Machine Learning Subhalo Properties (Summary)

Abstract

In this work we first use machine learning algorithms to predict the number of subhalos with peak masses of at least 10^{11} $M_\odot$/h within a given host halo. Then we provide a procedure for predicting the spatial distribution and other properties of such subhalos. We then show, as an example, how this can be used to predict statistical quantities like subhalo two point correlation function at small scales.

1 Introduction and Data Splittings

For this project we have used a halo catalog at redshift zero based on the Small MultiDark Planck (SMDPL) N-body simulation [1] which consists of 3840^3 particles in a box of side length 400 Mpc/h. The halos are identified via Rockstar halo finder [2]. To ensure that the accuracy of our predictions are consistent, we select four different cubic boxes, each with a side length of 200 Mpc/h, to separately serve as test data sets for four different training implementations. We refer to these as box 1, box 3, box 5 and box 7. Each time, we consider the data related to halos within one of these boxes as the test set, while the data from halos in the remaining volume is used as the training set. These four boxes are defined as follows:

$$\begin{aligned}
\text{box 1} &: \ 0 \leq x \leq 200, \ \ 0 \leq y \leq 200, \ \ 0 \leq z \leq 200 \\
\text{box 3} &: \ 200 < x \leq 400, \ \ 200 < y \leq 400, \ \ 0 < z \leq 200 \\
\text{box 5} &: \ 0 < x \leq 200, \ \ 0 < y \leq 200, \ \ 200 < z \leq 400 \\
\text{box 7} &: \ 200 < x \leq 400, \ \ 200 < y \leq 400, \ \ 200 < z \leq 400
\end{aligned}$$

2 Machine learning the number count

Our first goal is to predict the number, Y, of subhalos with peak masses greater than or equal to 10^{11} $M_\odot$/h within each host halo with a given "features" vector $\mathbf{x}$. Denoting the number of subhalos inside each host halo in the training set as $Y_{\rm tr}$, the machine learning (ML) algorithms try to "learn" the relationship between $Y_{\rm tr}$ and $\mathbf{x}$, or $Y_{\rm tr}(\mathbf{x})$, by minimizing the loss function which can be chosen to be the mean squared error (MSE) between the predicted and actual subhalo counts ,$Y_{\rm tr,\ pred}$ and $Y_{\rm tr,\ act}$:

$$\text{MSE} = \frac{\sum_{i=1}^{N} \left(Y^i_{\rm tr,pred}(\mathbf{x}^i) - Y^i_{\rm tr,act}(\mathbf{x}^i) \right)^2}{N} \tag{1}$$

with i denoting the i'th host halo with features $\mathbf{x}^i$ in the training set and N being the total number of host halos in that set.

After testing several ML algorithms with different combinations of host halo features, we found that the following combination has the most predictive power:

$$\mathbf{x}_{\rm opt} = \left\{ M_{\rm vir},\ M_{\rm peak},\ V_{\rm max},\ V_{\rm peak},\ c,\ a_{1/2},\ \lambda \right\} \tag{2}$$

where $c = R_{\rm vir}/R_{\rm s}$ is the concentration, $a_{1/2}$ is the half mass scale and λ is the Bullock spin. For ML algorithms, we found that a combination of Random Forests Regressor (RFR) with 60 estimators (trees) combined with a single Decision Tree Regressor (DTR) performs relatively

