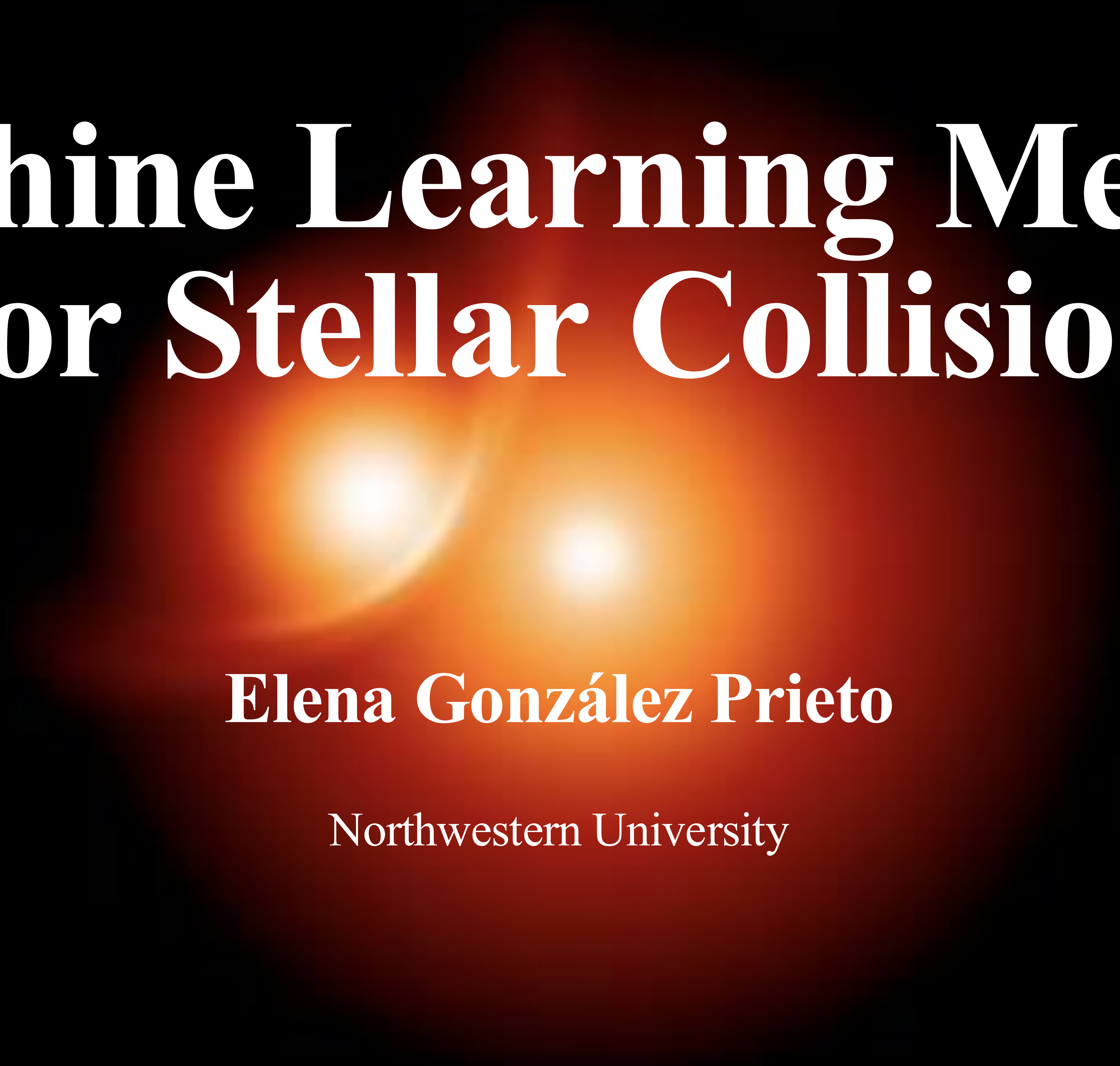
Machine Learning Methods for Stellar Collisions



