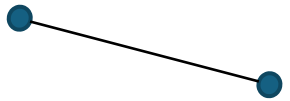
Calculate Npcf

1

Npcf Result
Calculate 2pcf 3pcf MGS

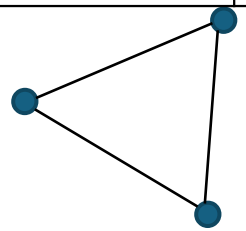


2pcf



	ISO	ANISO
R [Mpc]	180	180
Bins	36	360

3pcf

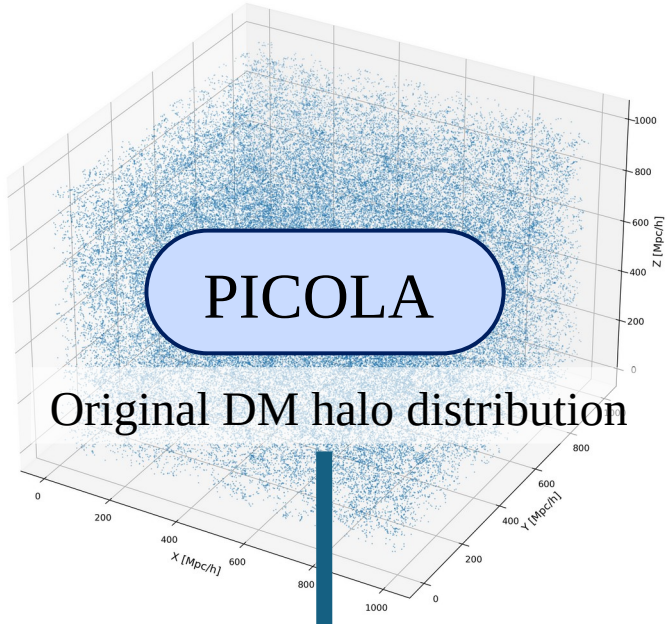


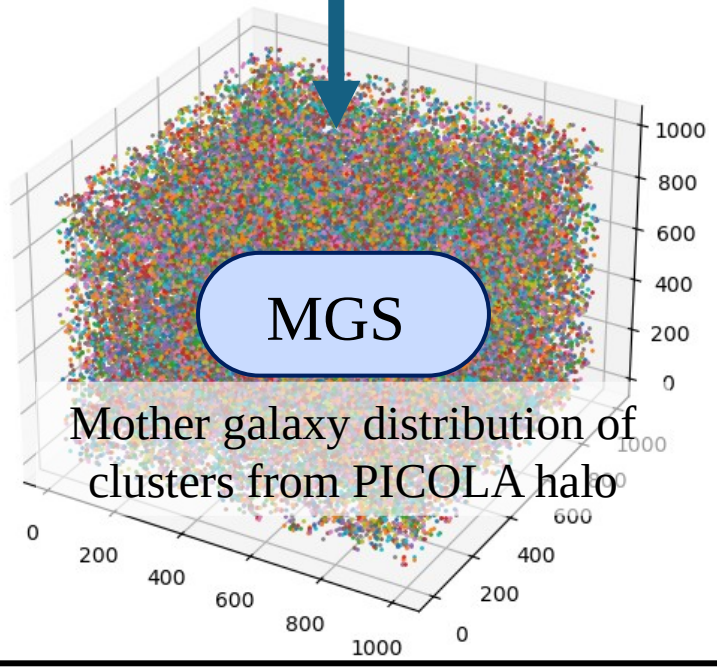
	ISO	ANISO
R [Mpc]	100	100
Bins	1540	15400