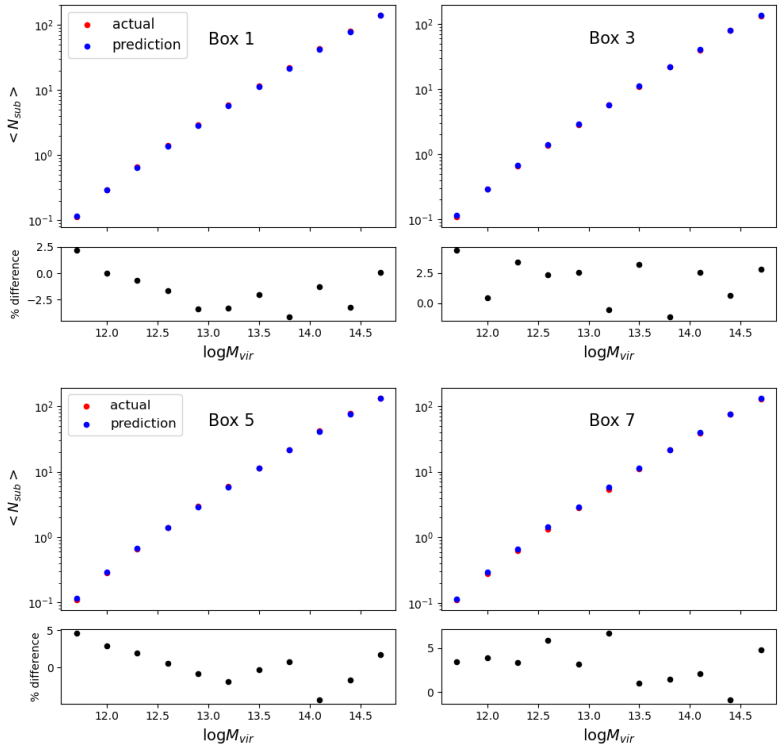

Figure 1: Average number of subhalos per host halo as a function of binned logarithm of virial masses. $M_{\rm vir}$ is in units of $\rm M_\odot/h$.

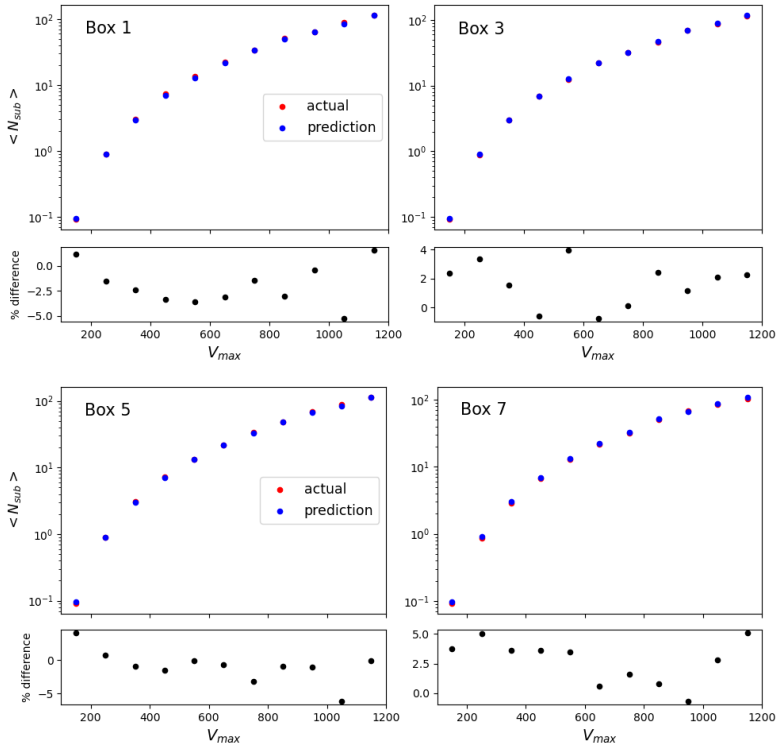

Figure 2: Average number of subhalos per host halo as a function of binned maximum circular velocity (km/s).

