Machine Learning in Large Scale Structure

Shirley Ho (Carnegie Mellon University)

Work with Xiaoying Xu,
Andrea Klein (CS), Shadab Alam, Zonngge Liu
Jeff Schneider (CS), Barnabas Poczos (CS), Junier Oliver (CS)

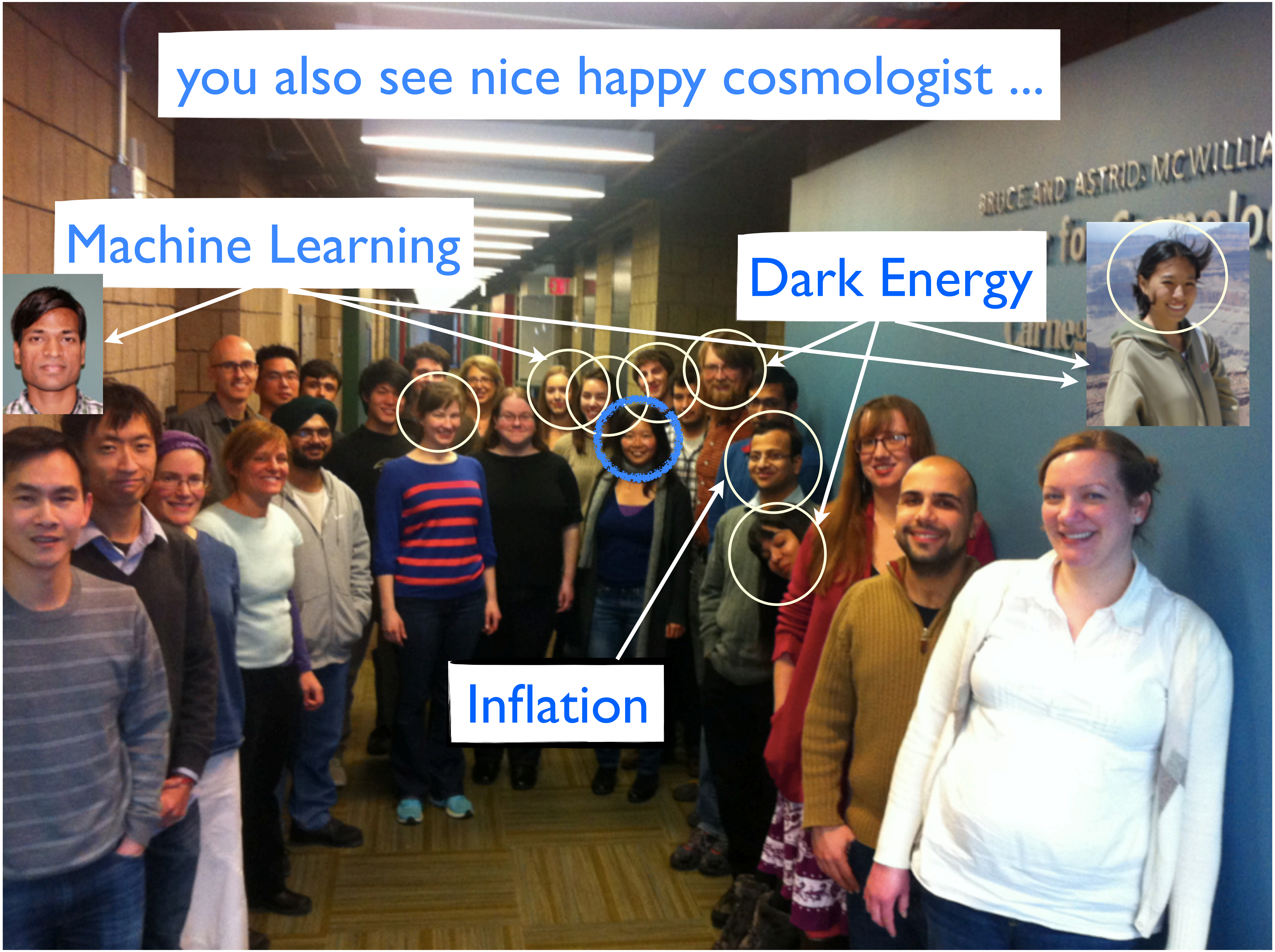
Carnegie Mellon University
What do you see on campus?

Carnegie Mellon University
What do you see on campus?

Bat vehicle + rioting Gotham citizens ...

Carnegie Mellon University
What do you see on campus?

Cat lady escaping from the Dean's office


you also see nice happy cosmologist ...
Machine Learning
Dark Energy
Inflation

Outline

- Simulation Production
  - Populating Halos with Galaxies using Machine Learning
  - Generating the Density field

Observations:

flux(x,y,band/wavelength), observational systematics(x,y,band/wavelength): psf, sky, dust, airmass and respective errors

Basic Reduction pipeline

Extracted object, observational systematics properties

LSS systematics removal, statistics estimation

Large Scale Structure (BAO, full clustering measurements)

Theory, Beyond Linear Model Predictions

Simulations (possibly with DM simulations + HOD), Covariance Matrix

Cosmology (cosmological parameters, formation and evolution of galaxies, quasars...)

Observations:

flux(positions,band/wavelength), observational systematics(positions,band/wavelength):
psf, sky, dust, airmass and respective errors

Basic Reduction pipeline

Extracted object, observational systematics properties

LSS systematics removal, statistics estimation

Large Scale Structure measurements
BAO (with reconstruction), full clustering

Linear Theory and
Beyond Linear Modeling

Covariance Matrix

Astronomy and Cosmology
[cosmological parameters, formation and evolution of galaxies, quasars...]

Why Populate Halos?

- Mock galaxy catalogues! These are necessary for LSS analysis – calculating the covariance matrix, testing the pipeline, etc.
- Alternatives:
  - Run N-body + hydro simulations, very expensive if you want many mocks.
  - Perturbation theory, not accurate enough on scales < 20Mpc, especially in redshift space.

Why Populate Halos?

- Mock galaxy catalogues! These are necessary for LSS analysis – calculating the covariance matrix, testing the pipeline, etc.
- Alternatives:
  - Run N-body + hydro simulations, very expensive if you want many mocks.
  - Perturbation theory, not accurate enough on scales < 20Mpc, especially in redshift space.

NB:
Here, by populating halos, I specifically mean determining the number of galaxies that will reside in a halo given its properties.

Why Populate Halos?

- Mock galaxy catalogues! These are necessary for LSS analysis – calculating the covariance matrix, testing the pipeline, etc.
- Alternatives:
  - Run N-body + hydro simulations, very expensive if you want many mocks.
  - Perturbation theory, not accurate enough on scales < 20Mpc, especially in redshift space.

Introducing Machine Learning

- Machine Learning algorithms learns trends from the data itself, it does not impose pre-assumed models on the data.
- The advantage of ML is that it is fully non-parametric:
  - The only assumption necessary is that some relationship **does** exist between halo properties (features) and the number of galaxies that will reside in it and that this relationship is continuous.

Cool Examples of ML applications (Courtesy Slide from Kayvon Fatahalian)

[Shrivastava 2011]
"Find images that are similar to a query image (even if not similar in individual pixel values)."

Query image (snowy day)

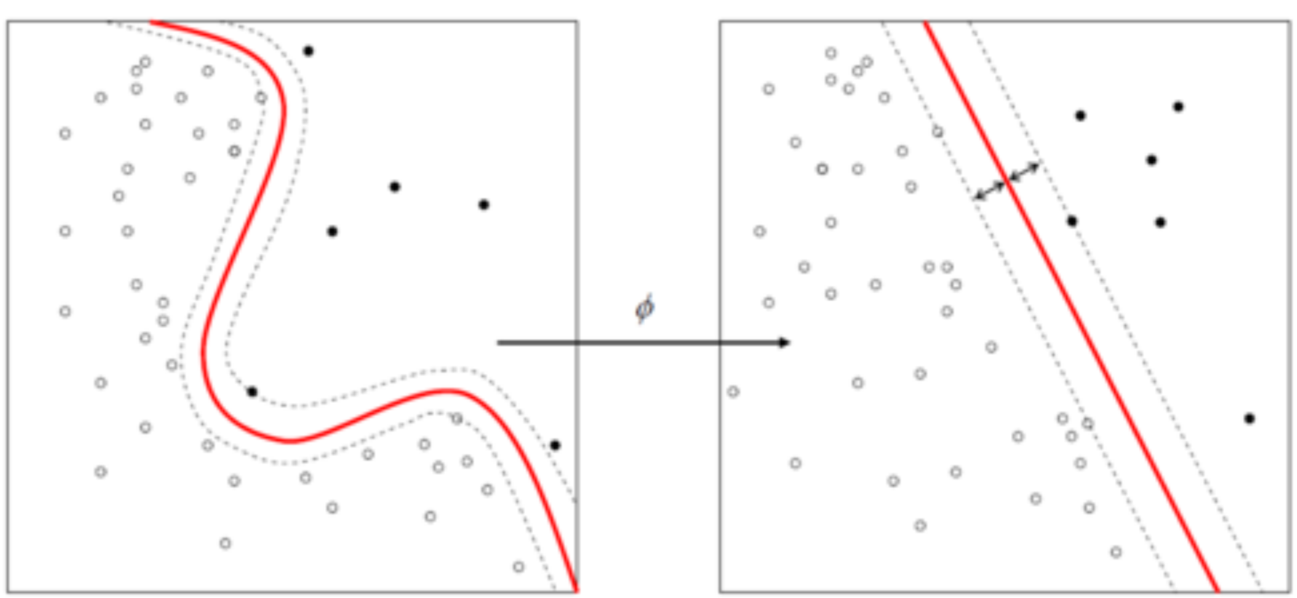
Matches

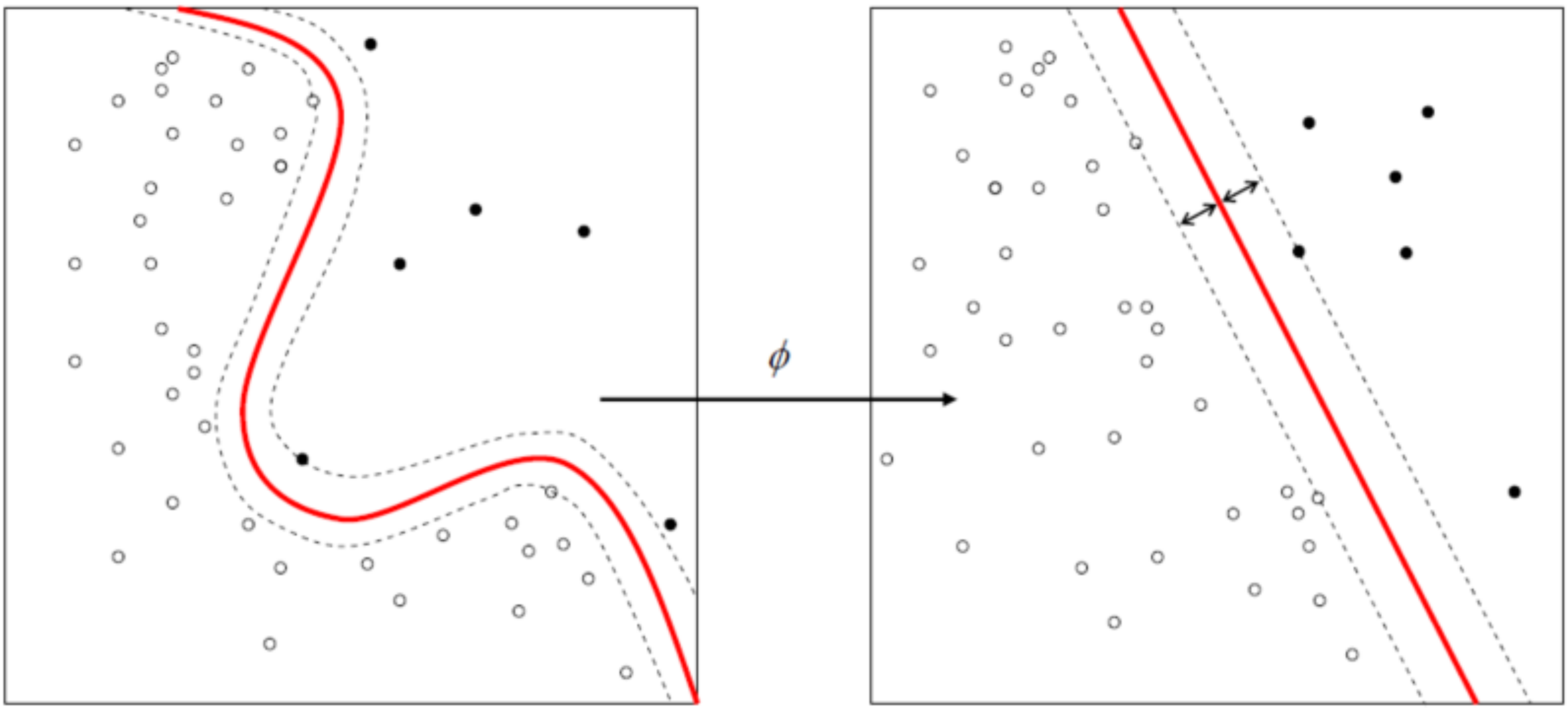
[Doersch 2012]
"Find meaningful visual elements that are unique to Paris"

Simple ML algorithms

SVM: Support Vector Machine

- SVM maps the data into higher dimensional space with a kernel. To train, separate data into classes using hyperplanes. It can generalized into regression (not only classification).




ϕ

Simple ML algorithms

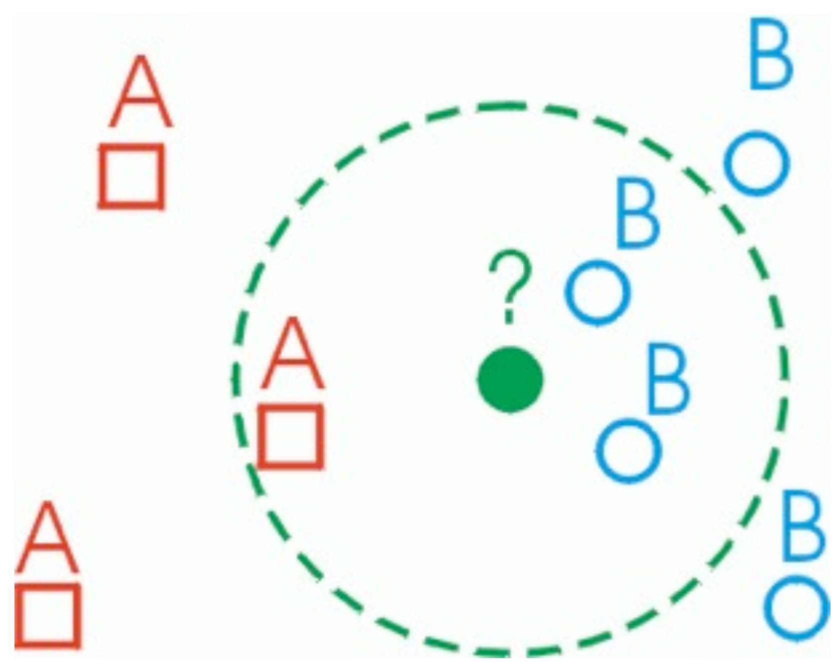
kNN: k–nearest neighbors

- kNN takes the average of k nearest neighbors to the point of interest in the training set.

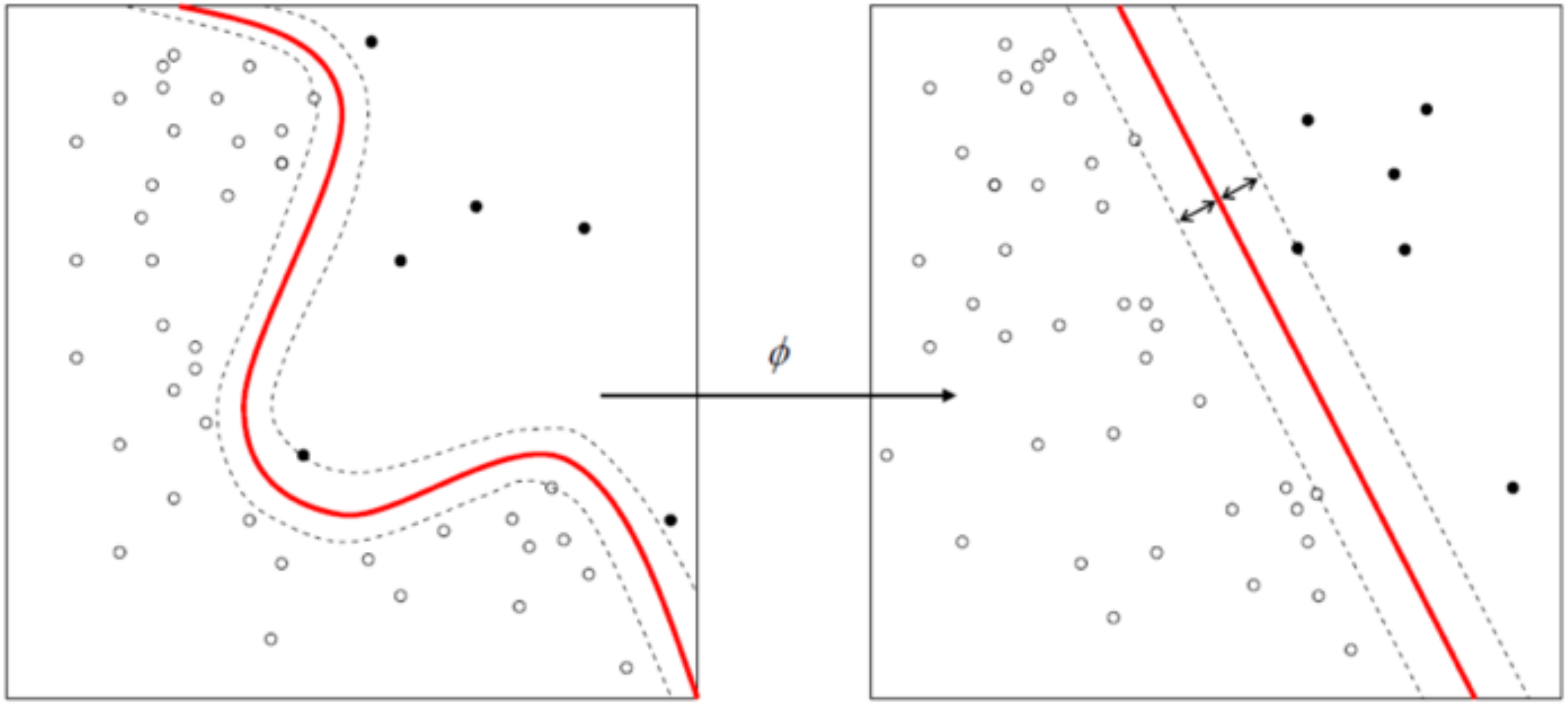
Simple ML algorithms

kNN: k–nearest neighbors

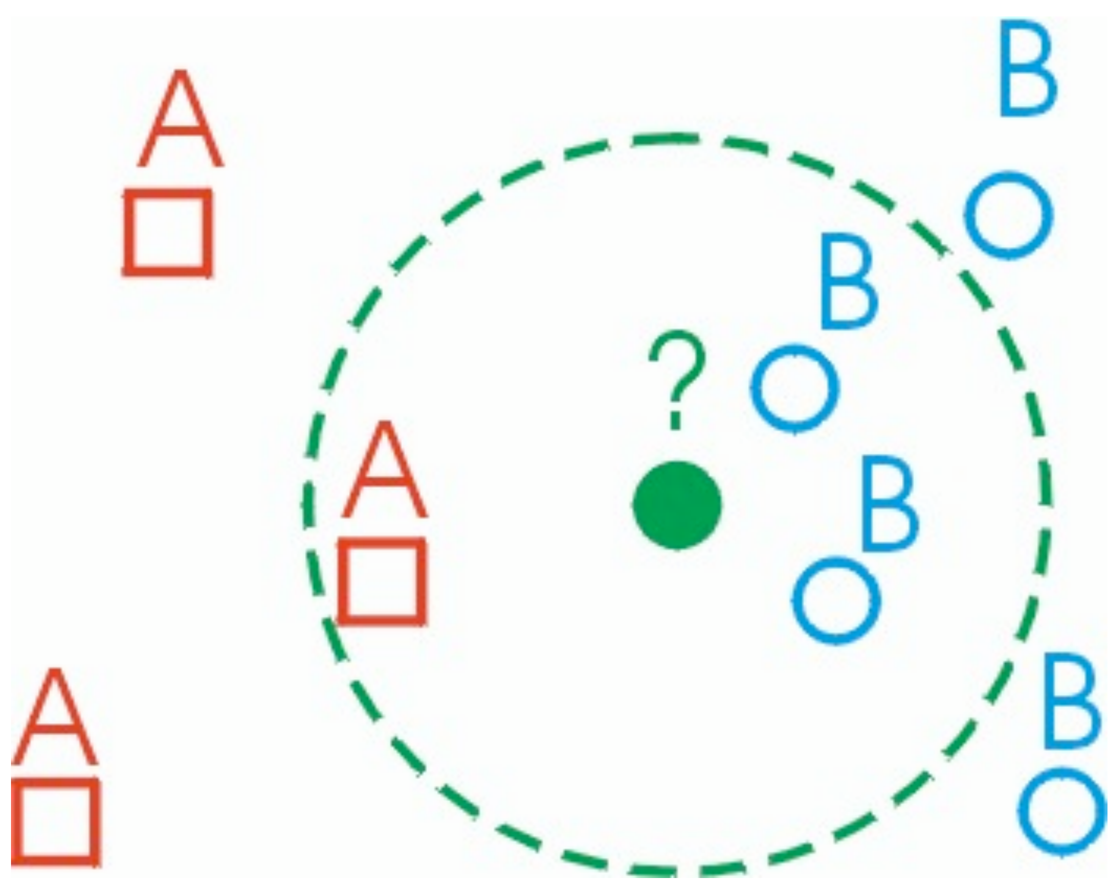
- kNN takes the average of k nearest neighbors to the point of interest in the training set.



