

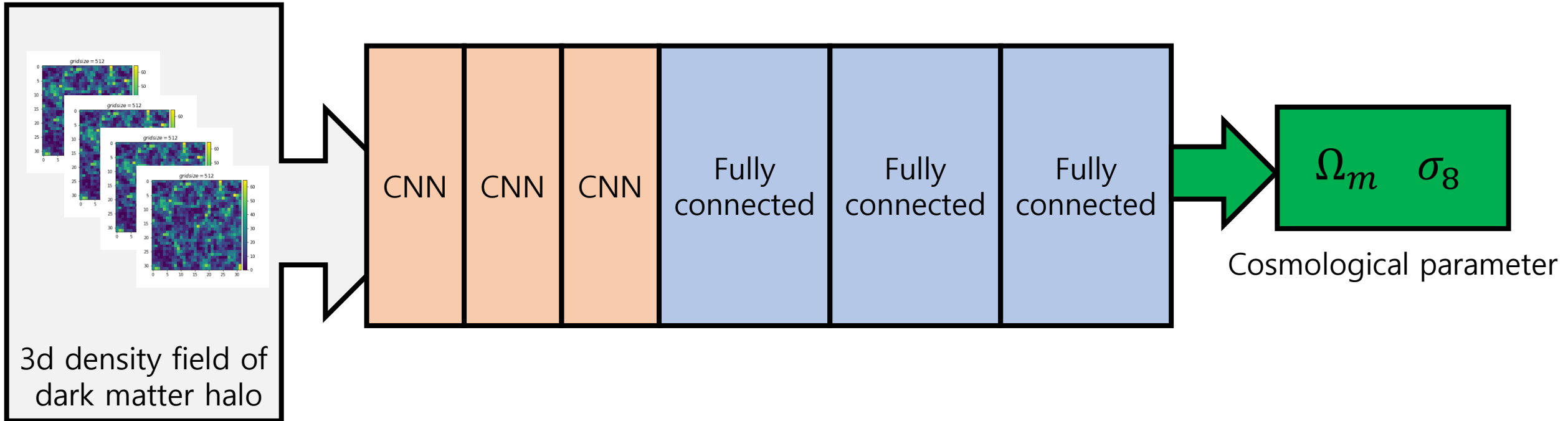
Cosmology with Large scale structure using Machine Learning

2022.02.07

<https://gitlab.sccc.uos.ac.kr/worldhsy/clml>

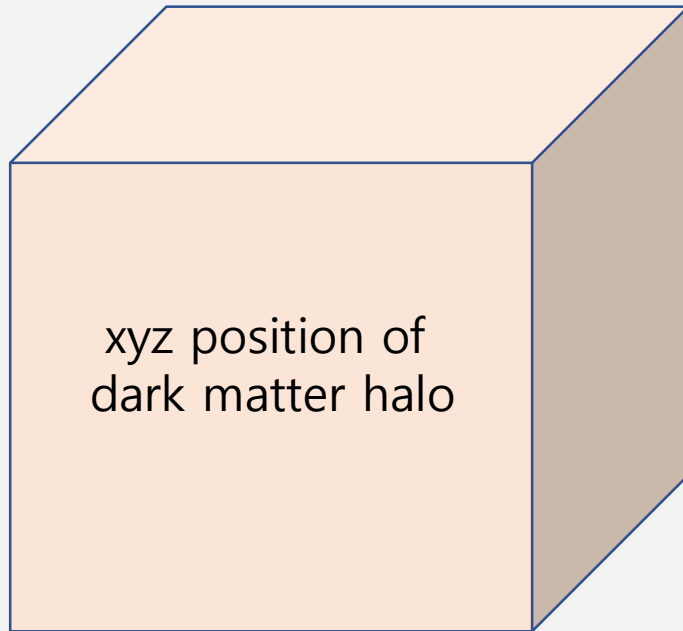
outlook

- I started on the machine leaning!([arXiv : 1908.10590](#))

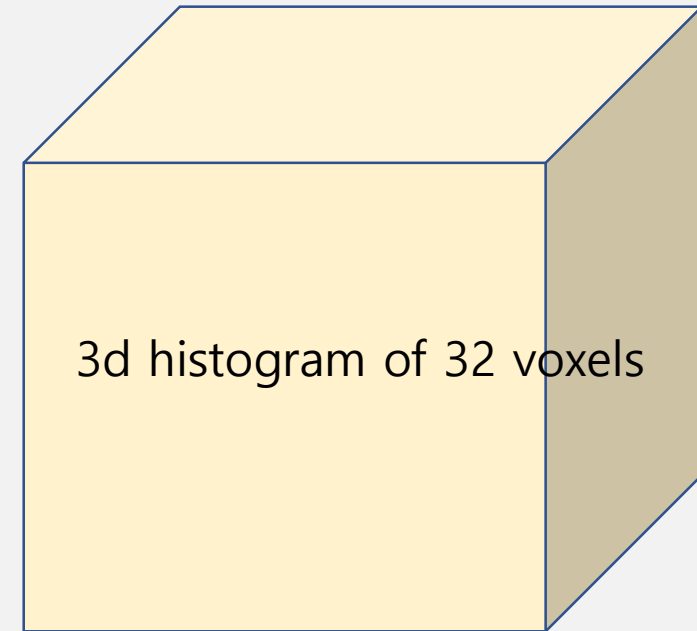
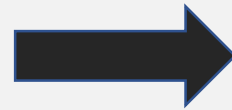


