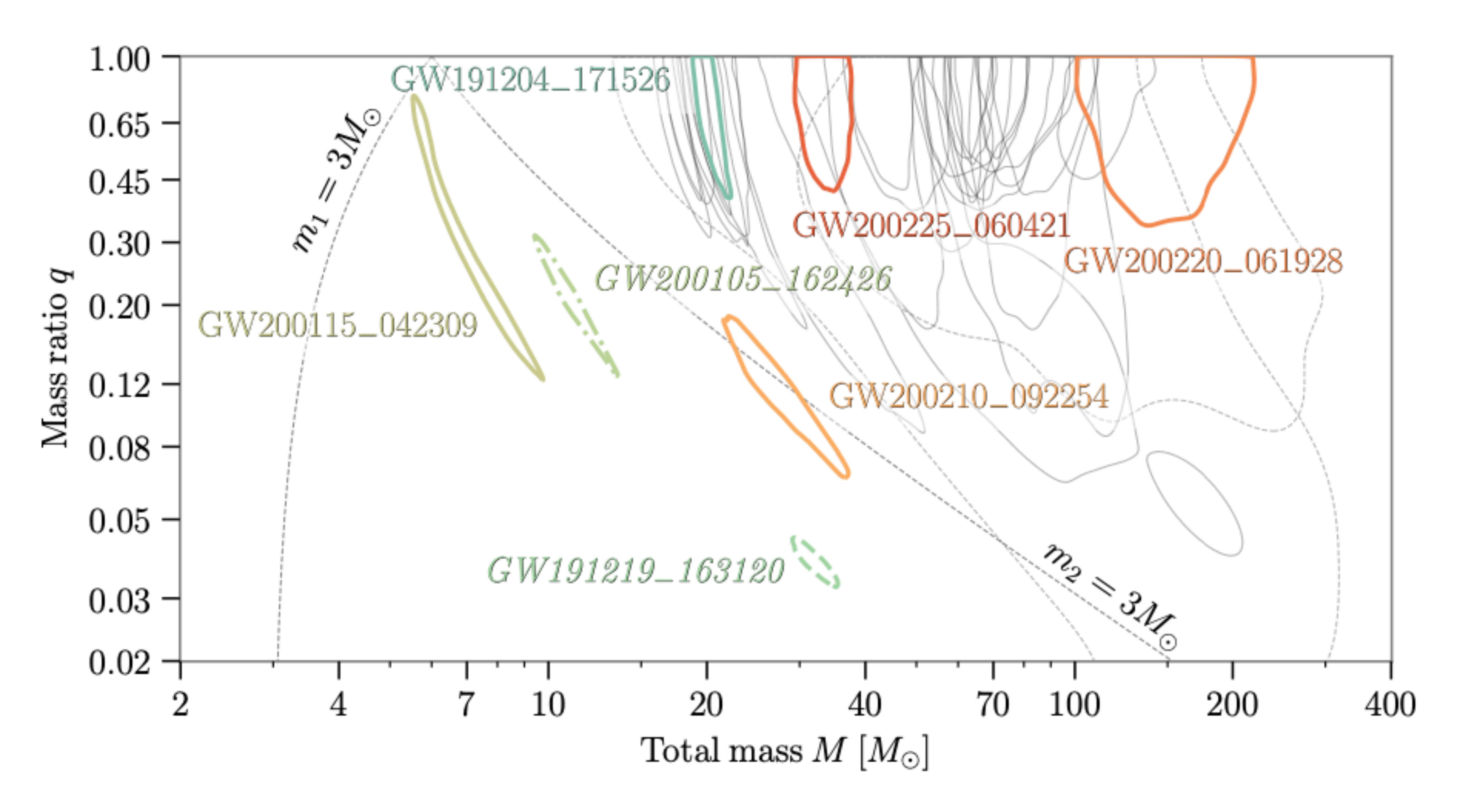
Using Machine Learning Algorithm to search for gravitational waves progenitors

PhD student: Julien Marchioro (APC)
Supervisors : S. Chaty & E. Chassande-Mottin (APC)

How can we train a machine learning algorithm to infer the origin of the low mass binary black holes detected in LIGO/Virgo ?

LIGO/Virgo mission

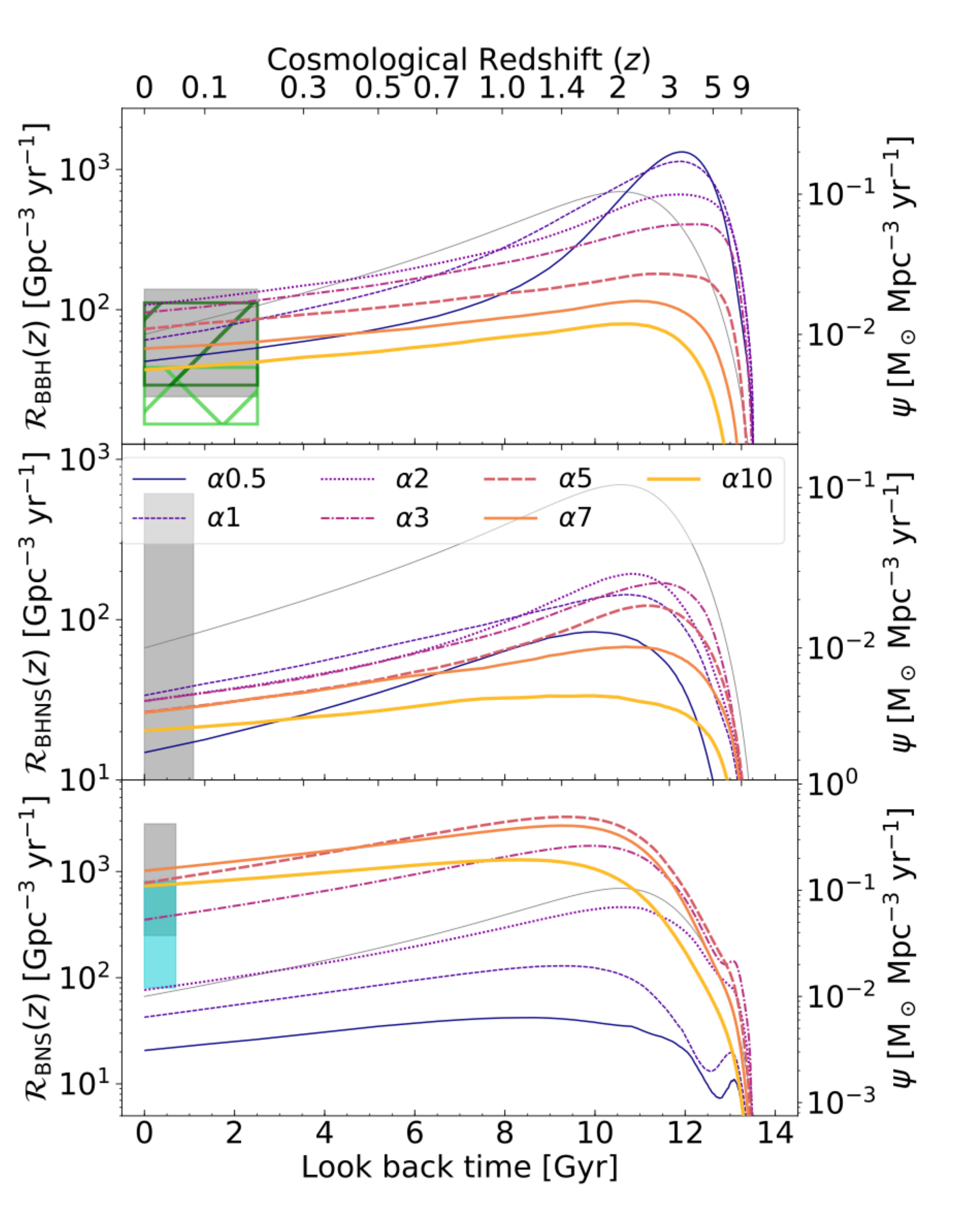
- Number of gravitational waves detections coming from compact binaries coalescence constantly increasing
- Different scenarios for binary black holes (BBH) formation (dynamic or isolated)
- Could we predict the origin of isolated BBH using machine learning ?



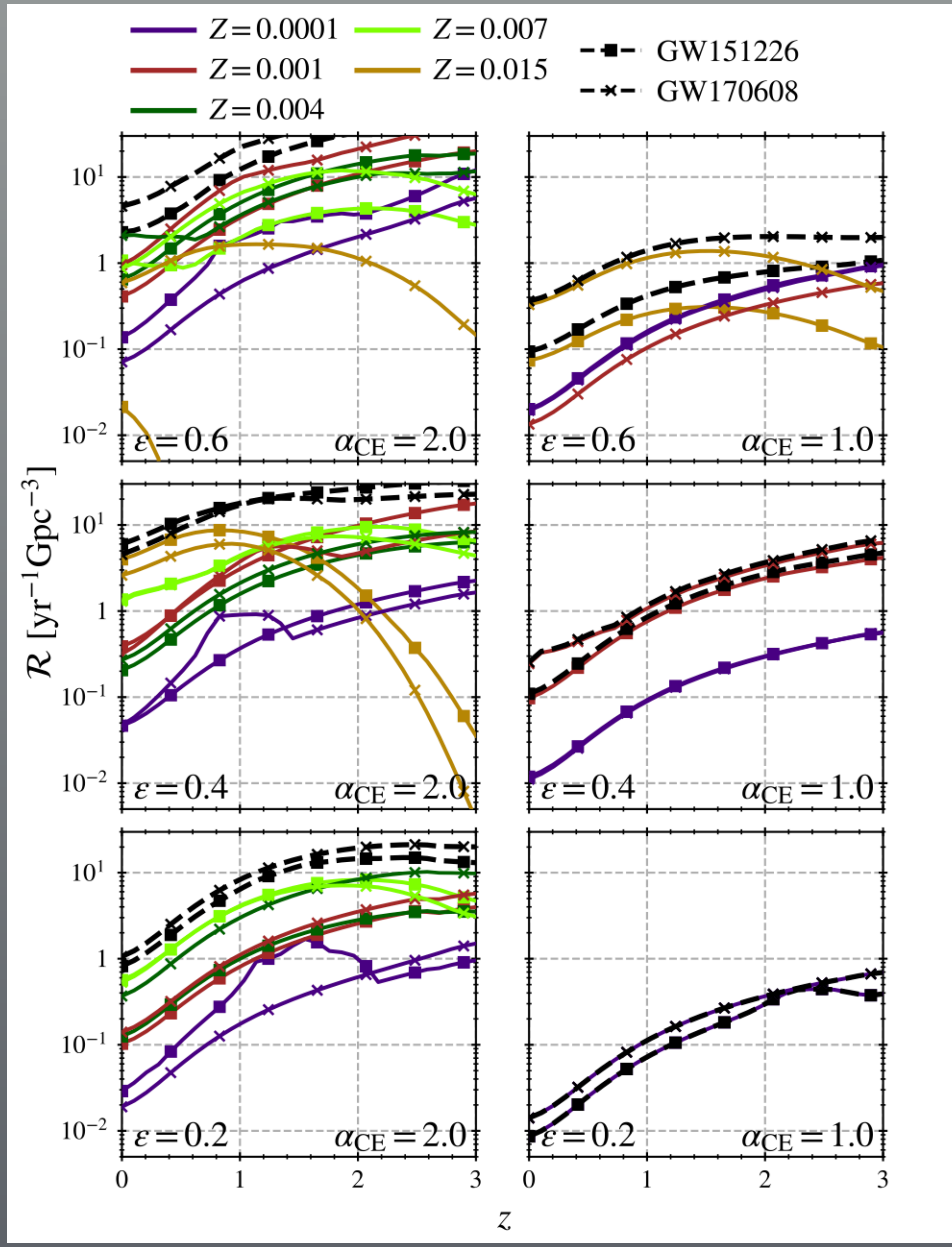
Total mass vs Mass ratio of the GWTC-3 detected events. Credits Abbott et al. (2021)

Simulations and predictions

- Binary Population Synthesis simulations are fast but not detailed
- Detailed binary evolution simulations are slow but detailed.
- Both can be used for statistics of double compact objects (DCO) merging rate with compatible results.