SVM



kNN



Introducing Machine Learning (cont'd)

- To evaluate how good a machine learning algorithm perform, we use the "mean-squared-error (MSE):
- $$\mathrm{MSE} = \frac{\sum_{i=1}^{N}(Y_{i,test,true} - Y_{i,test,predicted})^2}{N}$$
- We can compare the MSEs given by different algorithms and see which one does a better job, and we can also compare it to the **base MSE**, which basically replace $Y_{\text{test,predicted}}$ by average of Y_{train}

Machine Learning
Populating Halos with Galaxies

- We want to use Machine Learning to learn about how many galaxies (or specific kinds of galaxies) are in halos with certain properties.
- We can do this with real data, however, we are not in the era where we have lots of data on a lot of halos *yet*.
- We then use simulations which included 'all/some' physics (with lots of halos and halo properties).

Machine Learning
Populating Halos with Galaxies

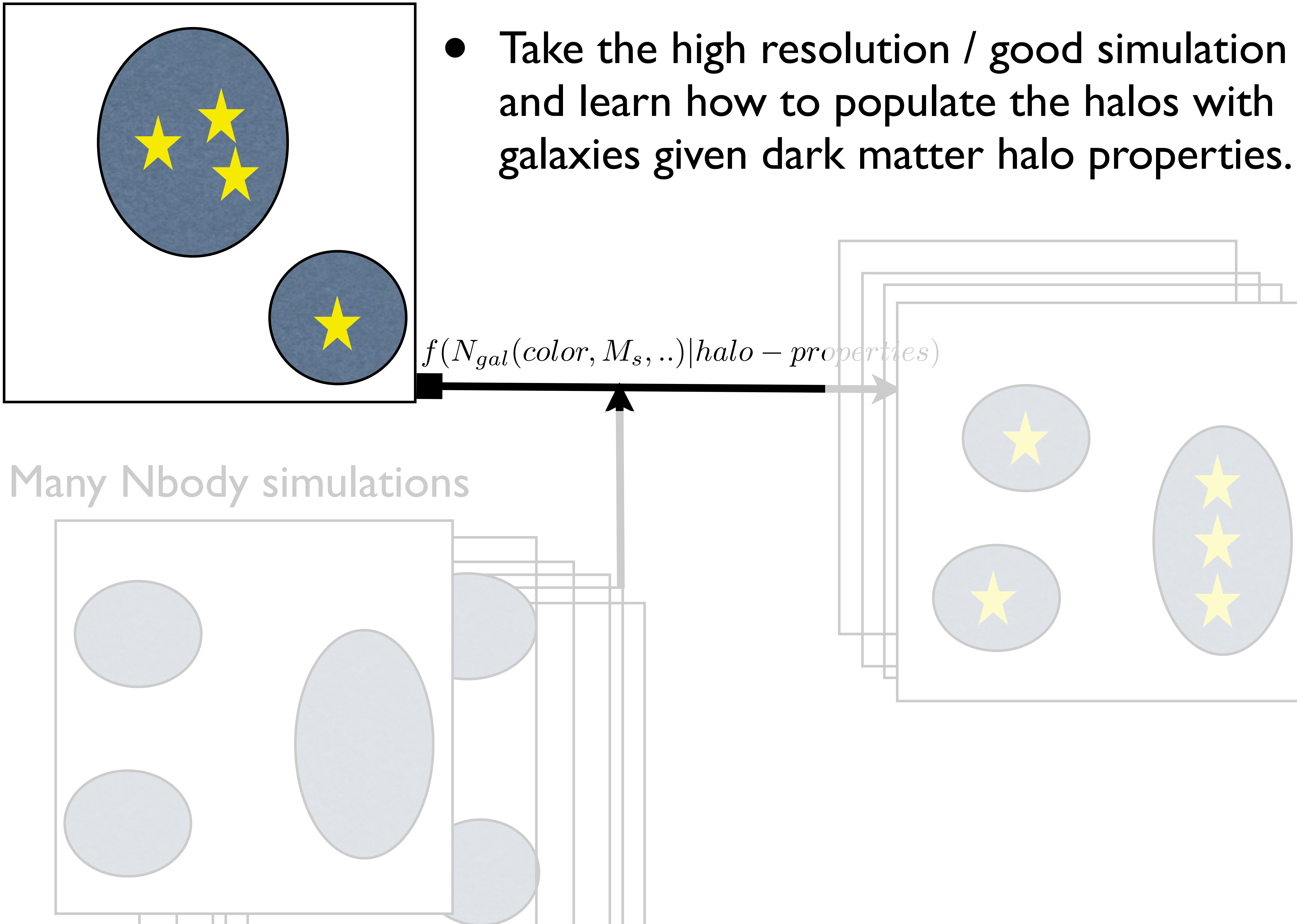
- We want to use Machine Learning to learn about how many galaxies (or specific kinds of galaxies) are in halos with certain properties.
- We can do this with real data, however, we are not in the era where we have lots of data on a lot of halos *yet.*
- We then use simulations which included 'all/some' physics (with lots of halos and halo properties).

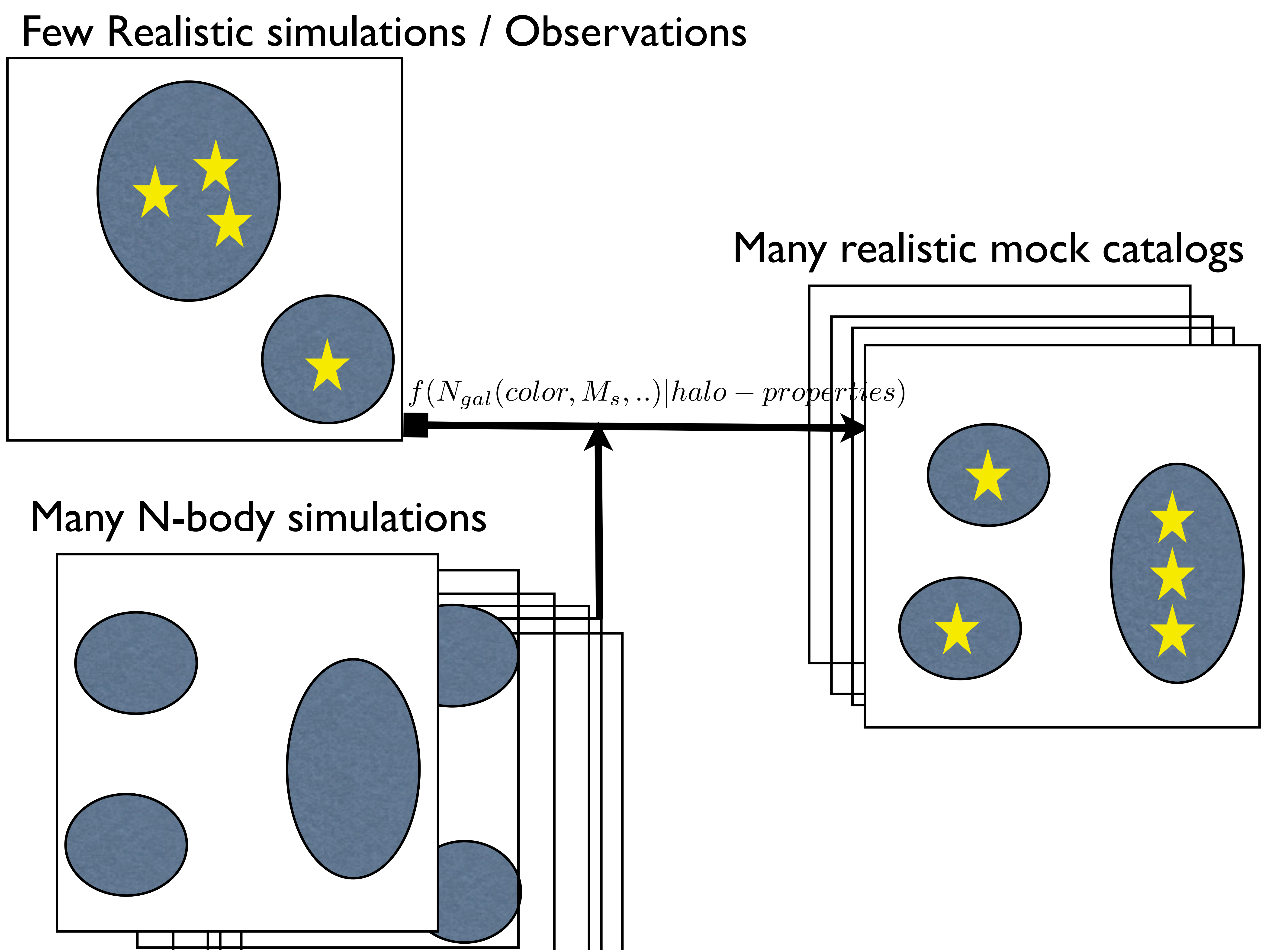
Machine Learning
Populating Halos with Galaxies

- We want to use Machine Learning to learn about how many galaxies (or specific kinds of galaxies) are in halos with certain properties.
- We can do this with real data, however, we are not in the era where we have lots of data on a lot of halos *yet*.
- We then use simulations which included ‘all/some’ physics (with lots of halos and halo properties).

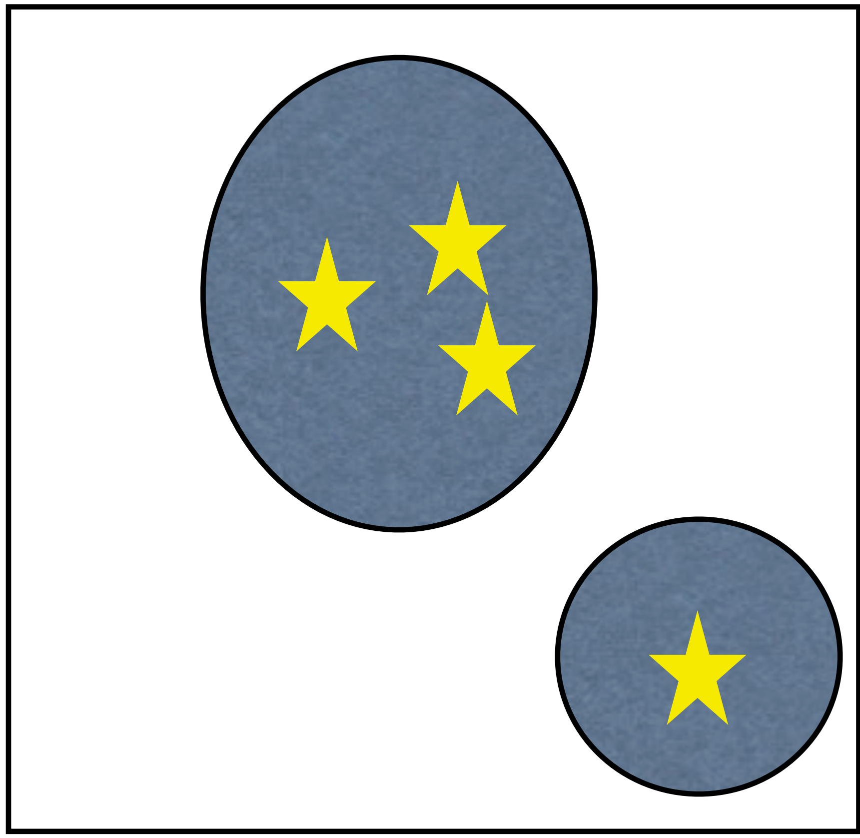
Realistic simulations / Observations

- Take the high resolution / good simulation and learn how to populate the halos with galaxies given dark matter halo properties.

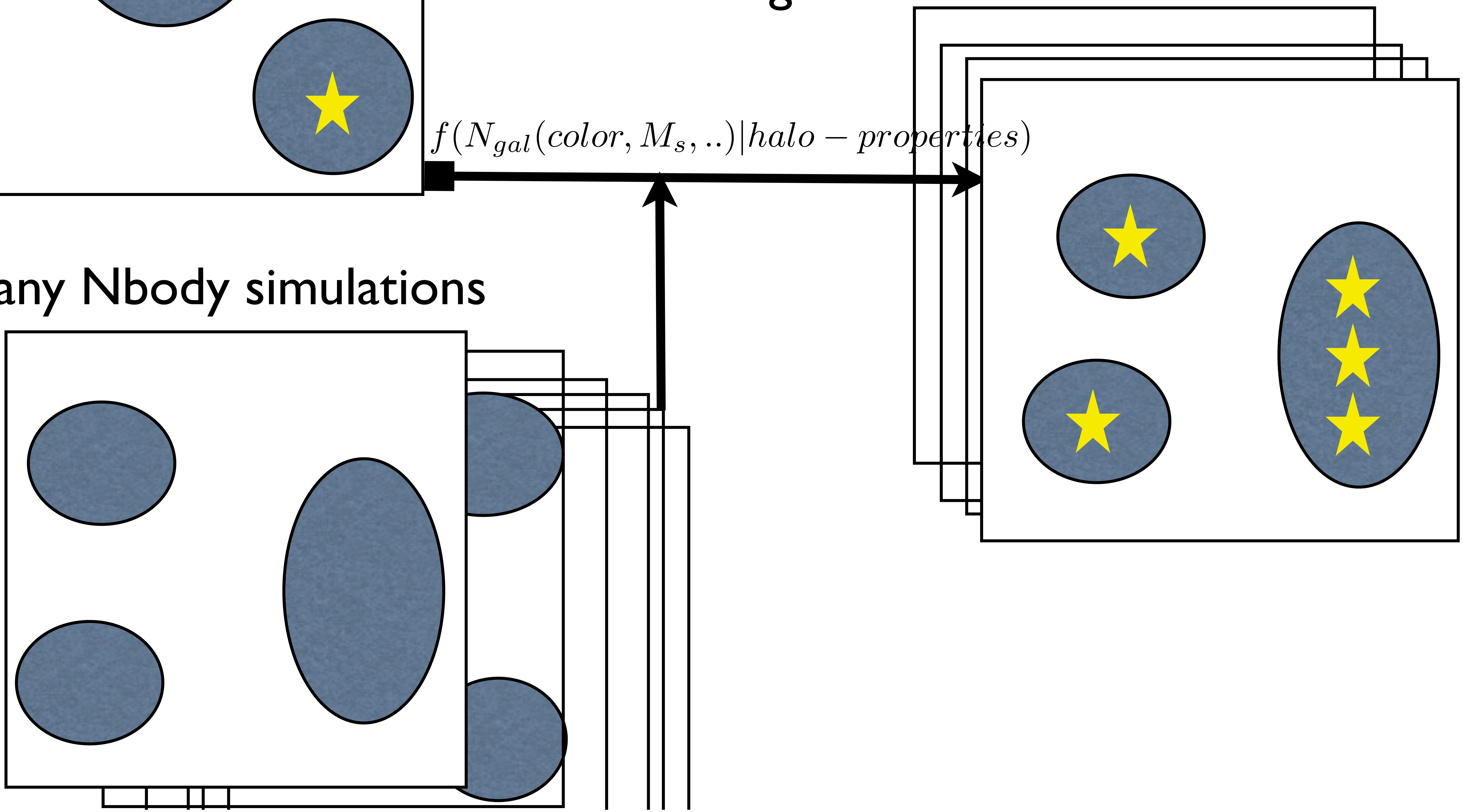



Few Realistic simulations / Observations
$f(N_{gal}(color, M_s, ..)|halo - properties)$
Many realistic mock catalogs
Many N-body simulations

Realistic simulations / Observations



- Can then take dark matter halos from large number of not-as-expensive DM only simulations, and make many independent mock catalogs.



Many Nbody simulations

Not enough observations... so we use Millenium

- We use the halo catalogues and semi-analytic galaxies from the Millennium simulation: Ω_m=0.25, Ω_b=0.045, Ω_Λ=0.75, h=0.73, n_s=1.0, σ_8=0.9.
- We only use the central and satellite galaxies of the primary Millennium halos with mass > $10^{12} M_{sun}/h$.
- There are about 400,000 of these halos. We subsample to 60,000 for some tests.
- We split the sample/subsample randomly and equally into training and test sets.