DATA

previous

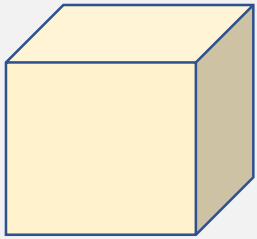


- Box size = 256(Mpc/h)
- Number of particles = 256^3
- Grid voxel = 256^3

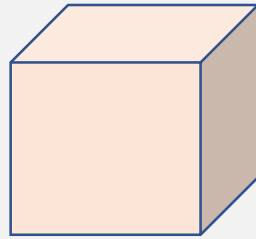


- 1 voxel have 8(Mpc/h) information

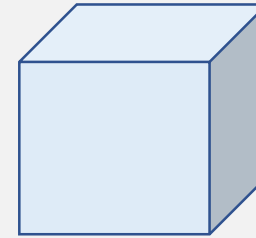
Change for grid size



- Box size = 256(Mpc/h)
- Number of particles = 128^3
- Grid voxel = 128^3



- Box size = 256(Mpc/h)
- Number of particles = 256^3
- Grid voxel = 256^3



- Box size = 256(Mpc/h)
- Number of particles = 512^3
- Grid voxel = 512^3

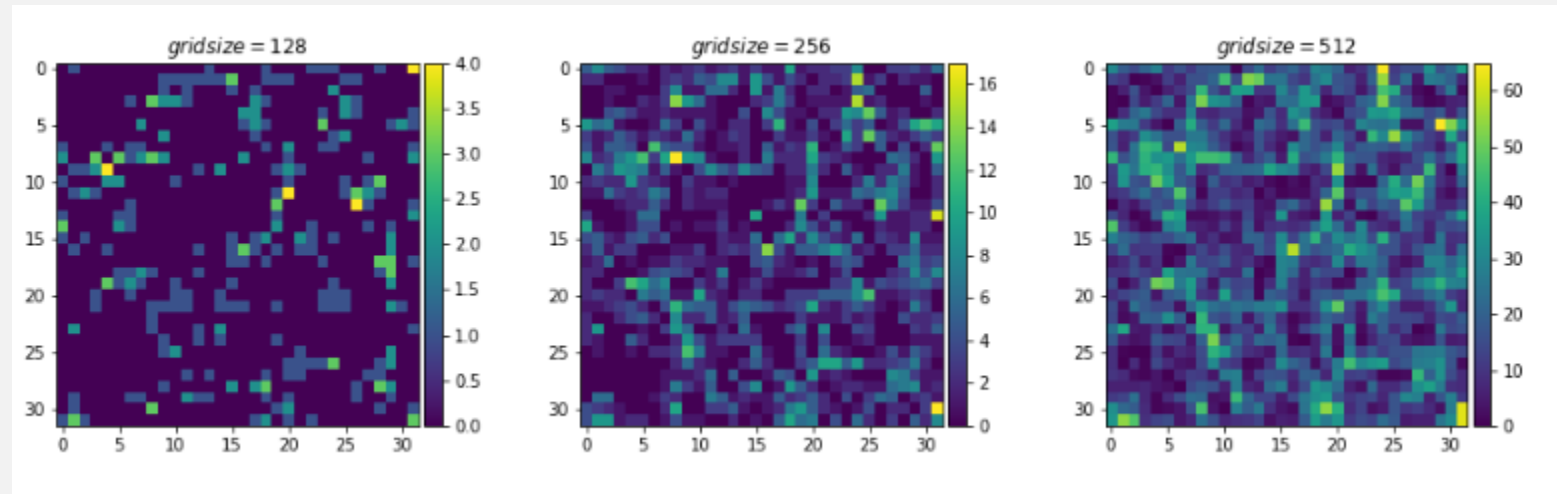
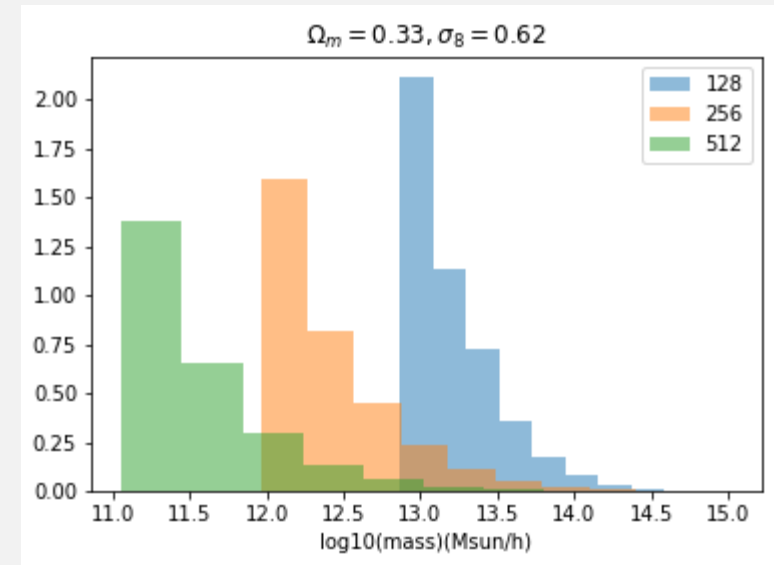
Data

Number of halo

```
the number of halo(128) : 9225  
the number of halo(256) : 74646  
the number of halo(512) : 504649
```

$$\rho = \frac{N * M}{V}$$

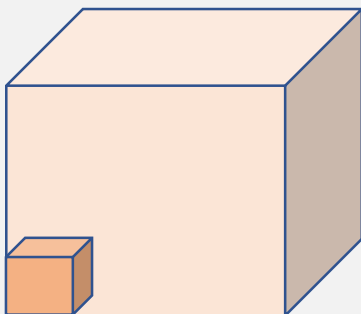
Halo mass



DATA

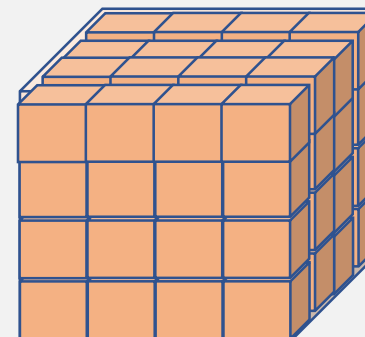
- 1 voxel have $2(\text{Mpc}/h)$ information

(Short version)



- Only using $0(\text{Mpc}/h) < x,y,z < 64 (\text{Mpc}/h)$
- Including only 0~64 information

(cut version)