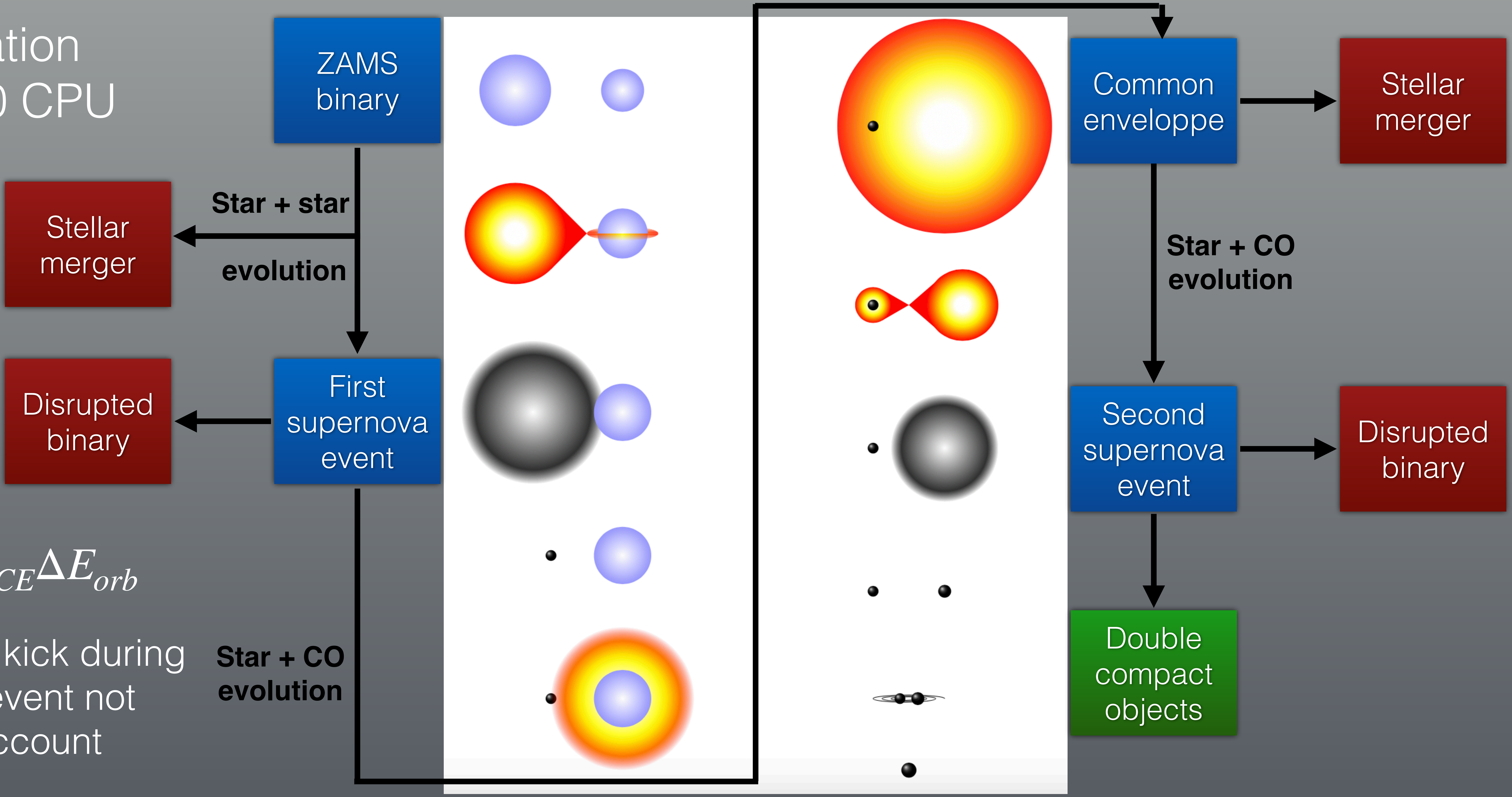
Merger rate density of DCO. Credits Santoliquido et al. (2020)



Merger rate density of DCO. Credits García et al. (2021)

MESA simulations

- 1 simulation ~10-100 CPU hours



- CE phase : $\Delta E_{bind} = \alpha_{CE} \Delta E_{orb}$

- Asymmetric kick during supernova event not taken into account

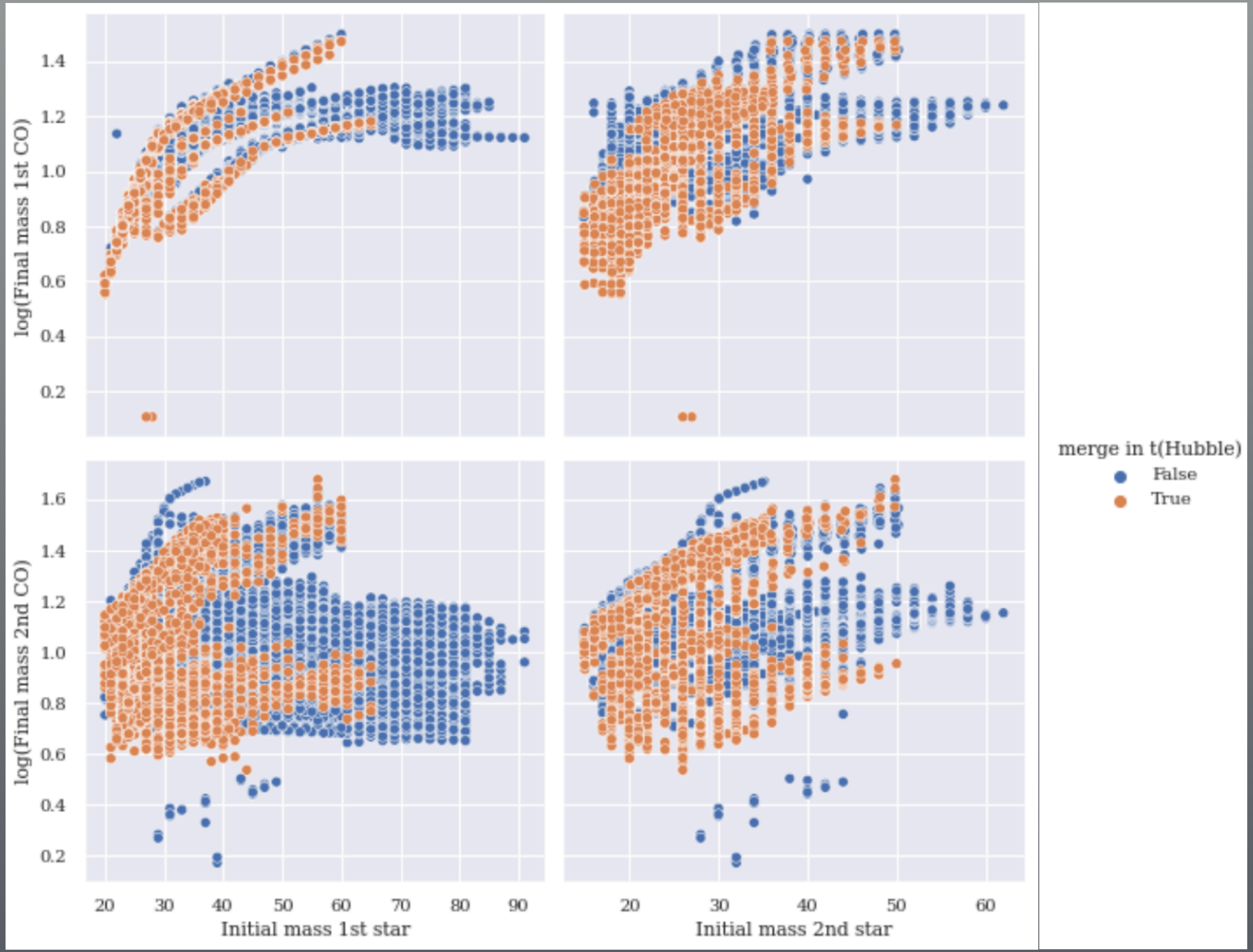
The dataset

- 27k simulations in total, 22k from García et al. (2021)
- Final compact binary :
  Mass of the first object : m_{1f}
  Mass of the second object : m_{2f}
  Orbital distance : a_f
  Merging time :t_{merge}

Initial binary	Lower value
Mass of the first star m_{1i}	$[20M_\odot, 91M_\odot]$
Mass of the first star m_{2i}	$[14M_\odot, 62M_\odot]$
Orbital distance a_i	$[30R_\odot, 500R_\odot]$
Metallicity $-\log(z)$	$\{1.8, 2.2, 2.4, 3, 4, 5\}$
Mass transfer efficiency β	$\{0.2, 0.4, 0.6, 0.8\}$
Common enveloppe efficiency α_{CE}	$\{1, 2\}$

Merger events in the dataset

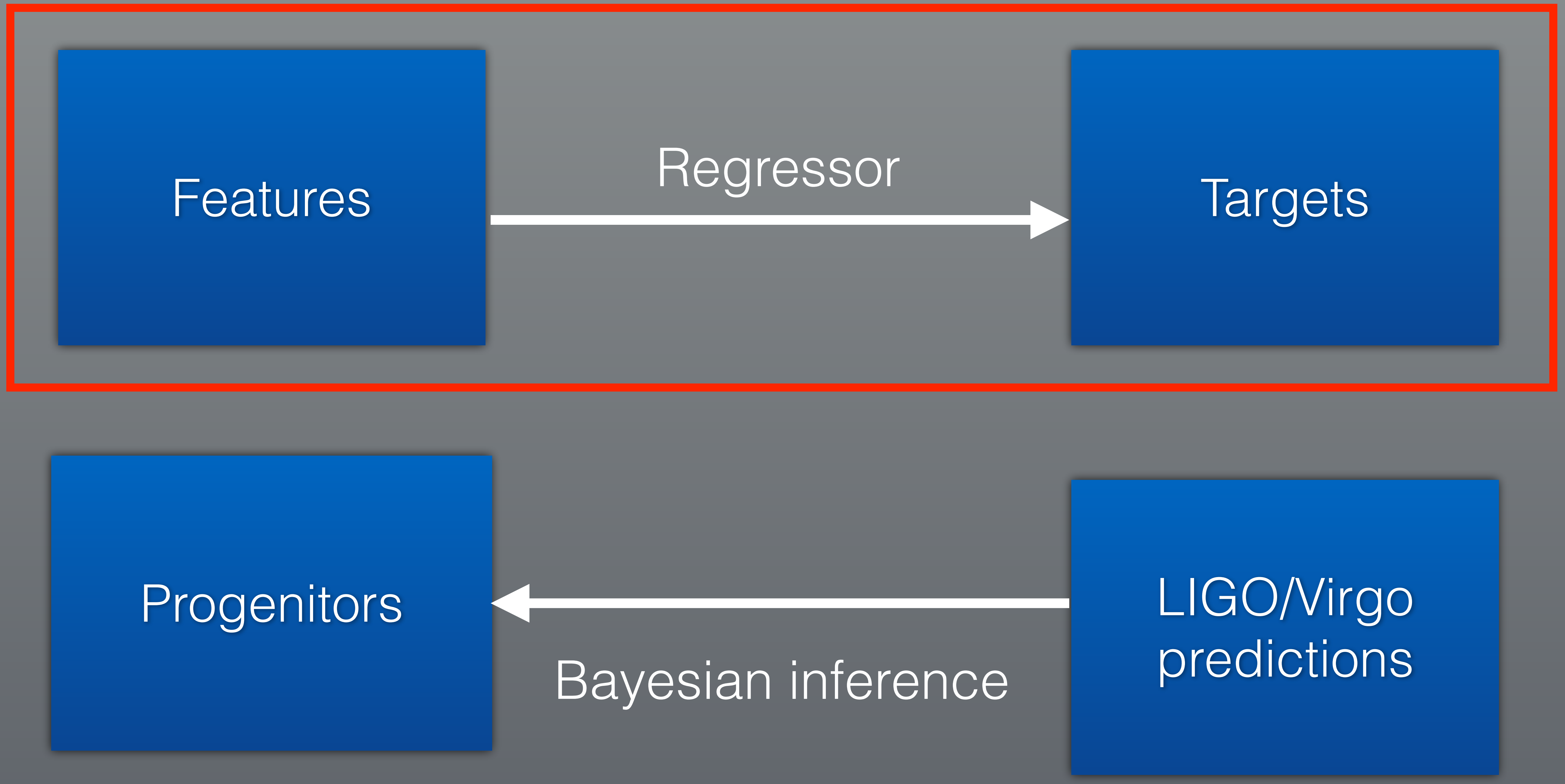
Termination	Number	Percentage of the total dataset
CE merge phase	3157	12 %
Binary disruption	33	< 1%
Numerical issues	2733	10 %
Mergers (within Hubble time)	21170 (3197)	78 % (12 %)
Total	27093	100 %