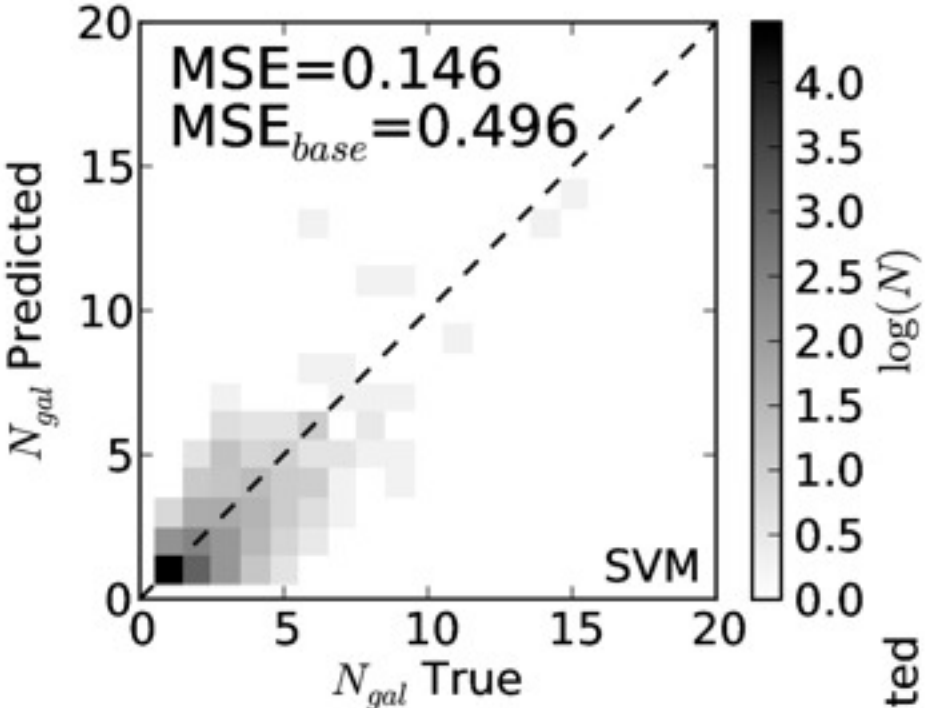
Process

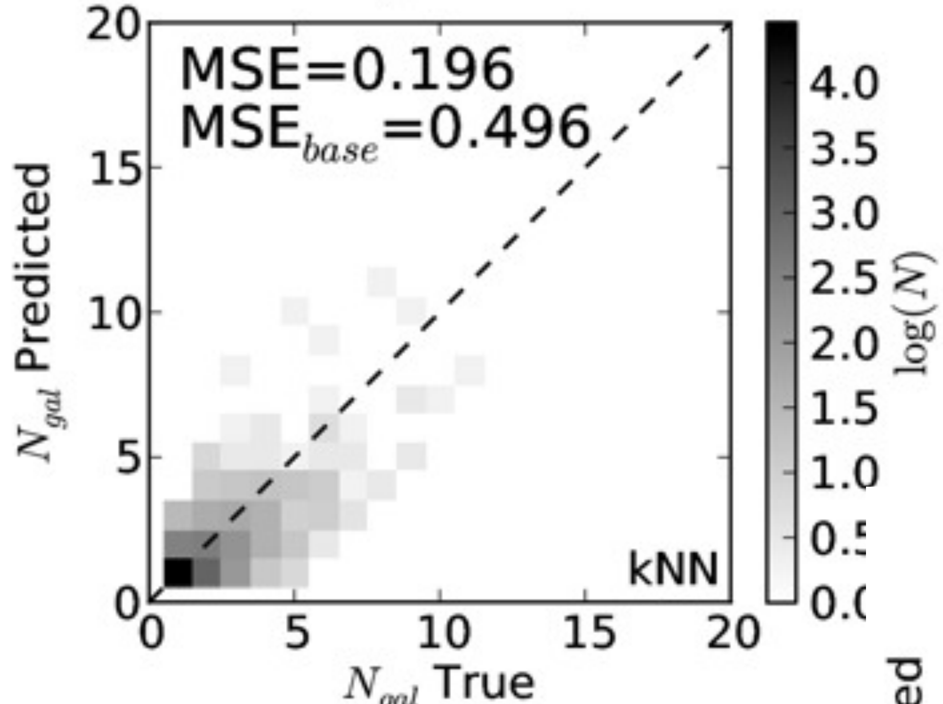
- We split the data into 3 sets
- One for training
- One for validation
- One for testing (we predict the number of galaxies given what we learn from the training set)

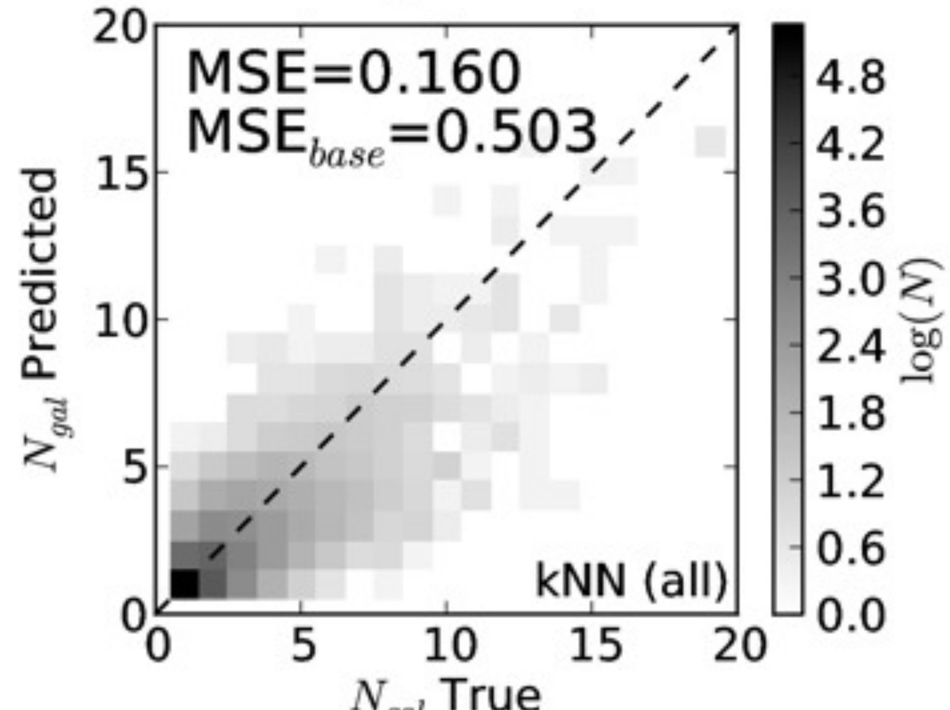
Results - Number of galaxies

Xu, S.H, Trac, Schneider, Poczos 2013

Results - Number of galaxies

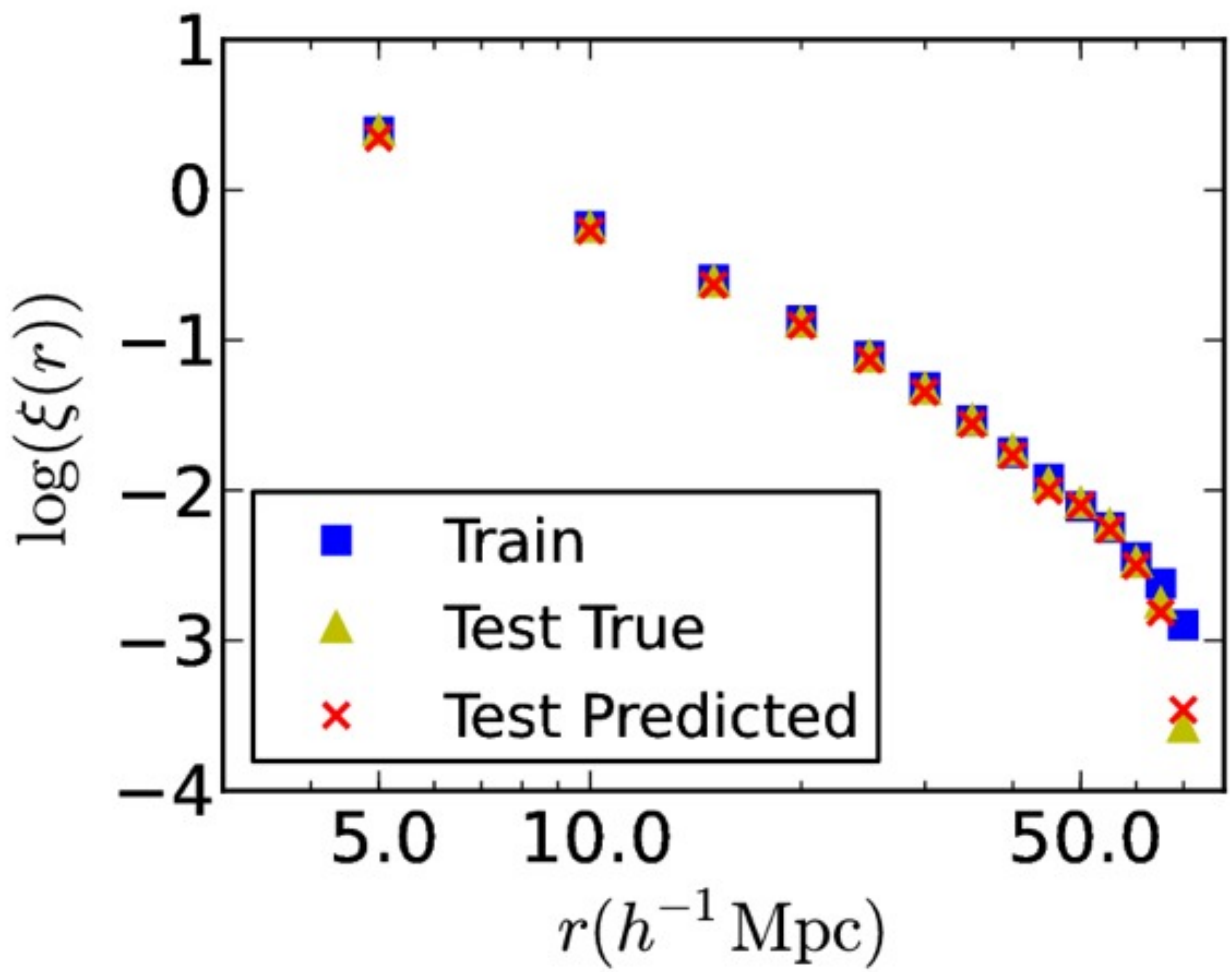






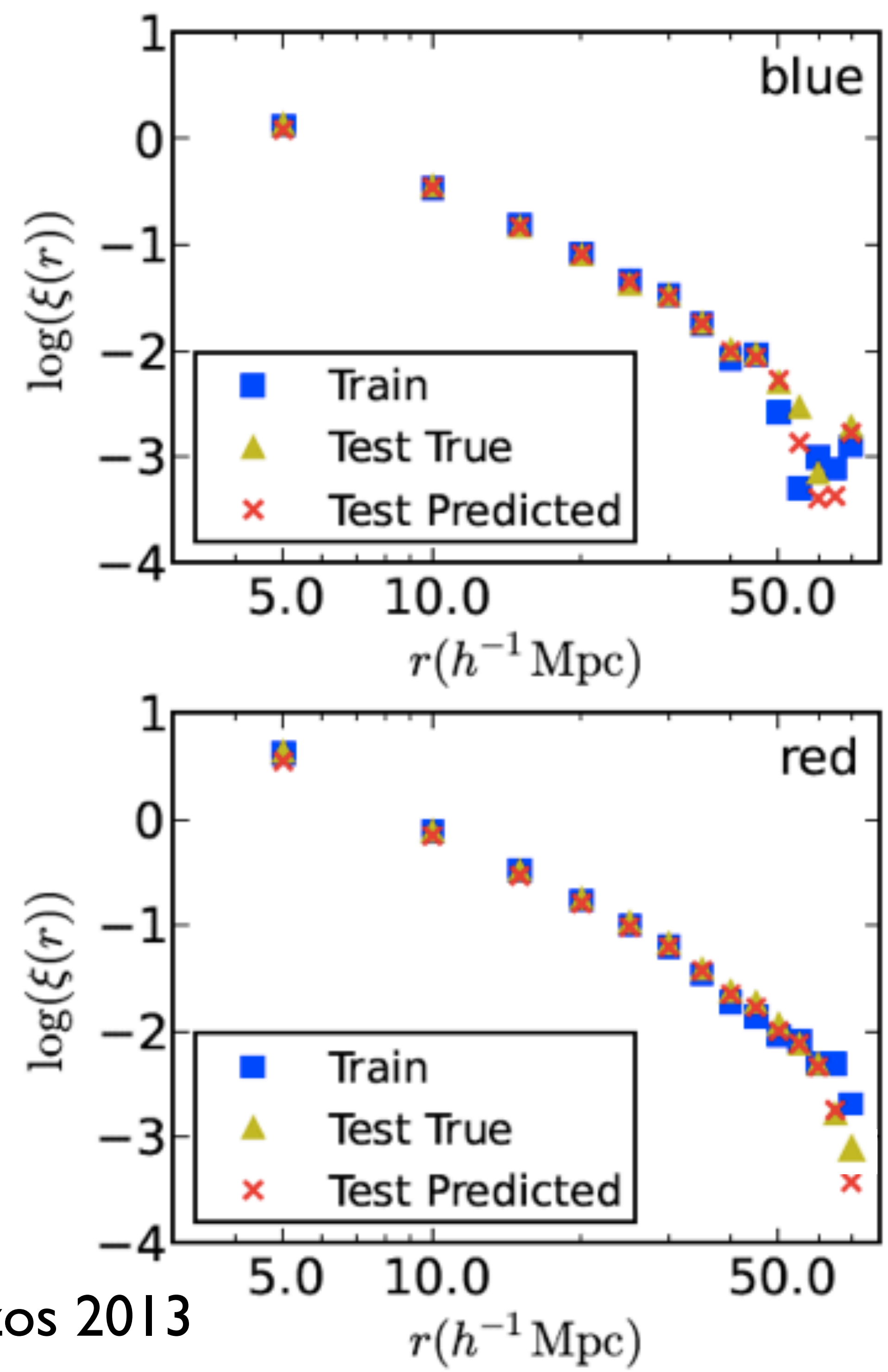
Xu, S.H, Trac, Schneider, Poczos 2013

Results - Correlation functions



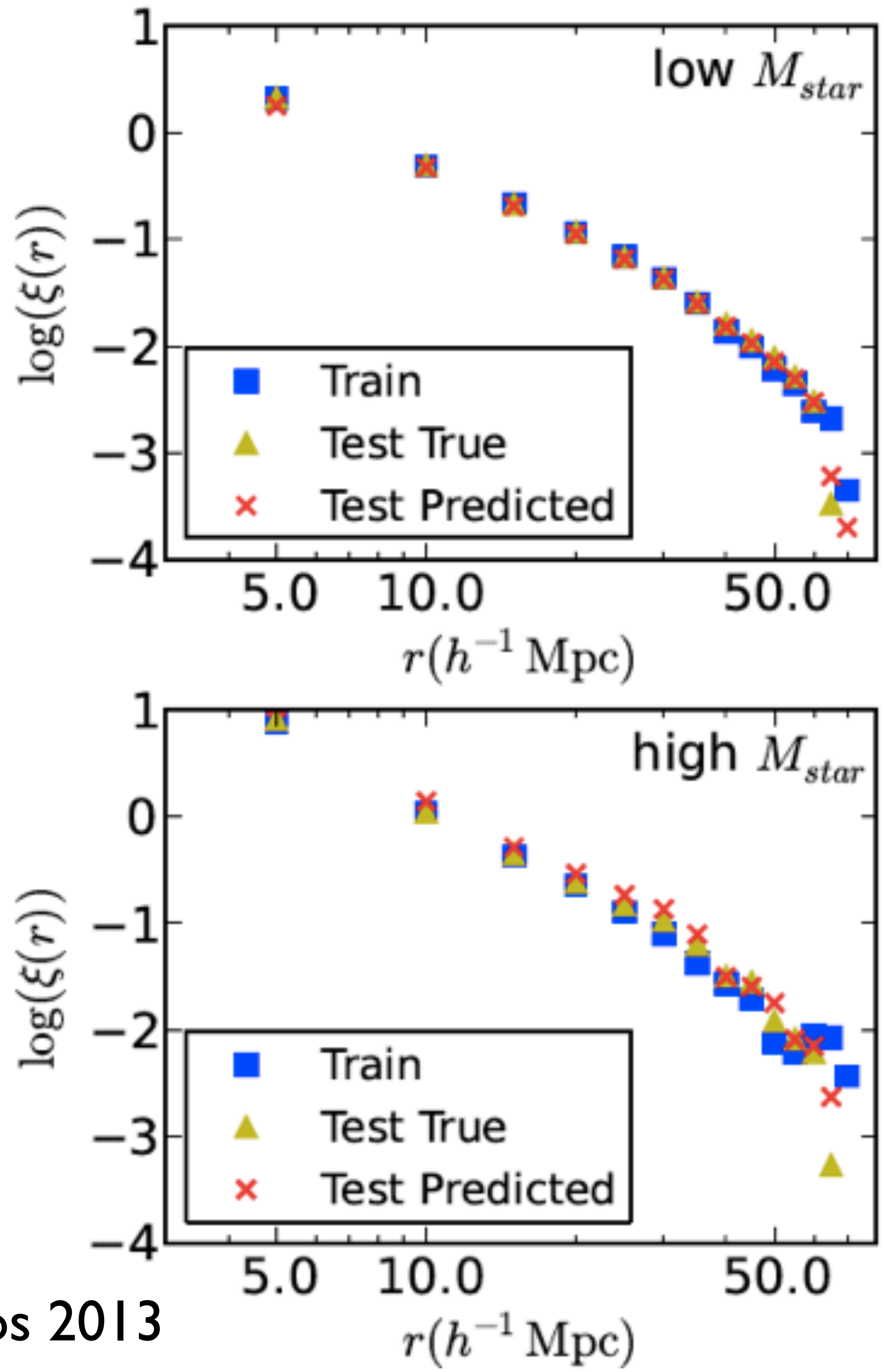
Xu, S.H, Trac, Schneider, Poczos 2013

Results - Correlation functions of blue and red galaxies

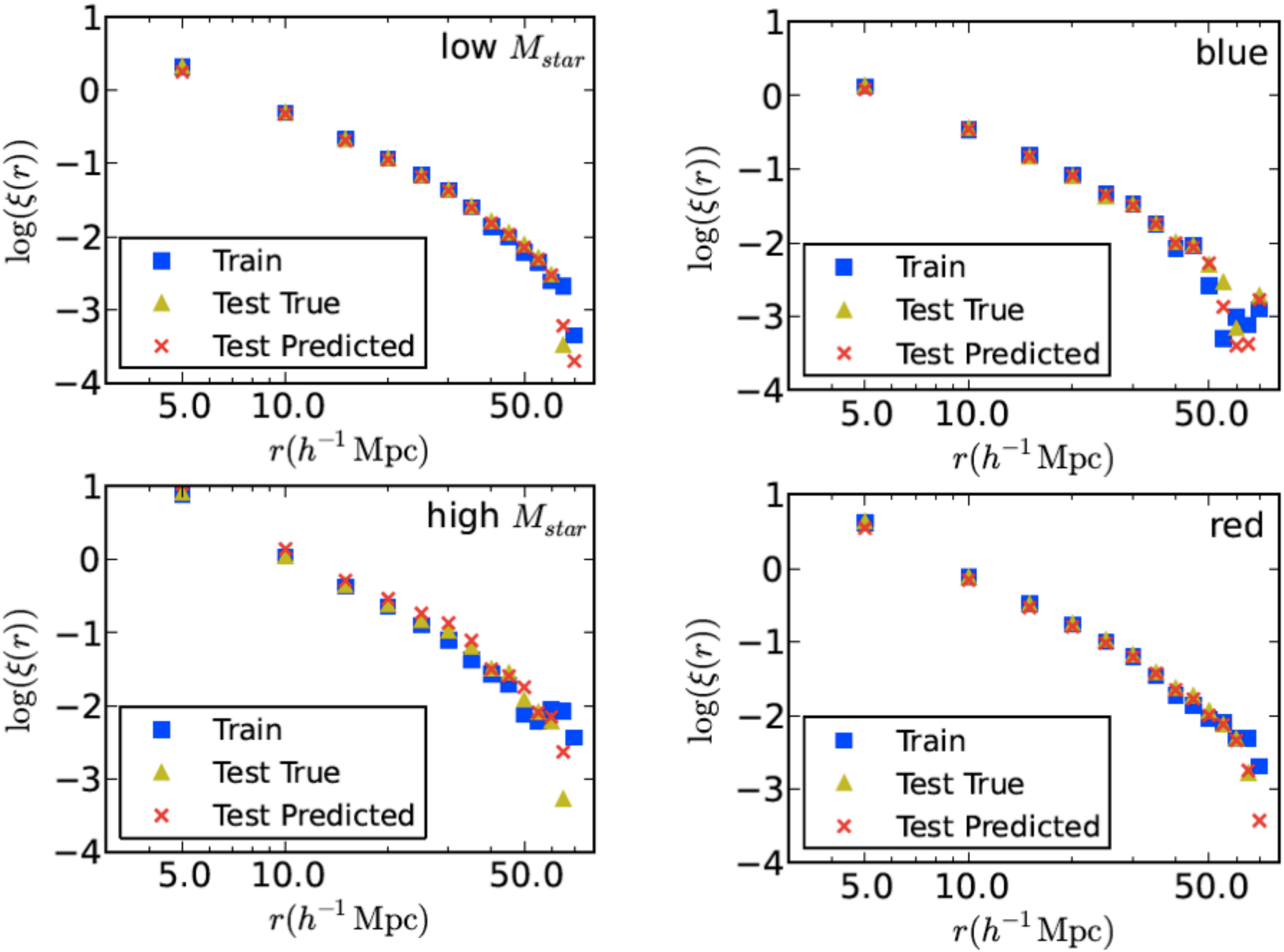


Xu, S.H, Trac, Schneider, Poczos 2013

Results - Correlation functions of galaxies with different stellar mass thresholds

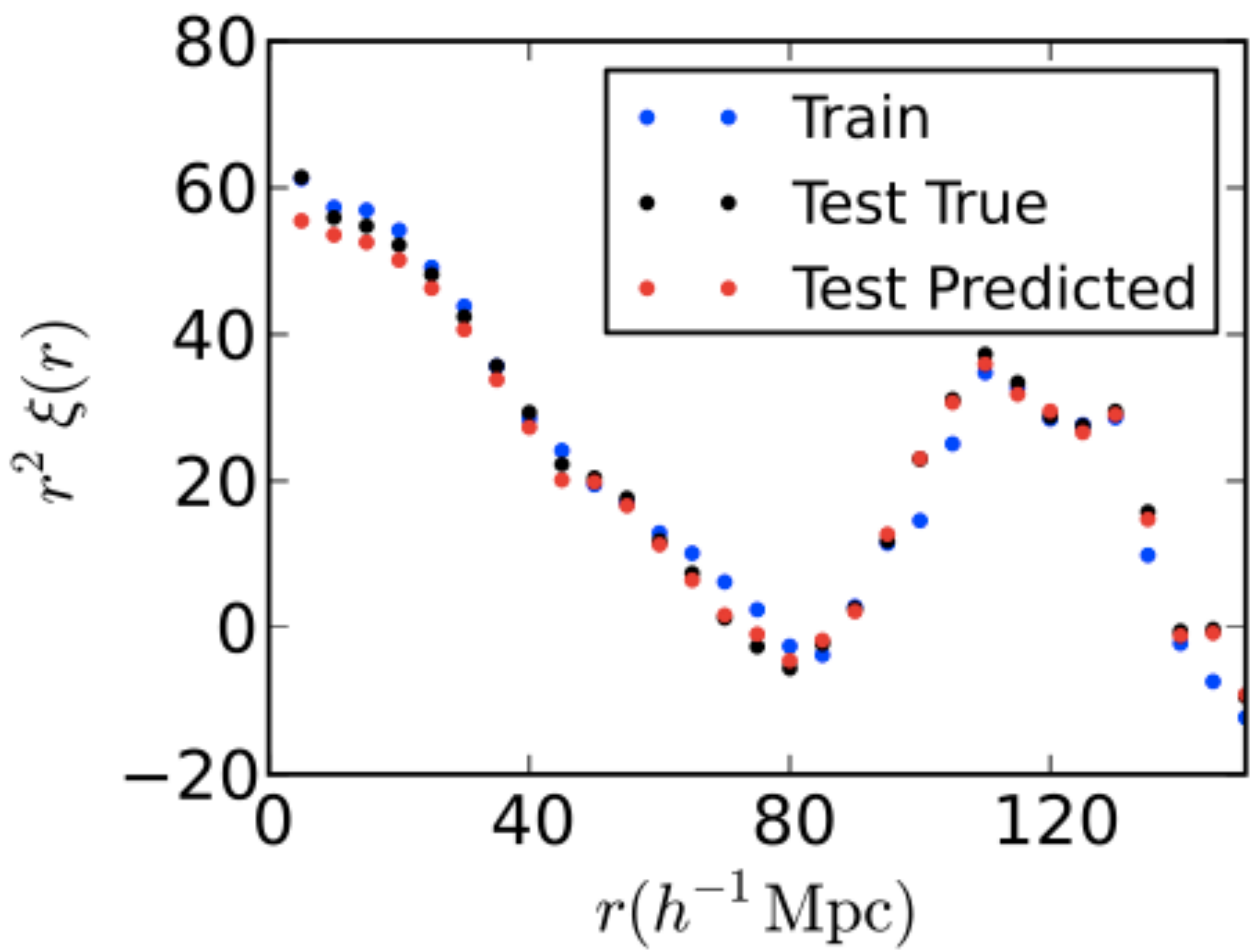


Xu, S.H, Trac, Schneider, Poczos 2013



low M_{star}
blue
high M_{star}
red
Train
Test True
Test Predicted
$\log(\xi(r))$
$r(h^{-1}\,\mathrm{Mpc})$
5.0
10.0
50.0

Results VI - Correlation function at large scale (BAO)

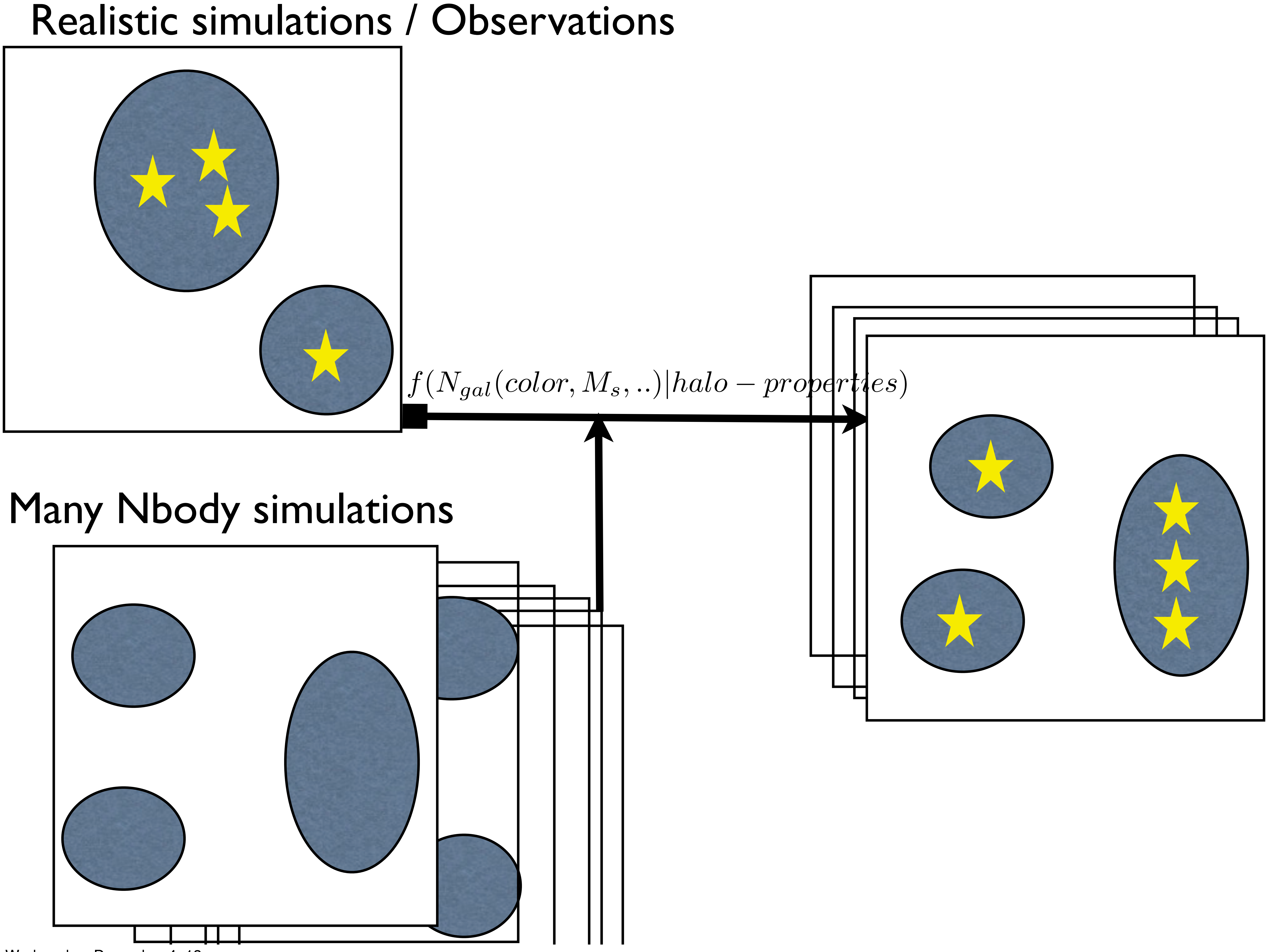


Note this is using only 60,000 galaxies, so there is quite a bit of shot noise.

Xu, S.H, Trac, Schneider, Poczos 2013

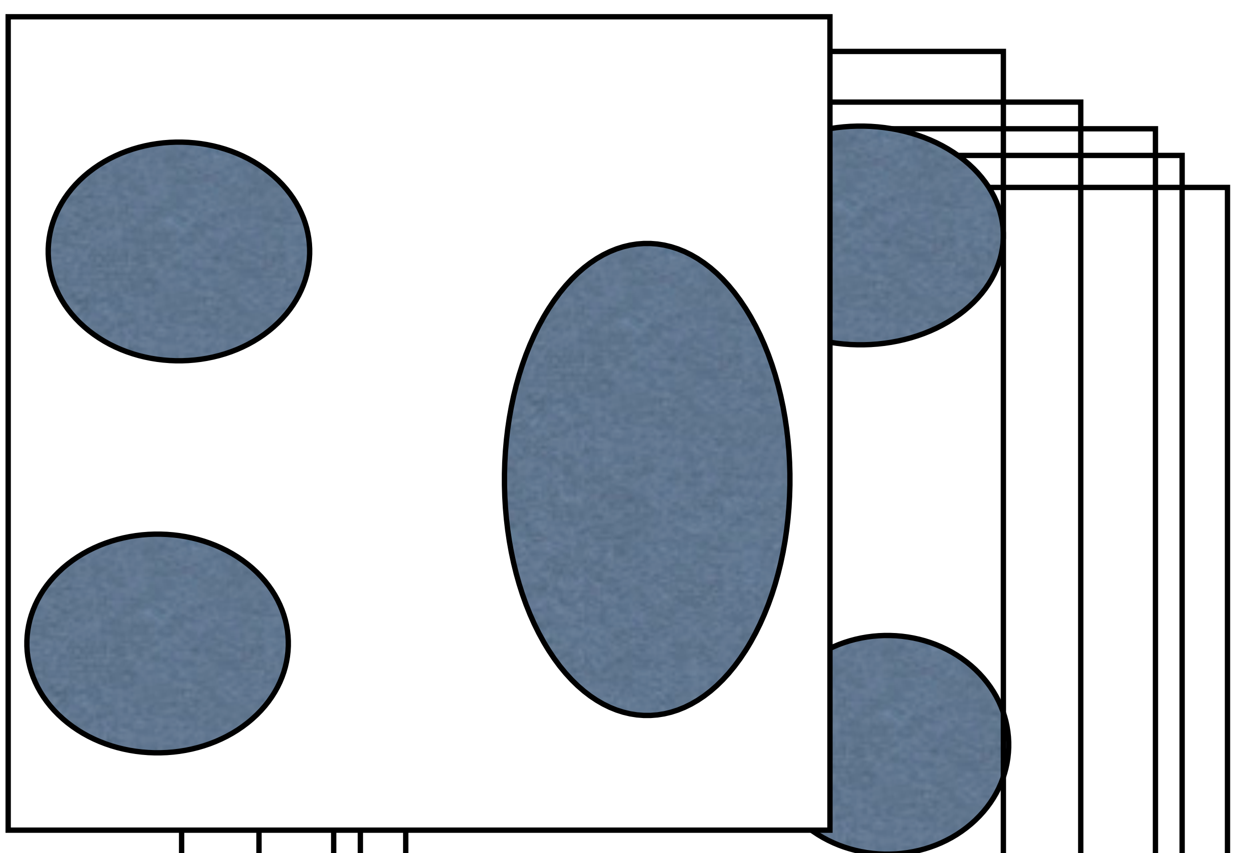
Mini-Conclusion

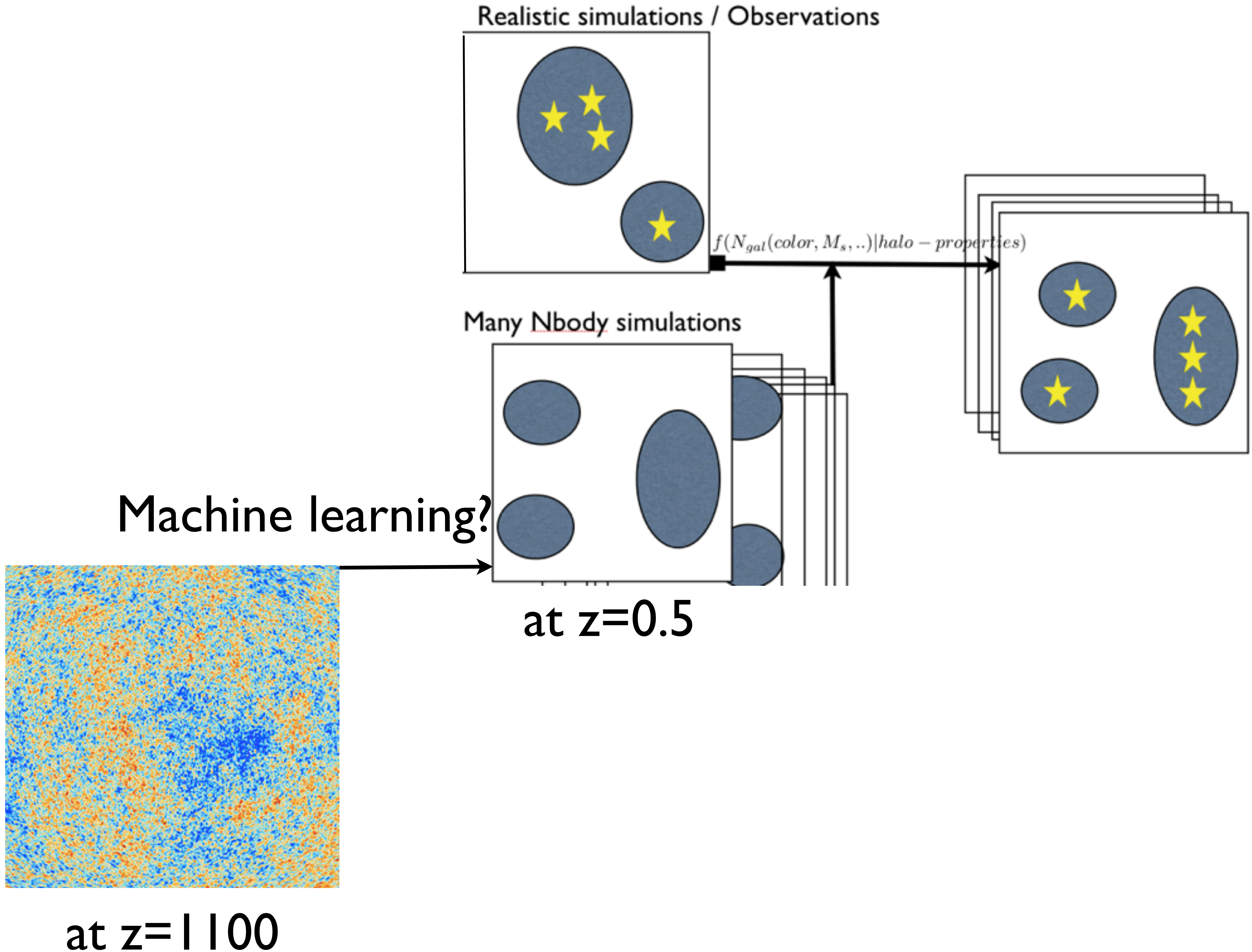
- Machine learning offers a completely **model-independent** method for understanding the halo-galaxy mapping while avoiding subhalo finding.
- ML techniques give robust predictions of the number of galaxies per halo, the distribution of halos with N_{gal} and the galaxy correlation function.
- **Now, we move to even more ambitious question:**
    - **Can we skip the whole process of N-body simulations with Machine Learning?**

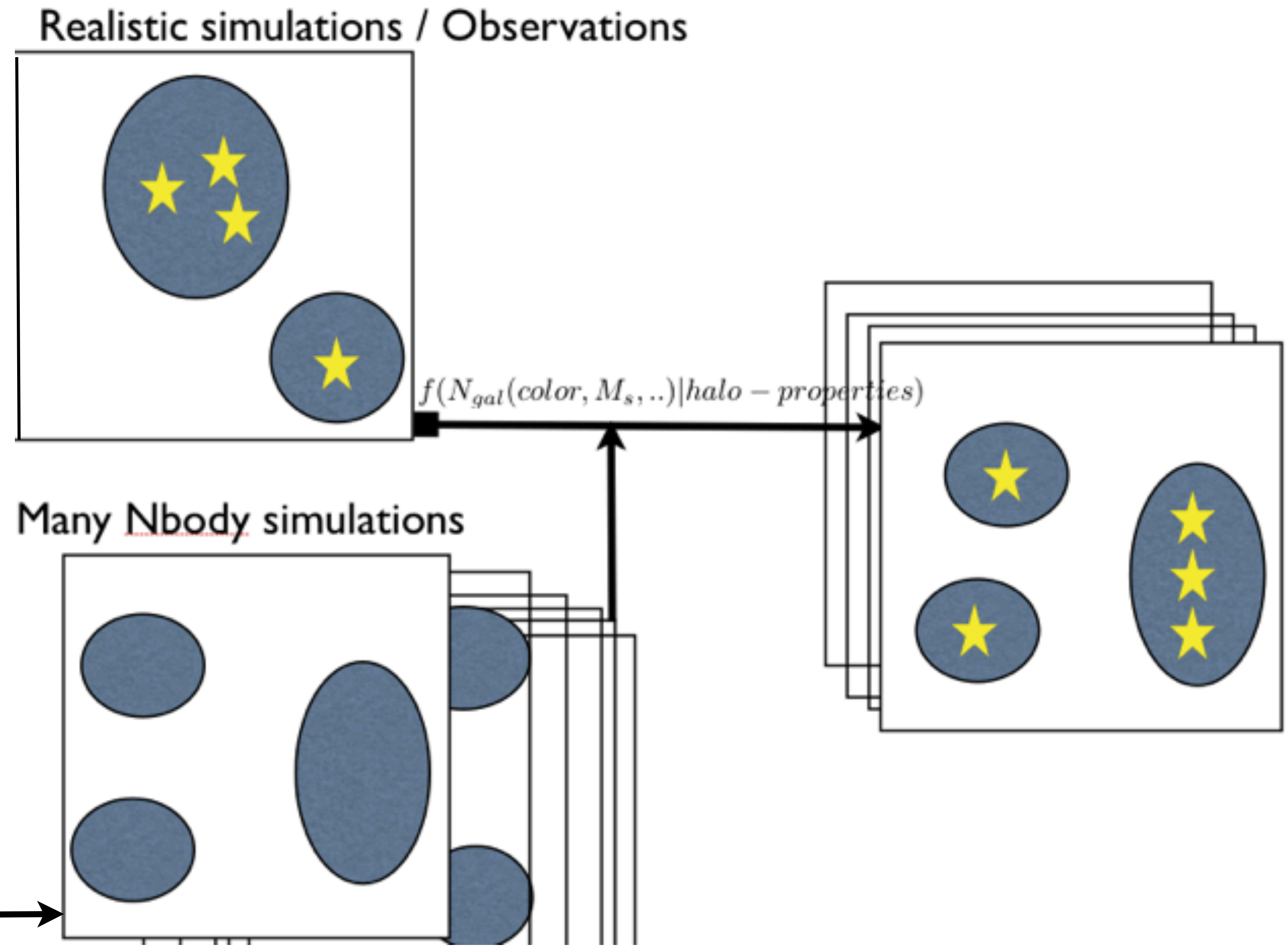


Realistic simulations / Observations
$f(N_{gal}(color, M_s, ..)|halo - properties)$
Many Nbody simulations