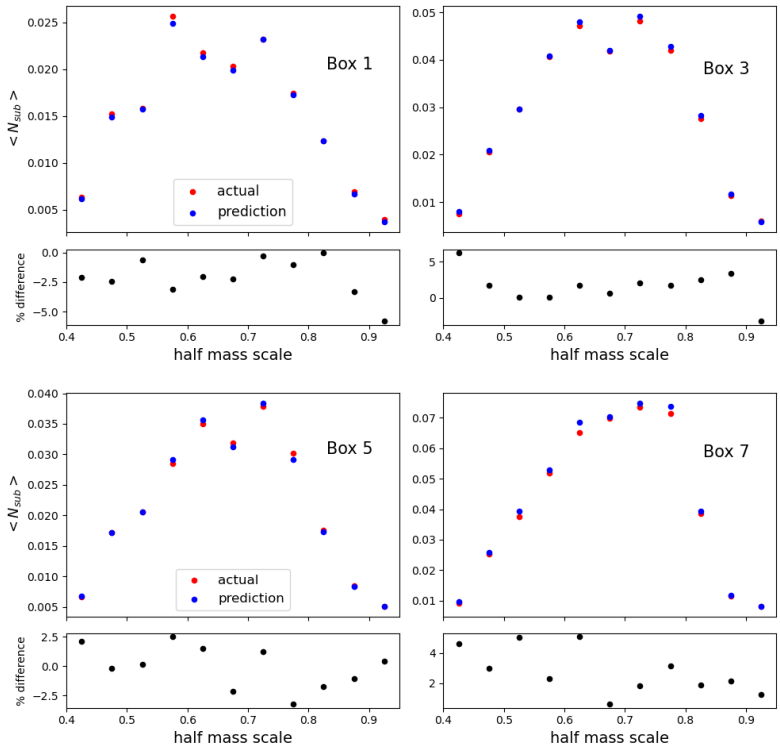

Figure 3: Average number of subhalos per host halo as a function of binned half-mass scales.

better. After fitting the regressors separately on the training data, we apply them to the test data for prediction. We then average the predicted number counts from each regressor to obtain the final prediction:

$$Y_{\rm pred}(\mathbf{x}_{\rm opt}) = \frac{Y_{\rm pred,\ RFR_{60}}(\mathbf{x}_{\rm opt}) + Y_{\rm pred,\ DTR}(\mathbf{x}_{\rm opt})}{2}\,. \tag{3}$$

We use the packages available in Scikit-learn library [3] to implement the ML algorithms needed for this work.

As mentioned in previous section, we test the accuracy of our predictions four times. Each time we take one of the aforementioned boxes as test set and train the regressors on the rest of the volume. Large panels in Figs 1 - 3 show the comparison between predicted and actual average number of subhalos within each host halo as a function of the binned values of log $M_{\rm vir}$, $V_{\rm max}$ and $a_{1/2}$ for different test sets (boxes). The lower panels in all figures show the percent difference, $((Y_{pred} - Y_{act})/Y_{act}) \times 100$. We can see that in all plots most of the errors are below 5%.

3 Predicting the Properties of Subhalos

After predicting the number counts, our next goal is to predict the properties of subhalos including their spatial distribution and velocities. We propose that an appropriate mapping of subhalos from certain host halos in the training set into the host halos in the test set can accomplish the goal. Let's assume that we have obtained the values of $Y_{\rm pred}(\mathbf{x}_{\rm opt})$ from Eq(3) for the test set. We define a target host halo in the test set as one with $Y_{\rm pred} \neq 0$. We use the superscript t to refer to these hosts, such as $Y^t_{\rm pred}$. For each target host with $Y^t_{\rm pred}$ number

of subhalos, we can find a set of host halos in the training set whose number of subhalos are equal to (or almost equal to) $Y^t_{\rm pred}$. Among these host halos, one can find a suitable one whose subhalos can be properly mapped into the target host halo in the test set. We call that suitable host halo, the "donor" host halo. To find the donor for each target host, we first need to define a set of "mapping" features, $\mathbf{x}_{\rm map}$, for both the target host halos and the host halos in the training set. These can be expressed as:

$$\mathbf{x}^t_{\rm map} = \{x^t_1, x^t_2, \ldots, x^t_n, \overbrace{Y^t_{\rm pred}, Y^t_{\rm pred}, \ldots, Y^t_{\rm pred}}^{K \text{ times}}\}, \tag{4}$$

$$\mathbf{x}^{\rm train}_{\rm map} = \{x^{\rm train}_1, x^{\rm train}_2, \ldots, x^{\rm train}_n, \overbrace{Y^{\rm train}, Y^{\rm train}, \ldots, Y^{\rm train}}^{K \text{ times}}\} \tag{5}$$

where x_i's can be features like virial mass, half mass scale, etc. The reason for repeating the Y values is to give the subhalo number count significantly more weight than other features.

