fit-predicted-alpha

June 15, 2021

1 Setup

[12]:

```
%load_ext autoreload
%autoreload 2
%aimport
```

```
The autoreload extension is already loaded. To reload it, use:
  %reload_ext autoreload
Modules to reload:
all-except-skipped

Modules to skip:
```

[13]:

```
%matplotlib inline
```

[21]:

```
import astropy
from pathlib import Path
import numpy as np
import matplotlib.pyplot as plt
import re
from astropy.table import Table
import astropy.table
import json
from scipy import stats
from copy import deepcopy
import warnings


from relaxed import halo_parameters, halo_catalogs, analysis
from relaxed.plot_defaults import *
```

2 Fits to obtain alpha

With Phil, I already discussed how adding β to the fitting procedure does not contribute much because it has a strong degeneracy with α. So in the following notebook we only use α for the fit.

Notes:

- There are two data points that don't reach a_{1/2} ever, this explains the runtime error below when obtaining a_{1/2}

[4]:
```
from relaxed.progenitors.catalog import get_alpha, lma_fit
from relaxed.analysis import setup, get_ma
from relaxed.halo_parameters import get_hparam
```

[5]:
```
hcat, indices, scales = setup('m11')
zs = (1/ scales) -1
ma = get_ma(hcat.cat, indices)
lma = np.log(ma)[:, :160] # towards very big scales many masses become really␣
 ↪close to 0.

# remove infs
keep = np.zeros(len(hcat.cat))
for indx in range(len(lma)):
    if np.isinf(lma[indx]).sum() == 0 and np.isnan(lma[indx]).sum() == 0:
        keep[indx] = 1

keep = keep.astype(bool)
hcat.cat = hcat.cat[keep]
ma = ma[keep]
lma = lma[keep]
n = lma.shape[1]
zs = zs[:n]
print(len(hcat.cat))
print(lma.shape)
print(zs.shape)
```

```
/home/imendoza/miniconda3/envs/alcca/lib/python3.8/site-
packages/astropy/table/column.py:1020: RuntimeWarning: invalid value encountered
in greater
  result = getattr(super(), op)(other)
```

```
9615
(9615, 160)
(160,)
```

[6]:
```
# calculate all alpha, betas and add them to table.
# log m(z) = - \alpha * z
alphas = []

for idx in range(len(hcat.cat)):
    alpha = get_alpha(zs, lma[idx])
    alphas.append(alpha)
c1 = astropy.table.Column(alphas, name='alpha')
```

```
hcat.cat.add_column(c1)
```

[7]:

```
# add a1/2
from relaxed.analysis import get_a2
a2 = get_a2(hcat.cat, scales, indices)
c = astropy.table.Column(a2, name='a2')
hcat.cat.add_column(c)
```

```
/home/imendoza/nbody-relaxed/relaxed/analysis.py:175: RuntimeWarning: invalid
value encountered in less
  idx = np.argmax(np.where(ma < 0.5, ma, -np.inf), 1)
```