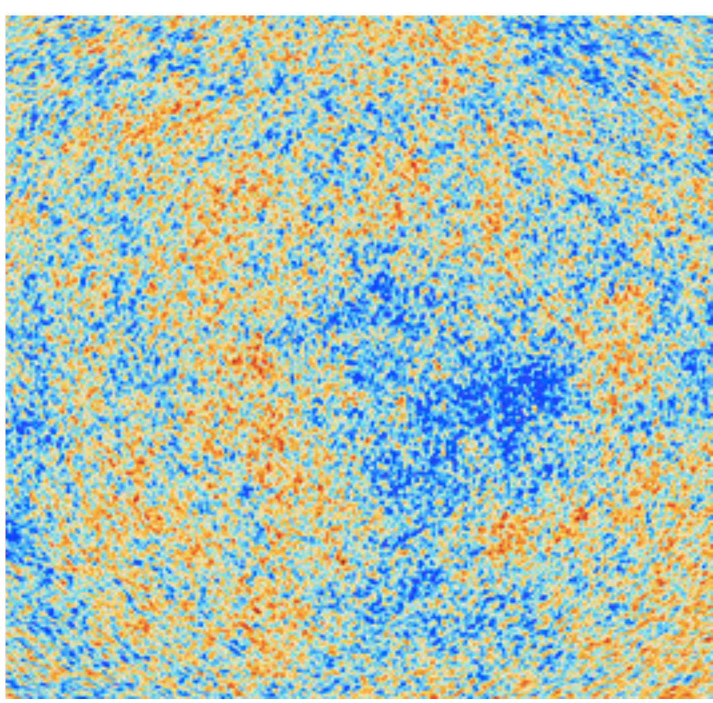
Many Nbody simulations




Realistic simulations / Observations
$f(N_{gal}(color, M_s, ..)|halo - properties)$
Many Nbody simulations
Machine learning?
at z=0.5
at z=1100

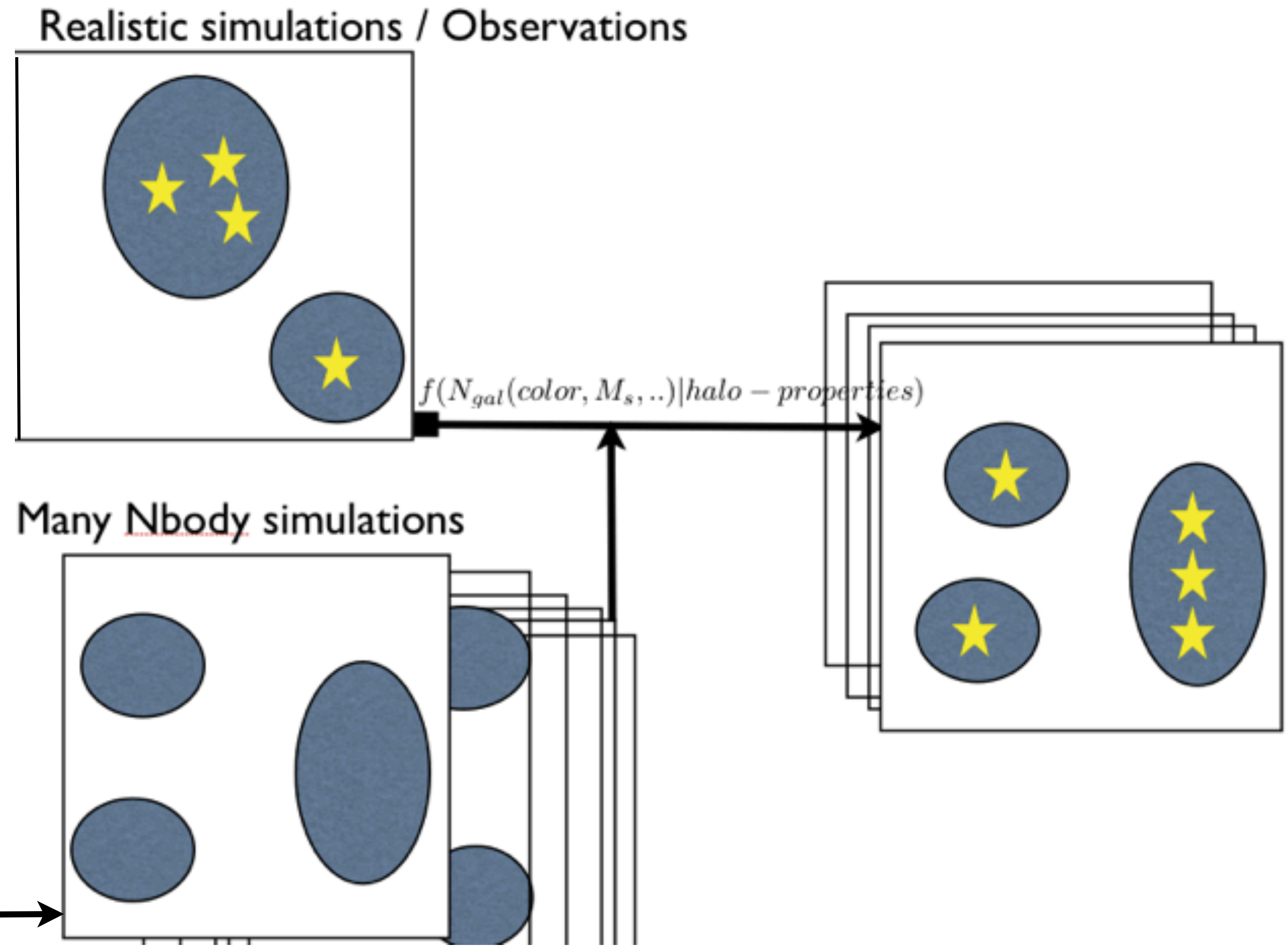


at z=0.5

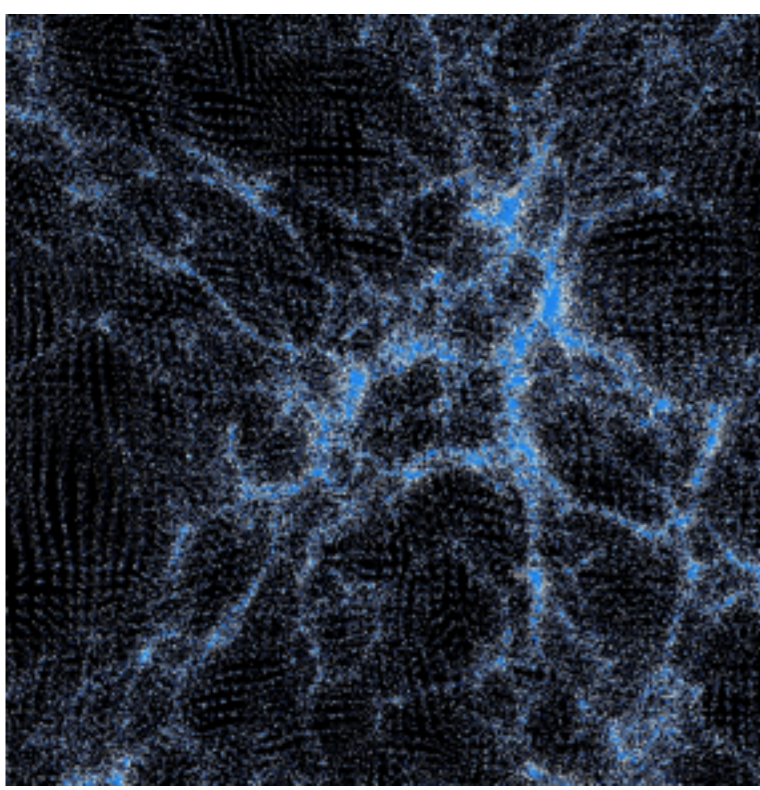


at z=1100

New method in machine learning allows distribution-to-distribution prediction

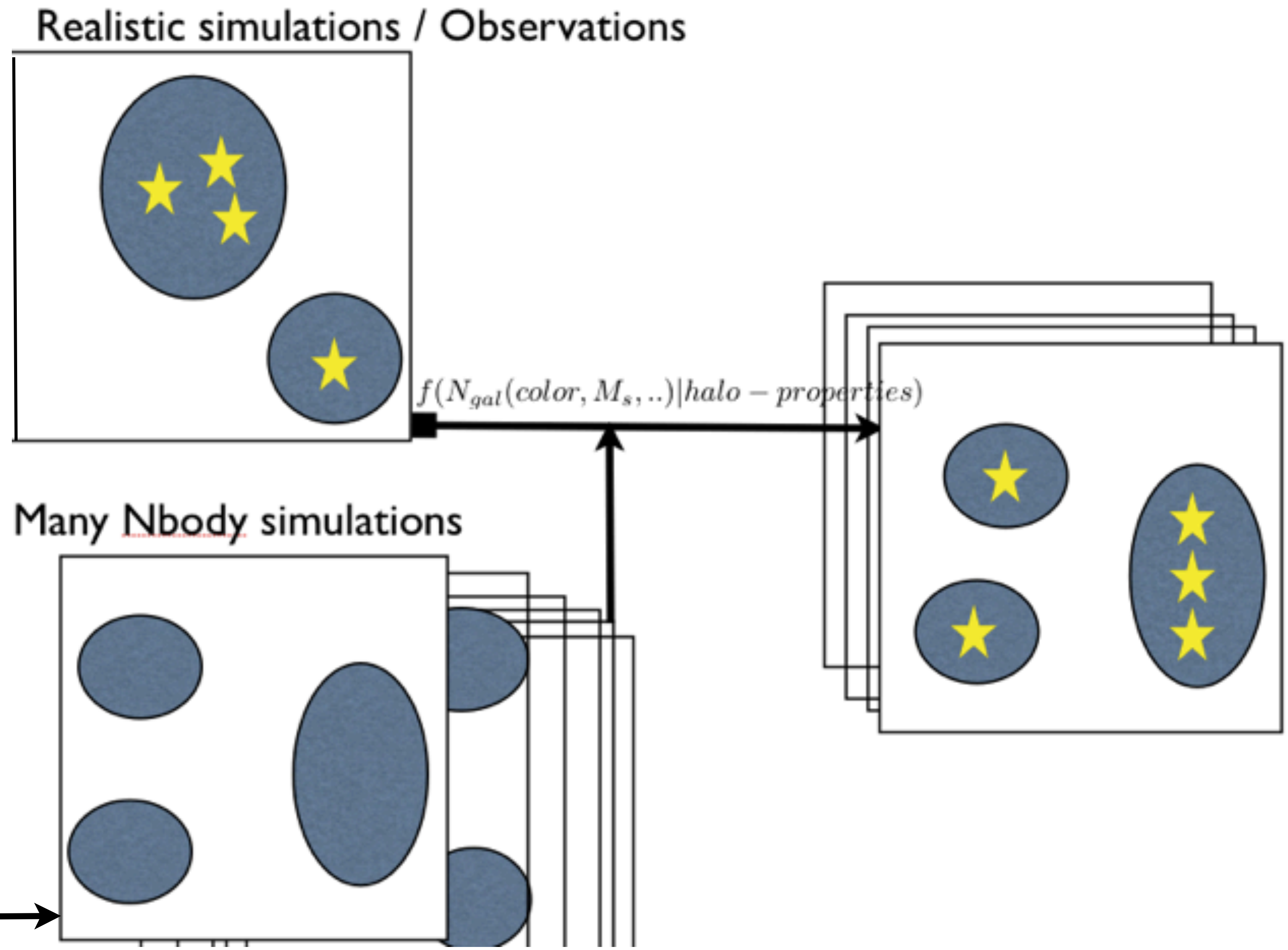


at z=0.5

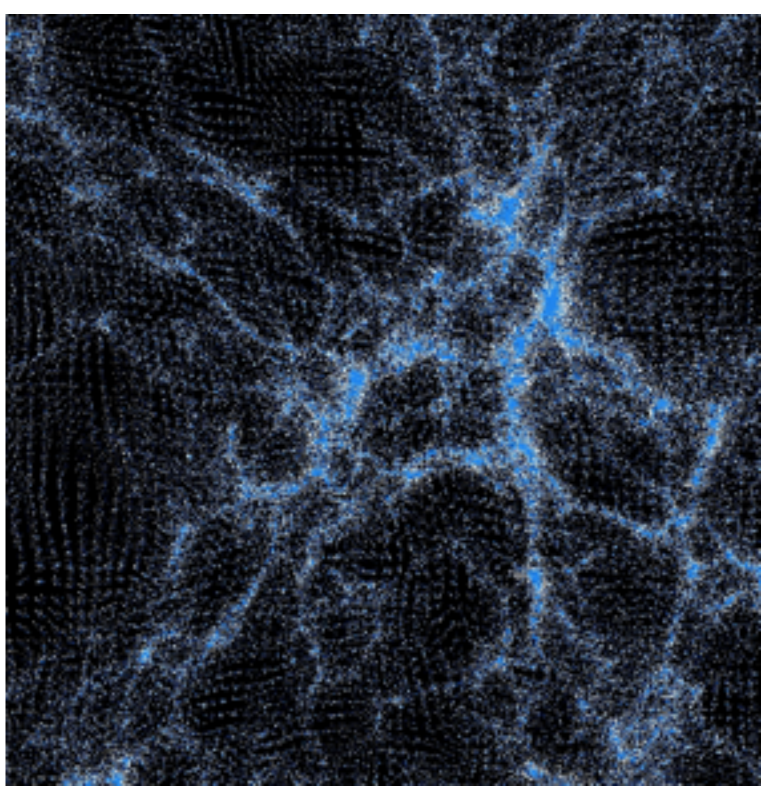


2LPT to z=0.5

To make this easier, we evolve the field from initial conditions to z=0.5 using 2LPT (~ mins)



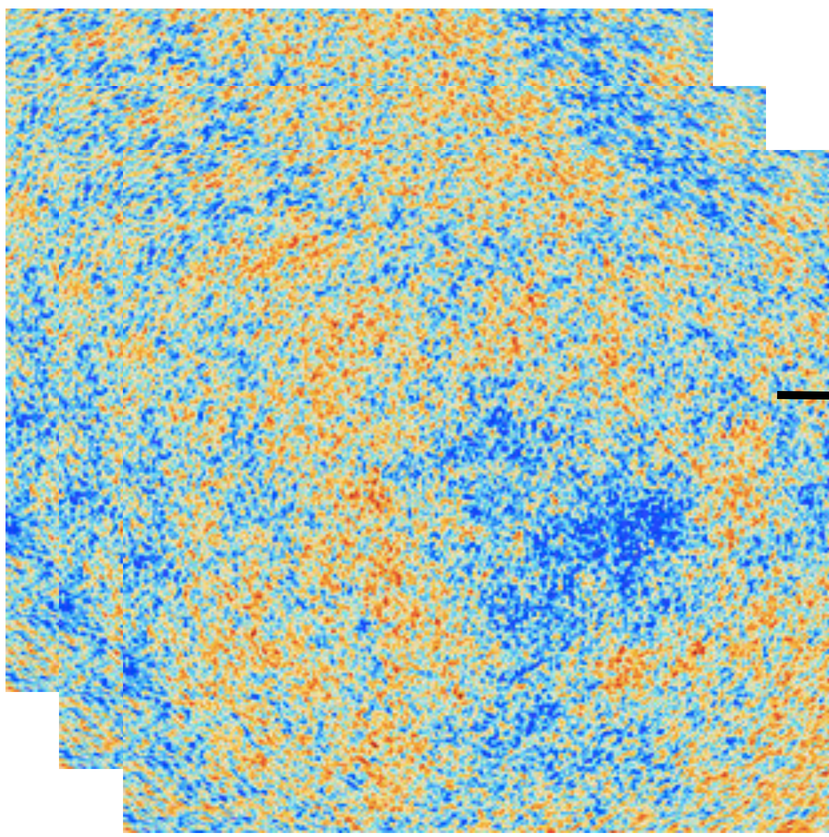
at z=0.5



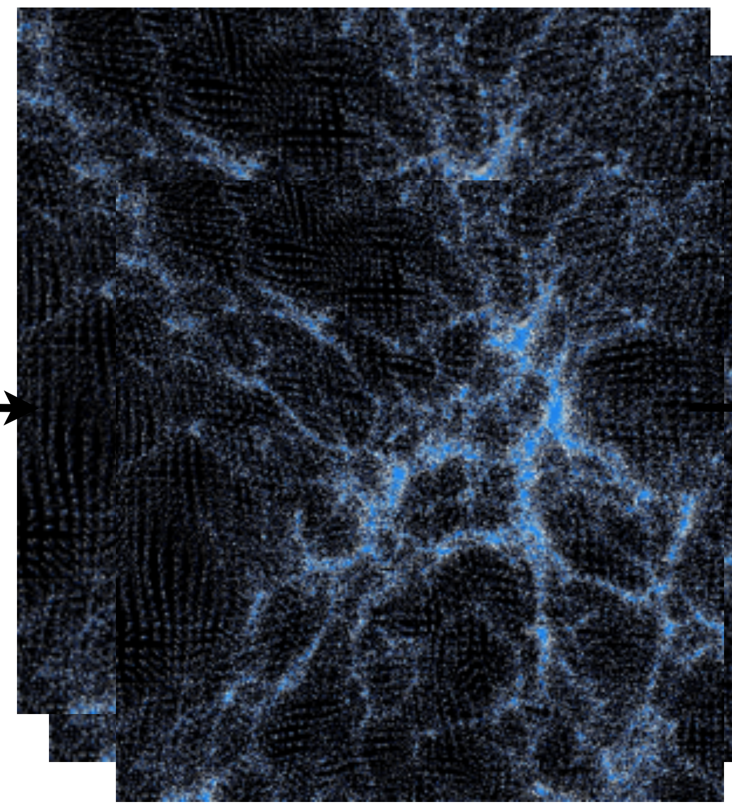
2LPT to z=0.5

Now the question is:
can we learn how to evolve from the 2LPT field to Nbody density field?