The task is to find the host halo in the train set whose $\mathbf{x}^{\rm train}_{\rm map}$ has minimal Euclidean distance to $\mathbf{x}^t_{\rm map}$ for a given target host halo. Of course, we apply scalers, such as the StandardScaler, to the mapping features to avoid any bias due to large numbers. The challenge is to find a suitable combination of x_i's as well as the optimal value of K that results in the most accurate predictions (in statistical sense) for subhalo properties. Once the parameters are selected, the host halo in the training set whose $\mathbf{x}^{\rm train}_{\rm map}$ has the minimal Euclidean distance to $\mathbf{x}^t_{\rm map}$ will be identified as the donor host for the corresponding target host halo in the test set. The donor will have almost the same number of subhalos as the target host due to the higher weight of Y's as mentioned above.

Next, we will appropriately map the desired properties of the donor host's subhalos into the target host. For example, we can map positions and velocities by requiring that the original and mapped subhalos have similar relative positions and velocities with respect to their corresponding host halos. This results in:

$$\mathbf{r}^t_{n,\rm sub} = \mathbf{r}^{\rm d}_{n,\rm sub} + \mathbf{r}^t - \mathbf{r}^{\rm d} \tag{6}$$

$$\mathbf{v}^t_{n,\rm sub} = \mathbf{v}^{\rm d}_{n,\rm sub} + \mathbf{v}^t - \mathbf{v}^{\rm d} \tag{7}$$

where $\mathbf{r}^t_{n,\rm sub}/\mathbf{v}^t_{n,\rm sub}$ are the position/velocity vectors of the nth mapped subhalo into the target host, $\mathbf{r}^{\rm d}_{n,\rm sub}/\mathbf{v}^{\rm d}_{n,\rm sub}$ are the position/velocity vectors of the nth subhalo inside the donor host that is being mapped, $\mathbf{r}^t/\mathbf{v}^t$ are the position/velocity vectors of the target host and, similarly, $\mathbf{r}^{\rm d}/\mathbf{v}^{\rm d}$ are the position/velocity vectors of the donor host. Other proper relations can be used to map other subhalo features such as masses from donor to target hosts.

4 Two point correlation function of subhalos at small scales

Following the ideas of previous section, we can predict the spatial distribution of subhalos within a given host halo in each of the test boxes defined earlier. This enables us to compute the subhalo two point correlation function at scales below 1 Mpc/h. To determine whether a host halo in the test set is a target host or not, i.e. if it satisfies $Y_{\rm pred} \neq 0$, we only check if the rounded value of $Y_{\rm pred,\ RFR_{60}}$ in Eq(3) is nonzero or not. Once a host is identified as target, we use rounded values from the entire Eq(3) for $Y^t_{\rm pred}$'s in Eq(4). We then use Eqs (4) and (5) with the following features: $x_1 = M_{\rm vir}$, $x_2 = a_{1/2}$ (half mass scale) and $K = 4$. This combination ensures that the donor host has not only a similar number of subhalos but also

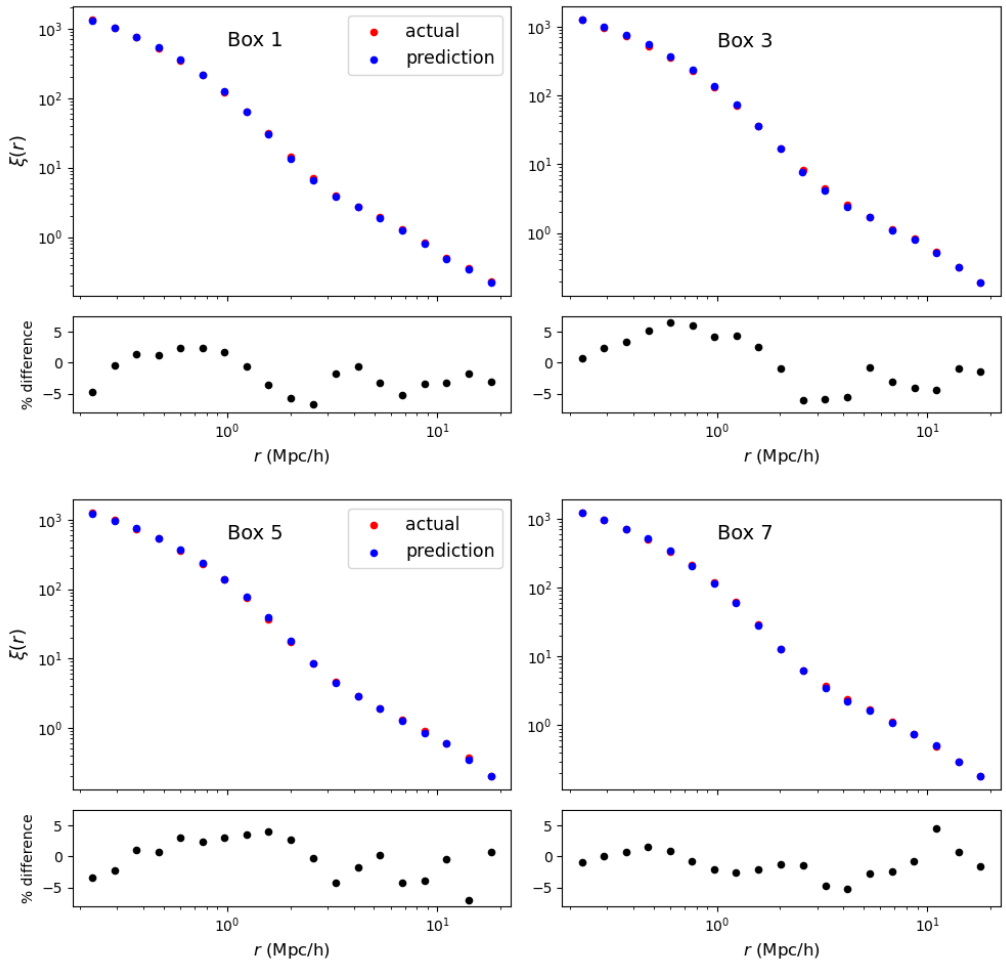

Figure 4: Two point correlation function of subhalos from 200 kpc/h to 20 Mpc/h in 20 logarithmic bins.

comparable mass and age which are among important features affecting the spatial distribution of subhalos. We found that this combination results in relatively accurate outcomes for two point correlation functions. For minimizing the Euclidian distance between each of the $\mathbf{x}^t_{\rm map}$'s and $\mathbf{x}^{\rm train}_{\rm map}$'s, we use a Scikit-learn based K-nearest neighbors algorithm properly adjusted for our purpose.

We have implemented this procedure for the four boxes, 1, 3, 5 and 7. Each time, one of these boxes served as the test set, while the remaining volume became the training set. Figure 4 shows the comparison between two-point correlation functions computed from the predicted and actual positions of subhalos in each test box. The computations are done using one of the Corrfunc [4] subpackages, Corrfunc.theory.xi, which assumes periodic boundary conditions and uses a Landy-Szalay estimator. The pair separations range from 200 kpc/h to 20 Mpc/h, divided into 20 logarithmic bins. As we can see from figure 4, most prediction errors fall below 5% which is a relatively good result at these scales.

References

[1] A. Klypin, G. Yepes, S. Gottlöber, F. Prada, and S. Heß, "Multidark simulations: the story of dark matter halo concentrations and density profiles," *Monthly Notices of the*

Royal Astronomical Society, vol. 457, p. 4340–4359, Feb. 2016.

[2] P. S. Behroozi, R. H. Wechsler, and H.-Y. Wu, "The rockstar phase-space temporal halo finder and the velocity offsets of cluster cores," *The Astrophysical Journal*, vol. 762, p. 109, Dec. 2012.

[3] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[4] M. Sinha and L. H. Garrison, "CORRFUNC - a suite of blazing fast correlation functions on the CPU," *MNRAS*, vol. 491, pp. 3022–3041, Jan 2020.