

PAPER • OPEN ACCESS

Non-parametric Lagrangian biasing from the insights of neural nets

To cite this article: Xiaohan Wu *et al* JCAP05(2023)040

View the [article online](#) for updates and enhancements.

You may also like

- [A MEASUREMENT OF SMALL-SCALE STRUCTURE IN THE \$2.2 \leq z \leq 4.2\$ Ly FOREST](#)
Adam Lidz, Claude-André Faucher-Giguère, Aldo Dall'Aglio et al.
- [Cosmological Parameter Estimation Using the Genus Amplitude—Application to Mock Galaxy Catalogs](#)
Stephen Appleby, Changbom Park, Sungwook E. Hong et al.
- [Comparative Analysis of TRGBs \(CATs\) from Unsupervised, Multi-halo-field Measurements: Contrast is Key](#)
J. Wu, D. Scolnic, A. G. Riess et al.

Non-parametric Lagrangian biasing from the insights of neural nets

Xiaohan Wu,^{a,b,*} Julian B. Muñoz^a and Daniel J. Eisenstein^a

^aHarvard-Smithsonian Center for Astrophysics,
60 Garden Street, Cambridge 02138, MA, U.S.A.

^bCanadian Institute for Theoretical Astrophysics, University of Toronto,
Toronto, ON, M5S 3H8, Canada

E-mail: xwu@cita.utoronto.edu, julianmunoz@cfa.harvard.edu,
deisenstein@cfa.harvard.edu

Received December 19, 2022

Revised April 5, 2023

Accepted May 1, 2023

Published May 22, 2023

Abstract. We present a Lagrangian model of galaxy clustering bias in which we train a neural net using the local properties of the smoothed initial density field to predict the late-time mass-weighted halo field. By fitting the mass-weighted halo field in the ABACUSSUMMIT simulations at $z = 0.5$, we find that including three coarsely spaced smoothing scales gives the best recovery of the halo power spectrum. Adding more smoothing scales may lead to 2–5% underestimation of the large-scale power and can cause the neural net to overfit. We find that the fitted halo-to-mass ratio can be well described by two directions in the original high-dimension feature space. Projecting the original features into these two principal components and re-training the neural net either reproduces the original training result, or outperforms it with a better match of the halo power spectrum. The elements of the principal components are unlikely to be assigned physical meanings, partly owing to the features being highly correlated between different smoothing scales. Our work illustrates a potential need to include multiple smoothing scales when studying galaxy bias, and this can be done easily with machine-learning methods that can take in high dimensional input feature space.

Keywords: cosmic web, redshift surveys, cosmological simulations

ArXiv ePrint: [2212.08095](https://arxiv.org/abs/2212.08095)

*Corresponding author.



Contents

1	Introduction	1
2	Methods	2
2.1	Neural net	2
2.2	Simulations and training	4
3	Exploring different combinations of smoothing scales	5
3.1	Comparison to least squares	5
3.2	Results with different smoothing scales	7
4	Reducing the dimensions of the input feature space	11
5	Modeling a thin mass range	14
6	Conclusions	16
A	Loss as a function of epoch	17

1 Introduction

Galaxy surveys that map the large-scale structure of the universe have been a powerful probe of the composition of our universe and how its structures form ([for a recent review, see 1]). In the canonical picture, structure formation originates from the gravitational collapse of initially small perturbations in the matter density field, which eventually grow into dark-matter halos. Galaxy formation then proceeds within these halos by gas accretion and cooling, producing galaxies that act as biased tracers of the underlying density field. An improved understanding of where dark-matter halos sit in the underlying matter field is crucial for modeling the formation of the large-scale structure and therefore constraining cosmological parameters. This, in turn, is key to understand the physics of inflation, and the dark sector of our universe.

Traditionally, the halo density is assumed to trace the underlying matter field in a polynomial form, with a series of bias parameters characterizing the relation [e.g. 2–4]. This bias expansion approach has achieved much success in describing summary statistics such as the galaxy power spectrum and bispectrum [5–11] as well as halos at the field level [12–15], and previous works have focused on evaluating the biases [16–24]. In ref. [25] (hereafter Paper I), we developed a fully non-parametric framework to calculate the distribution of halos at a given redshift given the initial Lagrangian density field. This goes beyond the traditional bias expansion approach and instead fits a function f that characterizes how much halo mass should form given properties of the initial density field. Specifically, we model the initial Lagrangian-space (pre-advection) halo overdensity δ_h as

$$1 + \delta_h = f(\delta, \nabla^2 \delta, \mathcal{G}_2), \quad (1.1)$$

where δ is the Lagrangian matter overdensity and \mathcal{G}_2 is the tidal operator, evaluated at each point in the initial Lagrangian space [3, 4, 26, 27]. We showed that for mass-weighted halos above a mass threshold, the shape of f clearly deviates from a polynomial of δ and other

quantities. Therefore while the bias expansion approach has been successful and physically intuitive [from the peak-background split argument, 28–30], it can yield an unphysical relation between the galaxy and matter fields such as a non-positive-definite f and enhanced biases for underdense regions. We demonstrated that our fitted f can recover the late-time halo power spectrum to sub-percent level at wavenumbers $k < 0.1 \, h \, \text{Mpc}^{-1}$

In this work we expand upon our previous Paper I and train a neural network (NN) to obtain f as a function of particle features associated with various smoothing scales. Our previous formalism involves dividing the feature space into a finite number of bins and solving for a piece-wise constant f with least-squares fitting. This limited us to using at most two features (such as δ and $\nabla^2\delta$) owing to memory restrictions. A NN, however, allows f to be a continuous function of a large number of features. It also captures the impact of the cross terms of the input features on the halo field as in the bias expansion approach, since the input features get interwoven after passing through two hidden layers. We show that with $\delta, \nabla^2\delta, \mathcal{G}_2$ associated with three smoothing scales appropriately chosen, the f function predicted by the NN is able to recover the halo power spectrum to within 1–2% at $k \lesssim 0.1 \, h \, \text{Mpc}^{-1}$ despite being trained at the field level. We find that for mass-weighted halos, f can be well described by two orthogonal directions in the original feature space. Projecting the original $\delta, \nabla^2\delta, \mathcal{G}_2$ into these two directions and re-training the NN reproduces the original training results of the halo power spectrum at least as well. While our work is closely related to refs. [31–33] who also studied halo formation and large-scale structure using machine learning methods, we focus on predicting the entire halo field instead of halo masses as done in these works.

This paper proceeds as follows. Section 2 presents our setup of the NN and the simulations. Section 3 discusses results of using different combinations of smoothing scales. Section 4 examines the structure of f using the principal components of ∇f . We briefly explore modeling a thin mass range instead of a mass threshold in section 5, and discuss future directions section 6.

2 Methods

2.1 Neural net

We aim to train a NN to predict the f weight that a particle should carry given its input features, so that such an f function best recovers the real-space halo field δ_h^{true} . To obtain the particle features, we first compute the smoothed overdensity field δ , its Laplacian $\nabla^2\delta$, and the corresponding tidal shear \mathcal{G}_2 using the initial condition of a simulation given a smoothing scale R_f . These fields are normalized to have standard deviation of 1. We then assign each particle in a simulation a certain set of $\delta, \nabla^2\delta, \mathcal{G}_2$ values according to its nearest grid point in the initial condition. In the presence of multiple smoothing scales R_{f1}, R_{f2}, \dots , a particle carries a vector of features

$$\theta_{\text{orig}} = (\delta_1, \nabla^2\delta_1, \mathcal{G}_{2,1}, \delta_2, \nabla^2\delta_2, \mathcal{G}_{2,2}, \dots)^T, \quad (2.1)$$

where the subscripts denote the indices of the smoothing scales. These features are not independent of each other. Specifically, the δ 's and $\nabla^2\delta$'s with different smoothing scales are highly correlated with each other. The \mathcal{G}_2 's are uncorrelated with δ 's and $\nabla^2\delta$'s since these are quadratic terms, but are highly correlated among themselves.

To expose the independent degrees of freedom to the NN, we opt to orthogonalize the original features. We thus calculate the covariance matrix of the features

$$\Sigma_{ij} = (\theta_{\text{orig},i}, \dots) \cdot (\theta_{\text{orig},j}, \dots)^T, \quad (2.2)$$

where the subscripts i, j denote the indices of particles, and the size of the covariance matrix is the number of original features squared. We then obtain the eigenvalues Λ and eigenvectors V of Σ , where Λ is a diagonal matrix with the eigenvalues as the diagonal elements, and the columns of the matrix V are the corresponding eigenvectors. Defining a matrix $\Lambda^{-1/2}$, which is diagonal and the elements are the inverse of the square root of Λ , the orthogonalized and normalized features are

$$\theta_{\text{orth}} = \Lambda^{-1/2} V^T \theta_{\text{orig}}. \quad (2.3)$$

Since the original features are highly correlated among each other, the range of the square root of their eigenvalues can span three or four magnitudes. It is only desirable to keep the transformed features with larger eigenvalues. Intuitively, the $\nabla^2 \delta$'s can be written as the derivative of the smoothed δ 's with respect to the smoothing scale, so can be approximated with a linear combination of δ 's with adjacent smoothing scales. For a number n_{R_f} of fine-spaced smoothing scales, we thus only need the n_{R_f} δ 's and one $\nabla^2 \delta$ with the smallest smoothing scale to approximate all the $2n_{R_f}$ δ 's and $\nabla^2 \delta$'s. This motivates us to keep the $n + 1$ eigenvectors with the largest eigenvalues calculated using only δ 's and $\nabla^2 \delta$'s.¹ We keep all of the n_{R_f} \mathcal{G}_2 's, since in the case of coarsely separated R_f 's the associated \mathcal{G}_2 's are not highly degenerate.² This leaves us with a total of $2n_{R_f} + 1$ orthogonalized and normalized features that should be input into the NN, which we represent by a projection matrix P

$$\theta_{\text{input}} = P \theta_{\text{orth}} = P \Lambda^{-1/2} V^T \theta_{\text{orig}}. \quad (2.4)$$

Here P is a diagonal matrix of size n^2 , with $2n_{R_f} + 1$ of the diagonal elements being 1 and the others being 0, which projects out the unused θ_{orth} .

For each particle, given its orthogonalized and normalized features θ , a NN predicts an f value that it should carry. We then sum up the f values of particles in each cell to obtain the model halo field:

$$1 + \delta_{h,j}^{\text{model}} = \frac{N_{\text{cell}}}{N_{\text{part}}} \sum_{i \in \text{cell}_j} f(\theta_{\text{input},i}) \quad (2.5)$$

where j denotes the index of a cell. Here the sum of f should be multiplied by the size of the halo grid N_{cell} divided by the total number of particles N_{part} to ensure that the mean of the halo overdensity is 0. The loss function is defined as

$$L = \sum_j (\delta_{h,j}^{\text{model}} - \delta_{h,j}^{\text{true}})^2. \quad (2.6)$$

The summation is carried over each batch in the training, where a batch is a number of grid cells with the particles in them.