3 Examples of fitting accretion history

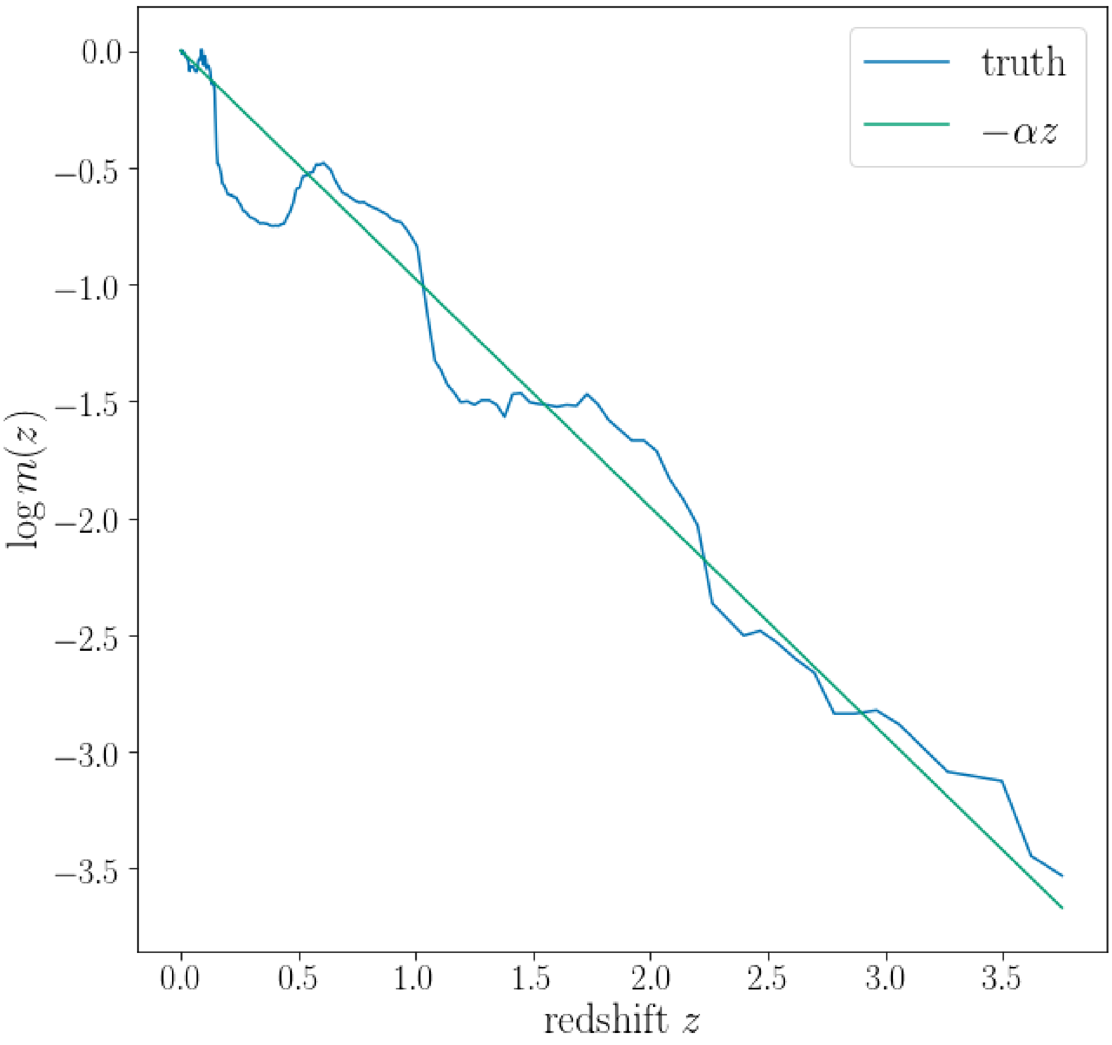
3.1 m(z)

[32]:

```
idx = 1
fit = -alphas[idx] * zs
plt.plot(zs, lma[idx], color=colors[0], label=r'\rm truth')
plt.plot(zs, fit, color=colors[1], label=r'\rm $-\alpha z$')
plt.xlabel(r'\rm redshift $z$')
plt.ylabel(r'$\log m(z)$')
plt.legend()
```

[32]:

```
<matplotlib.legend.Legend at 0x2b3c3f31a940>
```



3.2 a(m)

[38]:

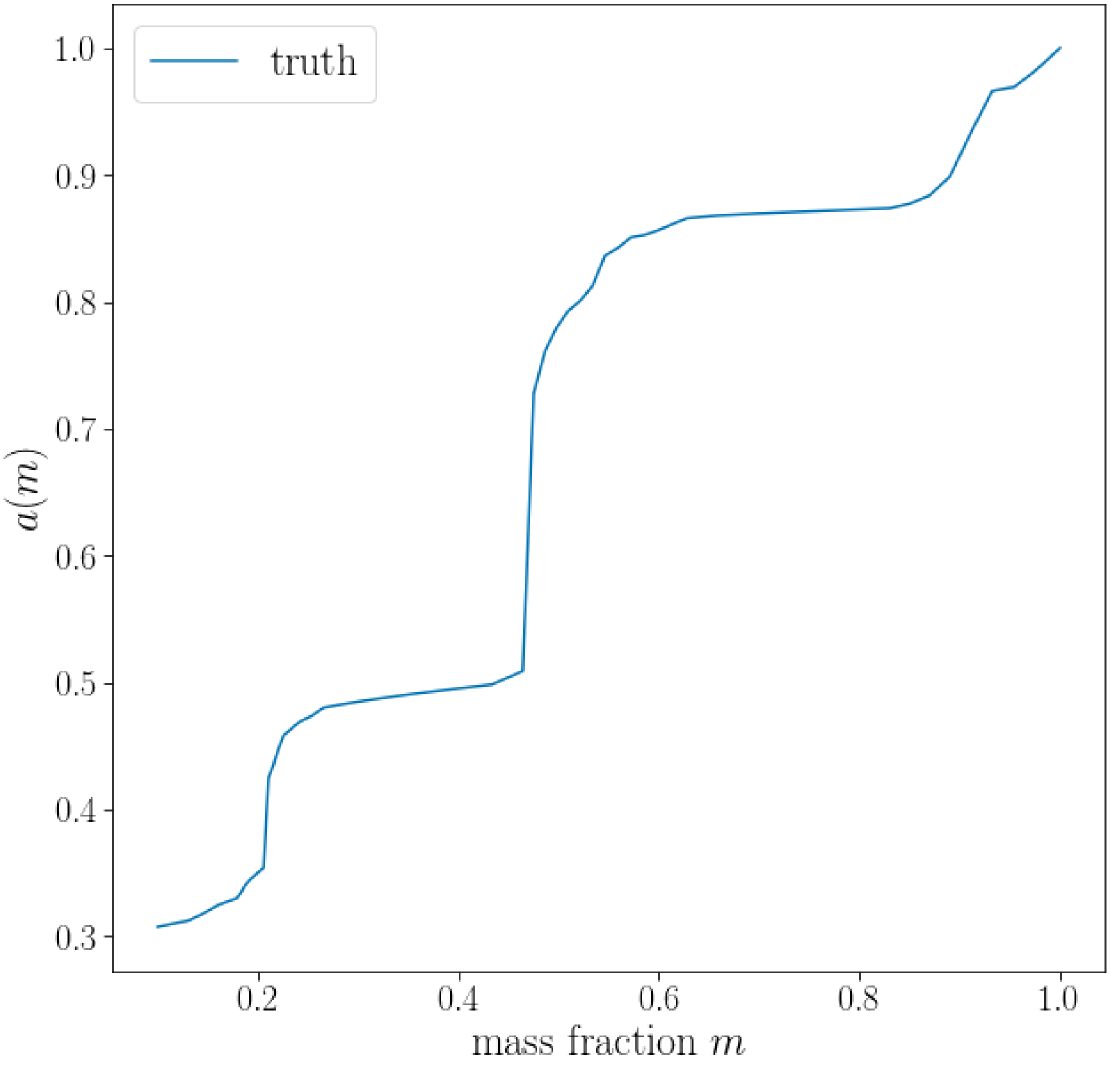
```
am,  mass_bins = analysis.get_am('m11', min_mass=0.1, path='../../temp/')
am = am[keep]
print(len(am))
```

```
9615
```

[40]:

```
idx = 1
plt.plot(mass_bins, am[idx], color=colors[0], label=r'\rm truth')
plt.xlabel(r'\rm mass fraction $m$')
plt.ylabel(r'$a(m)$')
plt.legend()
```

```
[40]: <matplotlib.legend.Legend at 0x2b3c3ed01c70>
```