- At each side : $[0,64] [64,128] [128, 192] [192, 256] >>$ 32 voxel histogram
- Including all information of the box scale
- Total $64(4^3)$ for one cosmology

Data set

128_short

128_cut

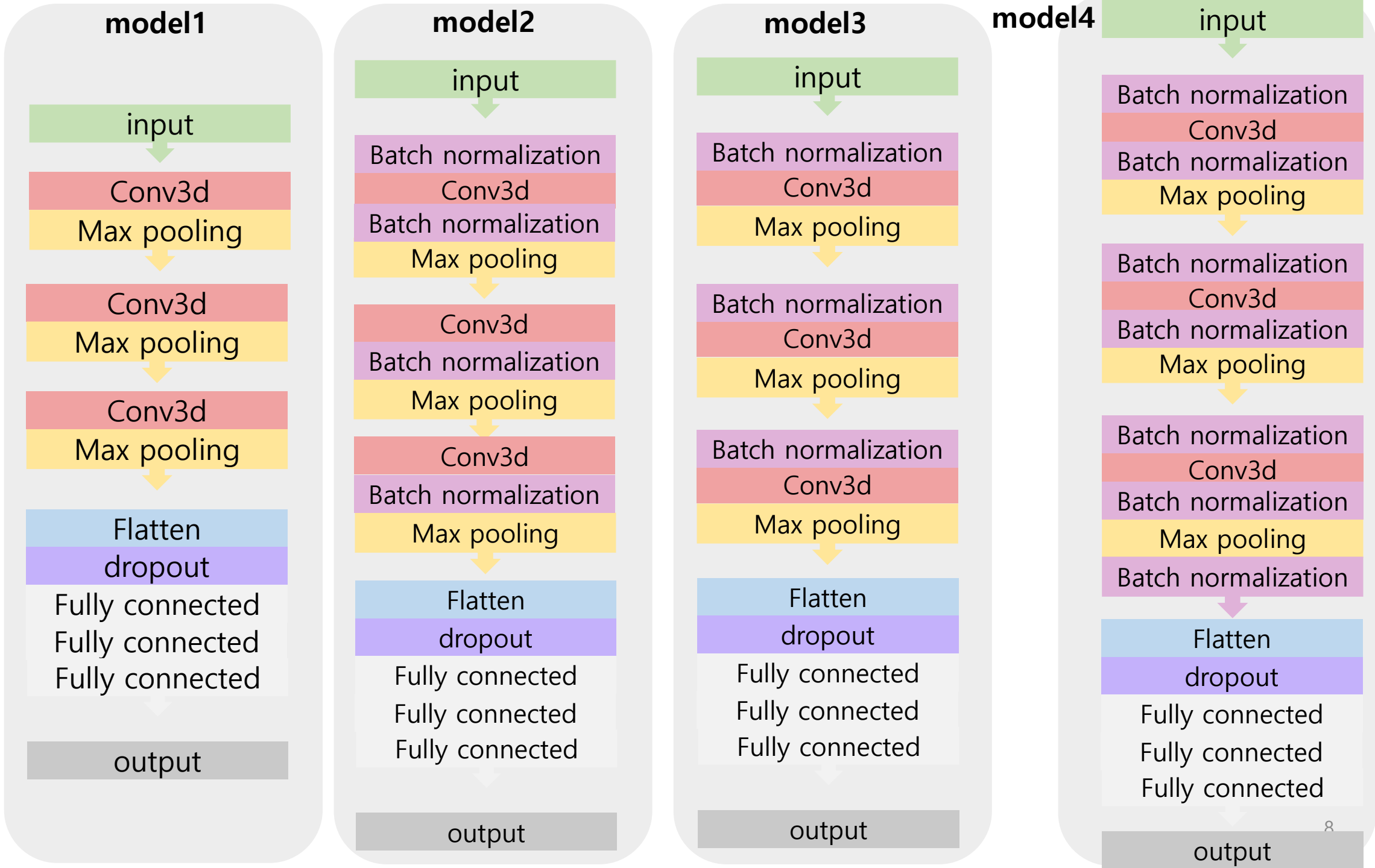
256_short

256_cut

512_short

512_cut

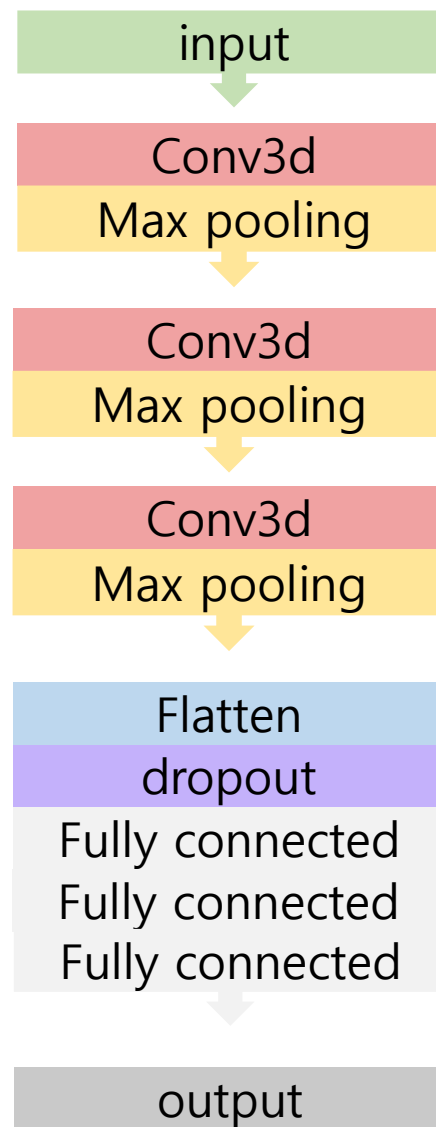
Mo
Del



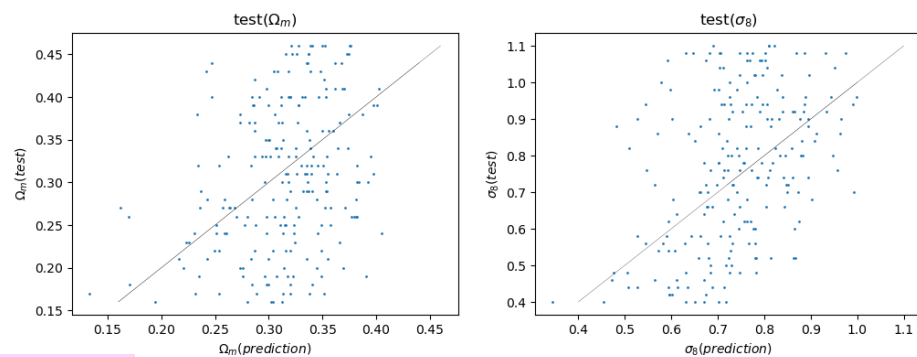
model1

Model: "model"