In addition to the squared loss, the predicted f 's should be non-negative and satisfy the integral constraint $\sum_i f_i / N_{\text{part}} = 1$ owing to mass conservation, where i denotes the particle indices. To enforce the non-negativity constraint, we make the NN predict $y = \log_{10}(f)$ instead of f , and add an additional penalty $(y + 5)^4$ for each particle to prevent y from going too negative and thus yielding a vanishing gradient. To implement the integral constraint,

¹We have verified that for small separation of the smoothing scales, inputting n_{R_f} δ 's and only the $\nabla^2 \delta$ of the smallest smoothing scale yields the same training results as inputting $2n_{R_f}$ δ 's and $\nabla^2 \delta$'s and keeping the largest $2n_{R_f} + 1$ eigenvectors. This no longer holds with large separations of the smoothing scales.

²We find that keeping $2n_{R_f} + 1$ features instead of choosing the features based on a cut-off of the ratios of the eigenvalues yields more stable training results. Specifically, if the eigenvectors and eigenvalues have small changes when calculated from a different simulation box, the resulting f appears to be robust.

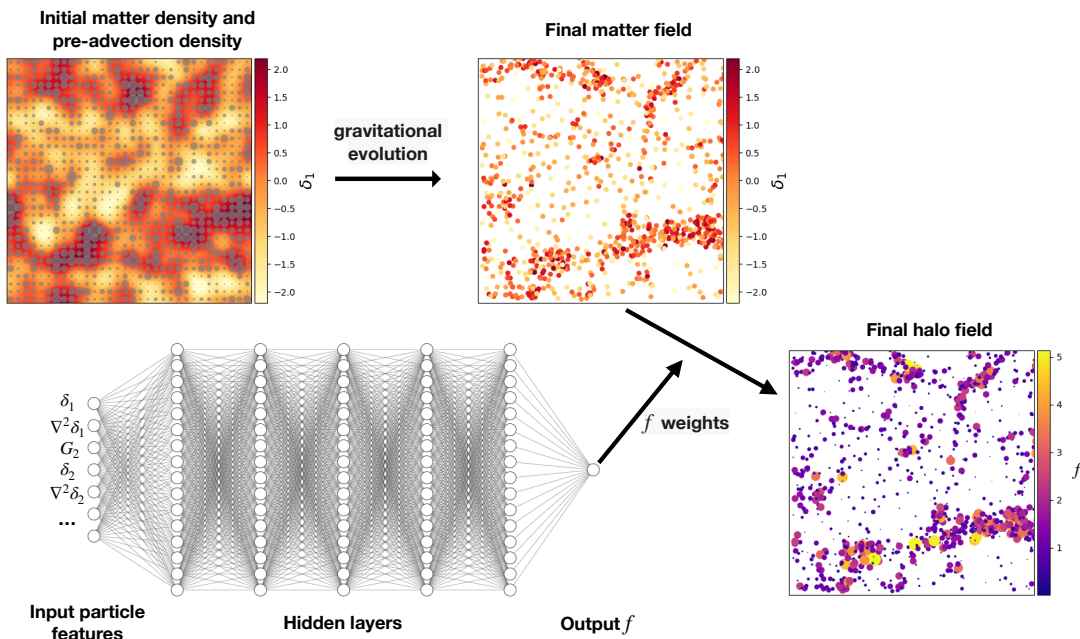


Figure 1. A schematic illustration of our training process. Top left panel shows the initial matter density field, illustrated by colors. The overlying gray circles represent the f weights that the particles should carry, with the size of the circles showing the amplitude of f . The f weights are predicted by the NN shown at the bottom left, by inputting the initial $\delta, \nabla^2 \delta, G_2$ at various smoothing scales. The top right panel illustrates the final matter field, where the particles shown by circles have been moved by gravity. The bottom right panel presents the final halo field, where the circles follow the locations of the particles and the colors and sizes of the circles represent the amplitude of f . We calculate the loss based on the residual in modeling the final halo field.

we choose the batch size so that each batch, which is a collection of cells, contains about $N_{\text{part, batch}} = 10^5$ particles in total. The batches are randomly initialized before the training and kept unchanged during the training. For each batch, in addition to the squared loss and the non-negativity loss, we add another loss $(\sum_i f_i / N_{\text{part, batch}} - 1)^2 / \epsilon^2$, where ϵ is the tolerance. During the first epoch of training, we do not include the integral constraint, and the NN recovers the integral constraint at a few percent level. Starting from the second epoch, we set ϵ as a function of the number of epochs $\max(10^{(-1-3\log_{10}(\text{epoch}))}, 10^{-4})$, so that it gradually decreases. This ensures that NN recovers the integral constraint at sub-percent level.

We stress that our NN makes predictions for particles, but our loss function is defined on the halo grid. The backward propagation is handled internally by `pytorch` as long as the loss function is implemented correctly. Figure 1 gives a schematic illustration of the training process, where the particles carry their corresponding f weights predicted from the NN (gray circles in the top left panel) given at the initial time to their final locations, forming the final halo field (bottom right panel).

2.2 Simulations and training

We use the *AbacusSummit* simulations [34] in this work to perform the training, which are run with the *Abacus* N-body simulation code [35–38]. These simulations were designed to meet and exceed the currently stated cosmological simulation requirements of the Dark Energy Spectroscopic Instrument (DESI) survey [39]. We utilize the small box simulations

which have a box size of $500 h^{-1}$ Mpc and 1728^3 particles, using the Planck2018 standard cosmology [40]: $\Omega_m = 0.14237$, $h = 0.6736$, $\sigma_8 = 0.807952$. This gives a particle mass of $2 \times 10^9 h^{-1} M_\odot$. Halos are identified on the fly with the **CompasO** halo finder which uses a hybrid FoF-SO algorithm [41]. We focus on fitting the mass-weighted halo field with halo masses $M > 3 \times 10^{11} M_\odot$ at $z = 0.5$, corresponding to > 150 particles.

The initial conditions were generated at $z = 99$ using the method proposed in [42]. To obtain the $(\delta, \nabla^2 \delta, \mathcal{G}_2)$ values associated with a particle, we interpolated the initial density field onto 576^3 grids and calculated the smoothed $(\delta, \nabla^2 \delta, \mathcal{G}_2)$ values on each grid point given a smoothing scale R_f . We then assign $(\delta, \nabla^2 \delta, \mathcal{G}_2)$ values to each particle by looking for the nearest grid point to the particle’s location in the initial space. These are then input into the NN for the training.

To train a NN, we produce halo grids by summing the f values of particles inside each cell. We note that this calculation of δ_h^{model} corresponds to using nearest neighbor interpolation (NNB), so the true halo field δ_h^{true} is computed with NNB as well. We use 100^3 grids, which gives a cell size of $5 h^{-1}$ Mpc. To reduce the computational cost, for each grid cell, we only sample 10% of the particles in it to perform the training. This fraction guarantees that for our grid size of $5 h^{-1}$ Mpc, there is at least one particle in each cell. The summed f ’s should then be scaled up by a factor of 10 to correctly obtain δ_h^{model} .

We implement a NN with 5 hidden layers, each with 64 neurons. We use the GeLU activation function for all layers and the Adam optimizer for gradient descent. We adopt an initial learning rate of 0.02, which decays as 0.9^{epoch} . This exponential decay ensures that the loss becomes rather flat after 20 epochs of training, and after 50 epochs of training the model predicted halo power spectrum converges as well. We chose these hyperparameters so that in the case of one smoothing scale, the model predicted halo power spectrum changes at $< 1 - 2\%$ level under different initializations of the NN and different sub-sampling of the particles. Utilizing even larger NNs does not improve the performance on the halo power spectrum any further. We find that the validation loss has the same trend of variation as the training loss, so we based the choice of the hyperparameters solely on the training loss and power spectrum.

Each training process takes one simulation as the training set and other simulations as the validation set. Here we use the phrase “validation set” interchangeably as “test set”, since we effectively did not use any validation simulation to tune the hyperparameters. Since our box size is only $500 h^{-1}$ Mpc, using multiple simulations as the validation set allows us to examine the impact of cosmic variance on the trained f . We train the NN five times and calculate the average f , each time with a different sub-sampling of the particles and initialization of the NN weights. During a training process, we do not change the sub-sampled particles. To examine the halo power spectrum, we apply the trained f function onto all particles and create the halo field on a 256^3 grid with Cloud-In-Cell interpolation. We choose to use this finer grid when computing the halo power spectrum to avoid aliasing effects [43]. We will assess the range of scatter in power owing to the changes of f in different times of training.

3 Exploring different combinations of smoothing scales

3.1 Comparison to least squares

As a first crosscheck, we begin by comparing the NN results to those of the least-squares formalism we developed in Paper I. In Paper I, we only include the δ and $\nabla^2 \delta$ values of one

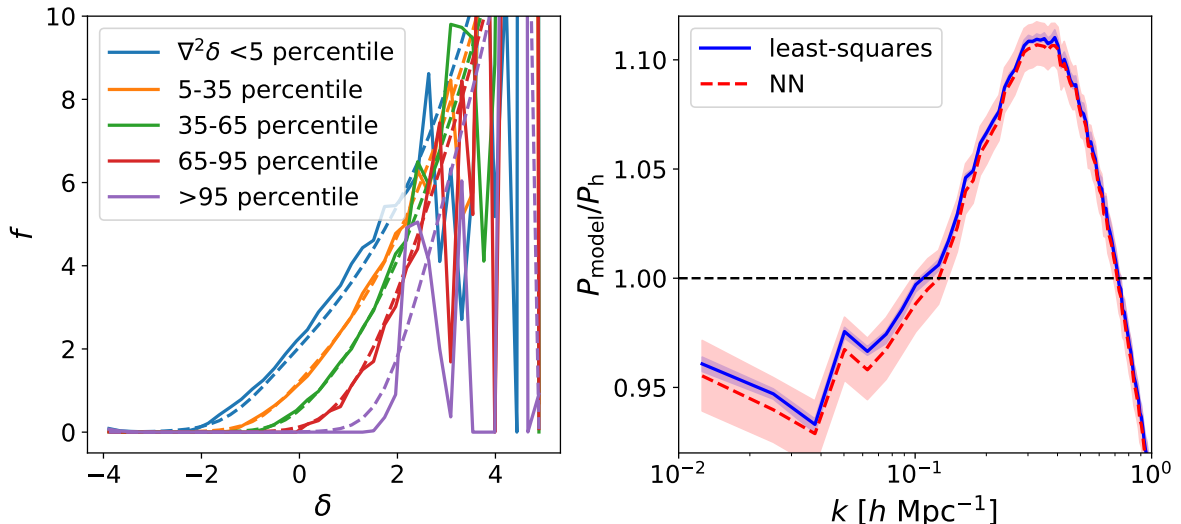


Figure 2. Comparison between the least-squares solution obtained from our formalism in Paper I and our NN results on one training simulation. We use one smoothing scale $R_f = 2.83 h^{-1} \text{ Mpc}$ and its associated $\delta, \nabla^2 \delta$, and trained on one training simulation. Left panel: f as a function of δ in different $\nabla^2 \delta$ bins, represented by different colors. Solid and dashed lines show the least-squares and the NN results, respectively. Right: the ratio of the model power spectrum to the halo power spectrum, evaluated on one validation simulation. Blue solid and red dashed lines illustrate results from least-squares and NN, respectively. Shades represent the minimum and maximum ranges obtained from 5 times of training and particle sub-sampling (of the same box), and lines show the results obtained from the average f . The NN recovers the least-squares results well and produces a much smoother f .

smoothing scale for the particles, and divided the δ - $\nabla^2 \delta$ plane into bins according to the percentiles of $\nabla^2 \delta$ at each δ value. We then obtained the least-squares solution to f in each bin by minimizing the mean squared error of the halo field, with the constraints that f is non-negative and should integrate to 1.