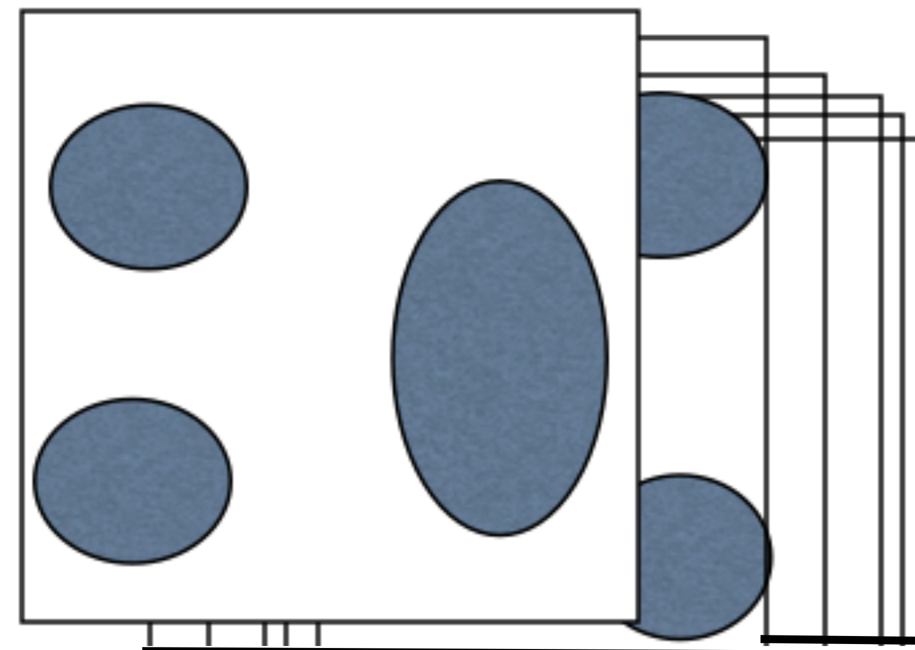
multiple initial conditions



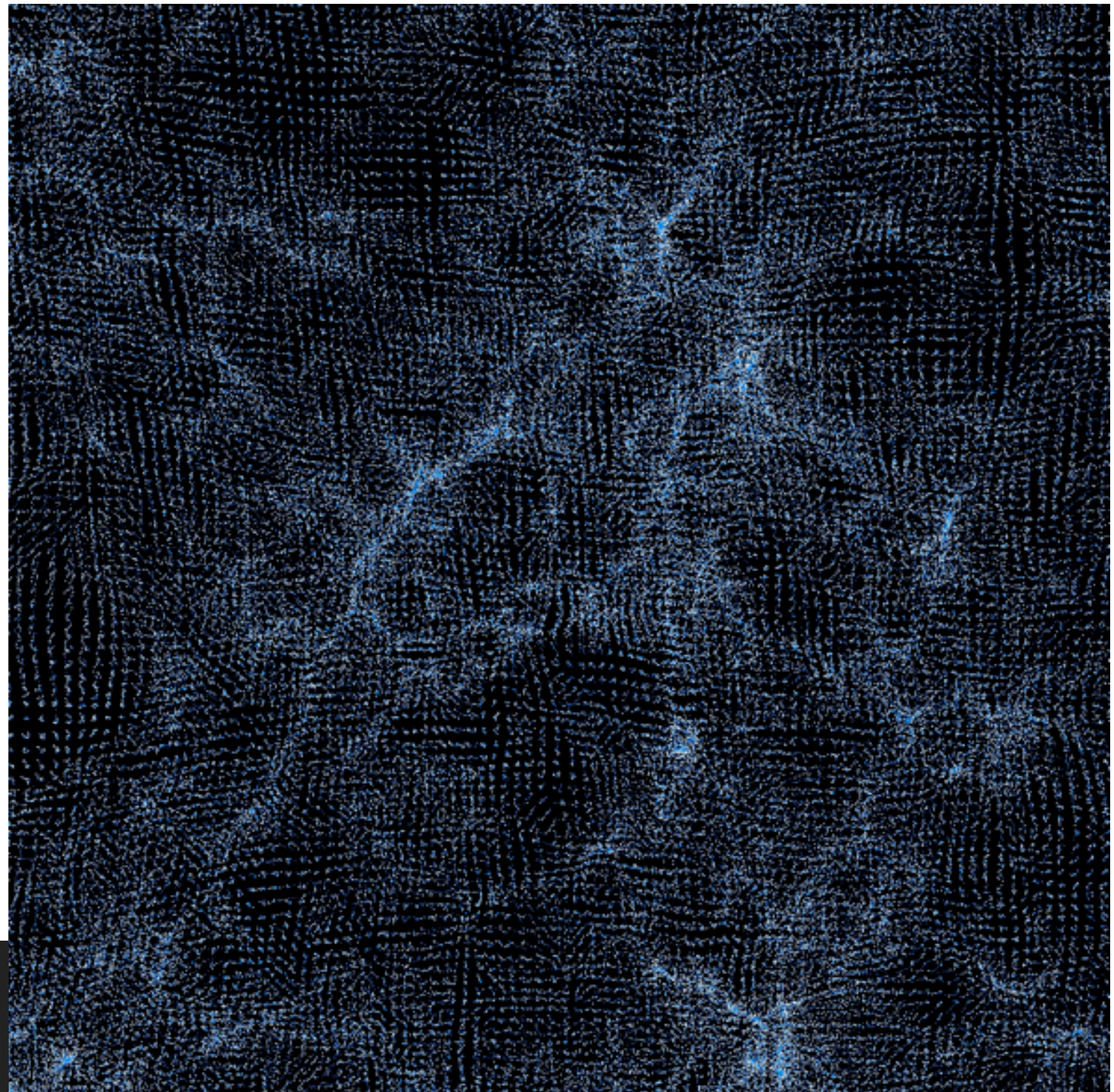
2LPT to z=0.5



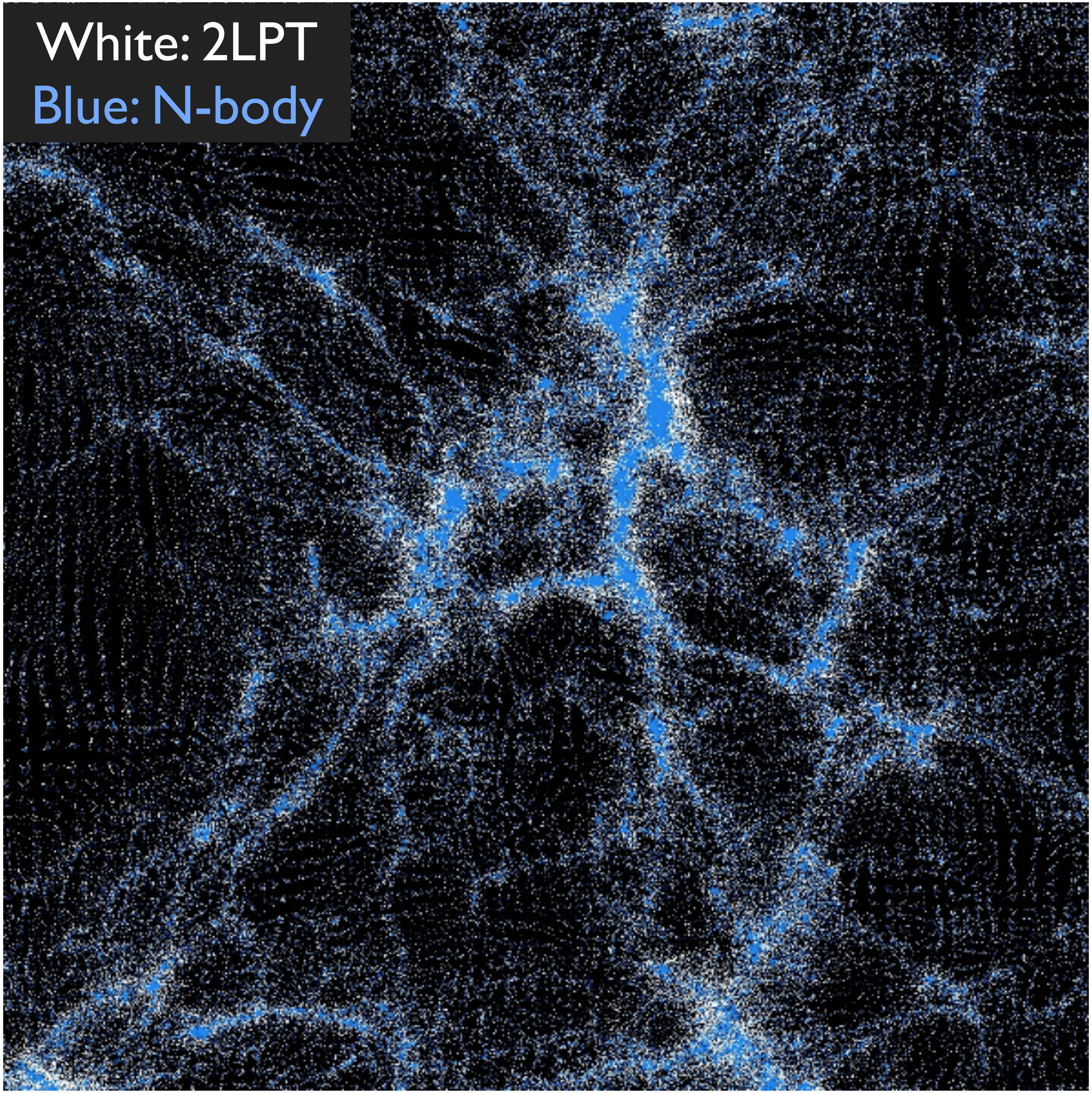
Multiple near N-body simulations



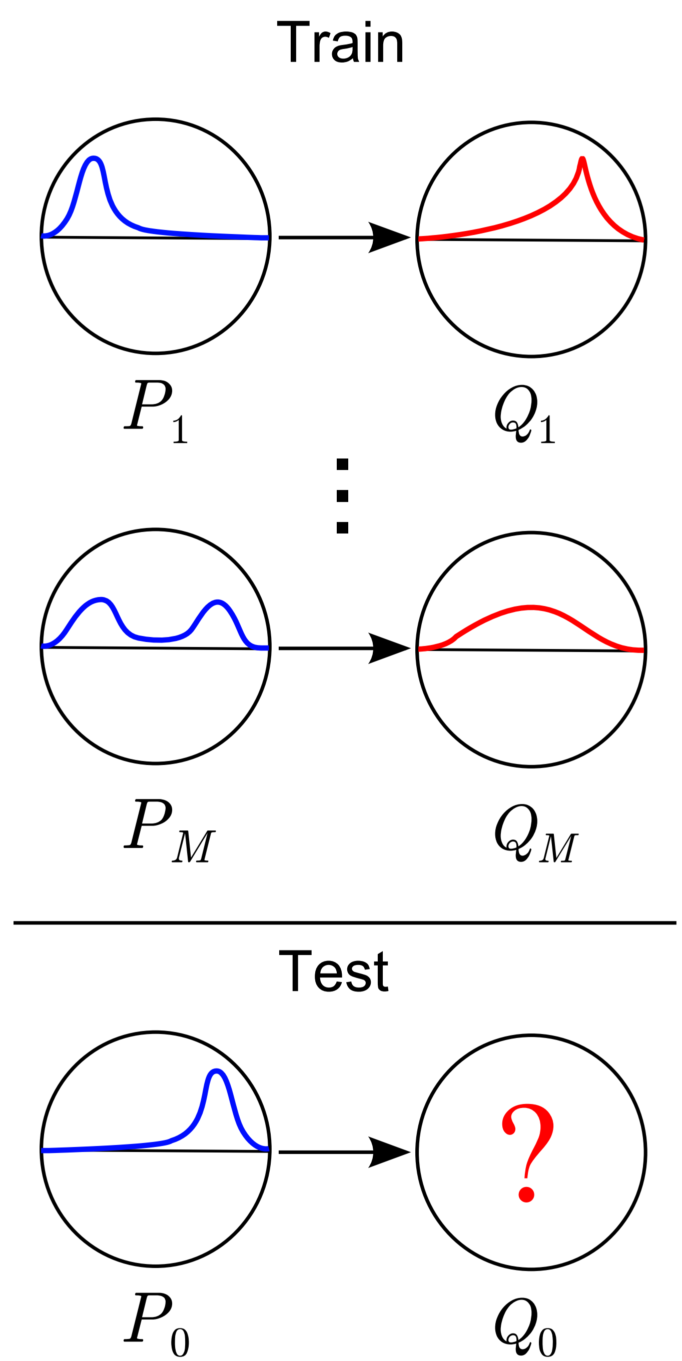
How does 2LPT field compares to N-body field?



White: 2LPT
Blue: N-body

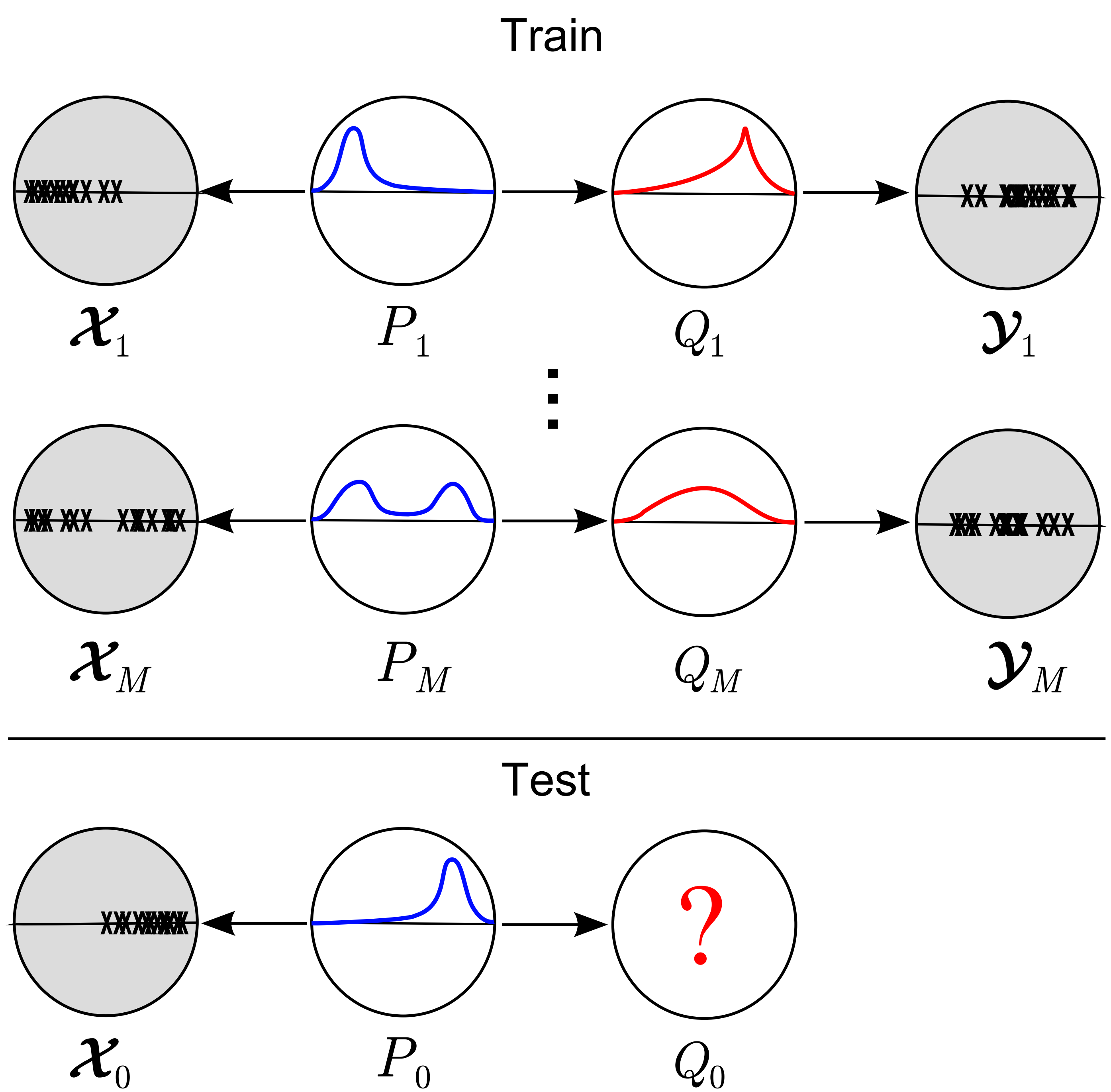

White: 2LPT
Blue: N-body

(2LPT) Distribution to (N-body) Distribution Machine Learning algorithms



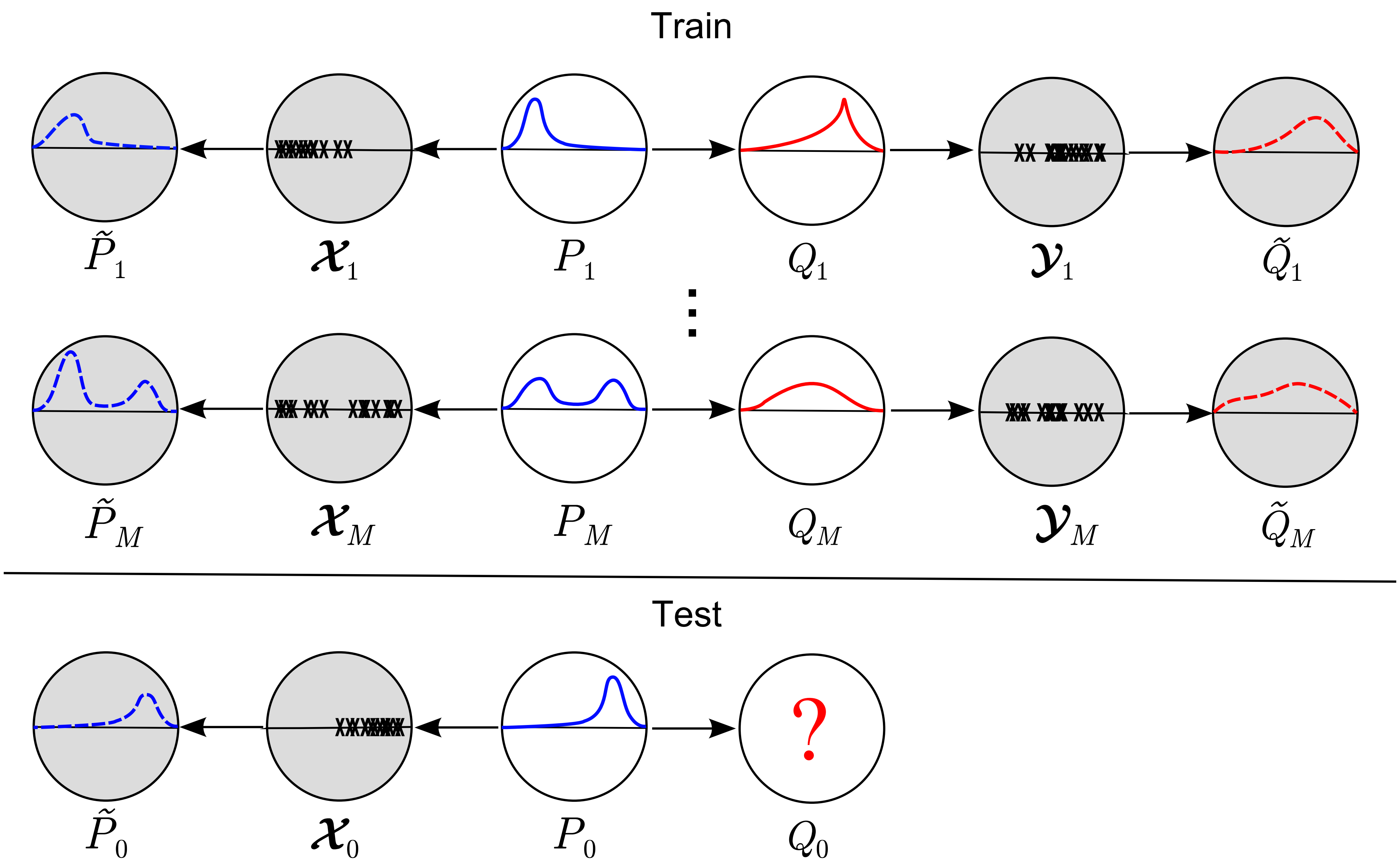
courtesy Junier Oliver

(2LPT) Distribution to (N-body) Distribution Machine Learning algorithms

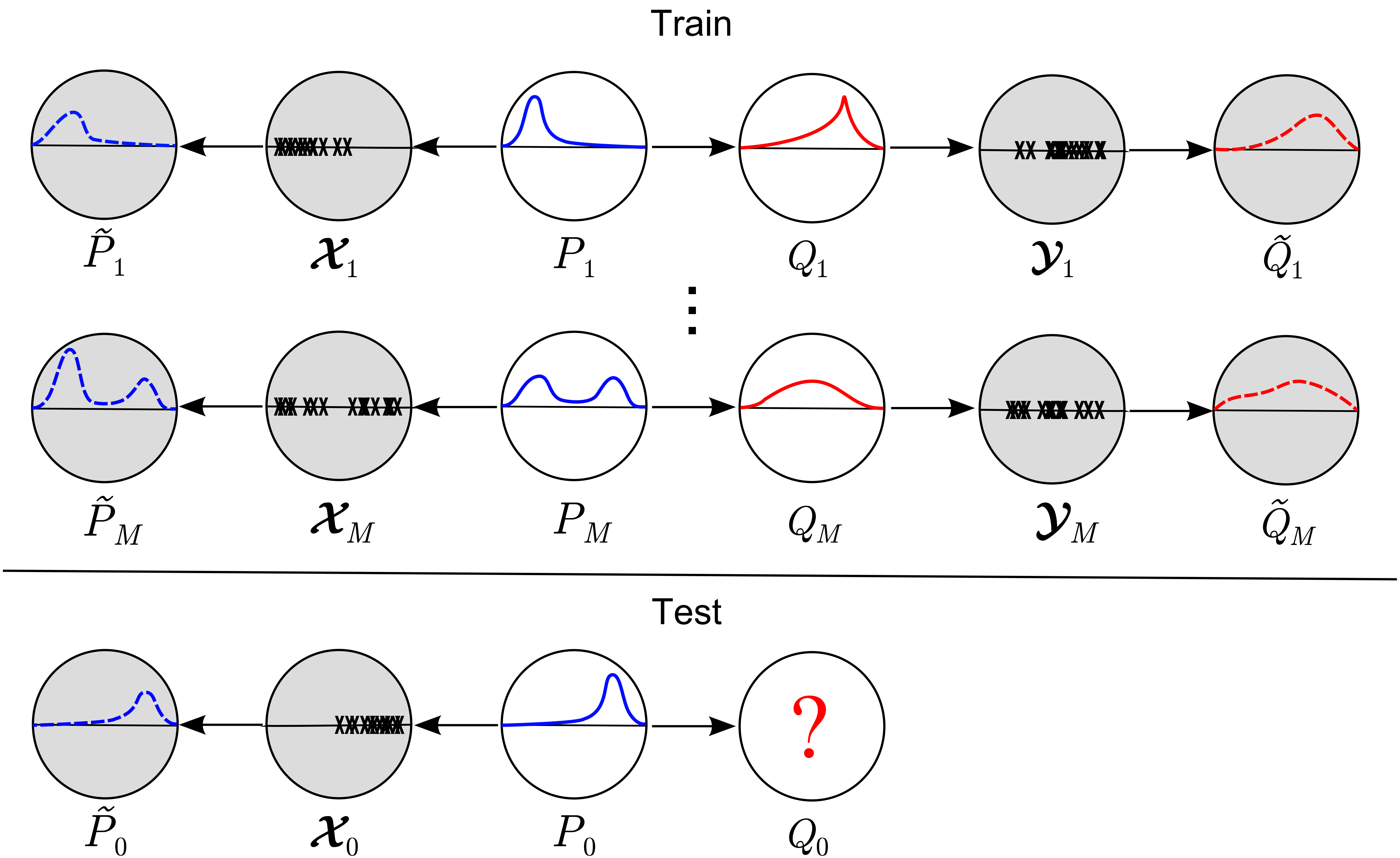


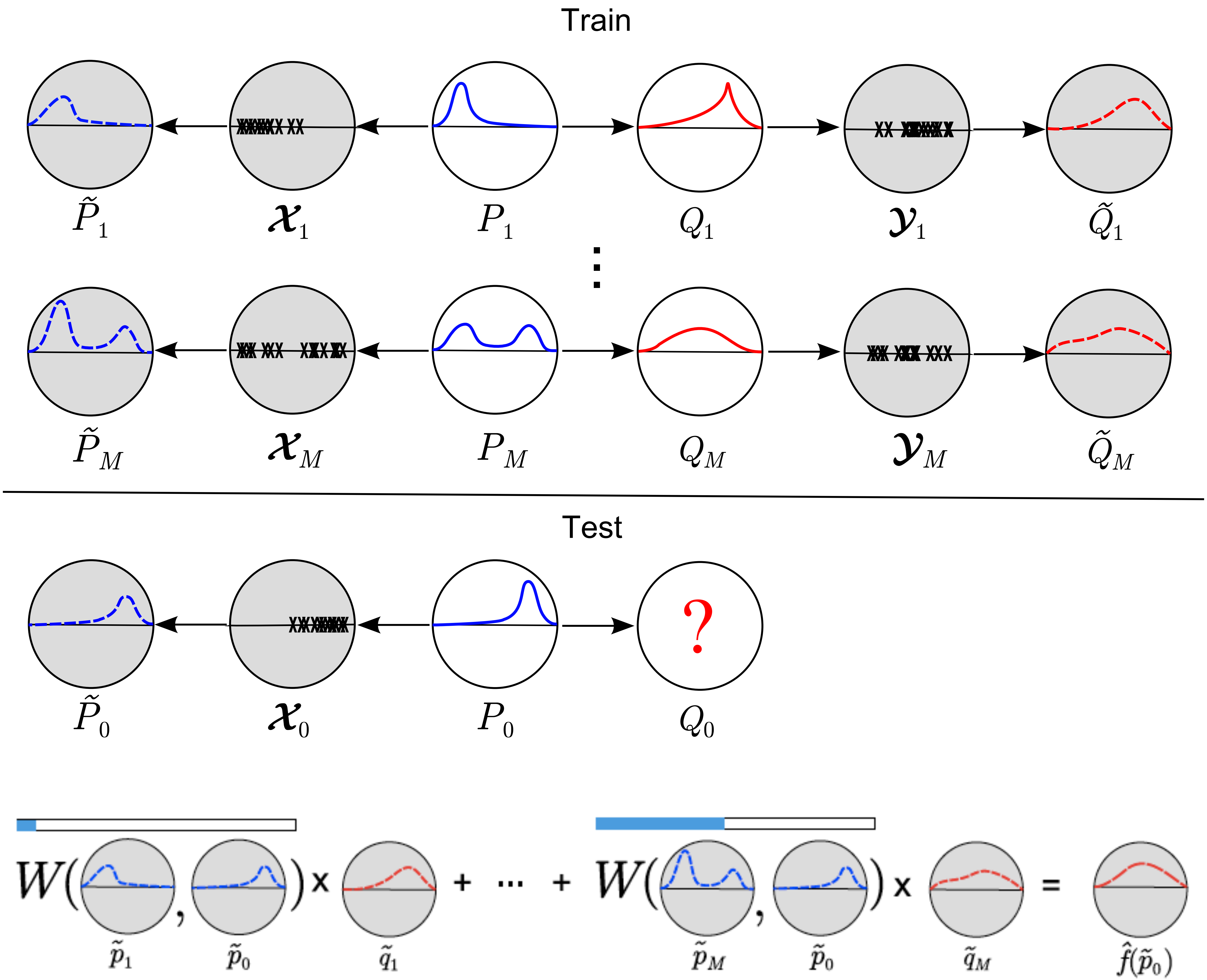
courtesy Junier Oliver

(2LPT) Distribution to (N-body) Distribution Machine Learning algorithms



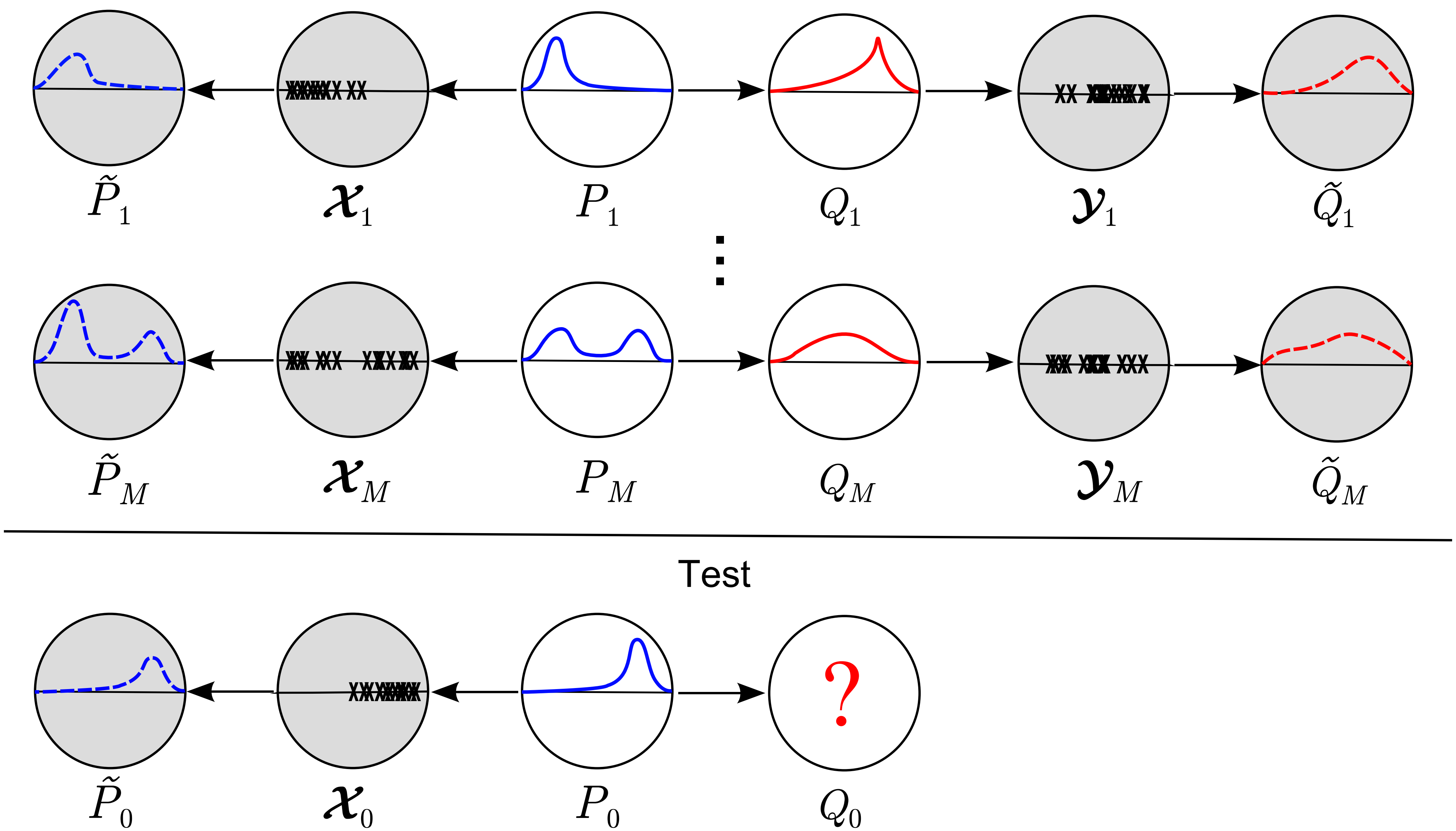
courtesy Junier Oliver


Train
$\tilde{P}_1$
$\mathcal{X}_1$
P_1
Q_1
$\mathcal{Y}_1$
$\tilde{Q}_1$
$\vdots$
$\tilde{P}_M$
$\mathcal{X}_M$
P_M
Q_M
$\mathcal{Y}_M$
$\tilde{Q}_M$
Test
$\tilde{P}_0$
$\mathcal{X}_0$
P_0
?
Q_0

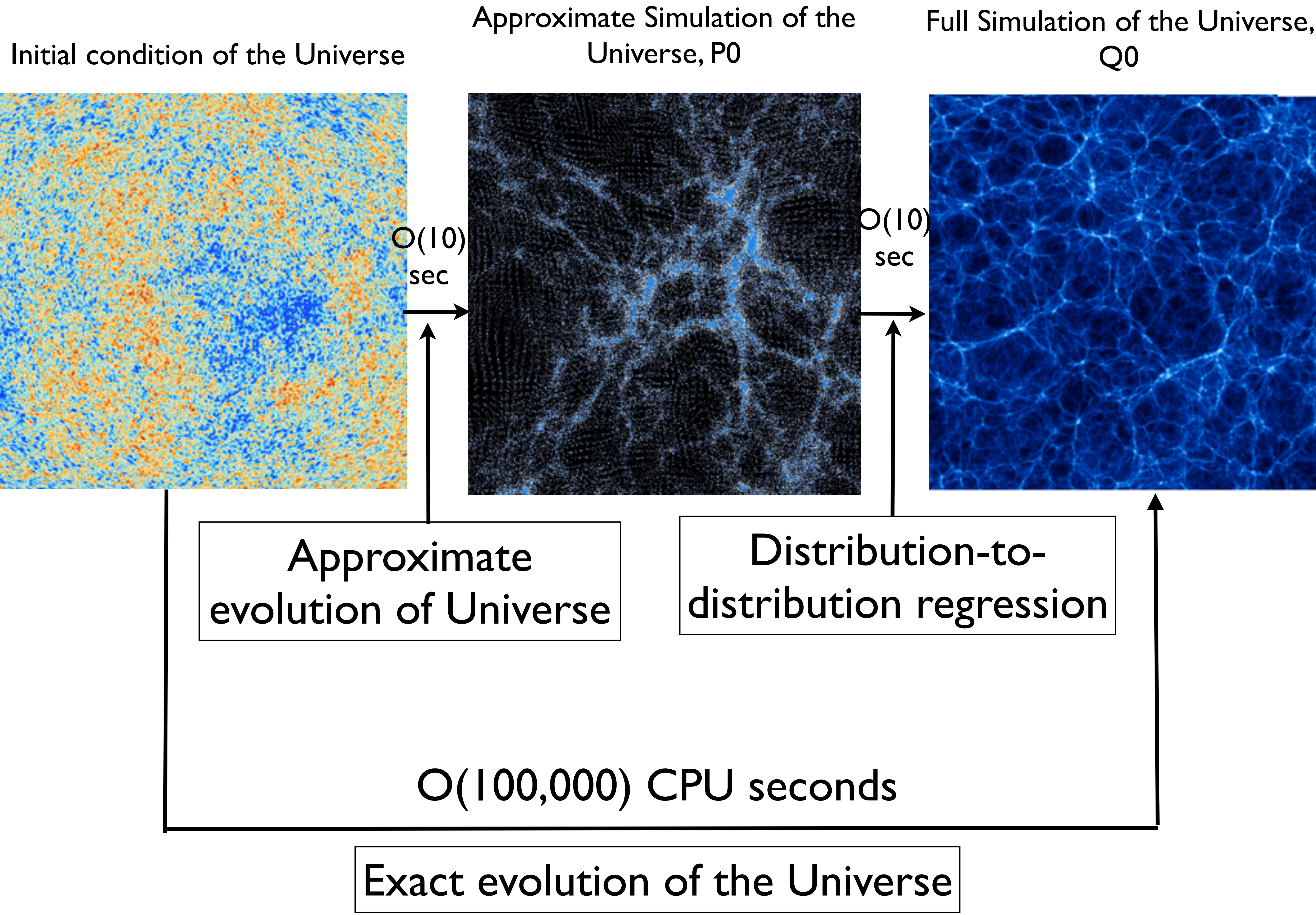

Train
$\tilde{P}_1$ $\mathcal{X}_1$ P_1 Q_1 $\mathcal{Y}_1$ $\tilde{Q}_1$
$\tilde{P}_M$ $\mathcal{X}_M$ P_M Q_M $\mathcal{Y}_M$ $\tilde{Q}_M$
Test
?
$\tilde{P}_0$ $\mathcal{X}_0$ P_0 Q_0
$W(\tilde{p}_1, \tilde{p}_0) \times \tilde{q}_1 + \cdots + W(\tilde{p}_M, \tilde{p}_0) \times \tilde{q}_M = \hat{f}(\tilde{p}_0)$

(2LPT) Distribution to (N-body) Distribution Machine Learning algorithms

2LPT (P) -> N-body(Q)



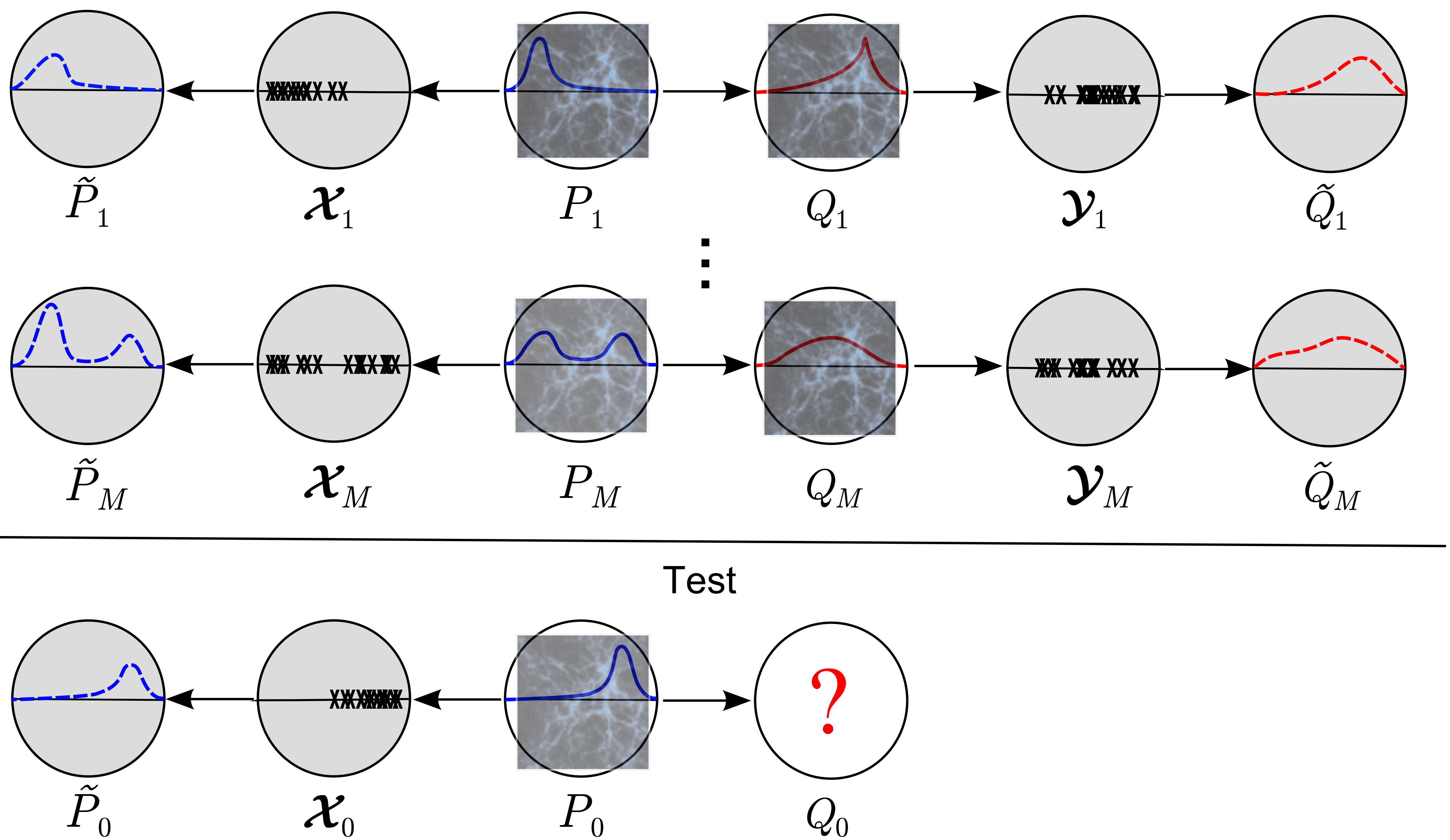
courtesy Junier Oliver



Initial condition of the Universe
Approximate Simulation of the Universe, P0
Full Simulation of the Universe, Q0
O(10) sec
O(10) sec
Approximate evolution of Universe
Distribution-to-distribution regression
O(100,000) CPU seconds
Exact evolution of the Universe

(2LPT) Distribution to (N-body) Distribution
Machine Learning algorithms

2LPT (P) -> N-body(Q)



Summary

- Making many similar synthetic Universes (same cosmology) of dark matter: possibly sped up by Machine Learning by a factor of 1000!
- Instead of running very high resolution simulation or extremely complicated semi-analytical modeling, we can machine learn from either high resolution sims or actual observations to create our own halo model (of not only galaxies, but may be many other things that we have trouble modeling!)