Layer (type)	Output Shape	Param #
=====		
input_1 (InputLayer)	[(None, 32, 32, 32, 1)]	0
conv3d (Conv3D)	(None, 30, 30, 30, 32)	896
max_pooling3d (MaxPooling3D)	(None, 15, 15, 15, 32)	0
conv3d_1 (Conv3D)	(None, 13, 13, 13, 64)	55360
max_pooling3d_1 (MaxPooling3D)	(None, 6, 6, 6, 64)	0
conv3d_2 (Conv3D)	(None, 4, 4, 4, 128)	221312
max_pooling3d_2 (MaxPooling3D)	(None, 2, 2, 2, 128)	0
flatten (Flatten)	(None, 1024)	0
dropout (Dropout)	(None, 1024)	0
dense (Dense)	(None, 1024)	1049600
dense_1 (Dense)	(None, 256)	262400
dense_2 (Dense)	(None, 2)	514
=====		
Total params: 1,590,082		
Trainable params: 1,590,082		
Non-trainable params: 0		



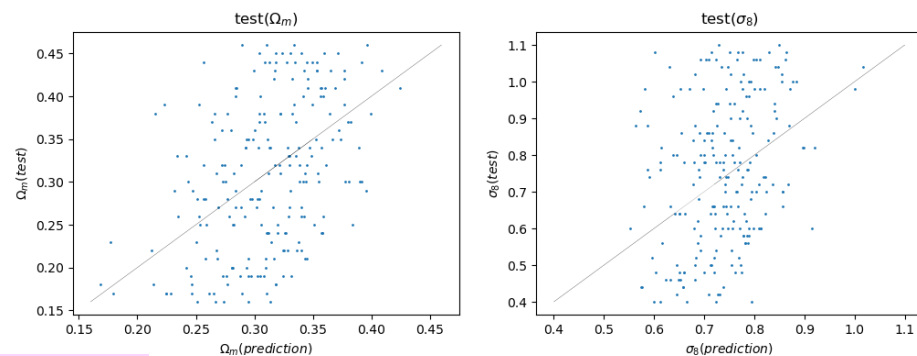
128_short



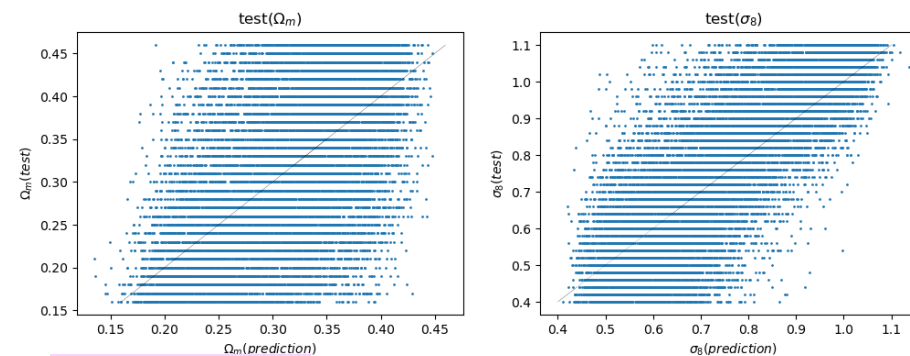
128_cut



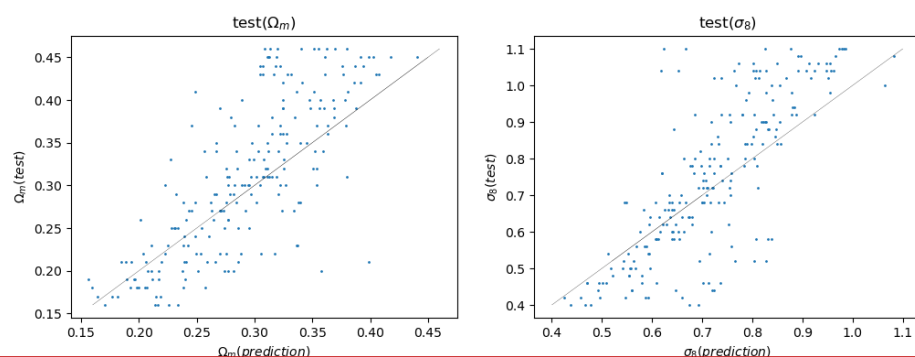
256_short



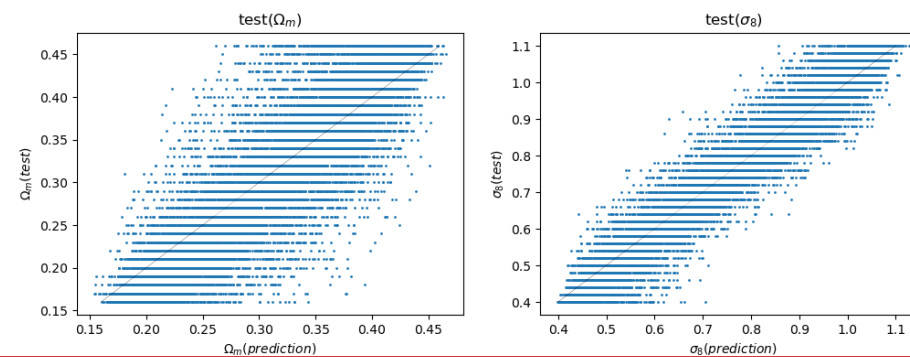
256_cut



512_short

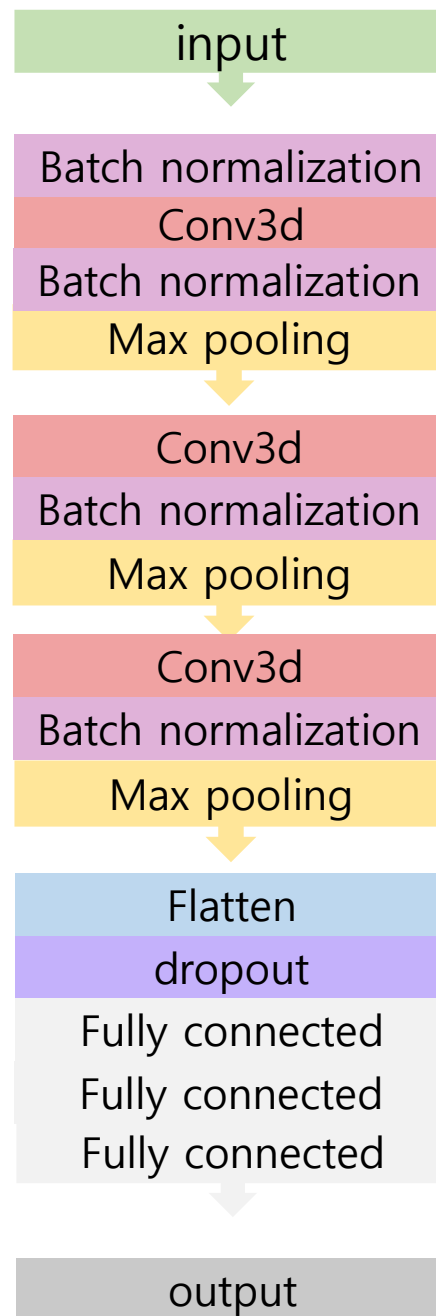


512_cut

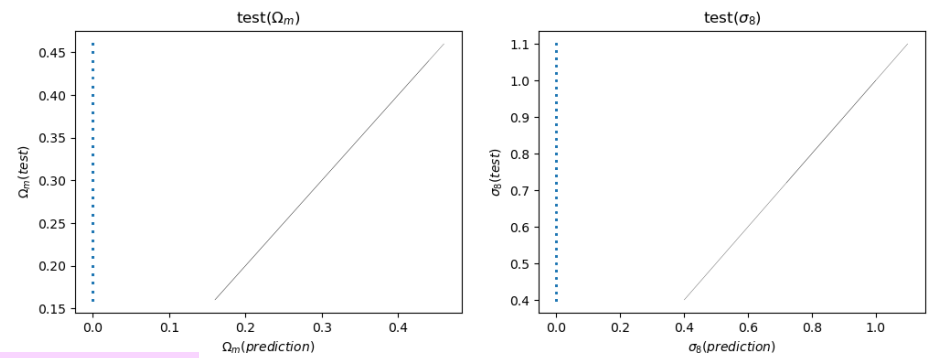