Using the mass-weighted halo field with $M > 3 \times 10^{11} M_\odot$ of one training simulation, we calculate the least-squares solution of f with $R_f = 2.83 h^{-1} \text{ Mpc}$. To make fair comparison, we train a NN with the δ and $\nabla^2 \delta$ of this R_f . Our R_f choice is based on the mass-weighted mean mass of $M > 3 \times 10^{11} h^{-1} M_\odot$ being $2.4 \times 10^{13} h^{-1} M_\odot$, corresponding to a Gaussian filter with $R_f = 2.6 h^{-1} \text{ Mpc}$. Since we train the NN five different times each with a different sub-sampling of particles, we also sub-sample 10% of the particles five times to obtain the least-squares solution. We then compute the average f solution and apply it on one validation simulation to calculate the halo power spectrum.

The left panel of figure 2 compares the f function obtained from least-squares (solid lines) and from the NN (dashed lines), with the colors indicating five $\nabla^2 \delta$ bins. The NN solution traces the least-squares solution, but is remarkably smoother. The least-squares solution, on the other hand, shows large noise at high δ values. The right panel illustrates the ratio of the model power spectrum to the halo power spectrum P_{model}/P_h calculated using one validation simulation. The solid and dashed lines show P_{model} obtained from the average f function from least-squares and NN, respectively. Shades represent the minimum and maximum ranges obtained from 5 times of training and random sampling of particles. Again, the average f of the NNs reproduces the least-squares power spectrum well.

Name	R_f (h^{-1} Mpc)
1 R_f	2.83
3 R_f	1.41, 2.83, 5.66
3 $R_f + 11.3$	1.41, 2.83, 5.66, 11.3
3 R_f coarse	1, 2.83, 8
3 R_f fine	2, 2.83, 4
5 R_f fine	1.41, 2, 2.83, 4, 5.66

Table 1. The sequences of smoothing scales used for training and their short names.

The NN-predicted f yields noticeably larger scatter ($\sim 1\%$) in the halo power spectrum at $k < 0.1 h \text{ Mpc}^{-1}$ when trained five different times, compared to the $\ll 1\%$ scatter produced by the least-squares f . Since the least-squares solution is deterministic and noisy but the NN solution is smooth, the NN tries to match the least-squares solution but can never reach the low loss created by least-squares. A small change in f , especially at the high- δ end, may not affect the real-space loss noticeably but will leave a larger imprint on the predicted bias of the halos. As we will show below, we find that such 1–2% fluctuations in power are persistent in all NN results, regardless of the smoothing scales used. We will examine these scatter in more detail below.

We also note that both the least-squares and the NN solutions underpredict the halo power by $\sim 5\%$ at $k \lesssim 0.1 h \text{ Mpc}^{-1}$. We showed in Paper I that this deficit can be alleviated by including a high- k cut-off to our loss function at $k = 1/R_f$. We do not perform this cut-off in this work, but rather try to improve the match to the halo field (and indirectly to the power spectrum) by including different smoothing scales.

3.2 Results with different smoothing scales

We now compare the NN results with different combinations of smoothing scales, since density fluctuations at different R_f 's can contribute to halo collapse with different weights. The least-square method becomes computationally unfeasible for more than 2 input features since the number of bins grows exponentially with the dimension of the feature spaces, but the NN is able to incorporate a much higher dimensional input space. For each R_f , we include its associated $\delta, \nabla^2\delta, \mathcal{G}_2$ into the input features. As mentioned in section 2, we orthogonalize the features and reduce the $3n_{R_f}$ -dimensional feature space to a $(2n_{R_f} + 1)$ -dimensional one based on the eigenvalues, where n_{R_f} is the number of smoothing scales. For each sequence of R_f 's, we make it a geometric series with a common ratio equal to some power of $\sqrt{2}$. Table 1 summarizes the R_f 's that we use and the names referring to them later in the plots. We use the ratio of the model power spectrum to the halo power spectrum $P_{\text{model}}/P_{\text{h}}$ as a metric, though we note that it is not included in our loss.

We first examine the effects of using multiple R_f 's instead of one. Figure 3 contrasts the training results using $R_f = 1.41, 2.83, 5.66 h^{-1} \text{ Mpc}$ (dashed lines) with those of $R_f = 2.83 h^{-1} \text{ Mpc}$ (solid lines). Left column shows the result of applying f from the training simulation to the training simulation. Top left and bottom left panels present $P_{\text{model}}/P_{\text{h}}$, and the ratio of the power spectra of the uncorrelated residual P_{uncorr} to the halo power spectrum

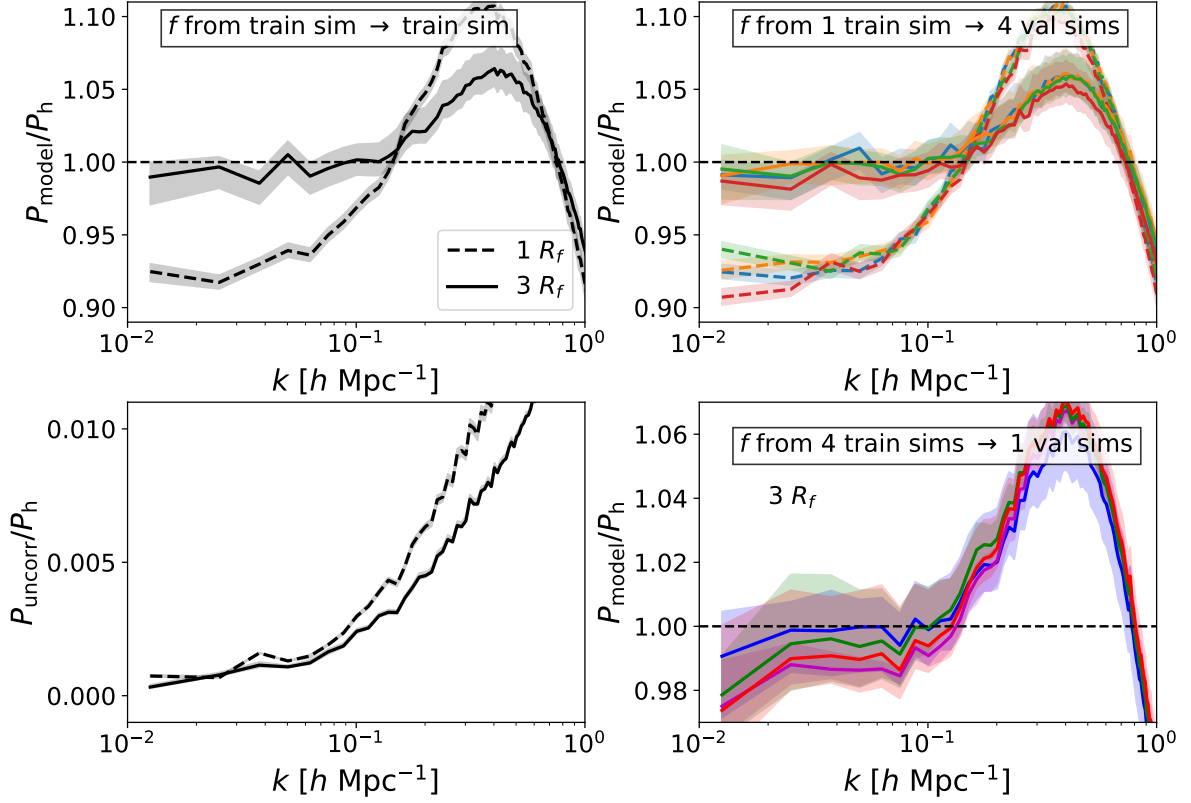


Figure 3. Comparison between the training results obtained using one smoothing scale (dashed lines) to using 3 smoothing scales (solid lines), and impact of cosmic variance on the model power spectrum. Shades represent the minimum and maximum ranges of the power spectra obtained from 5 times of training, and lines show the results obtained from the average f . Left column: results are obtained by applying f from the training simulation to the training simulation. Top left and bottom left panels show the ratio of the model power spectra to the halo power spectrum, and the ratio of the power spectra of the uncorrelated residual to the halo power spectrum respectively. Top right panel: $P_{\text{model}}/P_{\text{h}}$, calculated by applying f from the training simulation to 4 validation simulations (different colors). Bottom right panel: $P_{\text{model}}/P_{\text{h}}$, calculated by applying f from 4 training simulations (different colors) to 1 validation simulation. Adding the two additional smoothing scales significantly improves the match of the halo power spectrum, although cosmic variance leads to 1–2% scatter in the resulting halo power spectrum.

respectively. Here

$$P_{\text{uncorr}} = P_{\text{model}} - \frac{P_{\text{h,model}}^2}{P_{\text{h}}}, \quad (3.1)$$

where $P_{\text{h,model}}$ is the cross spectrum between the halo field measured and modeled. This uncorrelated residual characterizes the part in the model halo field that is uncorrelated with the true halo field [20, 25].

Using 3 smoothing scales dramatically improves the match of P_{model} to P_{h} , bringing the ratio to 1 within $\sim 2\%$ at $k < 0.1 \, h \, \text{Mpc}^{-1}$. With only 1 smoothing scale, we find $P_{\text{model}}/P_{\text{h}} \lesssim 0.95$ at $k < 0.1 \, h \, \text{Mpc}^{-1}$. We emphasize that we fit the real-space halo field without filtering the residuals at $k > 1/R_f$ in Fourier space as we did in Paper I. Top right panel shows $P_{\text{model}}/P_{\text{h}}$, calculated by applying f from the training simulation to 4 validation

simulations (different colors). Results on the validation simulations also trace those of the training simulations, although cosmic variance can lead to up to 5% scatter in the low- k power of these small boxes.