Calculate Npcf

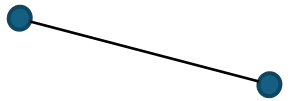
1

Npcf Result
Calculate 2pcf 3pcf MGS



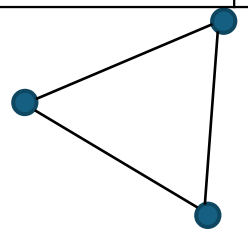


2pcf



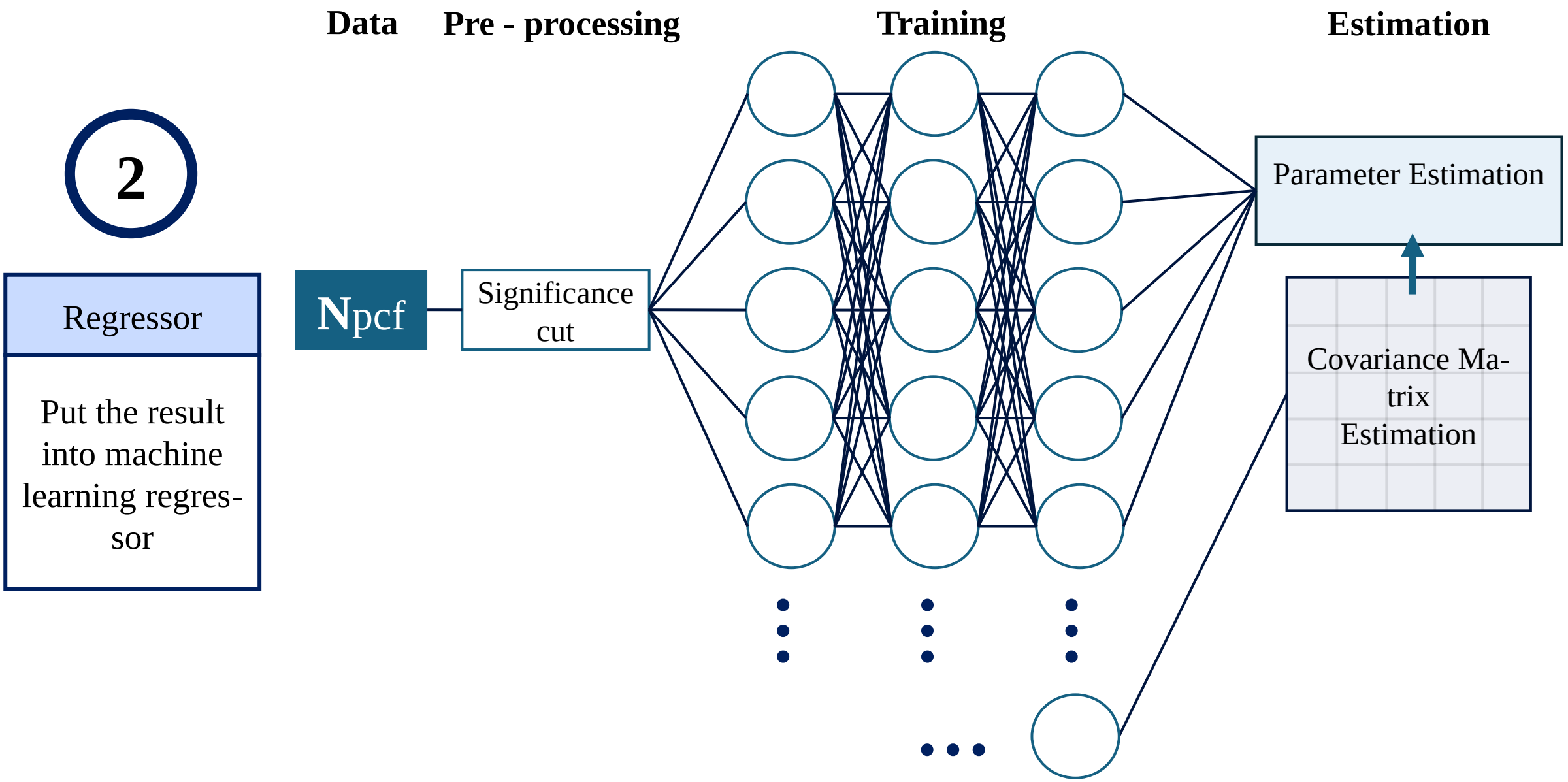
	ISO	ANISO
R [Mpc]	180	180
Bins	36	360

3pcf



	ISO	ANISO
R [Mpc]	100	100
Bins	1540	15400

Learning Structure



Estimation Range

3

Estimation
Cosmological parameter estimation

1000
Different Cosmology

	0.25 ~ 0.4
	0.75 ~ 0.85
	-1.3 ~ -0.7
	0.9 ~ 1.0
	0.6 ~ 0.8

100
Single Cosmology

	0.3136
	0.7814
	-1.0039
	0.9418
	0.7103

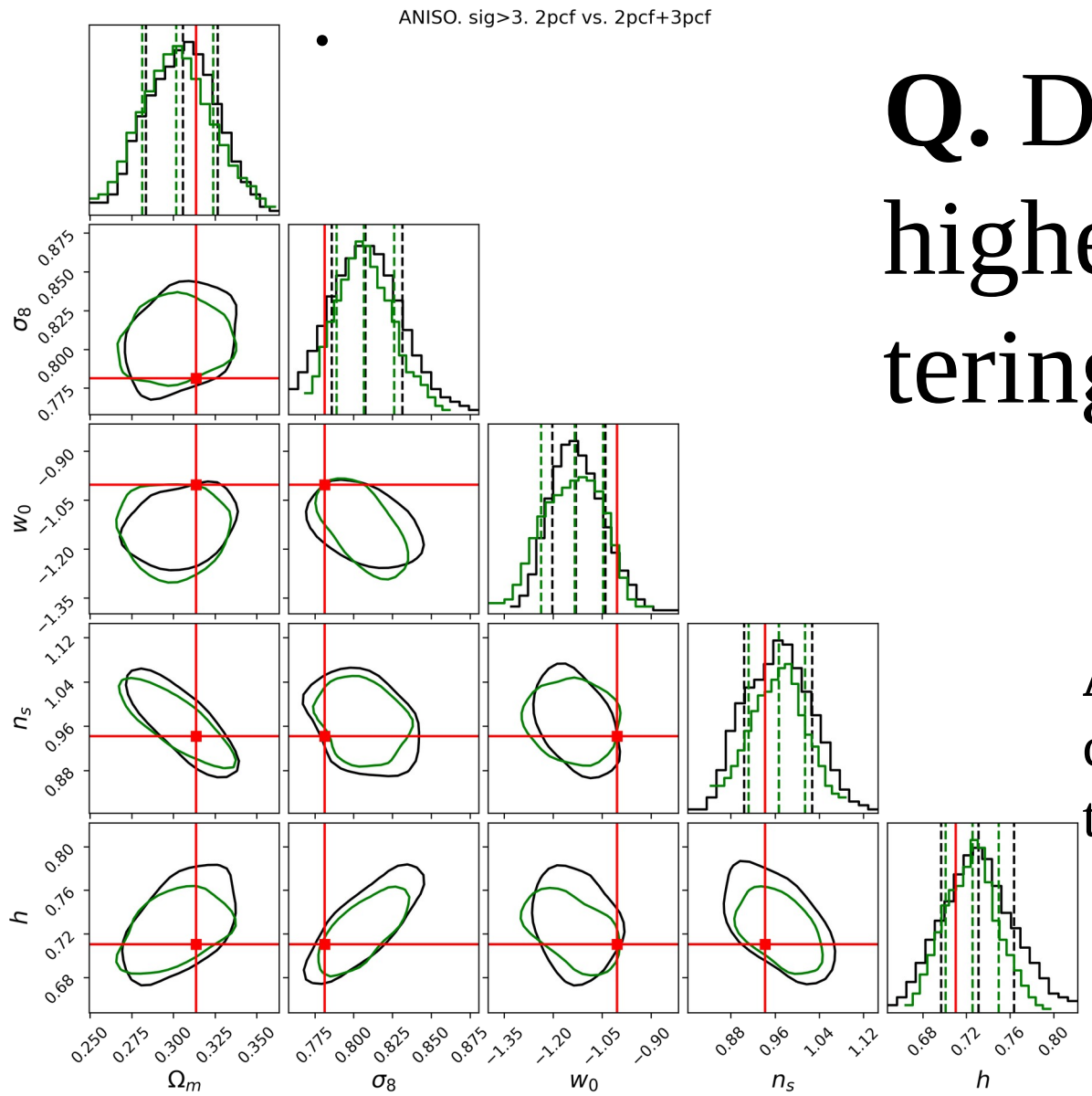
Result

2pcf vs **2pcf+3pcf**

4

Contour

Compare the contours of each parameters and Npcfs



Q. Do we need higher order clustering statistics?

A. Yes, it helps parameter contours more constrained than 2pcf

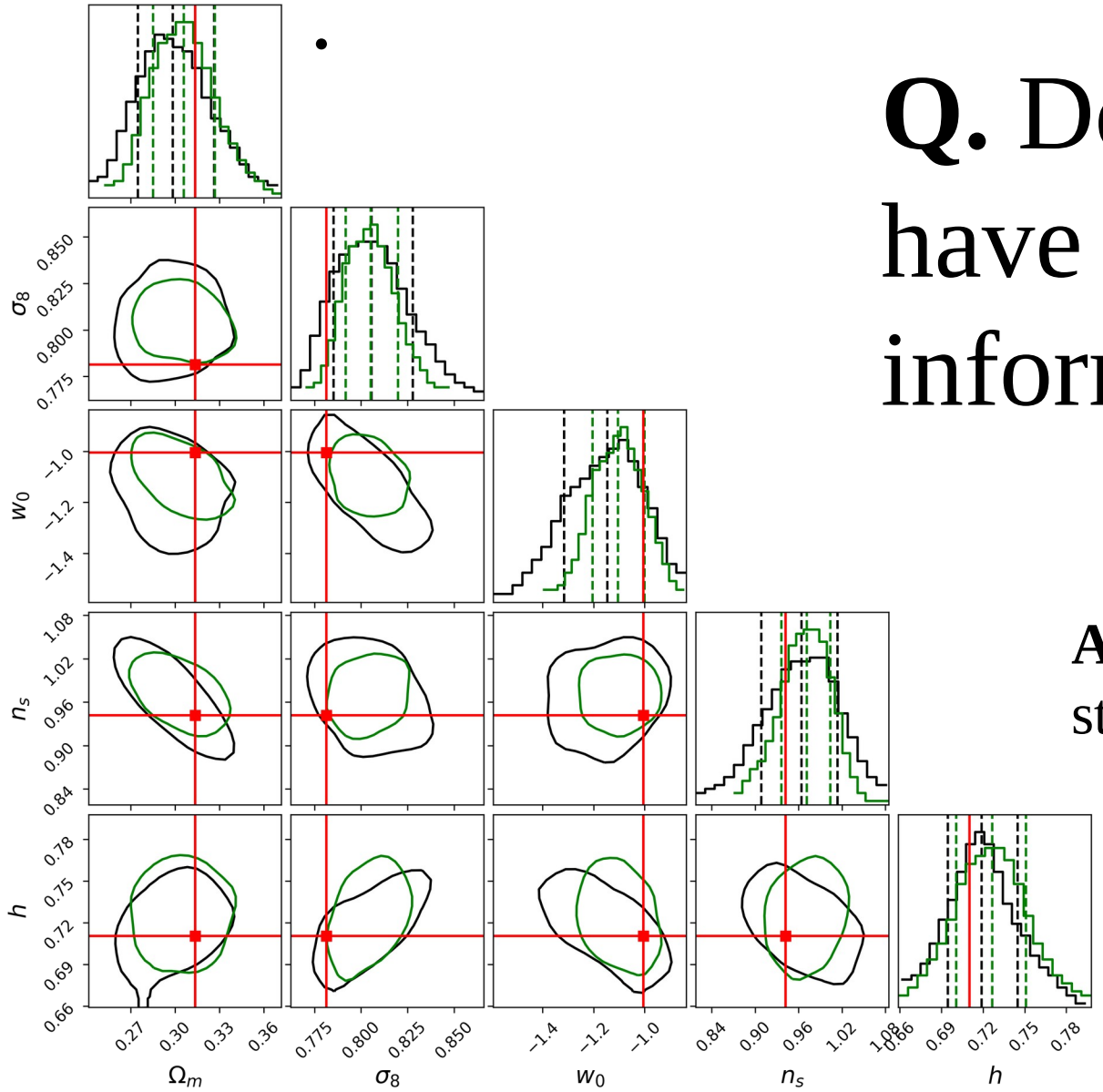
Result

Original vs **Original+MGS**

4

Contour

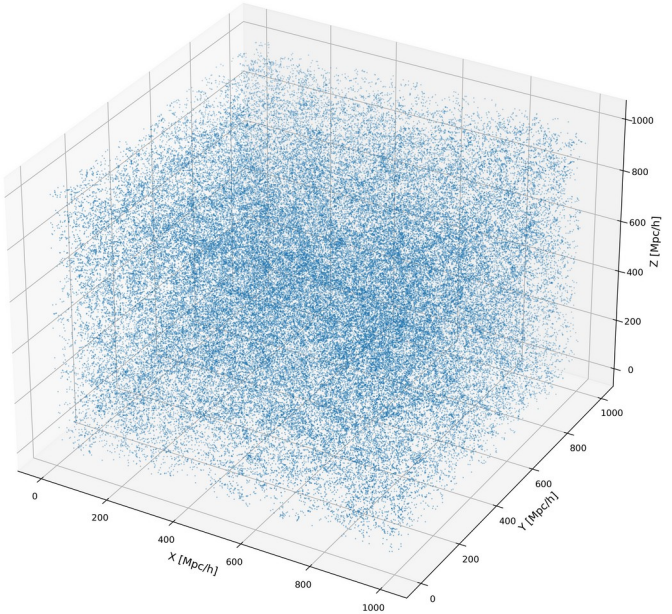
Compare the contours of each parameters and Npcfs