model2

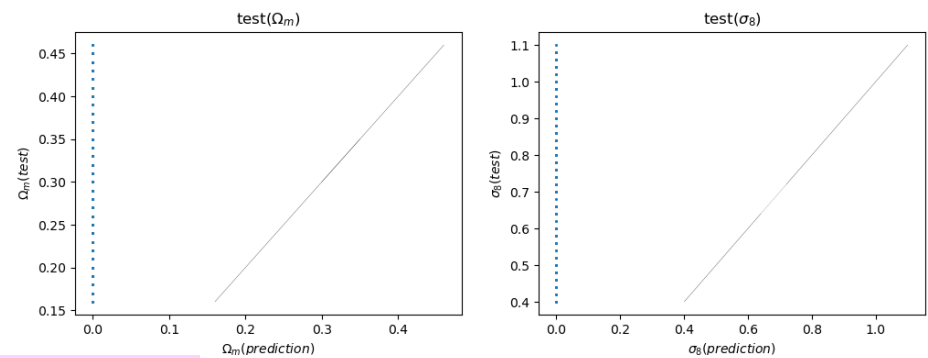
Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 32, 32, 32, 1)]	0
batch_normalization (Batch Normalization)	(None, 32, 32, 32, 1)	4
conv3d (Conv3D)	(None, 30, 30, 30, 32)	896
batch_normalization_1 (Batch Normalization)	(None, 30, 30, 30, 32)	128
max_pooling3d (MaxPooling3D)	(None, 15, 15, 15, 32)	0
conv3d_1 (Conv3D)	(None, 13, 13, 13, 64)	55360
batch_normalization_2 (Batch Normalization)	(None, 13, 13, 13, 64)	256
max_pooling3d_1 (MaxPooling3D)	(None, 6, 6, 6, 64)	0
conv3d_2 (Conv3D)	(None, 4, 4, 4, 128)	221312
batch_normalization_3 (Batch Normalization)	(None, 4, 4, 4, 128)	512
max_pooling3d_2 (MaxPooling3D)	(None, 2, 2, 2, 128)	0
flatten (Flatten)	(None, 1024)	0
dropout (Dropout)	(None, 1024)	0
dense (Dense)	(None, 1024)	1049600
dense_1 (Dense)	(None, 256)	262400
dense_2 (Dense)	(None, 2)	514
Total params: 1,590,982		
Trainable params: 1,590,532		
Non-trainable params: 450		