4 Plot more complete correlation matrix

```
[32]: plt.rc('text', usetex=True)
```

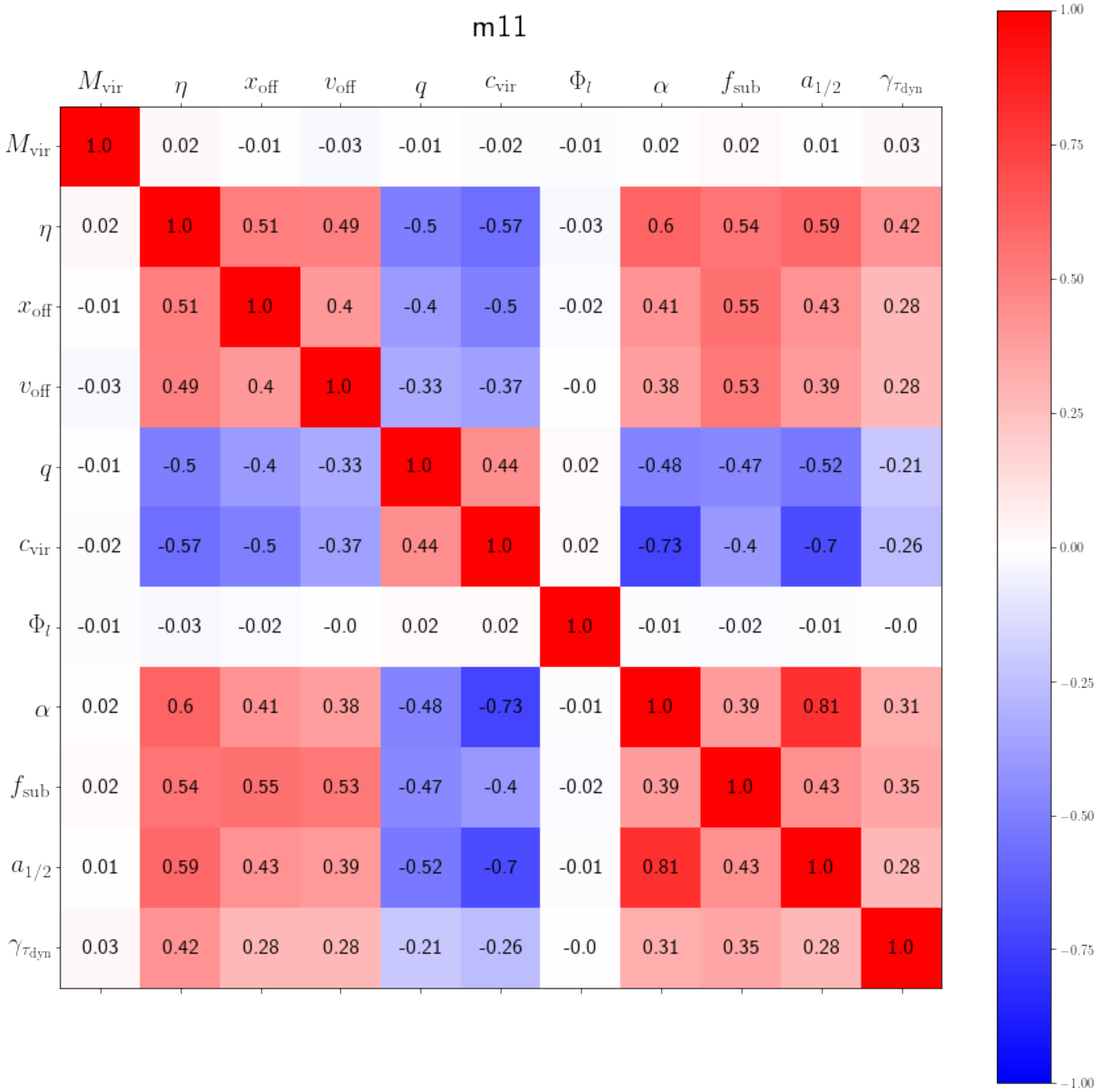
```
[33]: from scipy.stats import spearmanr

      def plot_correlation_matrix(ax, values, cell_text_size=14):
          # values: list of values of ech para
          n_params = len(values)
          matrix = np.zeros((n_params, n_params))
```

```
    for i, value1 in enumerate(values):
        for j, value2 in enumerate(values):
            matrix[i, j] = spearmanr(value1, value2)[0]

    # mask out lower off-diagonal elements if requested.
    im = ax.matshow(matrix, cmap="bwr", vmin=-1, vmax=1)
    plt.colorbar(im, ax=ax)

    # add the values to each cell
    for i in range(matrix.shape[0]):
        for j in range(matrix.shape[1]):
            _ = ax.text(
                j,
                i,
                round(matrix[i, j], 2),
                ha="center",
                va="center",
                color="k",
                size=cell_text_size,
            )
```