There are about 1.5% variations in the model power spectrum of individual simulations when using 3 R_f , while the one smoothing scale case yields 0.5% scatter in power. Since $P_{\text{uncorr}}/P_h < 0.002$ at $k < 0.1 h \text{ Mpc}^{-1}$, all of the variations in P_{model} come from the residuals in the model halo field that is correlated with the true halo field. We find that whether the model underpredicts or overpredicts the power is uncorrelated with the value of the real-space squared loss (see appendix A for a more detailed discussion). As mentioned earlier, a small change in f , especially at high δ values, may not affect the real-space loss much since the number of high δ_h^{true} cells are small. However, changes in f can leave a more evident imprint on the power spectrum indicating overestimating or underestimating the halo bias. This issue seems more prominent when the dimension of the input feature space is higher. This could be alleviated by using a k -space loss with some emphasis on the low- k end or by explicitly including in the power spectrum in our fitting.

The fact that our small $500 h^{-1} \text{ Mpc}$ box size contains very few low- k modes can also contribute to large low- k scatter. The bottom right panel shows P_{model}/P_h calculated by applying f from 4 training simulations (different colors) to 1 validation simulation. In addition to the 1–2% scatter around each power spectrum using the average f from a single training simulation, there is also 1–2% scatter in the power spectra from the 4 average f 's from 4 different training simulations. This indicates that the fluctuations in the halo power spectrum may be sourced by changes in f owing to cosmic/sample variance. We thus expect a better match to the power spectrum by training with multiple small boxes at the same time or with a larger box size. Given the exploratory nature of this first paper, we leave it for future work to make these improvements.

We now test the impact of using other combinations of smoothing scales, by either adding R_f 's or using another splitting of the R_f 's. Here we only evaluate P_{model}/P_h by applying f from one training simulation to one validation simulation, since the training and validation power spectra show similar trends. The left and right panels of figure 4 show P_{model}/P_h in the case of using coarser (common ratio of 2 or $2^{3/2}$ in R_f) and finer (common ratio of $2^{1/2}$) R_f 's respectively. Shades represent the minimum and maximum ranges obtained from 5 times of training, and lines show the results obtained from the average f . The black line in the left panel illustrates the $R_f = 1.41, 2.83, 5.66 h^{-1} \text{ Mpc}$ case, and the red line shows the result of adding $R_f = 11.3 h^{-1} \text{ Mpc}$ to it. The blue line represents the case of $R_f = 1, 2.83, 8 h^{-1} \text{ Mpc}$. The magenta and green lines in the right panel show $R_f = 2, 2.83, 4 h^{-1} \text{ Mpc}$ and $R_f = 1.41, 2, 2.83, 4, 5.66 h^{-1} \text{ Mpc}$, respectively.

As mentioned above, all of these cases of training in a very high dimensional feature space lead to about 1–2% scatter in the model power spectrum. Using 3 smoothing scales with coarse spacing results in the best match of the power spectrum, and P_{model} agrees with P_h to 1% level. Adding a large $R_f = 11.3 h^{-1} \text{ Mpc}$ drastically lowers the low- k power by up to 5%, which is not reflected in any changes of the loss (see appendix A). In Paper I, we found that using large smoothing scales with small halo grid cells can lead to a $\sim 10\%$ underestimation of the power spectrum, if we do not filter out the high- k residuals. It is thus likely that the NN outweigh the contribution of the $R_f = 11.3 h^{-1} \text{ Mpc}$ features which then lowers the low- k power. Since it is non-trivial to perform the training in Fourier space and determine a proper high- k cut-off in the case of multiple smoothing scales, we will defer for a future work to implement these improvements and just warn the reader about fitting with too high R_f 's.

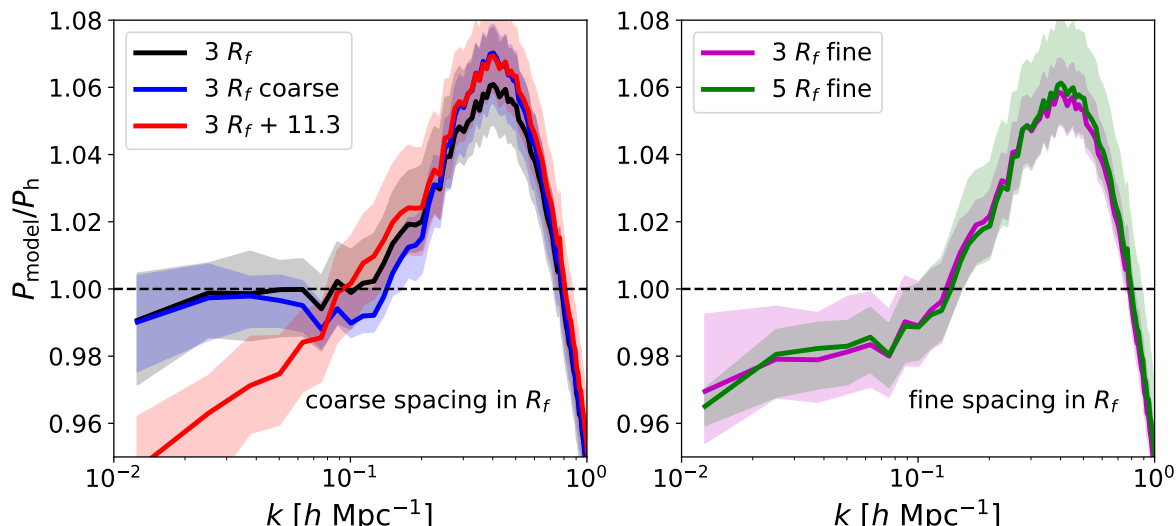


Figure 4. Comparison of training results using different combinations of smoothing scales, measured using the ratio of the model power spectra to the halo power spectra, evaluated for one validation simulation. The smoothing scales are separated by factors of 2, $2^{3/2}$ (coarse) in the left panels, and $\sqrt{2}$ (fine) in the right, respectively. Shades represent the minimum and maximum ranges obtained from 5 times of training, and lines show the results obtained from the average f . The black line in the left panel illustrates the $R_f = 1.41, 2.83, 5.66 \ h^{-1} \text{ Mpc}$ case, and the red line shows the result of adding $R_f = 11.3 \ h^{-1} \text{ Mpc}$. The blue line represents the case of $R_f = 1, 2.83, 8 \ h^{-1} \text{ Mpc}$. The magenta and green lines in the right panel show $R_f = 2, 2.83, 4 \ h^{-1} \text{ Mpc}$ and $R_f = 1.41, 2, 2.83, 4, 5.66 \ h^{-1} \text{ Mpc}$, respectively. Coarse-spaced smoothing scales in general result in much better fit to the halo power spectrum, while fine-spaced ones lead to 2–3% underprediction of the low- k power owing to the highly correlated features making the NN overfits. Adding in $11.3 \ h^{-1} \text{ Mpc}$ without filtering the residuals in k -space gives significantly underestimated low- k power.

Another possible improvement to make the model less sensitive to the input features is to implement other machine learning methods, such as random forest. This is, however, non-trivial to work out because we predict f based on a loss function that depends on the sum of f values in each grid cell. We do not know the true f 's a priori. Regression with gradient boosted trees or random forest instead requires inputting true f values. We leave it for future work to examine the possibility of implementing these other machine learning methods.

Compared to the coarse-spaced R_f 's, which match well the power spectrum, using fine-spaced R_f 's yields 2% underestimation of the power at $k < 0.1 \ h \text{ Mpc}^{-1}$. We find that the real-space squared loss is indistinguishable between the two. Since the features are more correlated when the R_f 's are closer, the orthogonalization and normalization of the original feature space greatly stretches the directions in the feature space that have tiny eigenvalues. This likely leads to the NN overfitting, in the presence of redundant features. We note that we are using the power spectrum as an indicator of overfitting instead of the real-space loss because of the uncorrelatedness of the loss and the low- k power (appendix A). We will show below, however, that re-training with the first 2 principal components of ∇f can effectively reduce this redundancy and provide a better match of P_{model} to P_h , as long as the required R_f 's are present.

In summary, we find that the NN-predicted f is much smoother than the least-squares (binned) f , and that using features associated with coarse-spaced R_f 's leads to the best recovery of the halo power spectrum. Below we will examine the structure of f in the input feature space in more detail.

4 Reducing the dimensions of the input feature space

Our results above showed that expanding the number of coarse-spaced smoothing scales gives a better fit to the halo field. However, the NN likely overfits in the presence of highly correlated features owing to fine-spaced R_f 's. We now examine whether f can be expressed in terms of a smaller number of input features that constitute a subset of the original input feature space. In other words, we would like to find if a lower-dimensional subspace of the initial features, beyond the projection already employed, might be used.

As an example, if f depended only upon a linear combination of the input features, we would expect ∇f to show a unique preferred direction in the feature space, even if f itself was a very nonlinear function. We therefore explore the structure of f by computing the principal components (PCs) of ∇f . The PCs are given by the eigenvectors of the covariance matrix of ∇f . To calculate the covariance matrix, we randomly sample $N_{\text{sample}} = 10^6$ particles, perturb around their orthogonalized and normalized features θ_{input} , and calculate their ∇f . The resulting gradient of f forms a matrix of size $(2n_{R_f} + 1) \times N_{\text{sample}}$, where n_{R_f} is the number of smoothing scales. We multiply this ∇f matrix by its transpose and obtain the eigenvector matrix U of the resulting covariance matrix, where the columns of U are the eigenvectors. The transformation from the original $\delta, \nabla^2 \delta, \mathcal{G}_2$ features to the PCs of ∇f is thus

$$\theta_{\text{PC}} = U^T \theta_{\text{input}} = \underbrace{U^T P \Lambda^{-1/2} V^T}_M \theta_{\text{orig}}, \quad (4.1)$$

where the second equality follows from equation (2.4). Given a set of R_f 's, we compute the average f from the five trained NNs and the corresponding PCs of ∇f . We then examine the structure of f as a function of θ_{PC} .

We use the 3 R_f case as an example, since we find that the behavior of f in the PC space is very similar between the sets of R_f 's that we have tested (see table. 1). Figure 5 shows the visualization of the f function in the PC space. The top panels illustrate f in the 2D planes of the zeroth, first, second, and third PCs, which have the largest eigenvalues. Bottom panels presents f as a function of the zeroth PC, with the color-bars showing values of the first, second, and third PCs.

We find that most of the variation in f is carried in the zeroth PC, with the largest eigenvalue. Very approximately f is 0 at $\text{PC0} \lesssim -1$, and increases monotonically with PC0 at larger values. This behavior of f is very similar to the dependence of f on δ seen in Paper I, thus making PC0 analogous to the overdensity. Intuitively, halos form in regions with high initial overdensity. The higher the halo mass, the larger the required δ . Since we fit the mass-weighted halo field with a mass threshold, this leads to a monotonically increasing f with δ , or PC0.

