Machine Learning Detection of Non-Axisymmetric Fast Flavor Instabilities in Compact Objects

Madhurima Chakraborty

Institute of Physics, Academia Sinica, Taiwan

The 16th Particle Physics Phenomenology Workshop

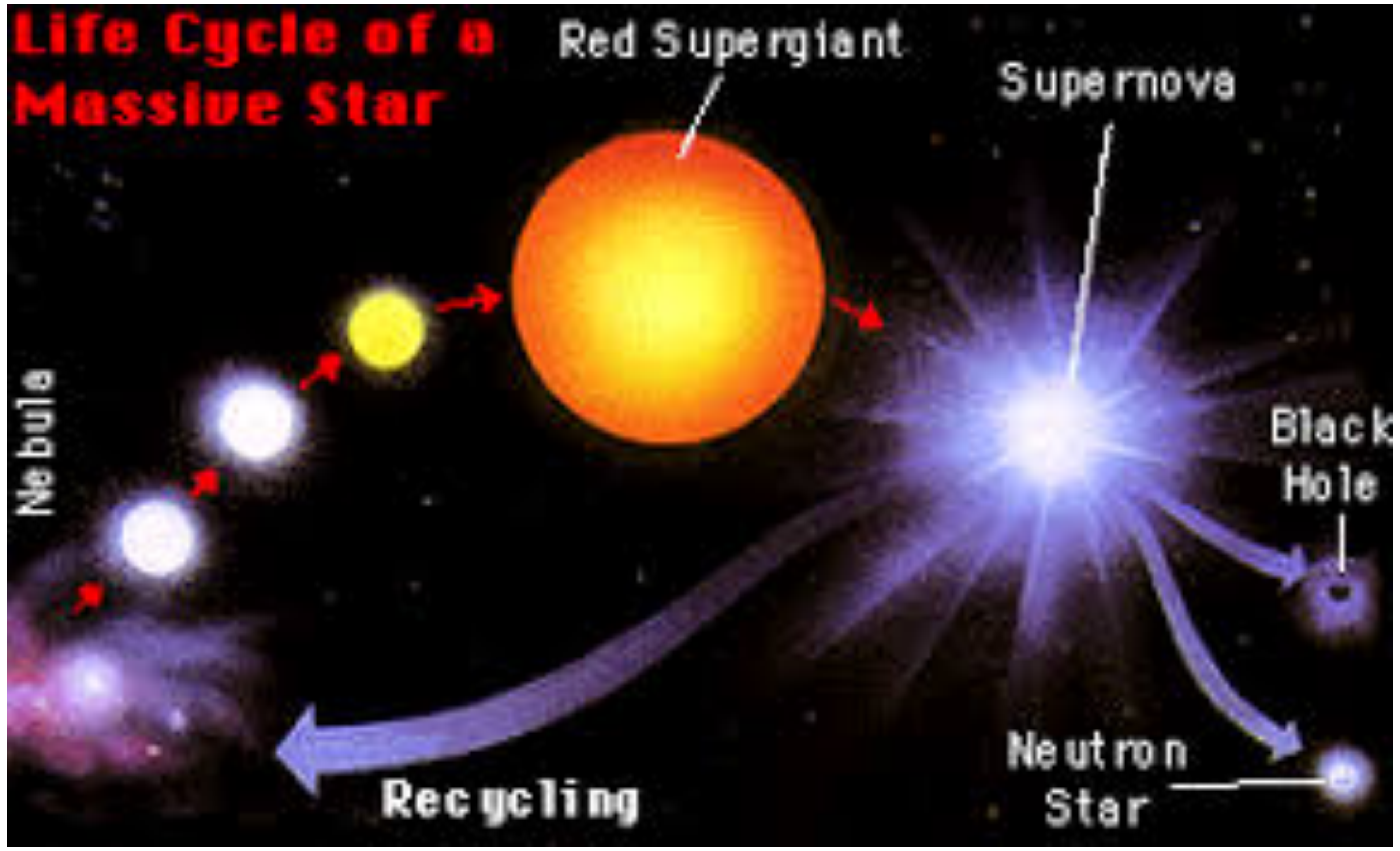
June 17, 2026

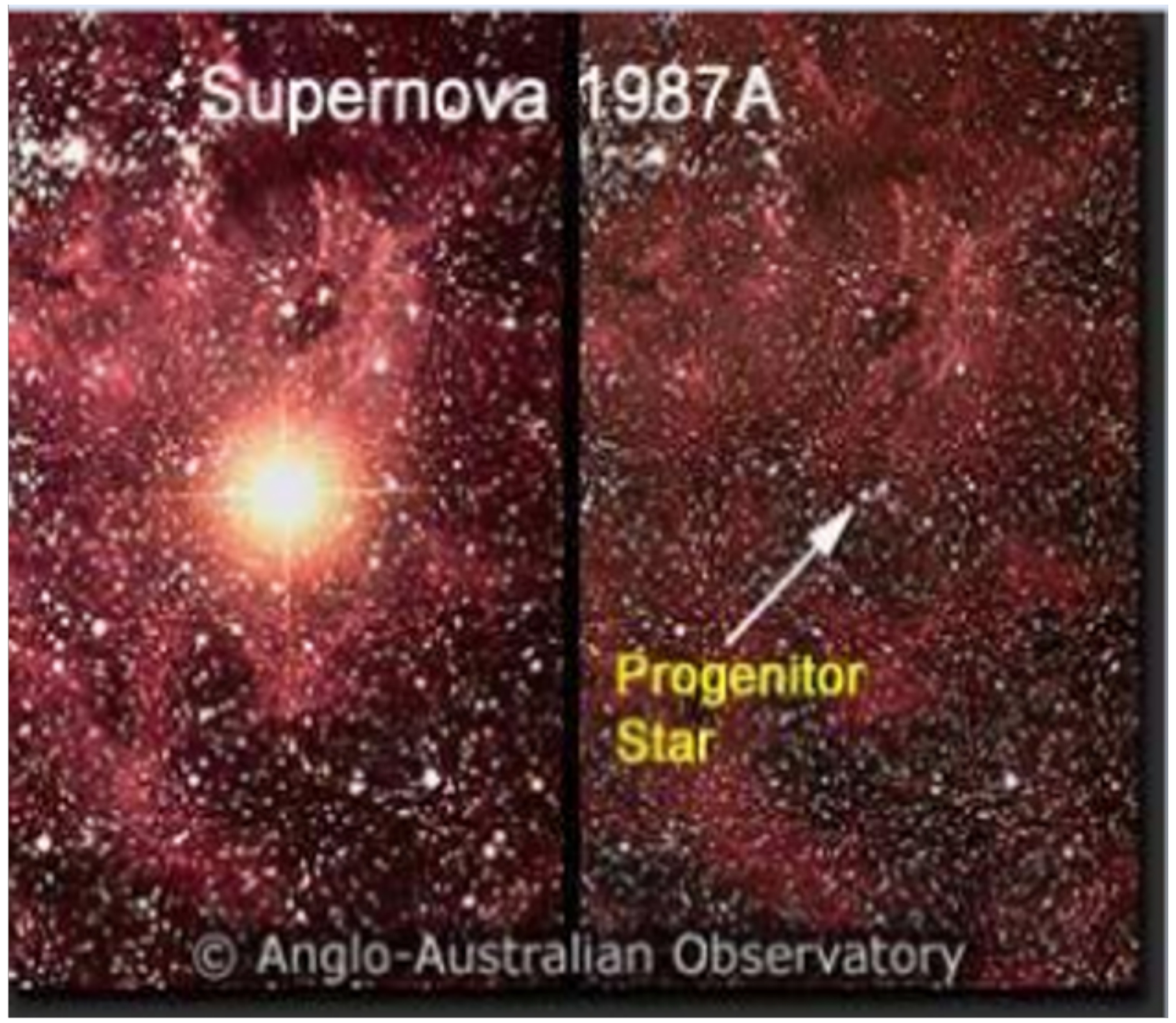
In Preparation with **Meng-Ru Wu** and **Sajad Abbar**

OUTLINE OF TALK

- Core- Collapse Supernova
- Collective Oscillations
- Fast Oscillations
- Machine Learning Detection of Fast Oscillations

CORE-COLLAPSE SUPERNOVA



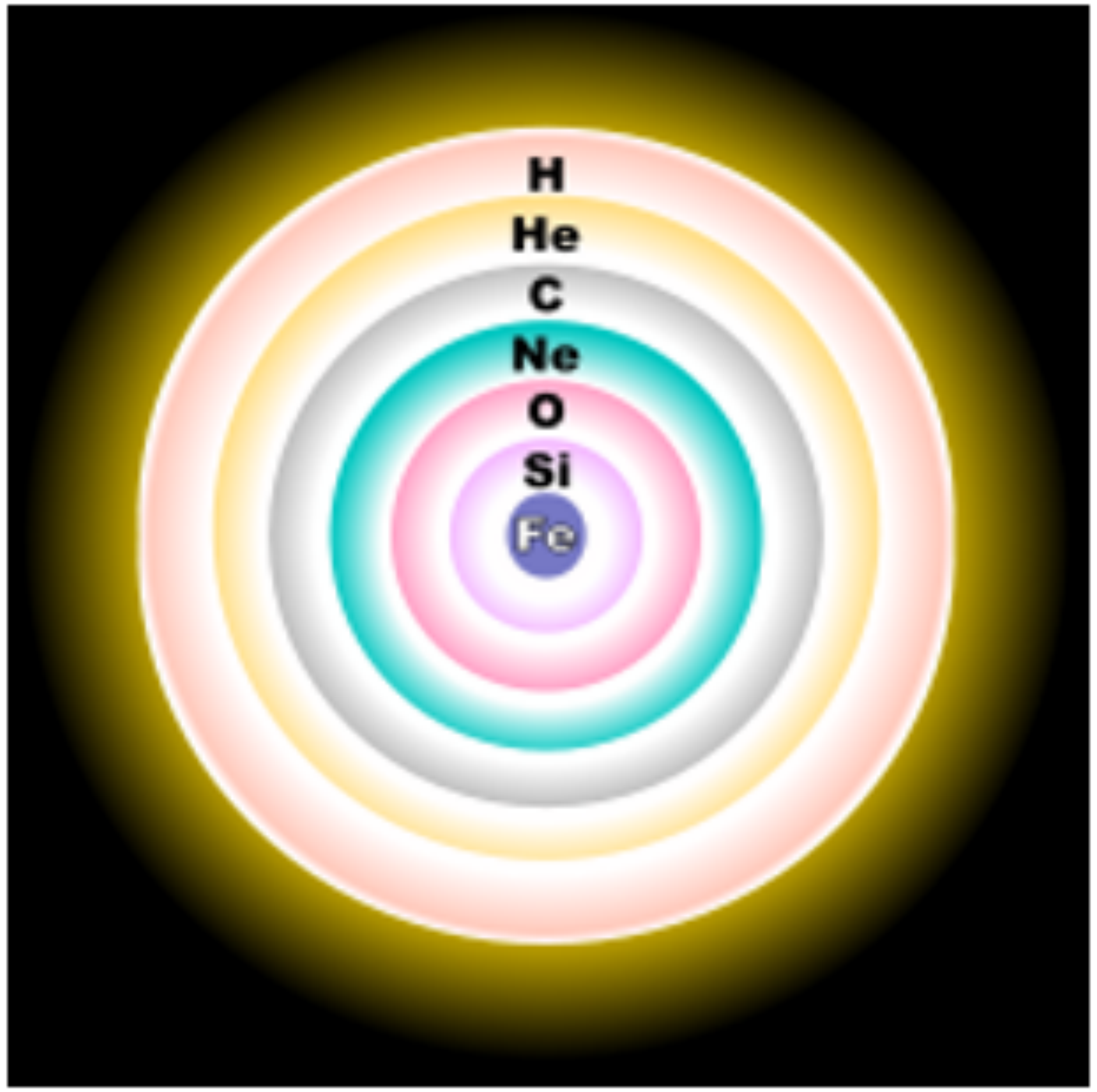


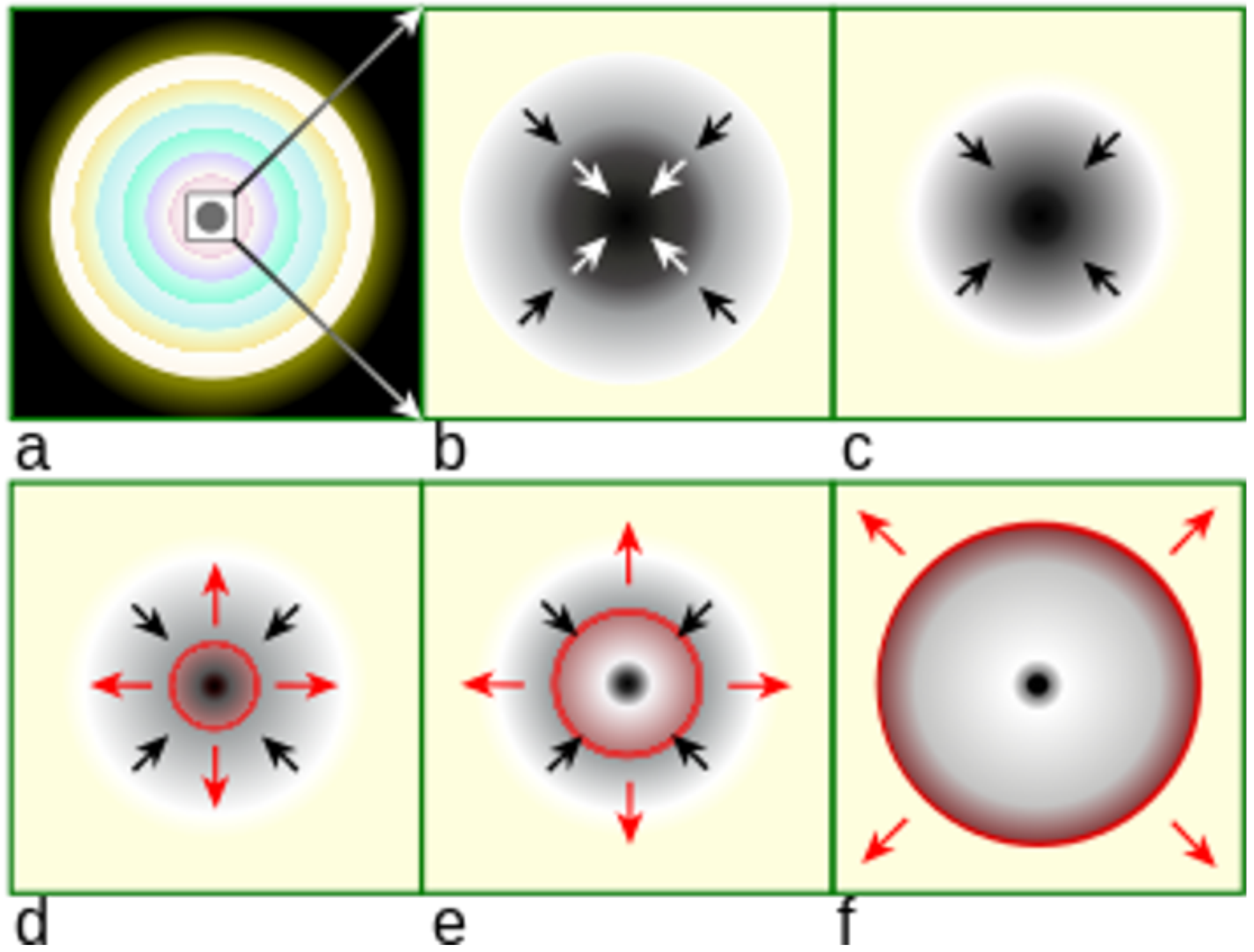
- Occurs from the death of a massive star of

  mass > 8 Msun

- Also known as Type II supernova

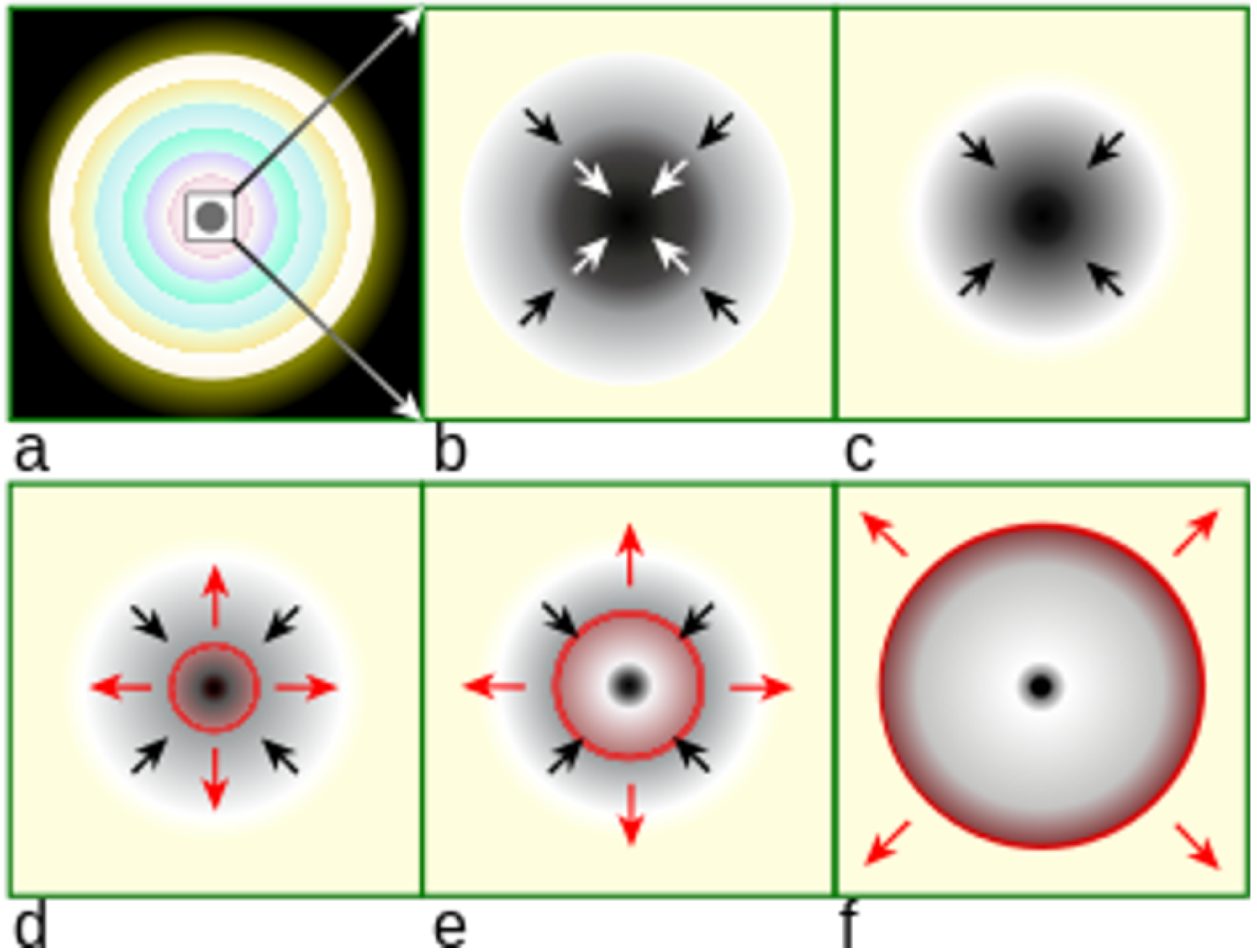
EXPLOSION MECHANISM





Onion ring like structure formed

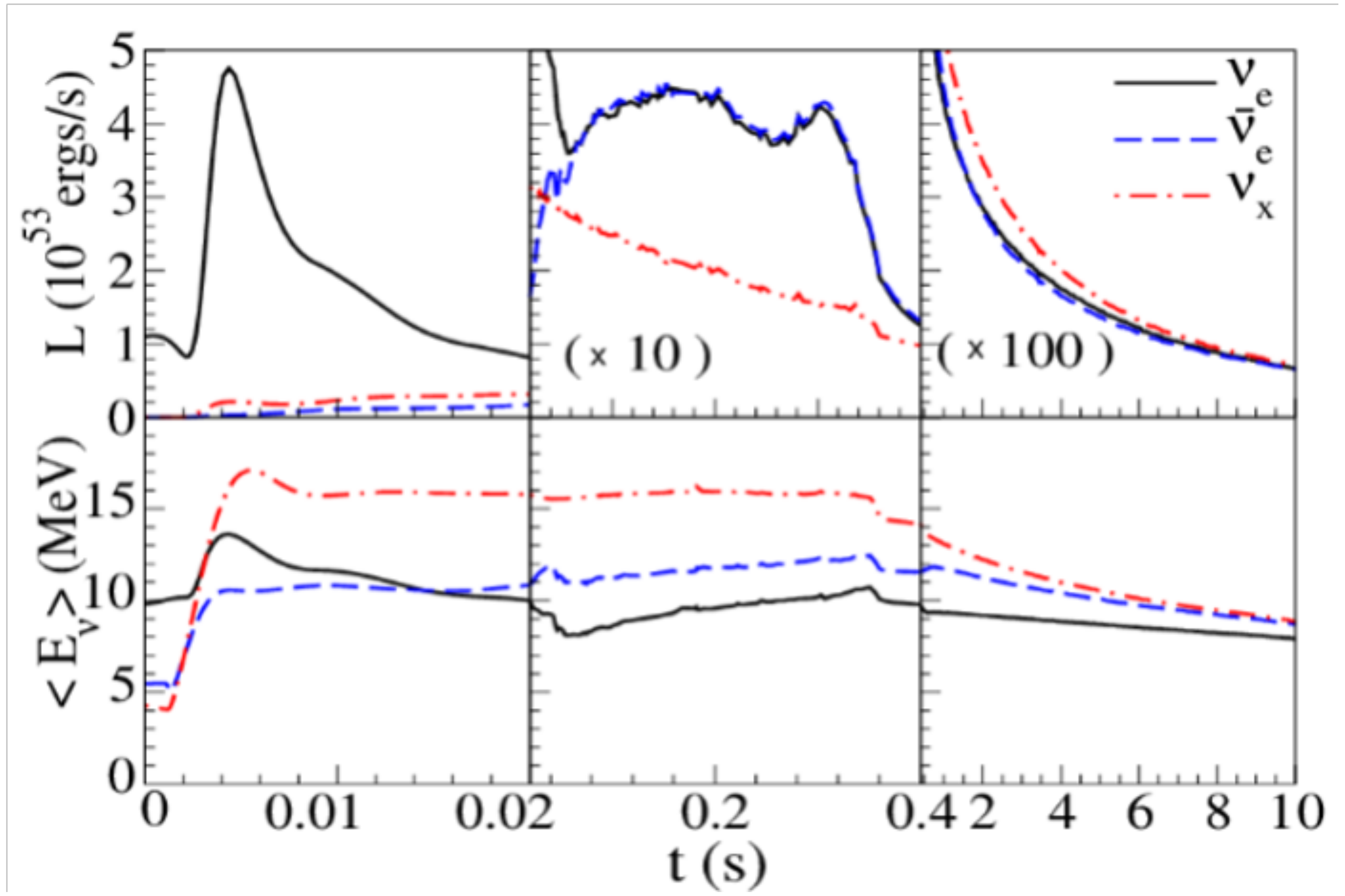
EXPLOSION MECHANISM



- Collapse initiated after star reaches $M_{chandrashekhar}$
- Super nuclear density reached
- Infalling matter bounces back
- Outward moving shockwave formed
- Shock starts to stall but regained due to **neutrino interaction??**
- Outer material blasted away and remnant formed

H. A. Bethe, Rev. Mod. Phys. 62, 801 (1990)

NEUTRINOS : INFORMATION CARRIERS



Obtained from the results of Basel/Darmstadt simulation of

a 18 $M_\odot$ progenitor

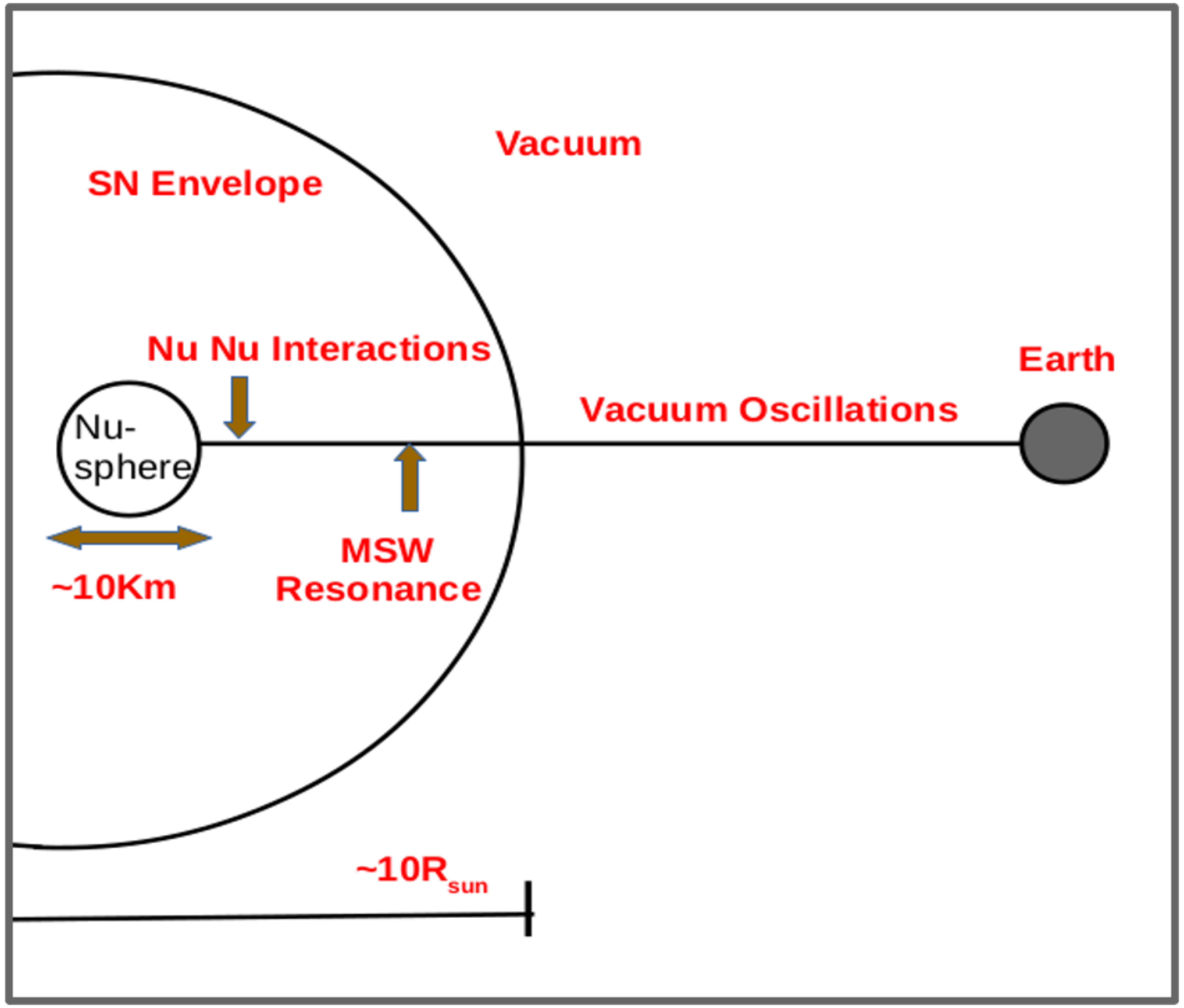
- 99% gravitational energy emitted as neutrinos
- Around 10^{58} neutrinos emitted
- Emission time ~ 10 sec
- Average energy of neutrinos ~ 10 Mev
- Neutrinos emitted in 3 phases :

  Neutronization Burst, Accretion, Cooling phases

H. A. Bethe, Rev. Mod. Phys. 62, 801 (1990)

Chakraborty et al Phys. Rev. D95 (2017)

WHAT HAPPENS TO NEUTRINOS AFTER EMISSION ?



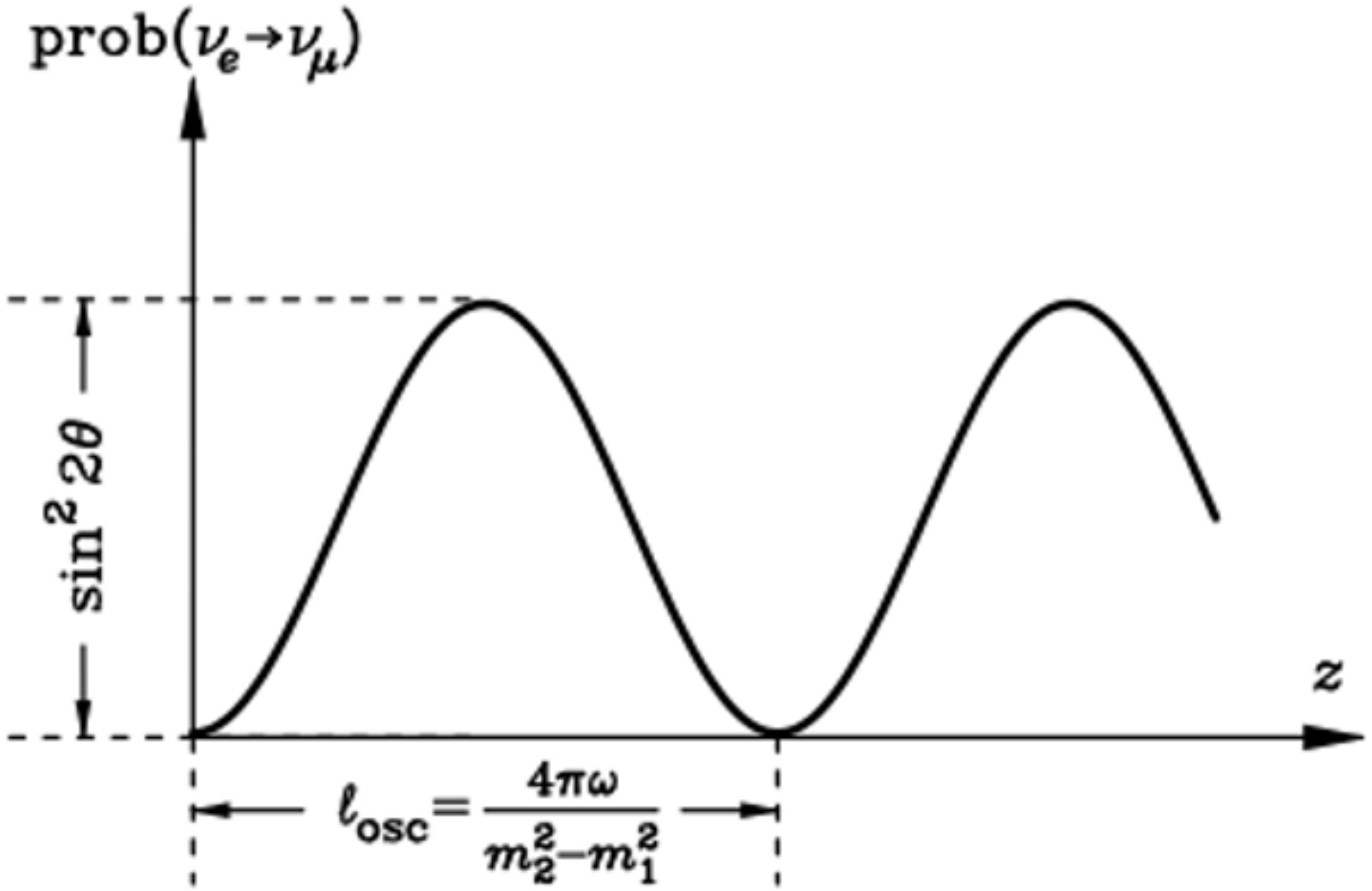
VACUUM OSCILLATIONS

Mass eigenstates - ν_1, ν_2

Flavor eigenstates - ν_e, ν_μ

$$\nu_e = \cos\theta_0\,\nu_1 + \sin\theta_0\,\nu_2$$
$$\nu_\mu = -\sin\theta_0\,\nu_1 + \cos\theta_0\,\nu_2$$

θ_0 → Mixing angle



$$\text{P}\ (\nu_e \to \nu_\mu; t) = P(\nu_\mu \to \nu_e; t) = Sin^2 2\theta_0 Sin^2(\frac{\Delta m^2 t}{4E})$$

$$\Delta m^2 = m_2^2 - m_1^2$$

E.K. Akhmedov, hep-ph/0001264

MATTER OSCILLATIONS

Propagation in matter $\neq$ Propagation in vacuum

Mikheyev-Smirnov-Wolfenstein effect (MSW)

$$V_{\text{eff}} = \sqrt{2}\, G_F\, n_e$$

$$\boxed{\sqrt{2}\, G_F\, n_e = \frac{\Delta m^2}{2E} \cos 2\theta_0}$$

MSW resonance condition

E.K. Akhmedov, hep-ph/0001264

COLLECTIVE OSCILLATIONS

- Occurs in dense environment like supernova and neutron star mergers
- Due to extremely high densities of neutrinos and antineutrinos (10^{31}/cc)
- Behave differently from the vacuum and MSW oscillations.
- Occurs as a result of neutrino-neutrino forward scattering.

Interested in studying the evolution of neutrinos as they are emitted

H. Duan et al, Ann. Rev. Nucl. Part. Sci. 60 (2010) 569–594

A. Mirizzi et al, Riv. Nuovo Cim. 39 (2016), no. 1-2 1–112

S. Chakraborty et al, Nucl. Phys. B908 (2016) 366–381

Capozzi F, Saviano N. Universe 8(2):94 (2022)

L. Johns et. al., Ann.Rev.Nucl.Part.Sci. 75 (2025) 1, 399-423

EVOLUTION EQUATION

$$\partial_t \rho_{x,p,t} + \mathbf{v} \,.\, \nabla_{\mathbf{r}} \rho_{\mathbf{x,p,t}} = -\,\mathbf{i}\,[\mathbf{H_p}, \rho_{\mathbf{x,p,t}}]$$

$$\mathrm{H} = \mathrm{H}_{vacuum} + H_{matter} + H_{\nu\nu}$$

$$H_{\nu\nu} = \sqrt{2} G_F \int dp' \,(1 - \mathbf{v} \,.\, \mathbf{v}')\,(\rho_{\mathbf{p}'} - \overline{\rho_{\mathbf{p}'}})$$

Makes the equation non-linear

$$\rho_p = \begin{pmatrix} \rho_p^{ee} & \rho_p^{e\mu} & \rho_p^{e\tau} \\ \rho_p^{\mu e} & \rho_p^{\mu\mu} & \rho_p^{\mu\tau} \\ \rho_p^{\tau e} & \rho_p^{\tau\mu} & \rho_p^{\tau\tau} \end{pmatrix}$$

Diagonal - overall flavour content

Off-diagonal - flavour coherence

Problem challenging to solve analytically as well as numerically

NEUTRINO DISTRIBUTION

$$\partial_t \rho_{x,p,t} + \mathbf{v} \, . \, \nabla_{\mathbf{r}} \rho_{\mathbf{x,p,t}} = - \, \mathbf{i} \, [\mathbf{H_p}, \rho_{\mathbf{x,p,t}}]$$

- Neutrinos and antineutrinos have some distribution
- Needed to solve the equation

- Energy spectrum - Important for slow oscillations
- Angular spectrum - Important for **fast** oscillations (Θ and Φ)

Presence of crossing in the spectrum necessary

FAST OSCILLATIONS

Neutrino lepton number (νLN):

$$G(\mathbf{v}) = \sqrt{2}\, G_F \int_0^\infty \frac{E_\nu^2\, dE_\nu}{(2\pi)^3} \left[f_{\nu_e}(\mathbf{p}) - f_{\bar{\nu}_e}(\mathbf{p}) - f_{\nu_x}(\mathbf{p}) + f_{\bar{\nu}_x}(\mathbf{p})\right]$$

Zero crossing is a necessary condition for fast flavor oscillations

Similar ν_x and $\bar{\nu}_x$ angular distributions

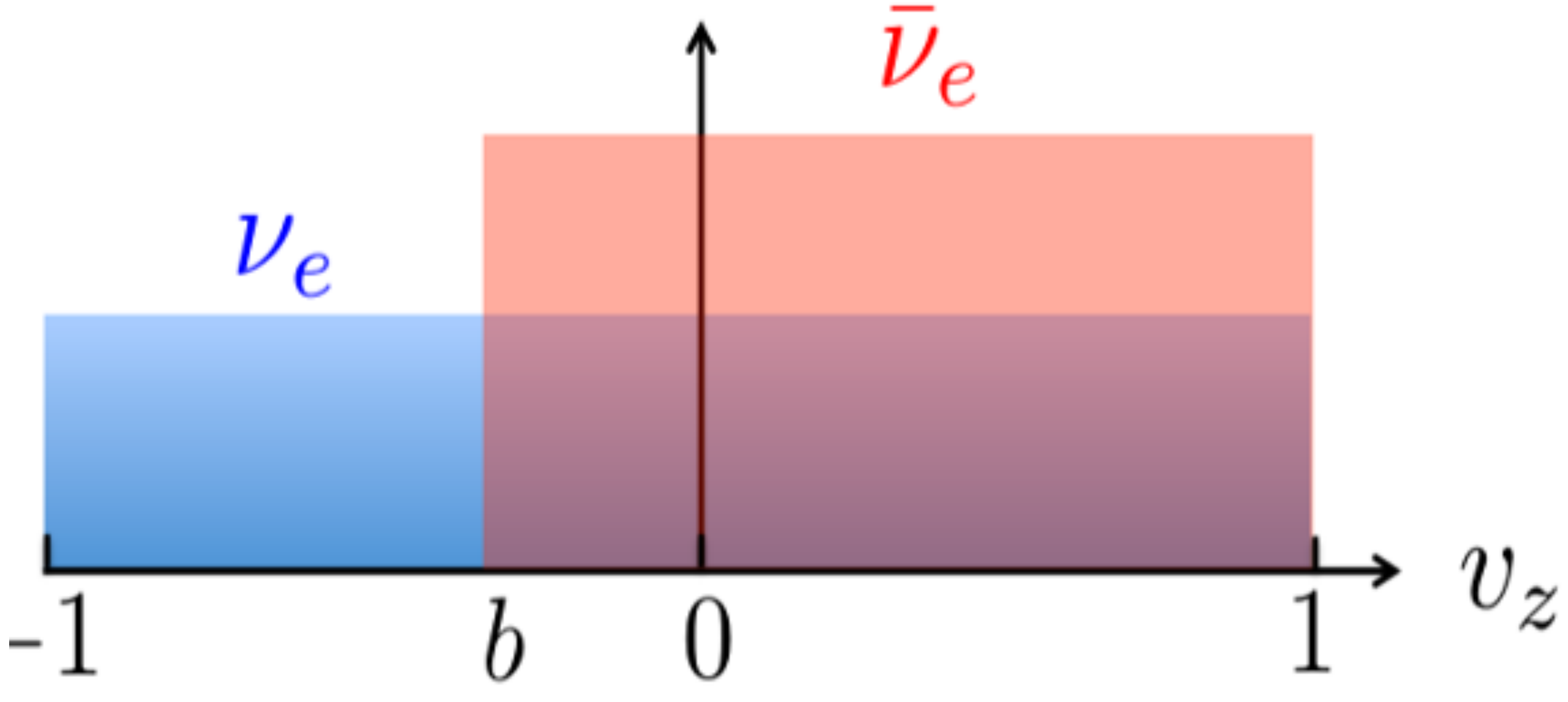
Known as electron lepton number (ELN)

R. F. Sawyer, Phys. Rev. D 72, 045003 (2005)

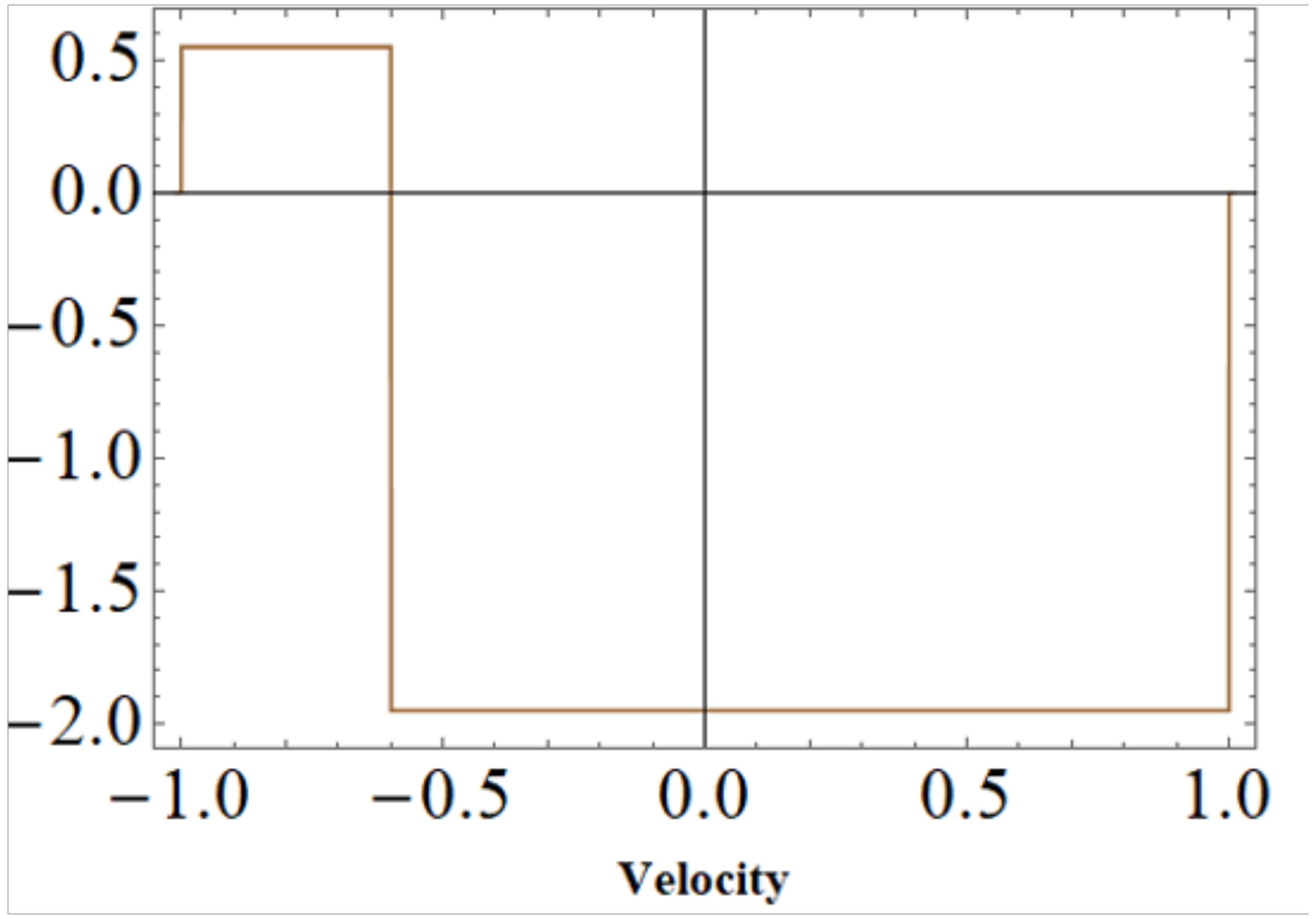
R. F. Sawyer, Phys. Rev. Lett. 116, 081101 (2016)

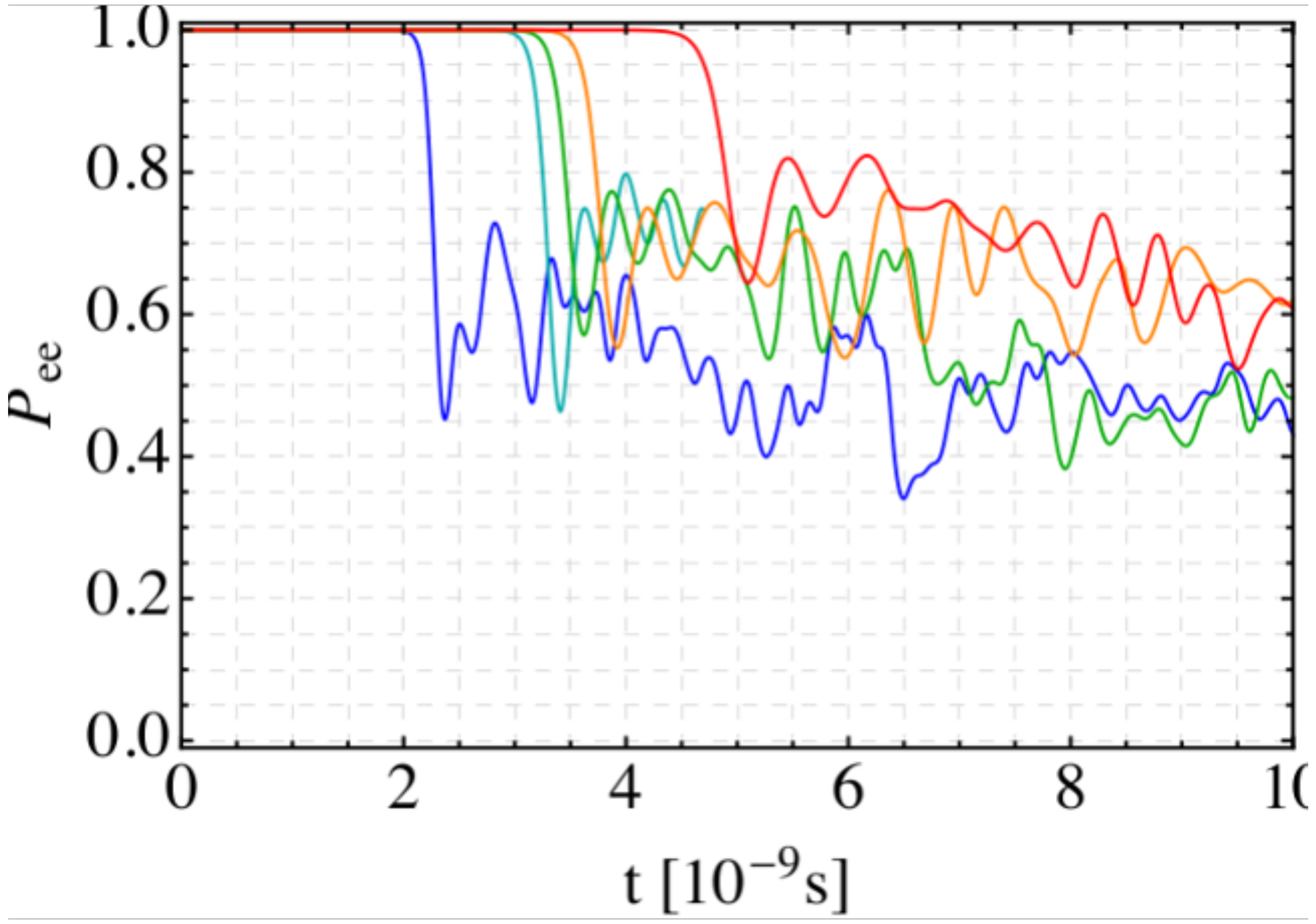
I. Izaguirre, Phys. Rev. Lett. 118, 021101 (2017)........................

FAST OSCILLATIONS



$v_z = \cos\Theta$





B.Dasgupta et. al., *JCAP* 02 (2017) 019

CROSSINGS

- Most hydrodynamic simulations do not provide full angular distributions
- Computationally expensive
- Provide only zeroth and first angular moments, i.e., number density and number density flux
- Analytical methods have been developed to extract angular information from these moments **L.Johns et. al., Phys. Rev. D 101, 043009 (2020)..................**
- But they have certain limitations like they can only be applied in post processing step

APPLICATIONS OF MACHINE LEARNING

- ML algorithms are applied to detect the crossings
- Found to be very efficient in detecting the crossings from the moments
- Provide the opportunity to evaluate the occurrence of fast flavor conversions in CCSN and NSM simulations on the fly
- Computationally cheap once the model is trained

PREVIOUS WORKS

Detecting fast oscillations in CCSN and NSM in axisymmetric setups

S. Abbar, *Phys.Rev.D* 107 (2023) 10, 103006

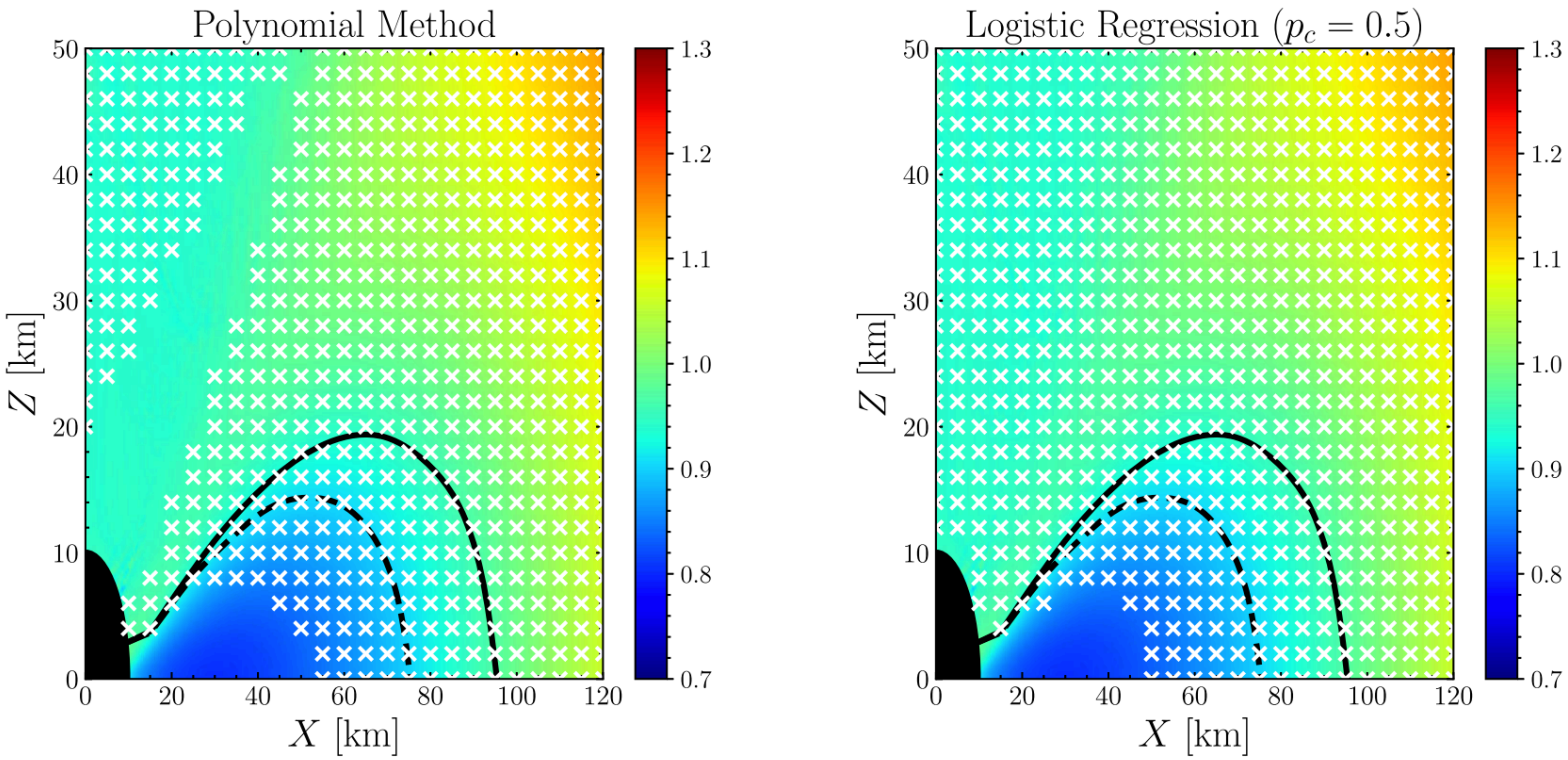
	Logistic regression (93%) accuracy		
	Precision (%)	Recall (%)	F_1-score (%)
No crossing	83	93	88
Crossing	97	93	95

	KNN ($n = 3$) (95%)		
	Precision (%)	Recall (%)	F_1-score (%)
No crossing	90	90	90
Crossing	96	96	96

PREVIOUS WORKS

Detecting fast oscillations in CCSN and NSM in axisymmetric setups

S. Abbar, *Phys.Rev.D* 107 (2023) 10, 103006



PREVIOUS WORKS

Detecting fast oscillations in CCSN in axisymmetric setups

S. Abbar, H. Nagakura, *Phys.Rev.D* 109 (2024) 2, 023033

TABLE I. A summary of the metric scores of the previously-trained ML algorithms (using artificial data) tested on the realistic dataset. This is to be compared with Table I. in Ref. [69]. Alongside each algorithm, one can find its corresponding accuracy score.

LR ($n = 9$) (68%) accuracy			
	Precision	Recall	F_1-score
No crossing	72%	82%	77%
Crossing	59%	43%	50%

KNN ($n = 3$) (77%)			
	Precision	Recall	F_1-score
No crossing	77%	89%	83%
Crossing	75%	55%	63%

SVM (87%)			
	Precision	Recall	F_1-score
No crossing	98%	81%	89%
Crossing	75%	98%	85%

TABLE II. A summary of the metric scores of ML algorithms trained on the combination of the realistic and artificial datasets, and then tested with the realistic data. Alongside each algorithm, one can find its corresponding accuracy score.

LR ($n = 2$) (94%) accuracy			
	Precision	Recall	F_1-score
No crossing	96%	95%	95%
Crossing	91%	93%	92%

KNN ($n = 3$) (98%)			
	Precision	Recall	F_1-score
No crossing	98%	99%	99%
Crossing	98%	97%	98%

PREVIOUS WORKS

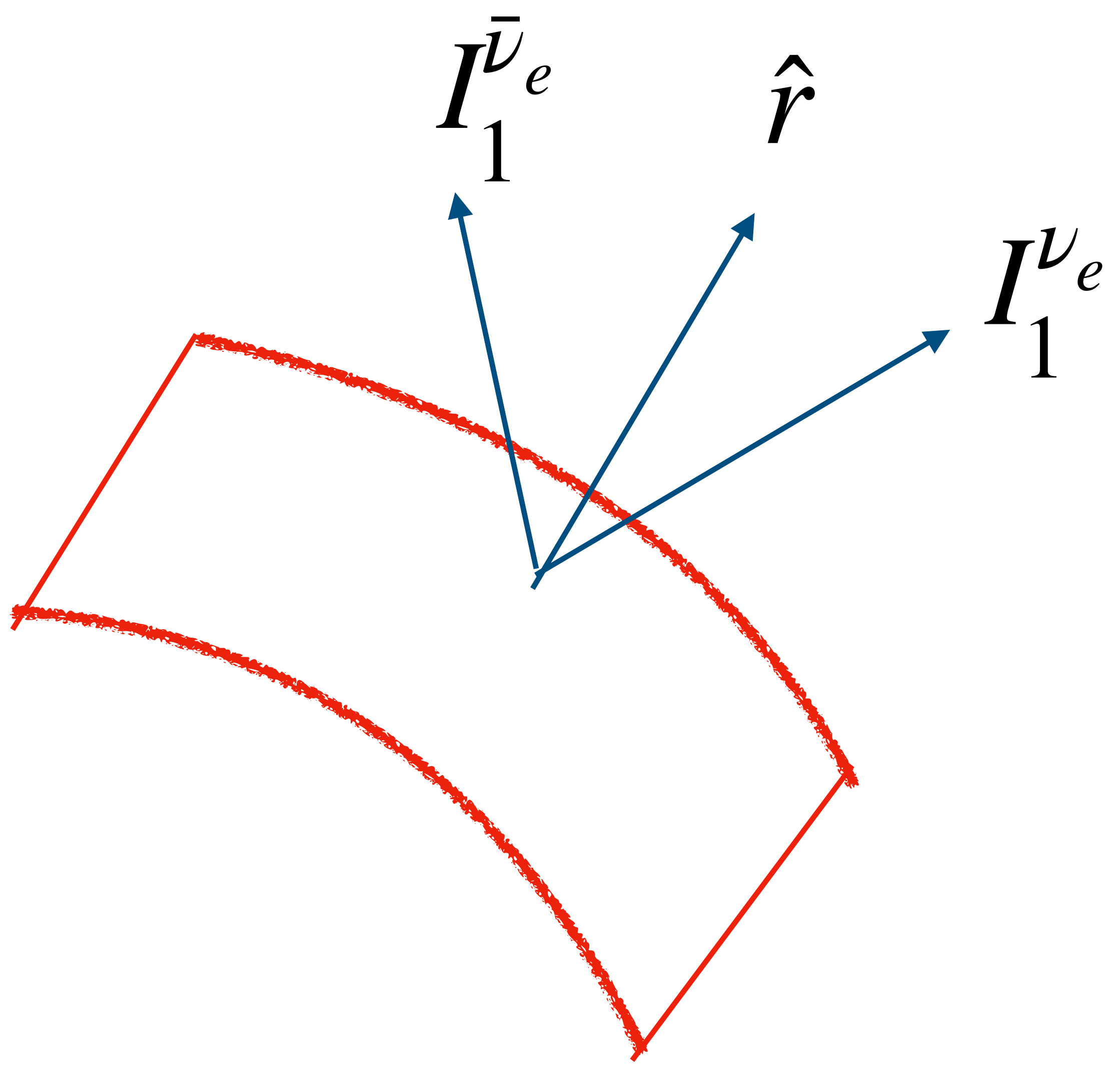
Detecting fast oscillations in CCSN in non-axisymmetric setup

S. Abbar, A. Harada, H. Nagakura, *Phys.Rev.D* 111 (2025) 6, 063077

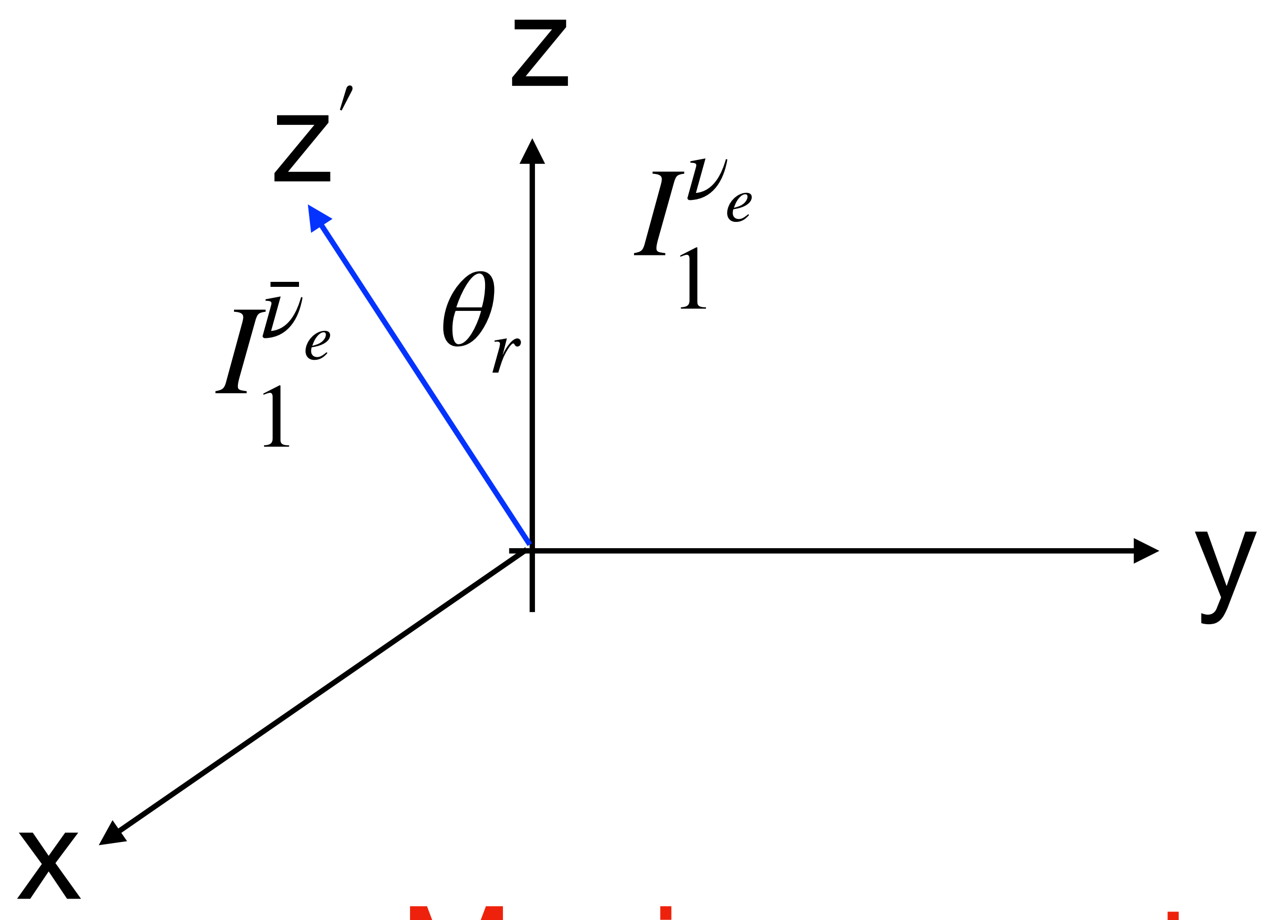
TABLE IV. Summary of the metric scores of the ML algorithms trained on both rotating and nonrotating SN models. Alongside each algorithm, one can find its corresponding accuracy score.

	KNN (n = 3) (99%) accuracy		
	Precision (%)	Recall (%)	F_1-score (%)
No crossing	100	100	100
Crossing	100	100	100
Nonaxisymmetric	93	91	92
	DT (100%)		
	Precision (%)	Recall (%)	F_1-score (%)
No crossing	100	100	100
Crossing	100	100	100
Nonaxisymmetric	93	92	92

OUR MODEL SETUP



Final configuration



Maximum entropy distribution

$$f_{\nu_e} = \exp(-\eta_{\nu_e} + a_{\nu_e} \mathrm{v}_z)$$

$$f_{\bar{\nu}_e} = \exp(-\eta_{\bar{\nu}_e} + a_{\bar{\nu}_e} \mathrm{v}_z')$$

$$\mathrm{v}_z' = \sin\theta_r \mathrm{v}_x + \cos\theta_r \mathrm{v}_z$$

$$\mathrm{v}_x = \sin\theta_\nu \cos\phi_\nu$$

$$\mathrm{v}_z = \cos\theta_\nu$$

MACHINE LEARNING FRAMEWORK

X - ML model -Y

Input

Output

Binary Classification Problem

Output - 0/1

Label 0 - No crossing

Label 1 - Crossing

$$I_n = \int d\Omega \int_0^\infty \frac{E_\nu^2\, dE_\nu}{(2\pi)^3} \mathbf{v}^n f_\nu(\mathbf{p})$$

Non-radial fluxes

$$\boxed{I_0^{\nu_e}, I_0^{\bar{\nu}_e}, I_{1z}^{\nu_e}, I_{1x}^{\bar{\nu}_e}, I_{1z}^{\bar{\nu}_e}}$$

Input Features :

$$\alpha = \frac{n_{\bar{\nu}_e}}{n_{\nu_e}}$$

$$F_z^{\nu_e} = \frac{I_1^{\nu_e}}{I_0^{\nu_e}}$$

$$F_x^{\bar{\nu}_e} = \frac{I_{1x}^{\bar{\nu}_e}}{I_0^{\bar{\nu}_e}}$$

$$F_z^{\bar{\nu}_e} = \frac{I_{1z}^{\bar{\nu}_e}}{I_0^{\bar{\nu}_e}}$$

TRAINING

- Randomly generate the values of $\alpha, F^{\nu_e}, F^{\bar{\nu}_e}$ and obtain the parameters of the angular distributions
- Determine the true labels of output (crossing/no crossing) from these angular distributions
- Train the ML model using these input features and the true output

ML algorithms :

- Logistic Regression
- k-Nearest Neighbours
- Support vector Machines
- Decision Tree Classifier

$$f_\nu = \exp(-\eta_\nu + a_\nu \mathrm{v}_z)$$

TRAINING

Rotation angle - $\theta_r = (0, \pi/2)$

For each θ_r, 100 values of α are uniformly sampled

in the range (0.001, 10)

For each pair of (θ_r, α), 10 random points are

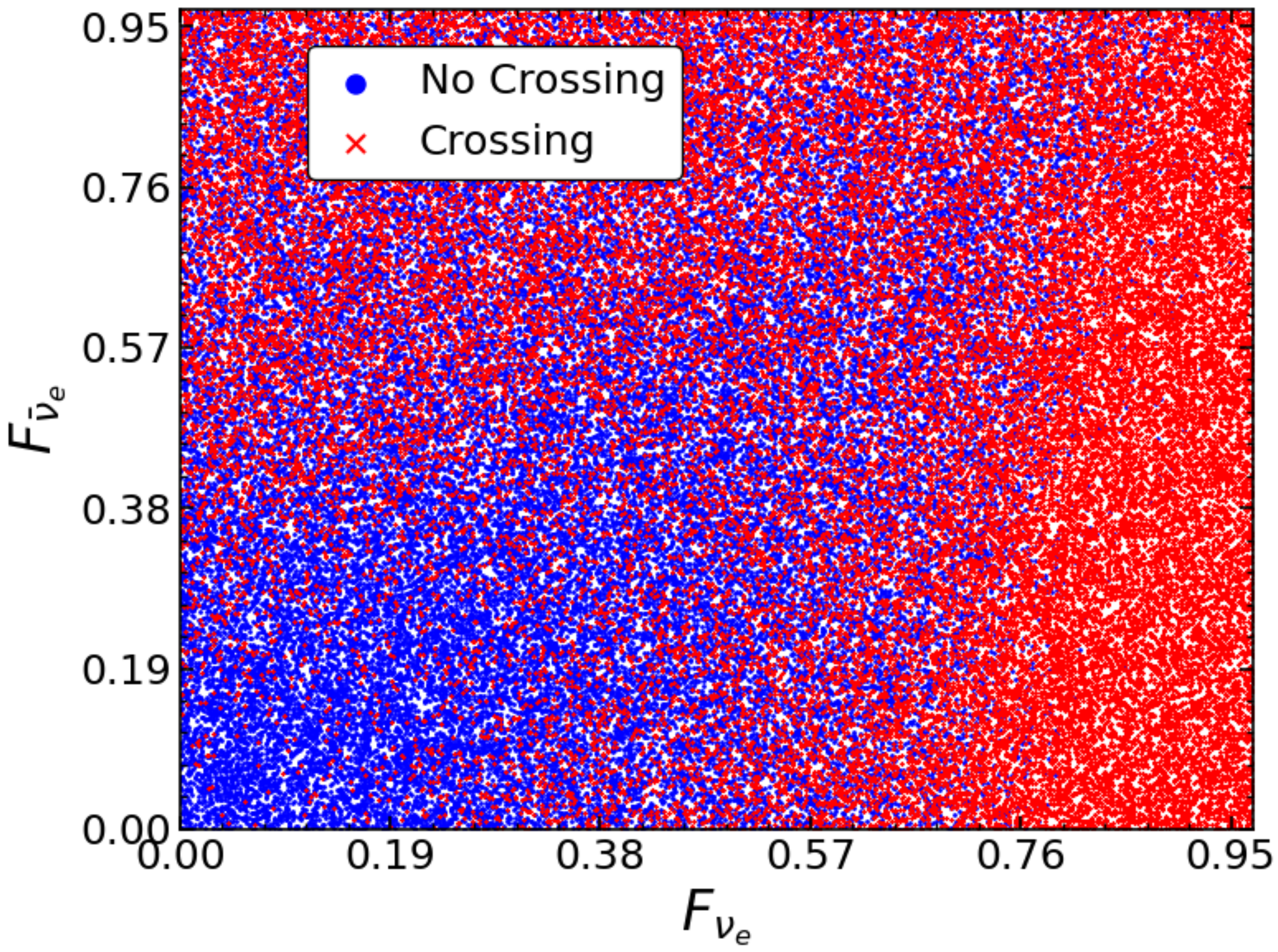
sampled in the $(F^{\nu_e} - F^{\bar{\nu}_e})$ space

Dataset has 10^5 sample points

70% training

15% validation

15% test



PERFORMANCE METRICS

$$\text{accuracy} = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}$$

overall correctness

$$\text{precision} = \frac{T_p}{T_p + F_p}$$

trusting predictions

$$\text{recall} = \frac{T_p}{T_p + F_n}$$

catching all real positives

T_p : True positive

T_n : True negative

F_p : False positive

F_n : False negative

Higher values imply better model

TEST MODELS

Dataset generated from the same method as the training model

Accuracies :

Logistic Regression : 96%

K-Nearest Neighbours : 97%

Support Vector Machines : 98%

Decision Tree : 97%

TEST MODELS

Angular distributions computed using the ray-tracing method

M. R. Wu, et. al. , *Phys.Rev.D* 96 (2017) 12, 123015

above the $\nu_e, \bar{\nu}_e$ emission surfaces (data from 3D NSM simulations)

R. A-Pulpillo, et. al., Mon. Not. Roy. Astron. Soc. 485, 4754–4789 (2019)

Accuracies :

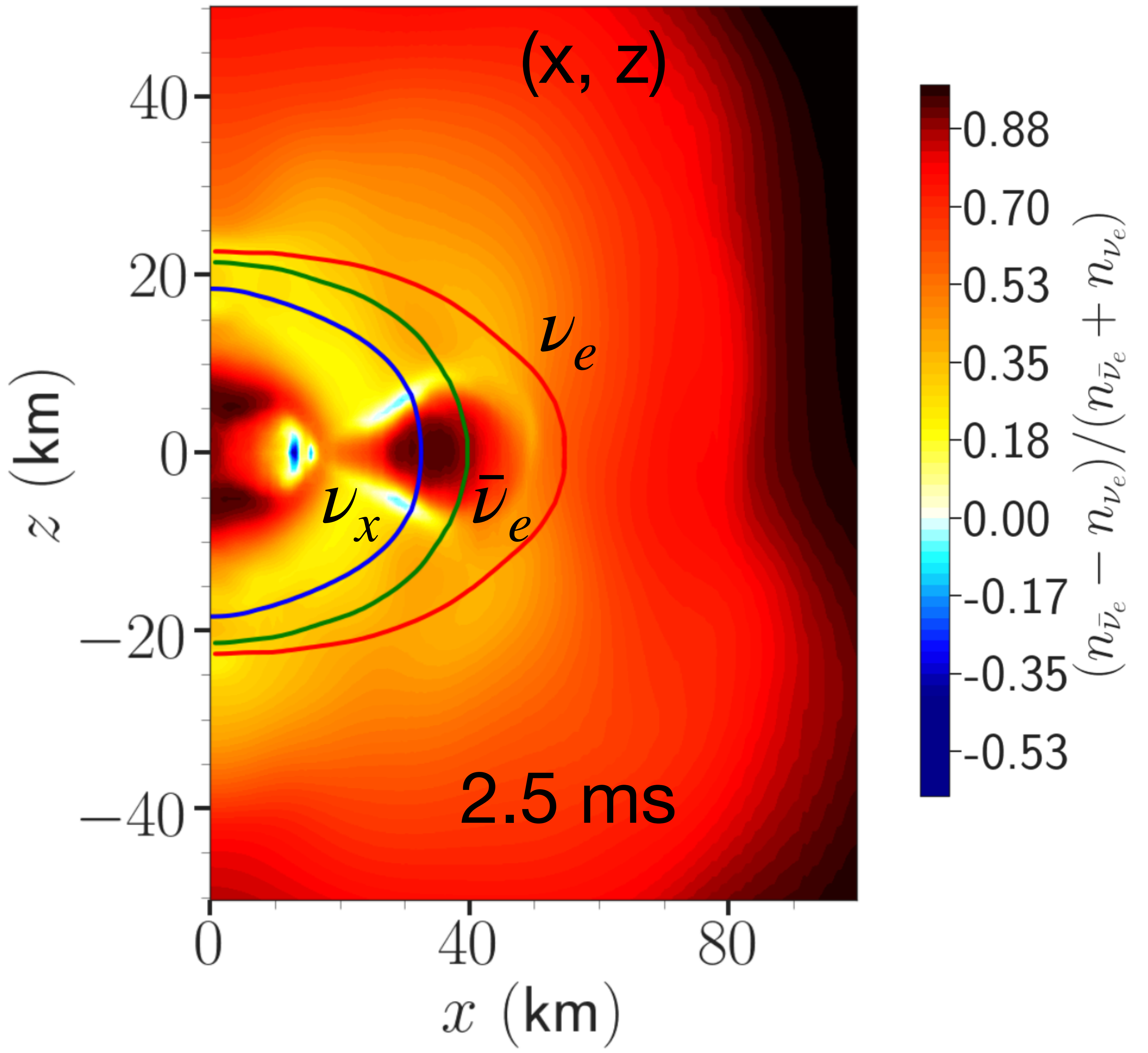
Logistic Regression : 100%

K-Nearest Neighbours : 100%

Support Vector Machines : 100%

Decision Tree : 95%

all samples have crossings



M. George et. al. , *Phys.Rev.D* 102 (2020) 10, 103015

TEST MODELS

Dataset generated from the angular distributions obtained from 1D supernova simulations

Z. Xiong, et. al. , *Phys.Rev.D* 109 (2024) 12, 123008

Axisymmetric

Accuracies :

Logistic Regression : 72%

K-Nearest Neighbours : 78%

Support Vector Machines : 75%

Decision Tree : 61%

Non - axisymmetric ($\theta_r = \pi/6$)

Accuracies :

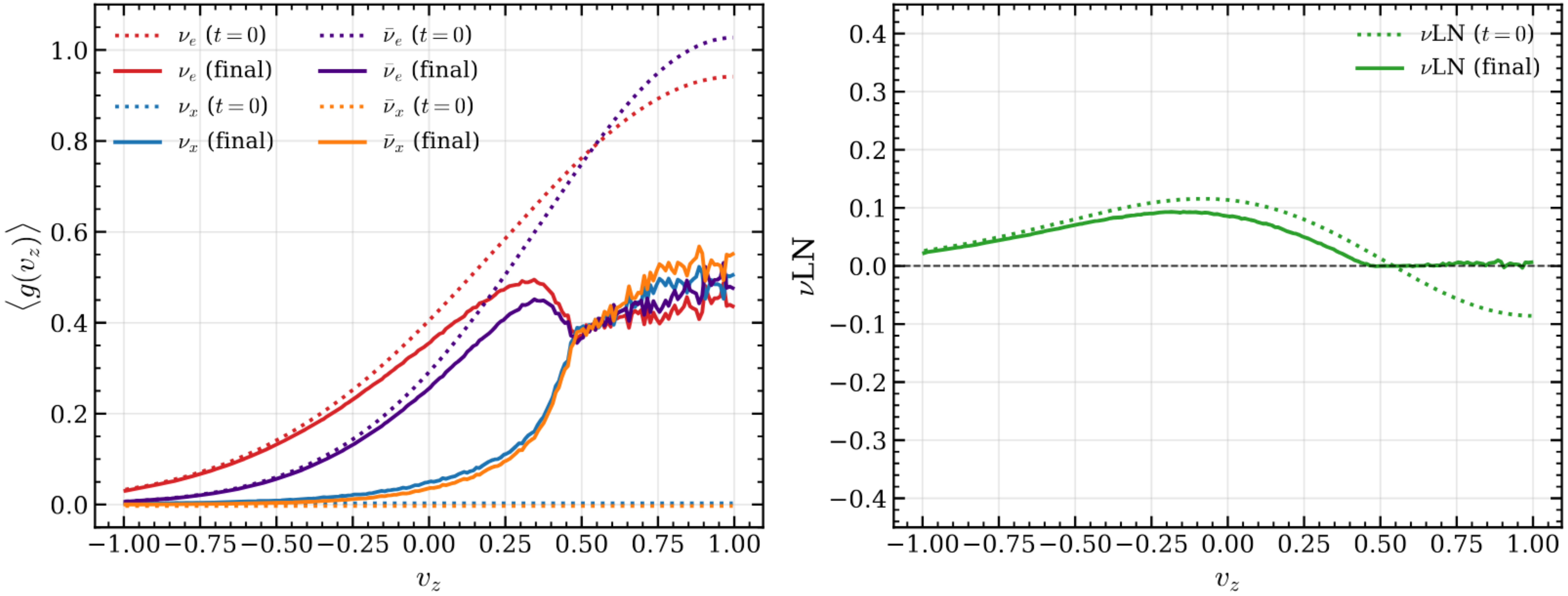
Logistic Regression : 94%

K-Nearest Neighbours : 96%

Support Vector Machines : 92%

Decision Tree : 94%

FLAVOR EQUILIBRATION



TEST MODELS

Apply “box-like” flavor equilibration scheme to the initial angular distributions

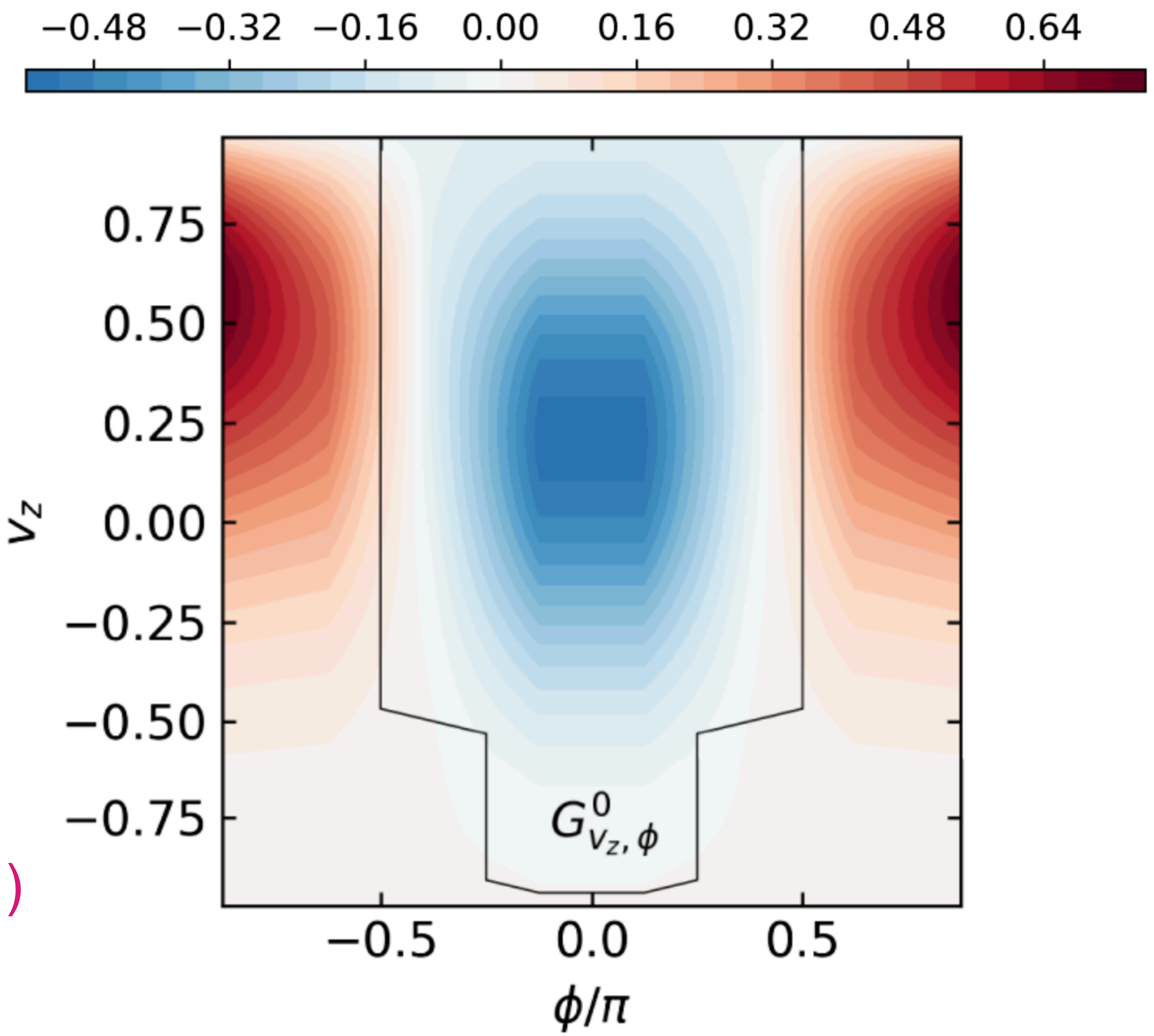
$$I_+ = \left| \int d\Gamma\, G(v_z,\phi)\, \Theta[G(v_z,\phi)] \right|,$$

$$I_- = \left| \int d\Gamma\, G(v_z,\phi)\, \Theta[-G(v_z,\phi)] \right|.$$

$$P_{ee}(\Gamma) = \begin{cases} \frac{1}{2}, & \text{for } \Gamma^< \\ 1 - \frac{I_<}{2 I_>}, & \text{for } \Gamma^> \end{cases}$$

$$I_< = \min(I_+, I_-)$$

$$I_> = \max(I_+, I_-)$$



S. Bhattacharyya, et. al. , Phys. Rev. Lett. 126, 061302 (2021)

M. Zaizen, et. al., , Phys. Rev. D 107, 103022 (2023)

M. George et. al. , *Phys.Rev.D* 110 (2024) 12, 123018

TEST MODELS

Apply "box-like" flavor equilibration scheme to the initial angular distributions

Accuracies :	
Logistic Regression : 55%	ν_x and $\bar{\nu}_x$ fluxes are different after FE
K-Nearest Neighbours : 54%	only no crossing samples
Support Vector Machines : 53%	
Decision Tree : 54%	

Accuracies :	
Logistic Regression : 94%	ν_x and $\bar{\nu}_x$ fluxes are same after FE
K-Nearest Neighbours : 98%	both crossings and no crossings
Support Vector Machines : 99%	
Decision Tree : 98%	

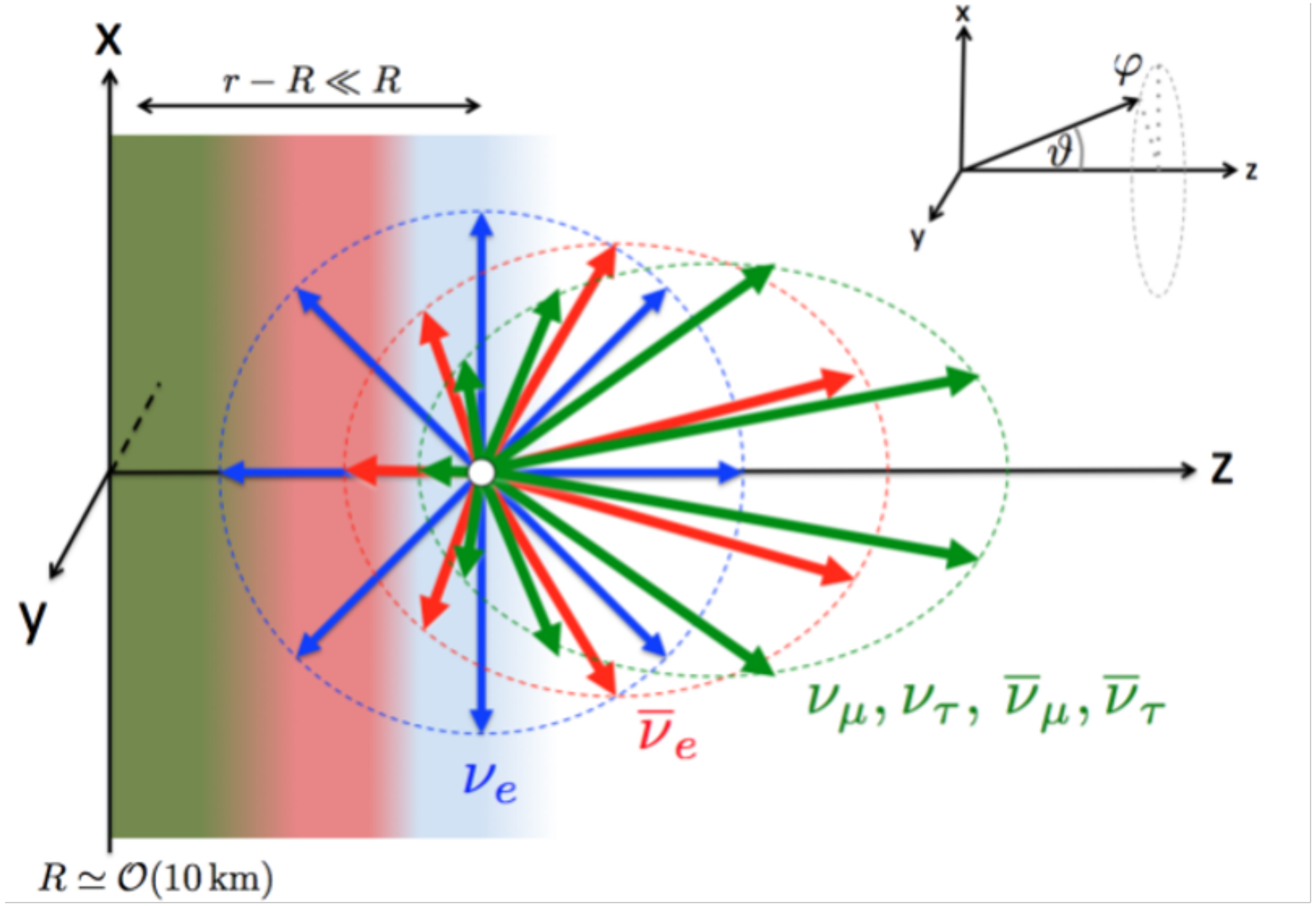
$$G(\mathbf{v}) = \sqrt{2}\, G_F \int_0^\infty \frac{E_\nu^2\, dE_\nu}{(2\pi)^3} [f_{\nu_e}(\mathbf{p}) - f_{\bar{\nu}_e}(\mathbf{p}) - f_{\nu_x}(\mathbf{p}) + f_{\bar{\nu}_x}(\mathbf{p})]$$

CONCLUSIONS

- Train and test our models on different setups
- Predictions are affected by the dataset-specific features
- Within a given setup, the ML model robustly captures the essential structure of ELN crossings
- Our ML model is able to efficiently detect the crossings/no crossings when the input features in the test set are consistent with that of training set

THANK YOU FOR YOUR ATTENTION

BACK UP



Different flavours :

- different interaction rates
- different angular distributions

ML algorithms

Logistic Regression :

Aim : to know yes/no

Probabilistic interpretation

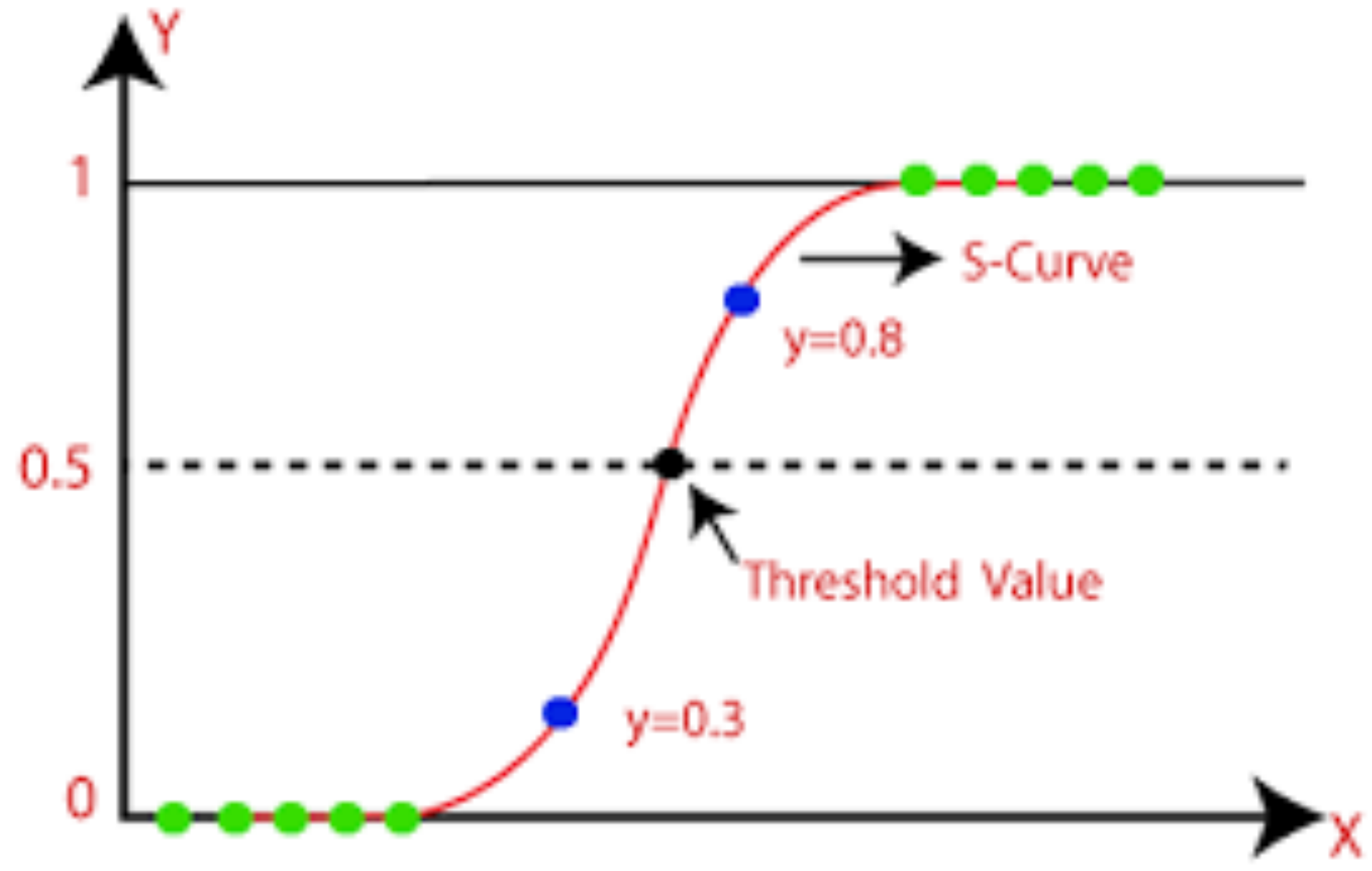
$$\sigma(z) = \frac{1}{1 + \exp(-z)}$$

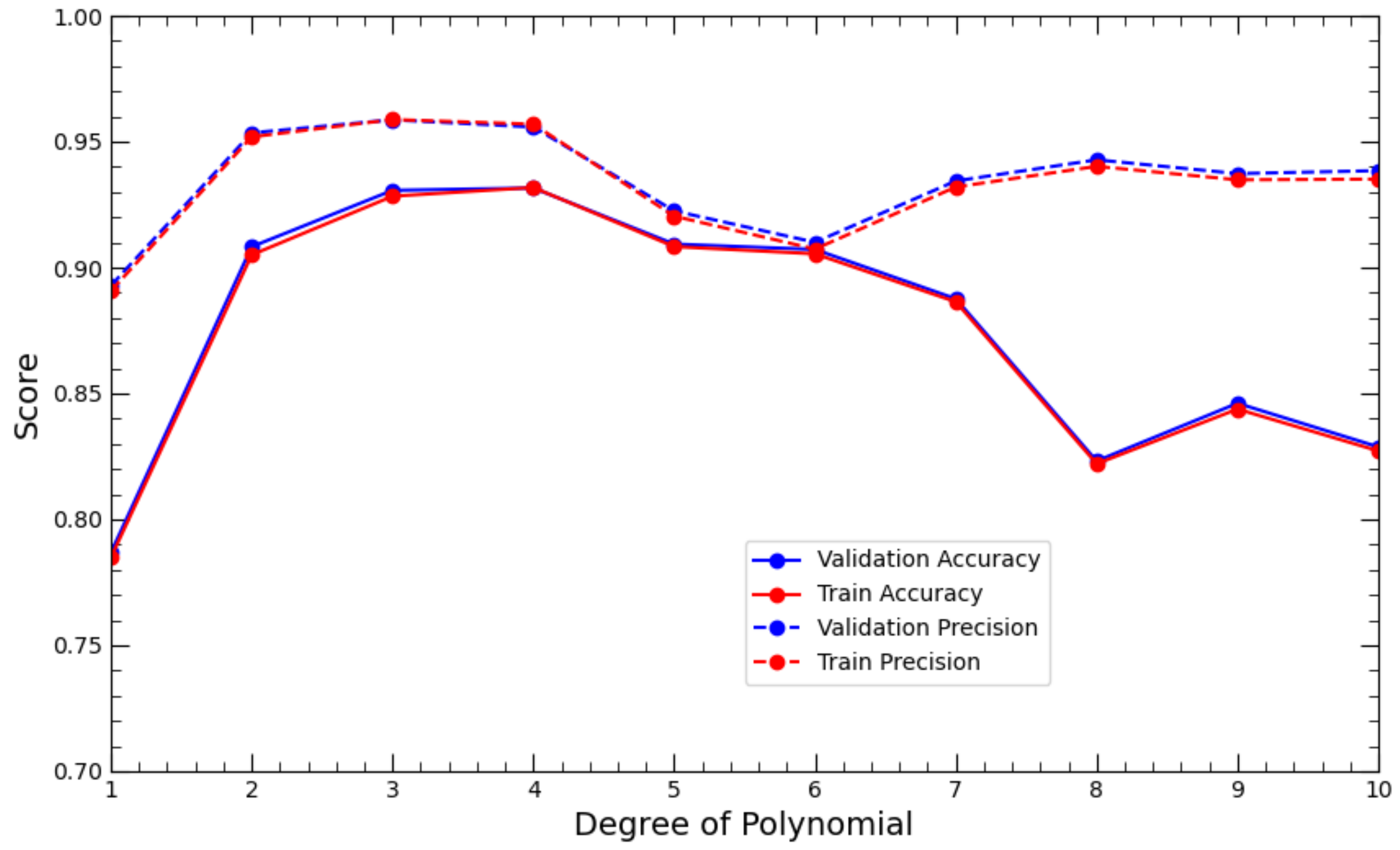
$$z = \mathbf{w} \cdot \mathbf{x}$$

Threshold value = 0.5

If $\sigma(z) > 0.5$, class 1

If $\sigma(z) < 0.5$, class 0





K- nearest neighbours

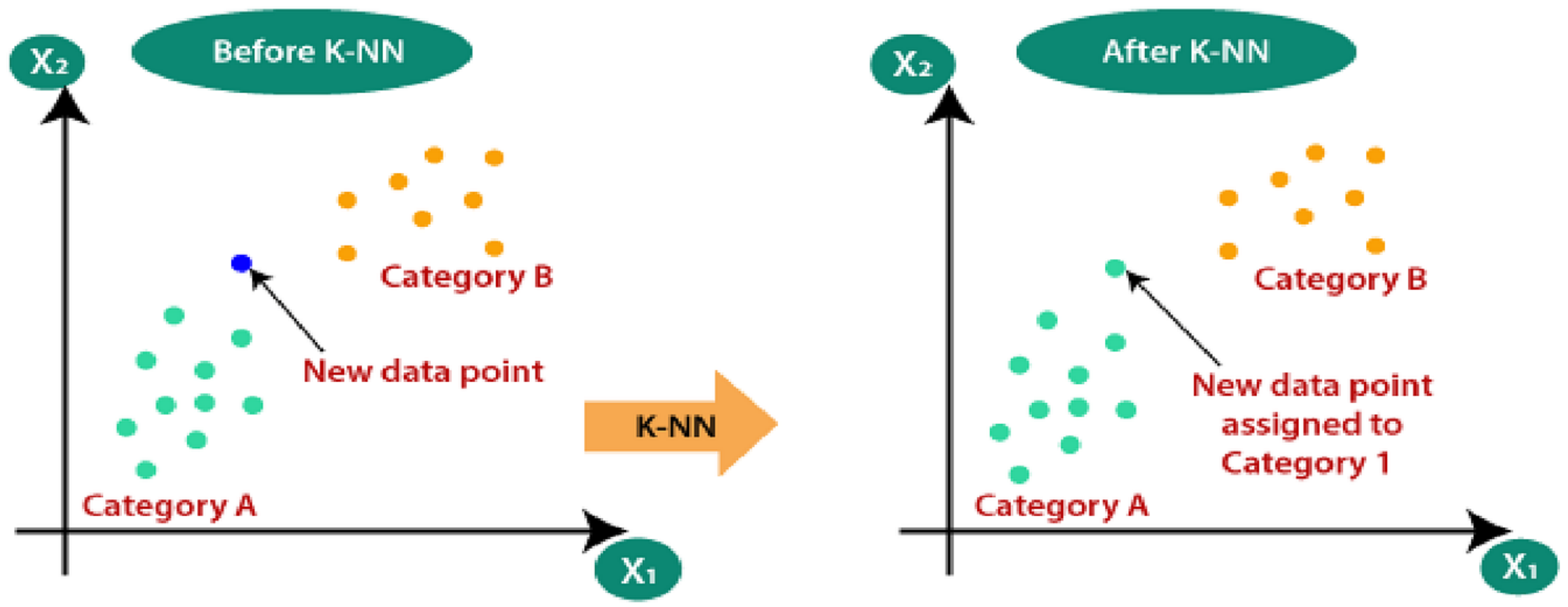
Aim : how many neighbours are there

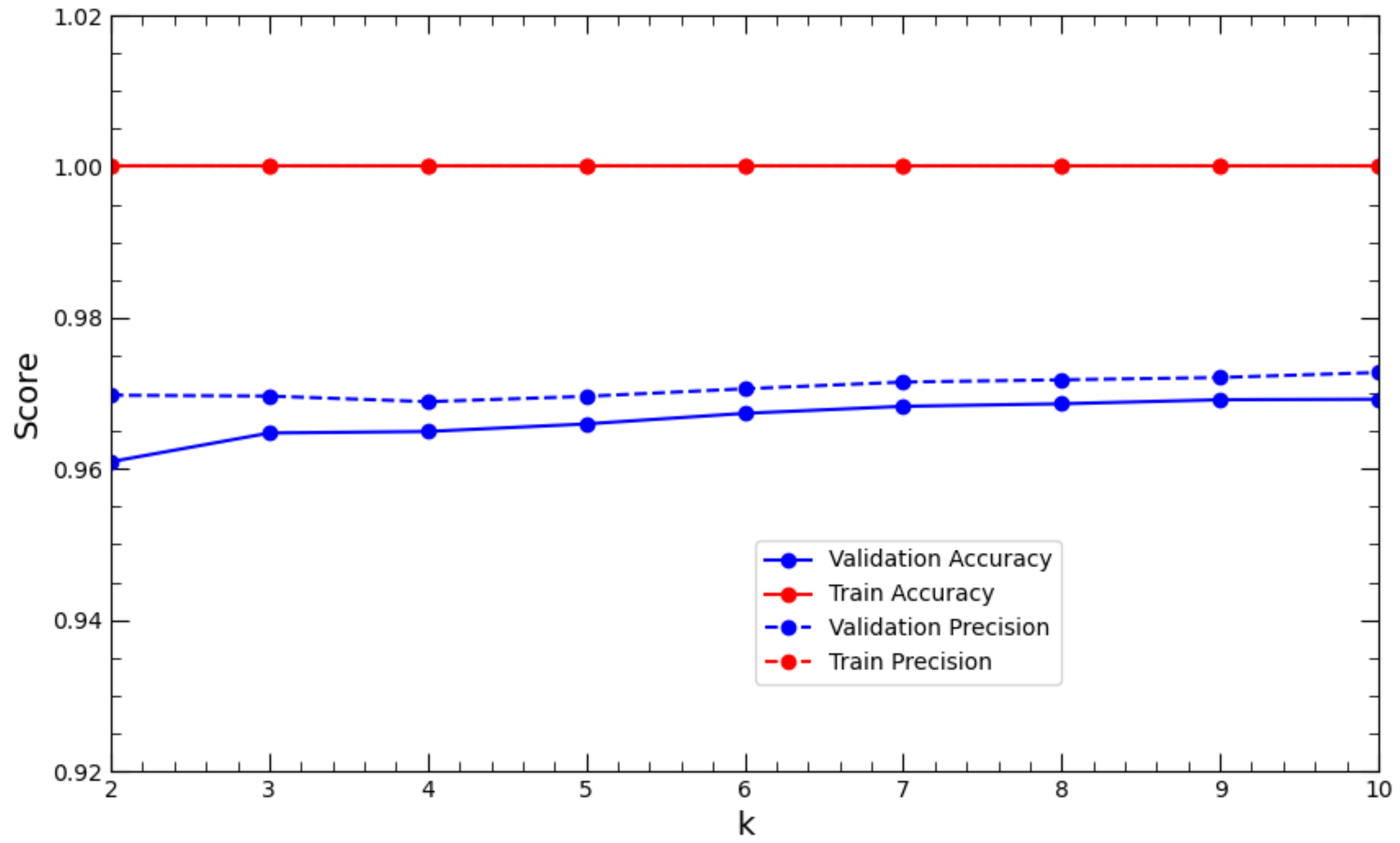
of each class

Based on the number of neighbours,

the new data point is classified

k - number of nearest neighbours

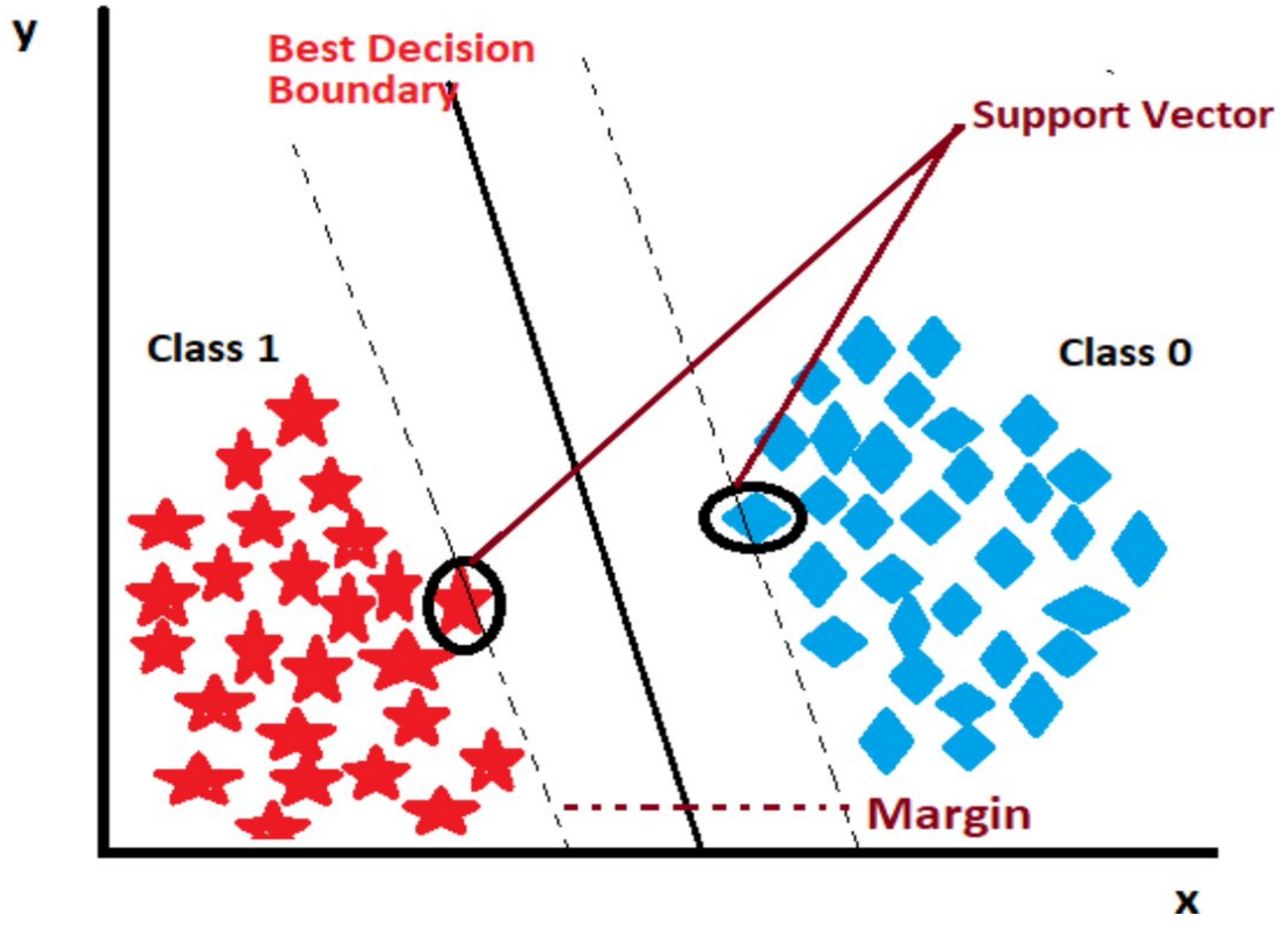




Support Vector Machine

Aim : To find the best decision boundary

separating the different classes

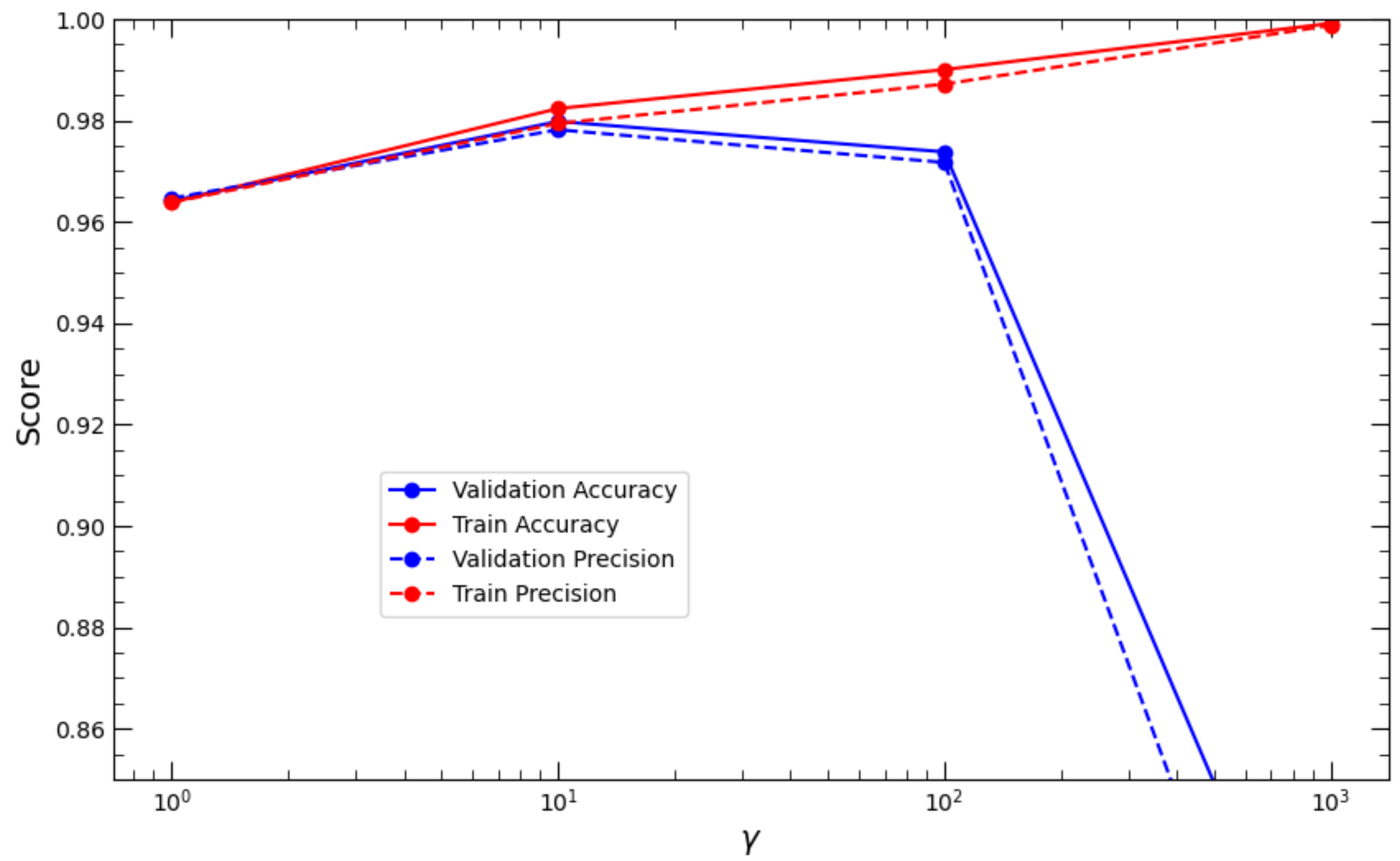


Distance between data point and boundary

is maximised

Radial basis function :

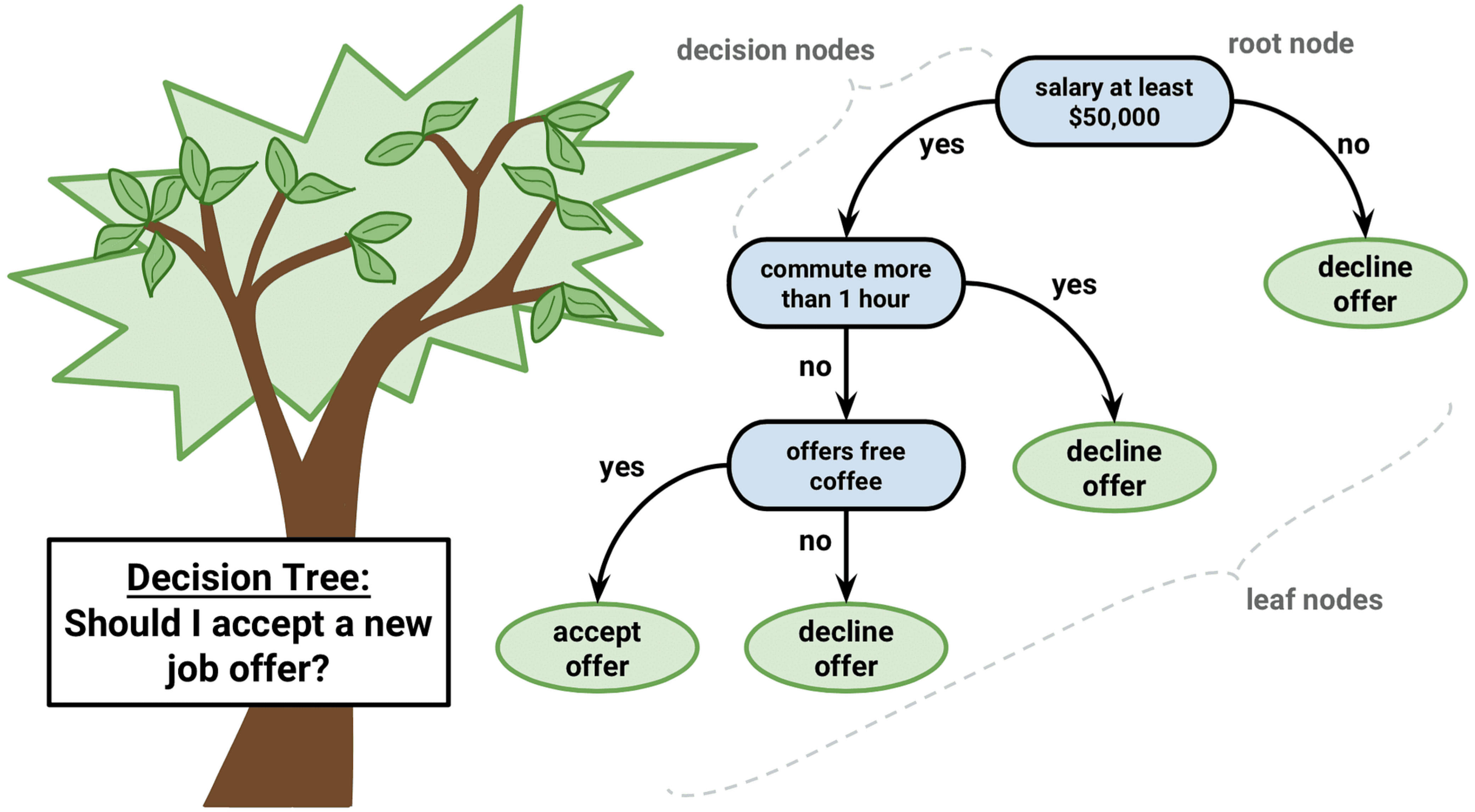
$$\mathscr{K}(\vec{x}, \vec{x}') = \exp(-\gamma ||\vec{x} - \vec{x}'||^2)$$



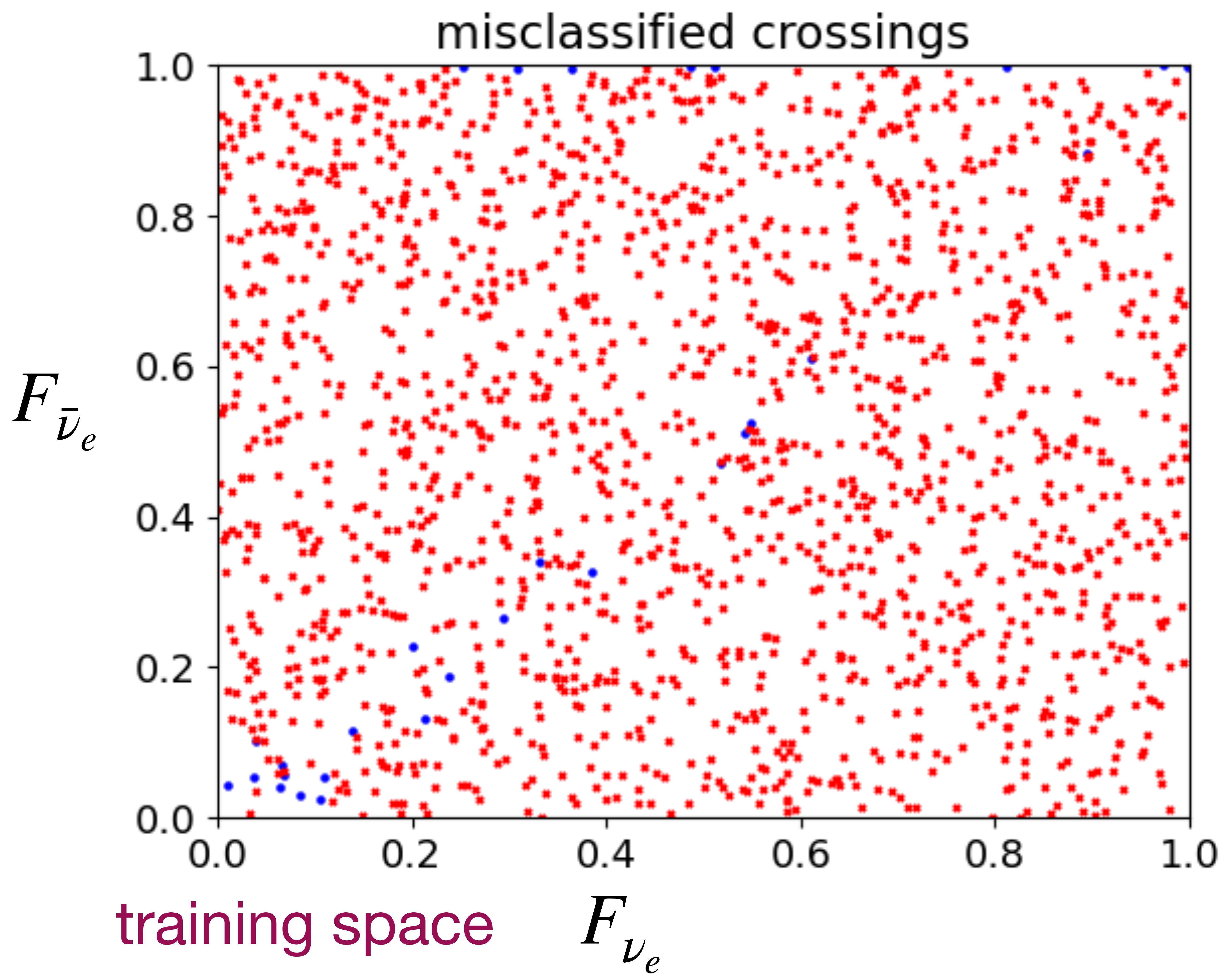
Decision Tree Classifier

At each node, feature is selected which best separates the two classes

Continues until the data is sufficiently classified



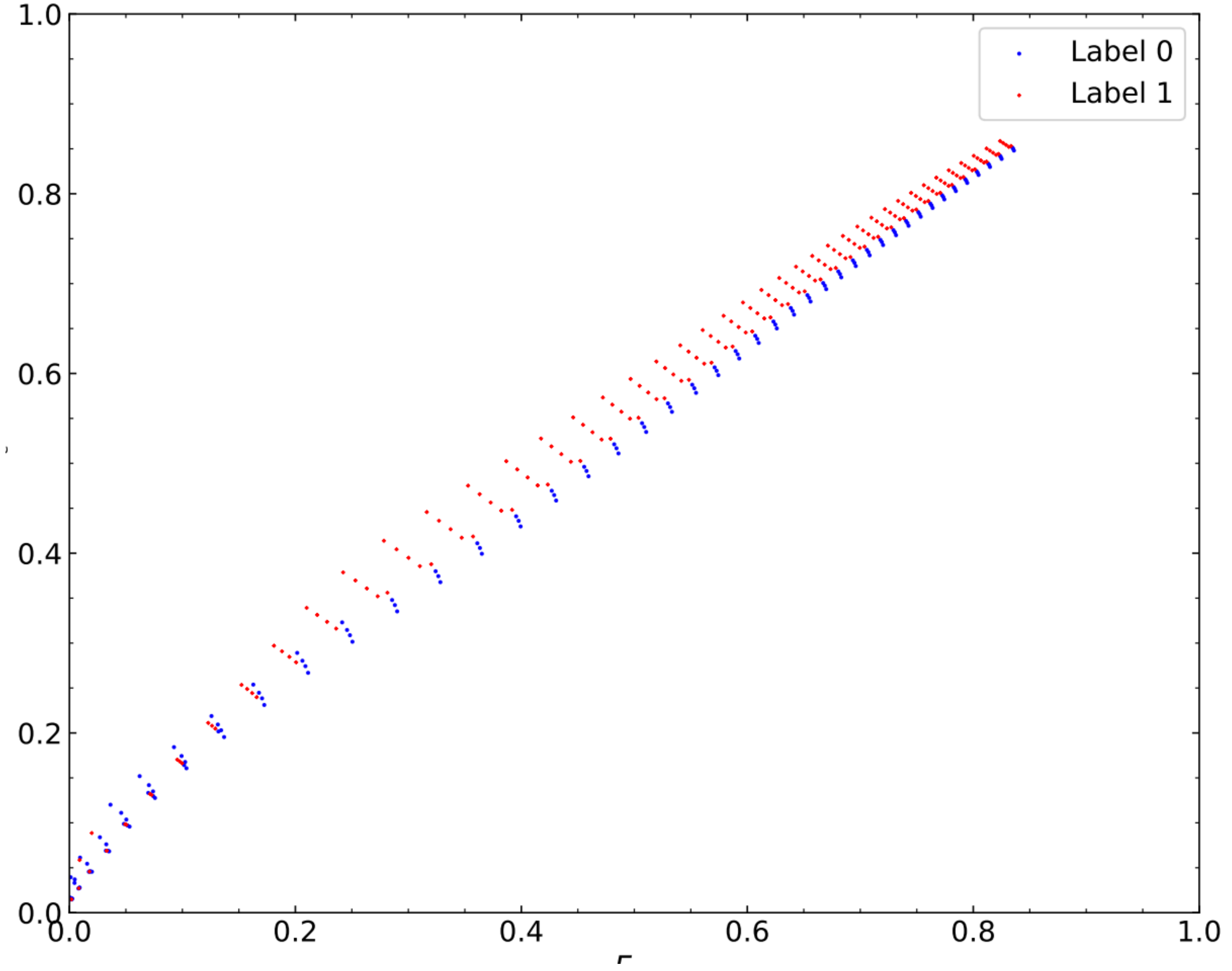
TEST MODELS



training space

• Crossing • No crossing

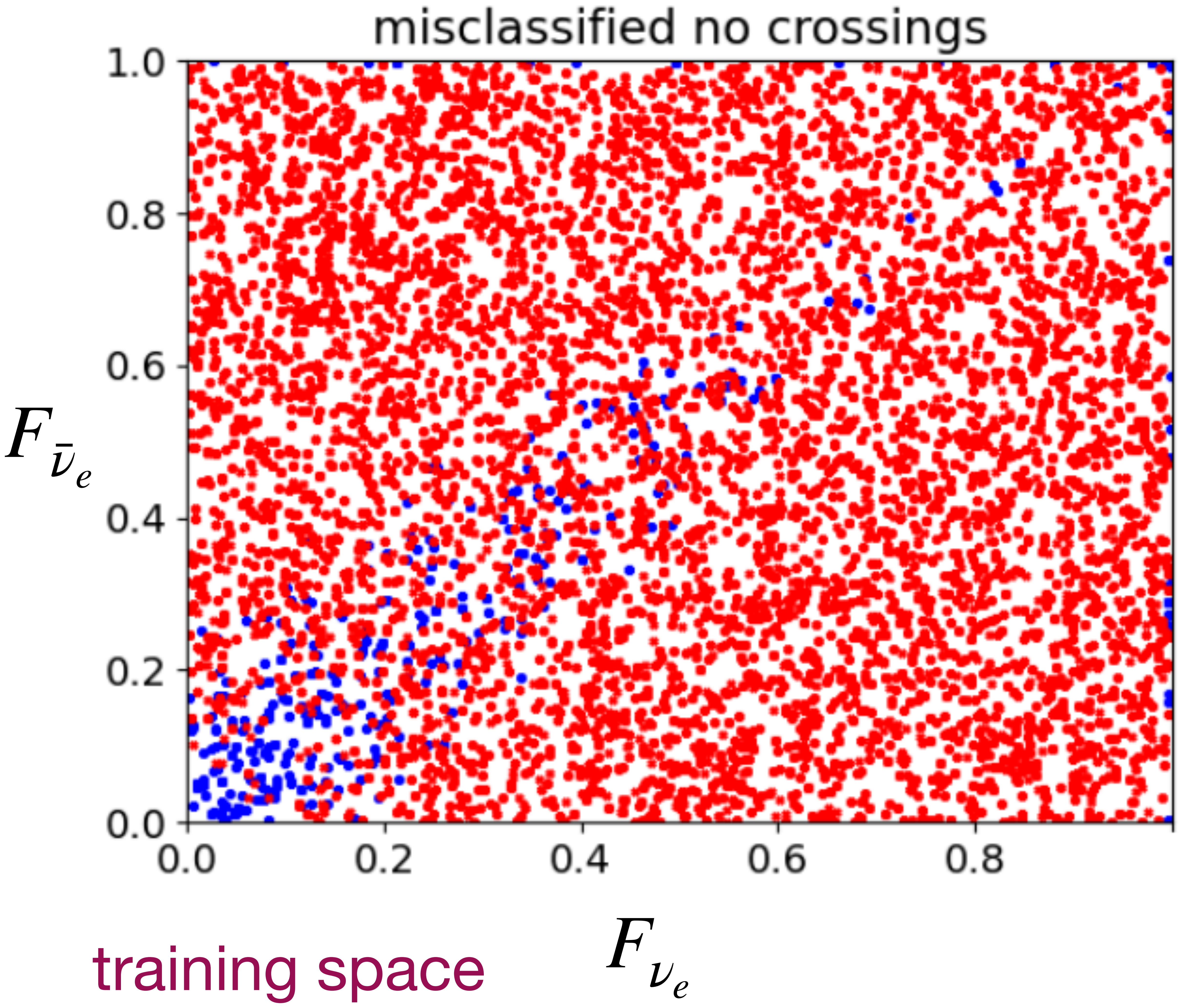
Axisymmetric



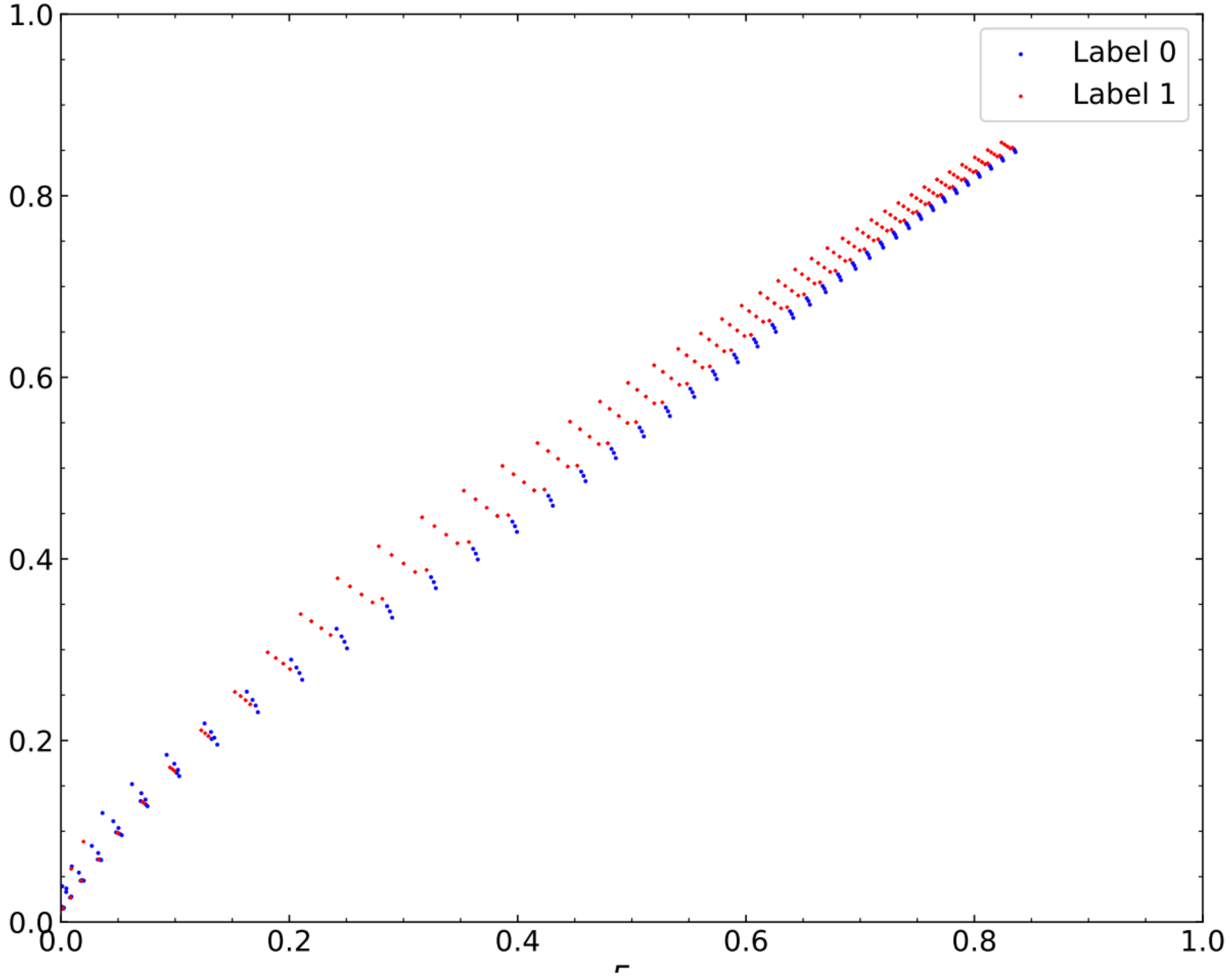
test space

TEST MODELS

Axisymmetric



training space

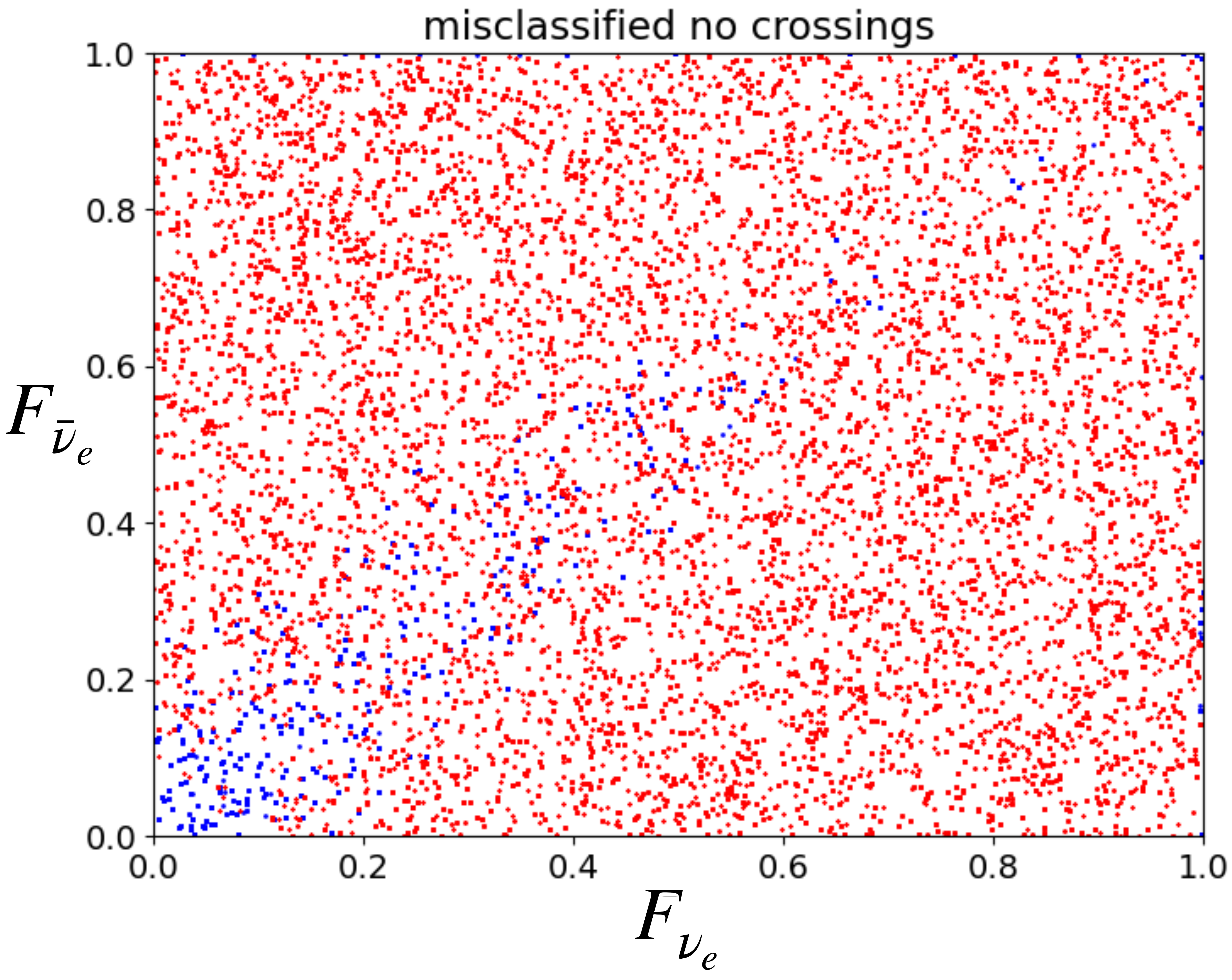


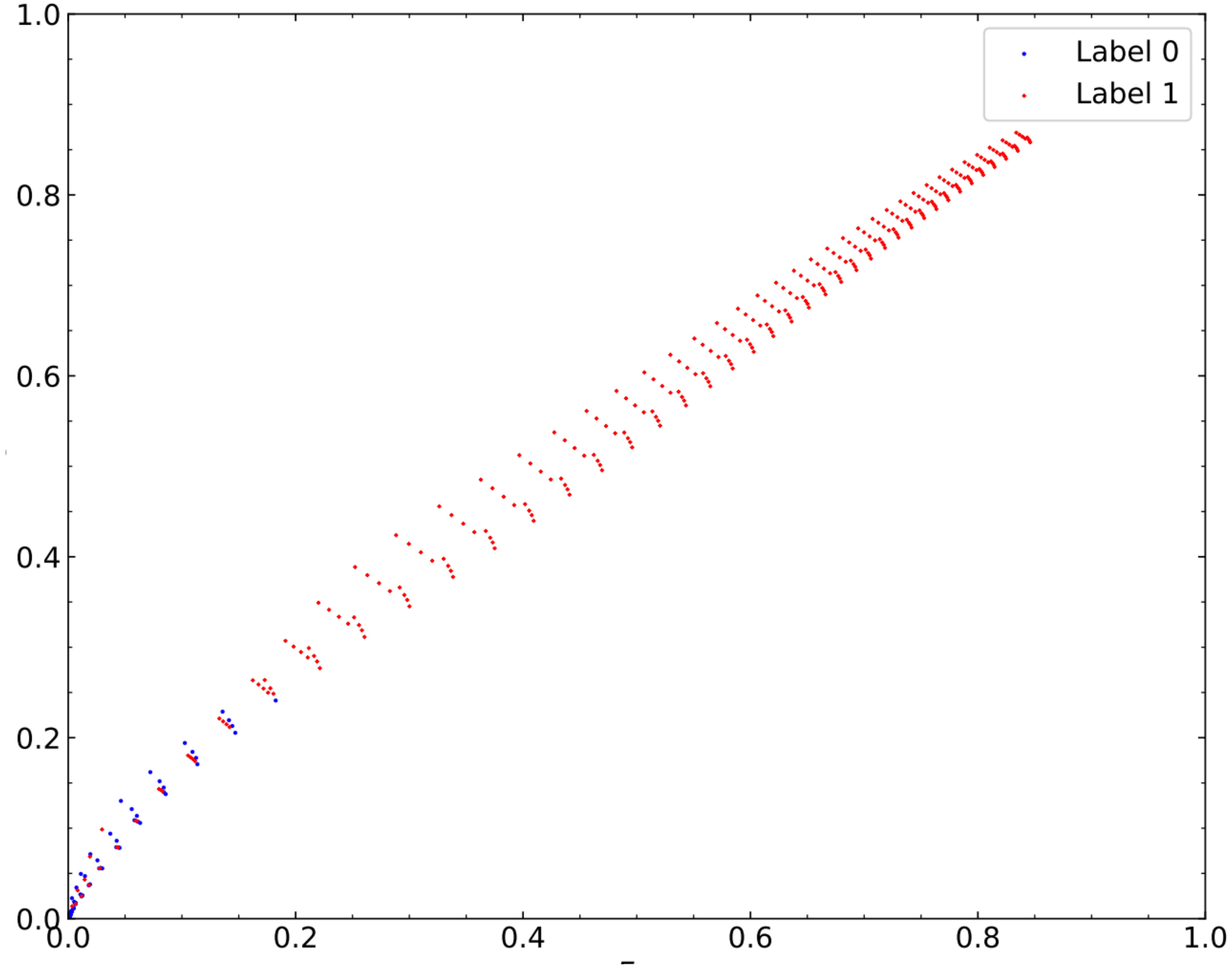
test space

- Crossing
- No crossing

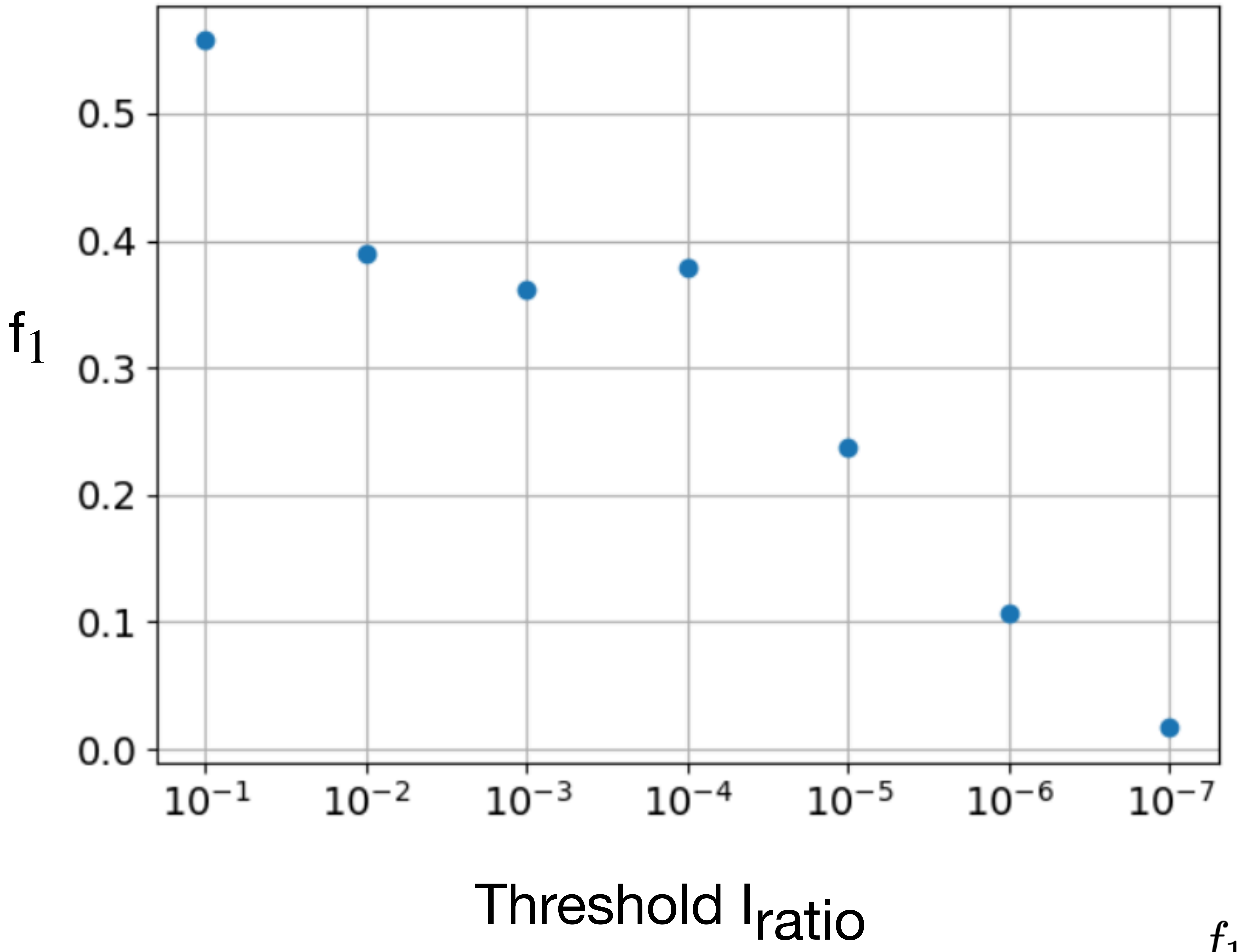
TEST MODELS

Non - axisymmetric ($\theta_r = \pi/6$)





- Crossing
- No crossing



- Axisymmetric setup
- Apply flavor equilibration
- Remap the flavor equilibrated fluxes to the max. entropy distribution
- Search for ELN crossings

$$I_{ratio} = \frac{I_<}{I_>}$$

$$f_1 = \frac{\text{number of crossings after remapping}}{\text{number of original crossings}}$$