[34]:

```
params = ["mvir", "eta", "x0", "v0", "q", "cvir", 'phi_l', 'alpha', 'f_sub',␣
 ↪'a2', 'gamma_tdyn']
hparams = {param: get_hparam(param, log=False) for param in params}
names = ['m11', 'm12']

fig, ax = plt.subplots(1, 1, figsize=(15, 15))


latex_params = [
    hparams[param].get_text(only_param=True)
    for param in params
]
values = [
    hparams[param].get_values(hcat.cat)
    for param in params
]

plot_correlation_matrix(ax, values, cell_text_size=16)

ax.set_xticks(range(len(latex_params)))
ax.set_xticklabels(latex_params, size=20)

ax.set_yticks(range(len(latex_params)))
ax.set_yticklabels(latex_params, size=20)

ax.set_title('m11', size=25, pad=20)
```

```
plt.show()
plt.tight_layout()
```



m11

	$M_{\rm vir}$	η	$x_{\rm off}$	$v_{\rm off}$	q	$c_{\rm vir}$	Φ_l	α	$f_{\rm sub}$	$a_{1/2}$	$\gamma_{\tau_{\rm dyn}}$
$M_{\rm vir}$	1.0	0.02	-0.01	-0.03	-0.01	-0.02	-0.01	0.02	0.02	0.01	0.03
η	0.02	1.0	0.51	0.49	-0.5	-0.57	-0.03	0.6	0.54	0.59	0.42
$x_{\rm off}$	-0.01	0.51	1.0	0.4	-0.4	-0.5	-0.02	0.41	0.55	0.43	0.28
$v_{\rm off}$	-0.03	0.49	0.4	1.0	-0.33	-0.37	-0.0	0.38	0.53	0.39	0.28
q	-0.01	-0.5	-0.4	-0.33	1.0	0.44	0.02	-0.48	-0.47	-0.52	-0.21
$c_{\rm vir}$	-0.02	-0.57	-0.5	-0.37	0.44	1.0	0.02	-0.73	-0.4	-0.7	-0.26
Φ_l	-0.01	-0.03	-0.02	-0.0	0.02	0.02	1.0	-0.01	-0.02	-0.01	-0.0
α	0.02	0.6	0.41	0.38	-0.48	-0.73	-0.01	1.0	0.39	0.81	0.31
$f_{\rm sub}$	0.02	0.54	0.55	0.53	-0.47	-0.4	-0.02	0.39	1.0	0.43	0.35
$a_{1/2}$	0.01	0.59	0.43	0.39	-0.52	-0.7	-0.01	0.81	0.43	1.0	0.28
$\gamma_{\tau_{\rm dyn}}$	0.03	0.42	0.28	0.28	-0.21	-0.26	-0.0	0.31	0.35	0.28	1.0

[ ]: