Data Compression and Inference in Cosmology with Self-Supervised Machine Learning

Aizhan Akhmetzhanova

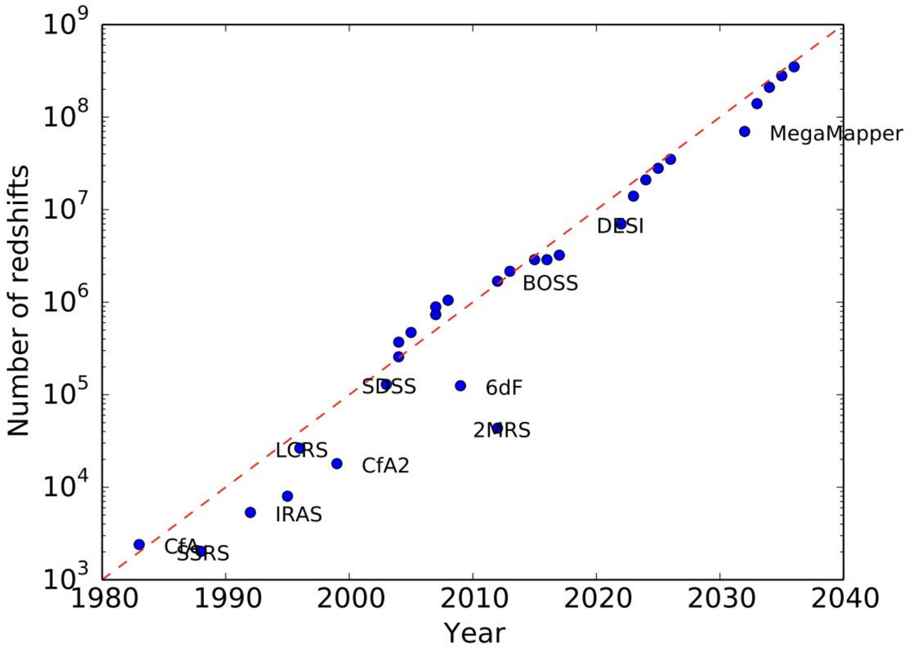
Based on arxiv:2308.09751 with Siddharth Mishra-Sharma and Cora Dvorkin

Outline

- Motivation
- Self-Supervised Learning (SSL)
  - SSL framework
  - Variance-Invariance-Covariance Regularization (VICReg)
- Self-Supervised Learning for Cosmology
  - Data Compression
  - Marginalization over Systematics and Nuisance Parameters
- Conclusions

Motivation

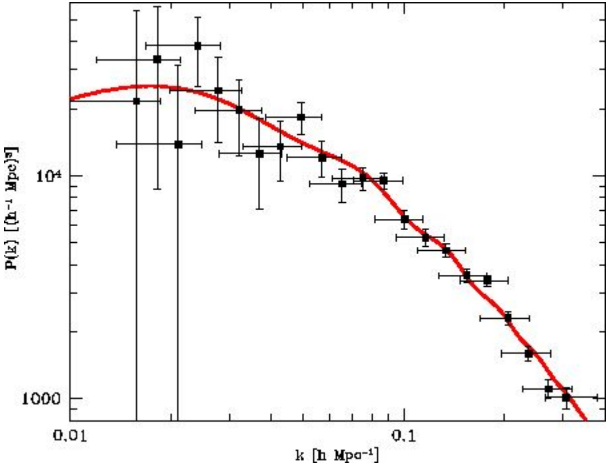
- Current and upcoming surveys (including DESI, Euclid, VRO, SKA) will provide *massive amounts of data* to probe new and existing questions in cosmology
- Making *full use of the data* provided by these surveys is a *challenging task*



(Image credit: Schlegel et al., BAAS (2019))

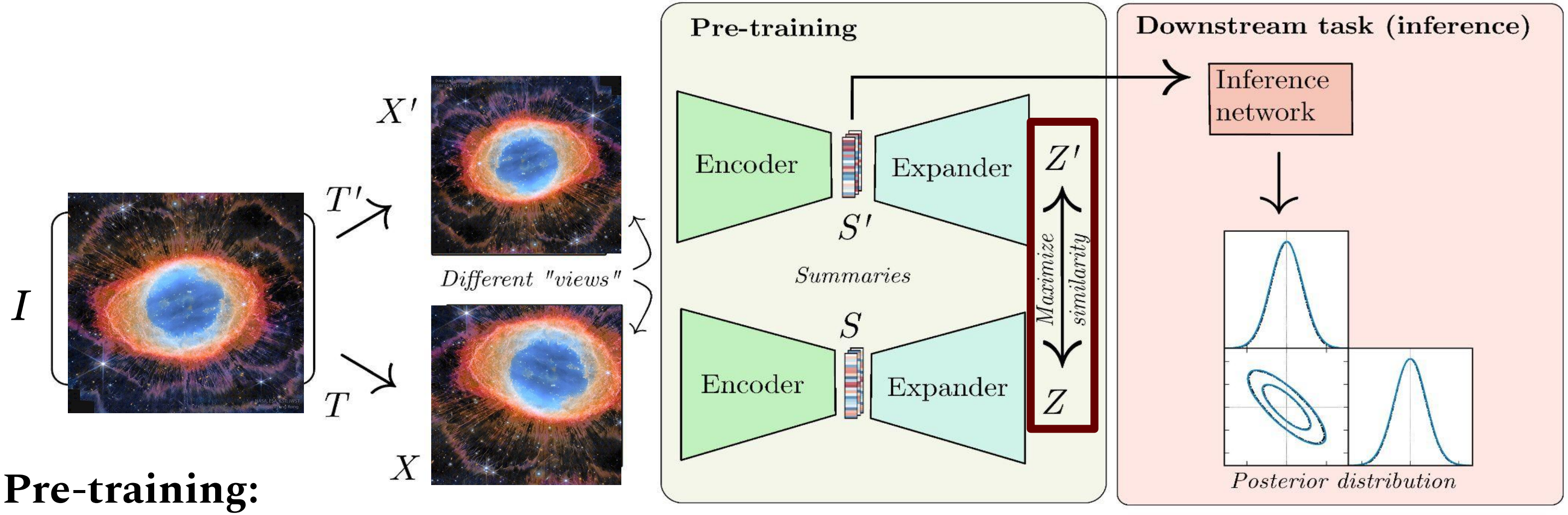
Motivation

- Analyzing the raw data is computationally expensive, so the data is first reduced to a ***set of informative summary statistics***:
    - Power spectrum
    - Bispectrum and higher-order correlation functions
    - Wavelet scattering coefficients
    - Overdensity probability distribution functions
- Potential considerations:
    - Might *not encode all of the physically-relevant information* from the input data
    - Even as a summary statistic, the data vectors might be *very high-dimensional*
- <u>Our approach</u>: Self-Supervised Learning with Physically-Motivated Augmentations



(Image credits:
top: SDSS; bottom: Tegmark et al., ApJ (2003))

Self-Supervised Learning Pipeline



(1) **Pre-training:**

- Given two "views" (augmentations) *X* and *X'* of an input vector *I*, the encoder is trained to produce low-dimensional summaries *S* and *S'* of the input according to some loss function, typically computed on embeddings *Z* and *Z'*.

(2) **Downstream task:**

- The summaries are used directly for downstream tasks (e.g. classification, parameter estimation) by training a simple neural network, such as an MLP with a few layers.

(1) Pre-Training Step:

- Variance-Invariance-Covariance Regularization (VICReg) is a non-contrastive method constructed with a **triple objective function:**

VICReg Loss = Invariance Loss + Variance Loss + Covariance Loss

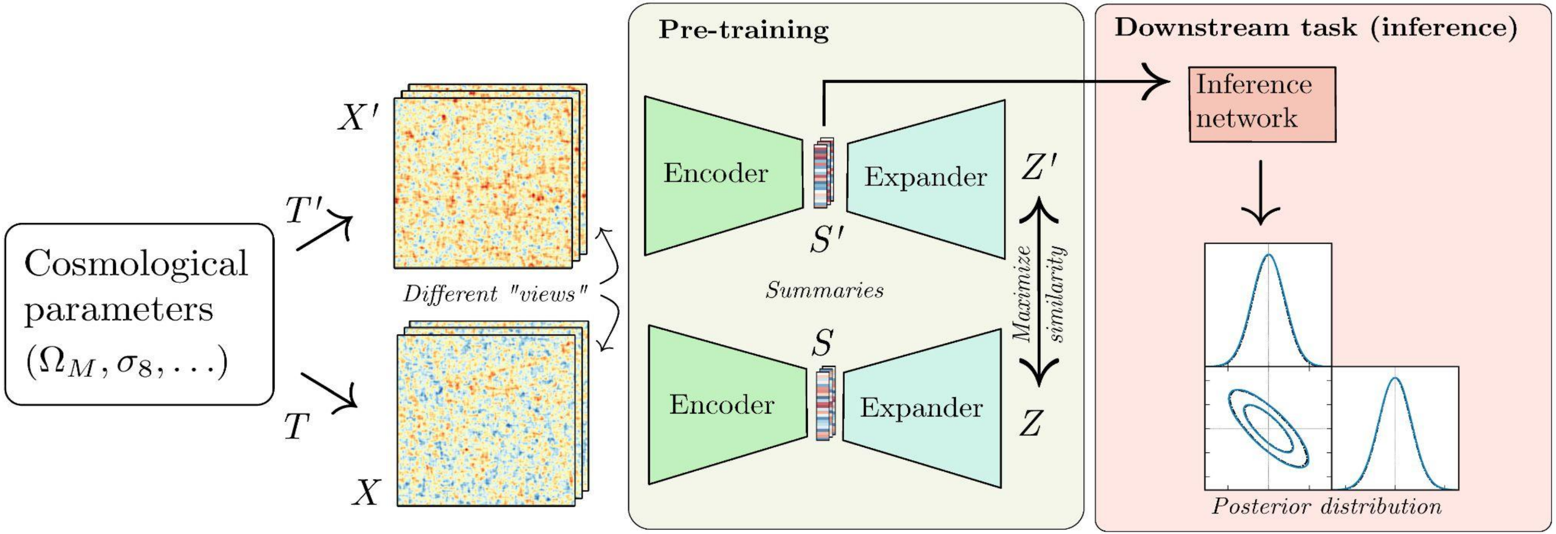
  - *maximizes the similarity* of the summaries corresponding to the same image
  - *minimizes the redundancy between different features* of the summary vectors
  - *maintains variance* between summaries within a training batch to avoid collapse to a trivial solution

(2) Downstream Task:

- Cosmological parameter inference
- Train an *inference* network to infer cosmological parameters of interest (means **θ** and covariance **Σ**) by minimizing the negative log-likelihood function:

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} \left[\frac{1}{2} \ln |\Sigma_n| + \frac{1}{2} (\theta_n - \mu_n)^T \Sigma_n^{-1} (\theta_n - \mu_n) \right]$$

This Work



- Using physically-motivated augmentations = augmentations that correspond to the same underlying physics of interest
- SSL Applications:
  - Data compression
  - Marginalization over systematics and nuisance parameters
  - Parameter inference with sequential simulation-based inference (in the paper)