Representation of the dataset

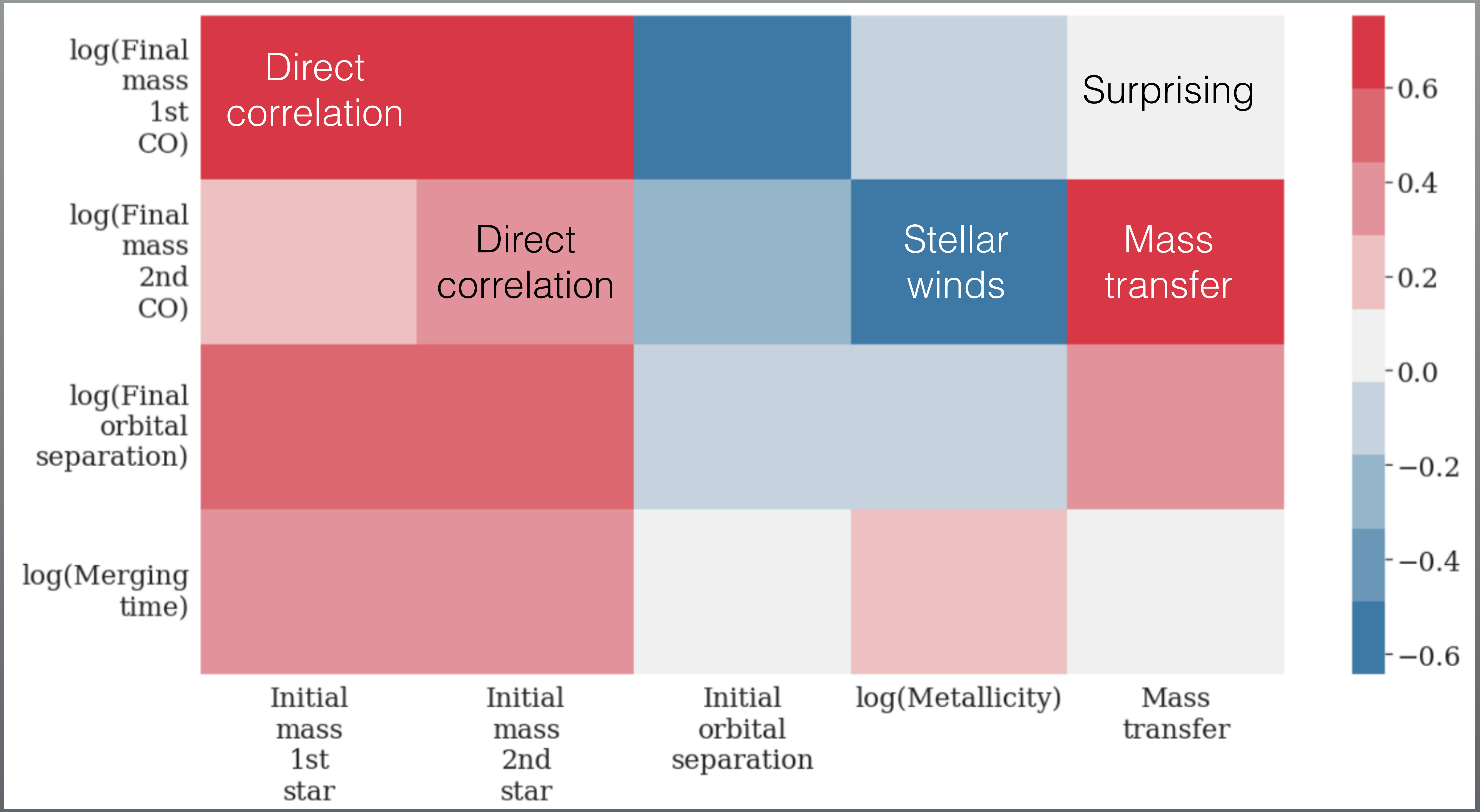
Machine Learning project

- Use MESA simulations to train regressor to infer final state from initial parameters of massive binary stars
- We want to train it such that err(target) < err(LVC)



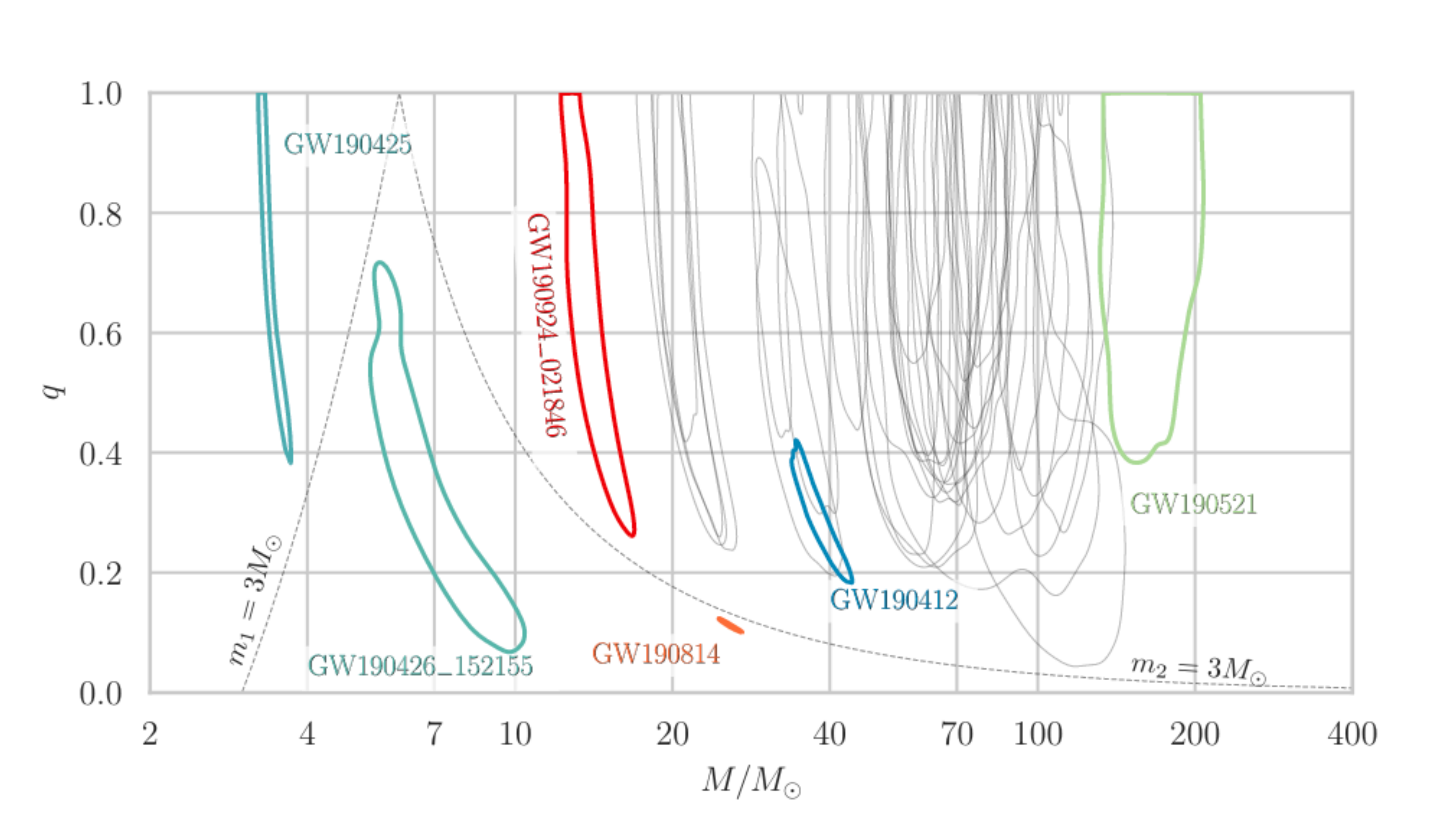
Results (1) : Sanity check

- Strong correlations between final masses and initial parameters
- Weaker correlations for final orbital separation and merging time

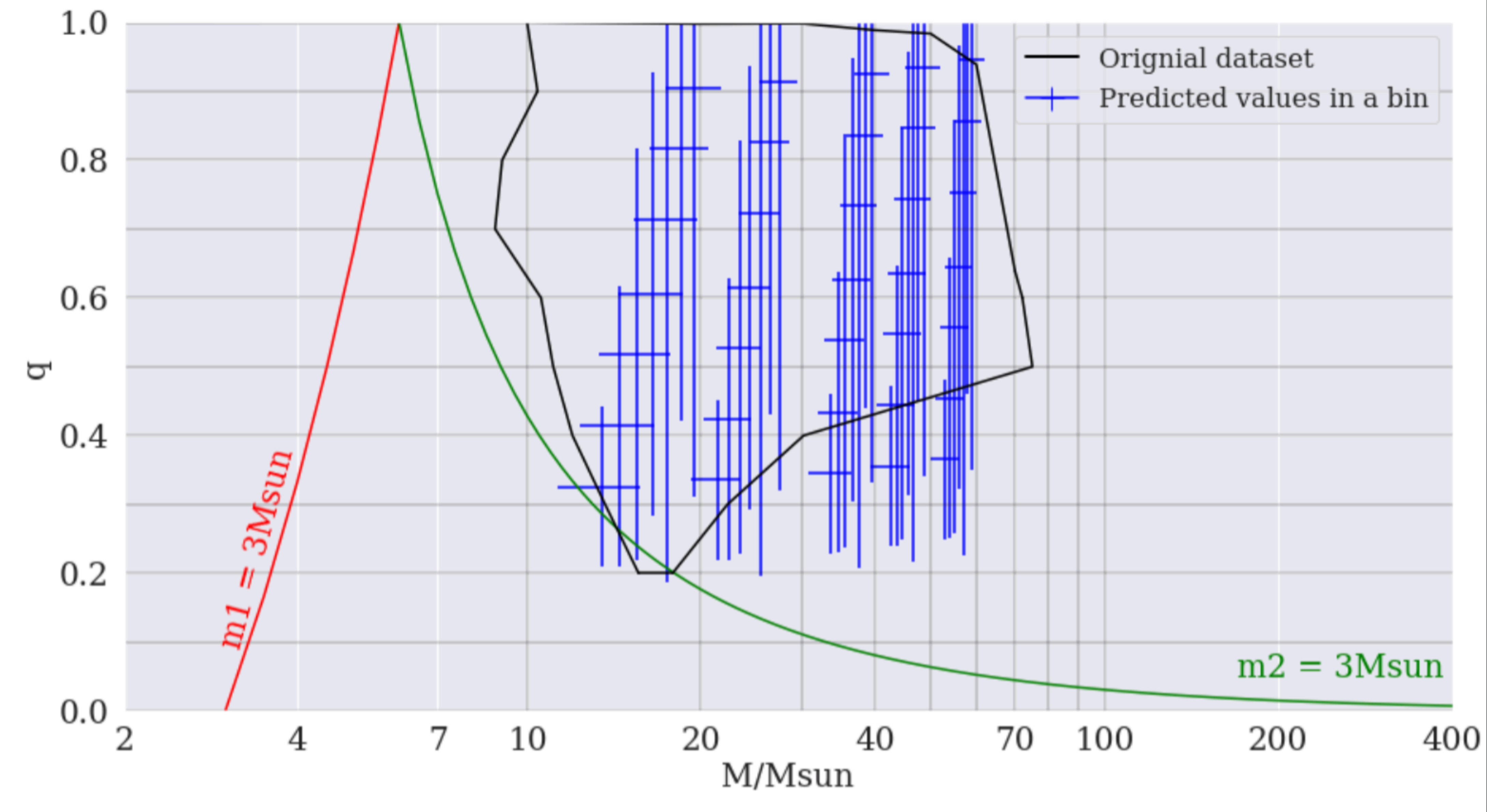


Correlation map between initial and final parameters

Results (2)

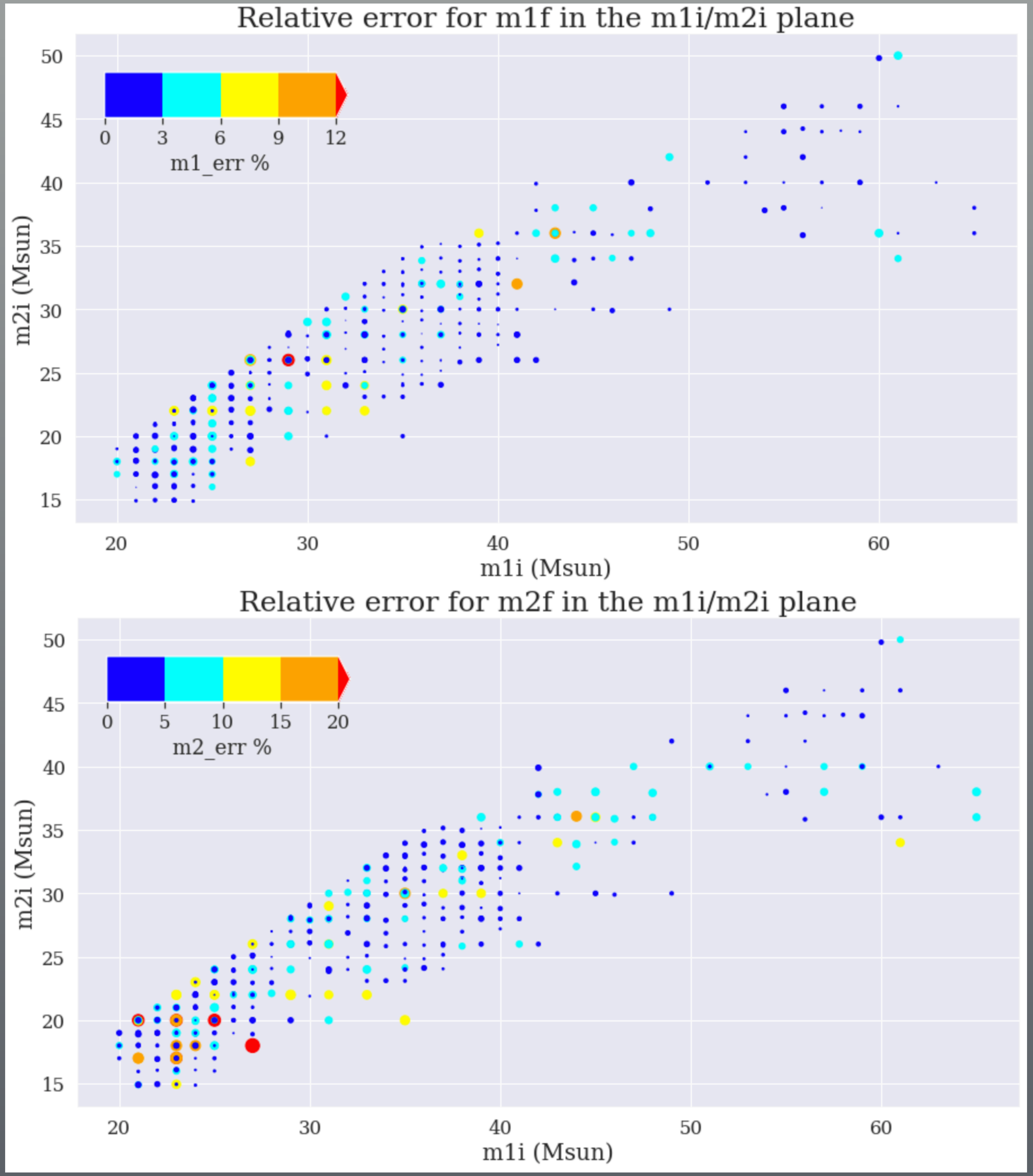


Total mass vs mass ratio of GWTC-2 detections Credits Abbott et al. (2021)

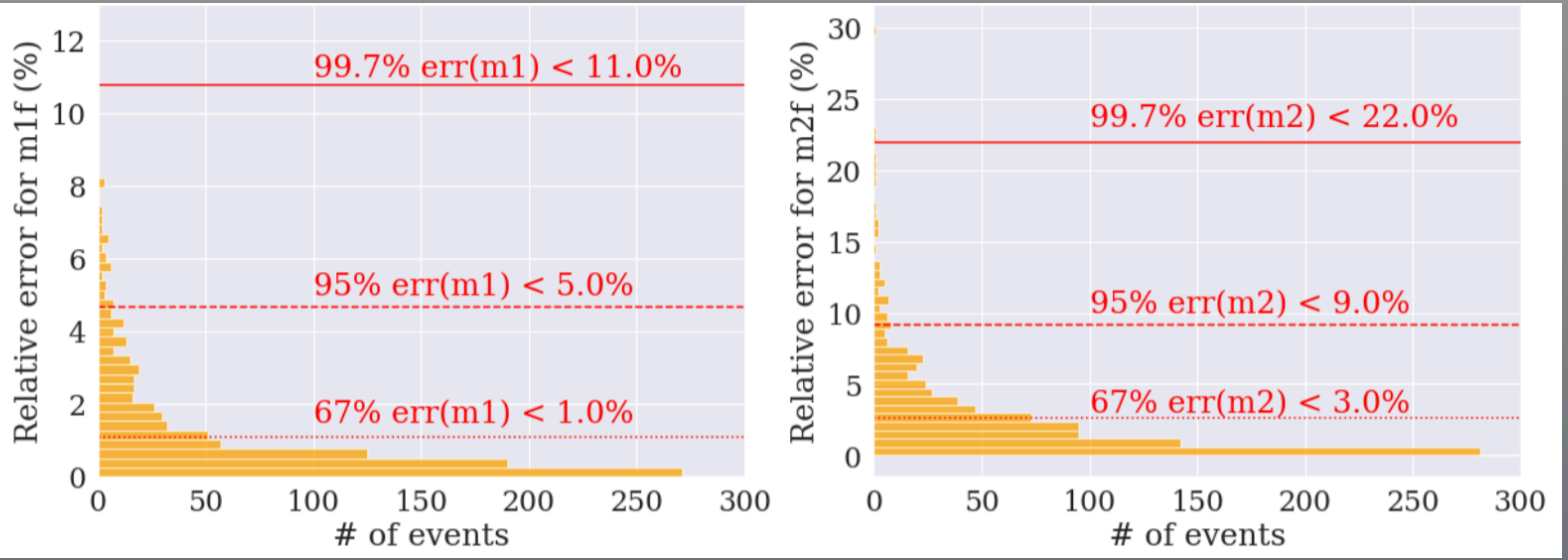


Total mass vs mass ratio for the tested dataset.

Results (3)



Relative error on final mass 1 (top) and final mass 2 (bottom) in the initial masses plane



Number of events in the testing dataset with a relative error for predicted final masses within a given bin. Red lines are values where 67% (dotted), 95% (dashed) and 99.5% (solid) of the dataset are below.

Summary

- MESA simulations describe with good precision physics behind the evolution of binary stars, giving a useful dataset to study progenitors.
- Limit of the approach : fixed phase space and large computing time
- ML based evolution model for black hole binaries in the $m_1 \in [3,30] - m_2 \in [3,40]\,\mathrm{M}_{\odot}$ mass range able to predict component mass with 20% accuracy.
- Final orbital separation and merging time less accurate : median relative error of 90% and above 100% respectively.
- Using Bayesian inference to retrieve statistics on progenitors for LIGO/Virgo detected low mass mergers events.

BACK UP

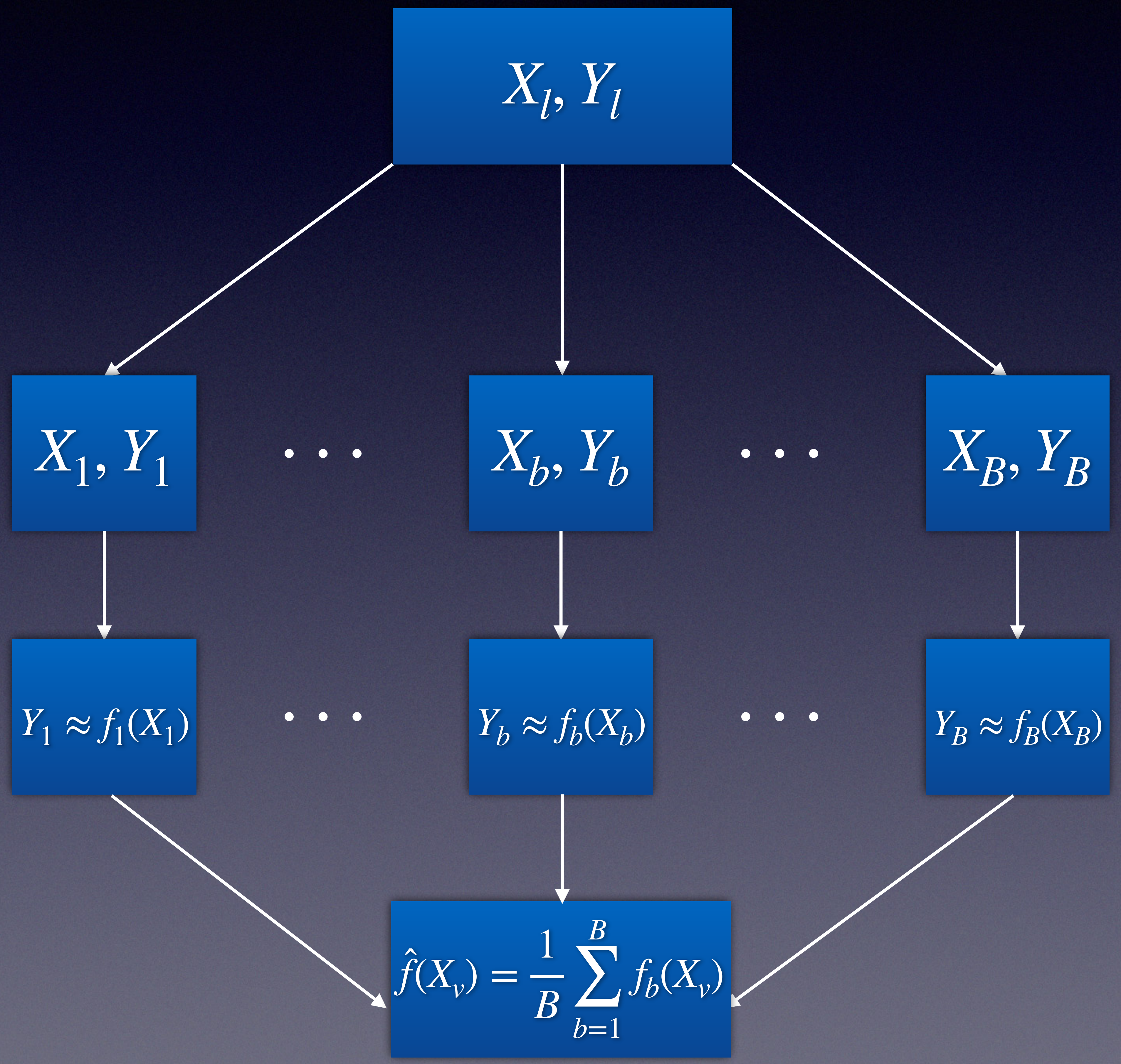
Machine Learning project

The algorithm :

$$\text{Learning set : } X_l = \{x_{l,i}\}_{i\in[1,n]}, Y_l = \{x_{l,i}\}_{i\in[1,n]}$$

$$\text{Validation set : } X_v = \{x_{v,i}\}_{i\in[1,n]}, Y_v = \{x_{v,i}\}_{i\in[1,n]}$$

A RFR algorithm with B trees will simulate B new datasets $(X_b, Y_b)_{b\in[1,B]}$ of size n, using selection with replacement from the original dataset, and apply regression tree on the B new datasets.



Schematic description of a Random Forest Regressor

BPS

- IBiS (Tutukov & Yungelson 1996, and references therein)
- Brussels' code (Vanbeveren et al. 1998a,b)
- Scenario Machine (Lipunov et al. 1996, 2009)
- SeBa (Portegies Zwart & Verbunt 1996; Toonen et al. 2012)
- BSE (Hurley et al. 2002)
- StarTrack (Belczynski et al. 2002, 2008)
- PNS (De Donder & Vanbeveren 2004)
- binary c (Izzard et al. 2004, 2006, 2009)
- SEVN (Spera et al. 2015)
- TRES (Toonen et al. 2016)
- BPASS (Eldridge & Stanway 2016; Stanway et al. 2016; Eldridge et al. 2017; Stanway & Eldridge 2018)
- COMPAS (Stevenson et al. 2017; Riley et al. 2021)
- ComBinE (Kruckow et al. 2018)
- COSMIC (Breivik et al. 2020)
- MOBSE (Giacobbo et al. 2018)

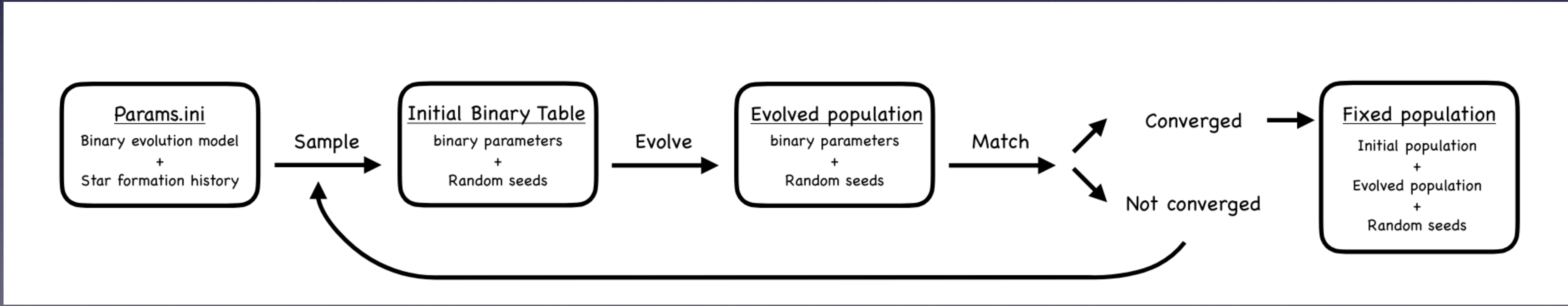
Binary Evolution Codes

- Cambridge STARS code (Eggleton 1971; Pols et al. 1995; Eldridge & Tout 2004)
- ev/STARS/TWIN (Pols et al.1995; Nelson & Eggleton 2001; Eggleton & Kiseleva-Eggleton 2002)
- BEC (Heger et al. 2000; Heger & Langer 2000)
- BINSTAR (Siess et al. 2013)
- MESA (Paxton et al. 2011, 2013, 2015, 2018, 2019)
- BPASS (Eldridge & Stanway 2016; Stanway et al. 2016; Eldridge et al. 2017; Stanway & Eldridge 2018)

State of the art in massive binary simulations and inferences on progenitors

Binary Population Synthesis (BPS)

- BPS is a great tool to get a huge amount of binaries, spanning wide ranges of masses and separation in a reasonable computing time.
- Parametric BPS (pBPS) use fitting formulae or look-up tables, but imply strong approximations for the binary systems.