Beyond PC0, additional trend of variation in f can be well explained by PC1, and the dependence of f on the rest of the PCs is much weaker. We find that regardless of the smoothing scales used for training, the eigenvalue of the zeroth PC is a factor of 3 and 4 larger than those of the first and the second PCs respectively, but the variation of f with PC2 at a fixed PC0 is much weaker than that with PC1. The rest of the PCs have rather flat amplitudes of the eigenvalues. We thus examine whether f can be recovered by only two PCs by re-training the NN with the first two rows of θ_{PC} that correspond to PC0 and PC1 of ∇f .

Figure 6 show $P_{\text{model}}/P_{\text{h}}$ after the re-training with PC0 and PC1. Black and red colors represent the original training results and the re-trained results, respectively. Shades represent the minimum and maximum ranges obtained from 5 times of training, and lines show the results

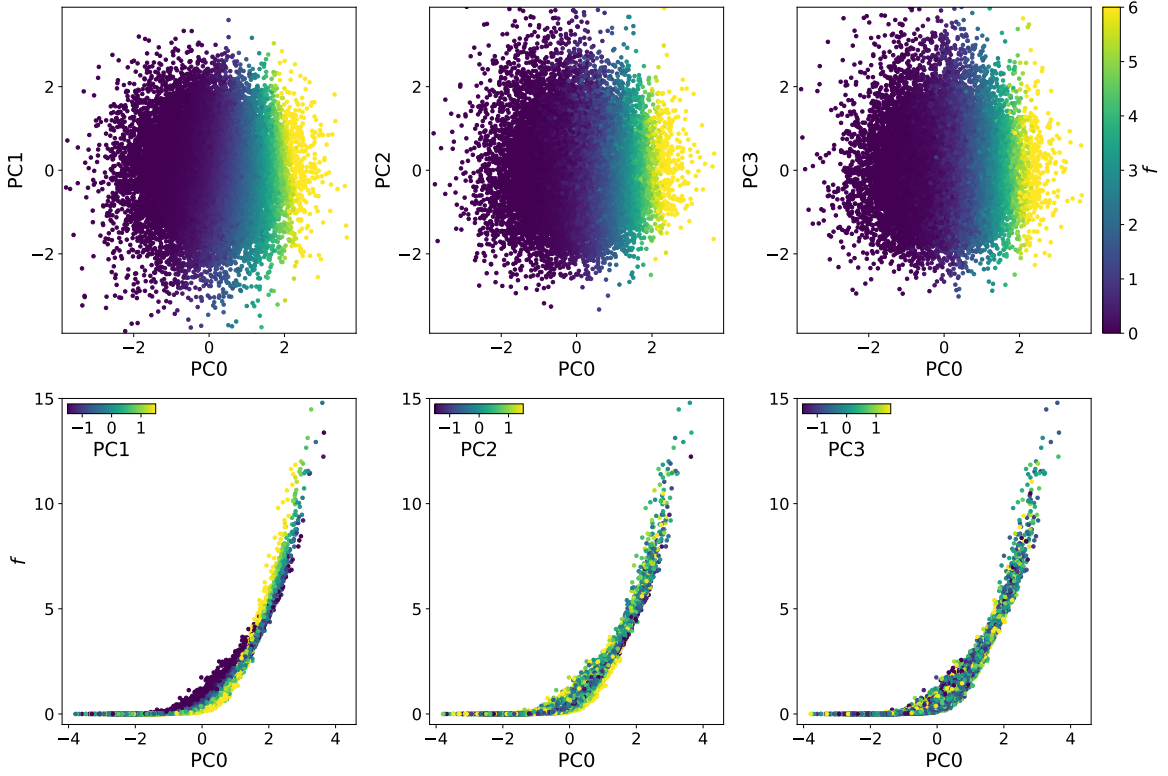


Figure 5. Visualization of the f function, obtained by training with $R_f = 1.41, 2.83, 5.66 h^{-1} \text{ Mpc}$ in the basis expanded by the principal components (PC) of ∇f . Top panels: f in the 2D planes of the zeroth, first, second, and third PCs, which have the largest eigenvalues. Bottom panels: f as a function of the zeroth PC, with the color-bars showing values of the first, second, and third PCs. Most of the structure of f is contained in the zeroth PC which has the largest eigenvalue, and the dependence of f beyond the first PC becomes rather weak.

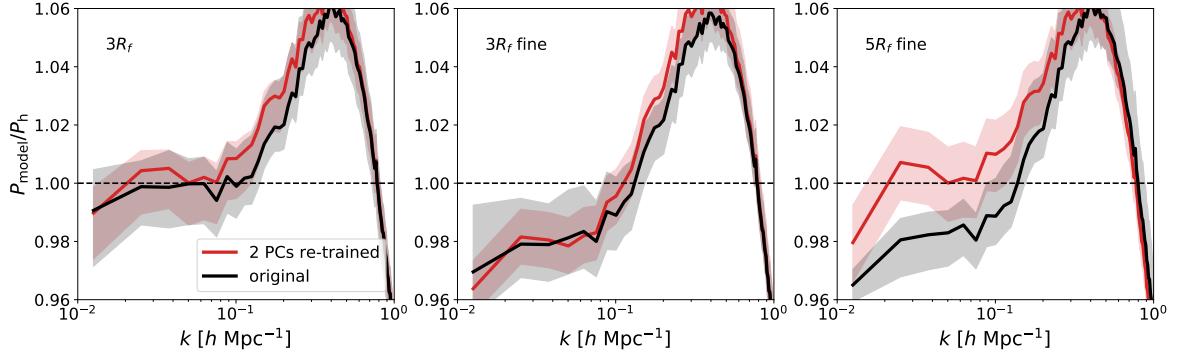


Figure 6. Results of re-training the NN after projecting the orthogonalized features onto the directions of the first two PCs of ∇f . Black and red represent the ratio of the model power spectra to the halo power spectra from the original training results and the re-trained results, respectively. Shades represent the minimum and maximum ranges obtained from 5 times of training, and lines show the results obtained from the average f . From left to right we show cases of different combinations of R_f 's. Re-training with only 2 PCs either recovers the original training result or outperform it owing to the reduction of the redundancy in the input feature space.

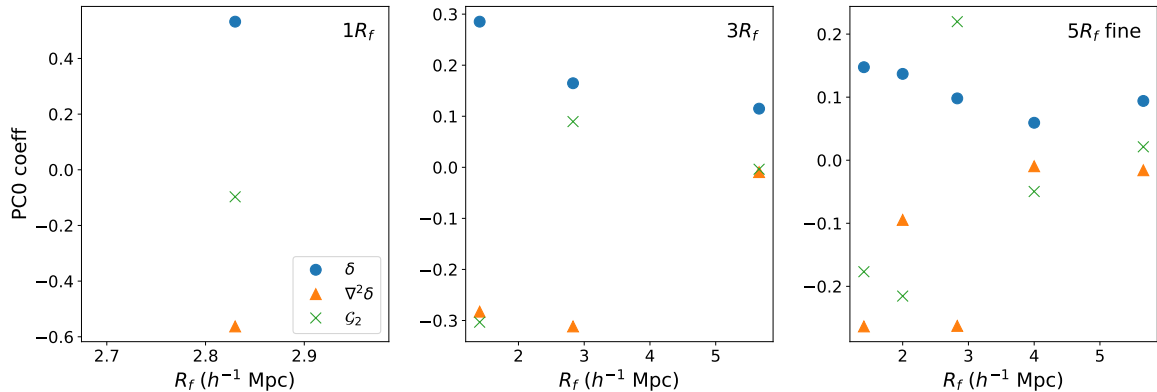


Figure 7. Coefficients of the zeroth principal component of ∇f at corresponding values of R_f . From left to right, the NN is trained with $R_f = 2.83 \, h^{-1} \text{ Mpc}$, $R_f = 1.41, 2.83, 5.66 \, h^{-1} \text{ Mpc}$, $R_f = 1.41, 2, 2.83, 4, 5.66 \, h^{-1} \text{ Mpc}$, respectively. Blue dots, orange triangles, and green crosses represent $\delta, \nabla^2 \delta, \mathcal{G}_2$ respectively. There is hardly any specific structure in the coefficients, so it is not desirable to interpret the numbers with any physical meanings.

obtained from the average f . From left to right we show cases of different combinations of R_f 's: $R_f = 1.41, 2.83, 5.66 \, h^{-1} \text{ Mpc}$, $R_f = 2, 2.83, 4 \, h^{-1} \text{ Mpc}$, $R_f = 1.41, 2, 2.83, 4, 5.66 \, h^{-1} \text{ Mpc}$. Although not shown, we find that re-training with the first two PCs also recovers the original training and validation losses.

For $R_f = 1.41, 2.83, 5.66 \, h^{-1} \text{ Mpc}$ and $R_f = 2, 2.83, 4 \, h^{-1} \text{ Mpc}$, the re-training produces $P_{\text{model}}/P_h \approx 1$ and 0.98 at $k < 0.1 \, h \text{ Mpc}^{-1}$ respectively, the same as the original training results. This demonstrates that the first two PCs are adequate to characterize f . For $R_f = 1.41, 2, 2.83, 4, 5.66 \, h^{-1} \text{ Mpc}$, the re-trained results bring P_{model}/P_h close to 1, while this ratio is 0.98 in the original training. This indicates that by reducing the size of the input feature space with only the first two PCs of ∇f , the NN no longer overfits owing to the redundancy in the highly correlated features. For the case of $R_f = 1.41, 2.83, 5.66, 11.3 \, h^{-1} \text{ Mpc}$, although not shown here, we find that the re-training raises the low- k power by 3% and brings P_{model}/P_h to 0.98 at low- k . The re-training thus improves the match to the power spectrum, in the absence of filtering the residuals.

We now examine whether the elements of the transformation matrix M , which we term the “coefficients of the PCs”, have any physical implications. Intuitively, these numbers might be interpreted as the bias coefficients. Figure 7 shows coefficients of PC0 of ∇f at corresponding R_f 's. From left to right, the NN is trained with $R_f = 2.83 \, h^{-1} \text{ Mpc}$, $R_f = 1.41, 2.83, 5.66 \, h^{-1} \text{ Mpc}$, $R_f = 1.41, 2, 2.83, 4, 5.66 \, h^{-1} \text{ Mpc}$, respectively. Blue dots, orange triangles, and green crosses represent the coefficients of $\delta, \nabla^2 \delta, \mathcal{G}_2$ respectively.