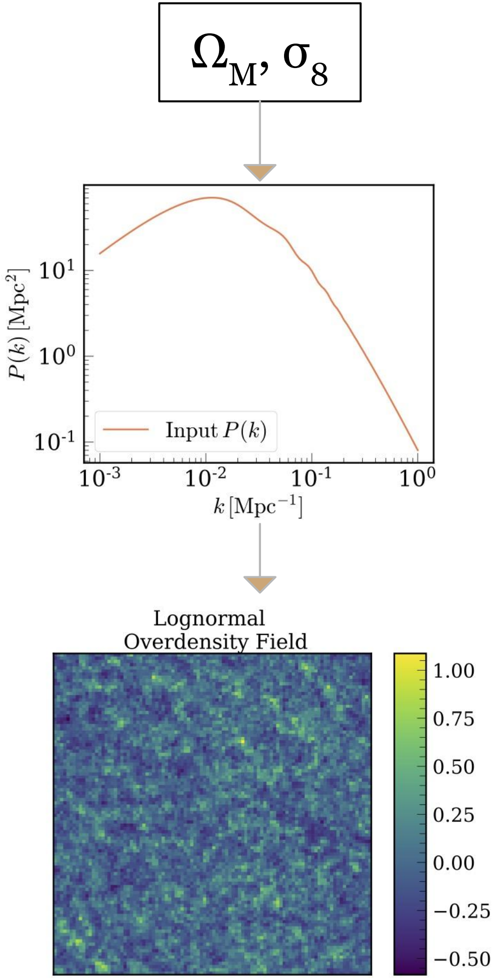
Self-Supervised Learning for Data Compression and Parameter Inference

Lognormal Fields

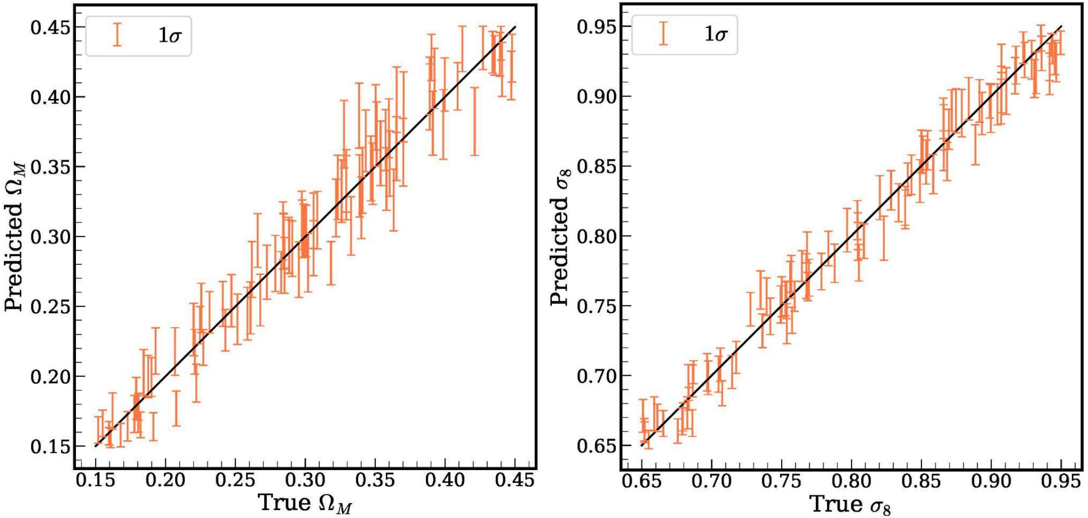
- We generate lognormal overdensity fields δ_{LN}
- 10,000 different cosmologies:
  - Vary $\mathbf{\Omega_M} \in [0.15, 0.45]$ and $\boldsymbol{\sigma_8} \in [0.65, 0.95]$
  - Remaining cosmological parameters are fixed
- Simulated field is a grid of N^2=100x100 points with area L^2=(1000 Mpc)^2

VICReg Setup

- **Augmentations/views**: different realizations of the same input cosmology with different initial conditions, rotated and flipped at random
- **Encoder network**: Compresses 100x100 maps to summaries of dim=16
- **Inference network**: Predicts *means* for Ω_M, σ_8 *and covariance matrix* Σ



Assessing the summaries: Inference on VICReg Summaries



- **The inference network** trained on the summaries is able to recover the true values of cosmological parameters with both accuracy and precision, with **relative errors on Ω_M and σ_8 equal to 5.2% and 1.3%**, respectively.
  - Similar errors from **the supervised baseline model: 5.1% and 1.3%** respectively

Assessing the summaries: Comparison to Fisher Constraints

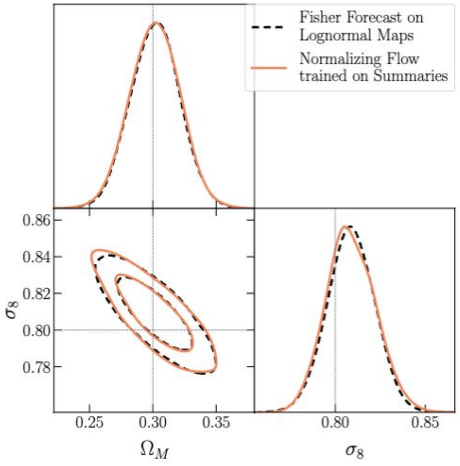
- Fisher information matrix for the lognormal maps:

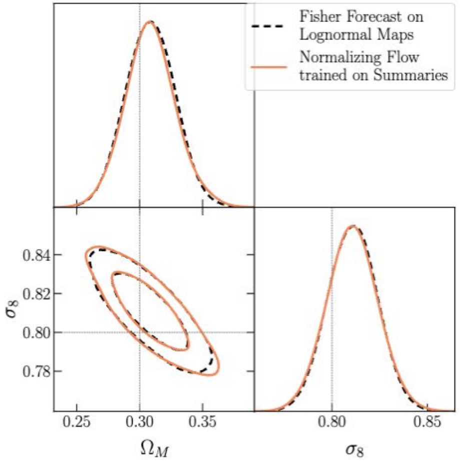
$$F_{\alpha\beta} = \frac{1}{2}\sum_k \frac{\partial P_G(k)}{\partial \theta_\alpha}\frac{\partial P_G(k)}{\partial \theta_\beta}\frac{1}{P_G(k)^2}$$

- Cramer-Rao bound:

$$\sigma_\alpha \geq [F^{-1/2}]_{\alpha\alpha}$$

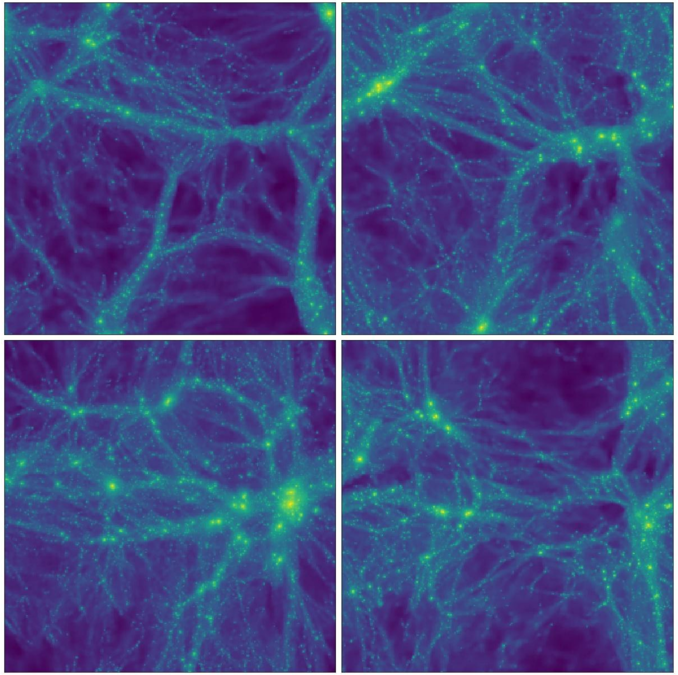
- We train a normalizing flow to estimate the posterior distribution of the parameters, given a VICReg summary of a corresponding lognormal field.
- For a fiducial cosmology with Ω_M=0.3 and σ_8=0.8, the Fisher constraints and posteriors from the normalizing flow show great agreement





CAMELS Total Matter Density Fields

- Two *hydrodynamic suites of simulations*, IllustrisTNG and SIMBA, from the CAMELS project (Villaescusa-Navarro et al., ApJ (2021))
  - Each simulation suite implements *distinct galaxy formation model*
- Total matter density maps represent spatial distribution of baryonic and dark matter at z=0
- 1,000 different cosmologies in each suite:
  - Cosmological parameters: $\Omega_M \in [0.1, 0.5]$ and $\sigma_8 \in [0.6, 1.0]$
  - Astrophysical parameters:
    - Stellar feedback parameters A_{SN1}, A_{SN2}
    - AGN feedback parameters A_{AGN1}, A_{AGN2}

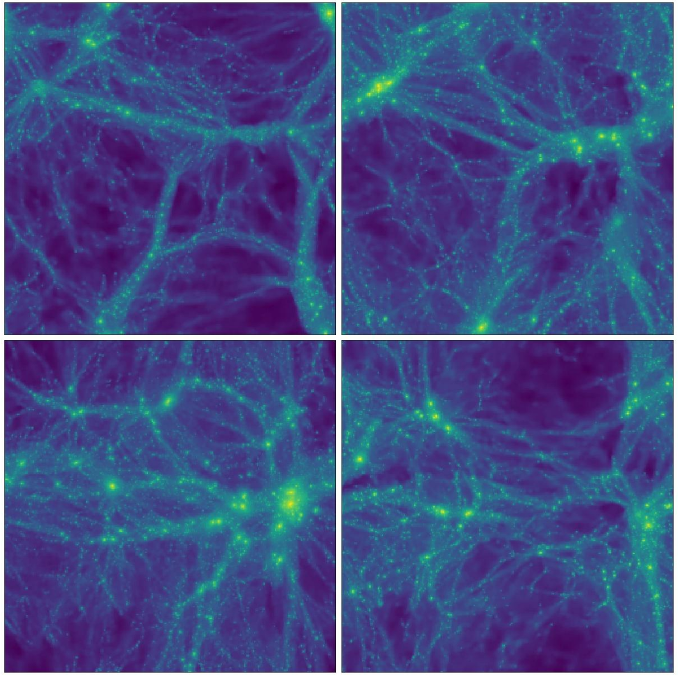


CAMELS: VICReg Setup

- **Augmentations/views**: different spatial slices of the simulation boxes, rotated and flipped at random
- Due to the complexity of the maps, we **modify the loss function** to include 5 pairs of different augmentations from each cosmology to allow the network to learn from more variations:

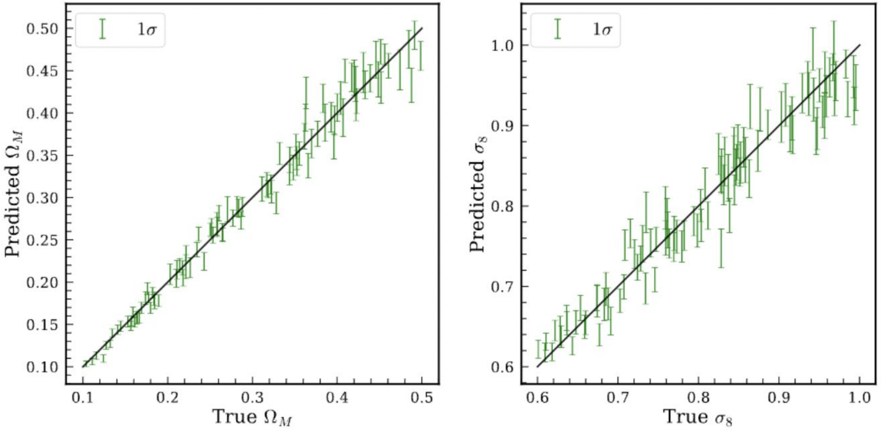
$$\mathcal{L} = \frac{1}{N_{\text{pairs}}} \sum_{i}^{N_{\text{pairs}}} \mathcal{L}_{\text{VICReg,i}}$$