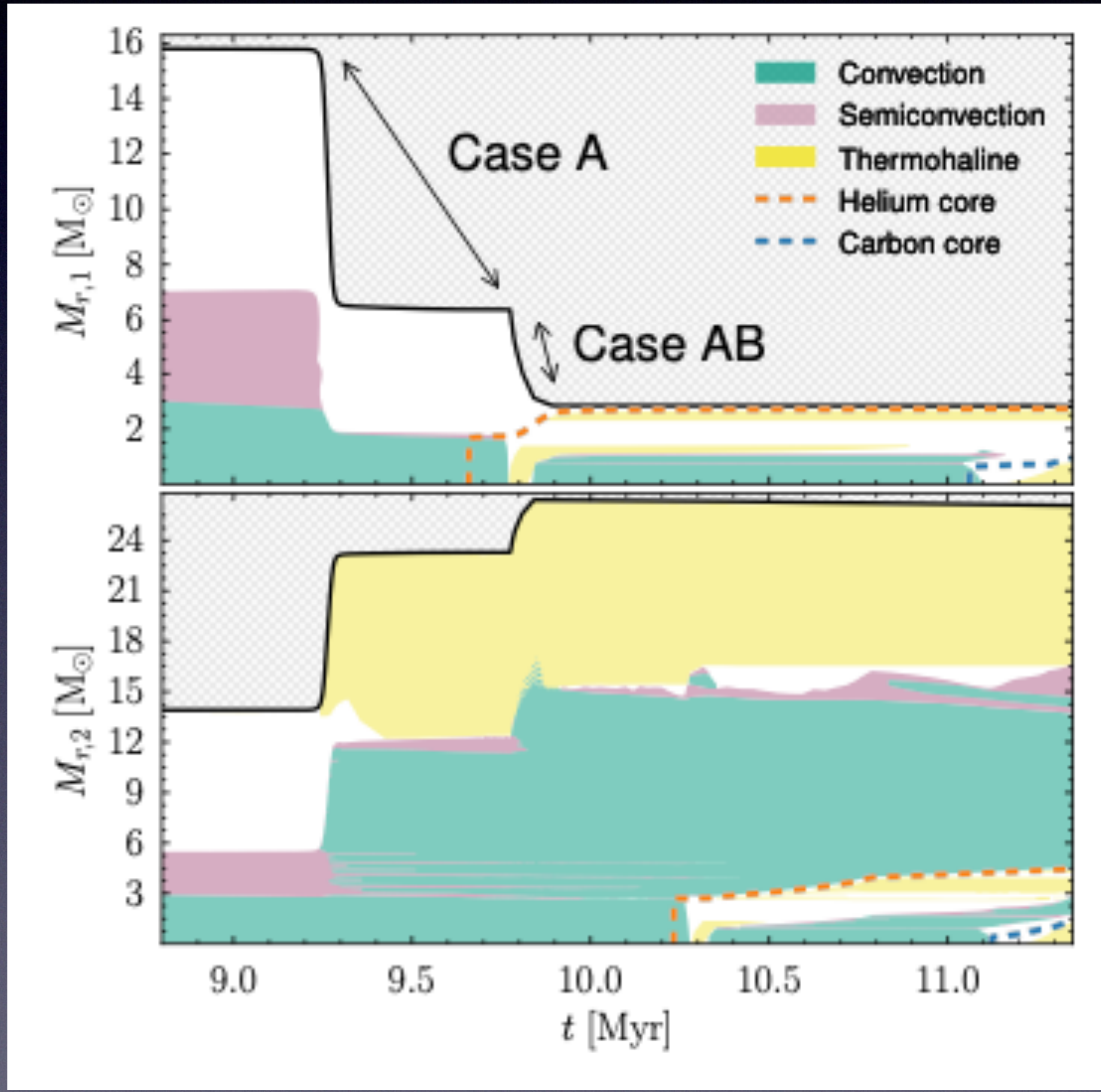


Schematic for the COSMIC BPS code. The evolution is made through a look-up table. Credits : Breivik et al. (2020)

State of the art in massive binary simulations and inferences on progenitors

Detailed Binary Evolution (dBE)

- Detailed Binary Evolution codes are much more time consuming with ~10 — 100 CPU hours per simulation (Paxton et al. 2019).
- They account for more precise physics of the binary and are easily customizable.

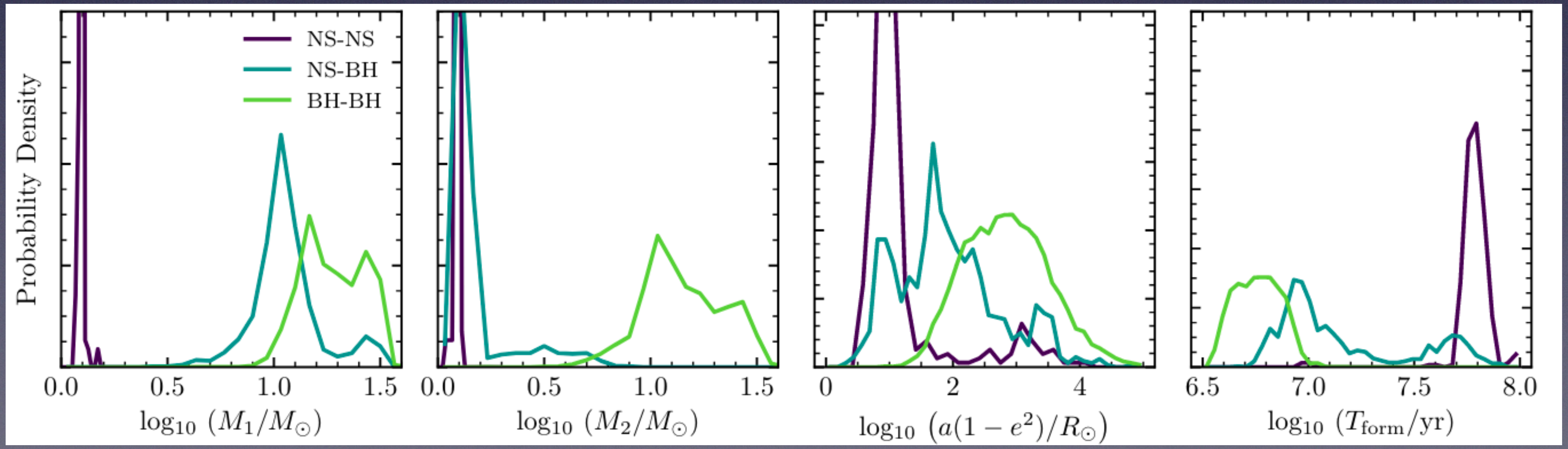


Kippenham diagram of a 14 + 16 binary with a 3 days orbital period. Credits Paxton et al. (2017)

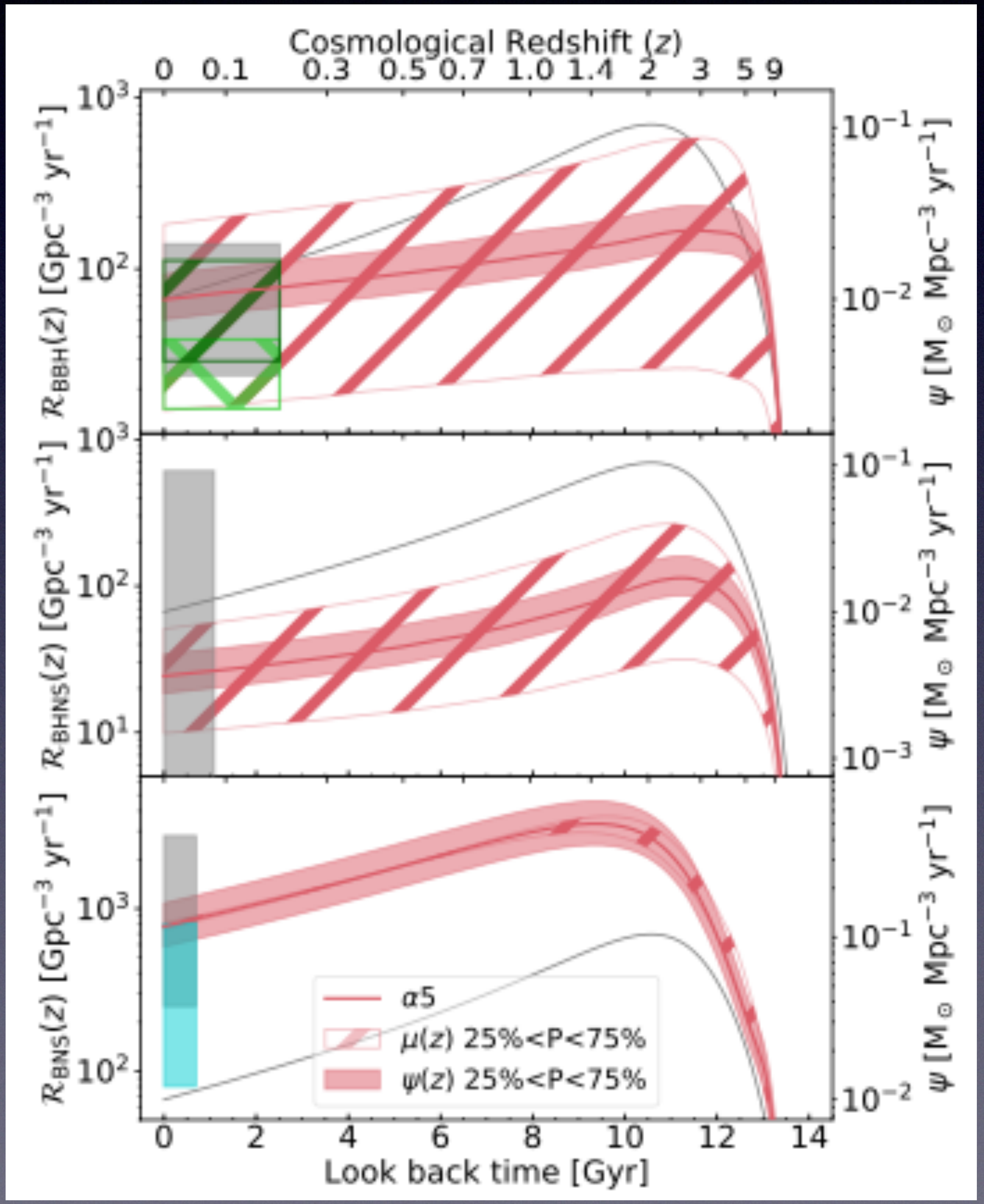
State of the art in massive binary simulations and inferences on progenitors

Predictions

- Statistics on merger rate density within the Hubble time.
- Statistics on total population of DCO.

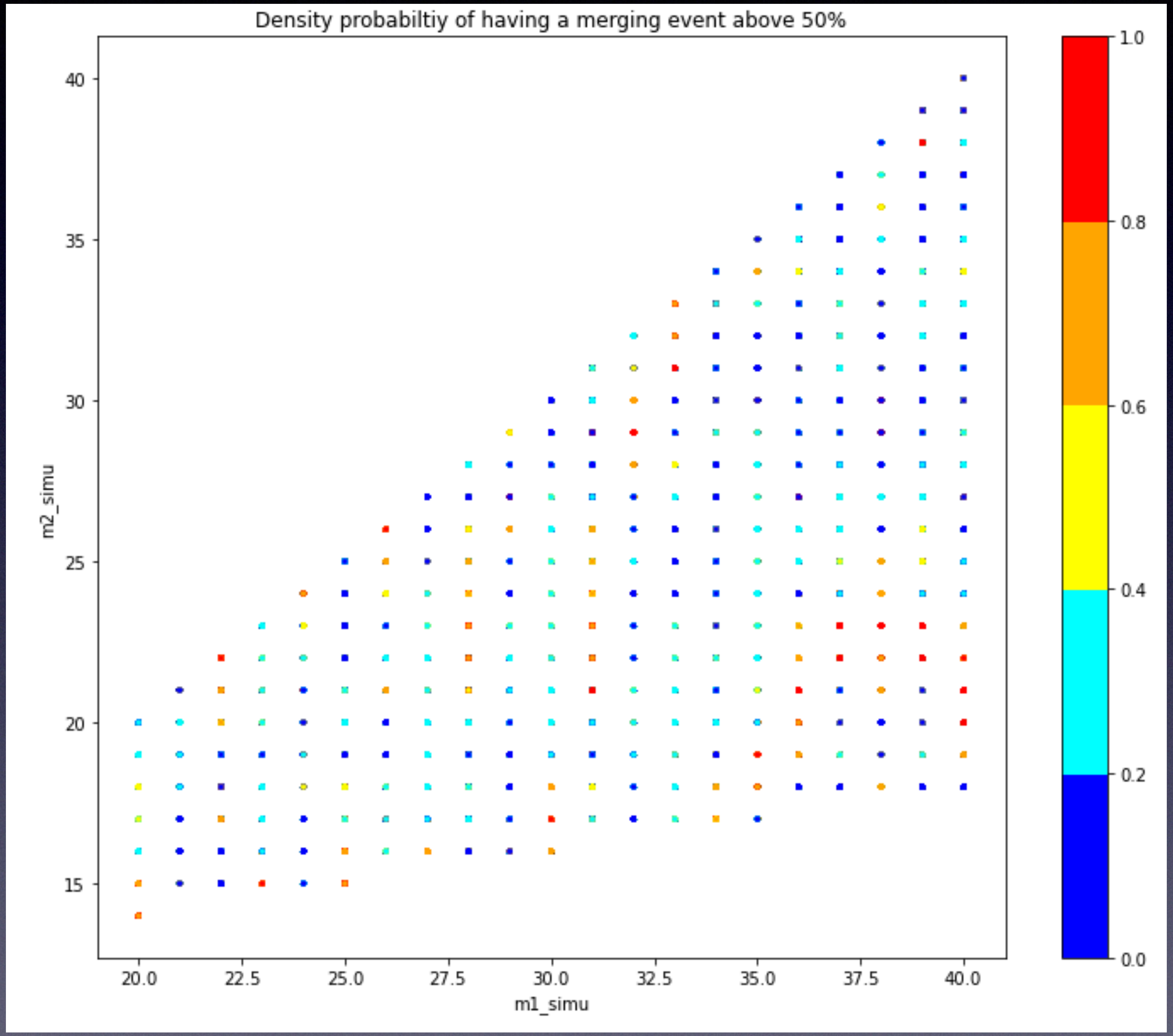


DCO population produced by POSYDON. Credits Fragos et al. (2022)

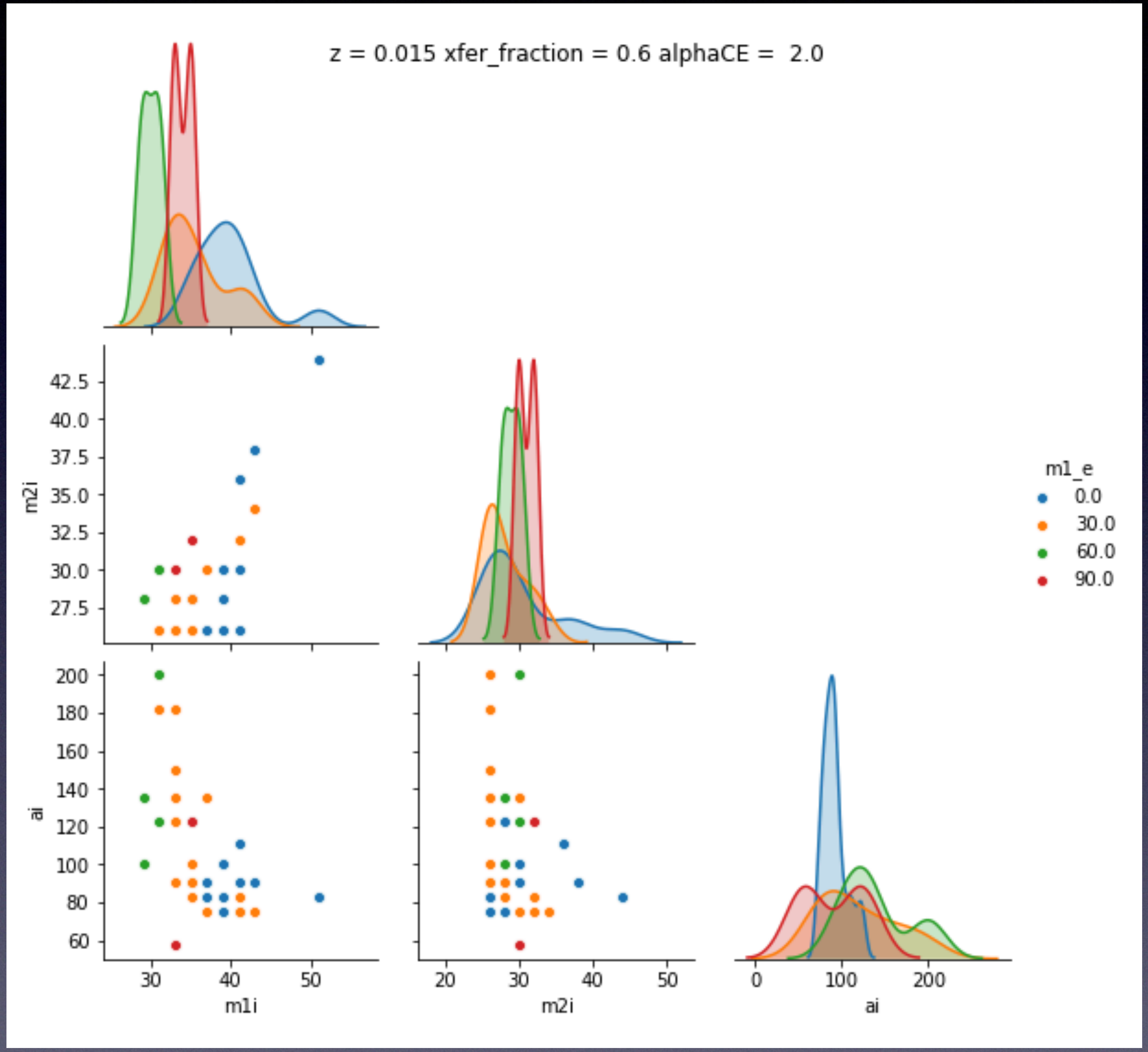


Merger rate density of DCO. Credits Santoliquido et al. (2020)

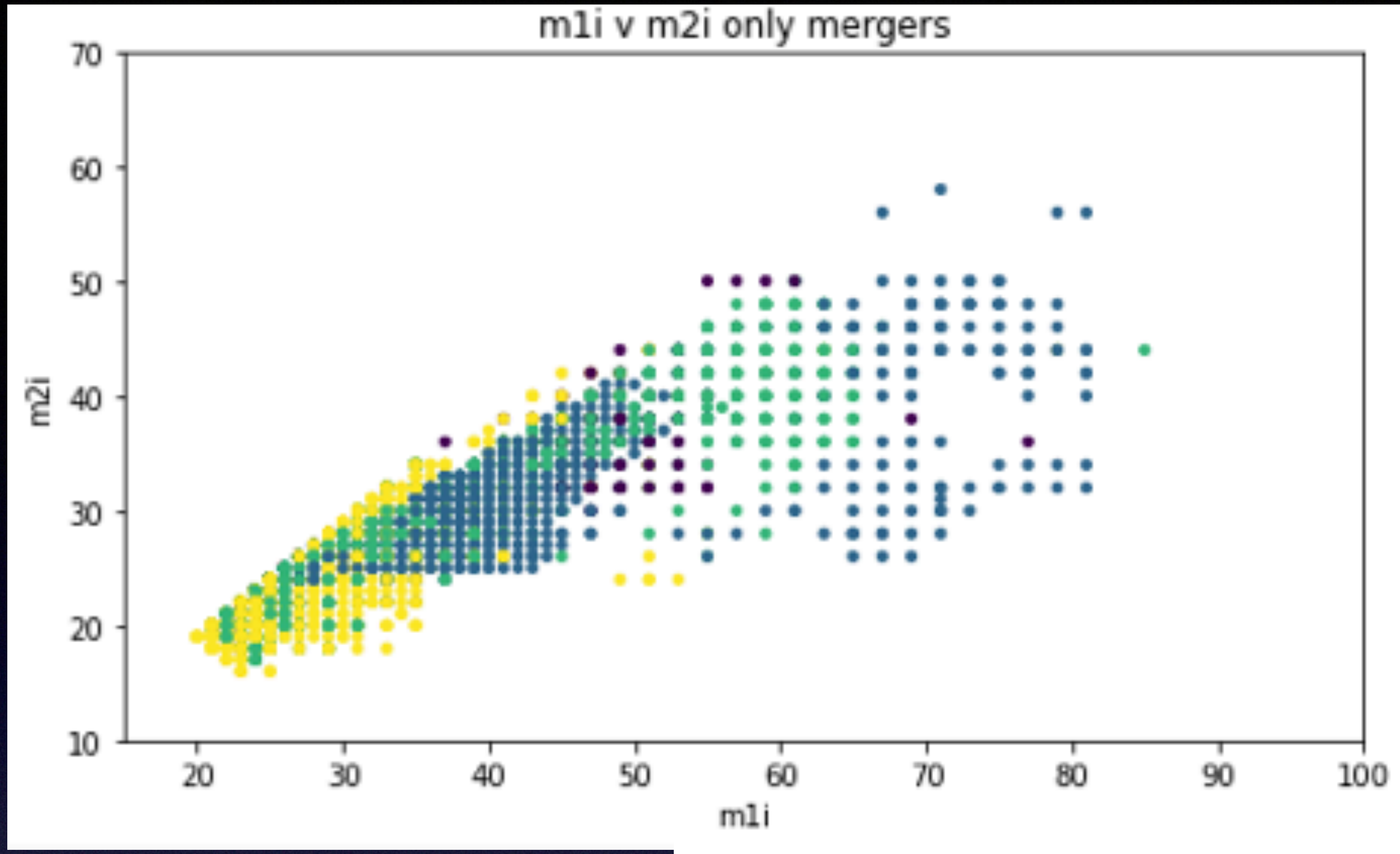
Precision of the model and holes

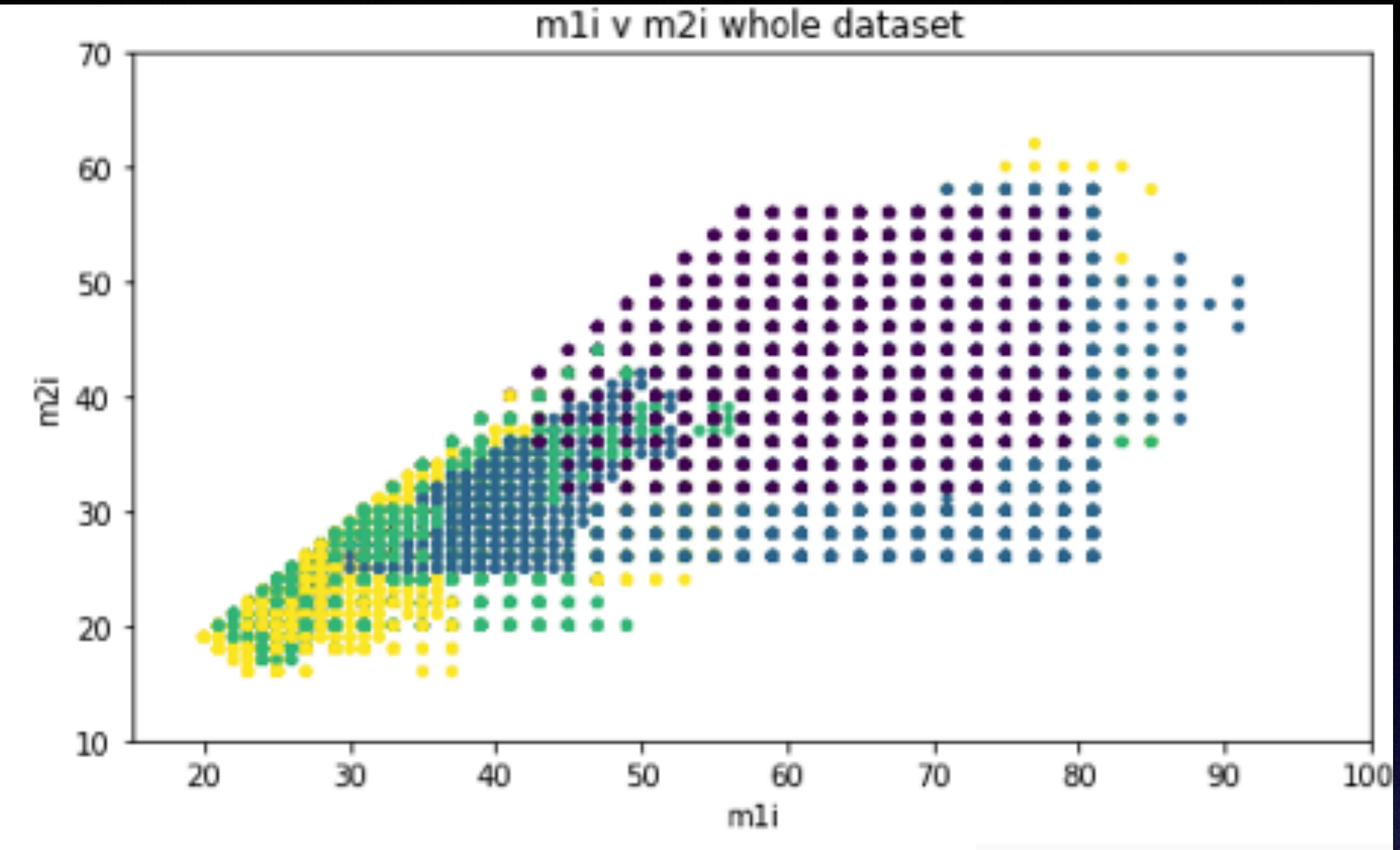


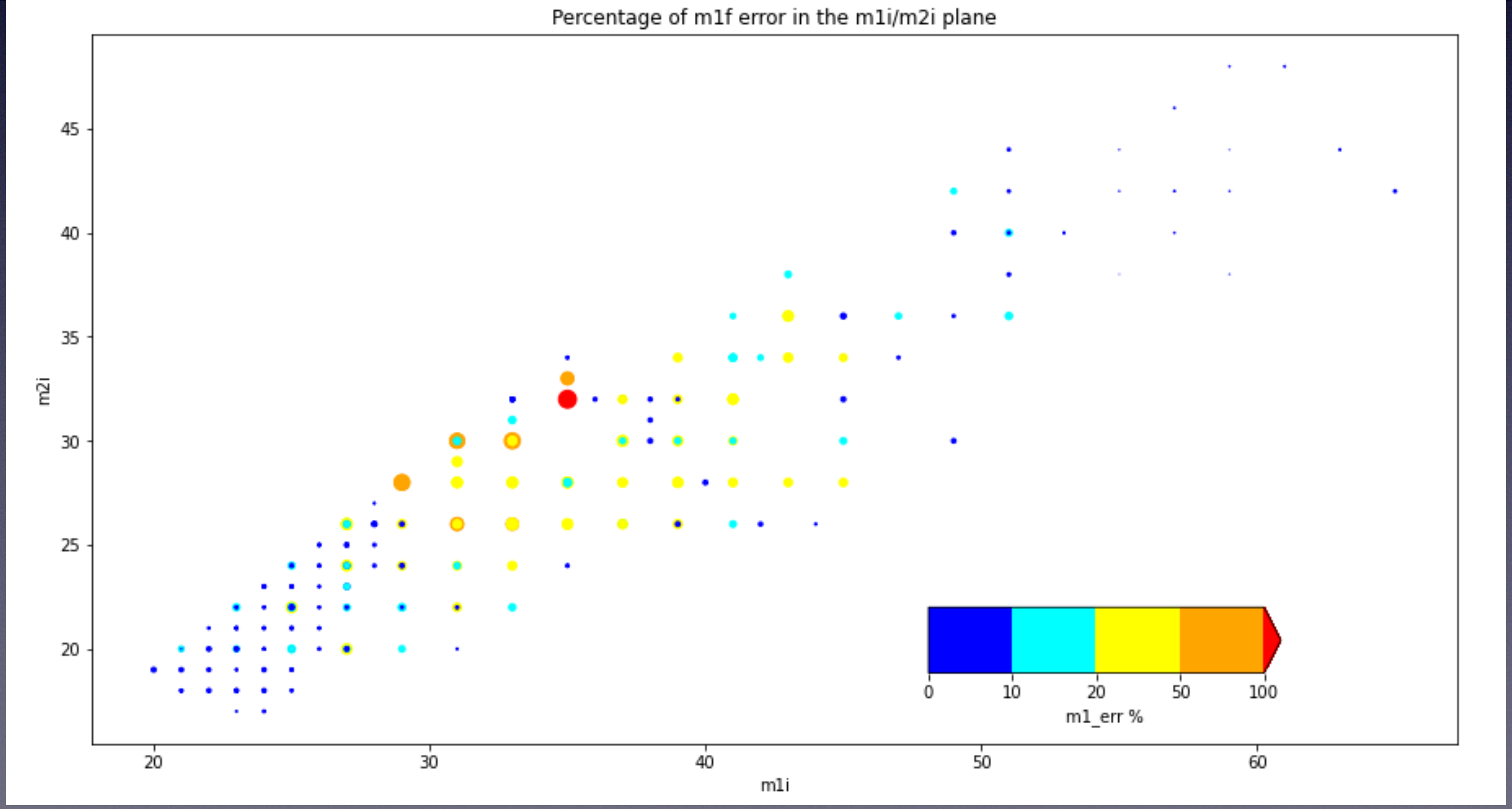
Probability of having a merger in M1i/M2i plane



Representation of a cut in the phase space.
(z=0.015, β=0.4, αCE=2)

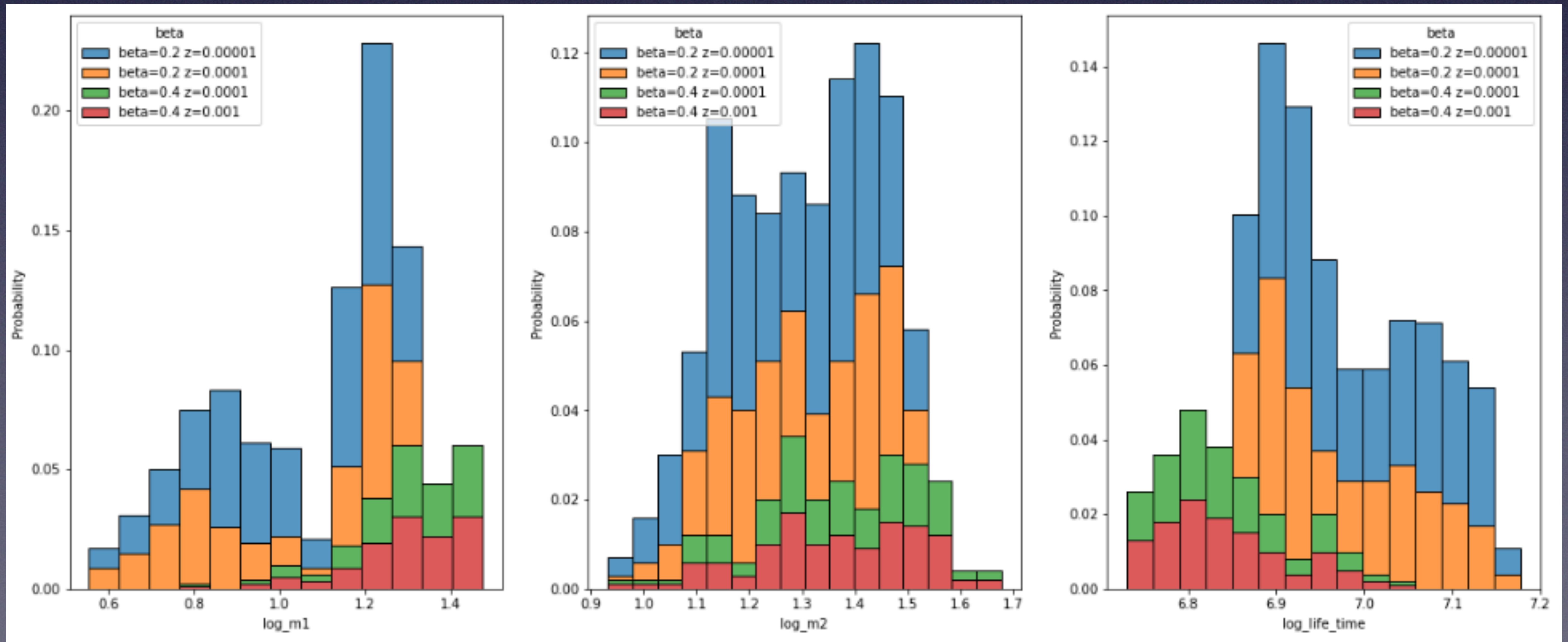

m1i v m2i only mergers
m2i
m1i
10
20
30
40
50
60
70
20
30
40
50
60
70
80
90
100


m1i v m2i whole dataset
m2i
m1i
10
20
30
40
50
60
70
20
30
40
50
60
70
80
90
100

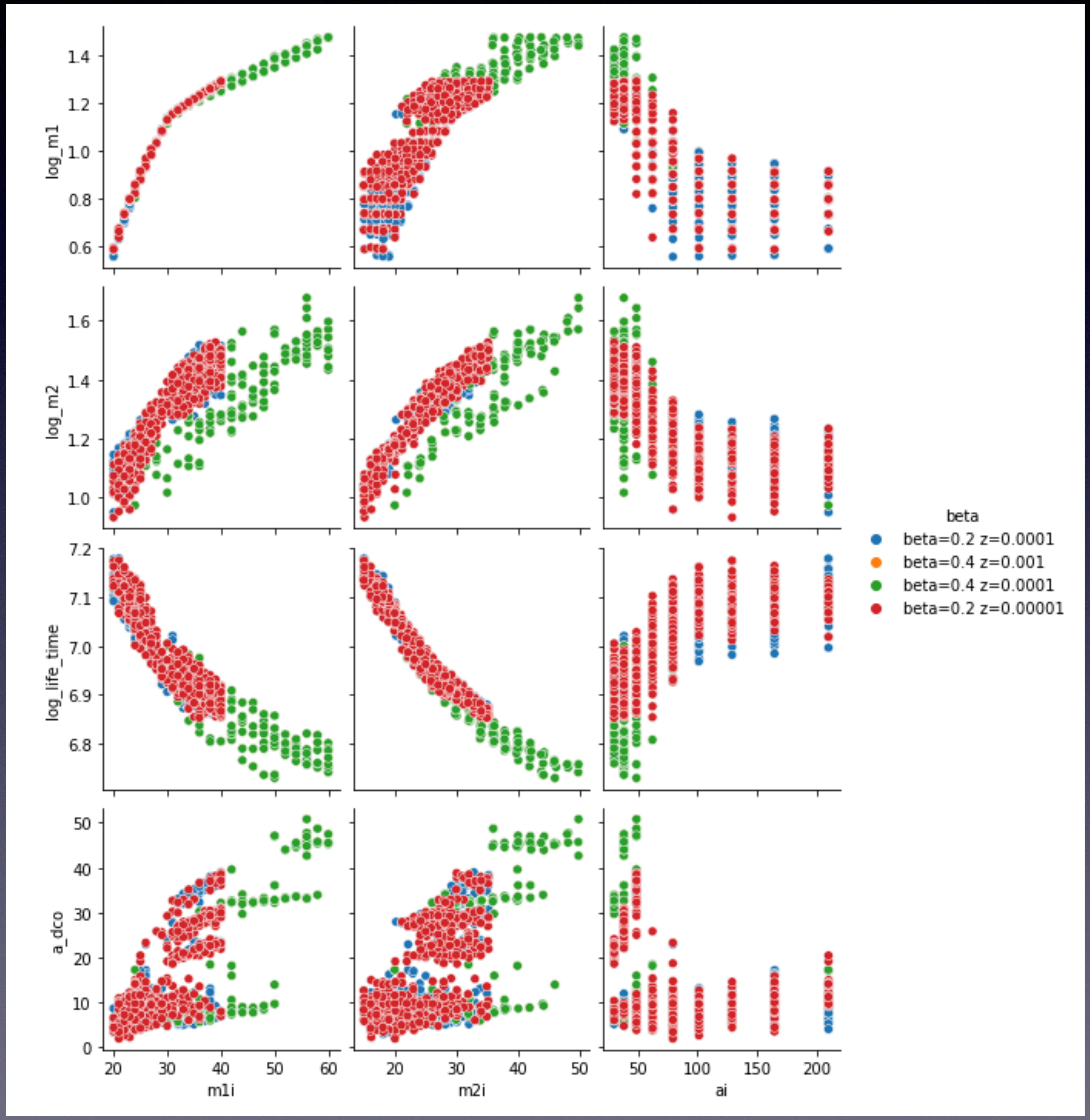

Percentage of m1f error in the m1i/m2i plane
m2i
m1i
20
25
30
35
40
45
20
30
40
50
60
0
10
20
50
100
m1_err %