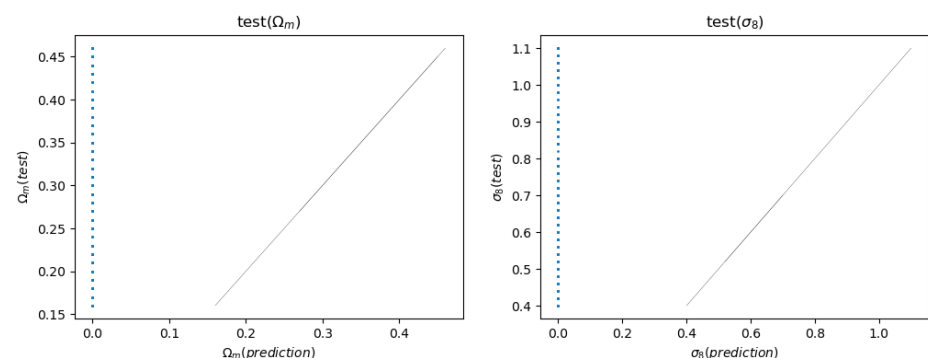
128_short



256_short



512_short



128_cut



256_cut

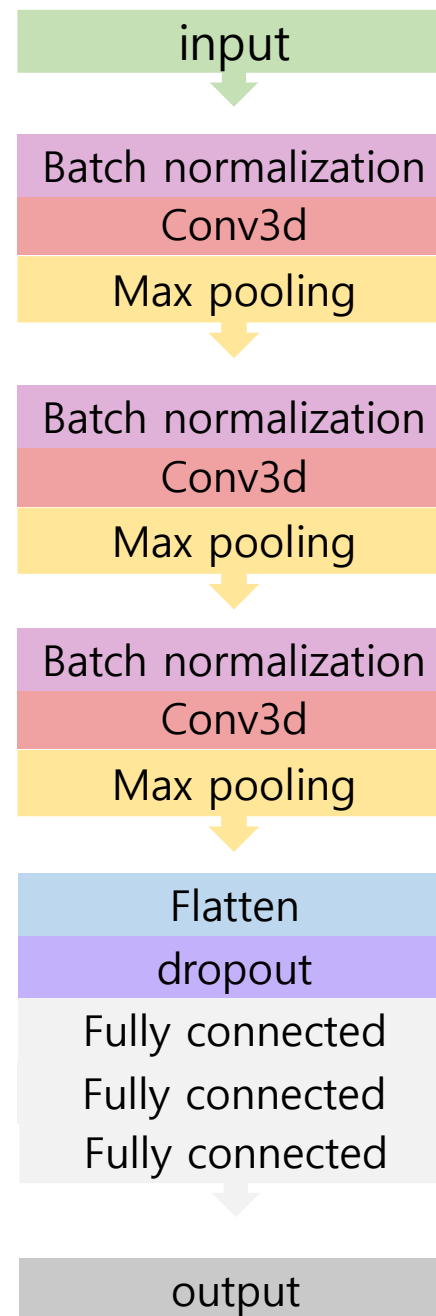


512_cut

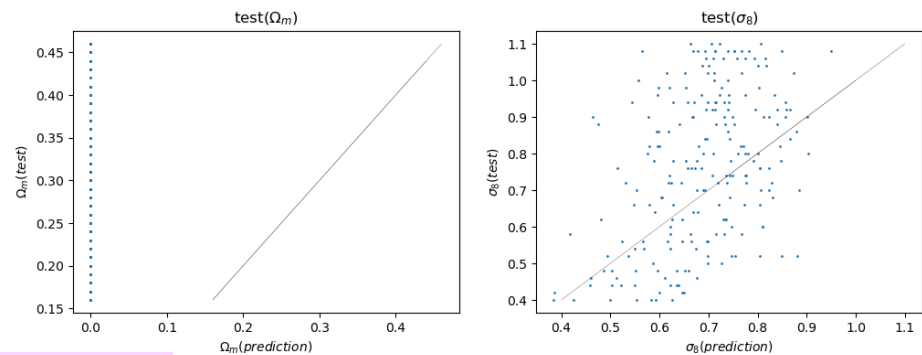


Model3

Layer (type)	Output Shape	Param #
=====		
input_1 (InputLayer)	[(None, 32, 32, 32, 1)]	0
batch_normalization (Batch Normalization)	(None, 32, 32, 32, 1)	4
conv3d (Conv3D)	(None, 30, 30, 30, 32)	896
max_pooling3d (MaxPooling3D)	(None, 15, 15, 15, 32)	0
batch_normalization_1 (Batch Normalization)	(None, 15, 15, 15, 32)	128
conv3d_1 (Conv3D)	(None, 13, 13, 13, 64)	55360
max_pooling3d_1 (MaxPooling3D)	(None, 6, 6, 6, 64)	0
batch_normalization_2 (Batch Normalization)	(None, 6, 6, 6, 64)	256
conv3d_2 (Conv3D)	(None, 4, 4, 4, 128)	221312
max_pooling3d_2 (MaxPooling3D)	(None, 2, 2, 2, 128)	0
flatten (Flatten)	(None, 1024)	0
dropout (Dropout)	(None, 1024)	0
dense (Dense)	(None, 1024)	1049600
dense_1 (Dense)	(None, 256)	262400
dense_2 (Dense)	(None, 2)	514
=====		
Total params: 1,590,470		
Trainable params: 1,590,276		
Non-trainable params: 194		