Q. Do MGS Npcfs have cosmological information?

A. Yes, and it does help constrain parameter contours.

Summary

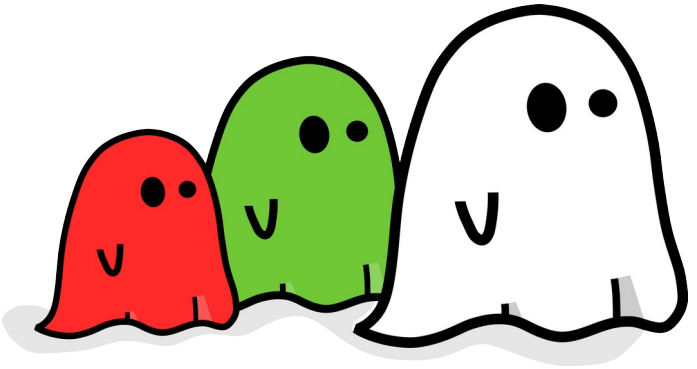


1. LSS has cosmological informations in it and N point correlation function is the common way to investigate them.

2. We tried to get cosmological information by calculating Npcf and put them into neural network regressor and estimate with covariance matrix at same time.

3. Our goal is to show …

- Higher order statistics gain more information
- MGS calculated Npcfs have cosmological information



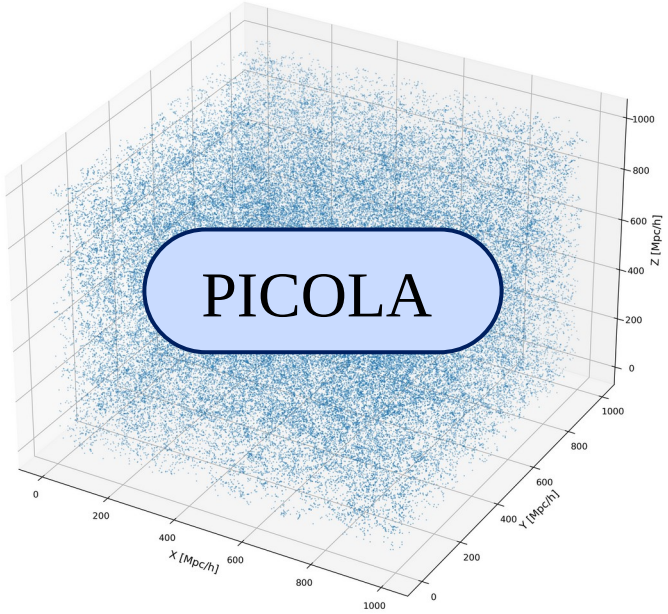
4. We are now working on comparing with ISO and ANISO and combining informations of all Npcfs we have.

Thanks for listening!