The one smoothing scale case produces a close-to-zero coefficient for \mathcal{G}_2 . This confirms our previous findings in Paper I that \mathcal{G}_2 is not as important as $\nabla^2 \delta$ in recovering the halo field with a cut at $M > 3 \times 10^{11} \, M_\odot$. Previous works also find that either the tidal bias is only important for halos with $M \gtrsim 10^{13} \, h^{-1} \, M_\odot$ [8], or there is a small negative shear bias regardless of halo mass [22, 44]. When including more smoothing scales, the coefficients of \mathcal{G}_2 jump up and down around zero and can have larger amplitudes than the coefficients of δ and $\nabla^2 \delta$. This is partially caused by the orthogonalization and normalization of the original features introducing small values in Λ , which when inverted and multiplied with U enlarges the corresponding U elements (equation (4.1)). In fact, the elements in U that are associated

with the orthogonalized \mathcal{G}_2 's have a factor of 2–3 lower amplitude than those associated with the orthogonalized δ and $\nabla^2\delta$. Moreover, since the \mathcal{G}_2 's are correlated, the jumping above and below zero of the \mathcal{G}_2 coefficients reflects the cancellation of the effects of the \mathcal{G}_2 terms. Our results with multiple smoothing scales thus are still consistent with \mathcal{G}_2 playing a minor role in determining the halo field with $M > 3 \times 10^{11} h^{-1} M_\odot$.

Comparing the $3R_f$ case and $5R_f$ one, the coefficients of δ and $\nabla^2\delta$ are similar at the R_f values of common. The addition of two intermediate smoothing scales brings in values that continues the spectrum of the coefficients of δ , but the spectrum of the coefficients of $\nabla^2\delta$ seems less regular. We find similar trends when examining the case of $R_f = 1.41, 2.83, 5.66, 11.3 h^{-1} \text{ Mpc}$ and $R_f = 1, 2.83, 8 h^{-1} \text{ Mpc}$. It is thus unclear whether these coefficients can be interpreted with any physical meaning. It might be possible to derive the spectrum of coefficients using the peak-patch formalism or the extended Press-Schechter theory, but such an exploration is beyond the scope of our paper. We leave it for a future work to examine this in detail.

In summary, we find that f can be well described by two directions in the original features space, and that training with these two principal components prevents the NN from overfitting. While previous works using the bias expansion approach to model the halo field only use one smoothing scale [e.g. 8, 20–23, 45], our results indicate that a linear combination of the features of multiple smoothing scales may lead to a better fit of the halo field.

5 Modeling a thin mass range

Having set up all the necessary tools for fitting f and examining its structure, we now briefly explore fitting the mass-weighted halo field with a thin mass range instead of a mass threshold. Intuitively, the larger the halo mass, the more it requires the involvement of higher δ regions. To obtain halos within a thin mass range, regions with too high δ thus cannot participate. As a consequence, instead of monotonically increasing with δ , as in the case of using mass thresholds, f will rise with δ but eventually drop to zero [3]. We also expect the peak of f to shift towards higher δ for higher-mass bins.

To test these intuitions, we train NNs on the mass-weighted halo fields with $3 \times 10^{11} M_\odot < M < 10^{12} M_\odot$ and $2 \times 10^{12} M_\odot < M < 10^{13} M_\odot$. The mass-weighted halo masses are $5.2 \times 10^{11} M_\odot$ and $6.2 \times 10^{12} M_\odot$, corresponding to Gaussian filters with $R_f = 0.8 h^{-1} \text{ Mpc}$ and $1.9 h^{-1} \text{ Mpc}$, respectively. Since our initial density field is created on a grid with cell size of $0.87 h^{-1} \text{ Mpc}$, we only produce smoothed density fields with $R_f \geq 1 h^{-1} \text{ Mpc}$. However, we expect the case of fitting a thin mass bin to require more strongly the involvement of multiple smoothing scales. Intuitively, the overdensity smoothed at the halo scale R_h should be large in order to form halos within a thin mass range, but the overdensity smoothed at $R_f \gg R_h$ needs to be small enough to not collapse to bigger objects. The NN therefore needs intake of features at a wide range of R_f 's. We thus train the NN with the $\delta, \nabla^2\delta, \mathcal{G}_2$'s at $R_f = 1, 2, 4 h^{-1} \text{ Mpc}$, which correspond to typical masses $M_h = 1.4 \times 10^{12}, 1.1 \times 10^{13}, 8.7 \times 10^{13} M_\odot$. As above, we perform five times of training on one training simulation with different initializations and obtain the average f .

We calculate the PCs of ∇f and examine f as a function of the PCs. Figure 8 shows f of the lower (left) and higher (right) mass range halos. The top panels illustrate f in the 2D planes of the zeroth and first PCs. The bottom panels present f as a function of the zeroth PC, with the color-bars showing values of the first PC.

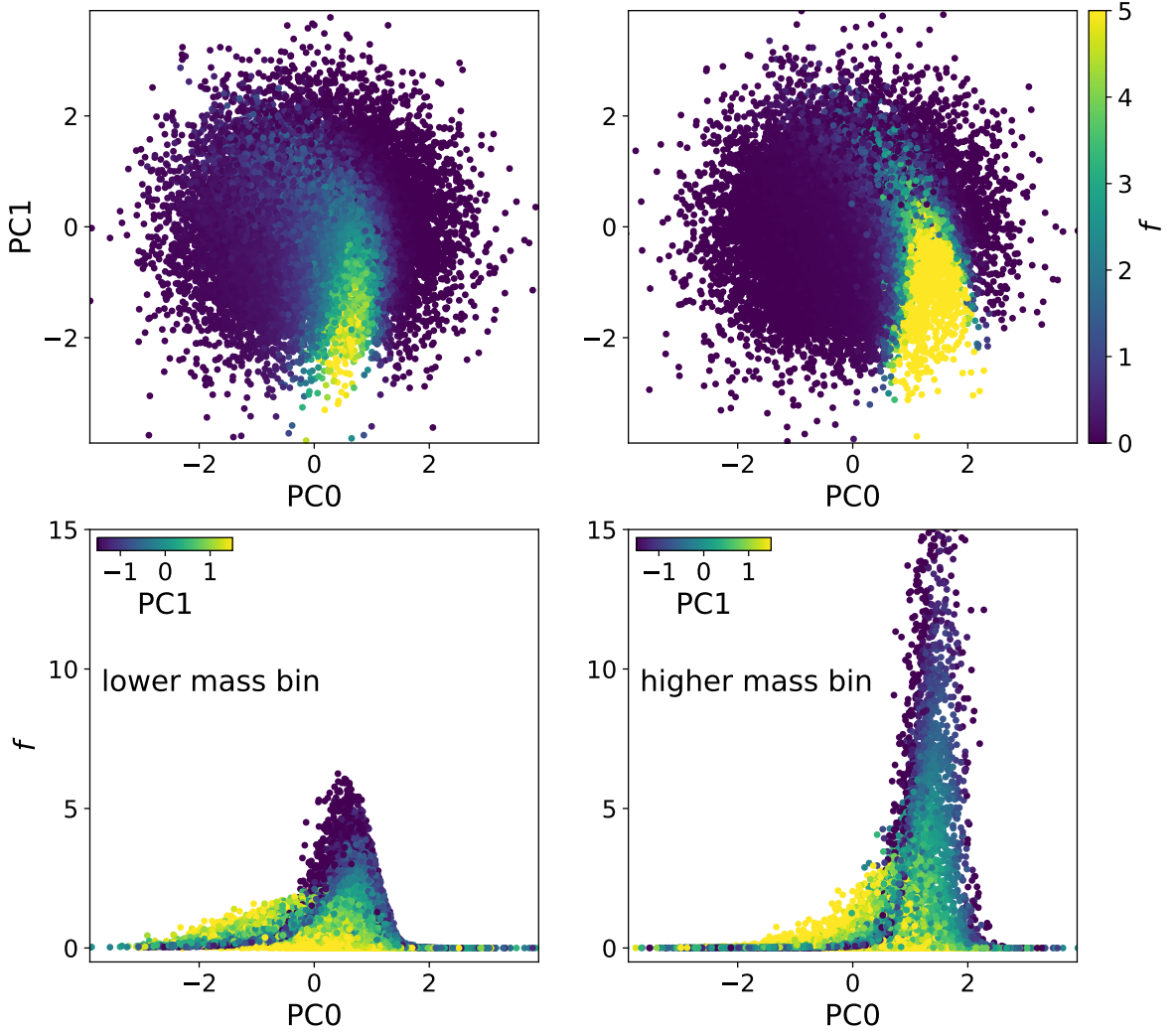


Figure 8. Visualization of the f function on two thin mass ranges, defined by $3 \times 10^{11} M_{\odot} < M < 10^{12} M_{\odot}$ (left) and $2 \times 10^{12} M_{\odot} < M < 10^{13} M_{\odot}$ (right), obtained by training with $R_f = 1, 2, 4 h^{-1} \text{ Mpc}$. Top panels: f in the 2D planes of the zeroth and first PCs. Bottom panels: f as a function of the zeroth PC, with the color-bars showing values of the first PC. The structure of f in the case of a thin mass range is substantially different from training with a mass threshold. The peak of f shifts to higher values of PC0 as the halos become more massive, as PC0 is analogous to the overdensity.

As expected, f in the case of a thin mass range rises and drops with increasing values of PC0, substantially different from the case of a mass threshold where f increases monotonically with PC0. The peak of f occurs at $\text{PC0} \sim 0.5$ for the lower-mass halos, but at $\text{PC0} \sim 1.5$ for the higher-mass ones. As PC0 is analogous to the overdensity, this matches our intuition that higher overdensities contribute more to the formation of more massive halos. f also exhibits a tilt in the PC0-PC1 plane, compared to its more regular structure in the case of mass threshold (figure 5). These complex structures of f demonstrate the necessity of using NNs to explore halo formation in a large parameter space. We will examine the training on thin-mass-range halos in detail in a future work, including the performance of NNs on matching the halo power spectrum.

6 Conclusions

In this work we train neural nets to obtain a non-parametric Lagrangian model of galaxy clustering bias, expanding our previous formalism that uses least-square fitting on a binned input feature space. We measure the halo-to-mass ratios f for mass-weighted halos in N-body simulations, assuming f depends on the linear overdensity δ , the tidal operator \mathcal{G}_2 , and a non-local term $\nabla^2\delta$, all of which are smoothed over multiple smoothing scales. We find that for mass-weighted halos above a mass threshold, using 3 coarsely separated smoothing scales gives a much better recovery of the halo power spectrum than 1 smoothing scale that corresponds to the halo mass. Adding more smoothing scales or using fine-spaced ones leads to overfitting and thus a 2–5% underestimation of the low- k power. By calculating the principal components (PC) of ∇f , we find that f can be well described by the first two PCs of ∇f and that f is a monotonically increasing function of the PC with the largest eigenvalue (PC0). Re-training the NN with these two PCs either recovers the original training results or outperforms it by better matching the halo power spectrum, indicating that they prevent the NN from overfitting. The coefficients of these linear combinations may be interpreted as the bias in the case of multiple smoothing scales, but a detailed examination is beyond the scope of our paper.