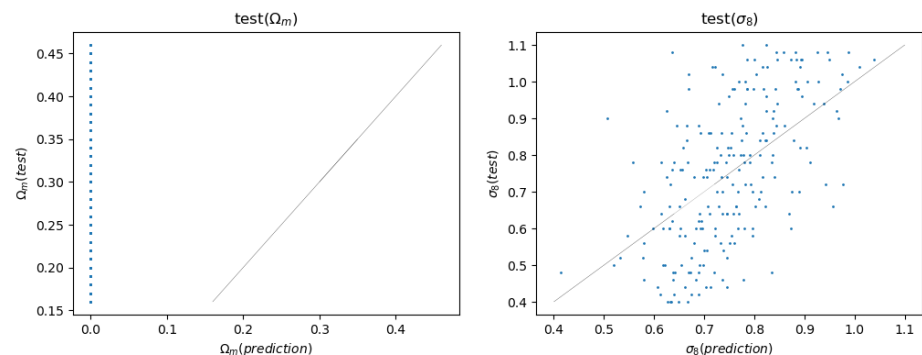
128_short



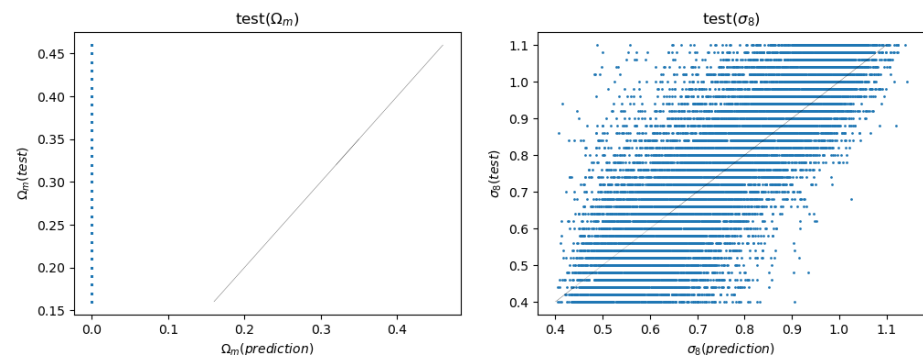
128_cut



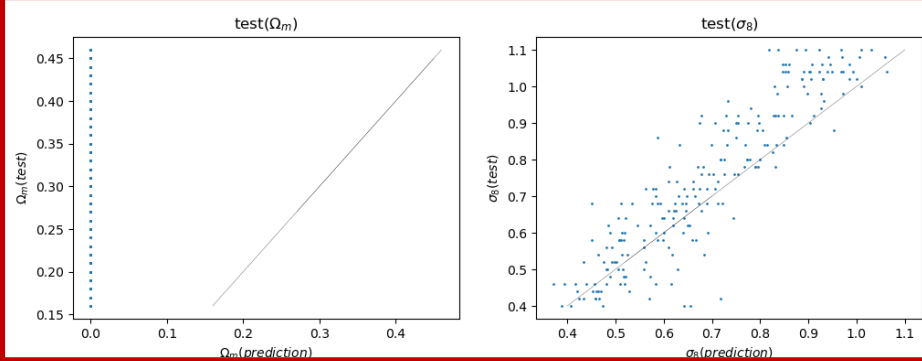
256_short



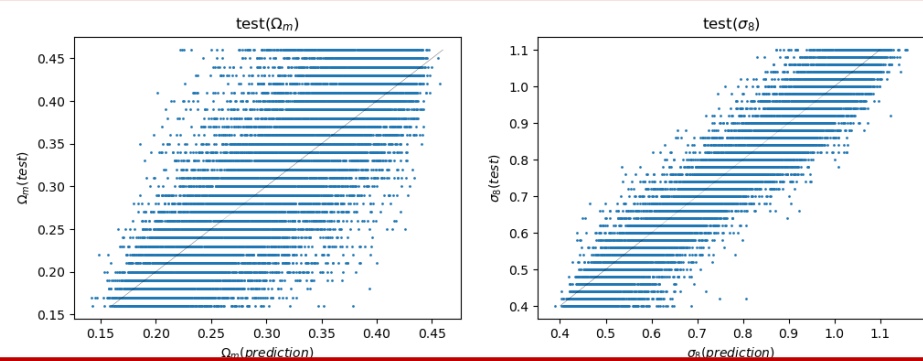
256_cut



512_short



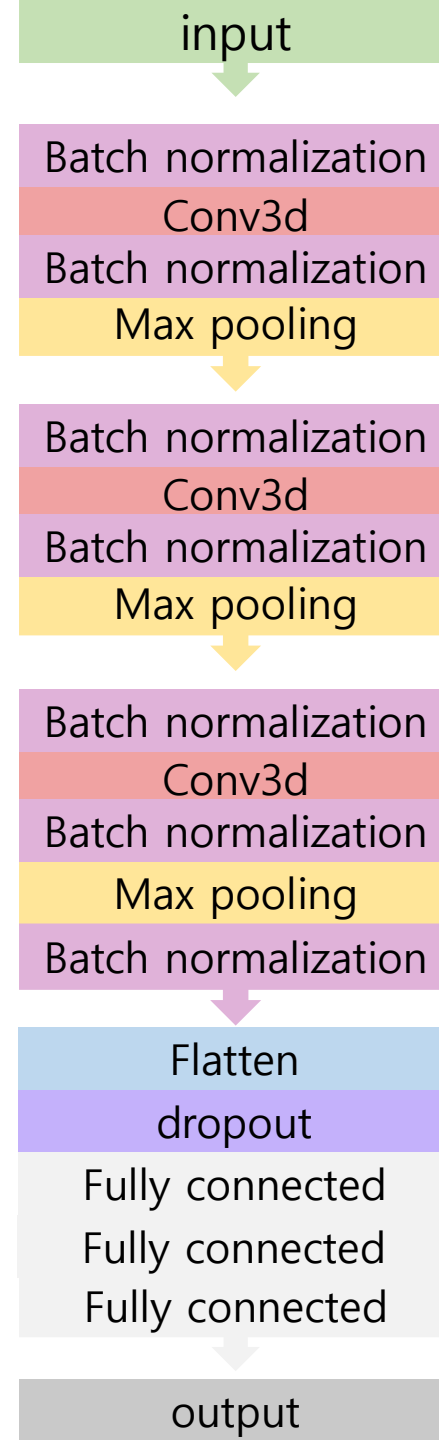
512_cut



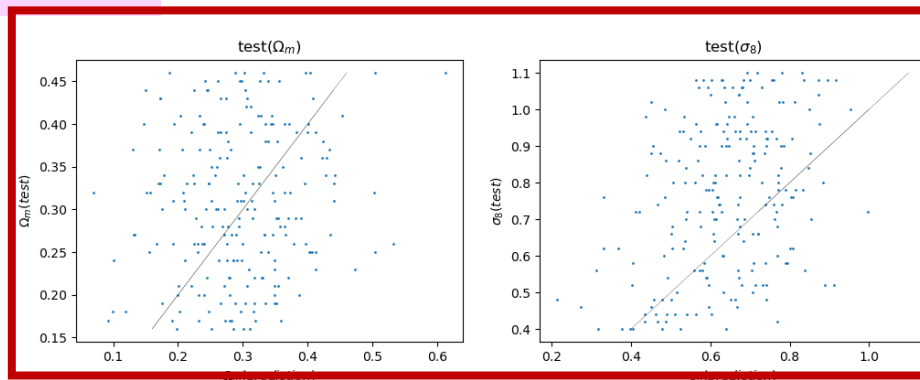
?