Back Up

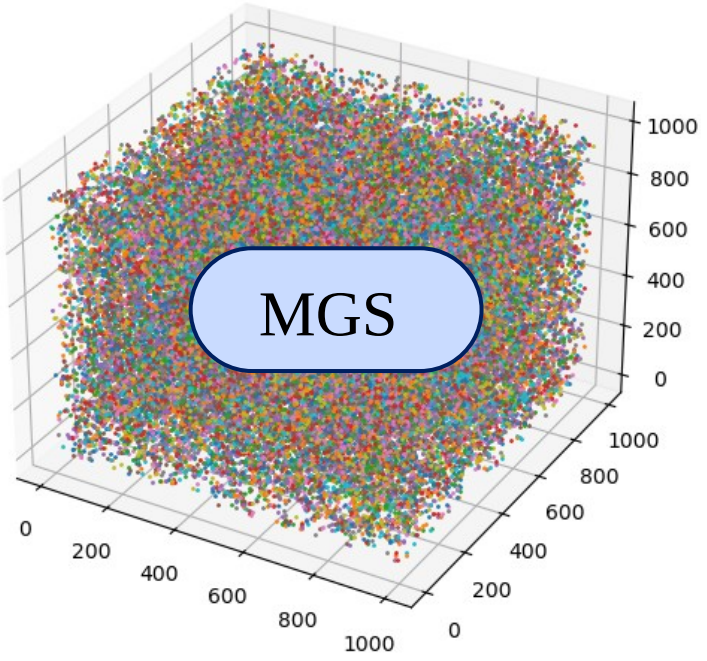
Particle differernce

Dark matter halo distribution



N

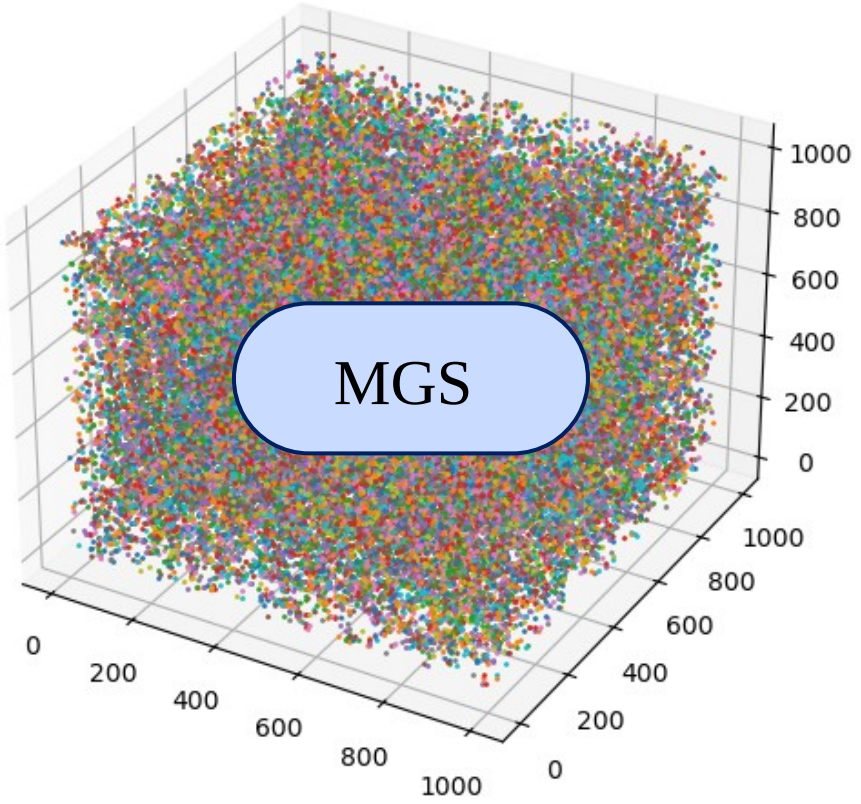
Cluster from DM halo distribution



$0.79N$

Linking Length

$$l_{link} = b\left(\frac{V_u}{N}\right)^{1/3}$$



$$¿ 0.2\left(\frac{10^9 (Mpc/h)^3}{N}\right)^{1/3}$$

4.76

Data Pre – processing

x = Single Cosmology

```
mx=np.mean(x,axis=0)
sx=np.std(x,axis=0)
sig=np.abs(mx/sx)

msk=np.abs(mx/sx)>3.
```

y = Different Cosmology

```
y=[]
for i in range(1000):
    y.append(x[i,msk])
y=np.asarray(y)

maxy=np.max(y,axis=0)
miny=np.min(y,axis=0)
meany=np.mean(y,axis=0)
stdy=np.std(y,axis=0)

z=(y-meany)/(3*stdy)
single_norm=(single-meany)/
(3*stdy)
```

Loss function & Estimation

Custom Loss function using covariance matrix

Make the values of covariance matrix at minimum

```
def custom_loss(y_true, y_pred, model):
    """
    Custom loss function using global covariance matrix.

    Args:
        y_true: True values, shape (batch_size, n_params)
        y_pred: Predicted mean values, shape (batch_size, n_params)
        model: The model instance to access the covariance matrix.

    Returns:
        Negative log-likelihood loss
    """
    # Get the global covariance matrix
    cov_matrix = model.get_covariance_matrix()

    # Create a multivariate normal distribution with the global covariance matrix
    mvn = tfp.distributions.MultivariateNormalFullCovariance(loc=y_pred, covariance_matrix=cov_matrix)

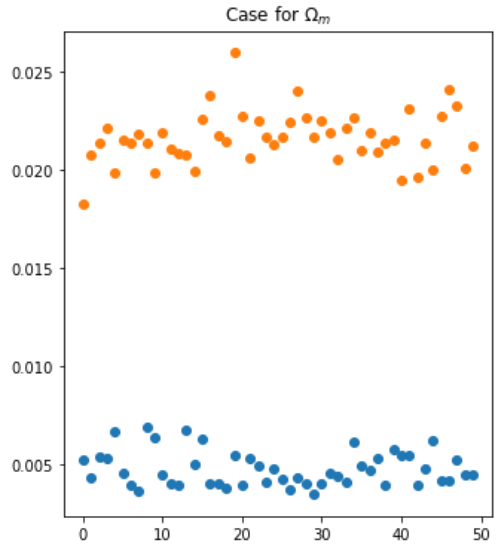
    # Calculate the negative log-likelihood
    neg_log_likelihood = -mvn.log_prob(y_true)

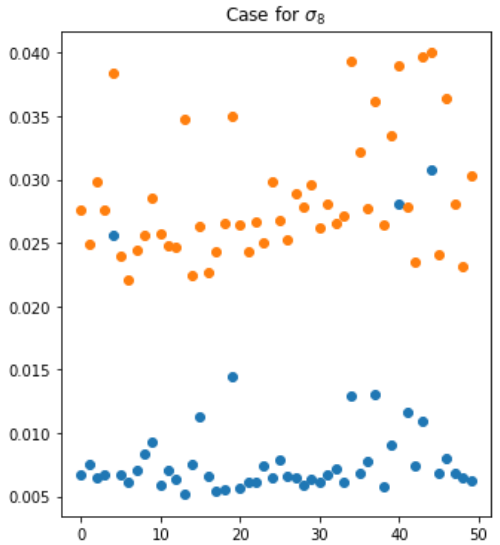
    # Return mean loss over the batch
    return tf.reduce_mean(neg_log_likelihood) +  model.covariance_regularization()
```

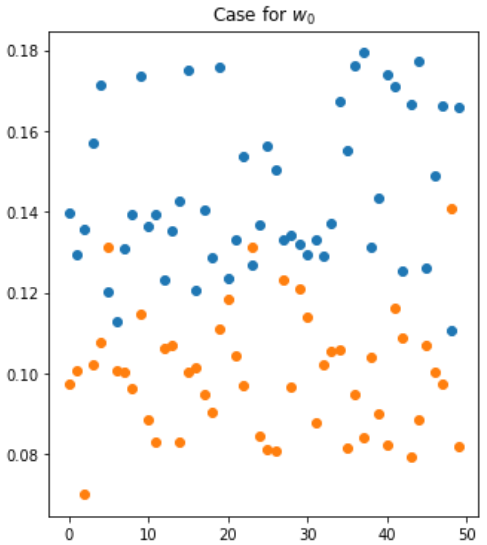
Covariance Matrix

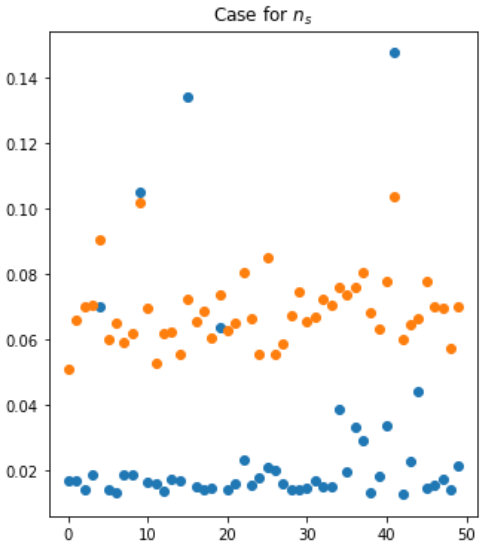
- : covariance error
- : contour prediction error

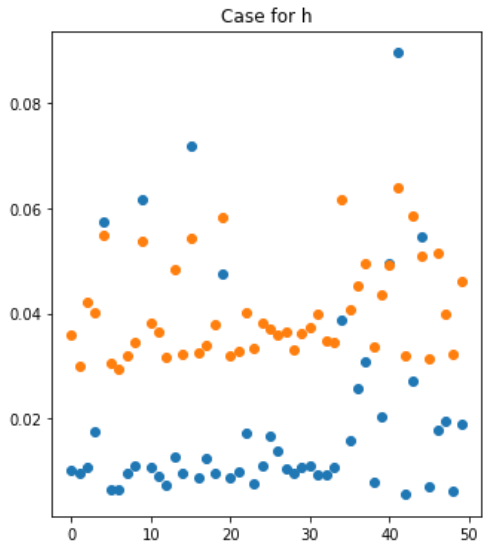
ANISO_2pcf. sig>3. Error check



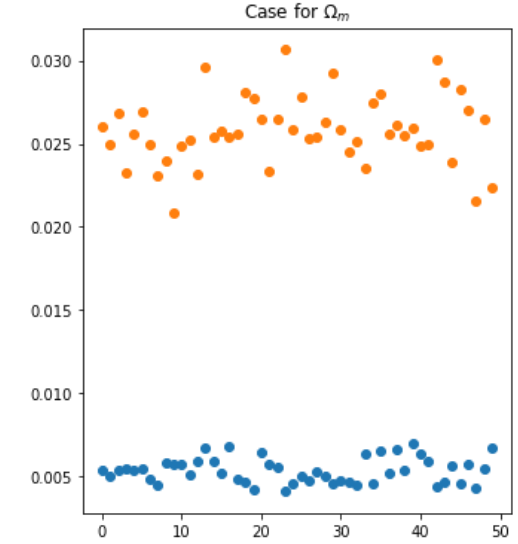


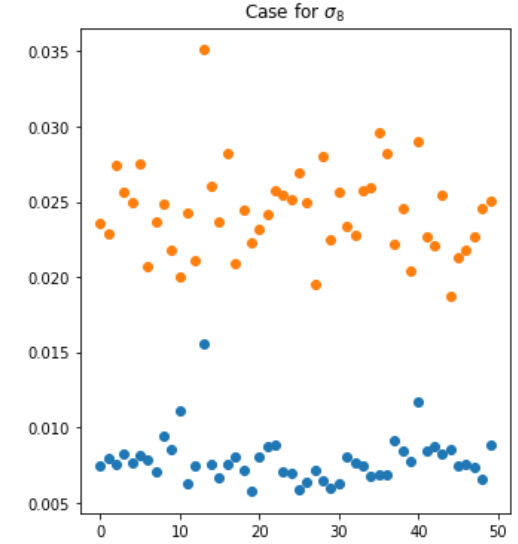


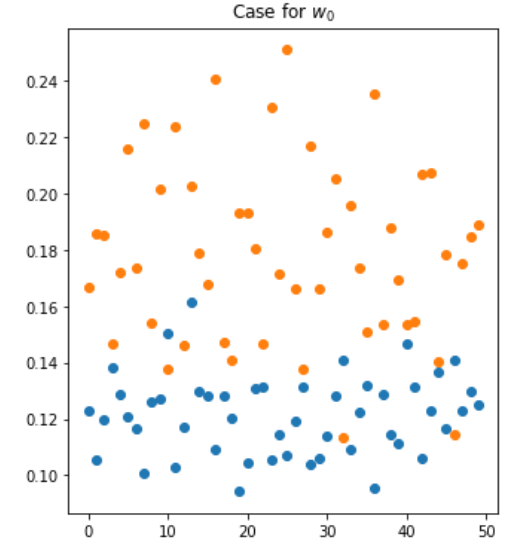


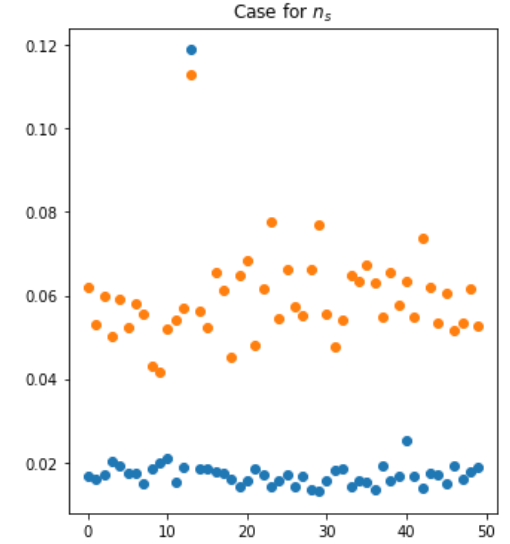


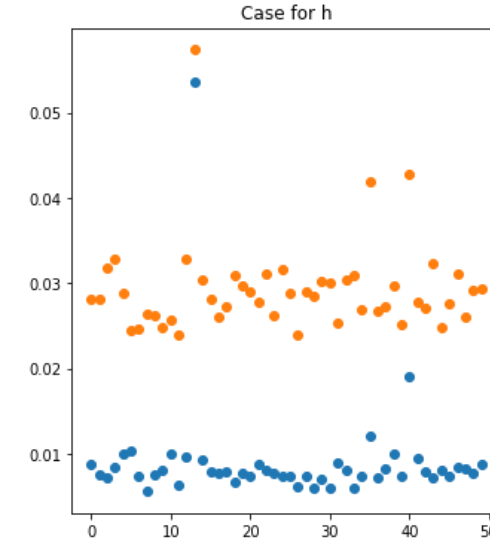
ANISO_3pcf. sig>3. Error check











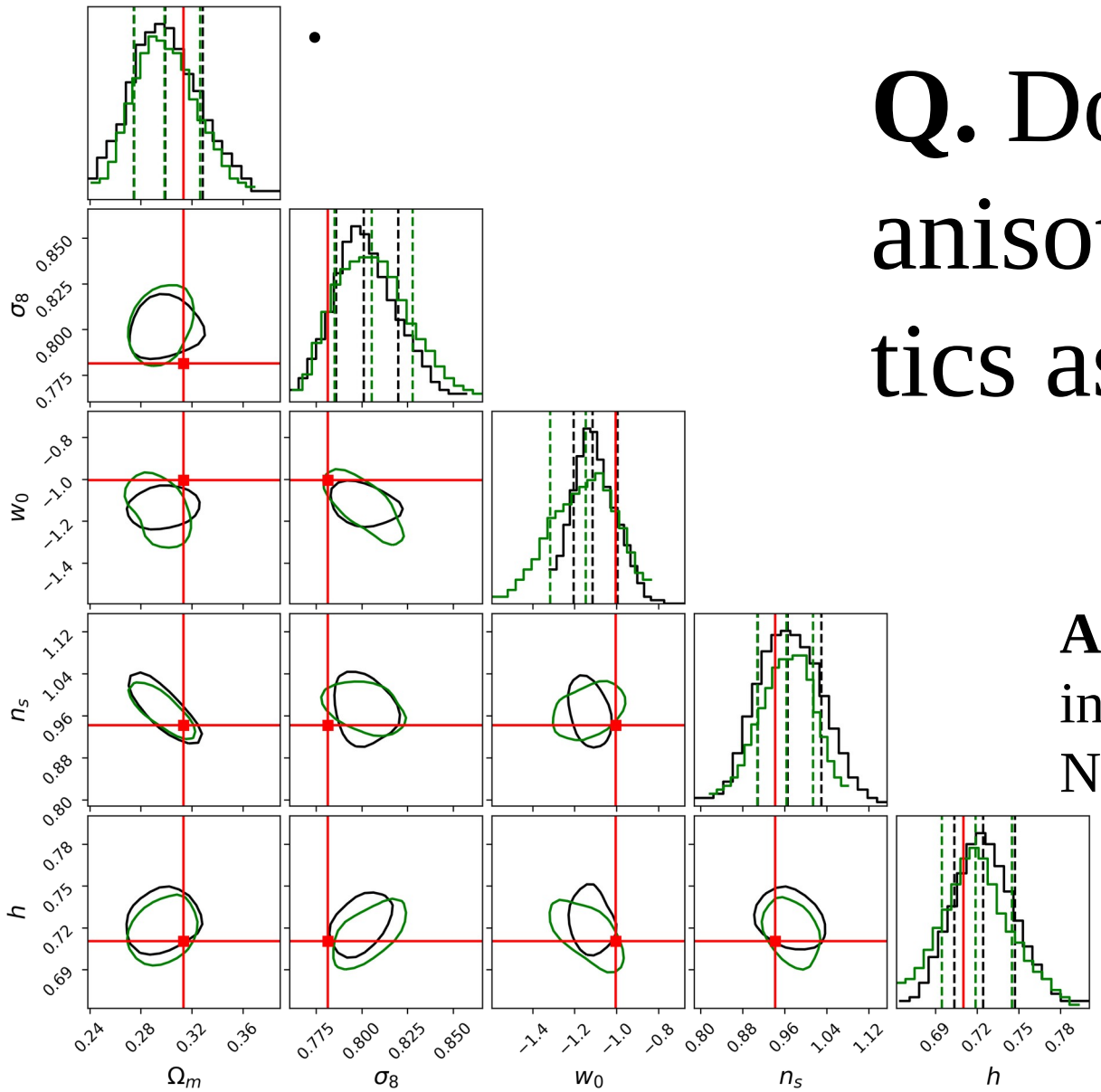
Result

ISO vs ANISO

4

Contour

Compare the contours of each parameters and Npcfs



Q. Do we need anisotropic statistics as well?

A. Yes, it includes different information from isotropic Npcfs

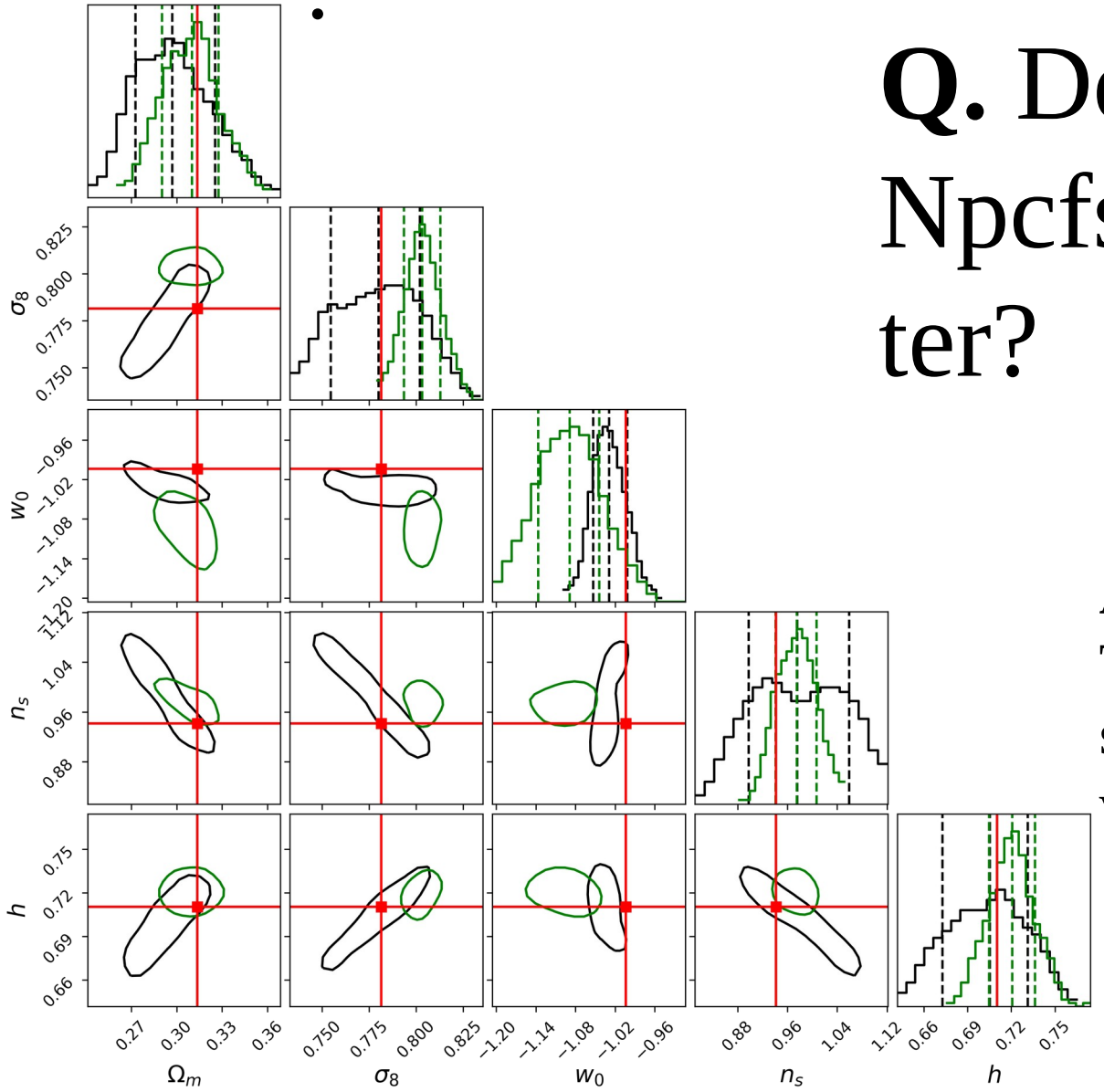
Result

ISO 2p vs Combined

4

Contour

Compare the contours of each parameters and Npcfs

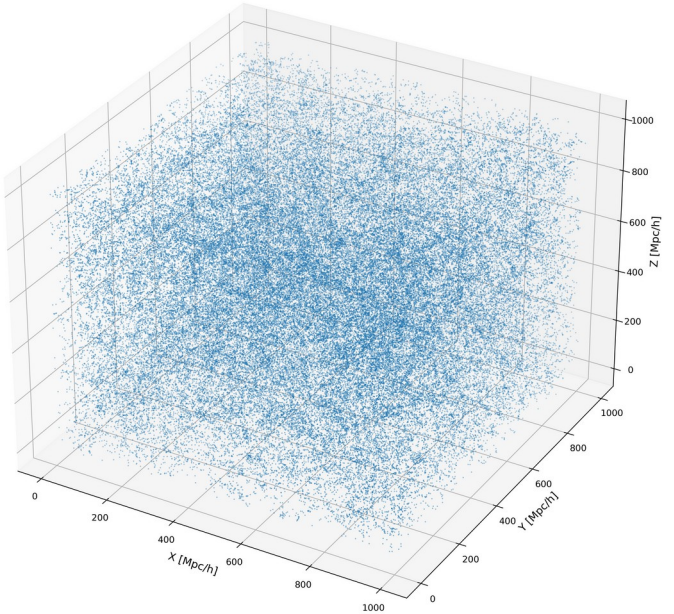


Q. Do combined Npcfs estimate better?

A. Yes, In some cases. They're mostly more constrained but miss the truth value.

Summary

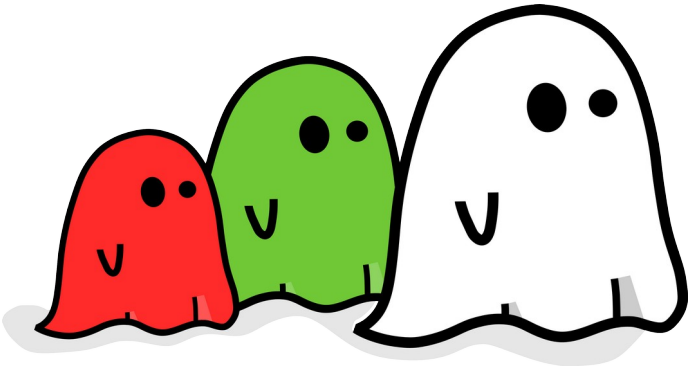
1. LSS has cosmological informations in it and N point correlation function is the common way to investigate them.



2. We tried to get cosmological information by calculating Npcf and put them into neural network regressor and estimate with covariance matrix at same time.

3. Our goal is to show …

- Higher order statistics gain more information
- ~~Anisotropic Npcfs gain more information~~
- Simplified distribution(using MGS) have different information



~~4.~~ ~~But it seems adding all different Npcfs is not doing much better.~~

5. We are now working on it and will investigate more about this.