We briefly explored fitting the mass-weighted halo field over a thin mass range instead of a mass threshold, and find that f rises and drops with PC0 instead of being monotonically increasing. This matches our physical intuition that f should only be non-zero at a certain range of overdensities in the case of a thin mass range. The complex structure of f in the PC space demonstrates the usefulness of using a NN to examine structure formation.

As we find that the real-space squared loss is not correlated with the recovery of the power spectrum, one way to improve our formalism is to express the loss function in Fourier space and filtering out high- k residuals. Producing and fitting the halo fields using Cloud-in-Cell interpolation with varying kernel size instead of nearest neighbor interpolation may have a similar effect as filtering the high- k residuals and thus can also help alleviate the difficulty of matching the low- k power. We will also extend our code to perform the fitting in $2h^{-1}$ Gpc large box simulations, which may largely reduce the 1–2% scatter in the model power spectrum.

Despite all these future improvements, we have demonstrated the ability of using a NN to predict complex-structured f in a high-dimensional input feature space, and that multiple smoothing scales are needed to fully capture the halo field. This is especially true in the thin mass range case. We will explore in a future work the use of number-weighted halos or a halo occupation distribution model [46–48], which should also require complicated structures of f . Another interesting direction of future work is to examine f of halos at higher redshifts, as these are more biased than the $z = 0.5$ halos we study here. These insights might transform the way we analyze observational data from galaxy surveys, especially those targeting higher redshifts and larger halo masses.

Acknowledgments

JBM is supported by a Clay fellowship at the Smithsonian Astrophysical Observatory. DJE is supported by U.S. Department of Energy grant, now DE-SC0007881, by the National Science Foundation under Cooperative Agreement PHY-2019786 (the NSF AI Institute for Artificial Intelligence and Fundamental Interactions, <http://iaifi.org/>), and as a Simons Foundation Investigator.

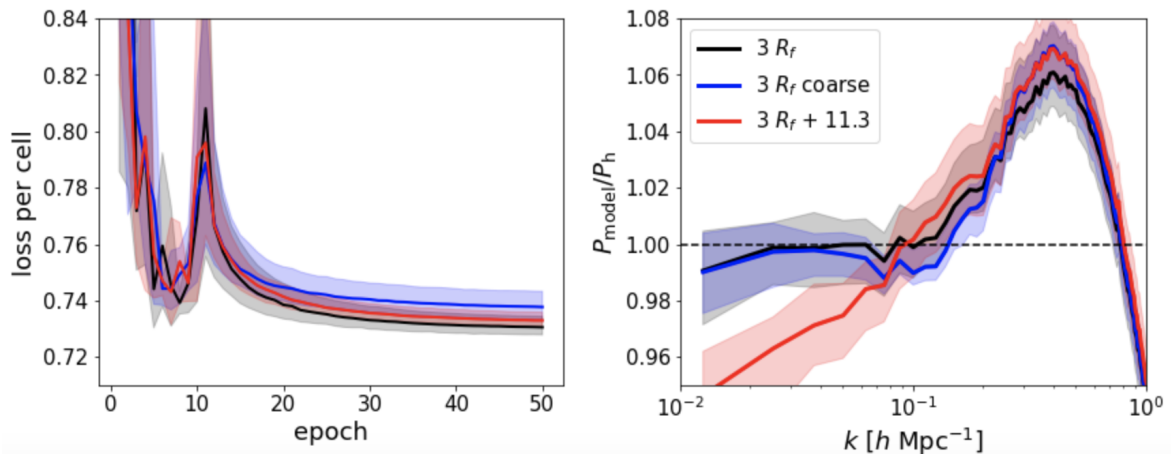


Figure 9. Average loss per cell as a function of epoch (left panel) and the ratio of the model to true halo power spectrum (right panel), evaluated for one validation simulation using different smoothing scales. Line colors are the same as in figure 4.

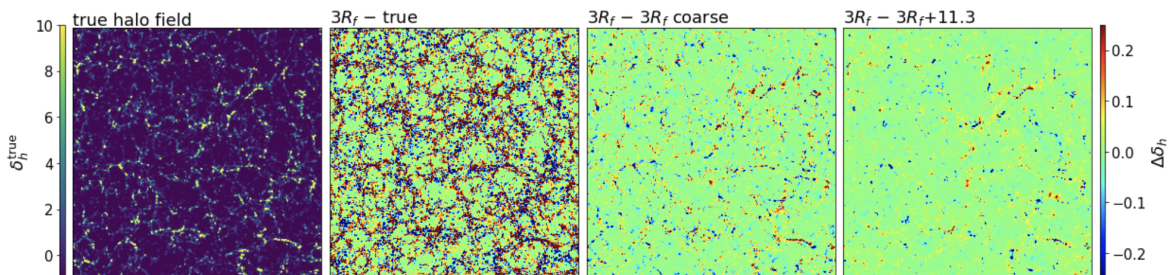


Figure 10. The true halo field (leftmost panel) and the differences in the predicted halo fields using different R_f 's. The slices are $10 h^{-1}$ Mpc thick. The second column shows the model halo field in the $3R_f$ case minus the true halo field. The third and fourth columns illustrate the model halo field with $3R_f$ coarse and $3R_f + 11.3$ subtracted by the $3R_f$ case, respectively.

A Loss as a function of epoch

Here we show that the match between the predicted and true halo power spectrum is not strongly correlated with the real-space loss of our NN.

The left panel of figure 9 shows the average loss per cell as a function of epoch, for the same validation simulation as in figure 4. Line colors represent the same combinations of R_f 's as figure 4. The losses differ at sub-percent level at the 50th epoch which we use to calculate the halo power spectrum. The “ $3R_f + 11.3$ ” case even has a slightly lower average loss than “ $3R_f$ coarse”. However, “ $3R_f + 11.3$ ” produces a halo power spectrum that undershoot the true power spectrum by $\sim 5\%$ at the lowest k , while “ $3R_f$ coarse” provides a fair match of P_{model} to P_h (right panel). The value of the real-space loss is thus not a good indicator of the behavior of the halo power spectrum in the case of multiple smoothing scales. This may be due to our implementation of the loss in this first work.

Figure 10 illustrates the true halo field and the differences in the predicted halo fields, for $3R_f$, $3R_f$ coarse, and $3R_f + 11.3$. The slices are $10 h^{-1}$ Mpc thick. The second column, showing the $3R_f$ halo field minus the true halo field, indicates that scatter around the input

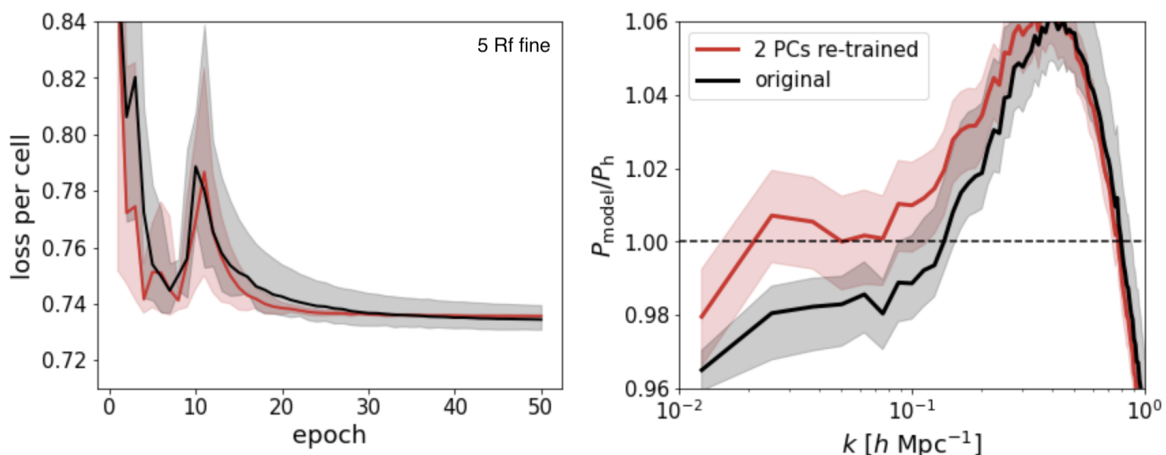


Figure 11. Average loss per cell as a function of epoch (left panel) and the ratio of the model to true halo power spectrum (right panel), evaluated for one validation simulation in the case of “ $5R_f$ fine” and retraining with 2 PCs. Line colors are the same as in figure 6.

halo field is roughly uniform at each local overdensity. The $3R_f$ coarse halo field also appears to scatter rather randomly around the $3R_f$ halo field at a given local overdensity, as illustrated by the third column. These seem consistent with the $3R_f$ and $3R_f$ coarse halo power spectra matching the true power spectrum well at low- k . Since the $3R_f + 11.3$ power spectrum underpredicts the low- k power, one might expect its real-space halo field to undershoot the local overdensities. However, the scatter around the $3R_f$ halo field does not show any particular pattern. This again demonstrates an uncorrelatedness between the real-space loss and the power spectrum.

This uncorrelatedness between the real-space loss and the low- k power is also reflected when retraining with the PCs. Figure 11 illustrates the loss and the halo power spectra in the case of “ $5R_f$ fine” and retraining with 2 PCs. Line colors are the same as in figure 6. While the original and the retraining losses are nearly indistinguishable at the 50th epoch, the model power spectra differ by 2% at $k < 0.1 h \text{ Mpc}^{-1}$. We will revisit the issue of linking the loss to the relevant observables (e.g., power spectrum) in future work.

References

- [1] V. Desjacques, D. Jeong and F. Schmidt, *Large-Scale Galaxy Bias*, *Phys. Rept.* **733** (2018) 1 [[arXiv:1611.09787](#)] [[INSPIRE](#)].
- [2] J.N. Fry and E. Gaztañaga, *Biasing and hierarchical statistics in large scale structure*, *Astrophys. J.* **413** (1993) 447 [[astro-ph/9302009](#)] [[INSPIRE](#)].
- [3] T. Matsubara, *Nonlinear perturbation theory with halo bias and redshift-space distortions via the Lagrangian picture*, *Phys. Rev. D* **78** (2008) 083519 [Erratum *ibid.* **78** (2008) 109901] [[arXiv:0807.1733](#)] [[INSPIRE](#)].
- [4] Z. Vlah, E. Castorina and M. White, *The Gaussian streaming model and convolution Lagrangian effective field theory*, *JCAP* **12** (2016) 007 [[arXiv:1609.02908](#)] [[INSPIRE](#)].
- [5] K.C. Chan, R. Scoccimarro and R.K. Sheth, *Gravity and Large-Scale Non-local Bias*, *Phys. Rev. D* **85** (2012) 083509 [[arXiv:1201.3614](#)] [[INSPIRE](#)].
- [6] T. Baldauf, U. Seljak, V. Desjacques and P. McDonald, *Evidence for Quadratic Tidal Tensor Bias from the Halo Bispectrum*, *Phys. Rev. D* **86** (2012) 083540 [[arXiv:1201.4827](#)] [[INSPIRE](#)].