model4

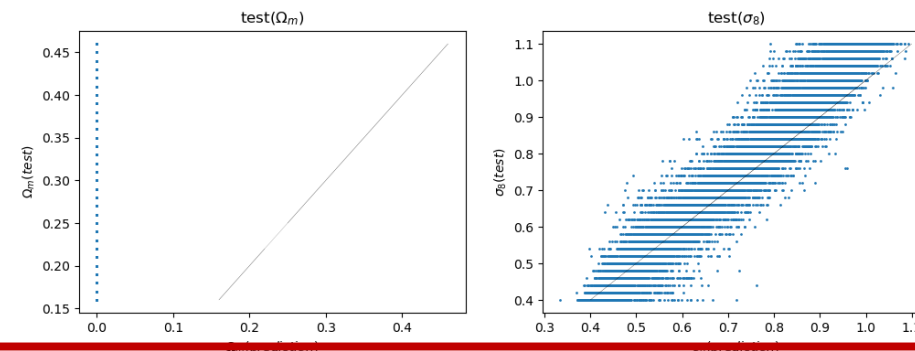
Layer (type)	Output Shape	Param #
=====		
input_1 (InputLayer)	[(None, 32, 32, 32, 1)]	0
batch_normalization (Batch Normalization)	(None, 32, 32, 32, 1)	4
conv3d (Conv3D)	(None, 30, 30, 30, 32)	896
batch_normalization_1 (Batch Normalization)	(None, 30, 30, 30, 32)	128
max_pooling3d (MaxPooling3D)	(None, 15, 15, 15, 32)	0
batch_normalization_2 (Batch Normalization)	(None, 15, 15, 15, 32)	128
conv3d_1 (Conv3D)	(None, 13, 13, 13, 64)	55360
batch_normalization_3 (Batch Normalization)	(None, 13, 13, 13, 64)	256
max_pooling3d_1 (MaxPooling3D)	(None, 6, 6, 6, 64)	0
batch_normalization_4 (Batch Normalization)	(None, 6, 6, 6, 64)	256
conv3d_2 (Conv3D)	(None, 4, 4, 4, 128)	221312
batch_normalization_5 (Batch Normalization)	(None, 4, 4, 4, 128)	512
max_pooling3d_2 (MaxPooling3D)	(None, 2, 2, 2, 128)	0
batch_normalization_6 (Batch Normalization)	(None, 2, 2, 2, 128)	512
flatten (Flatten)	(None, 1024)	0
dropout (Dropout)	(None, 1024)	0
dense (Dense)	(None, 1024)	1049600
dense_1 (Dense)	(None, 256)	262400
dense_2 (Dense)	(None, 2)	514
=====		
Total params: 1,591,878		
Trainable params: 1,590,980		
Non-trainable params: 898		



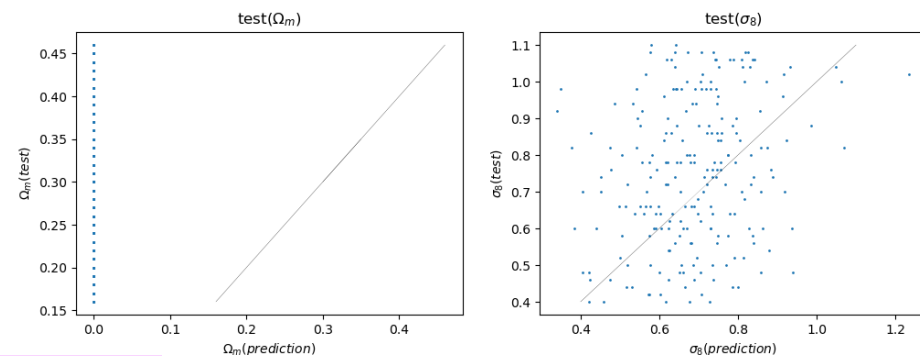
128_short



128_cut



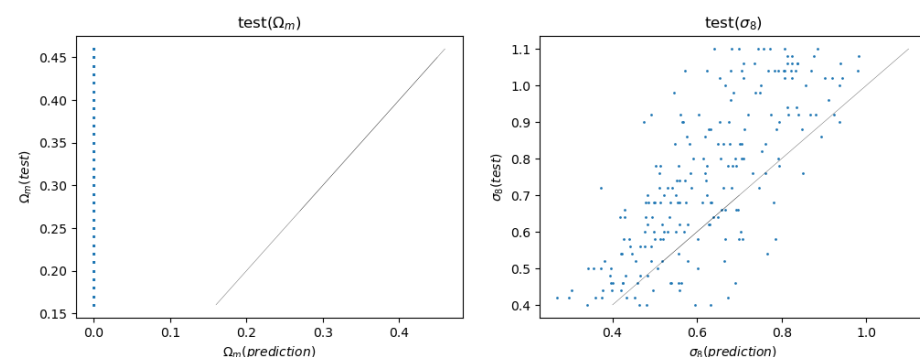
256_short



256_cut



512_short



512_cut



conclusion

- I'm not sure about what is going on..
- There are possibility of my mistakes + possibility of different results each time when I run it even I used same model and same data(I have to check)

Next step

- I'm going to try running with different cosmology + same seed
- Also, I have to understand about mathematics on how each layer works