- **Encoder network**:
  - Compresses 256x256 maps to summaries of dim=128
- **Inference network**:
  - Predicts *means and covariance matrix* for cosmological parameters Ω_M and σ_8 from the summaries

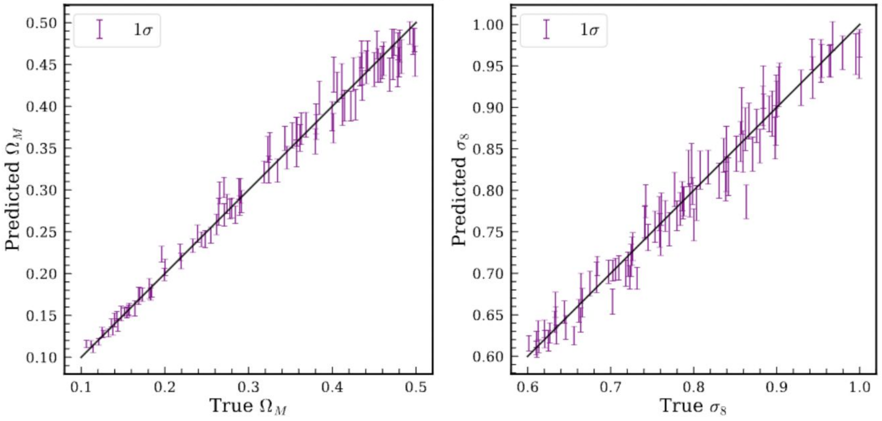


Assessing the summaries: Inference on the VICReg Summaries

- Despite considerable reduction in the dimensionality of the data, the VICReg model still able to infer cosmological parameters for Ω_M and σ_8 with percent-level accuracy:
  - *SIMBA* suite: 3.8% and 2.5%
  - *IllustrisTNG* suite: 3.7% and 1.9%
- Slightly lower errors from the baseline supervised model:
  - *SIMBA* suite: 3.3% and 2.3%
  - *IllustrisTNG* suite: 3.3% and 1.8%



(a) VICReg: SIMBA.

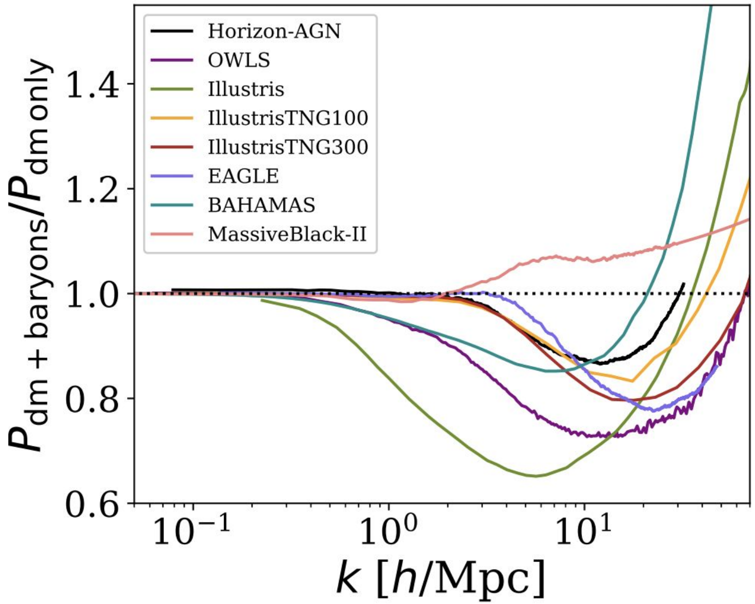


(b) VICReg: IllustrisTNG.

Self-Supervised Learning for Marginalization Over Systematics and Nuisance Parameters

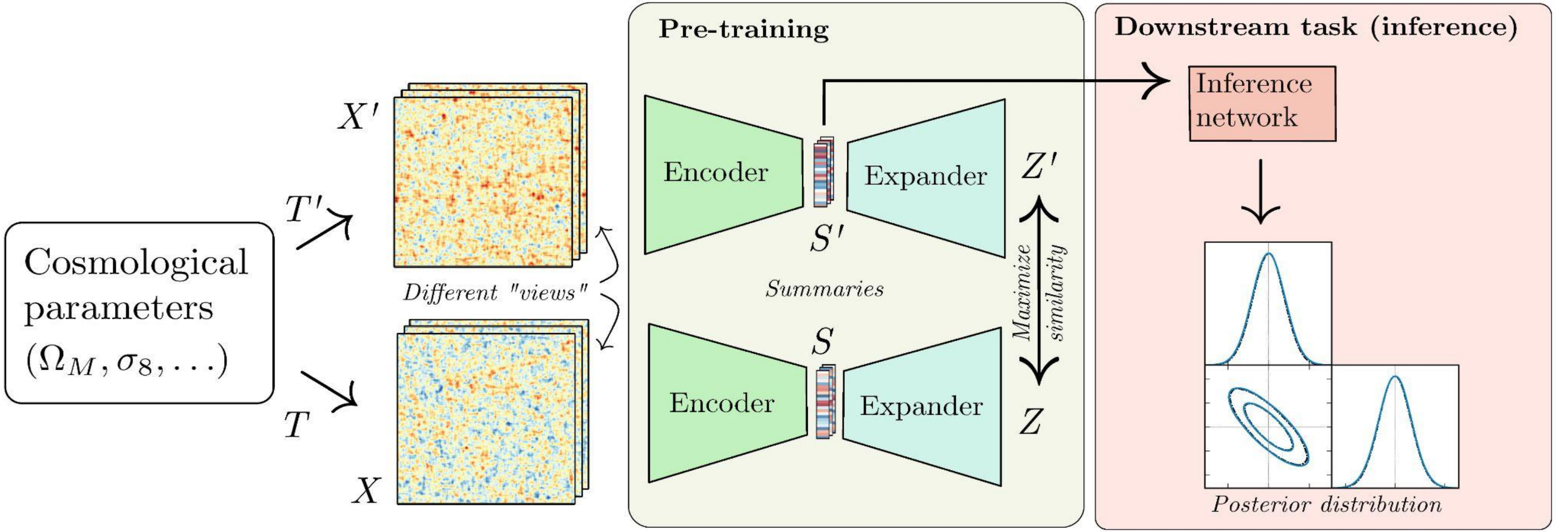
Baryonic Effects in Cosmology

- Baryonic effects modify total matter distribution on small scales:
    - AGN feedback, SNe feedback, Star formation
- These effects are, in general, *complex and poorly understood*:
    - Some of these effects cannot be resolved in simulations → different prescriptions ("sub-grid" models) for these processes
    - Different hydrodynamical simulations have *different predictions* on the resulting *modifications to matter power spectrum* $P_M(k)$
- Particularly important for future-generation weak-lensing surveys like VRO, Euclid, Roman which require accurate theoretical modelling of $P_M(k)$



(Image credit: Chisari et al. 2019)

Baryonic Effects as an Augmentation



- It would be interesting to use ***different implementations of baryonic effects in hydrodynamical simulations*** (e.g. SIMBA, IllustrisTNG) as ***different augmentations*** of the **same cosmology** (with the same initial conditions)
- Such a dataset is *unavailable* at present → We use a simple proof-of-principle example instead

Based on a model from
Villaescusa-Navarro et al., ApJ (2022)

Toy $P(k)$ model

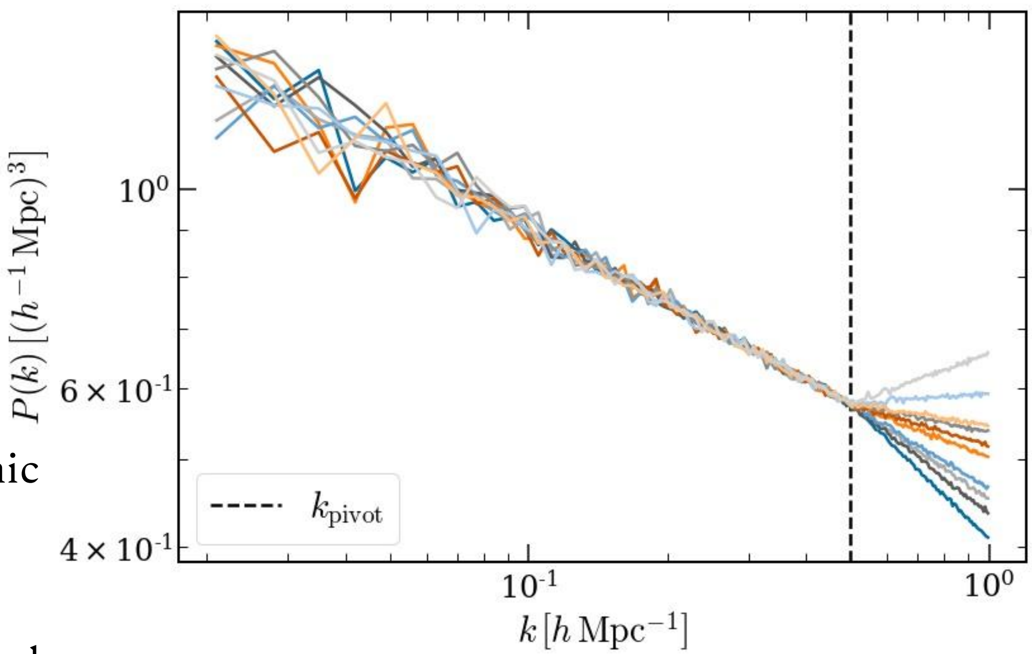
$$P(k) = \begin{cases} Ak^B & k \leq k_{\text{pivot}} \\ Ck^D & k > k_{\text{pivot}} \end{cases}$$