- [7] S. Saito et al., *Understanding higher-order nonlocal halo bias at large scales by combining the power spectrum with the bispectrum*, *Phys. Rev. D* **90** (2014) 123522 [[arXiv:1405.1447](#)] [[INSPIRE](#)].
- [8] M.M. Abidi and T. Baldauf, *Cubic Halo Bias in Eulerian and Lagrangian Space*, *JCAP* **07** (2018) 029 [[arXiv:1802.07622](#)] [[INSPIRE](#)].
- [9] T. Fujita et al., *Very Massive Tracers and Higher Derivative Biases*, *JCAP* **01** (2020) 009 [[arXiv:1609.00717](#)] [[INSPIRE](#)].
- [10] C. Modi, S.-F. Chen and M. White, *Simulations and symmetries*, *Mon. Not. Roy. Astron. Soc.* **492** (2020) 5754 [[arXiv:1910.07097](#)] [[INSPIRE](#)].
- [11] N. Kokron et al., *The cosmology dependence of galaxy clustering and lensing from a hybrid N -body-perturbation theory model*, *Mon. Not. Roy. Astron. Soc.* **505** (2021) 1422 [[arXiv:2101.11014](#)] [[INSPIRE](#)].
- [12] M. Schmittfull, M. Simonović, V. Assassi and M. Zaldarriaga, *Modeling Biased Tracers at the Field Level*, *Phys. Rev. D* **100** (2019) 043514 [[arXiv:1811.10640](#)] [[INSPIRE](#)].
- [13] M. Schmittfull et al., *Modeling Galaxies in Redshift Space at the Field Level*, *JCAP* **05** (2021) 059 [[arXiv:2012.03334](#)] [[INSPIRE](#)].
- [14] C. Modi, M. White, A. Slosar and E. Castorina, *Reconstructing large-scale structure with neutral hydrogen surveys*, *JCAP* **11** (2019) 023 [[arXiv:1907.02330](#)] [[INSPIRE](#)].
- [15] A. Barreira, T. Lazeyras and F. Schmidt, *Galaxy bias from forward models: linear and second-order bias of IllustrisTNG galaxies*, *JCAP* **08** (2021) 029 [[arXiv:2105.02876](#)] [[INSPIRE](#)].
- [16] N. Kaiser, *On the Spatial correlations of Abell clusters*, *Astrophys. J. Lett.* **284** (1984) L9 [[INSPIRE](#)].
- [17] V. Desjacques, M. Crocce, R. Scoccimarro and R.K. Sheth, *Modeling scale-dependent bias on the baryonic acoustic scale with the statistics of peaks of Gaussian random fields*, *Phys. Rev. D* **82** (2010) 103529 [[arXiv:1009.3449](#)] [[INSPIRE](#)].
- [18] M. Musso, A. Paranjape and R.K. Sheth, *Scale dependent halo bias in the excursion set approach*, *Mon. Not. Roy. Astron. Soc.* **427** (2012) 3145 [[arXiv:1205.3401](#)] [[INSPIRE](#)].
- [19] T. Baldauf, V. Desjacques and U. Seljak, *Velocity bias in the distribution of dark matter halos*, *Phys. Rev. D* **92** (2015) 123507 [[arXiv:1405.5885](#)] [[INSPIRE](#)].
- [20] C. Modi, E. Castorina and U. Seljak, *Halo bias in Lagrangian Space: Estimators and theoretical predictions*, *Mon. Not. Roy. Astron. Soc.* **472** (2017) 3959 [[arXiv:1612.01621](#)] [[INSPIRE](#)].
- [21] T. Lazeyras, C. Wagner, T. Baldauf and F. Schmidt, *Precision measurement of the local bias of dark matter halos*, *JCAP* **02** (2016) 018 [[arXiv:1511.01096](#)] [[INSPIRE](#)].
- [22] T. Lazeyras and F. Schmidt, *Beyond LIMD bias: a measurement of the complete set of third-order halo bias parameters*, *JCAP* **09** (2018) 008 [[arXiv:1712.07531](#)] [[INSPIRE](#)].
- [23] T. Lazeyras and F. Schmidt, *A robust measurement of the first higher-derivative bias of dark matter halos*, *JCAP* **11** (2019) 041 [[arXiv:1904.11294](#)] [[INSPIRE](#)].
- [24] T. Lazeyras, A. Barreira and F. Schmidt, *Assembly bias in quadratic bias parameters of dark matter halos from forward modeling*, *JCAP* **10** (2021) 063 [[arXiv:2106.14713](#)] [[INSPIRE](#)].
- [25] X. Wu, J.B. Muñoz and D. Eisenstein, *A fully Lagrangian, non-parametric bias model for dark matter halos*, *JCAP* **02** (2022) 002 [[arXiv:2109.13948](#)] [[INSPIRE](#)].
- [26] P. McDonald and A. Roy, *Clustering of dark matter tracers: generalizing bias for the coming era of precision LSS*, *JCAP* **08** (2009) 020 [[arXiv:0902.0991](#)] [[INSPIRE](#)].
- [27] V. Assassi, D. Baumann, D. Green and M. Zaldarriaga, *Renormalized Halo Bias*, *JCAP* **08** (2014) 056 [[arXiv:1402.5916](#)] [[INSPIRE](#)].

- [28] J.M. Bardeen, J.R. Bond, N. Kaiser and A.S. Szalay, *The Statistics of Peaks of Gaussian Random Fields*, *Astrophys. J.* **304** (1986) 15 [[INSPIRE](#)].
- [29] H.J. Mo and S.D.M. White, *An Analytic model for the spatial clustering of dark matter halos*, *Mon. Not. Roy. Astron. Soc.* **282** (1996) 347 [[astro-ph/9512127](#)] [[INSPIRE](#)].
- [30] R.K. Sheth and G. Tormen, *Large scale bias and the peak background split*, *Mon. Not. Roy. Astron. Soc.* **308** (1999) 119 [[astro-ph/9901122](#)] [[INSPIRE](#)].
- [31] L. Lucie-Smith, H.V. Peiris, A. Pontzen and M. Lochner, *Machine learning cosmological structure formation*, *Mon. Not. Roy. Astron. Soc.* **479** (2018) 3405 [[arXiv:1802.04271](#)] [[INSPIRE](#)].
- [32] L. Lucie-Smith, H.V. Peiris and A. Pontzen, *An interpretable machine learning framework for dark matter halo formation*, *Mon. Not. Roy. Astron. Soc.* **490** (2019) 331 [[arXiv:1906.06339](#)] [[INSPIRE](#)].
- [33] L. Lucie-Smith et al., *Deep learning insights into cosmological structure formation*, [arXiv:2011.10577](#) [[INSPIRE](#)].
- [34] N.A. Maksimova et al., *AbacusSummit: a massive set of high-accuracy, high-resolution N-body simulations*, *Mon. Not. Roy. Astron. Soc.* **508** (2021) 4017 [[arXiv:2110.11398](#)] [[INSPIRE](#)].
- [35] L.H. Garrison et al., *The Abacus Cosmos: A Suite of Cosmological N-body Simulations*, *Astrophys. J. Suppl.* **236** (2018) 43 [[arXiv:1712.05768](#)] [[INSPIRE](#)].
- [36] L.H. Garrison, D.J. Eisenstein and P.A. Pinto, *A High-Fidelity Realization of the Euclid Code Comparison N-body Simulation with Abacus*, *Mon. Not. Roy. Astron. Soc.* **485** (2019) 3370 [[arXiv:1810.02916](#)] [[INSPIRE](#)].
- [37] L.H. Garrison et al., *The abacus cosmological N-body code*, *Mon. Not. Roy. Astron. Soc.* **508** (2021) 575 [[arXiv:2110.11392](#)] [[INSPIRE](#)].
- [38] M.V.L. Metchnik, *A fast N-body scheme for computational cosmology*, Ph.D. thesis, The University of Arizona, U.S.A. (2009).
- [39] DESI collaboration, *The DESI Experiment Part I: Science, Targeting, and Survey Design*, [arXiv:1611.00036](#) [[INSPIRE](#)].
- [40] PLANCK collaboration, *Planck 2018 results. VI. Cosmological parameters*, *Astron. Astrophys.* **641** (2020) A6 [*Erratum ibid.* **652** (2021) C4] [[arXiv:1807.06209](#)] [[INSPIRE](#)].
- [41] B. Hadzhiyska et al., *compaso: A new halo finder for competitive assignment to spherical overdensities*, *Mon. Not. Roy. Astron. Soc.* **509** (2021) 501 [[arXiv:2110.11408](#)] [[INSPIRE](#)].
- [42] L.H. Garrison et al., *Improving Initial Conditions for Cosmological N-Body Simulations*, *Mon. Not. Roy. Astron. Soc.* **461** (2016) 4125 [[arXiv:1605.02333](#)] [[INSPIRE](#)].
- [43] Y.P. Jing, *Correcting for the alias effect when measuring the power spectrum using FFT*, *Astrophys. J.* **620** (2005) 559 [[astro-ph/0409240](#)] [[INSPIRE](#)].
- [44] J. Bel, K. Hoffmann and E. Gaztañaga, *Non-local bias contribution to third-order galaxy correlations*, *Mon. Not. Roy. Astron. Soc.* **453** (2015) 259 [[arXiv:1504.02074](#)] [[INSPIRE](#)].
- [45] M. Zennaro et al., *The BACCO simulation project: biased tracers in real space*, [arXiv:2101.12187](#) [[INSPIRE](#)].
- [46] S. Yuan, D.J. Eisenstein and L.H. Garrison, *Exploring the squeezed three-point galaxy correlation function with generalized halo occupation distribution models*, *Mon. Not. Roy. Astron. Soc.* **478** (2018) 2019 [[arXiv:1802.10115](#)] [[INSPIRE](#)].
- [47] B. Hadzhiyska et al., *Limitations to the ‘basic’ HOD model and beyond*, *Mon. Not. Roy. Astron. Soc.* **493** (2020) 5506 [[arXiv:1911.02610](#)] [[INSPIRE](#)].
- [48] B. Hadzhiyska, S. Bose, D. Eisenstein and L. Hernquist, *Extensions to models of the galaxy-halo connection*, *Mon. Not. Roy. Astron. Soc.* **501** (2021) 1603 [[arXiv:2008.04913](#)] [[INSPIRE](#)].