Observations:

flux(x,y,band/wavelength), observational systematics(x,y,band/wavelength): psf, sky, dust, airmass and respective errors

Basic Reduction pipeline

Extracted object, observational systematics properties

LSS systematics removal, statistics estimation

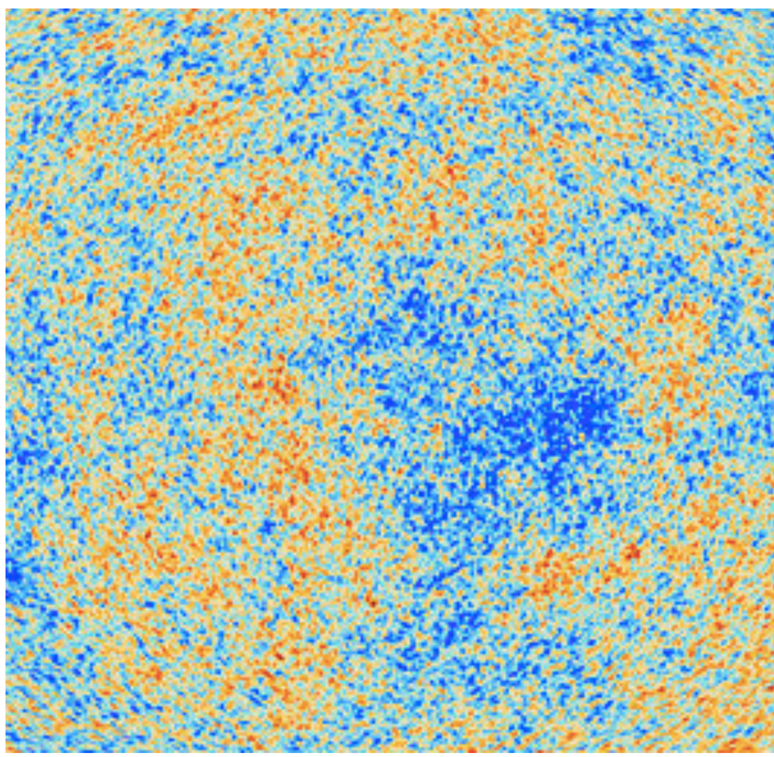
Large Scale Structure (BAO, full clustering measurements)

Theory, Beyond Linear Model Predictions

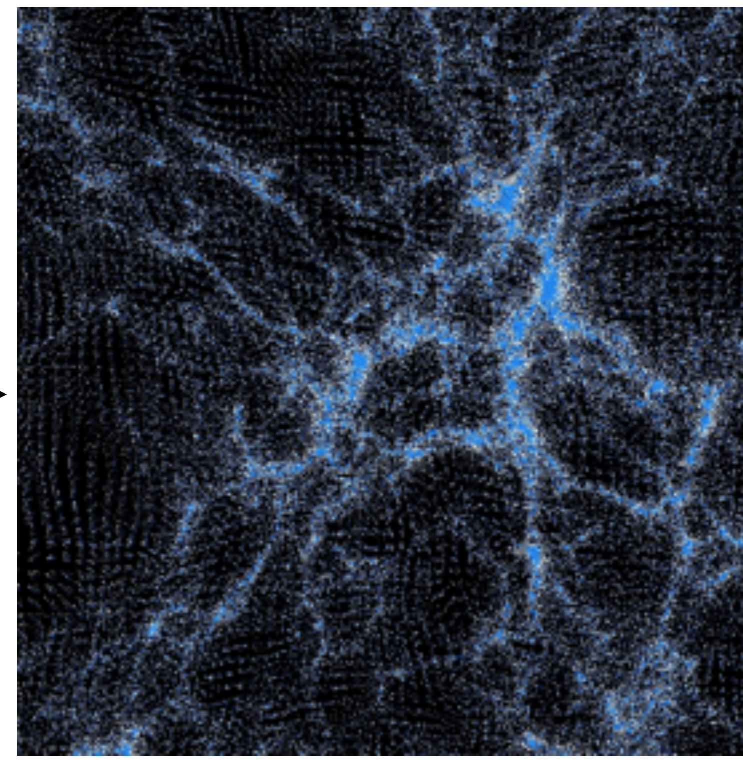
Simulations (possibly with DM simulations + HOD), Covariance Matrix

Cosmology (cosmological parameters, formation and evolution of galaxies, quasars...)

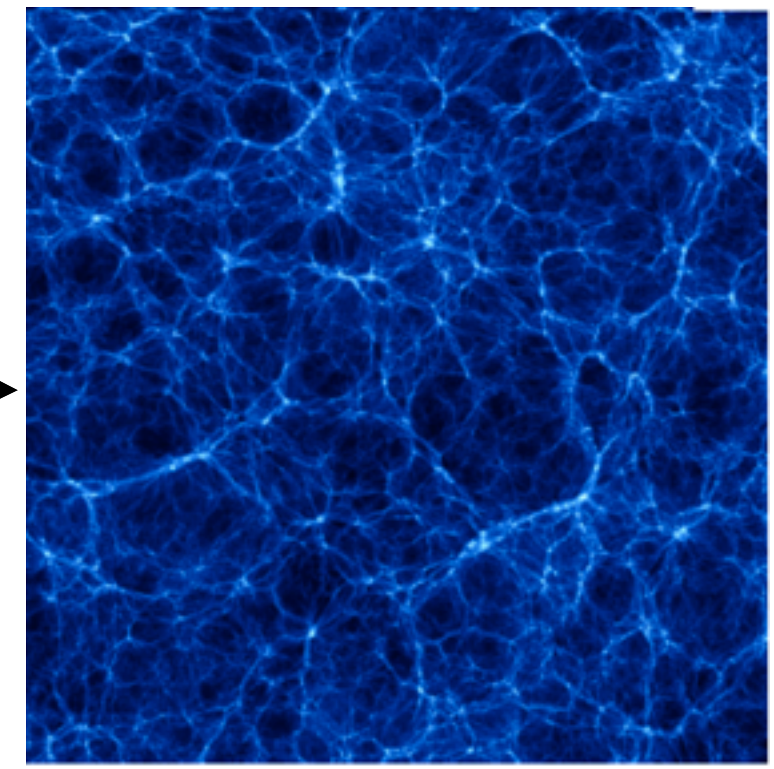
Initial condition of the Universe



Approximate Simulation of the Universe, P_0

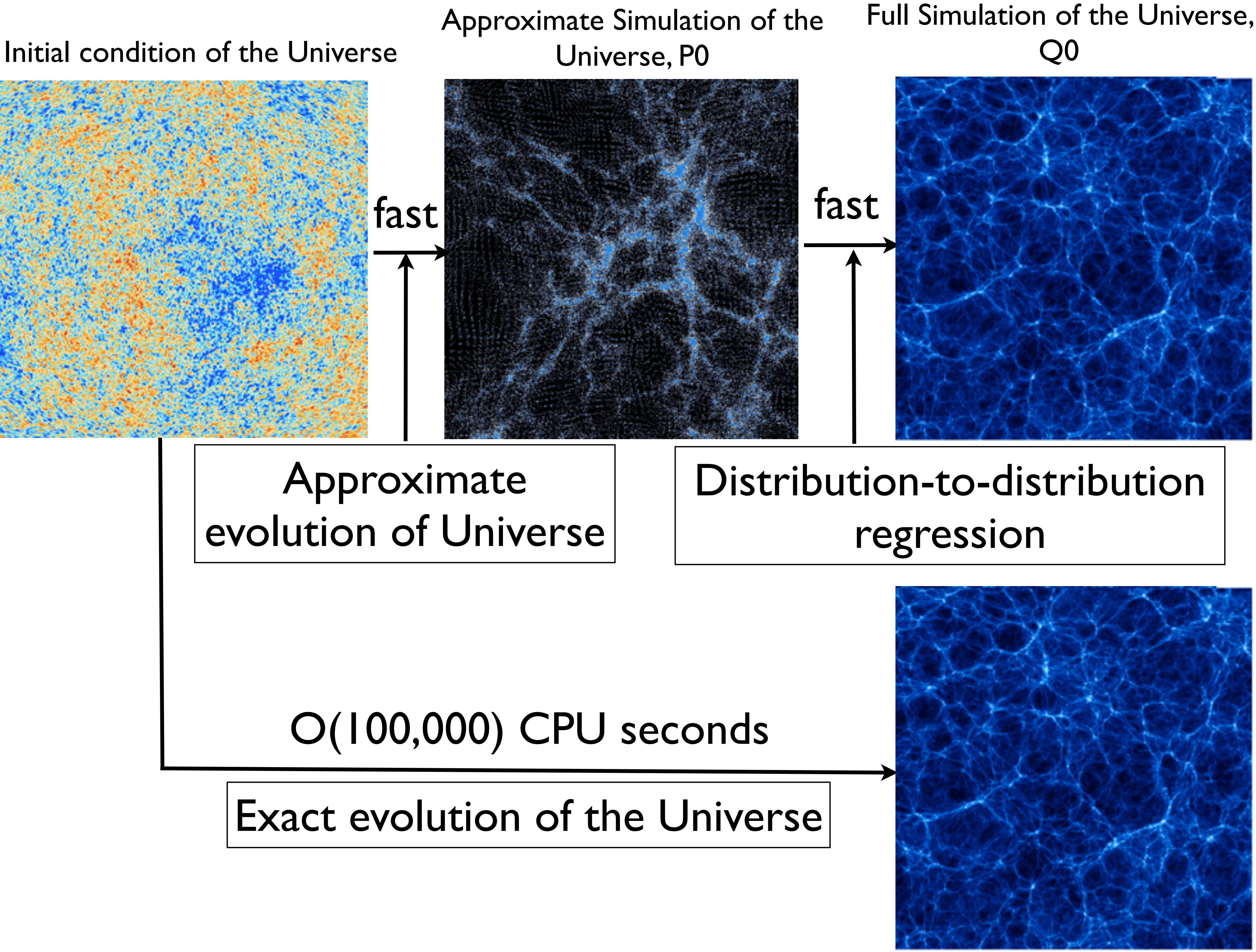


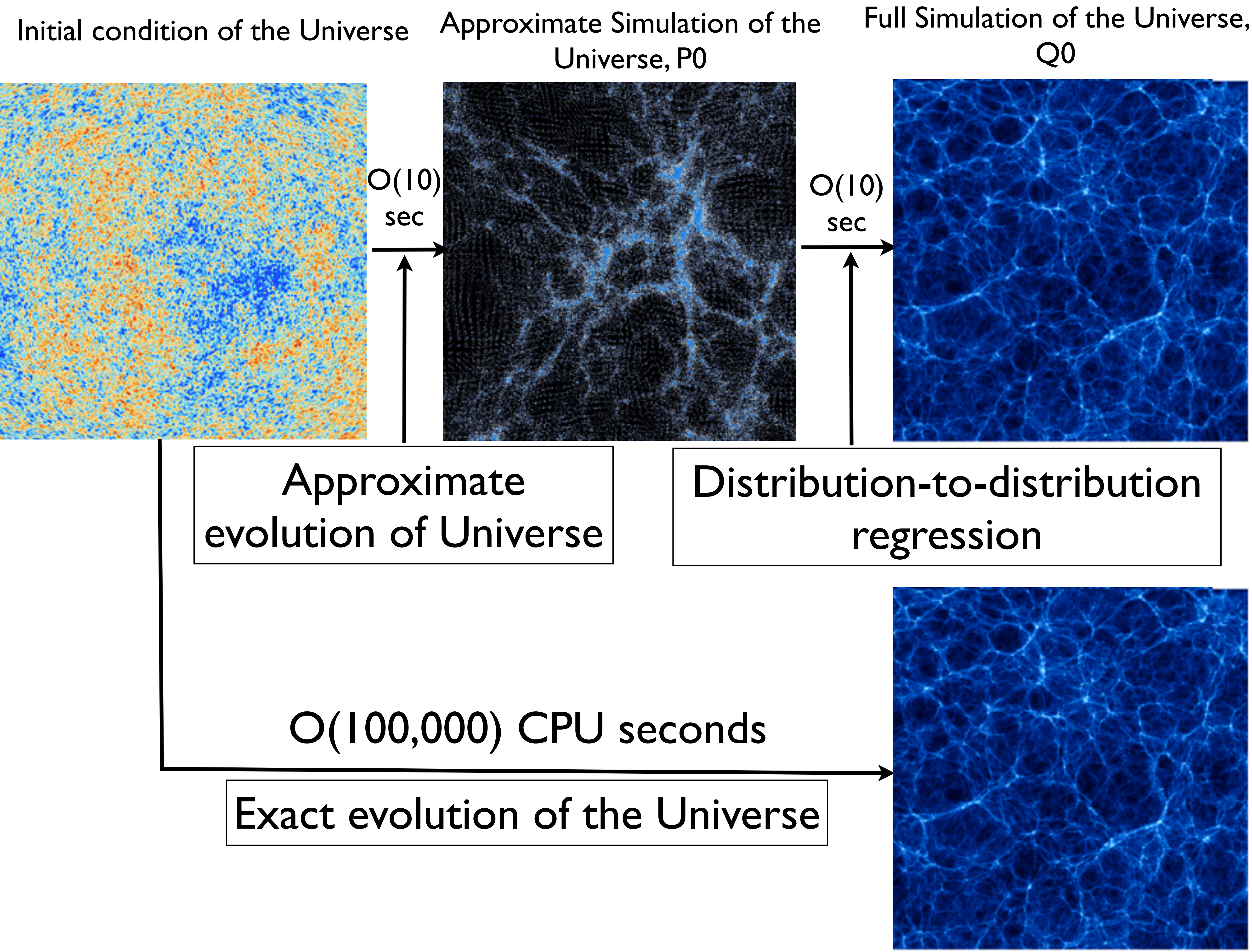
Full Simulation of the Universe, Q_0

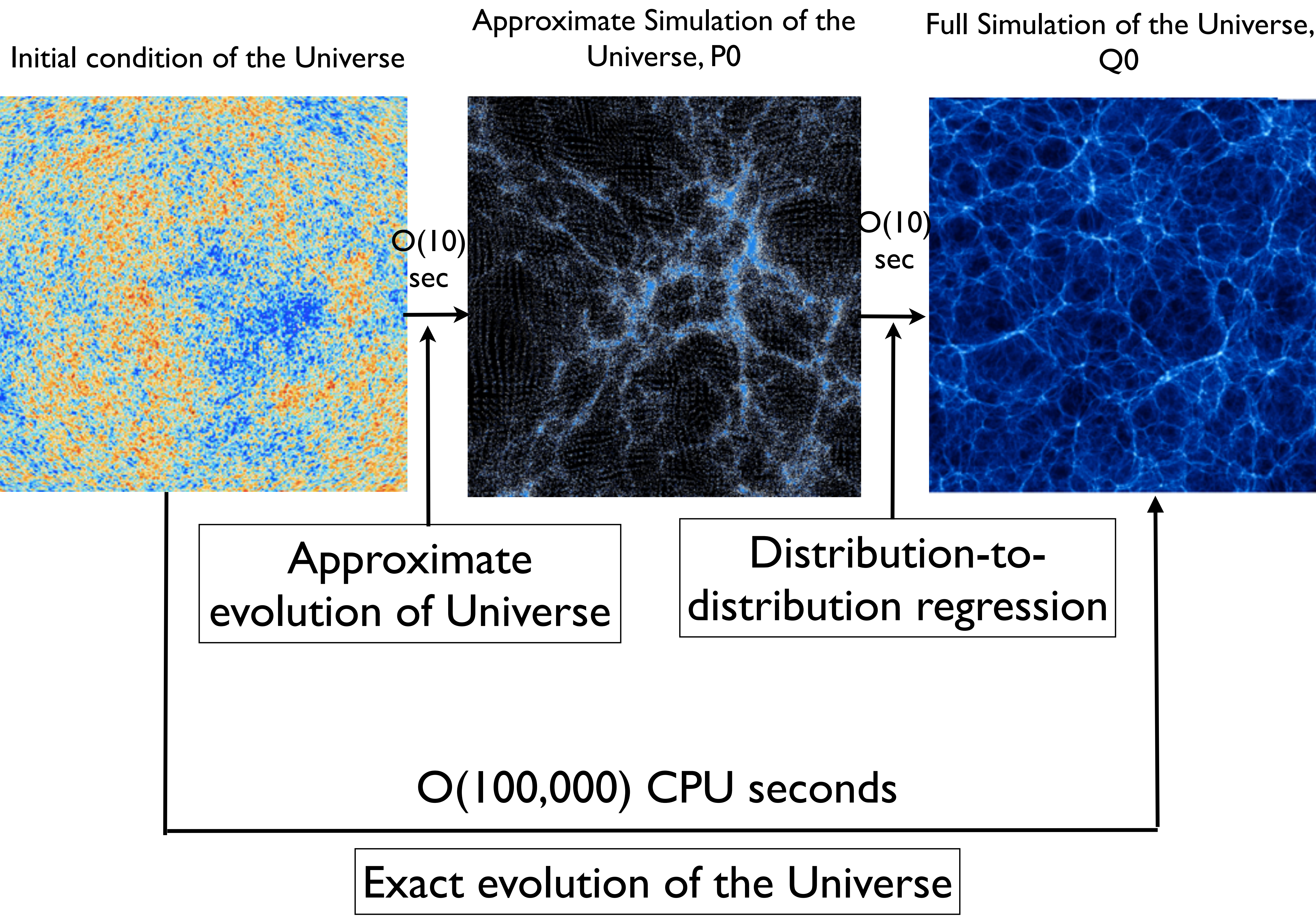


Approximate evolution of Universe with simple Physics

Machine Learning Distribution-to-distribution regression


Initial condition of the Universe
Approximate Simulation of the Universe, P0
Full Simulation of the Universe, Q0
fast
fast
Approximate evolution of Universe
Distribution-to-distribution regression
O(100,000) CPU seconds
Exact evolution of the Universe


Initial condition of the Universe
Approximate Simulation of the Universe, P0
Full Simulation of the Universe, Q0
O(10) sec
O(10) sec
Approximate evolution of Universe
Distribution-to-distribution regression
O(100,000) CPU seconds
Exact evolution of the Universe


Initial condition of the Universe
Approximate Simulation of the Universe, P0
Full Simulation of the Universe, Q0
O(10) sec
O(10) sec
Approximate evolution of Universe
Distribution-to-distribution regression
O(100,000) CPU seconds
Exact evolution of the Universe