Posterior simulations

- 4984 have been done 998 mergers (within Hubble time) have been obtained
- Different z and β tested $(\log(z) \in \{-2\,;-3;-4\};\ \beta \in \{0.2\,;0.4\})$



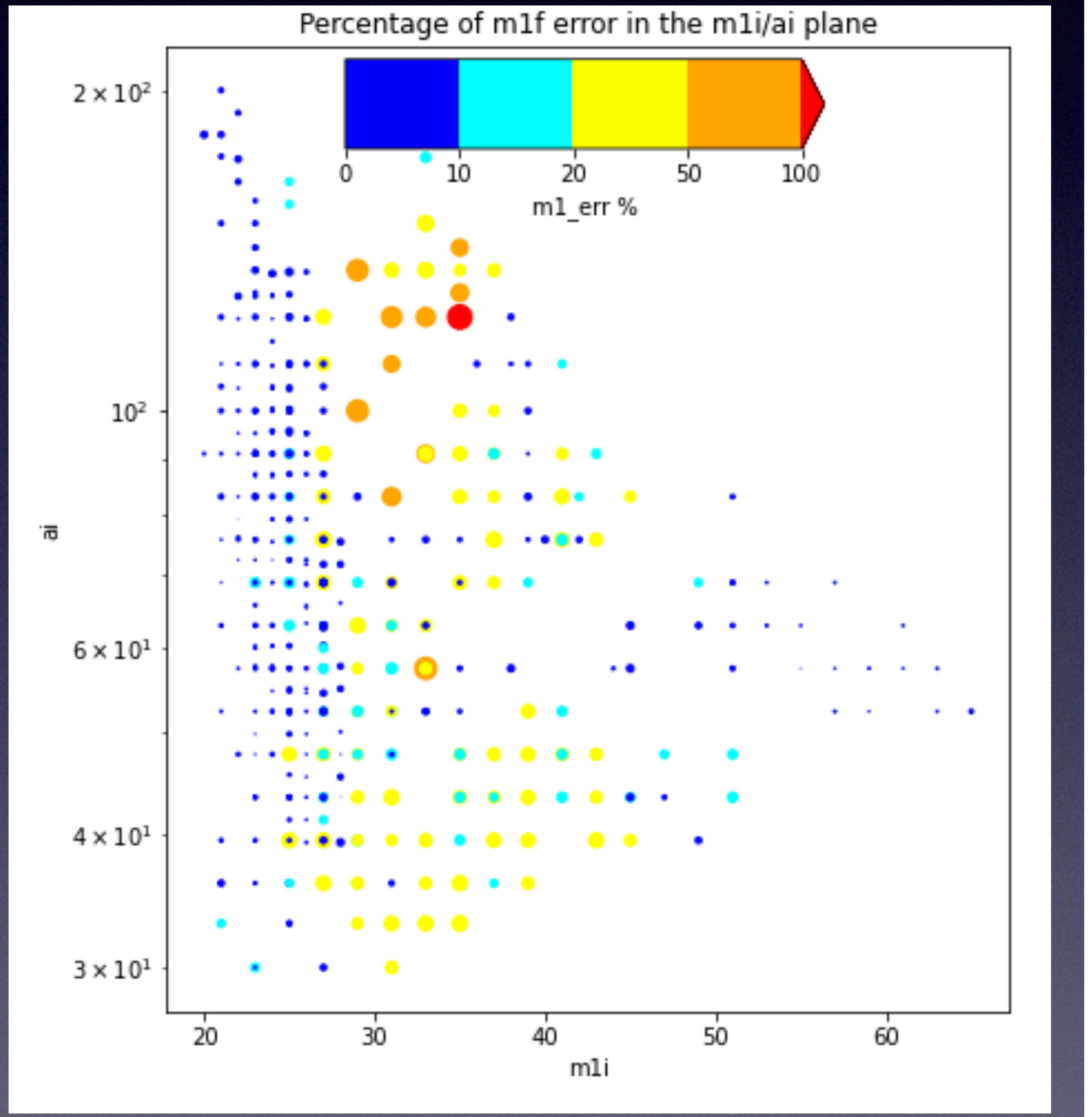
Final parameters histogram repartition. Color indicate β and z.



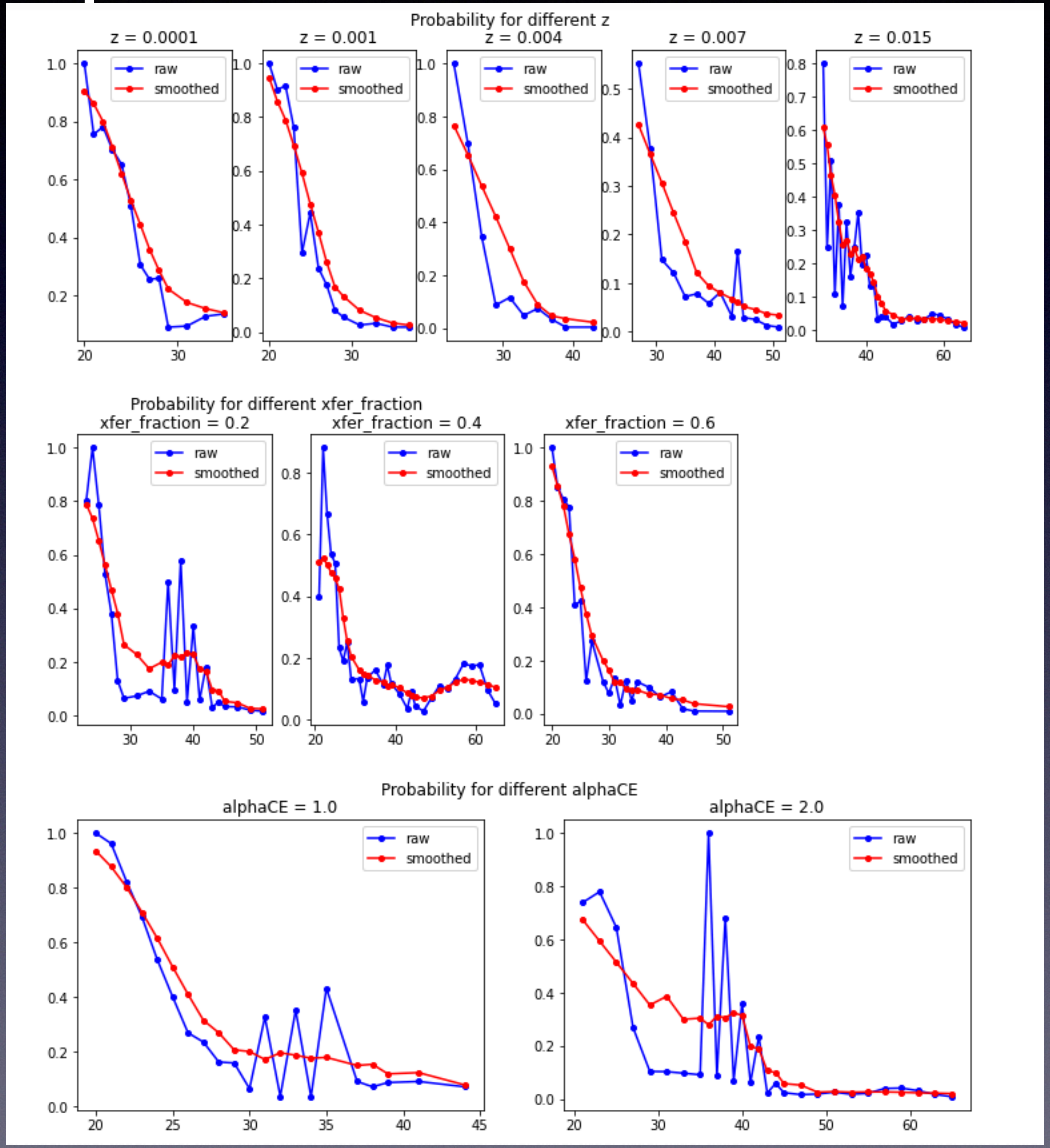
Initial vs final parameters for the simulations. Colors represent β and z

Changing the strat, populate smart

- High computational cost
- Some areas have just holes that have to be filled



Showing holes in the phase space



Some regions are less populated, leading to those oscillations in the merging rate density. (Red is what more simulations would look like)

Comparisons with GWTC-2

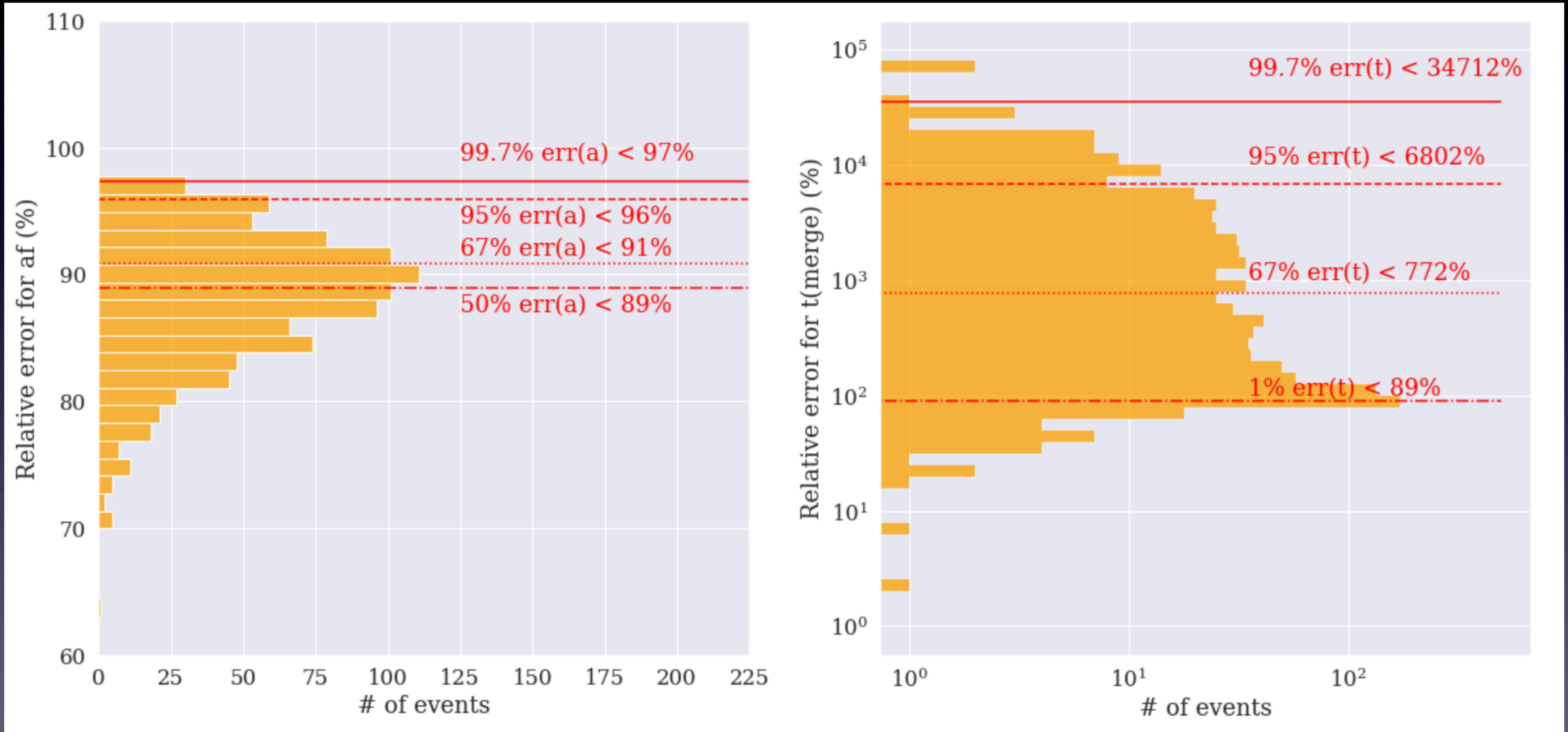
- Given N_{obs} LIGO/Virgo detections $\{x\} = \{x_1, \ldots, x_{N_{obs}}\}$

- Also given hyper-parameters θ, with prior p(θ), we have :

$$p(\theta | \{x\}, N_{obs}) \propto p(\{x\}, N_{obs} | \theta) p(\theta)$$

- Which can be rewritten as :

$$p(\theta | \{x\}, N_{obs}) \propto p(\theta) \prod_{i=1}^{N_{obs}} \frac{\int p(x_i | \theta, \alpha) p_{pop}(\alpha | \theta) d\alpha}{\int p_{det}(\alpha, \theta) p_{pop}(\alpha | \theta) d\alpha}$$



Number of events in the testing dataset with a relative error for predicted final orbital separation and merging time within a given bin. Red lines are values where 67% (dotted), 95% (dashed) and 99.5% (solid) of the dataset are below. Dashdot correspond to 50% (left), and 1% (right)