- ***A***, ***B***: "cosmological" parameters, ***D***: "baryonic physics" parameter
- Change in slope on small scales ($k > k_{pivot} = 0.5$ h/Mpc) represents the effects of "baryonic physics"
  - Treat different realizations of "baryonic physics" as a possible augmentation
- Small scales still contain "cosmological" information via ***C***
- No noise modelling, but we account for cosmic variance effects via

$$P_{\text{obs}}(k) \sim \mathcal{N}\left(P(k), \sigma_k^2\right)$$

Dataset: Broken Power Law with varying $\boldsymbol{D}$

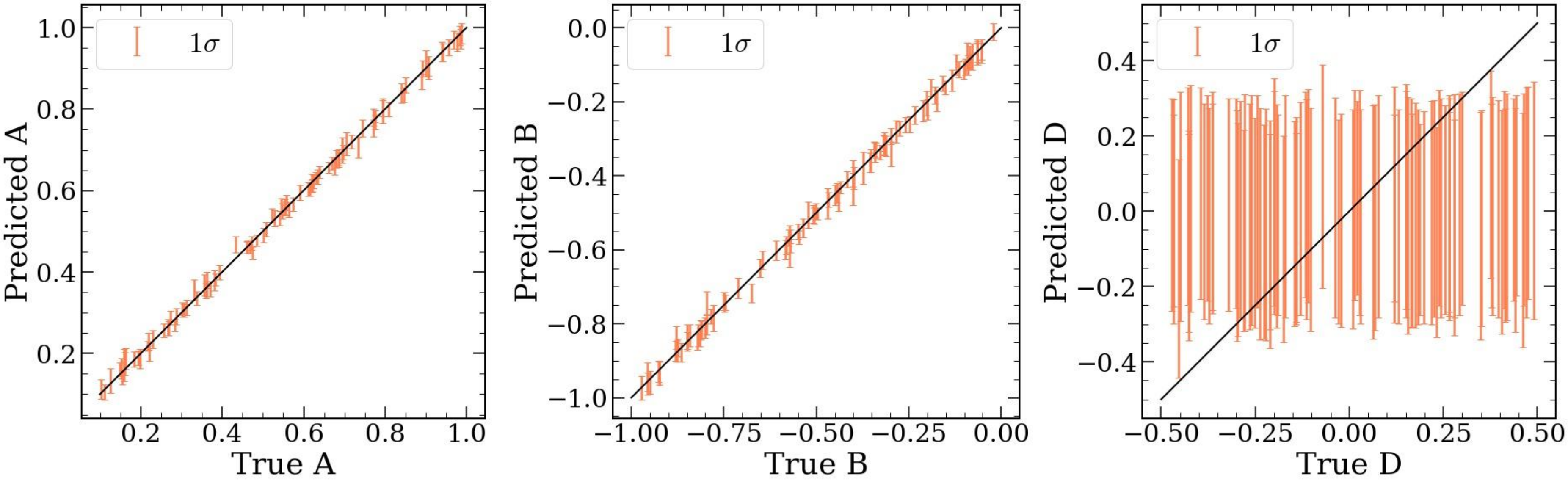
- 1,000 different cosmologies (with different values of A, B)
  - $\boldsymbol{A} \in [0.1, 1.0]$
  - $\boldsymbol{B} \in [-1., 0.0]$
  - $\boldsymbol{D} \in [-0.5, 0.5]$
- $k \in [0.021, 0.994]$ h/Mpc
- $k_{pivot} = 0.5$ h/Mpc



VICReg Setup

- **Augmentations:** variations in baryonic effects (different values of $\boldsymbol{D}$)
- **Encoder network** compresses *P(k)* from dim=140 to dim=32
- **Inference network** predicts means and covariance matrix for $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{D}$

Parameter Inference: Broken Power Law with varying *D*

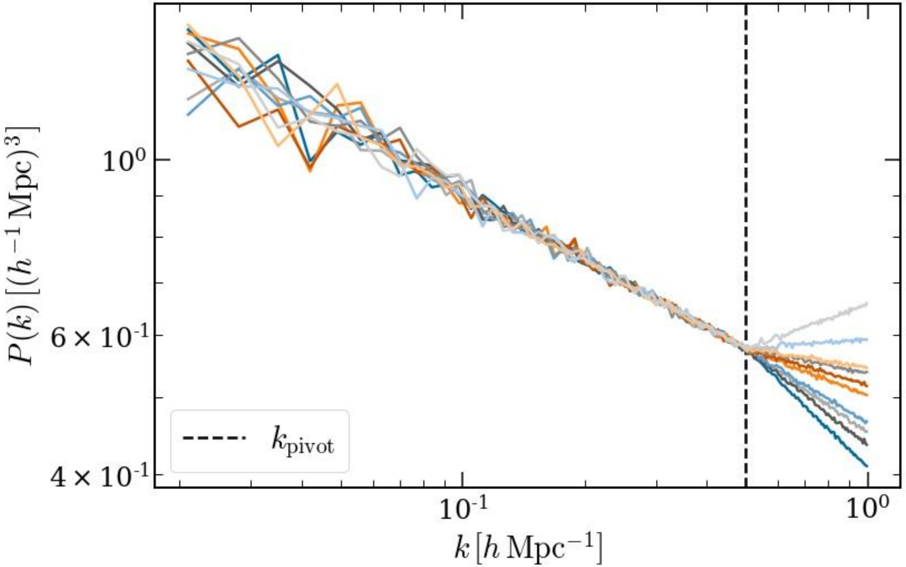


Analyzing the Summaries

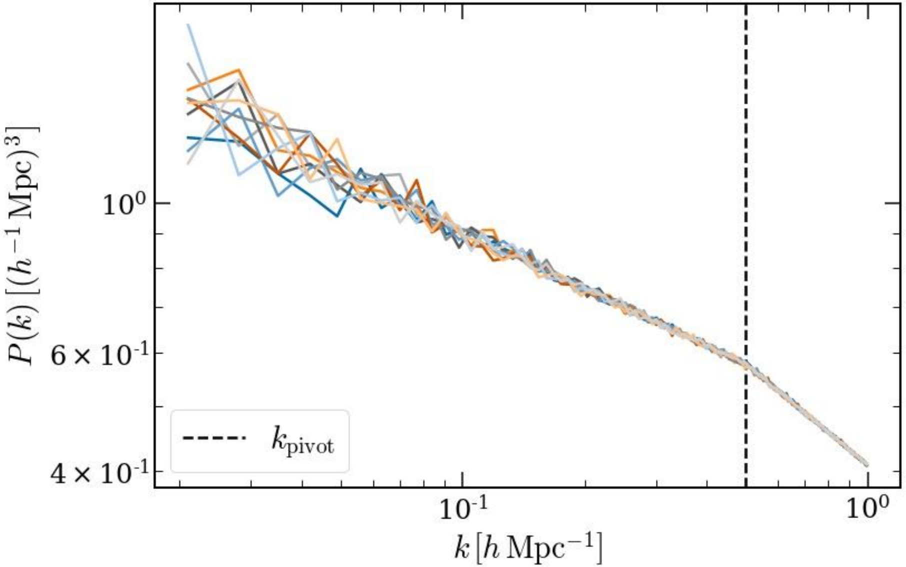
- Small scales still encode information about cosmological parameters ***A***, ***B*** via ***C***
- Do the summaries ignore the small scales? Or do they still use the information from them to infer cosmological parameters?
- How do the summaries *S* depend on the values of *P(k)* in different *k*-bins?
  - Distance Correlation
    - captures both linear and non-linear dependence between random variables
  - Mutual Information
    - quantifies how much information one gains about a random variable X by observing another random variable Y

Two Datasets for Comparison

Broken Power Law (BPL) with varying ***D***
(cosmic variance and baryonic effects)

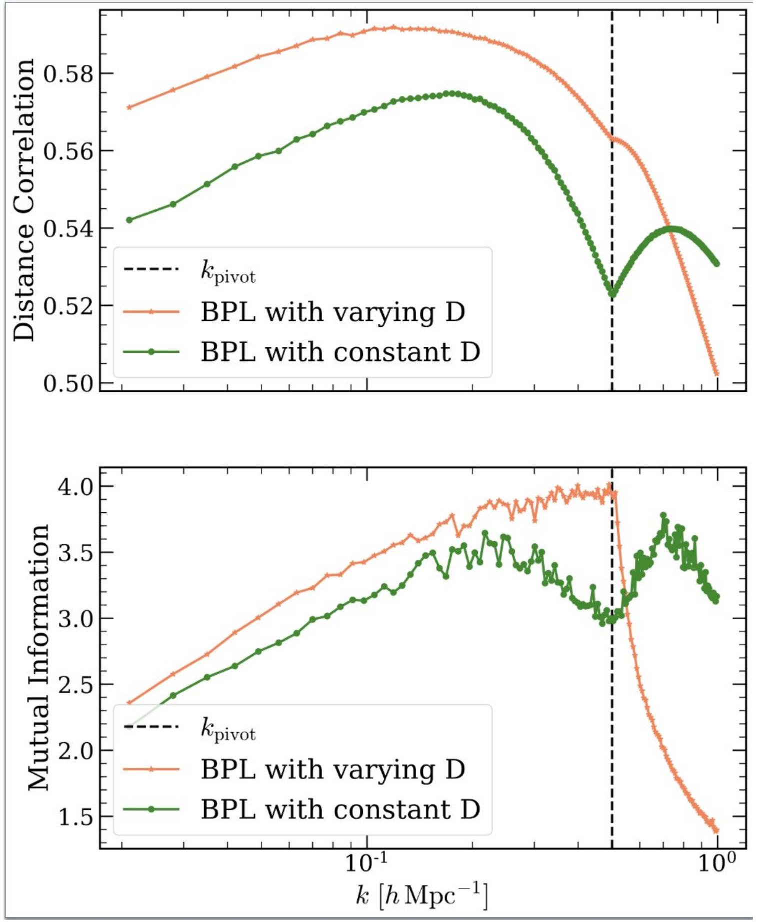


Broken Power Law (BPL) with constant ***D***
(cosmic variance only)



Distance Correlation and Mutual Information

- Both metrics follow similar trends
- Similar behaviour for the two datasets up to the pivot scale k_{pivot}
  - On these scales, P(k) contains information only about `cosmological' parameters
- Past the pivot scale, we also get information about `baryonic' parameters:
  - For ***BPL w/ varying D***, `baryonic' parameters are ***not of interest*** → dCorr and MI decrease
  - For ***BPL w/ constant D***, `baryonic' parameters are ***relevant*** → dCorr and MI increase again



Conclusions

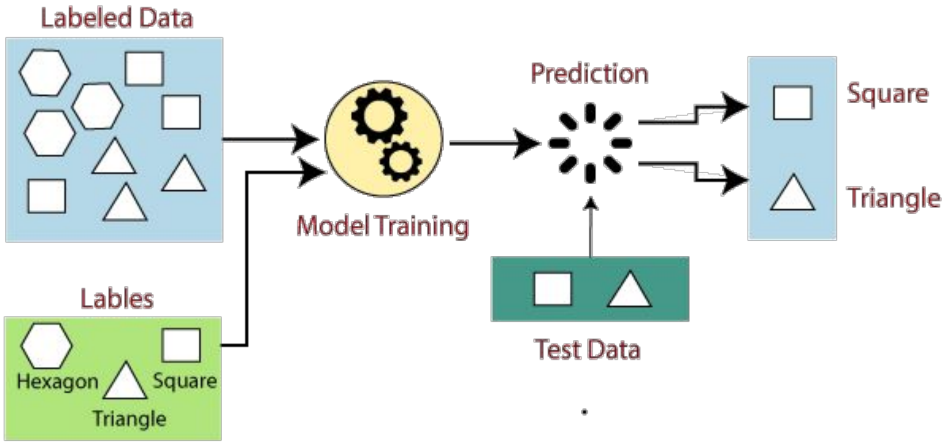
- We have explored an SSL approach for constructing compact informative summary statistics and extended it by including augmentations that correspond to the same underlying physics of interest
- We demonstrated the applications of the method and its potential in cosmological context:
  - Data Compression
  - Marginalization over Nuisance Parameters and Systematics
  - Simulation-Based Inference (in the paper)
- Additional follow-up studies are necessary before deploying self-supervised learning methods on real cosmological data:
  - Complexifying the models:
    - Applying the self-supervised learning framework to cosmological power spectra (or other observables) with more realistic modelling of baryonic feedback (e.g. HMcode)
  - Finding a more principled way to decide on the optimal size of the summary vectors
  - Determining new ways to assess how informative and unbiased the summaries are

Thank you!
Questions?

Bonus/Back-up Slides

Supervised Learning

- In the supervised learning framework, a neural network model is trained to perform a *specific task* based on a *dataset with associated labels*
- Downsides:
  - *Limited by the availability* of quality *labeled datasets*
  - *New downstream tasks* usually *require new models* to be trained from scratch



(Image credits: javaTpoint)

Self-Supervised Learning

SSL framework combines unsupervised and supervised learning:

(1) Learn to **construct meaningful (lower-dimensional) summaries** (or representations) of data **from an unlabeled dataset**

(2) **Use the learnt summaries** for a **downstream supervised task** of interest

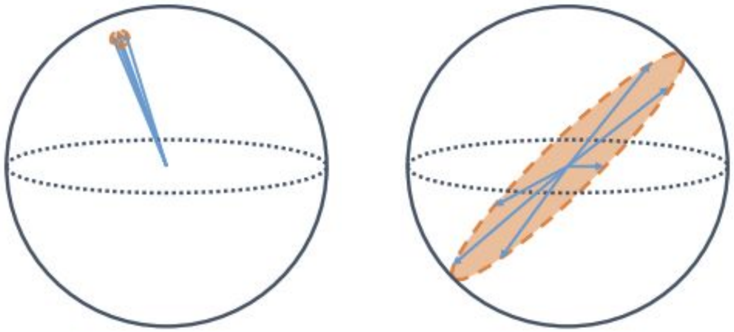
- Advantages over supervised learning (SL) framework:
  - Can make *use* of both *vast unlabeled datasets* and *smaller labeled datasets*
  - Summaries can *be used for a range of downstream tasks* (as opposed to a specific predetermined task in supervised learning)

Self-Supervised Learning in Astrophysics and Cosmology

- Galaxy morphology classification:
  - Classifying SDSS galaxy images (Hayat et al., ApJL (2020))
  - Radio galaxy classification using data from FIRST survey (Slijpcevic et al., MNRAS (2020))
- Self-similarity search and anomaly detection:
  - Building self-similarity search tools for galaxy images from DES (Stein et al., NeurIPS 2021)
  - Detecting galaxies with tidal features using HSC images (Desmons et al., ICML 2023)
- Neural posterior estimation:
  - Estimating black hole merger parameters from the gravitational waves (Shen et al., Mach. Learn.: Sci. Technol. (2021))

Collapse in Self-Supervised Learning

- **Norm collapse**:
  - Key challenge in implementing self-supervised learning methods
  - The encoder learns a ***trivial solution***: maps different input vectors to the same summaries
- **Dimensional collapse**:
  - *Different dimensions* of the summaries are *redundant* (encode the similar information)
  - Might lead to poorer performance as the network is not using its full capacity
- Approaches to the collapse problems:
  - Contrastive methods:
    - Distinguish between *negative* and *positive samples*:
      - Push positives closer together and negatives further apart in the embedding space
  - Non-Contrastive methods:
    - Employ various *regularization methods* to prevent collapse
    - Examples: **VICReg**



(b) complete collapse (c) dimensional collapse

(Image credit: Jing et al., ICLR 2022)

- Let $\boldsymbol{Z} = [Z_1, \dots, Z_n]$, and $\boldsymbol{Z'} = [Z'_1, \dots, Z'_n]$ be two batches of $\boldsymbol{n}$ embeddings
- Each embedding $\boldsymbol{Z_i}$ is a $\boldsymbol{d}$-dimensional vector
- $\boldsymbol{Z_i}$ and $\boldsymbol{Z'_i}$ are embeddings of the two transformed views of the same image

$$\text{VICReg Loss} = \boxed{\text{Invariance Loss}} + \text{Variance Loss} + \text{Covariance Loss}$$

$$\boxed{s(Z, Z') = \frac{1}{n} \sum_{i=1}^{n} \|Z_i - Z'_i\|_2^2}$$

- The invariance component *s(Z, Z')* *<u>measures the similarity</u>* between the outputs of the encoder *Z, Z'* corresponding to the *same image*

- Let $\boldsymbol{Z} = [Z_1, \ldots, Z_n]$, and $\boldsymbol{Z'} = [Z'_1, \ldots, Z'_n]$ be two batches of $\boldsymbol{n}$ embeddings
- Each embedding $\boldsymbol{Z_i}$ is a $\boldsymbol{d}$-dimensional vector
- $\boldsymbol{Z_i}$ and $\boldsymbol{Z'_i}$ are embeddings of the two transformed views of the same image

VICReg Loss = Invariance Loss + $\boxed{\text{Variance Loss}}$ + Covariance Loss

$$\boxed{v(Z) = \frac{1}{d} \sum_{j=1}^{d} \max\left(0, \gamma - S\left(Z^j, \epsilon\right)\right)}$$

where

- Z^j is a vector that consists of the values of the embeddings $\boldsymbol{Z}$ at $\boldsymbol{j}$-th dimension
- $S(x, \epsilon) = \sqrt{\mathrm{Var}(x) + \epsilon}$
- γ, ε are hyperparameters

- The variance $v(Z, Z')$ component is intended to <u>avoid the *norm collapse*</u>
- Measures <u>*the overall variance in a given batch*</u> across d different dimensions in the embedding space
- Encourages the variance along each dimension to be *close to some constant* γ

- Let $\boldsymbol{Z} = [Z_1, \ldots, Z_n]$, and $\boldsymbol{Z'} = [Z'_1, \ldots, Z'_n]$ be two batches of $\boldsymbol{n}$ embeddings
- Each embedding $\boldsymbol{Z_i}$ is a $\boldsymbol{d}$-dimensional vector
- $\boldsymbol{Z_i}$ and $\boldsymbol{Z'_i}$ are embeddings of the two transformed views of the same image

VICReg Loss = Invariance Loss + Variance Loss + Covariance Loss

- The covariance *c(Z)* component is intended to <u>avoid the *dimensional collapse*</u>
- <u>*Decorrelates different features*</u> of the summaries
- Drives the covariance matrix **C(Z)** to be close to a *diagonal matrix*

$$c(Z) = \frac{1}{d} \sum_{k \neq l} [\mathbb{C}(Z)]^2_{k,l}$$

$$\mathbb{C}(Z) = \frac{1}{n-1} \sum_{i=1}^{n} (Z_i - \bar{Z})(Z_i - \bar{Z})^T, \text{ where } \bar{Z} = \frac{1}{n} \sum_{i=1}^{n} Z_i$$

VICReg Loss = Invariance Loss + Variance Loss + Covariance Loss

$$= \lambda s(Z, Z') + \mu[v(Z) + v(Z')] + \eta[c(Z) + c(Z')]$$

$$s(Z, Z') = \frac{1}{n}\sum_{i=1}^{n} \|Z_i - Z'_i\|_2^2$$

$$v(Z) = \frac{1}{d}\sum_{j=1}^{d} \max\left(0, \gamma - S\left(Z^j, \epsilon\right)\right)$$

$$c(Z) = \frac{1}{d}\sum_{k \neq l} [\mathbb{C}(Z)]_{k,l}^2$$

λ, μ, η: hyperparameters controlling the weights of the terms

Self-Supervised Learning for Data Compression

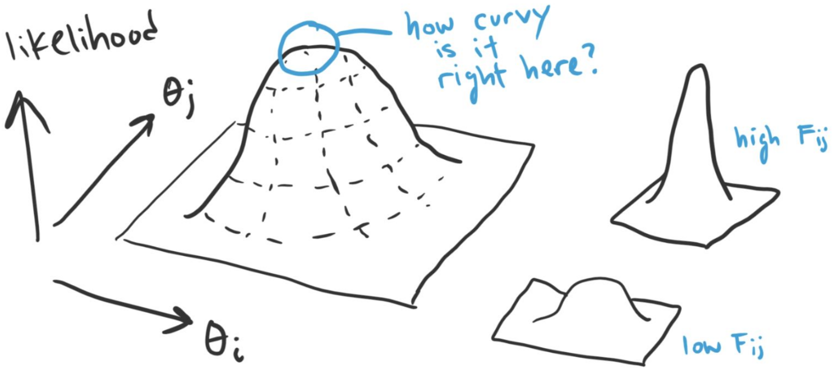
Assessing the summaries: Fisher Information

- Fisher information $\boldsymbol{F_{\alpha\beta}(\theta)}$ is a way of measuring of the amount of information a data vector $\boldsymbol{d}$ carries about parameters $\boldsymbol{\theta}$.
- $\boldsymbol{F_{\alpha\beta}(\theta)}$ can be computed as the variance of the score of the likelihood at fiducial parameters $\boldsymbol{\theta_{fid}}$:

$$\mathbf{F}_{\alpha\beta}(\boldsymbol{\vartheta}) = \left\langle \frac{\partial \ln \mathcal{L}(\mathbf{d}|\boldsymbol{\vartheta})}{\partial \vartheta_\alpha} \frac{\partial \ln \mathcal{L}(\mathbf{d}|\boldsymbol{\vartheta})}{\partial \vartheta_\beta} \right\rangle \Bigg|_{\boldsymbol{\vartheta}=\boldsymbol{\vartheta}^{\mathrm{fid}}}$$

$$= -\left\langle \frac{\partial^2 \ln \mathcal{L}(\mathbf{d}|\boldsymbol{\vartheta})}{\partial \vartheta_\alpha \partial \vartheta_\beta} \right\rangle \Bigg|_{\boldsymbol{\vartheta}=\boldsymbol{\vartheta}^{\mathrm{fid}}}$$

- Cramer-Rao bound: The inverse of the Fisher matrix is the lower bound on variance of any unbiased estimator of $\boldsymbol{\theta}$: $\boldsymbol{\sigma_\theta \geq [F^{-1/2}]_{\theta\theta}}$



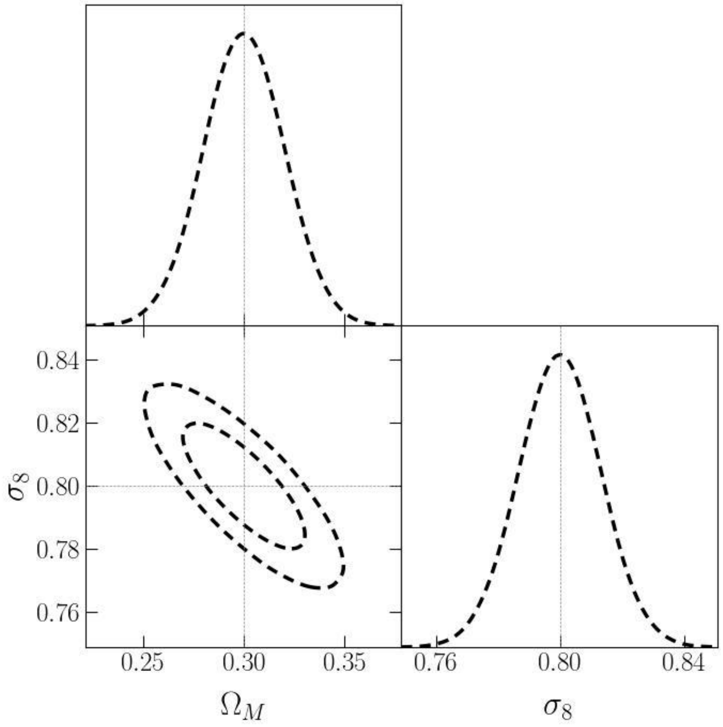
(Image credit:Zack Li)

Assessing the summaries: Fisher Constraints for Lognormal Fields

Computing Fisher matrix from lognormal fields:

- We expect the lognormal fields to preserve the Fisher information content of underlying Gaussian fields.
- We estimate the Fisher information matrix for lognormal fields by computing the Fisher information matrix for the associated Gaussian fields (with power spectrum $P_G(k)$):

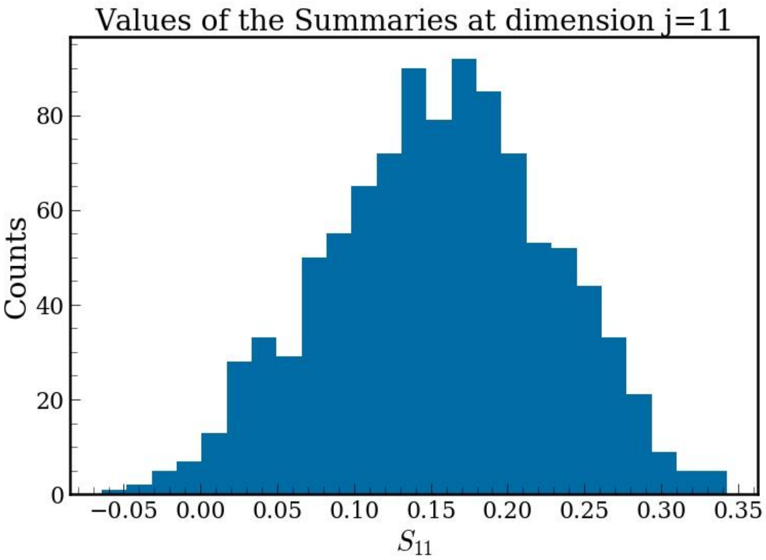
$$F_{\alpha\beta} = \frac{1}{2}\sum_k \frac{\partial P_G(k)}{\partial \theta_\alpha}\frac{\partial P_G(k)}{\partial \theta_\beta}\frac{1}{P_G(k)^2}$$

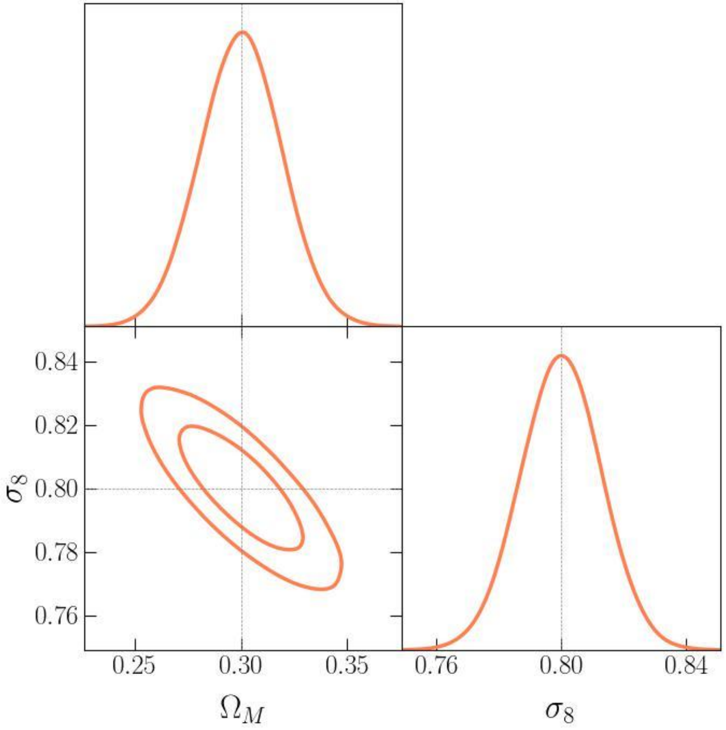


Assessing the summaries: Fisher Constraints for the VICReg Summaries

Assuming Gaussian likelihood, the Fisher matrix elements can be computed as:

$$F_{\alpha\beta} = \frac{\partial S}{\partial \theta_\alpha} C^{-1} \frac{\partial S}{\partial \theta_\beta}$$





Assessing the summaries: Comparison of Fisher Constraints

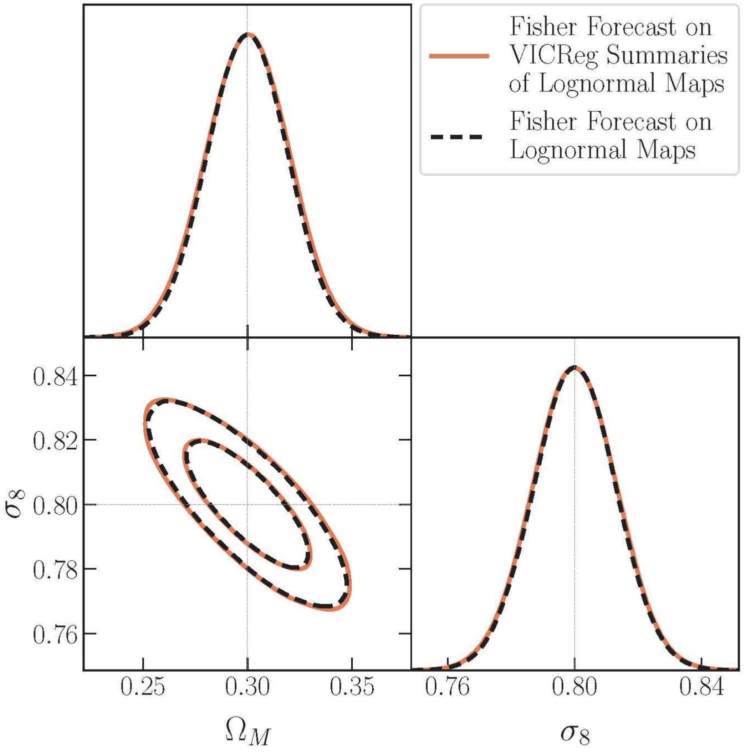
- Fisher information matrix for the lognormal maps:

$$F_{\alpha\beta} = \frac{1}{2}\sum_k \frac{\partial P_G(k)}{\partial\theta_\alpha}\frac{\partial P_G(k)}{\partial\theta_\beta}\frac{1}{P_G(k)^2}$$

- Fisher information matrix for the summaries of the lognormal maps:

$$F_{\alpha\beta} = \frac{\partial S}{\partial\theta_\alpha} C^{-1} \frac{\partial S}{\partial\theta_\beta}$$

- Cramer-Rao bound: $\sigma_\alpha \geq [F^{-1/2}]_{\alpha\alpha}$
- For a fiducial cosmology with Ω_M=0.3 and σ_8=0.8, we find good agreement between Fisher constraints on the cosmological parameters



Assessing the summaries: Comparison of Fisher Constraints

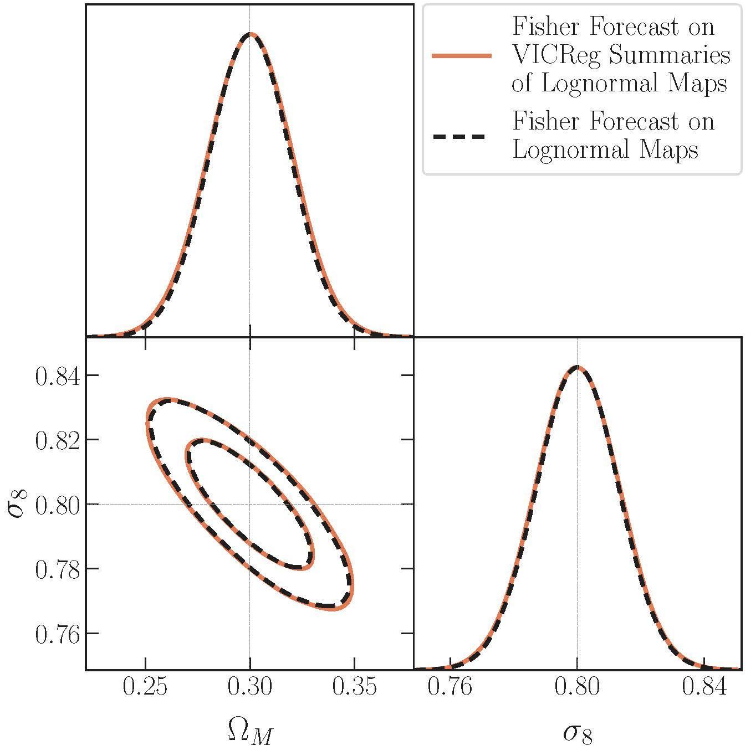
- We expect the lognormal fields to preserve the information content of underlying Gaussian fields
- Fisher information matrix for Gaussian fields:

$$F_{\alpha\beta} = \frac{1}{2}\sum_k \frac{\partial P_G(k)}{\partial\theta_\alpha}\frac{\partial P_G(k)}{\partial\theta_\beta}\frac{1}{P_G(k)^2}$$

- Fisher information matrix for the summaries (assuming Gaussian likelihood):

$$F_{\alpha\beta} = \frac{\partial S}{\partial\theta_\alpha}C^{-1}\frac{\partial S}{\partial\theta_\beta}$$

- Cramer-Rao bound: $\sigma_\alpha \geq [F^{-1/2}]_{\alpha\alpha}$.
- For a fiducial cosmology with Ω_M=0.3 and σ_8=0.8, we find excellent agreement between Fisher constraints on the cosmological parameters



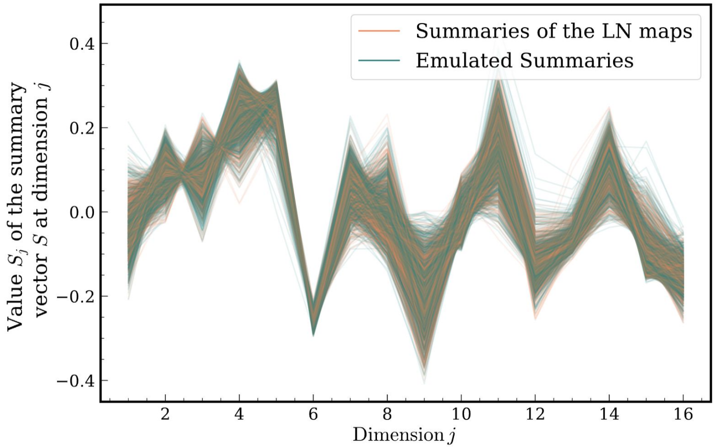
Self-Supervised Learning for Parameter Inference with Sequential Simulation-Based Inference (SBI)

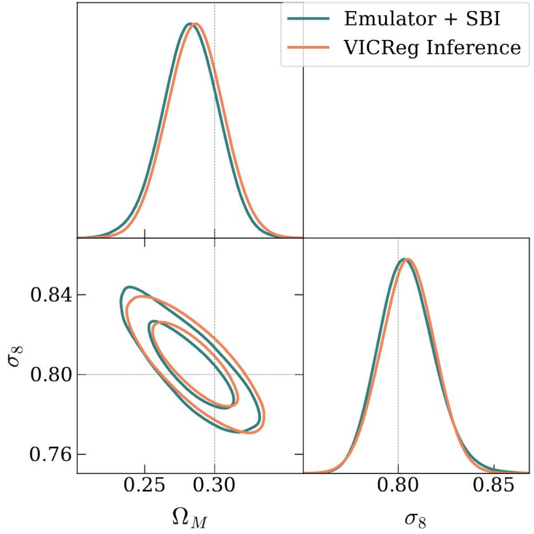
Simulation-Based Inference

- **SBI** is a broad set of methods designed to infer parameters of interest $\boldsymbol{\theta}$ when the likelihood $\mathbf{p}(\boldsymbol{x}_{obs}|\boldsymbol{\theta})$ describing the observed data $\boldsymbol{x}_{obs}$ is unknown or intractable
  - Rely on forward models (simulators) which implicitly define the likelihood
  - Neural SBI methods enable efficient and accurate posterior inference, even for complex high-dimensional distributions
  - ***Bottleneck***: computational complexity of the simulator
- One potential application of the self-supervised compression scheme is using it to <u>build an emulator of summaries</u> to address the computational bottleneck:
  - (1) Train an emulator on the summaries $\boldsymbol{S}$
    - Should be easier and faster than training an emulator to produce the uncompressed data (e.g. maps) due to lower-dimensionality of $\boldsymbol{S}$
  - (2) Use the emulator as the forward model in the inference process

Emulated Summaries

- **Data**: lognormal maps
- **Emulator**: a stack of masked autoregressive flows
- **Observed data:** a random realization of a lognormal maps with Ω_M=0.3 and σ_8=0.8
- **Training settings:**
  - Sequential Neural Posterior Estimation (SNPE)
  - 10 rounds of inference with 1000 simulations per round
- **Posteriors:**
  - The constraints obtained using the inference network are consistent with the SNPE-informed constraints, with true values well within the posterior contours

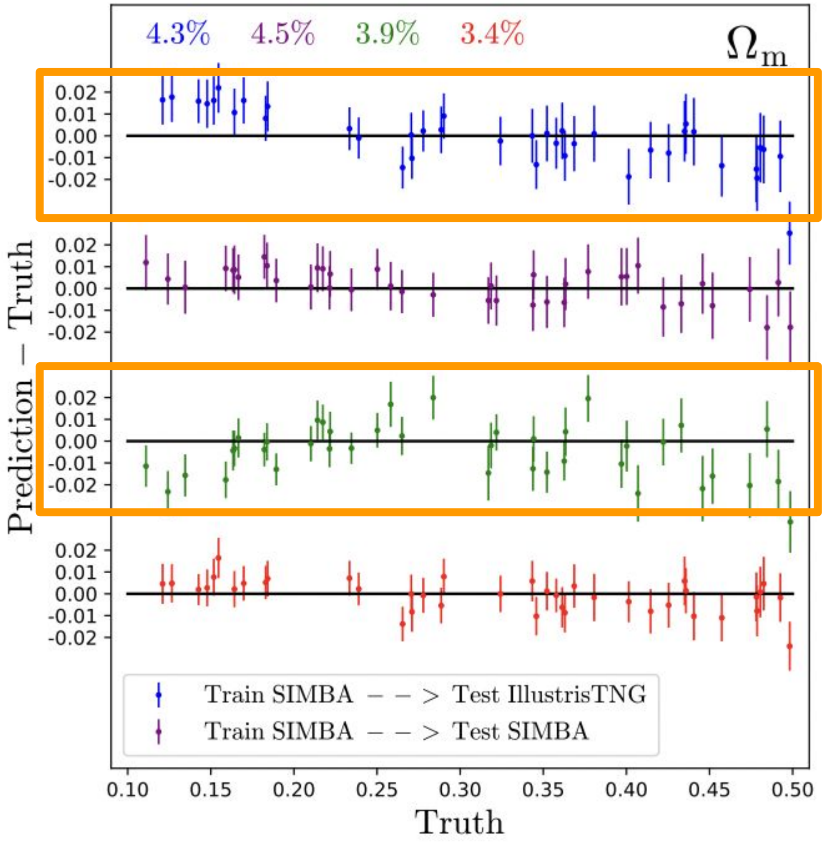




Self-Supervised Learning for Marginalization Over Systematics and Nuisance Parameters

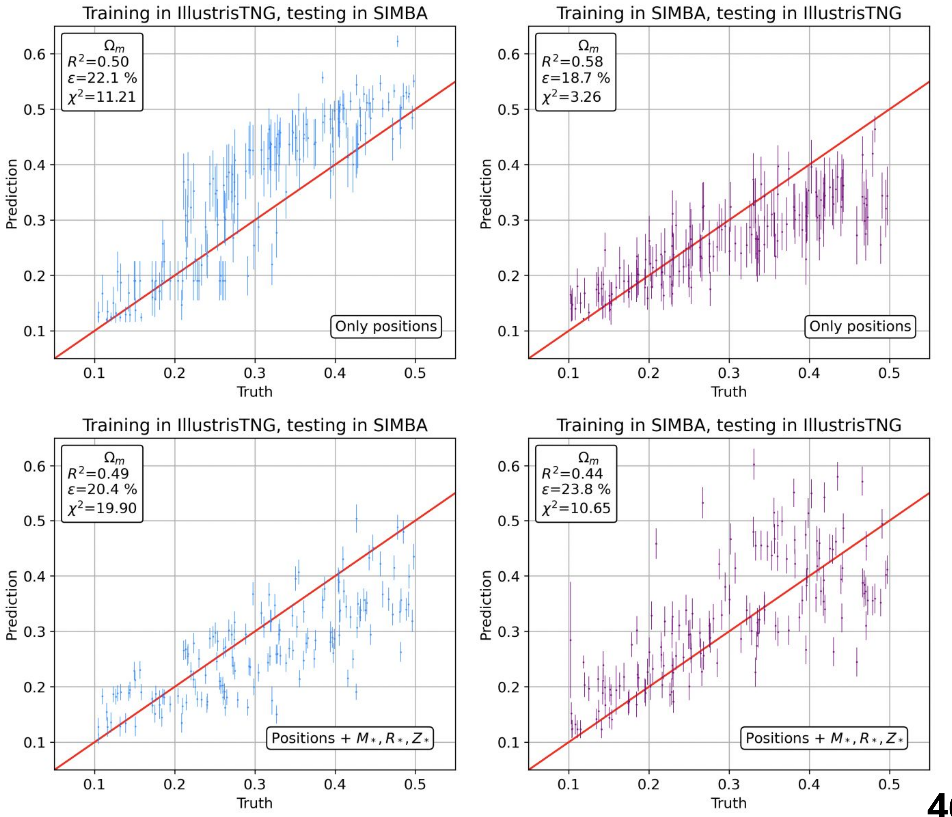
Robustness of ML models

- *Some studies* have found machine learning models that are *robust to variations in ‘sub-grid’ physics* across different simulations.
- Villaescusa-Navarro et al. 2021:
  - CNNs trained on mass density maps from one suite simulation, tested on maps from the other (SIMBA and IllustrisTNG)
  - Were able to recover *percent-level errors* on Ω_M and σ_8



Robustness of ML models

- Others, however, *do not generalize well* when applied to data from *new, previously unseen suites* of simulation
- Villanueva-Domingo et al. 2022:
  - Constructed Graph Neural Networks (GNNs) using information about positions and properties of galaxies to predict Ω_M and σ_8 from simulations suites
  - *Models fail to generalize*, even when using additional information about the galaxies



Distance Correlation (dCorr)

- **Distance correlation** is a measure of dependence between two random vectors X, Y
  - Captures both linear and non-linear dependence between the vectors
  - Only zero if the two vectors are independent (otherwise, 0< dCorr ≤ 1)
  - The two vectors do not have to have the same dimensionality
- Distance correlation can be computed as follows:

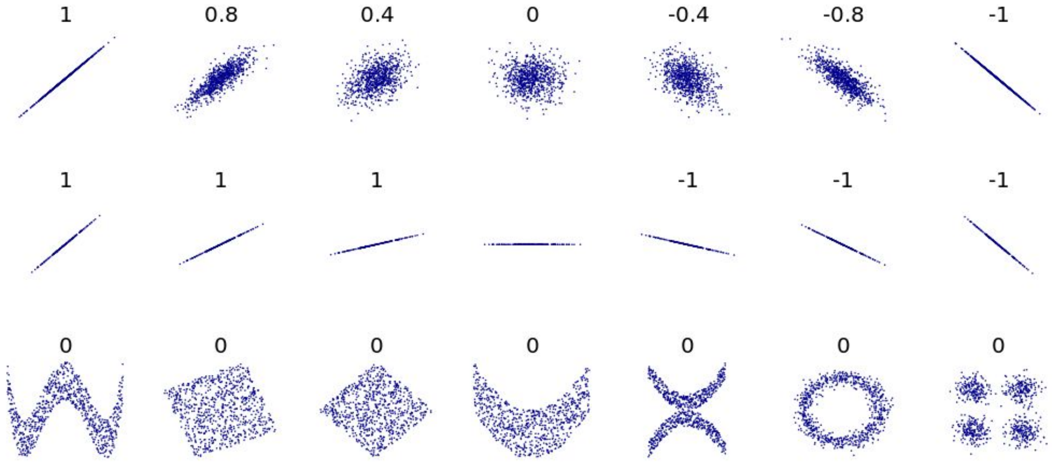
$$\mathcal{R}(X,Y) = \begin{cases} \frac{\mathcal{V}^2(X,Y)}{\sqrt{\mathcal{V}^2(X)}\sqrt{\mathcal{V}^2(Y)}}, & \mathcal{V}^2(X)\mathcal{V}^2(Y) > 0 \\ 0, & \mathcal{V}^2(X)\mathcal{V}^2(Y) = 0 \end{cases}$$

where the distance variance is defined as an element-wise product of doubly-centered distance matrices *A,B*:

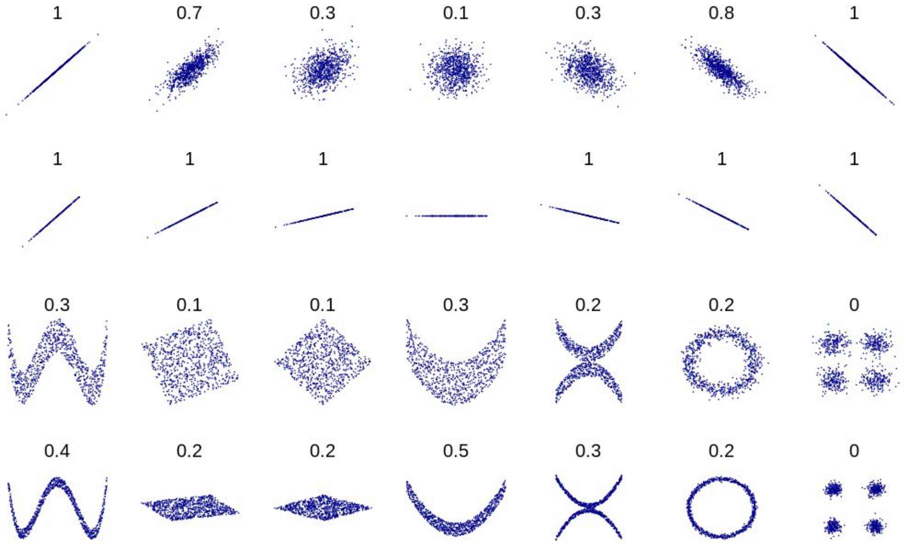
$$\mathcal{V}_N^2(X,Y) = \frac{1}{N^2}\sum_{k,l=1}^{n} A_{kl}B_{kl}$$ where $A_{j,k} = a_{j,k} - \bar{a}_{j\cdot} - \bar{a}_{\cdot k} + \bar{a}_{\cdot\cdot},$

and the distance matrix is $a_{j,k} = \|X_j - X_k\|$ with $\bar{a}_{j\cdot}$, $\bar{a}_{\cdot k}$, $\bar{a}_{\cdot\cdot}$ defined as the row, column, and overall means of the distance matrix.

Pearson Correlation Coefficient



Distance Correlation



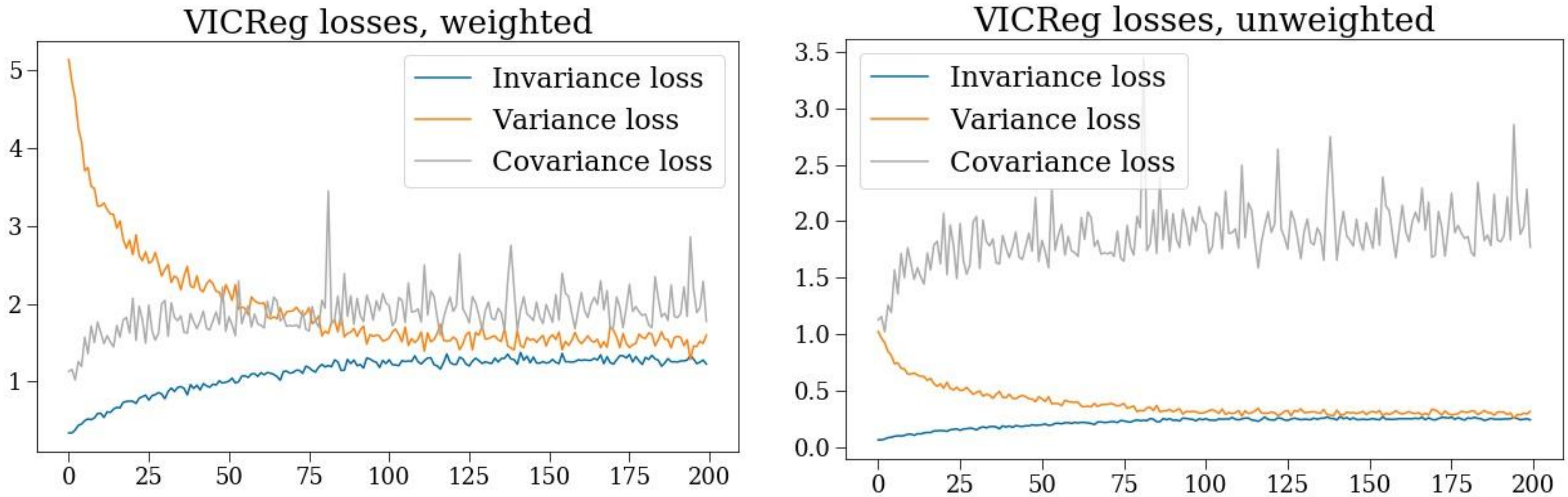
Mutual Information (MI)

$$I(X,Y) \equiv D_{\mathrm{KL}}\left(P_{XY} || P_X \otimes P_Y\right) = \int P_{XY}(x,y) \log \frac{P_{XY}(x,y)}{P_X(x)P_Y(y)} dxdy,$$

- MI is another measure of mutual dependence (beyond linear) between random variables
  - Quantifies how much information one gains about a random variable ***X*** by observing another random variable ***Y***
  - Can be expressed in terms of entropy: $I(X, Y) = H(X) - H(X|Y) = I(Y, X)$
  - MI is non-negative $I(X, Y) \geq 0$; $I(X, Y) = 0$ only if X and Y are independent
  - In general, estimating MI for variables in higher-dimensional spaces is challenging
- We estimate MI with a variational method approach called MINE (Mutual Information Neural Estimation)
  - The idea of MINE is to estimate a lower bound on the MI (the Donsker-Varadhan bound) by training a neural network with the corresponding cost function

Self-Supervised Learning for Data Compression and Parameter Inference

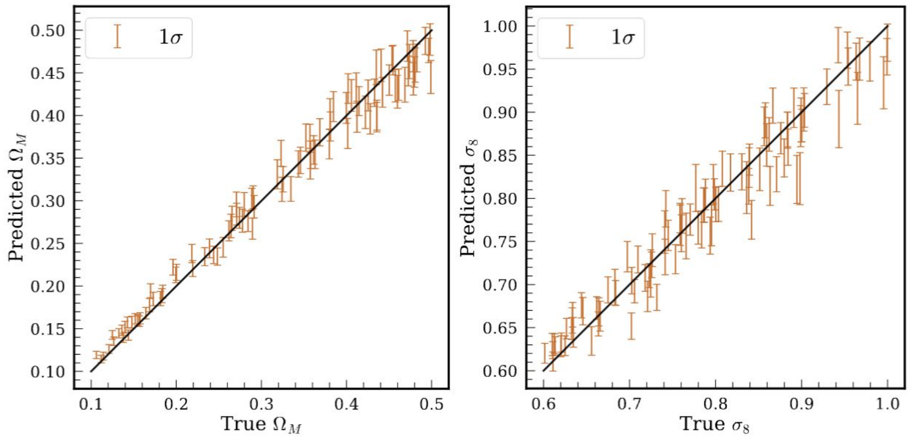
Contributions of different loss terms (lognormal fields dataset)



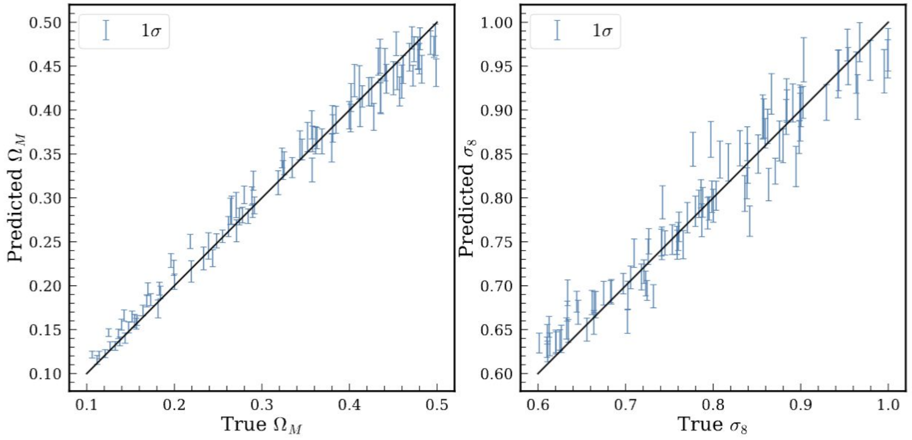
Testing on Out of Distribution Data (Trained on SIMBA, Tested on IllustrisTNG)

Method	Loss	MSE	MSE on Ω_M (Relative error)	MSE on σ_8 (Relative error)
VICReg	-2.55	4.73×10^{-4}	3.19×10^{-4} (4.65%)	6.26×10^{-4} (2.53%)
Supervised	-3.29	3.92×10^{-4}	2.55×10^{-4} (4.36%)	5.28×10^{-4} (2.21%)

(a) Trained on SIMBA, tested on IllustrisTNG.



(a) Supervised: trained on SIMBA, tested on IllustrisTNG.

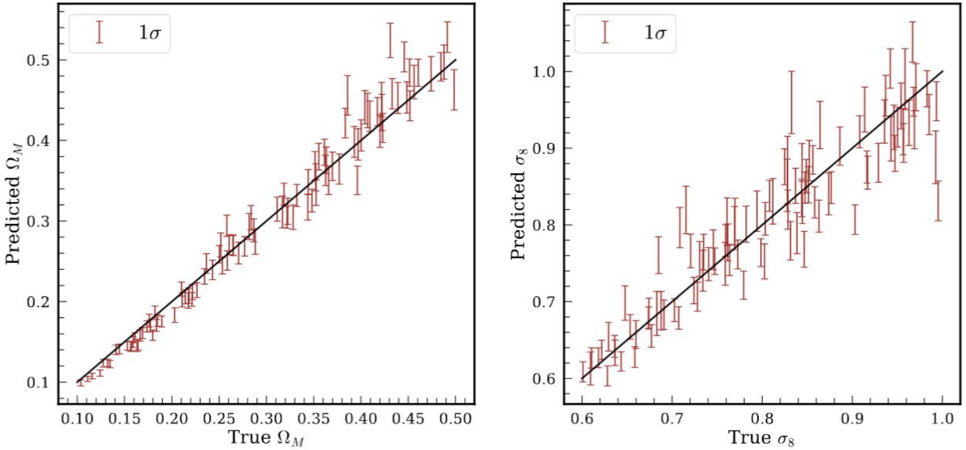


(b) VICReg: trained on SIMBA, tested on IllustrisTNG.

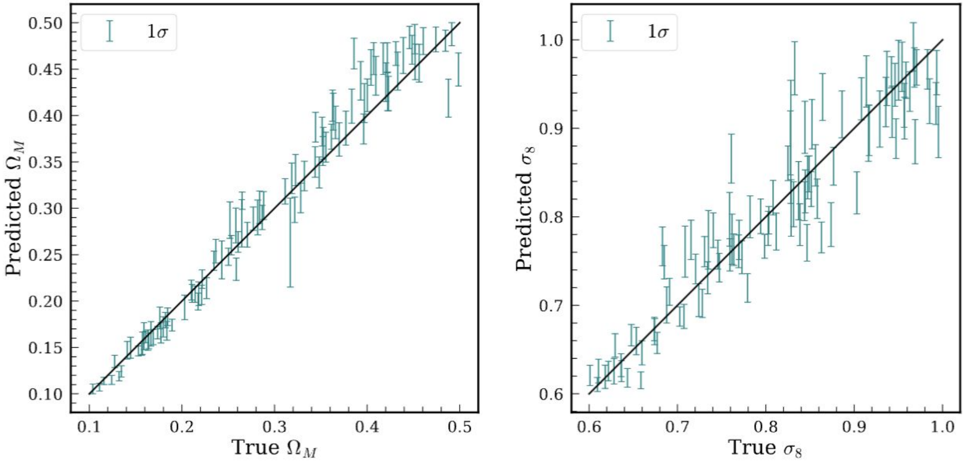
Testing on Out of Distribution Data (Trained on IllustrisTNG, Tested on SIMBA)

Method	Loss	MSE	MSE on Ω_M (Relative error)	MSE on σ_8 (Relative error)
VICReg	-2.00	9.14×10^{-4}	5.08×10^{-4} (5.27%)	13.2×10^{-4} (3.24%)
Supervised	-2.54	8.06×10^{-4}	3.92×10^{-4} (4.96%)	12.2×10^{-4} (3.17%)

(b) Trained on IllustrisTNG, tested on SIMBA.



(a) Supervised: trained on IllustrisTNG, tested on SIMBA.



(b) VICReg: trained on IllustrisTNG, tested on SIMBA.