Elena González Prieto

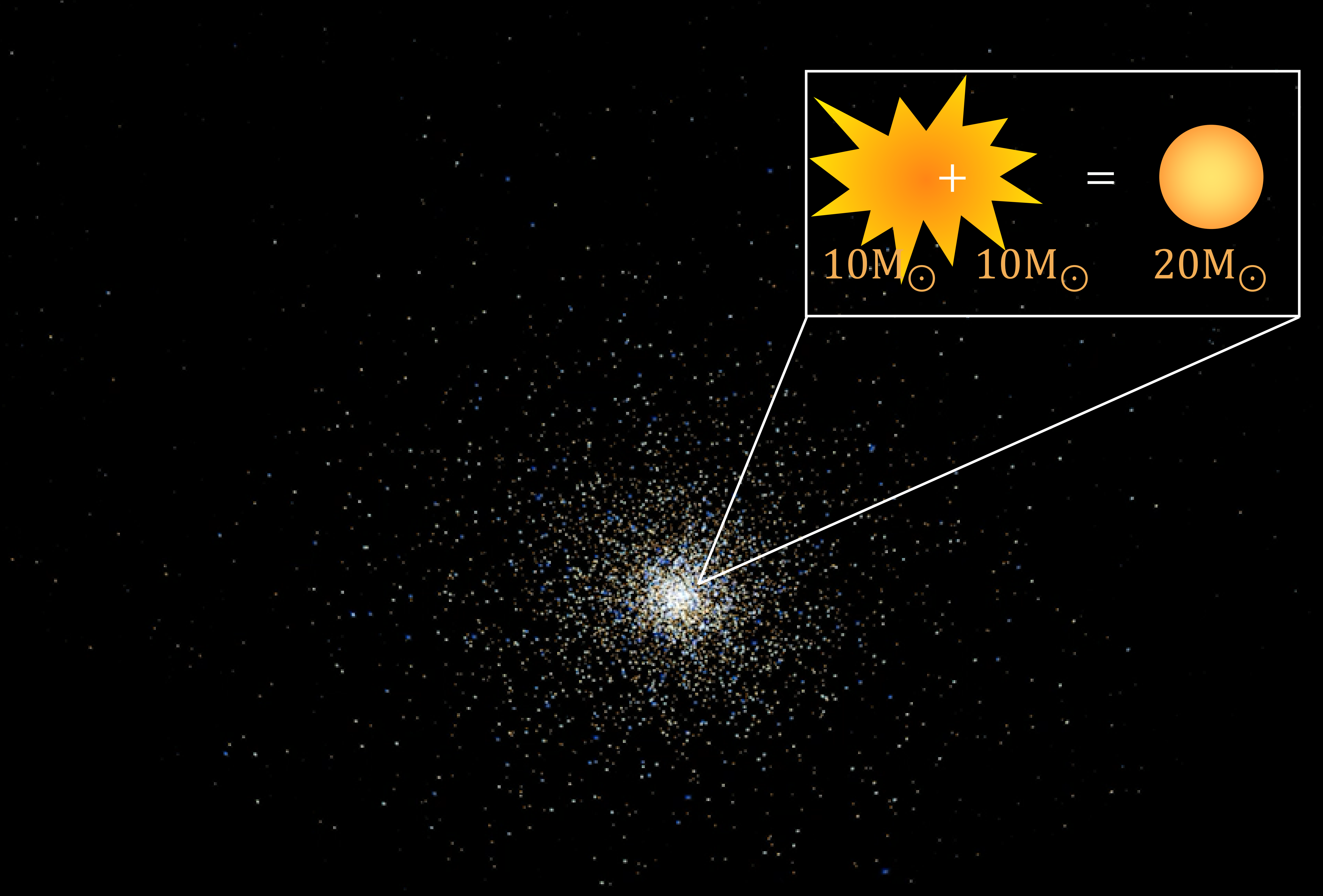
Northwestern University

CIERA
Center for Interdisciplinary Exploration and Research in Astrophysics

NSF

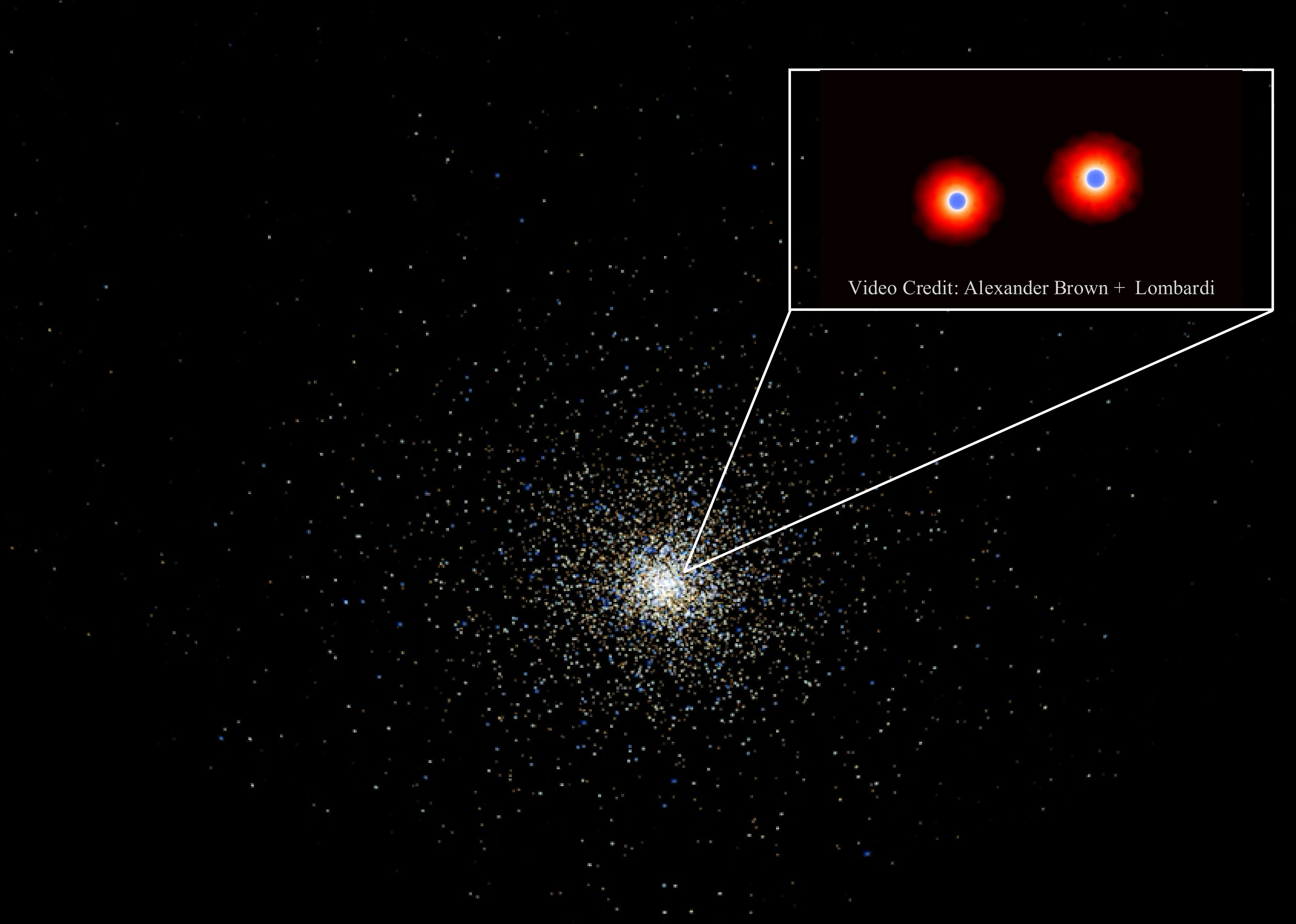
SkAI
Funded by the U.S. National Science Foundation and the Simons Foundation

In Collaboration With: James C. Lombardi, Sanaea C. Rose, Charles F.A. Gibson, Christopher E. O'Connor, Tjitske Starkenburg, Fulya Kıroğlu, Kyle Kremer, Tristan C. Sand, Frederic A. Rasio



Video Credit: NASA & ESA

elena.prieto@northwestern.edu



Video Credit: Alexander Brown + Lombardi

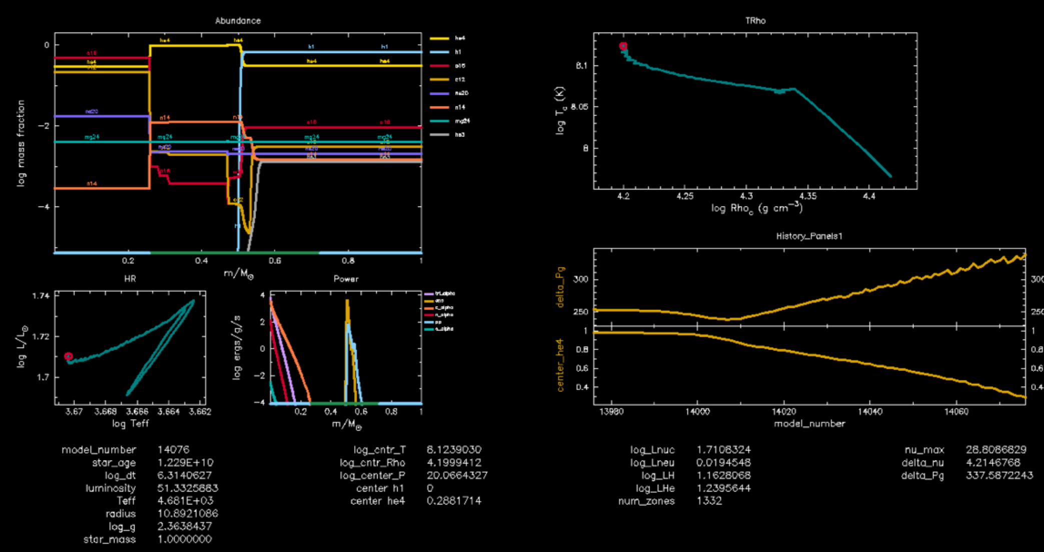
elena.prieto@northwestern.edu

Framework

Realistic Stellar Structure

Machine Learning Model

Hydrodynamics Simulation



MESA

(Paxton et al. 2011-2019, Jermyn et al. 2023)

+

StarSmasher

collAIder

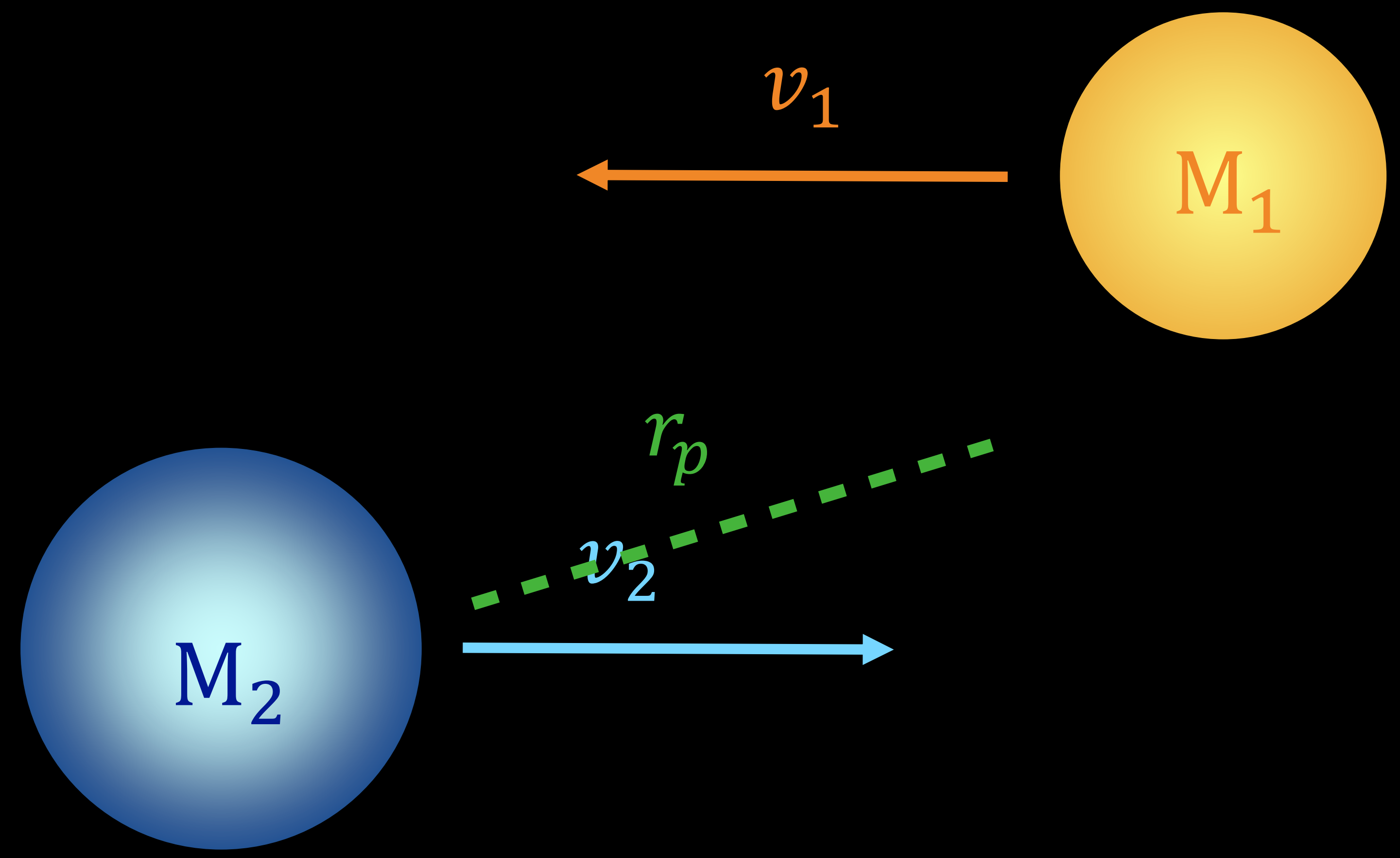
(Rasio et al. 1991, Gaburov et al. 2010)

elena.prieto@northwestern.edu

Stellar Collisions

- Assume all stars are born at the same time

$$\text{Input} = \begin{cases} \text{Age [Gyr]} \\ \mathrm{M}_1\,[\mathrm{M}_\odot] \\ \mathrm{M}_2\,[\mathrm{M}_\odot] \\ v_\infty\,[\mathrm{km/s}] \\ r_p\,[R_\odot] \end{cases}$$



$$v_\infty = |\vec{v}_1 - \vec{v}_2|$$

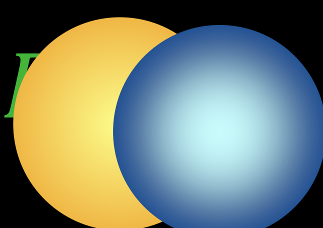
Stellar Collisions

27,720 Simulations

$$\text{Input} = \begin{cases} \text{Age [Gyr]} & 10^{-3} - 10\ \text{Gyr} \\ \mathrm{M}_1[\mathrm{M}_\odot] & 0.2 - 64M_\odot \\ \mathrm{M}_2[\mathrm{M}_\odot] & 0.2 - 64M_\odot \\ v_\infty[\mathrm{km/s}] & 10 - 16{,}000\mathrm{km/s} \\ r_p[R_\odot] & 0 - 100\%\ \text{enclosed radii} \end{cases}$$

Stellar Collisions

Input = $\left\{ \begin{array}{l} \text{Age [Gyr]} \\ M_1\,[M_\odot] \\ M_2\,[M_\odot] \\ v_\infty\,[\text{km/s}] \\ r_p\,[\end{array} \right.$



Output =

Classification

- Number of surviving stars

  0 : Mutual Disruption

  1 : Merger

  2 : Fly-by

  3 : Stripped Star

- Masses of surviving stars

  $M_{1,\,\text{final}}$

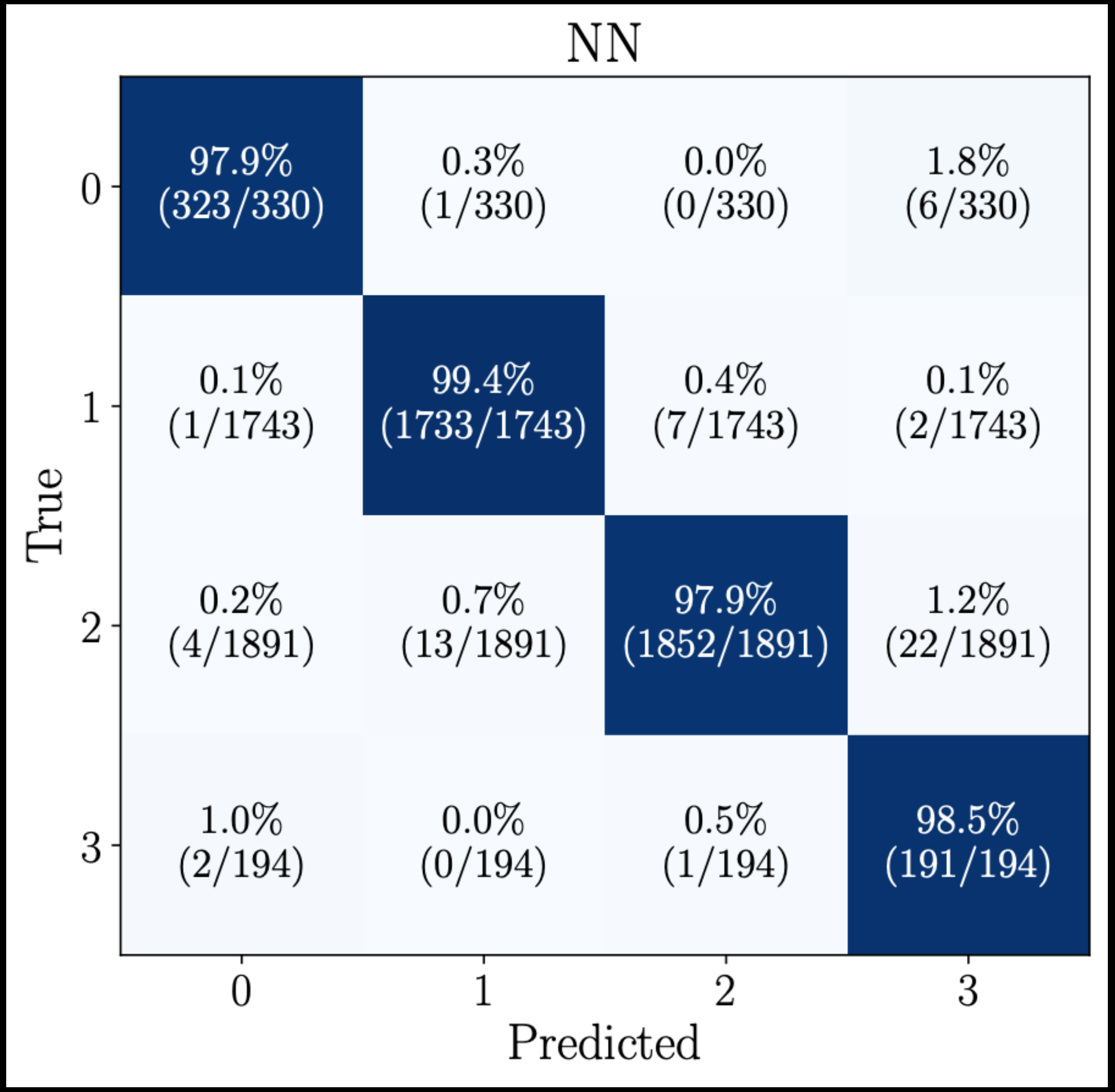
  $M_{2,\,\text{final}}$

Regression

We compared performance for k-Nearest Neighbors, Support Vector Machines, and Neural Networks

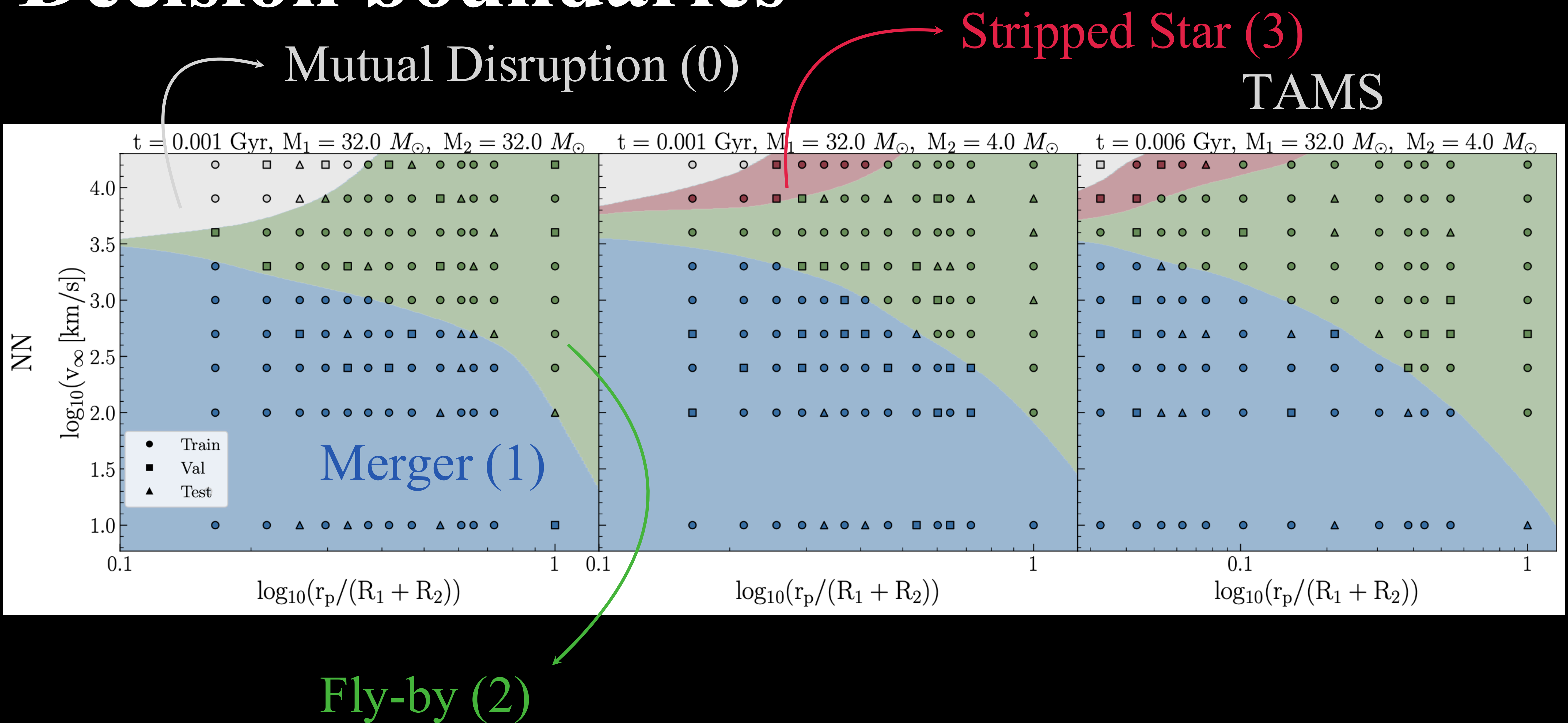
Predicting the final number of stars

- Training set is imbalanced, with only **4.8% stripping** and **8.1% mutually destructive** collisions
- Models are scored based on balanced accuracy



elena.prieto@northwestern.edu

Decision boundaries



- Recovers the symmetries expected for equal-mass collisions
- Learns the dependence on the evolution of the stellar structure

Predicting the final masses

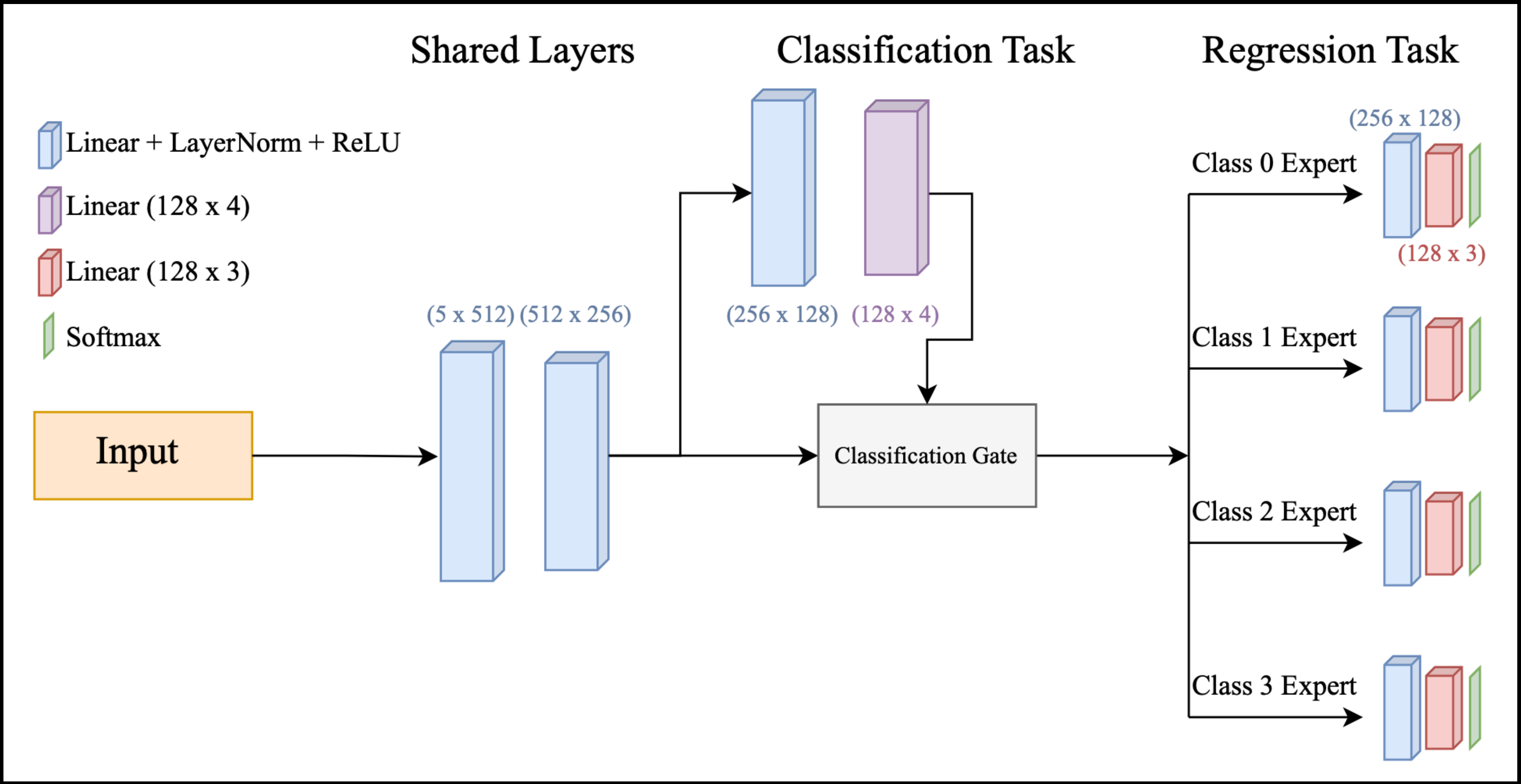
- The NN is trained to predict

$$\left\{\frac{M_{1,\mathrm{f}}}{M_{\mathrm{tot,i}}}, \frac{M_{2,\mathrm{f}}}{M_{\mathrm{tot,i}}}, \frac{M_{u,\mathrm{f}}}{M_{\mathrm{tot,i}}}\right\}$$

which ensures mass conservation.

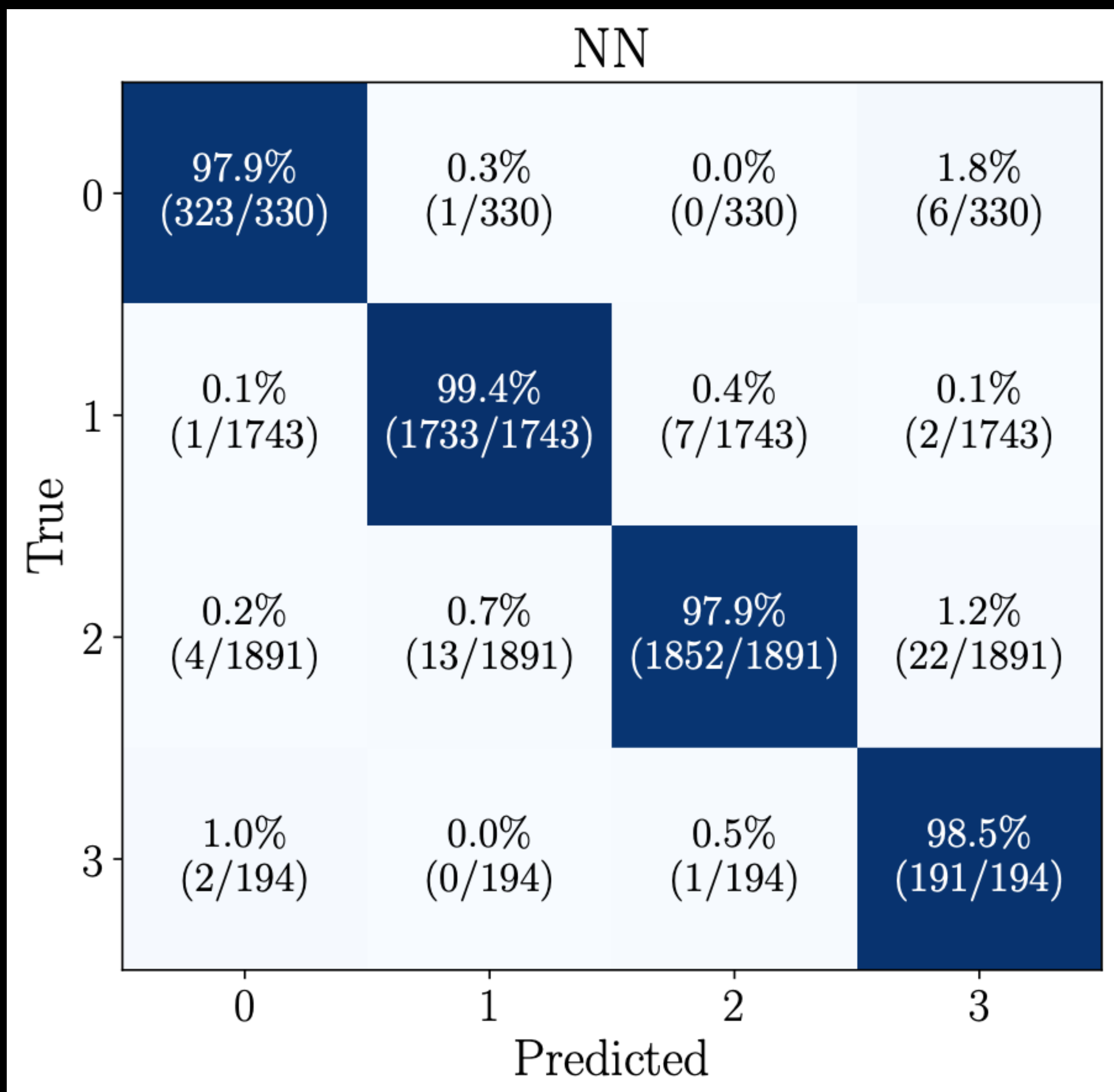
- The NN outperforms all methods, with relative median errors below 0.15% in the final masses

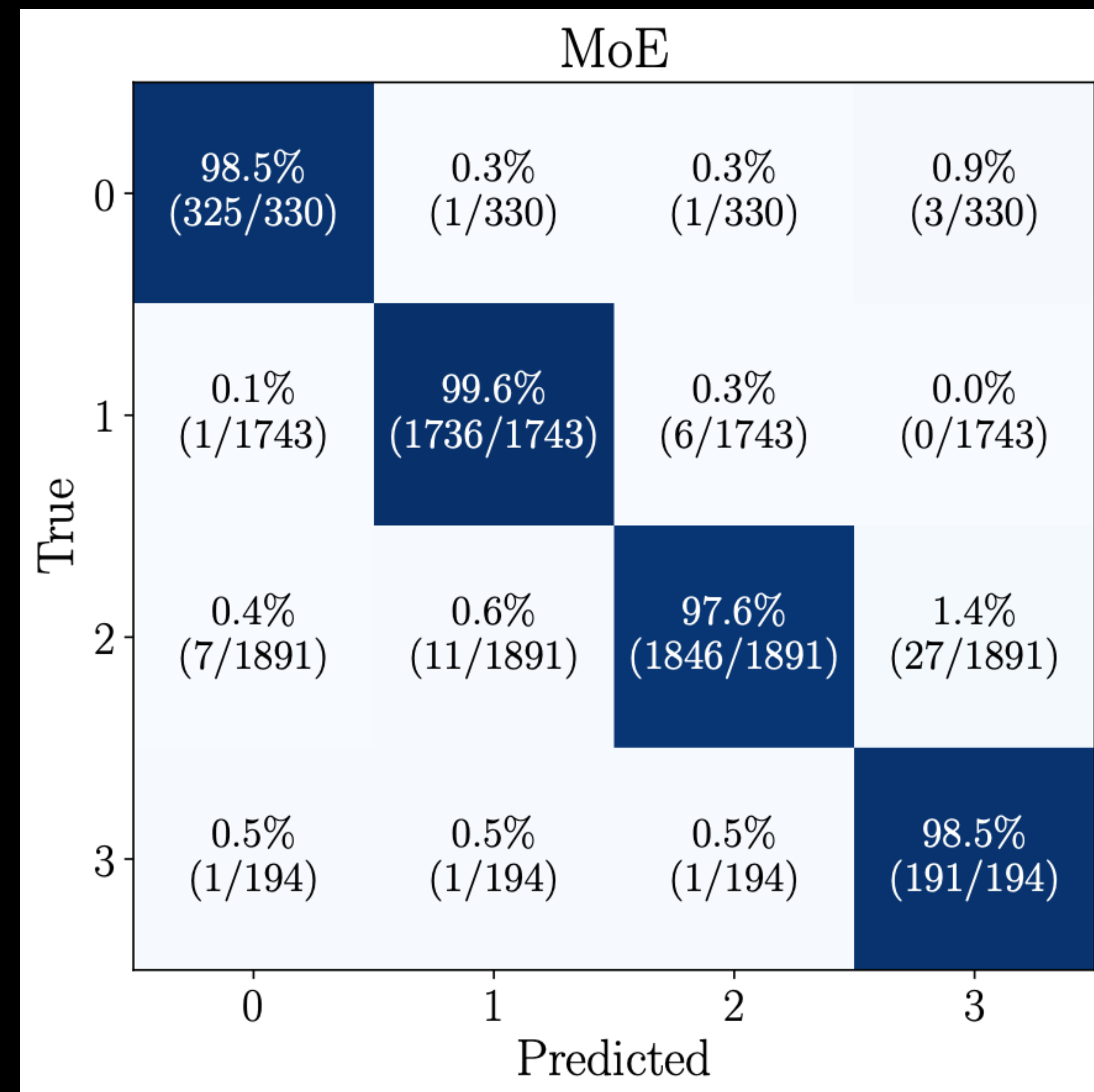
Mixture of Experts Model



elena.prieto@northwestern.edu

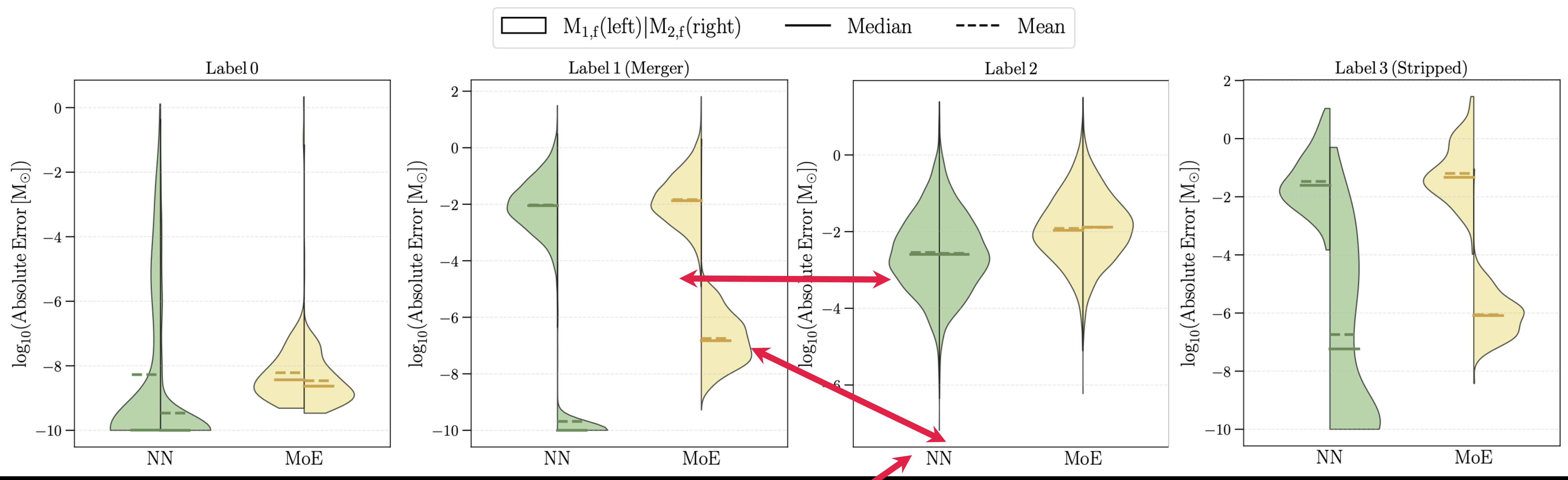
Classification NN vs MoE





Balanced Accuracy: 97.9 ± 0.2 98.2 ± 0.2

Comparison of Errors



Across all labels, the NN has consistently lower median error

Ongoing and Future Work

Sanaea C. Rose
Lindheimer Postdoctoral Fellow at CIERA
Undergraduate at Allegheny College

Machine Learning

- Uncertainty quantification
- Adding kinematic information
- Expanding to other metallicities
- Collisions beyond the main sequence

N-body Simulations

- Modeling stellar collisions in N-body simulations of globular and nuclear star clusters with collAIder
- Integrating collAIder into N-body gravitational scattering experiments

Thank you!

arXiv: 2602.10191

Elena González Prieto

Jamie Lombardi

Fred Rasio

Sanaea C.Rose

Charles F.A. Gibson

Chris O’Connor

Tjitske Starkenburg

Fulya Kiroğlu

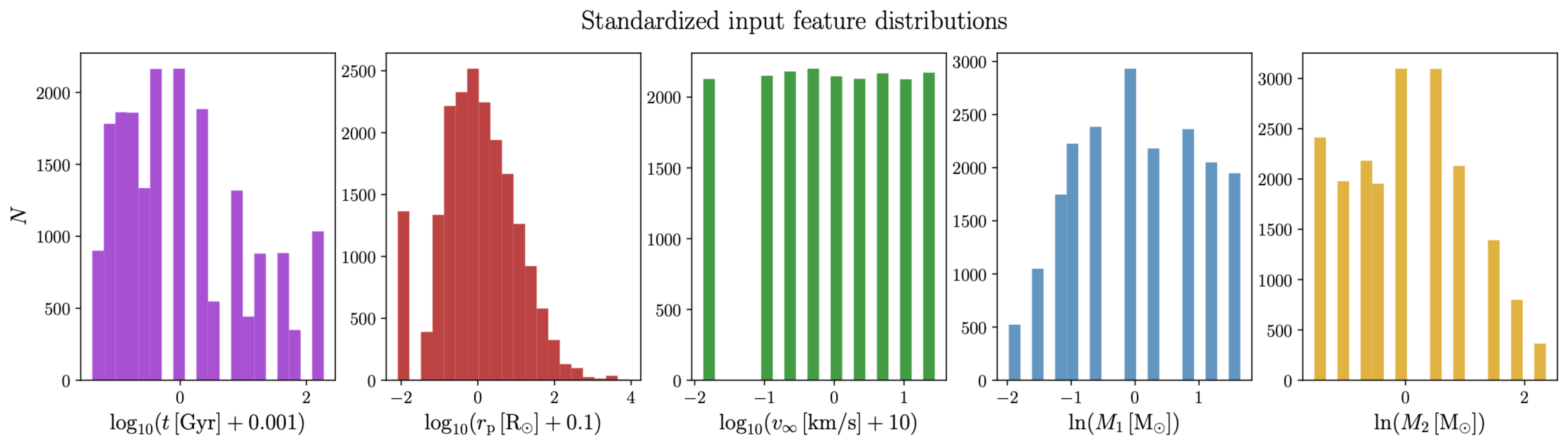
Kyle Kremer

Tristan Sand

elena.prieto@northwestern.edu

Extra Slides

Input Feature Distributions

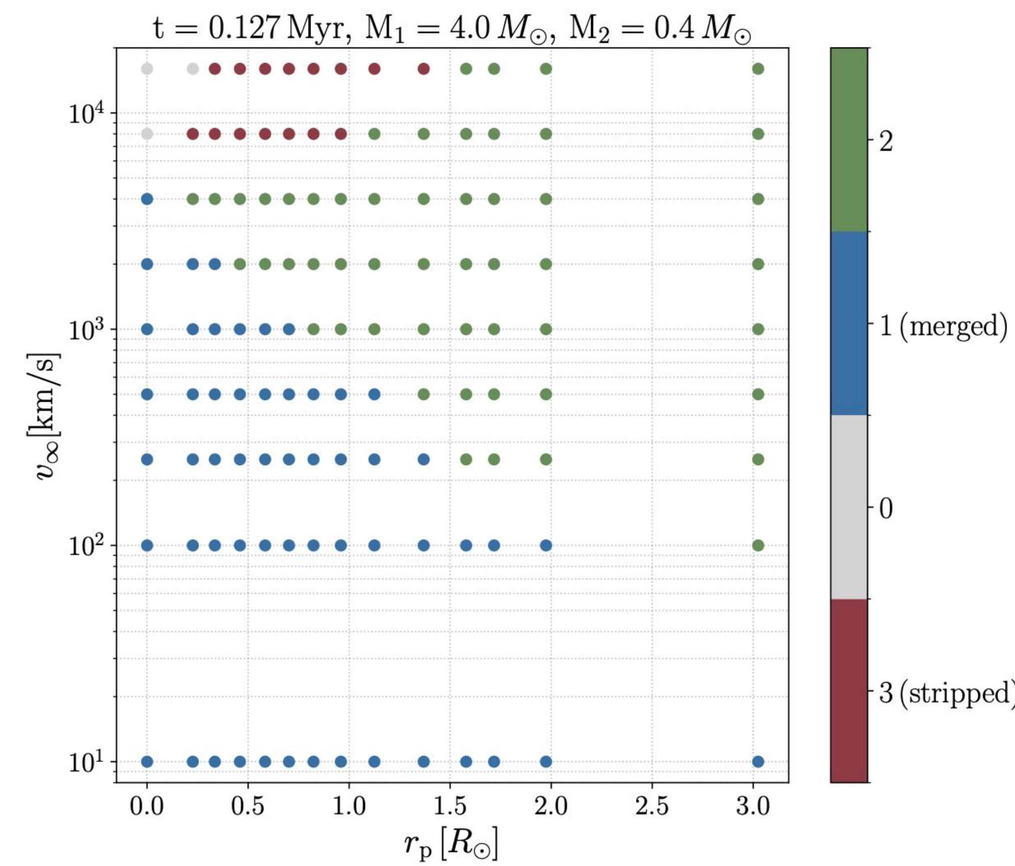


70% Training (19, 404 samples)

15% Validation / Testing (4,158 samples each)

Grid

M_1 \ M_2	$0.2\,M_\odot$	$0.4\,M_\odot$	$0.8\,M_\odot$	$1\,M_\odot$	$2\,M_\odot$	$4\,M_\odot$	$8\,M_\odot$	$16\,M_\odot$	$32\,M_\odot$	$64\,M_\odot$
$64\,M_\odot$					3.00 Myr 4.00 Myr	0.80 Myr 1.00 Myr 3.00 Myr 4.00 Myr	0.20 Myr 1.00 Myr 3.00 Myr 4.00 Myr	0.09 Myr 1.00 Myr 3.00 Myr 4.00 Myr	0.04 Myr 1.00 Myr 3.00 Myr 4.00 Myr	0.02 Myr 1.00 Myr 3.00 Myr 4.00 Myr
$32\,M_\odot$					3.00 Myr 4.00 Myr 5.60 Myr	0.80 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr	0.20 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr	0.09 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr	0.04 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr	
$16\,M_\odot$				13 Myr	3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr	0.80 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr	0.20 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr	0.09 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr		
$8\,M_\odot$			40 Myr	13 Myr 40 Myr	3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr 40 Myr	0.80 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr 40 Myr	0.20 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr 40 Myr			
$4\,M_\odot$	100 Myr 127 Myr	100 Myr 127 Myr	32 Myr 40 Myr 100 Myr 127 Myr	13 Myr 32 Myr 40 Myr 100 Myr 127 Myr	3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr 32 Myr 40 Myr 100 Myr 127 Myr	0.80 Myr 1.00 Myr 3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr 32 Myr 40 Myr 100 Myr 127 Myr				
$2\,M_\odot$	100 Myr 316 Myr 600 Myr	100 Myr 316 Myr 600 Myr	32 Myr 40 Myr 100 Myr 316 Myr 600 Myr	13 Myr 32 Myr 40 Myr 100 Myr 316 Myr 600 Myr	3.00 Myr 4.00 Myr 5.60 Myr 10 Myr 13 Myr 32 Myr 40 Myr 100 Myr 316 Myr 600 Myr					
$1\,M_\odot$	100 Myr 316 Myr 1 Gyr 3.2 Gyr 5 Gyr	100 Myr 316 Myr 1 Gyr 3.2 Gyr 5 Gyr	32 Myr 40 Myr 100 Myr 316 Myr 1 Gyr 3.2 Gyr 5 Gyr	13 Myr 32 Myr 40 Myr 100 Myr 316 Myr 1 Gyr 3.2 Gyr 5 Gyr						
$0.8\,M_\odot$	100 Myr 316 Myr 1 Gyr 3.2 Gyr 10 Gyr 11.2 Gyr	100 Myr 316 Myr 1 Gyr 3.2 Gyr 10 Gyr 11.2 Gyr	32 Myr 40 Myr 100 Myr 316 Myr 1 Gyr 3.2 Gyr 10 Gyr 11.2 Gyr							
$0.4\,M_\odot$	100 Myr 316 Myr 1 Gyr 3.2 Gyr 10 Gyr 13.7 Gyr	100 Myr 316 Myr 1 Gyr 3.2 Gyr 10 Gyr 13.7 Gyr								
$0.2\,M_\odot$	100 Myr 316 Myr 1 Gyr 3.2 Gyr 10 Gyr 13.7 Gyr									



elena.prieto@northwestern.edu

k-NN/SVC Classification Hyperparameters

kNN

Number of neighbors : 3 - 25 (5)

Distance Metric : Euclidean or Manhattan

Weighting Scheme : uniform or inverse distance

SVC

Kernel : Gaussian radial basis function, polynomial (2 - 3)

C (regularization) : 0.1 - 10,000

Gamma : 0.001 - 1 (0.1)

NN Classification Hyperparameters

Table 2
Classification Hyperparameters

Parameter	Optimized Value	Search Range
Batch size	64	64–256
Epochs	500	500–2000
Optimizer	`AdamW`	`AdamW`, `sgd`
Scheduler	`CA-LR`	`CA-LR`, `RLRP`
Learning rate	0.0002	10^{-5}–0.1
`patience`	...	10–60
`factor`	...	0.3–0.5

elena.prieto@northwestern.edu

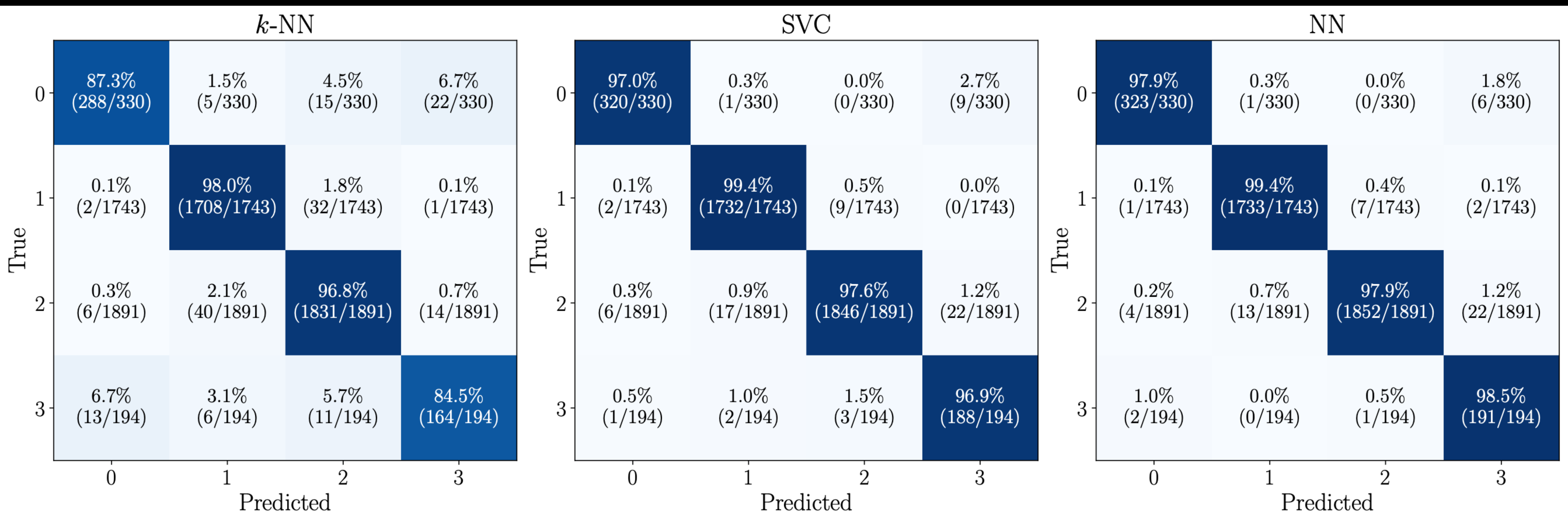
ML Methods Comparison: Classification

Table 1
Classification Performance

Method	Accuracy (%)	Balanced Accuracy (%)
k-NN	96.0	91.7
SVC	98.3	97.7
NN	98.5 ± 0.4 (Best: 98.6)	97.9 ± 0.2 (Best: 98.4)

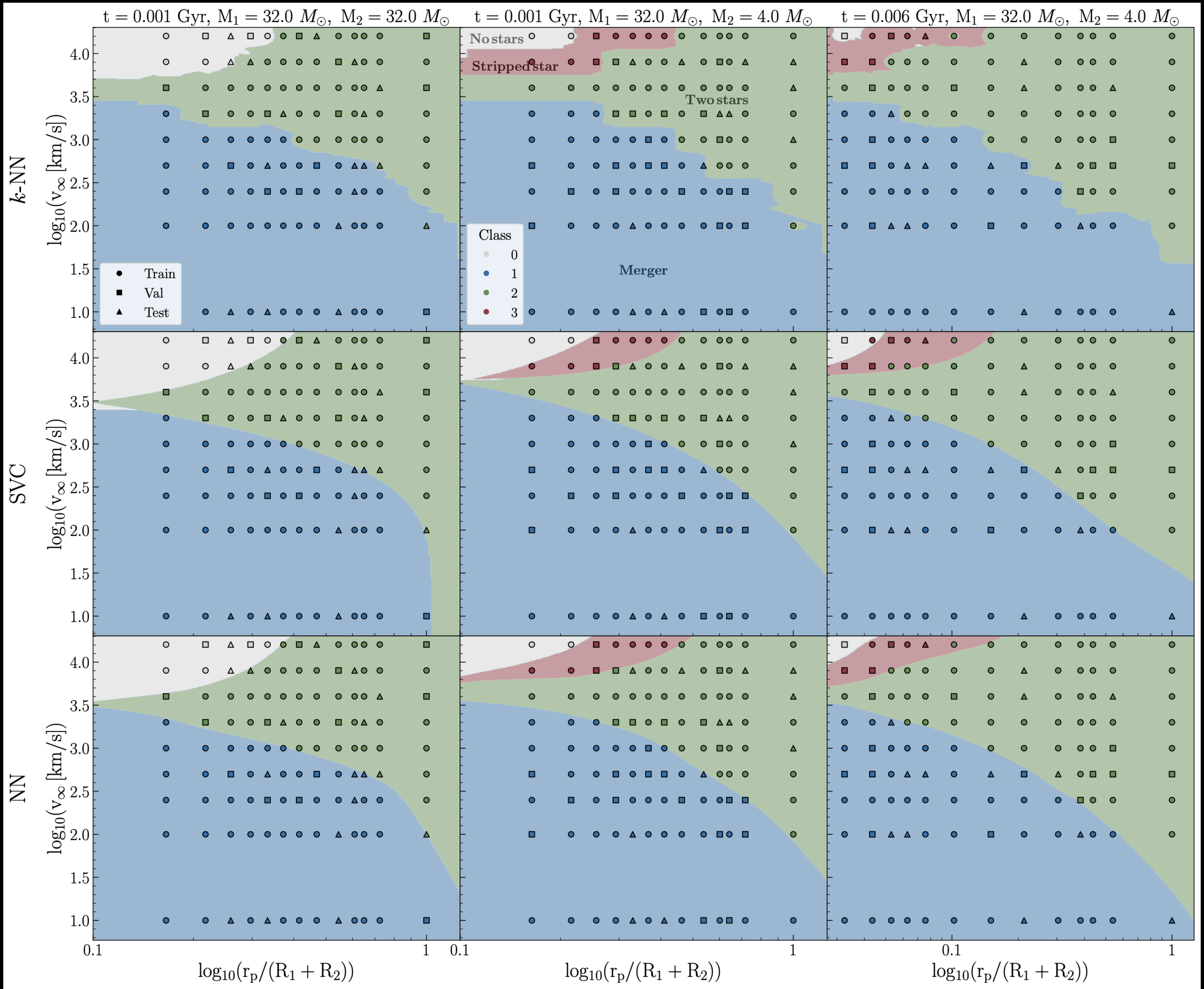
elena.prieto@northwestern.edu

Method Comparison: Classification



elena.prieto@northwestern.edu

Method Comparison: Classification



k-NN Regression

- Evaluated on Mean Absolute Error for the predicted mass fractions

  Number of neighbors : 3 - 25

  Distance Metric : Euclidean or Manhattan

  Weighting Scheme : uniform or inverse distance

elena.prieto@northwestern.edu

Support Vector Machine: Regression

Table 8
SVR Hyperparameter Grid Search Ranges

Quantity	Kernel	`C`	`gamma`	ϵ	Degree
		Initial Coarse Grid Search			
$q_{1,f}$	`rbf`	0.1–100	0.01–10	0.001–0.1	⋯
$q_{1,f}$	Polynomial	0.1–10	0.01–1	0.01–0.1	2–3
$q_{1,f}$	Sigmoid	0.1–1000	0.01–10	0.0001–0.1	⋯
$q_{2,f}$	`rbf`	0.1–100	0.01–10	0.001–0.1	⋯
$q_{2,f}$	Polynomial	0.1–10	0.01–1	0.001–0.1	2–3
$q_{2,f}$	Sigmoid	0.1–1000	0.01–10	0.0001–0.1	⋯
$q_{\mathrm{u},f}$	`rbf`	0.1–100	0.01–10	0.001–0.1	⋯
$q_{\mathrm{u},f}$	Polynomial	0.1–10	0.01–1	0.001–0.1	2–3
$q_{\mathrm{u},f}$	Sigmoid	0.1–1000	0.01–10	0.0001–0.1	⋯
		Refined Fine Grid Search			
All	`rbf`	1	10, 50	0.00001–0.001	⋯
		Optimized Values			
All	`rbf`	1	10	0.00001	⋯

elena.prieto@northwestern.edu

NN Regression Hyperparameters

Table 4
Regression Hyperparameters

Parameter	Optimized Value	Hyperparameter Search Range
Epochs	2000	500–2000
Batch size	64	64–512
Optimizer	`AdamW`	`AdamW, sgd`
Scheduler	`CA-LR`	`CA-LR, RLRP`
Learning rate	0.0002	10^{-5}–0.1
`patience`	⋯	10–120
`factor`	⋯	0.2–0.8
`auxweight`	0.31	0.05–1.0

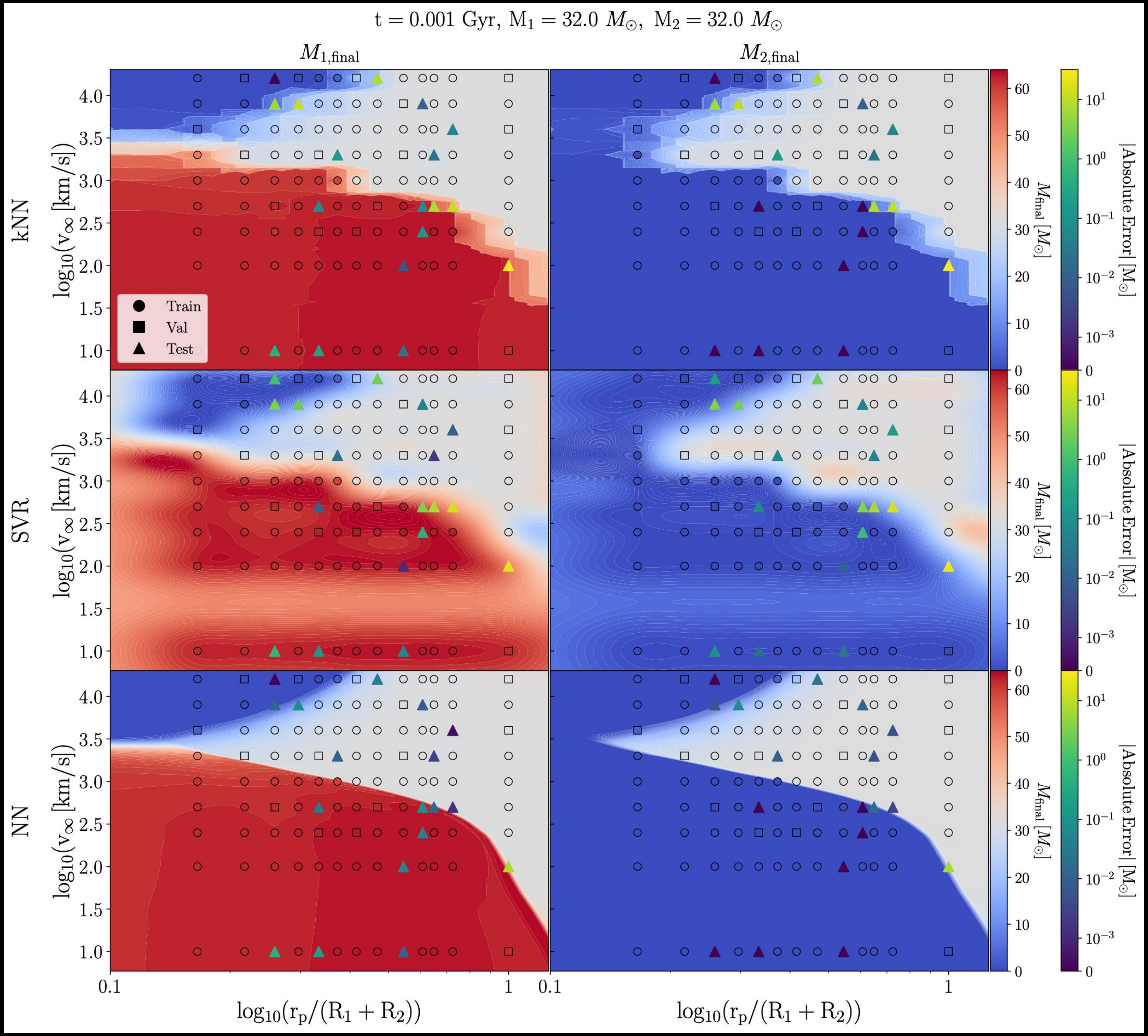
elena.prieto@northwestern.edu

ML Methods Comparison: Regression

Table 3
Regression Performance Metrics

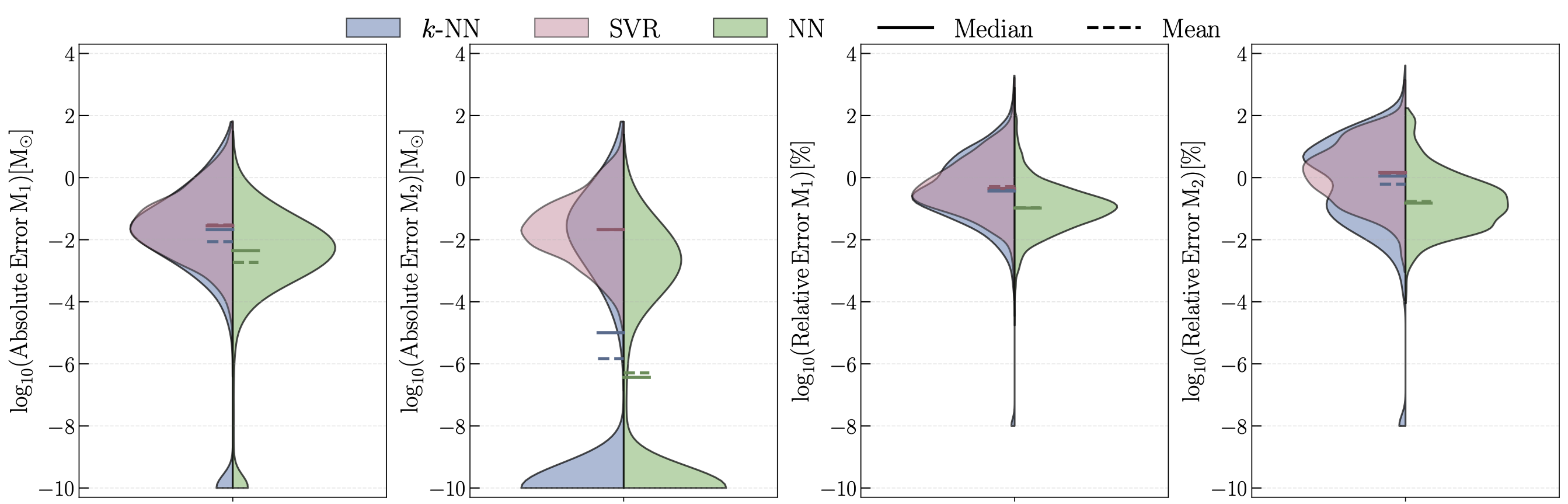
Method	Median Absolute Error ($M_\odot$)		Median Relative Error	
	$M_{1,f}$	$M_{2,f}$	$M_{1,f}$	$M_{2,f}$
k-NN	0.021	1×10^{-5}	0.0037	0.011
SVR	0.028	0.021	0.0045	0.015
NN ($\times 10^{-3}$)	4.67 ± 0.23 (Best: 4.41)	0.00047 ± 0.00038 (Best: 0.00037)	1.09 ± 0.05 (Best: 1.05)	1.76 ± 0.15 (Best: 1.51)

Method Comparison: Regression



elena.prieto@northwestern.edu

Method Comparison: Regression



elena.prieto@northwestern.edu

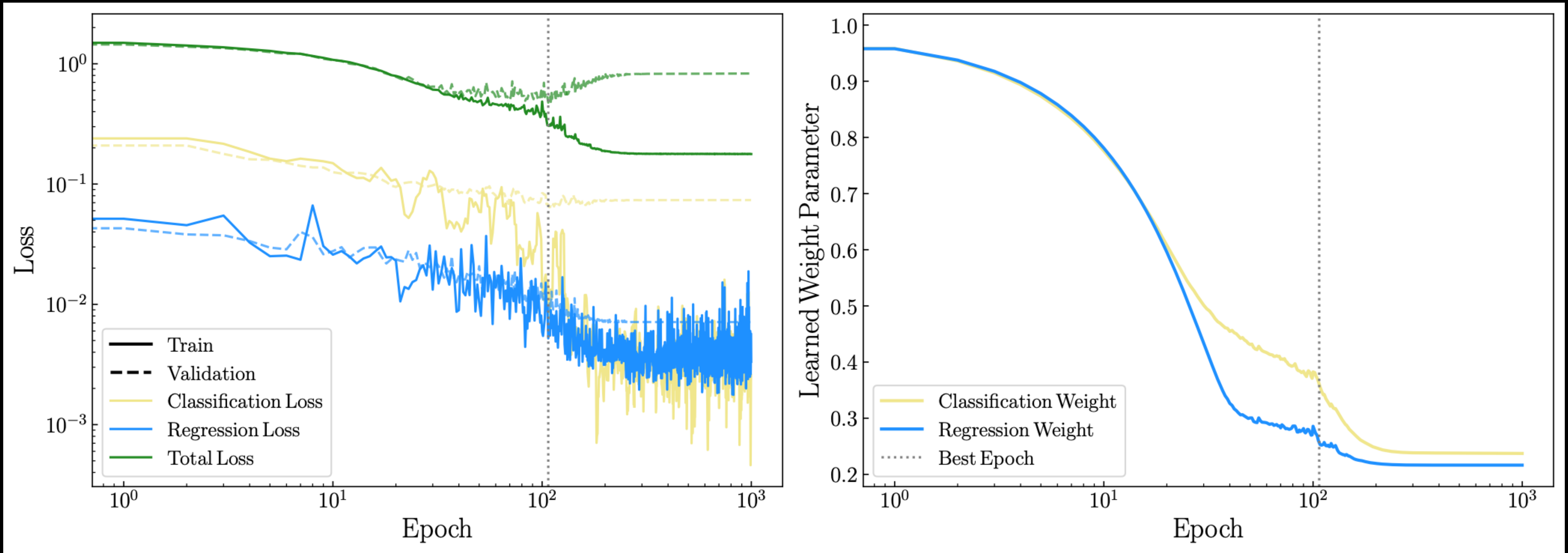
MoE Hyperparameters

Table 5
MoE Hyperparameters

Parameter	Optimized Value	Hyperparameter Search Range
Epoch	500	500–2000
Batch size	128	64–512
Learning rate	0.0004	10^{-5}–0.1
Optimizer	`AdamW`	`AdamW, sgd`
Scheduler	`CA-LR`	`CA-LR, RLRP`
`patience`	···	5–80
`factor`	···	0.2–0.8
`auxweight`	0.77	0.05–1.0

elena.prieto@northwestern.edu

MoE Training



elena.prieto@northwestern.edu

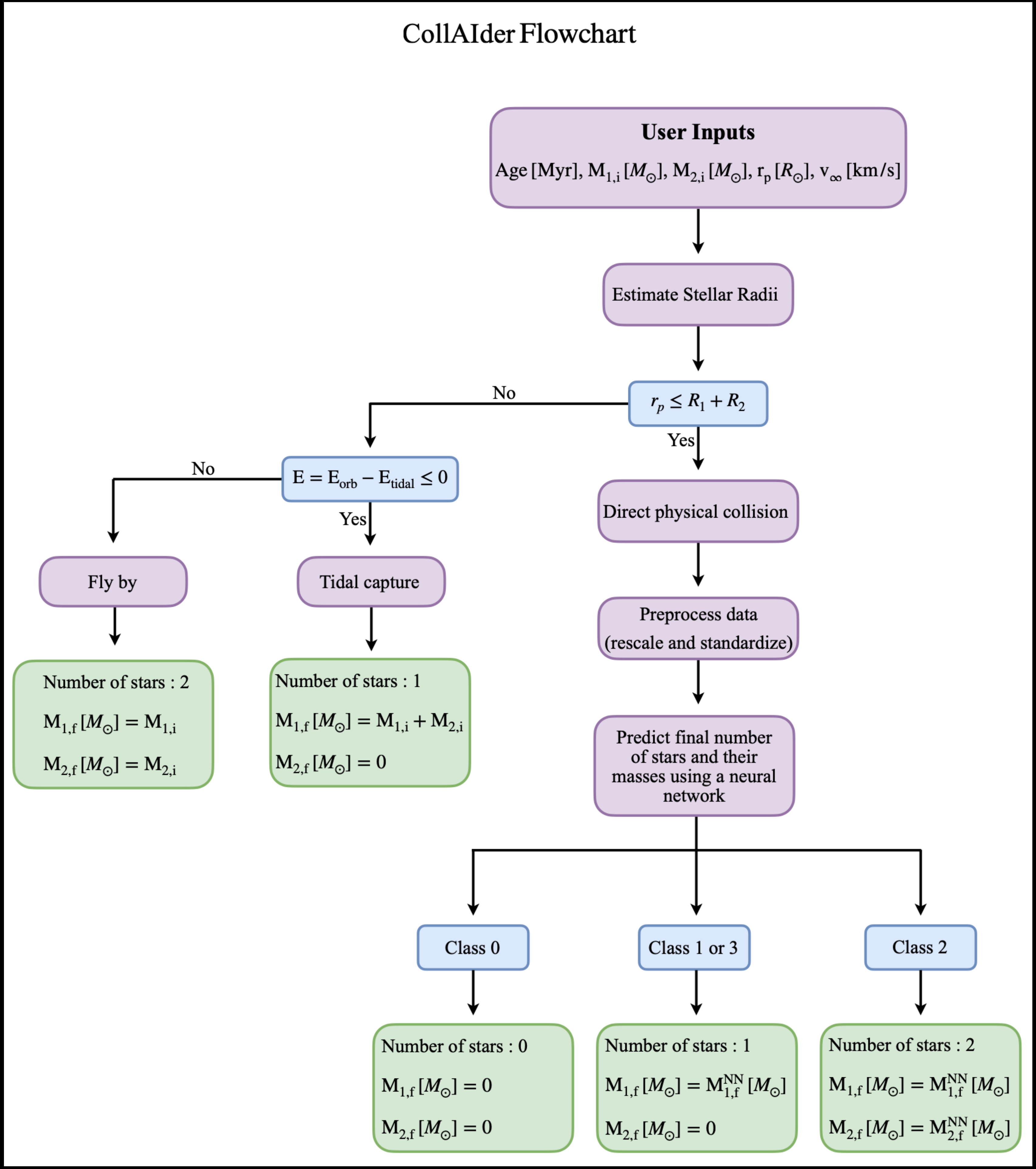
MoE vs NN

Table 6
MoE and Separate NN Performance Metrics

Model	Accuracy (%)	Balanced Accuracy (%)	Median Absolute Error ($M_\odot$)		Median Relative Error	
			$M_{1,f}$ [$\times 10^{-3}$]	$M_{2,f}$ [$\times 10^{-3}$]	$M_{1,f}$ [$\times 10^{-3}$]	$M_{2,f}$ [$\times 10^{-3}$]
MoE	98.5 ± 0.1 (Best: 98.6)	98.2 ± 0.2 (Best: 98.5)	12.5 ± 1.5 (Best: 10.4)	0.01 ± 0.01 (Best: 0.004)	3.1 ± 0.3 (Best: 2.7)	8.4 ± 1.7 (Best: 7.5)
Separate NN	98.5 ± 0.4 (Best: 98.6)	97.9 ± 0.2 (Best: 98.4)	4.67 ± 0.23 (Best: 4.41)	0.00047 ± 0.00038 (Best: 0.00037)	1.09 ± 0.05 (Best: 1.05)	1.76 ± 0.15 (Best: 1.51)

elena.prieto@northwestern.edu

CollAIder Flowchart



CollAIder Performance Metrics

Table 7
`collAIder` Performance Metrics

Dataset	Accuracy (%)	Balanced Accuracy (%)	Median Absolute Error ($M_\odot$)		Median Relative Error	
			$M_{1,f}$	$M_{2,f}$	$M_{1,f}$	$M_{2,f}$
Testing	98.2	98.2	0.0043	0.0	0.0011	0.0014
Interpolation	100	100	0.091	0.0	0.0022	0.0036
TAMS	94.9	89.4	0.0086	0.0	0.0041	0.0044
Extrapolation	90.6	83.8	1.25	0.0	0.011	0.012

- *Interpolation:* different masses within the original grid
- *TAMS:* near the TAMS of a collision within the grid
- *Extrapolation:* out-of-distribution data (massive stars before/at TAMS)

Across all datasets, collAIder achieves a classification balanced accuracy above 83.8% and median relative errors below 1.2% in the final mass predictions