Loss is more

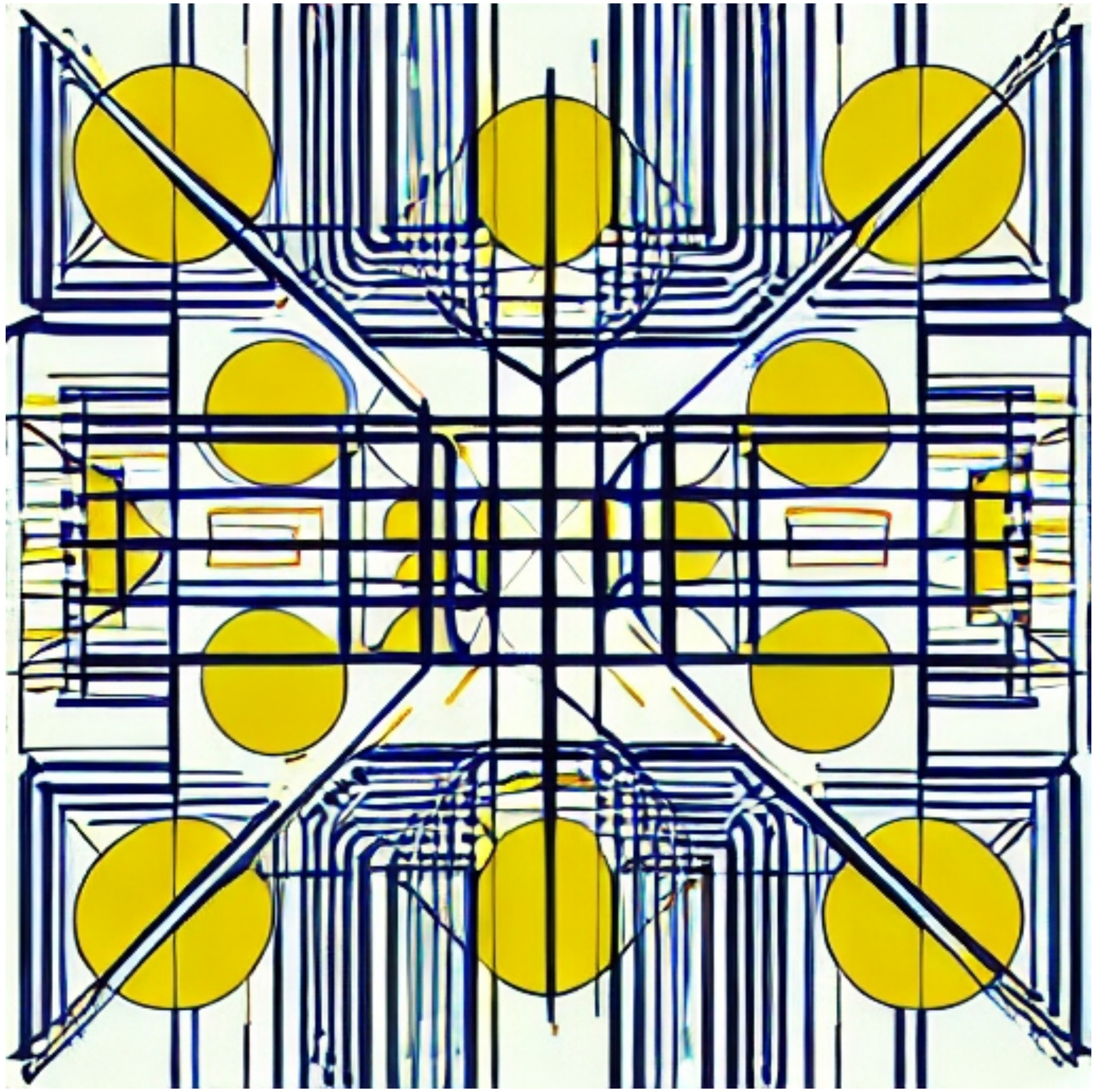
Exploring the weight space of a perceptron via enhanced sampling techniques

UNIVERSITÀ DI TRENTO

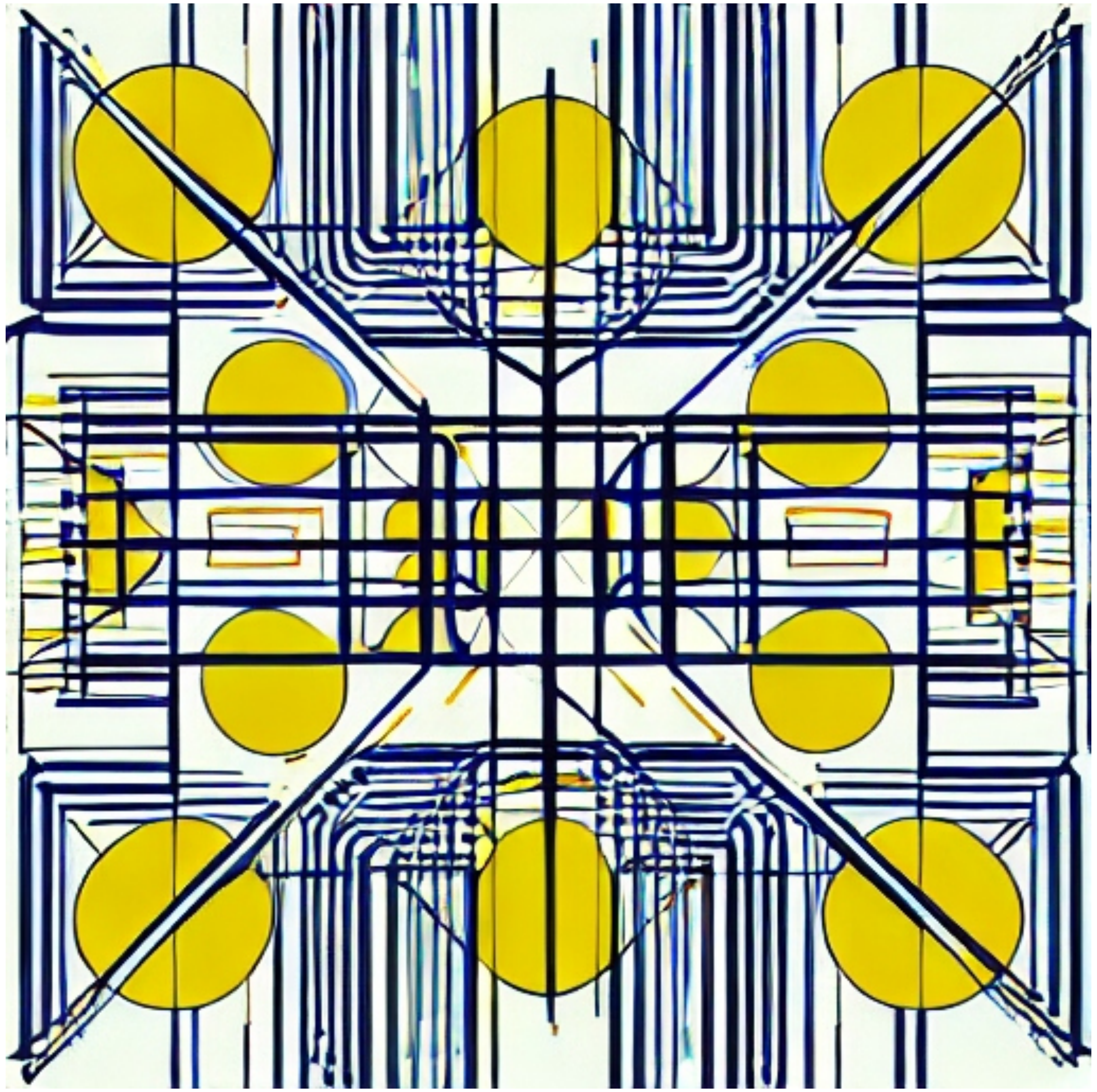
ECT* - BRIDGING SCALES 2024

margherita.mele@unitn.it - Margherita Mele

HOW DO NETWORKS LEARN?

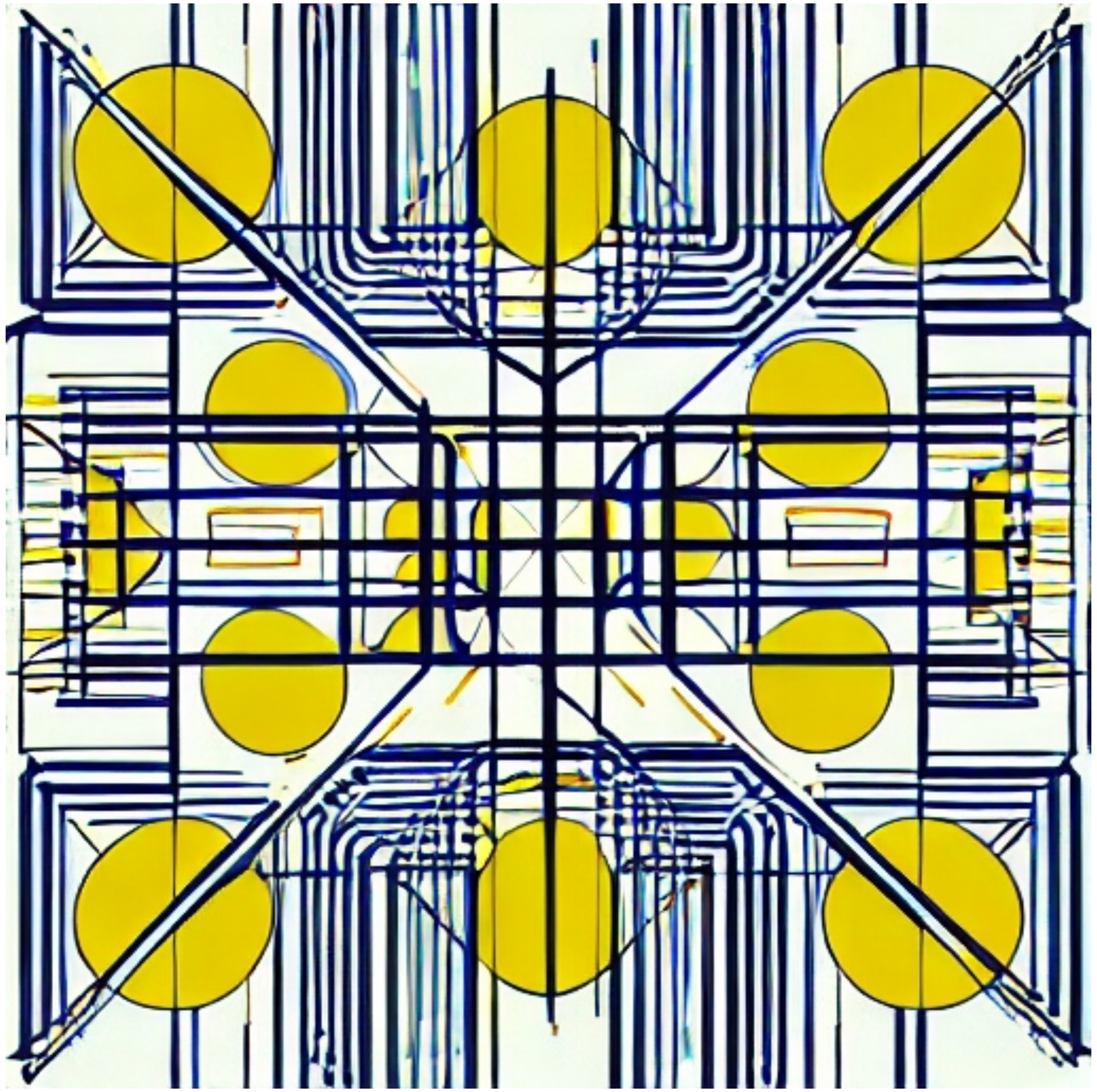


HOW DO NETWORKS LEARN?



Neural Networks identify pattern that we are not able to see

HOW DO NETWORKS LEARN?

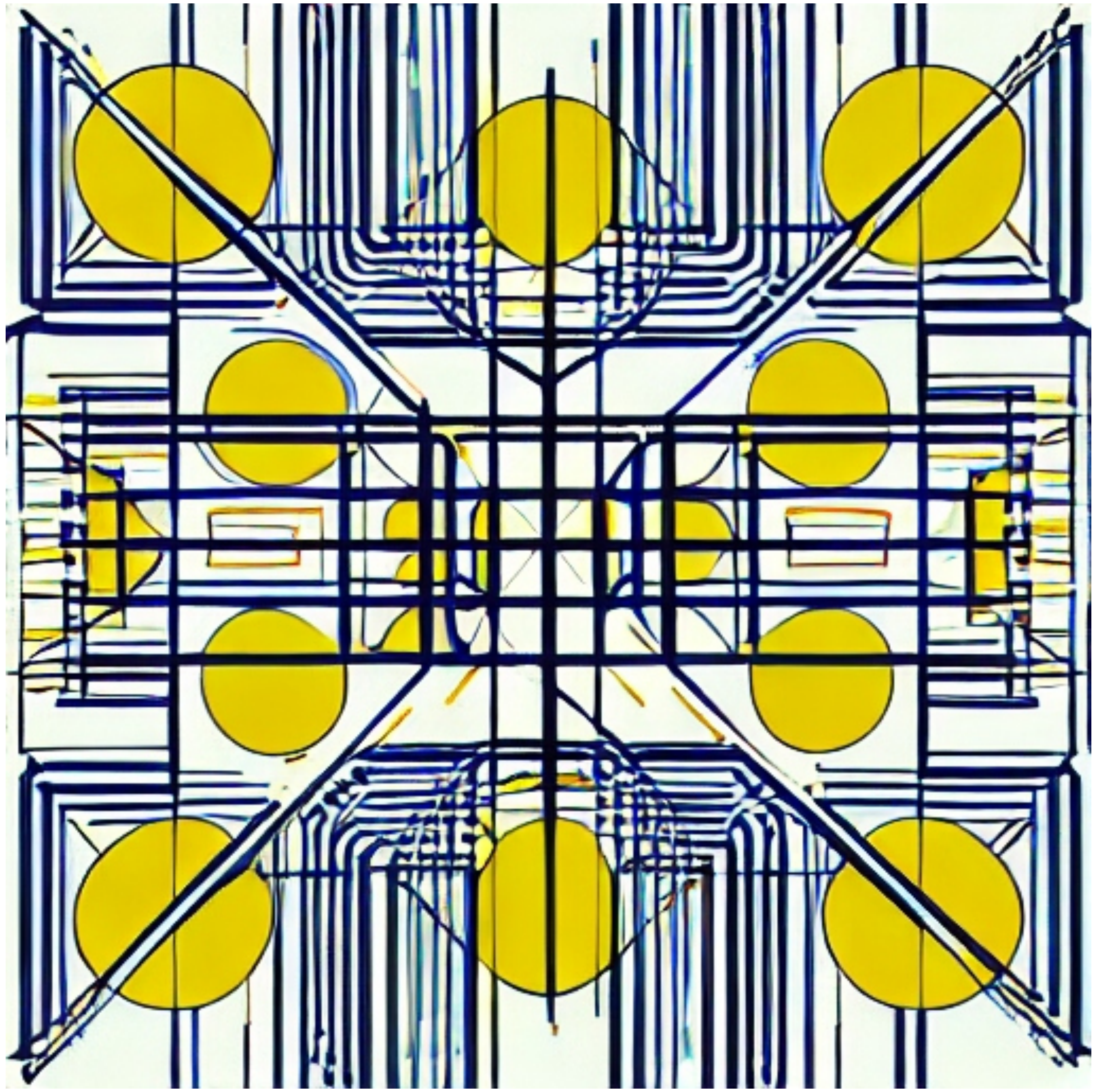


Neural Networks identify pattern
that we are not able to see

↓

Can we identify the key data traits
networks learn from?

HOW DO NETWORKS LEARN?

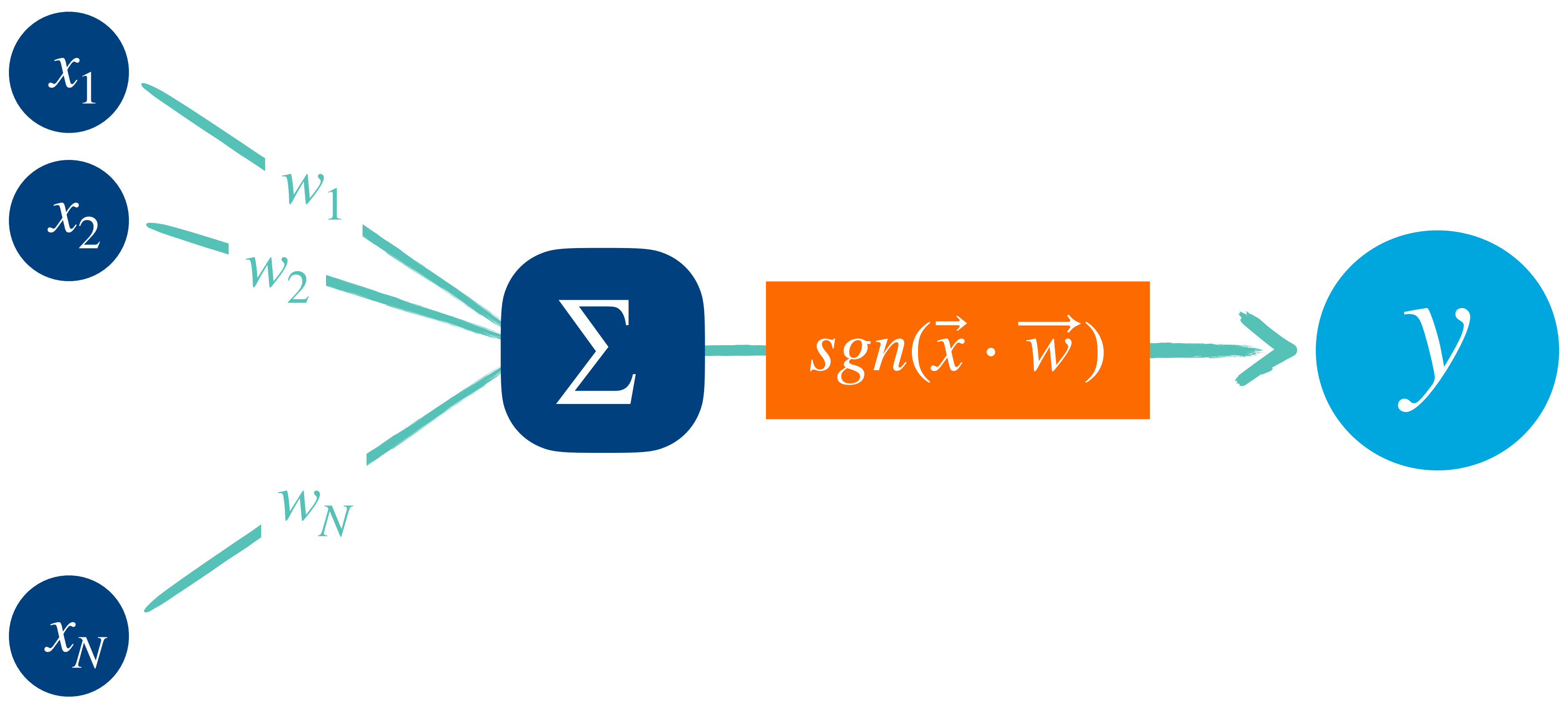


Neural Networks identify pattern that we are not able to see

Can we identify the key data traits networks learn from?

Describe the network configuration space while varying the input data structure

Perceptron



Inputs $\{x_i\}_1^P$

$$[1,1,-1,\ldots,1,-1,-1]$$
$$[-1,1,1,\ldots,-1,1,-1]$$
$$\vdots$$
$$[-1,-1,1,\ldots,-1,1,1]$$

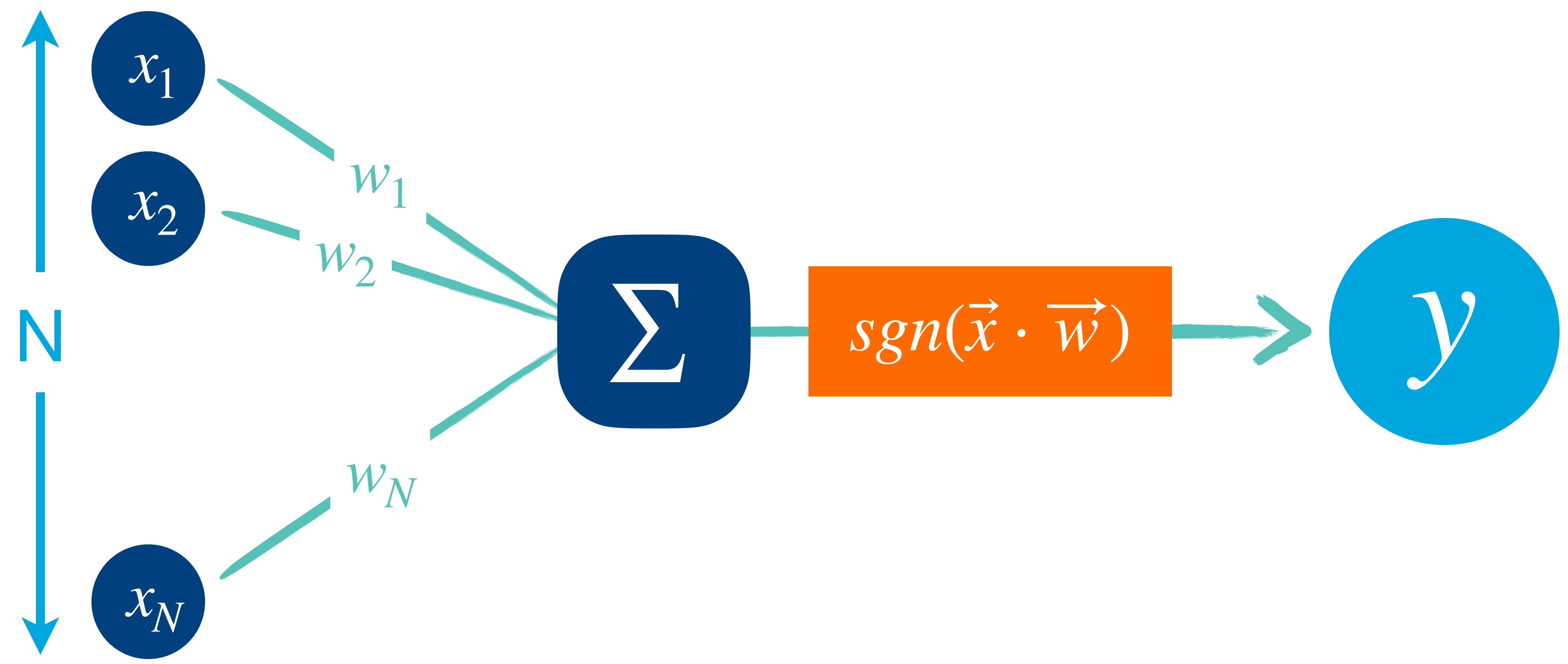
Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \text{sgn}\left(w\,x_i\right)$$

Perceptron



Inputs $\{x_i\}_1^P$

$$\begin{gathered}[1,1,-1,\ldots,1,-1,-1] \\ [-1,1,1,\ \ldots,-1,1,-1] \\ \vdots \\ [-1,-1,1,\ \ldots,-1,1,1]\end{gathered}$$

N

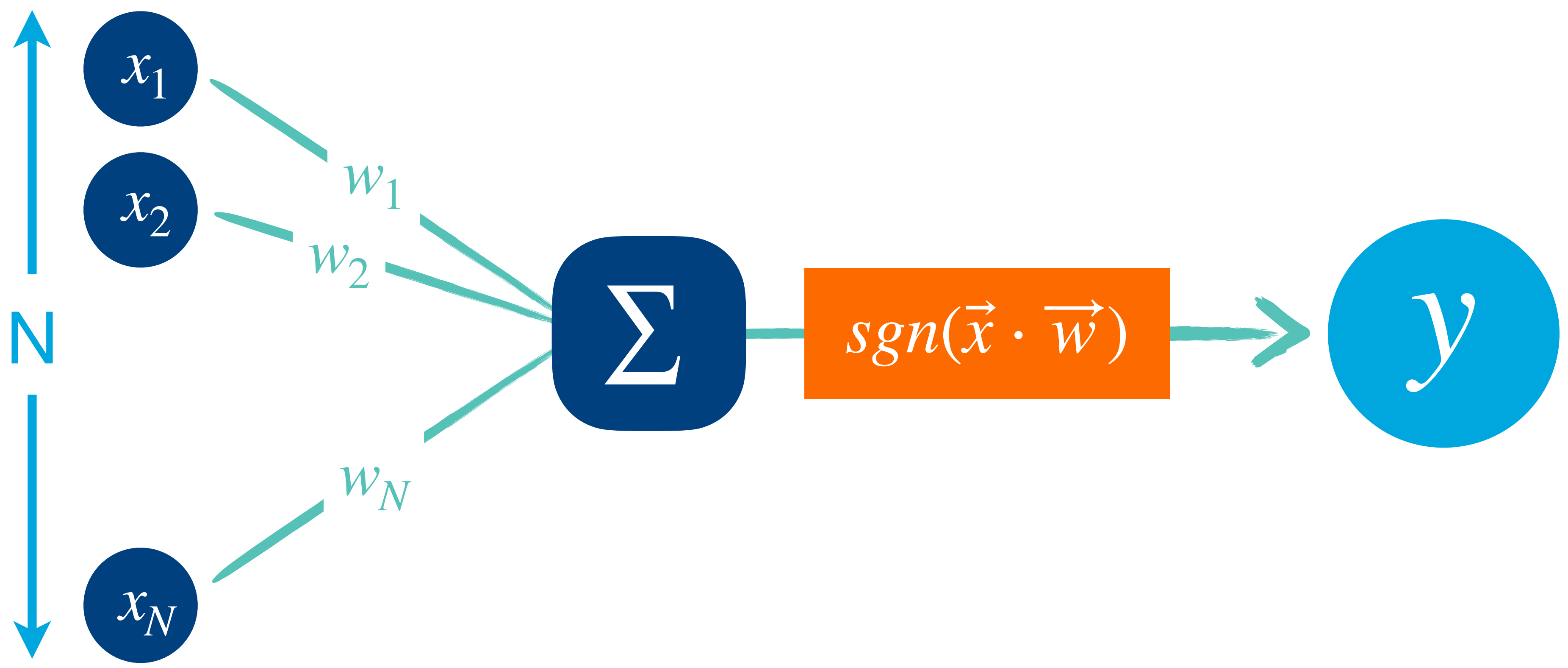
Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \text{sgn}\left(w\, x_i\right)$$

Perceptron



Inputs $\{x_i\}_1^P$

$$
\begin{array}{c}
[1,1,-1,\ldots,1,-1,-1] \\
[-1,1,1,\ \ldots,-1,1,-1] \\
\vdots \\
[-1,-1,1,\ \ldots,-1,1,1]
\end{array}
$$

P

N

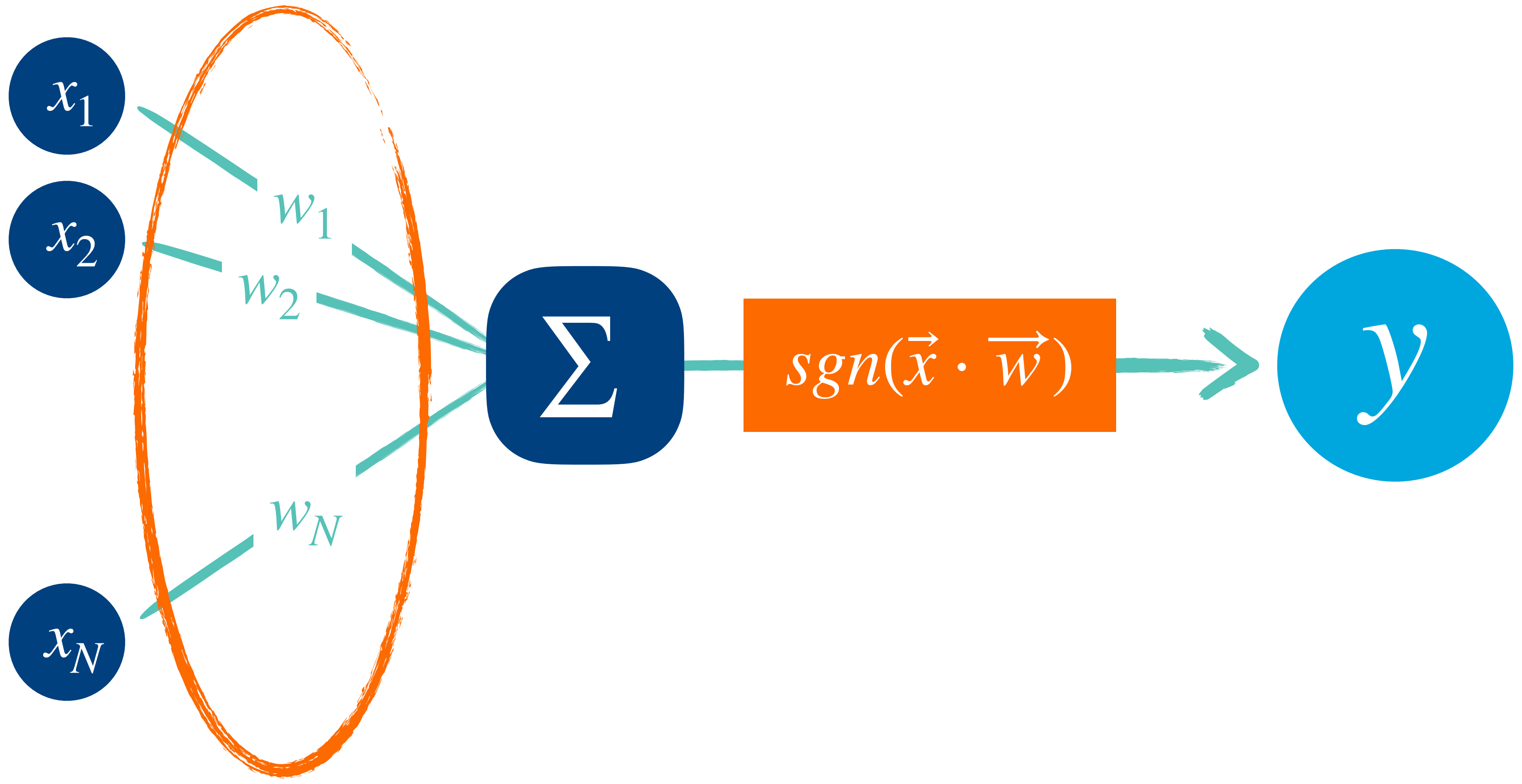
Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\left(w\, x_i\right)$$

Perceptron



Inputs $\{x_i\}_1^P$

$$\begin{array}{c} [1,1,-1,\ldots,1,-1,-1] \\ [-1,1,1,\ \ldots,-1,1,-1] \\ \vdots \\ [-1,-1,1,\ \ldots,-1,1,1] \end{array}$$

P

N

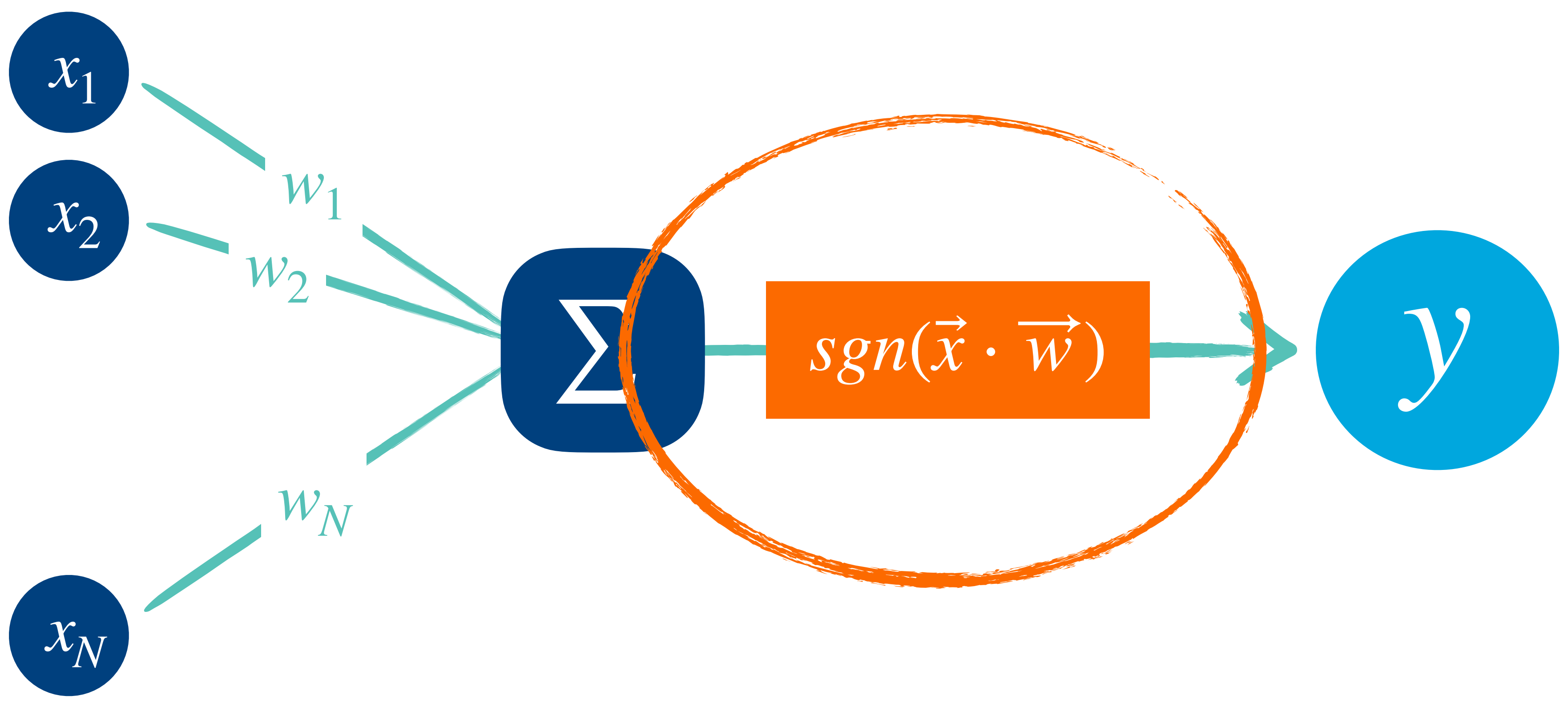
Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\left(w\, x_i\right)$$

Perceptron



Inputs $\{x_i\}_1^P$

$$[1,1,-1,\ldots,1,-1,-1]$$
$$[-1,1,1,\ \ldots,-1,1,-1]$$
$$\vdots$$
$$[-1,-1,1,\ \ldots,-1,1,1]$$

P

N

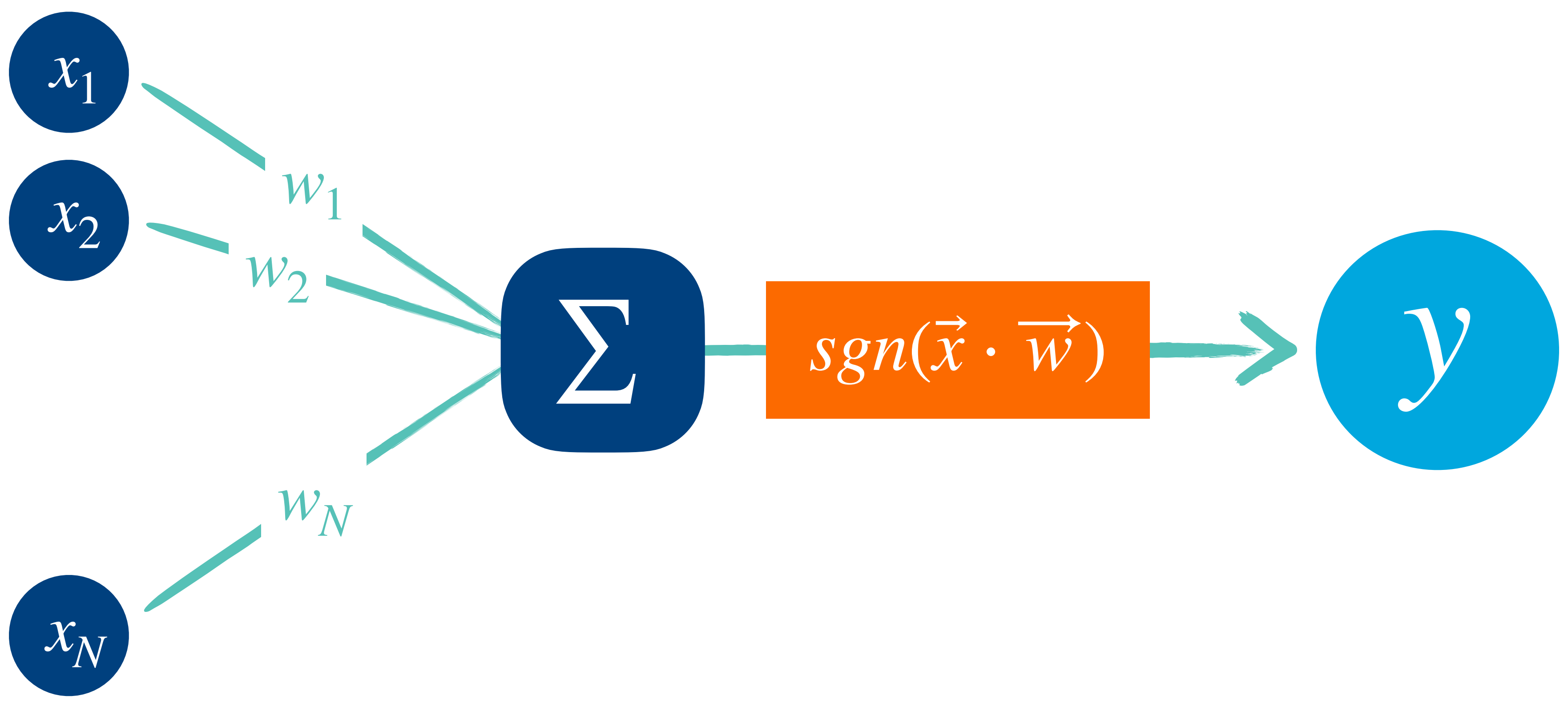
Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \text{sgn}\left(w\, x_i\right)$$

Perceptron



Energy

$$E_w = \sum_i^P \Theta\left(-y_i \cdot \mathrm{sgn}\left(w\, x_i\right)\right)$$

Entropy

$$S(\bar{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \bar{E})\right)$$

Inputs $\{x_i\}_1^P$

$$\begin{array}{cc} & \begin{bmatrix}1,1,-1,\ldots,1,-1,-1\end{bmatrix} \\ P & \begin{bmatrix}-1,1,1,\ \ldots,-1,1,-1\end{bmatrix} \\ & \vdots \\ & \begin{bmatrix}-1,-1,1,\ \ldots,-1,1,1\end{bmatrix} \\ & N \end{array}$$

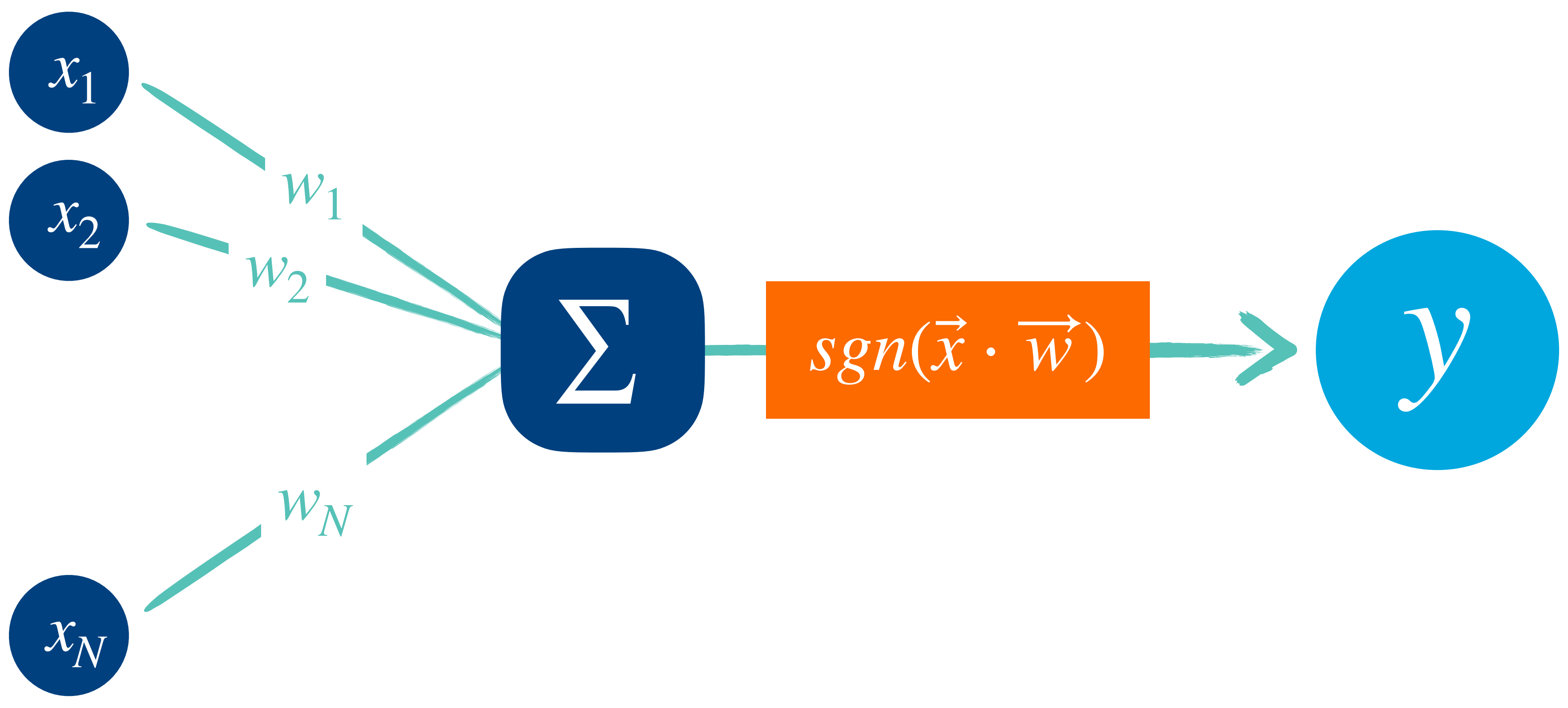
Weight vector w

$$\begin{bmatrix}-1,1,-1\ \ldots,-1,-1,1\end{bmatrix}$$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\left(w\, x_i\right)$$

Perceptron



Inputs $\{x_i\}_1^P$

$[1,1,-1,\dots,1,-1,-1]$
$[-1,1,1,\dots,-1,1,-1]$
$\vdots$
$[-1,-1,1,\dots,-1,1,1]$

P

N

Weight vector w

$[-1,1,-1 \dots,-1,-1,1]$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\left(w\, x_i\right)$$

Energy

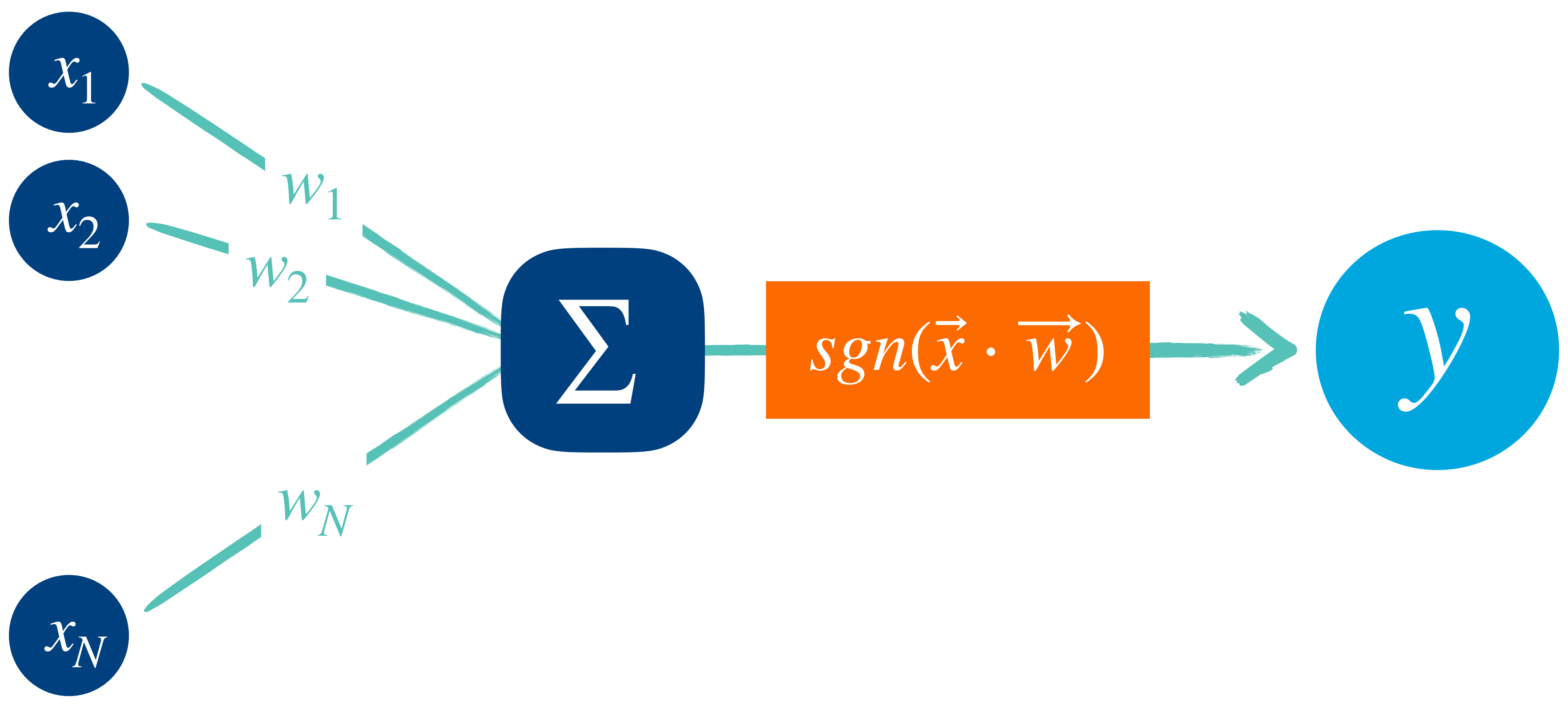
$$E_w = \sum_i^P \Theta\left(-y_i \cdot \mathrm{sgn}\left(w\, x_i\right)\right)$$

Number of Errors

Entropy

$$S(\overline{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \overline{E})\right)$$

Perceptron



Energy

$$E_w = \sum_i^P \Theta\left(-y_i \cdot \mathrm{sgn}\left(w\, x_i\right)\right)$$

Number of Errors

Entropy

$$\boxed{S(\overline{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \overline{E})\right)}$$

Inputs $\{x_i\}_1^P$

$$P \left\{ \begin{array}{l} [1,1,-1,\ldots,1,-1,-1] \\ [-1,1,1,\ \ldots,-1,1,-1] \\ \vdots \\ [-1,-1,1,\ \ldots,-1,1,1] \end{array} \right.$$

N

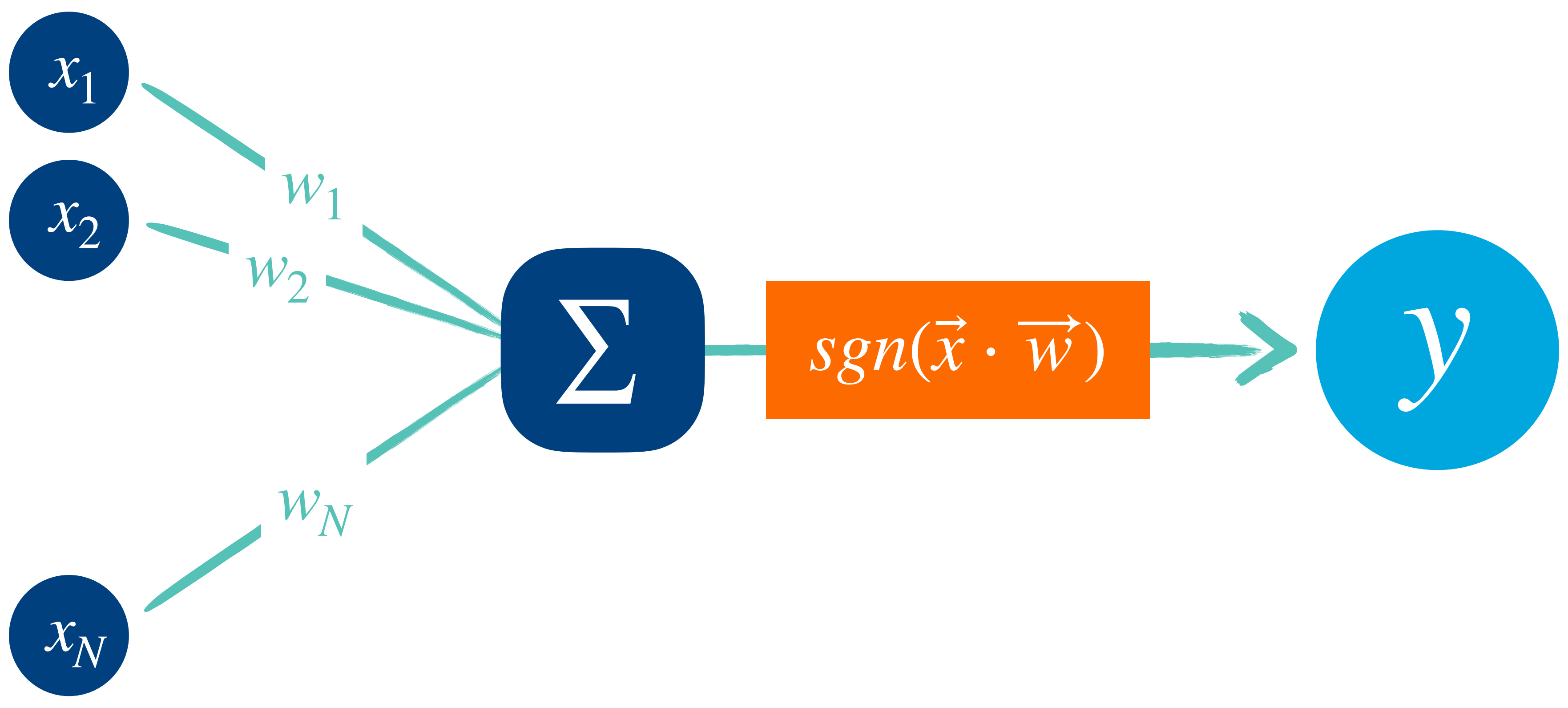
Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\left(w\, x_i\right)$$

Perceptron



Energy

$$E_w = \sum_i^P \Theta\left(-y_i \cdot \mathrm{sgn}\left(w\, x_i\right)\right)$$

Number of Errors

Entropy

$$S(\bar{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \bar{E})\right)$$

N small (< 25)

Inputs $\{x_i\}_1^P$

$[1,1,-1,\ldots,1,-1,-1]$
$[-1,1,1,\ \ldots,-1,1,-1]$
$\vdots$
$[-1,-1,1,\ \ldots,-1,1,1]$

P

N

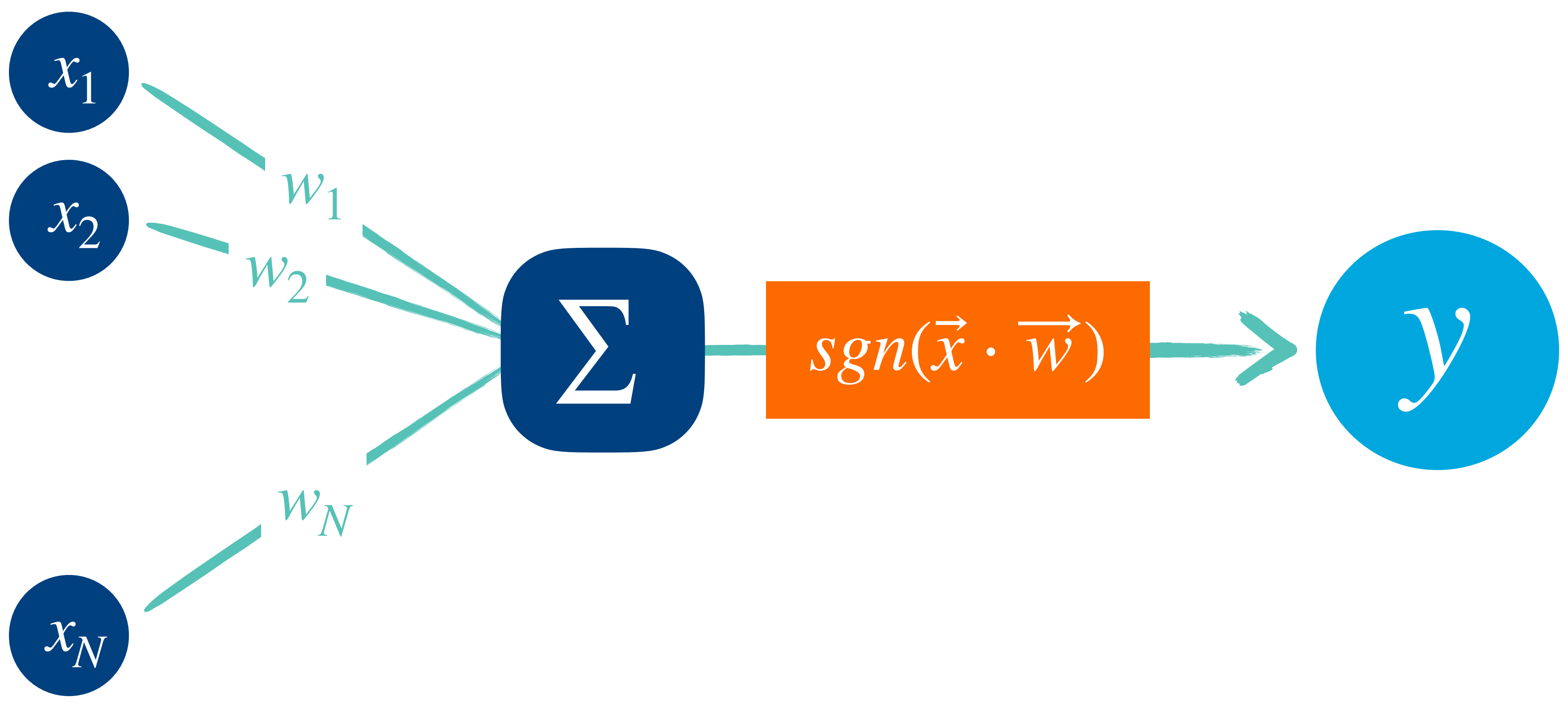
Weight vector w

$[-1,1,-1\ \ldots,-1,-1,1]$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\left(w\, x_i\right)$$

Perceptron



Energy

$$E_w = \sum_i^P \Theta\left(-y_i \cdot \text{sgn}\left(w\, x_i\right)\right)$$

Number of Errors

Entropy

$$S(\overline{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \overline{E})\right)$$

N small (< 25) → **EASY**

Inputs $\{x_i\}_1^P$

$$\begin{array}{c} [1,1,-1,\ldots,1,-1,-1] \\ [-1,1,1,\ \ldots,-1,1,-1] \\ \vdots \\ [-1,-1,1,\ \ldots,-1,1,1] \end{array}$$

P

N

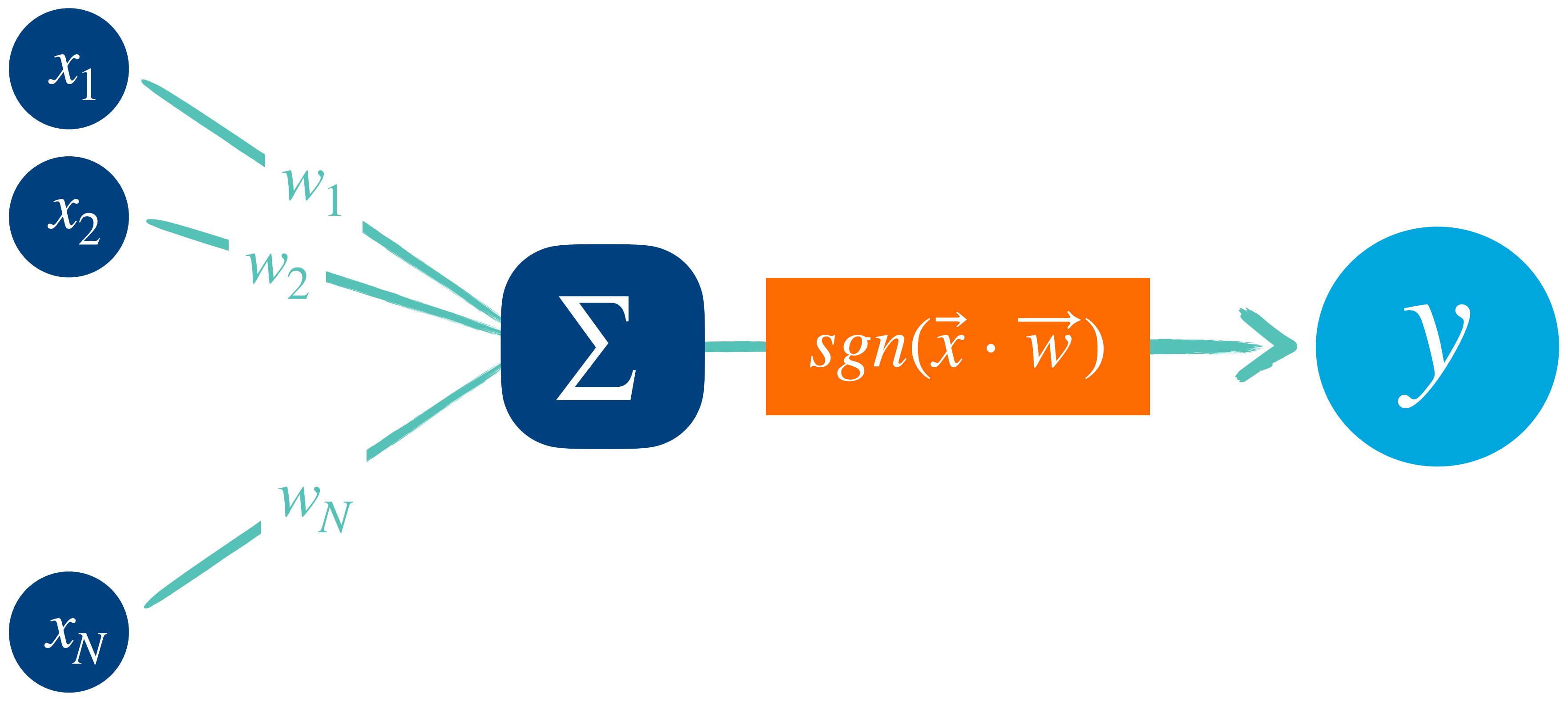
Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \text{sgn}\left(w\, x_i\right)$$

Perceptron



Energy

$$E_w = \sum_i^P \Theta\left(-y_i \cdot \mathrm{sgn}\left(w\, x_i\right)\right)$$

Number of Errors

Entropy

$$\boxed{S(\overline{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \overline{E})\right)}$$

N small (< 25) **EASY**

N big (>30)

Inputs $\{x_i\}_1^P$

P

$[1,1,-1,\ldots,1,-1,-1]$

$[-1,1,1,\ \ldots,-1,1,-1]$

$\vdots$

$[-1,-1,1,\ \ldots,-1,1,1]$

N

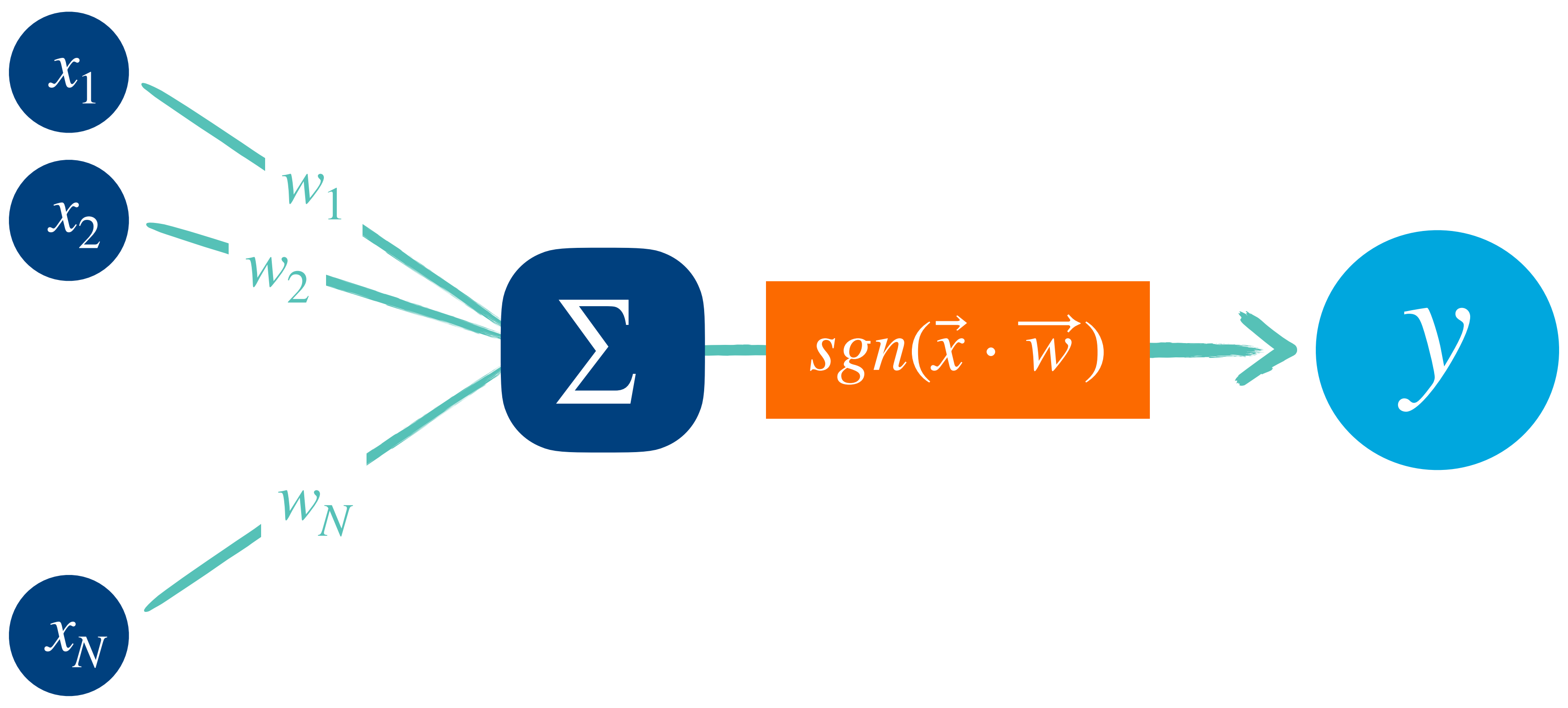
Weight vector w

$[-1,1,-1\ \ldots,-1,-1,1]$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\left(w\, x_i\right)$$

Perceptron



Energy

$$E_w = \sum_i^P \Theta\left(-y_i \cdot \mathrm{sgn}\left(w\, x_i\right)\right)$$

Number of Errors

Entropy

$$S(\bar{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \bar{E})\right)$$

N small (< 25) → **EASY**

N big (>30) → **HARD**

Inputs $\{x_i\}_1^P$

$$\begin{array}{c} [1,1,-1,\ldots,1,-1,-1] \\ [-1,1,1,\ \ldots,-1,1,-1] \\ \vdots \\ [-1,-1,1,\ \ldots,-1,1,1] \end{array}$$

P

N

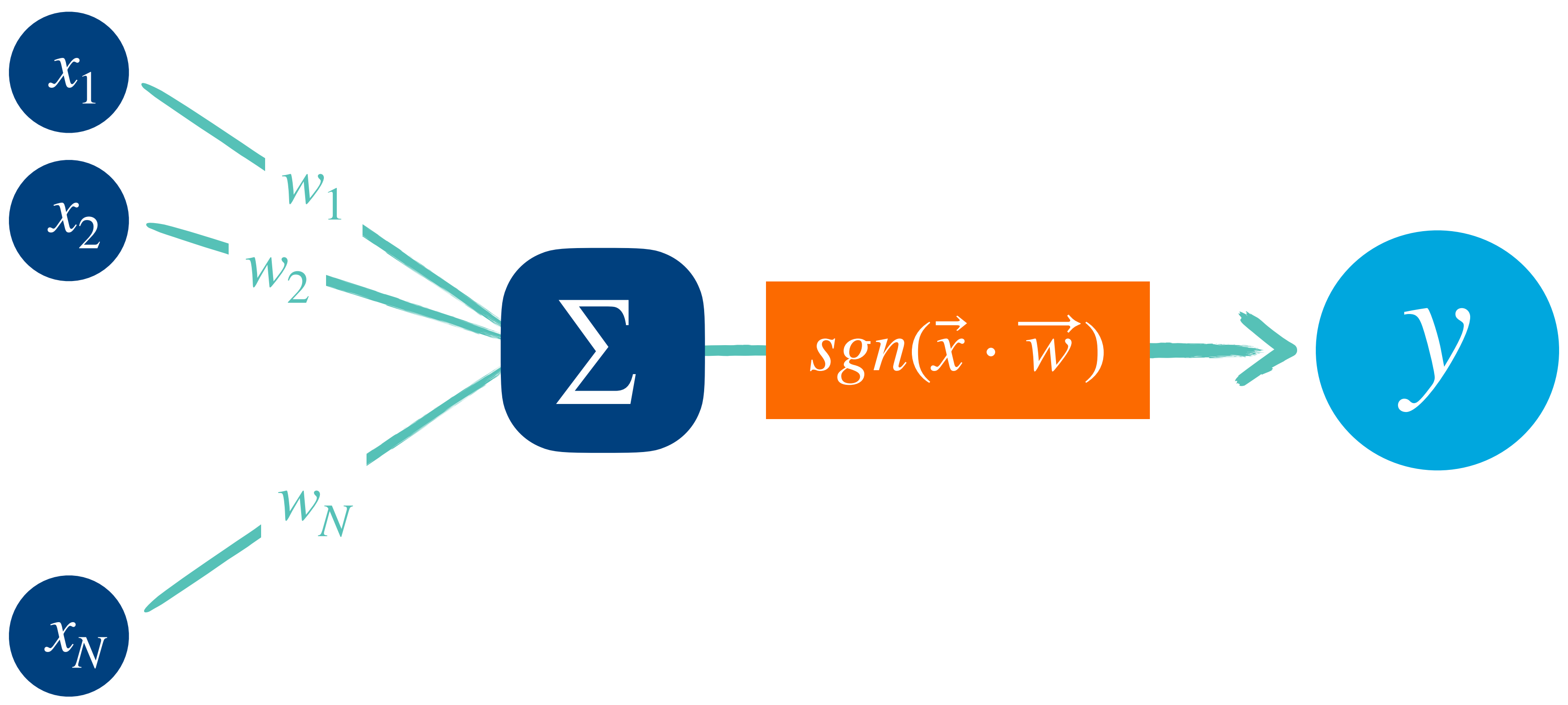
Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\left(w\, x_i\right)$$

Perceptron



Inputs $\{x_i\}_1^P$

$$\begin{array}{c} [1,1,-1,\ldots,1,-1,-1] \\ [-1,1,1,\ \ldots,-1,1,-1] \\ \vdots \\ [-1,-1,1,\ \ldots,-1,1,1] \end{array}$$

P

N

Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\left(w\,x_i\right)$$

Energy

$$E_w = \sum_i^P \Theta\left(-y_i \cdot \mathrm{sgn}\left(w\,x_i\right)\right)$$

Number of Errors

Entropy

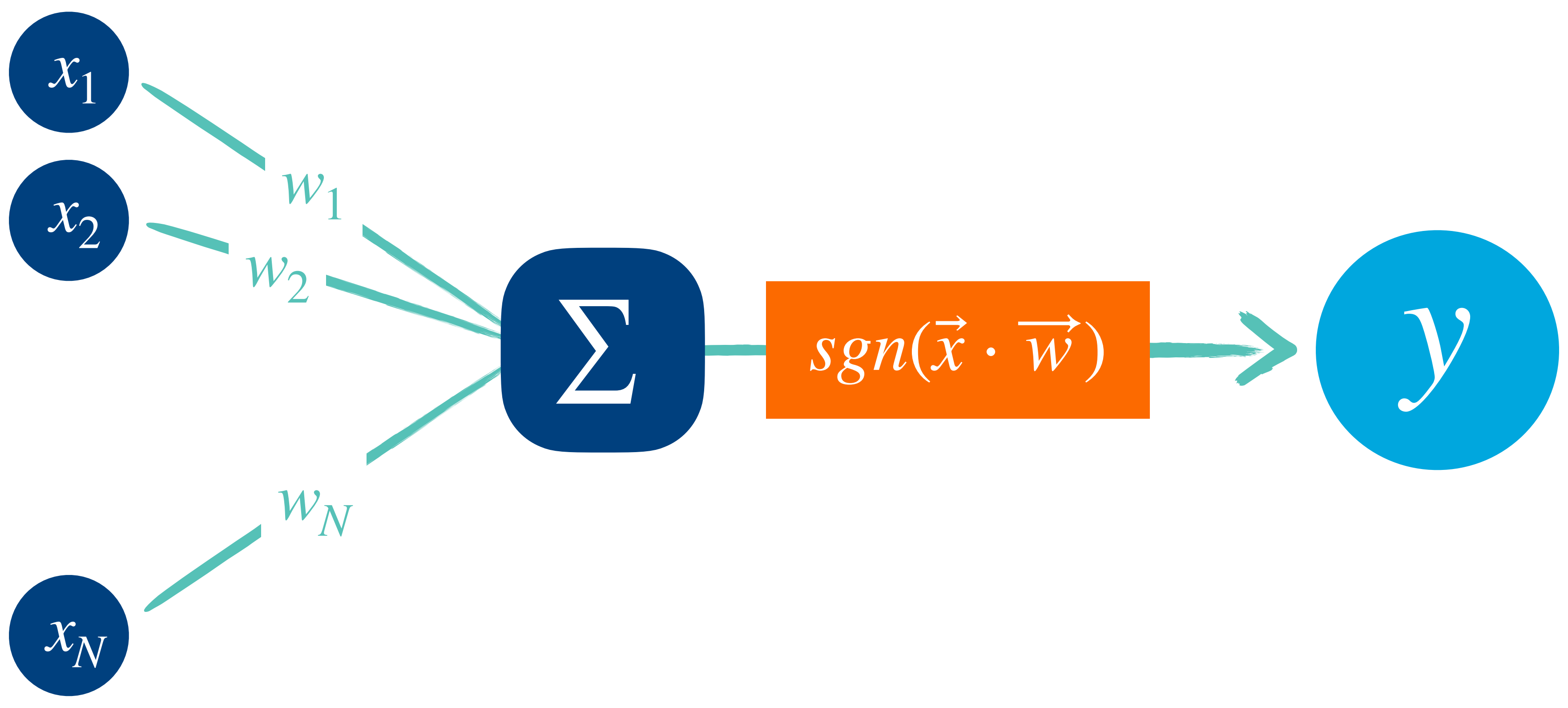
$$\boxed{S(\overline{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \overline{E})\right)}$$

N small (< 25) → **EASY**

N big (>30) → **HARD**

- Huge configurational space $2^{30} = 1073741824$

Perceptron



Inputs $\{x_i\}_1^P$

$$\begin{array}{c|c} & [1,1,-1,\ldots,1,-1,-1] \\ P & [-1,1,1,\ldots,-1,1,-1] \\ & \vdots \\ & [-1,-1,1,\ldots,-1,1,1] \end{array}$$

N

Weight vector w

$$[-1,1,-1 \ldots, -1,-1,1]$$

Labels $\{y_i\}$

$$y_i = \mathrm{sgn}\,(w\,x_i)$$

Energy

$$E_w = \sum_i^P \Theta\left(-y_i \cdot \mathrm{sgn}\,(w\,x_i)\right)$$

Number of Errors

Entropy

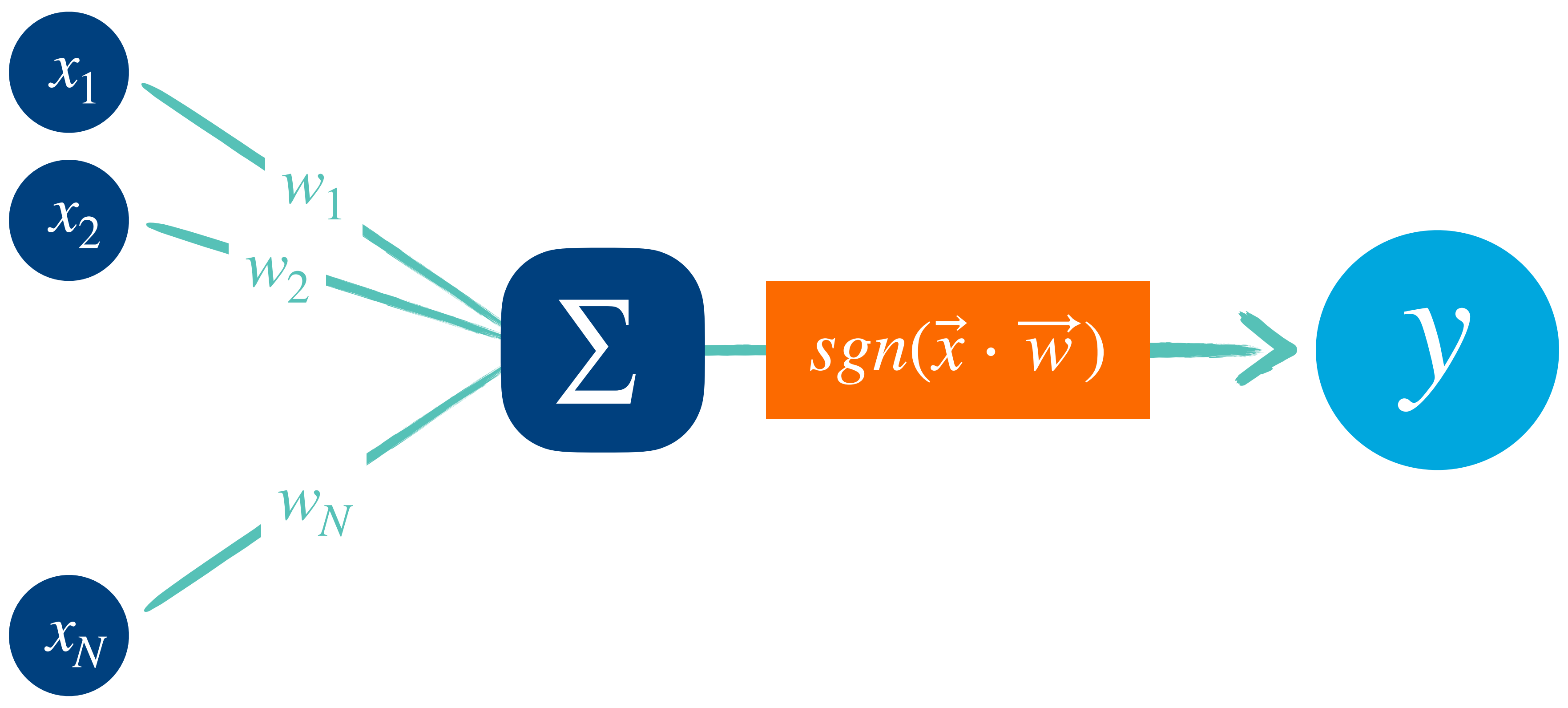
$$\boxed{S(\bar{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \bar{E})\right)}$$

N small (< 25) $\longrightarrow$ **EASY**

N big (>30) $\longrightarrow$ **HARD**

- Huge configurational space $2^{30} = 1073741824$
- Sampling algorithms get trapped in local minima

Perceptron



Energy

$$E_w = \sum_i^P \Theta\left(-y_i \cdot \mathrm{sgn}\left(w\, x_i\right)\right)$$

Number of Errors

Entropy

$$\boxed{S(\overline{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \overline{E})\right)}$$

N small (< 25) → **EASY**

N big (>30) → **HARD**

Inputs $\{x_i\}_1^P$

$$\begin{array}{c} [1,1,-1,\ldots,1,-1,-1] \\ [-1,1,1,\ldots,-1,1,-1] \\ \vdots \\ [-1,-1,1,\ldots,-1,1,1] \end{array}$$

P (rows), N (columns)

Weight vector w

$$[-1,1,-1\ \ldots,-1,-1,1]$$

Labels $\{y_i\}$

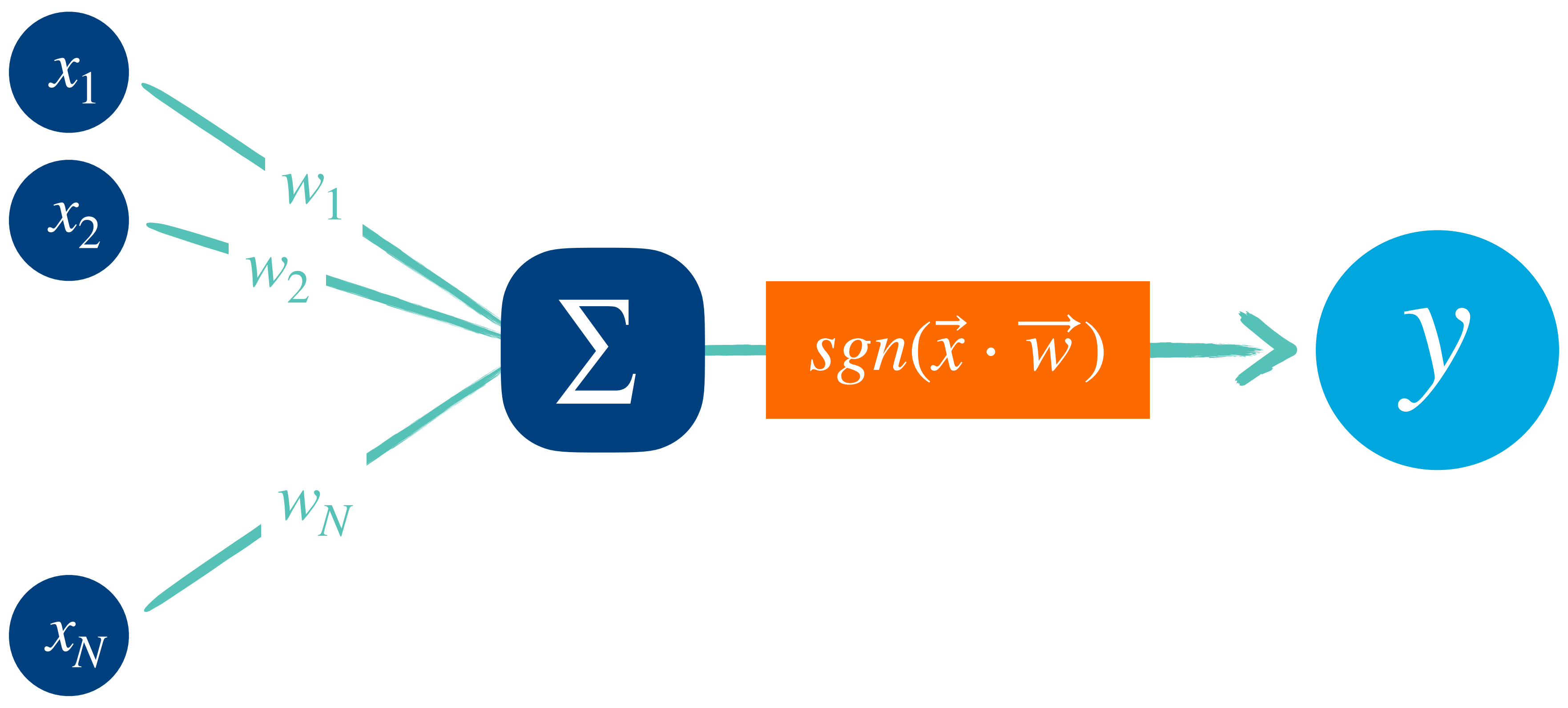
$$y_i = \mathrm{sgn}\left(w\, x_i\right)$$

- Huge configurational space $2^{30} = 1073741824$
- Sampling algorithms get trapped in local minima

→ **Self-consistent entropy estimation**

→ **Uniform exploration of the energy spectrum**

Perceptron



Energy

$$E_w = \sum_{i}^{P} \Theta\left(-y_i \cdot \operatorname{sgn}\left(w\, x_i\right)\right)$$

Number of Errors

Entropy

$$S(\overline{E}) = \log\left(\sum_{\{w\}} \delta(E_w - \overline{E})\right)$$

Inputs $\{x_i\}_1^P$

P

$[1,1,-1,\ldots,1,-1,-1]$

$[-1,1,1,\ \ldots,-1,1,-1]$

$\vdots$

$[-1,-1,1,\ \ldots,-1,1,1]$

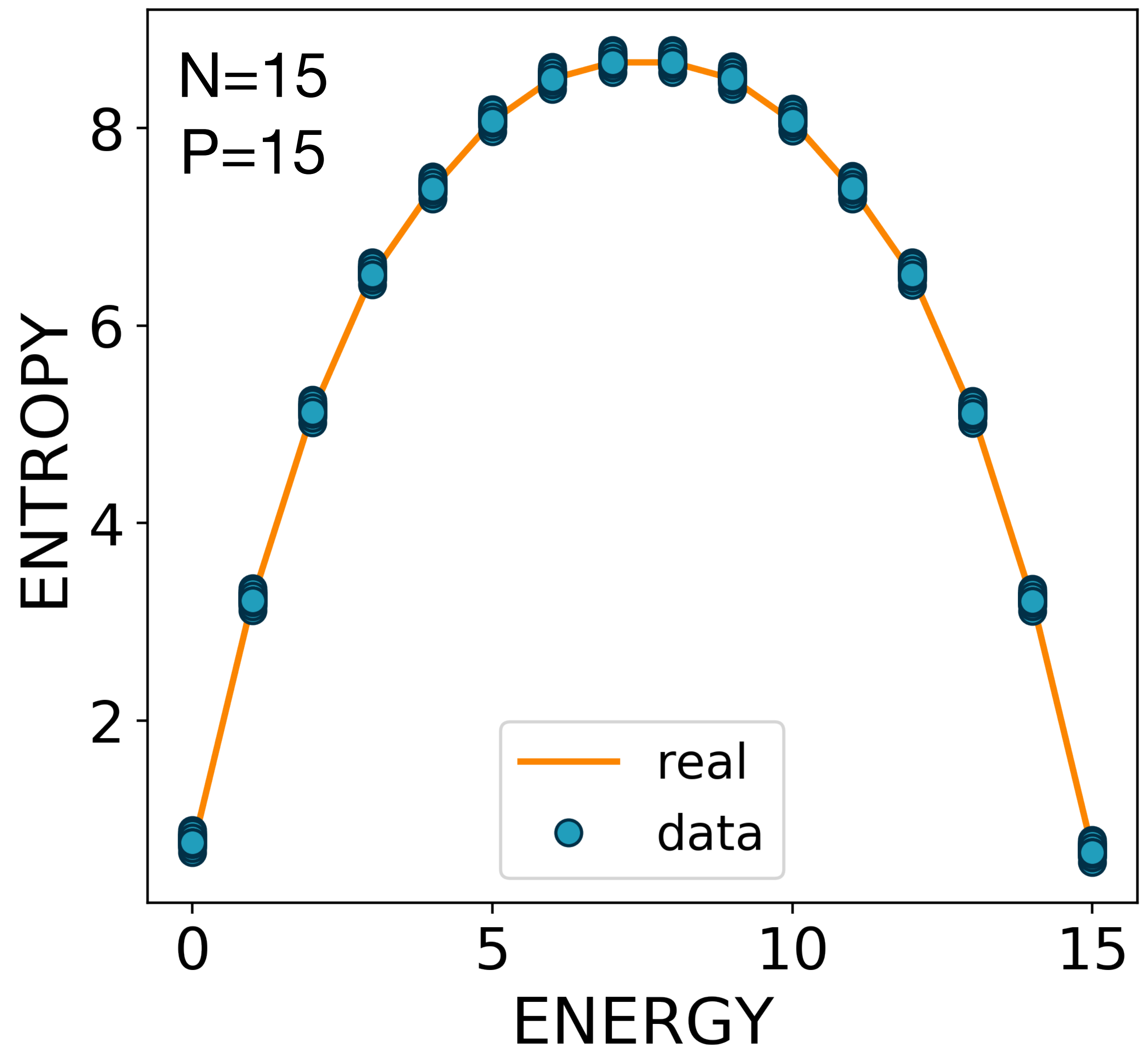
N

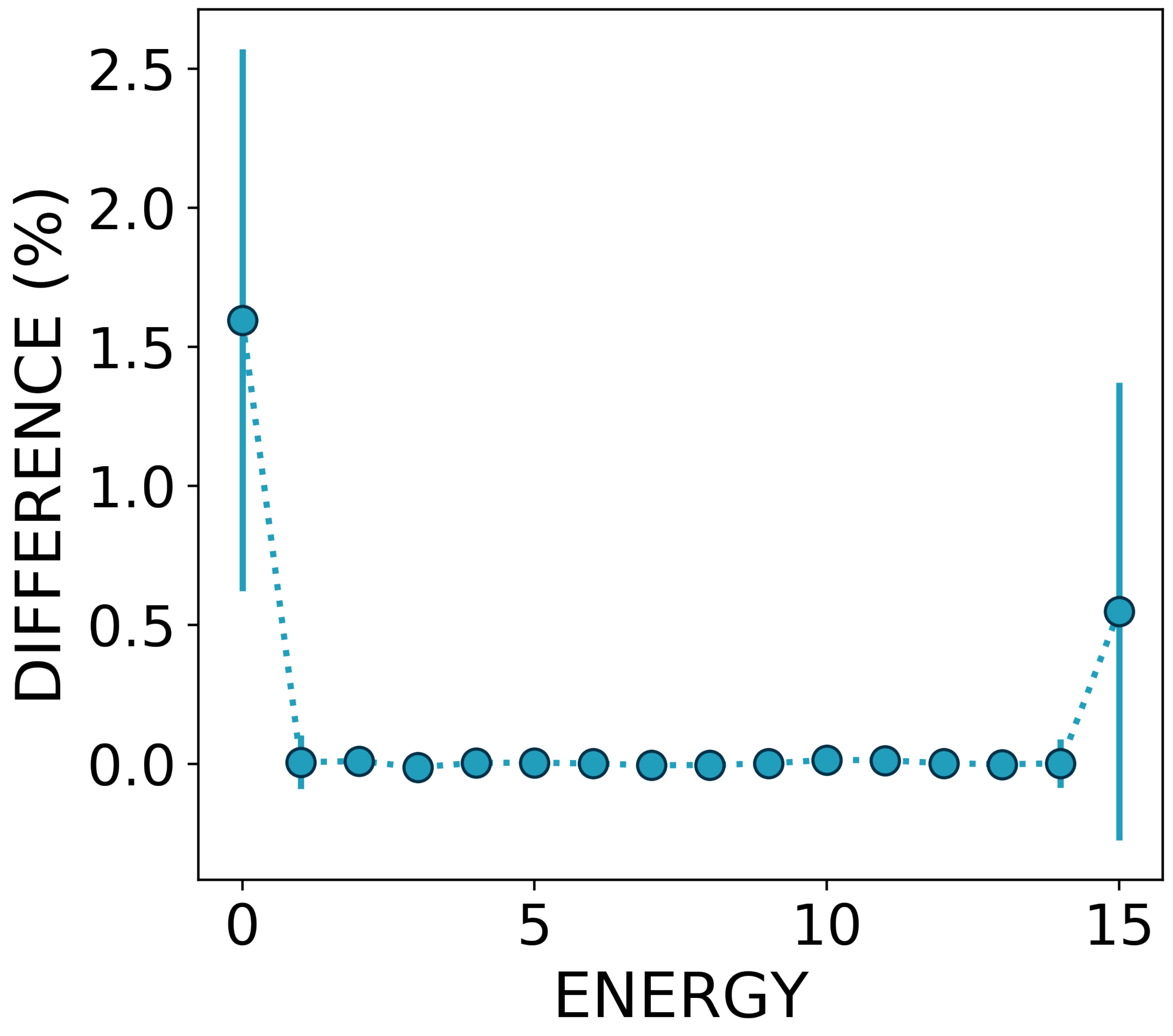
Weight vector w

$[-1,1,-1\ \ldots,-1,-1,1]$

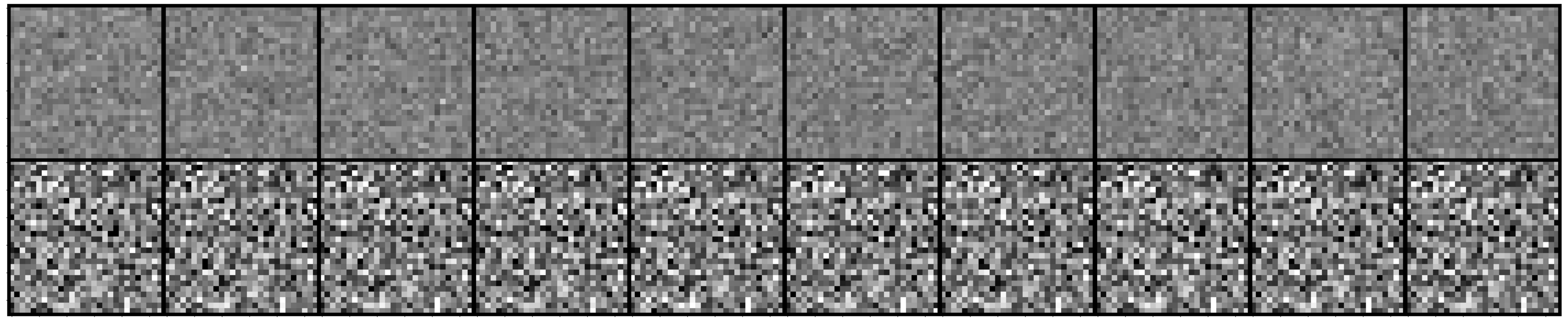
Labels $\{y_i\}$

$$y_i = \operatorname{sgn}\left(w\, x_i\right)$$

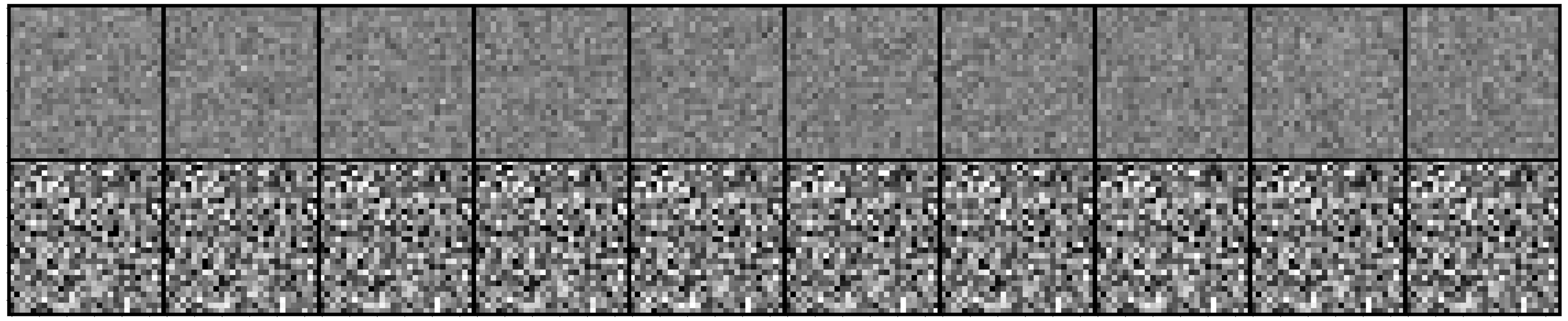


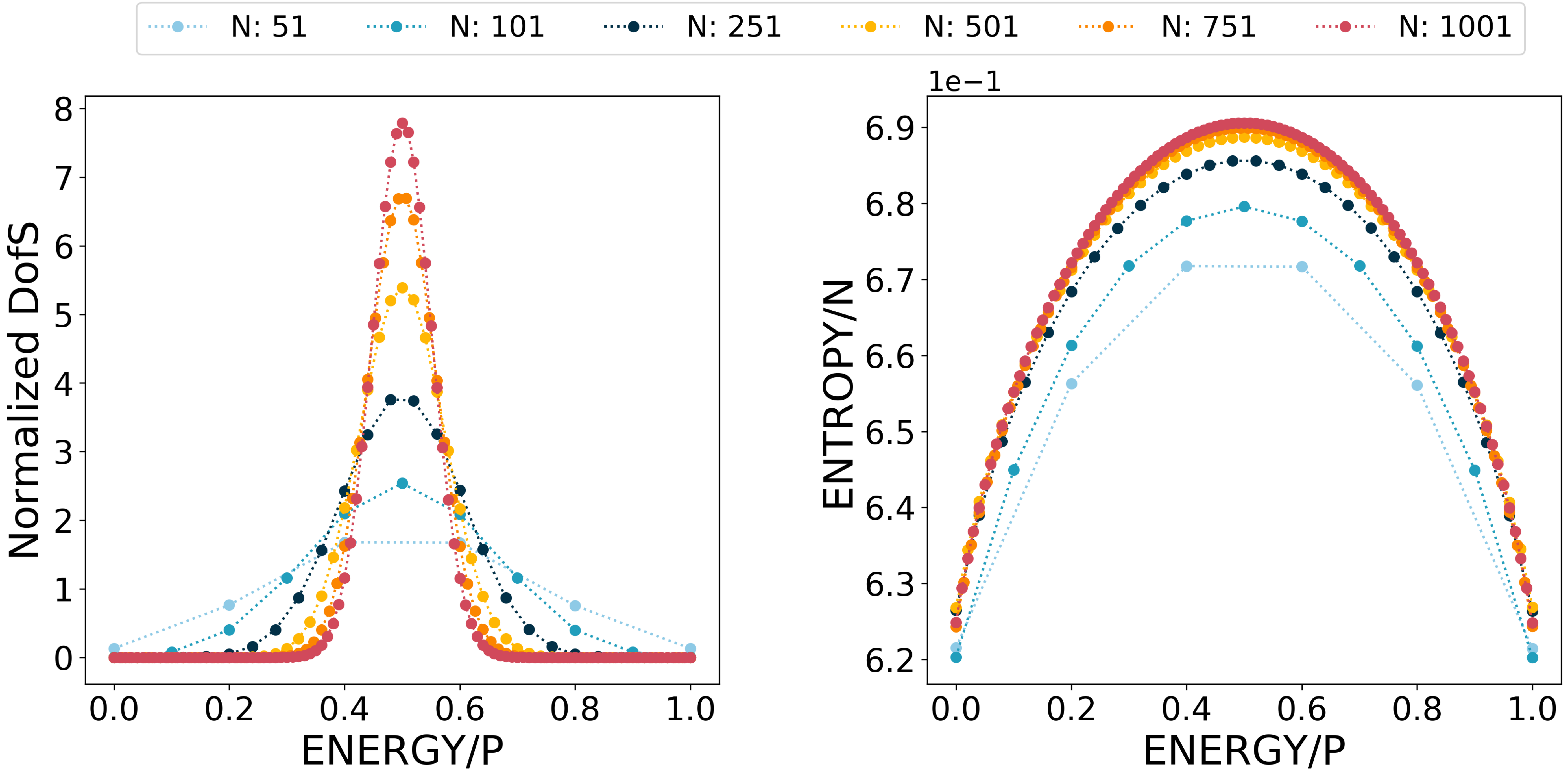


Random Data

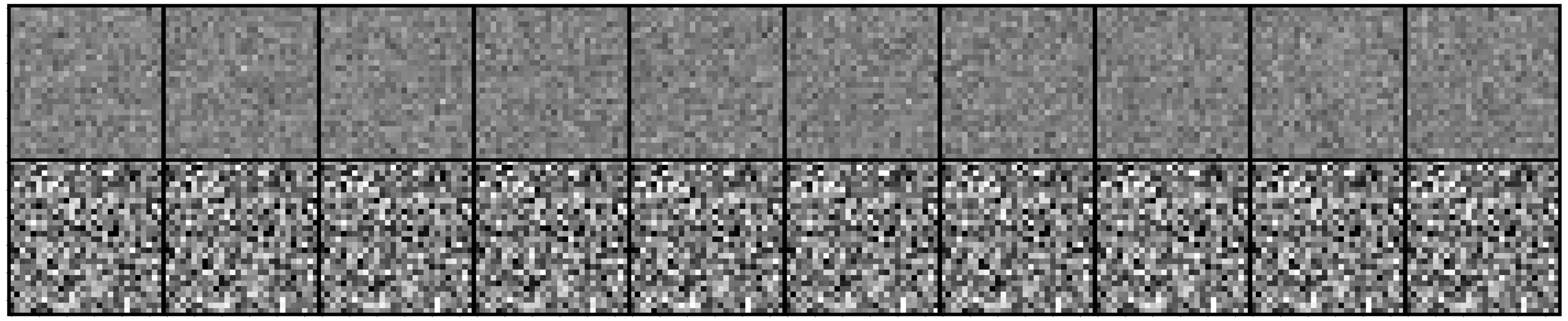


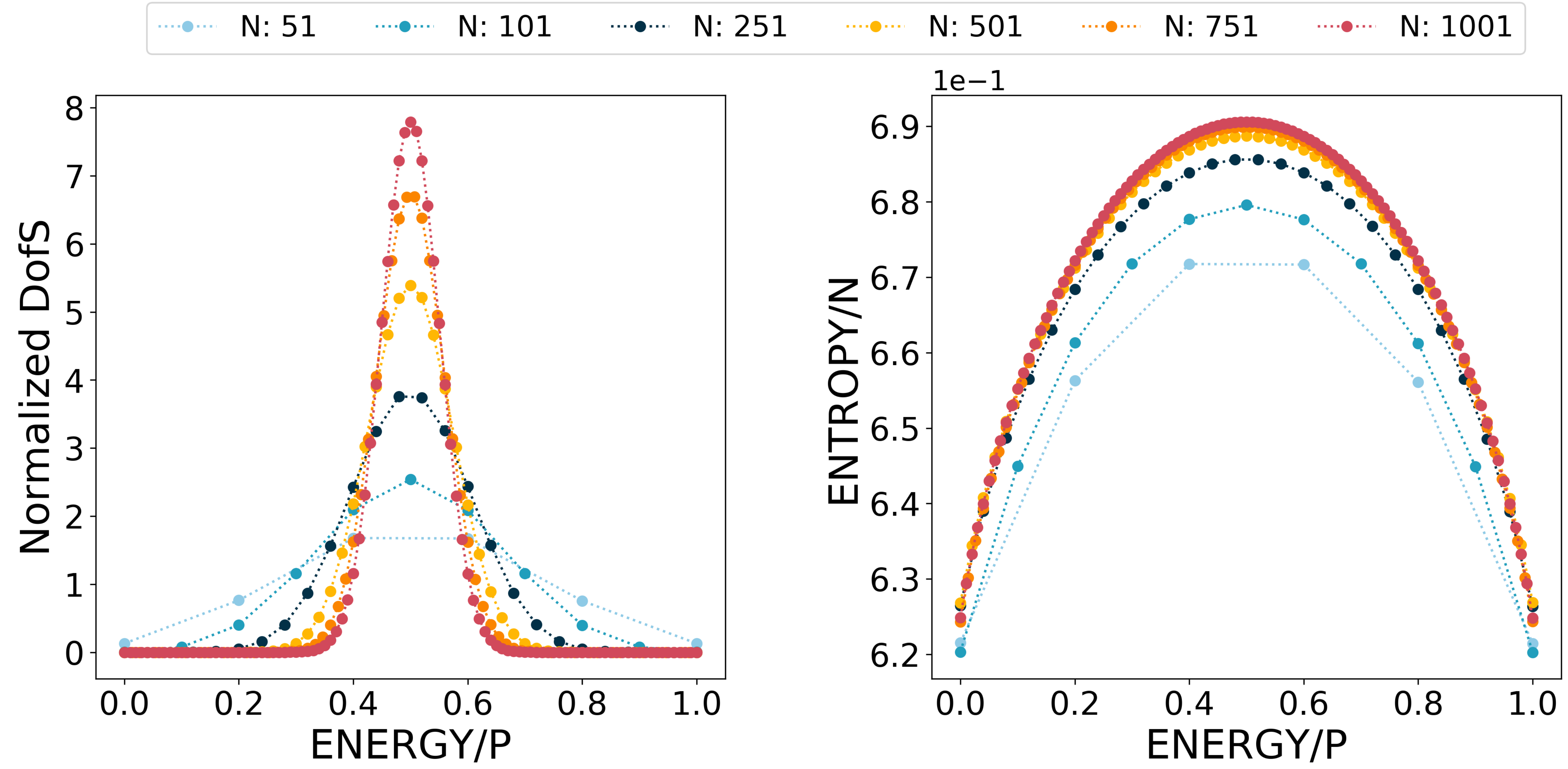
Random Data



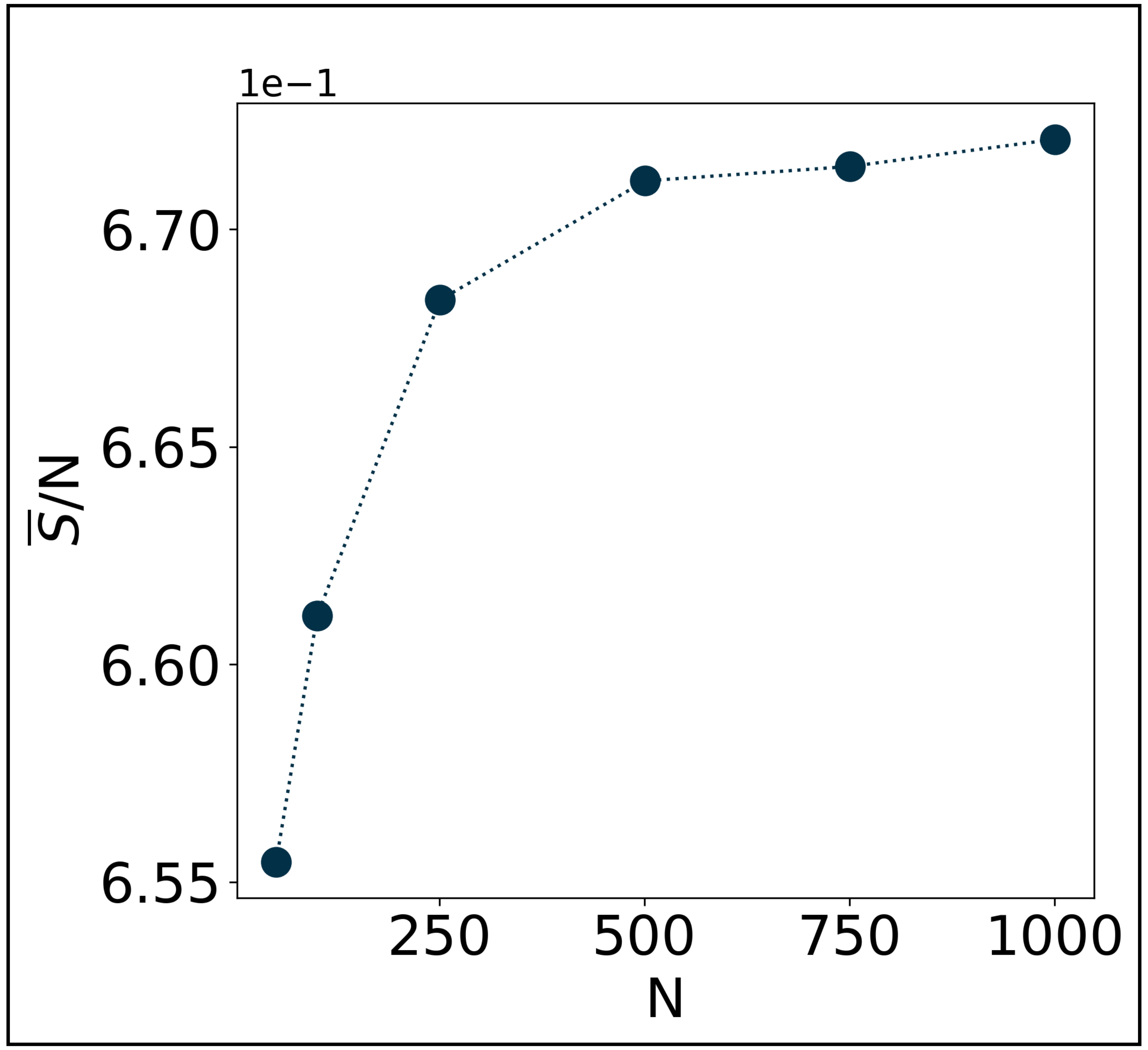

N: 51
N: 101
N: 251
N: 501
N: 751
N: 1001
Normalized DofS
ENERGY/P
1e−1
ENTROPY/N
ENERGY/P

Random Data





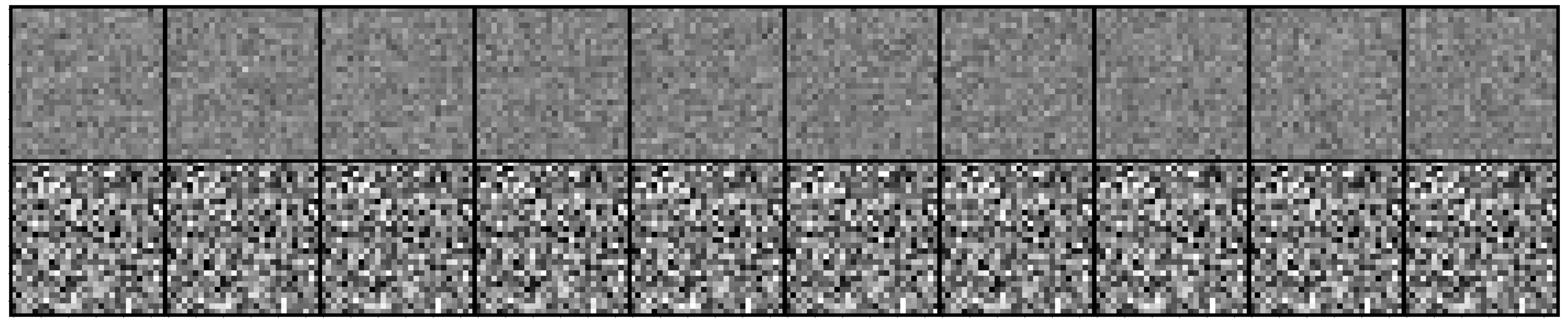
N: 51
N: 101
N: 251
N: 501
N: 751
N: 1001
Normalized DofS
ENERGY/P
1e−1
ENTROPY/N
ENERGY/P

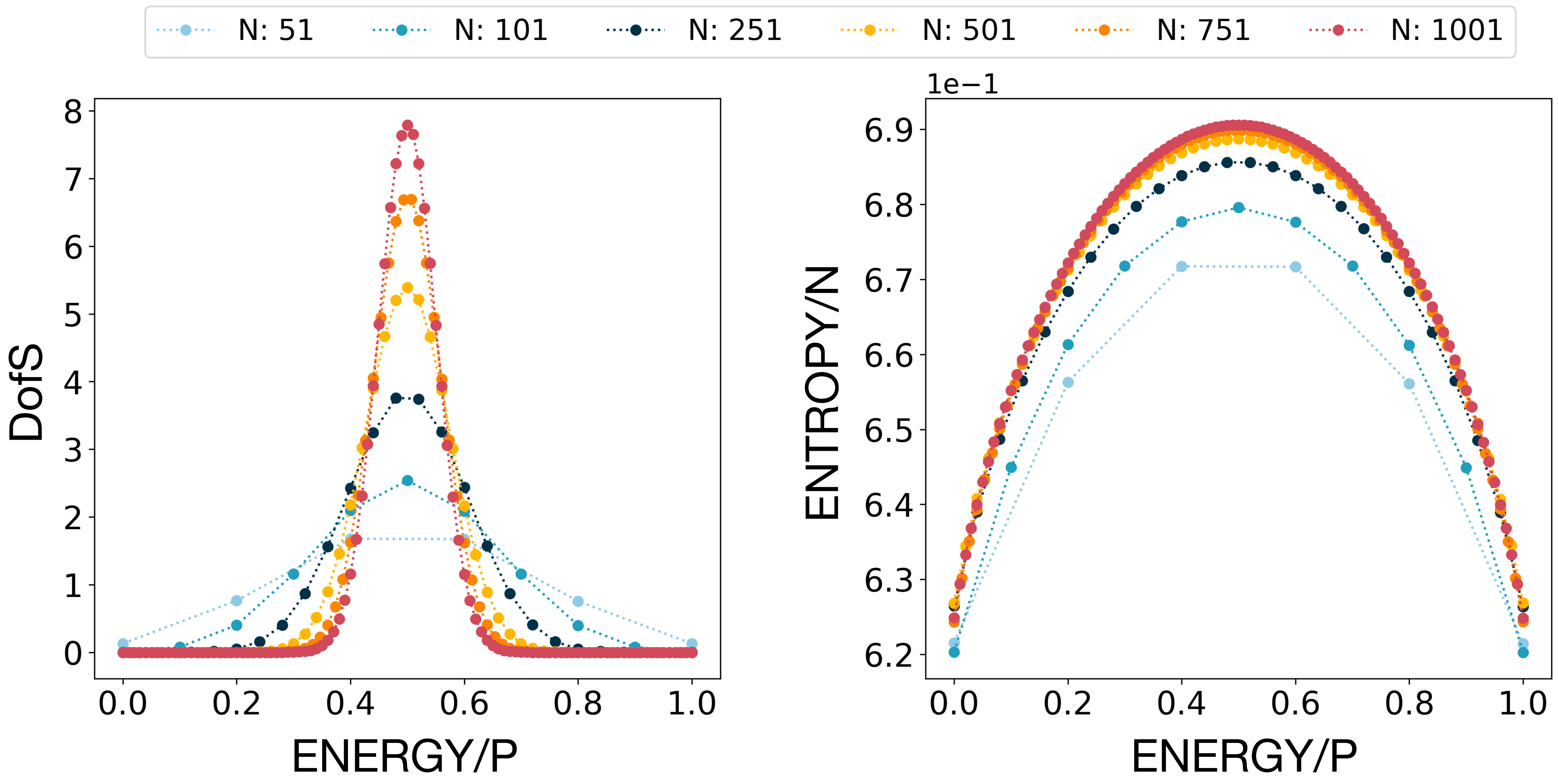


1e−1
$\overline{S}$/N
N

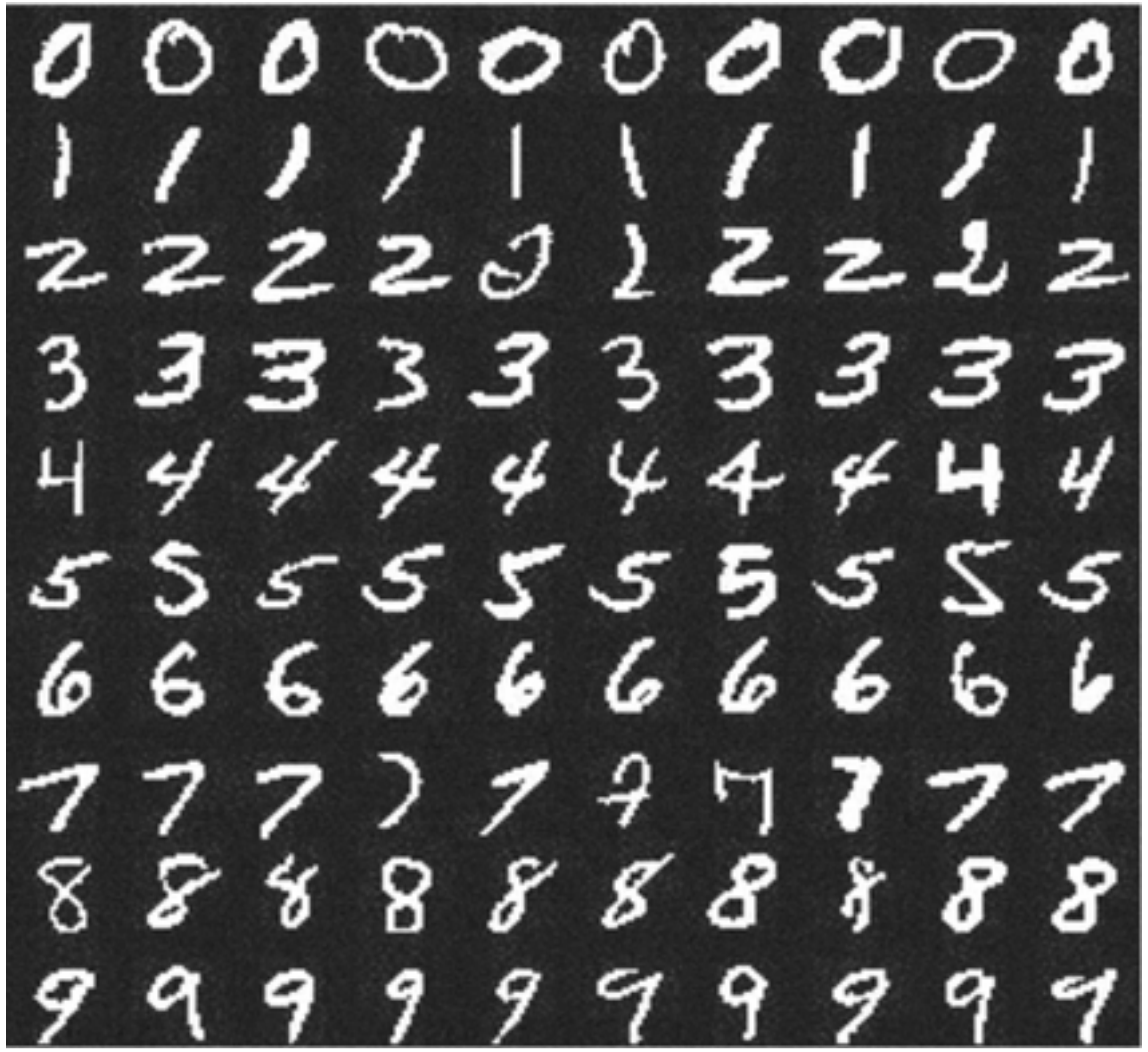
Random vs Real

Random



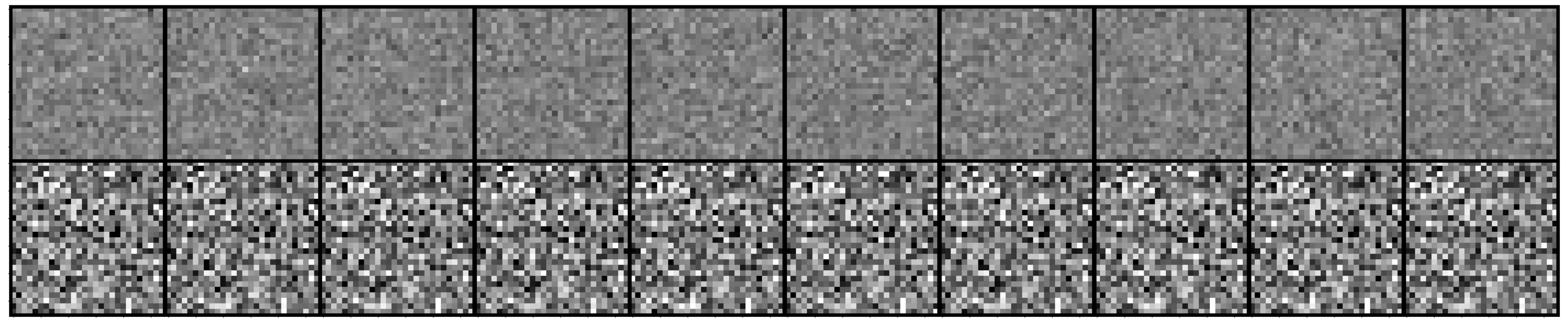


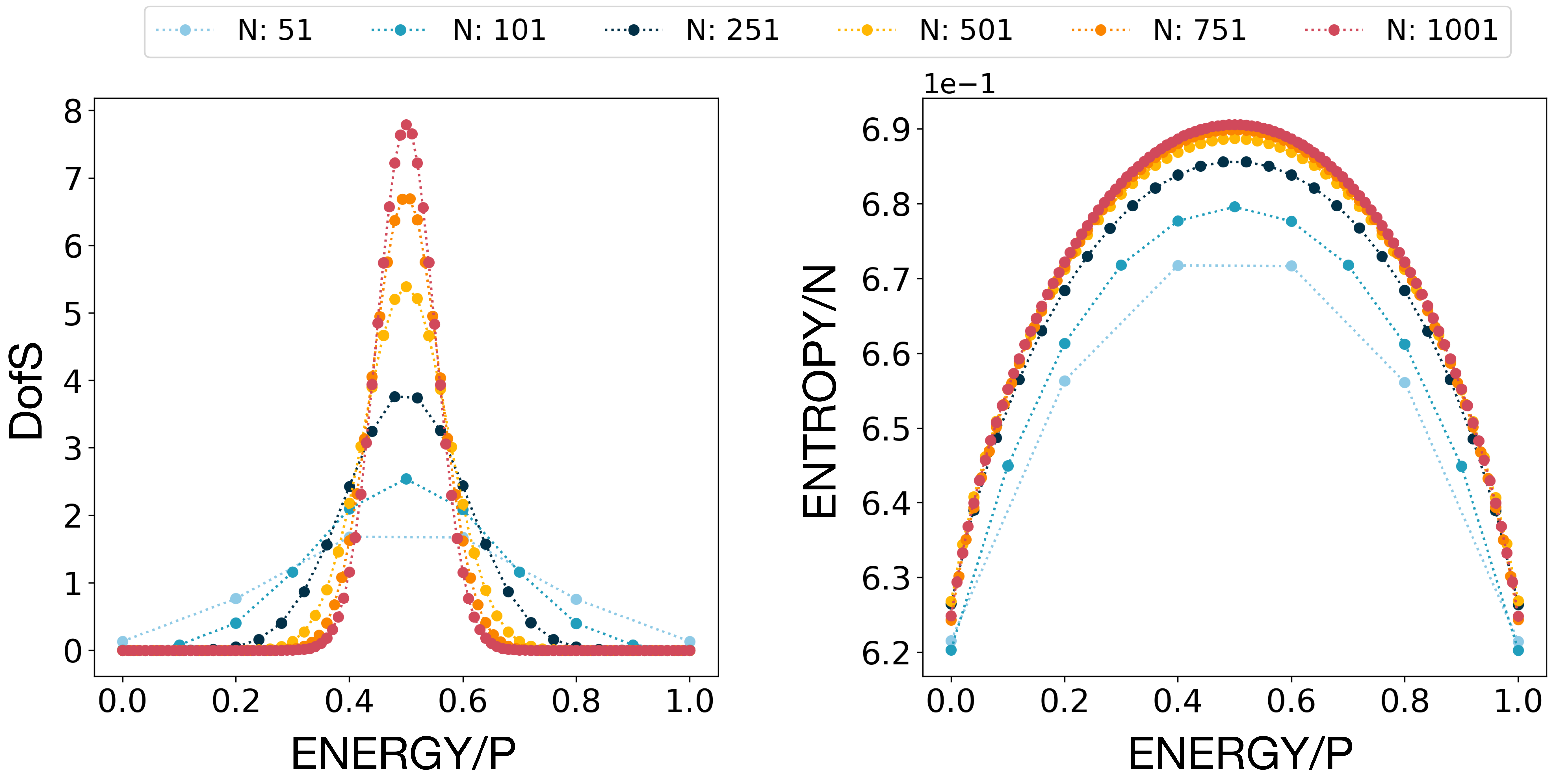
MNIST



Random vs Real

Random

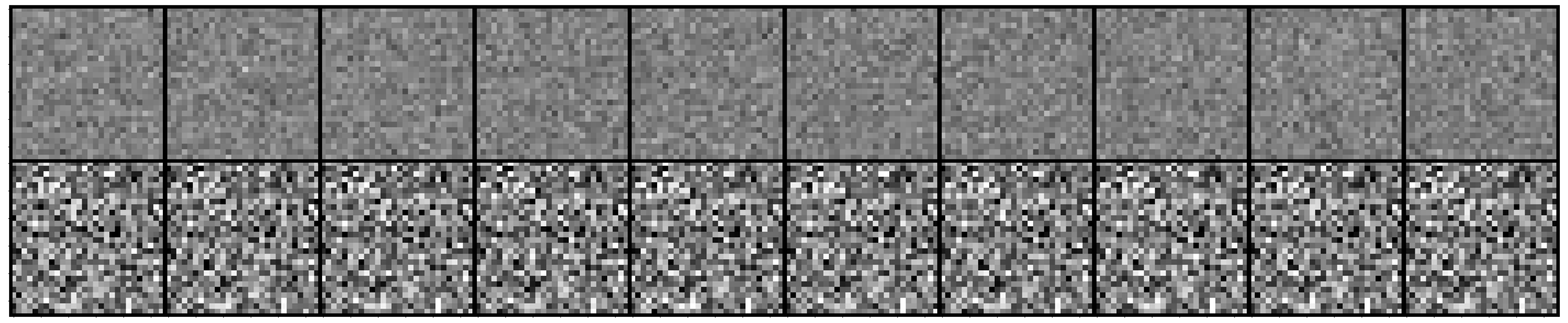


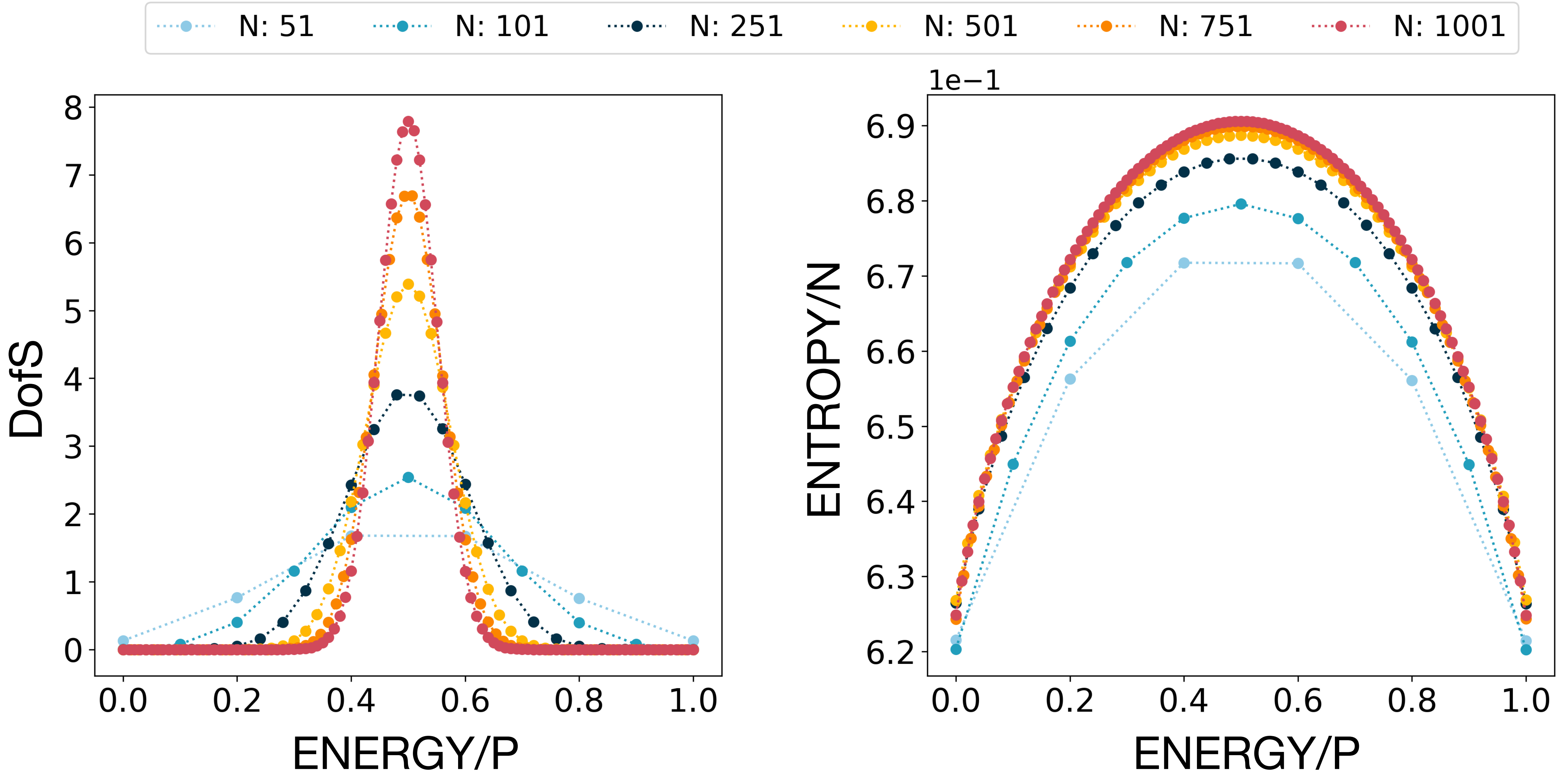


MNIST

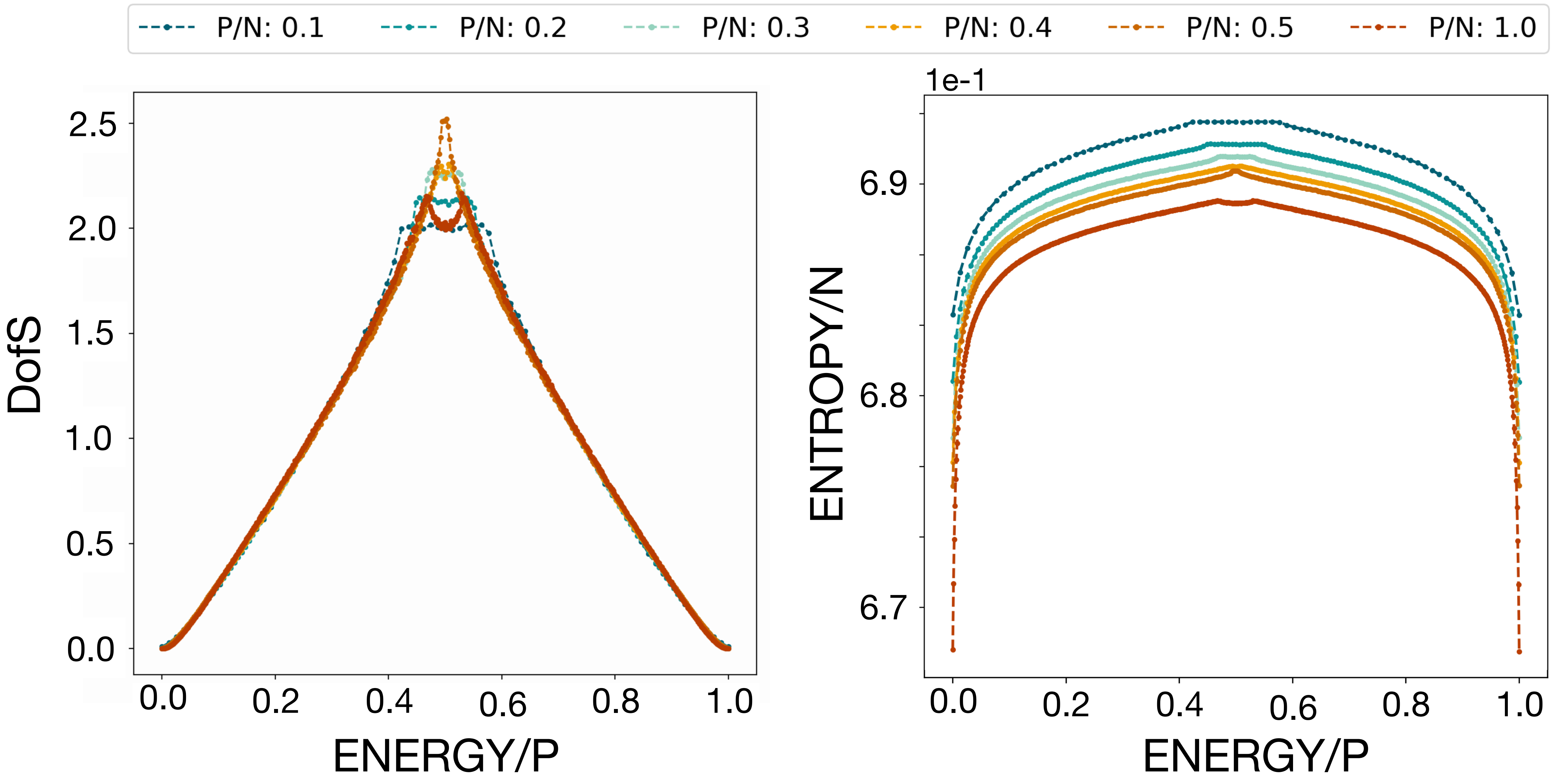
Random vs Real

Random

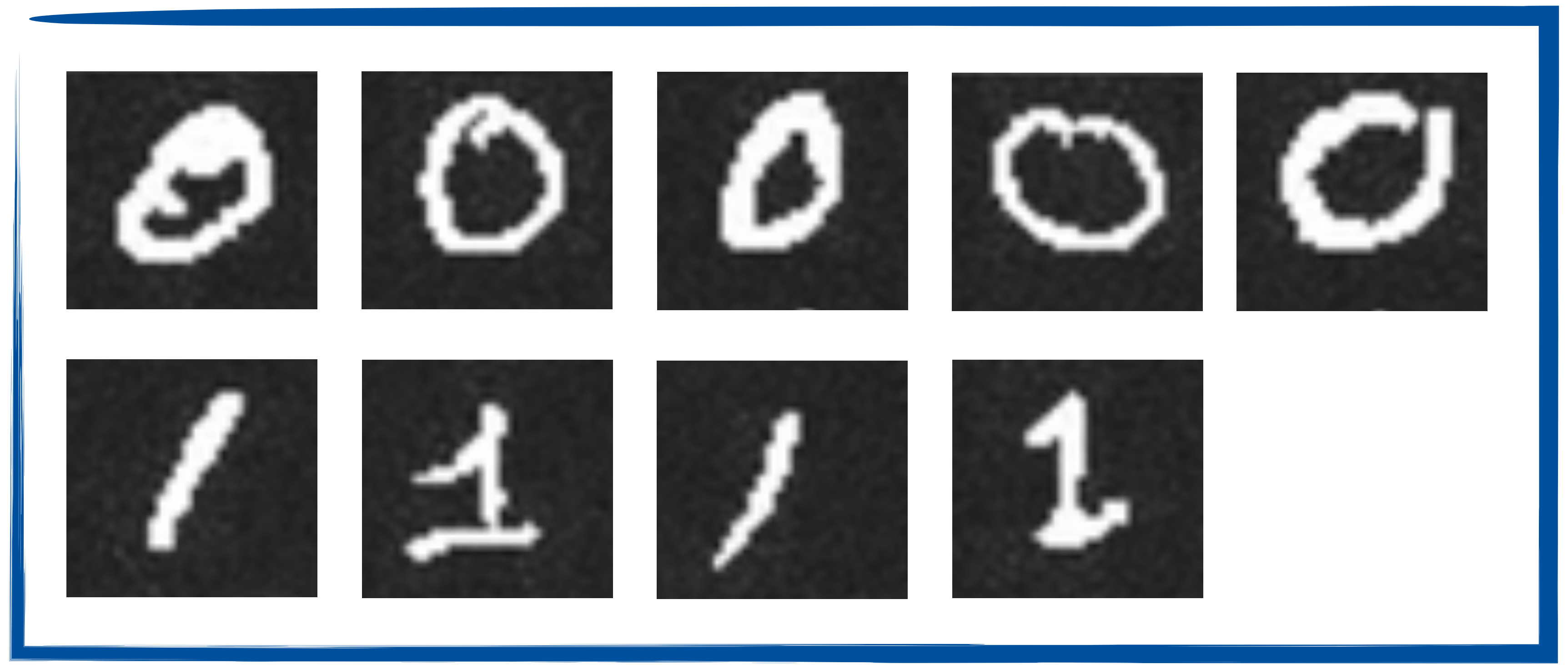




MNIST



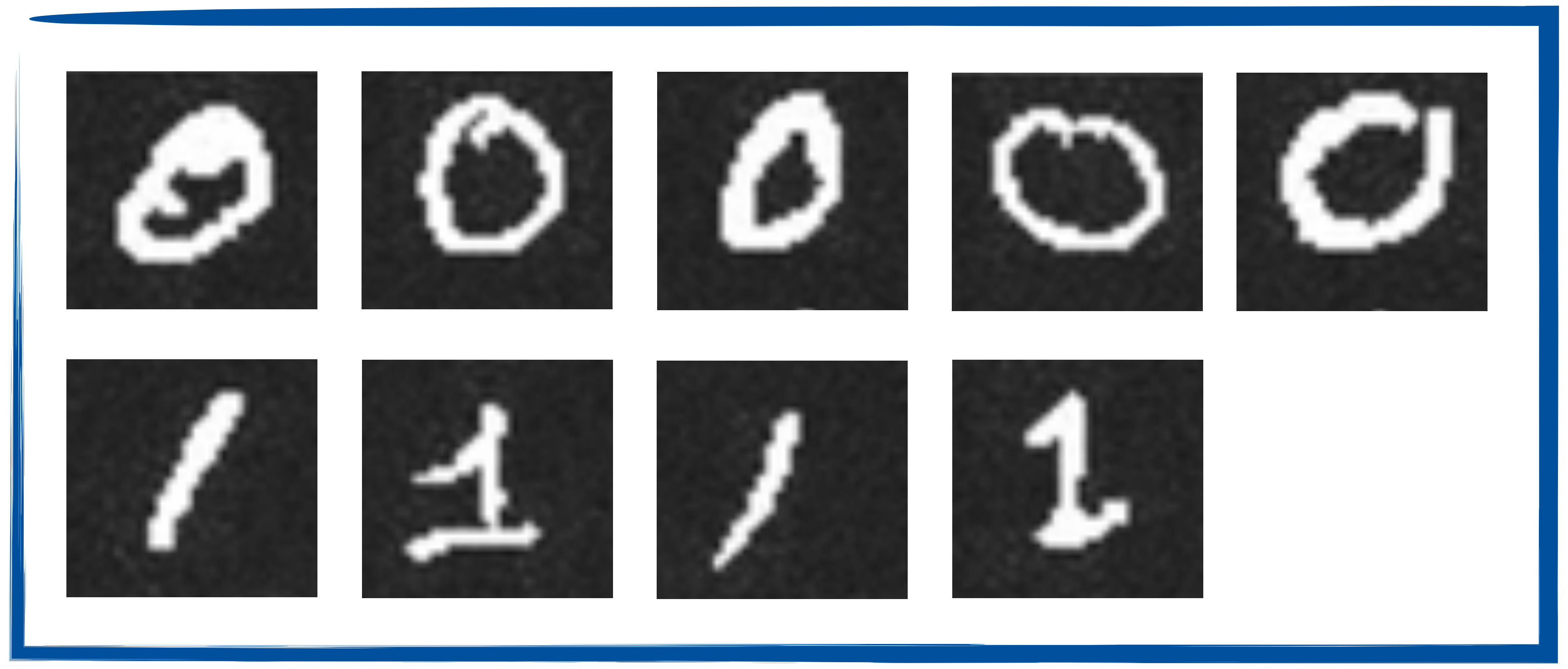
Class Unbalancing



$$P_0 \neq P_1$$

$$\frac{P_1}{P} \neq 0.5$$

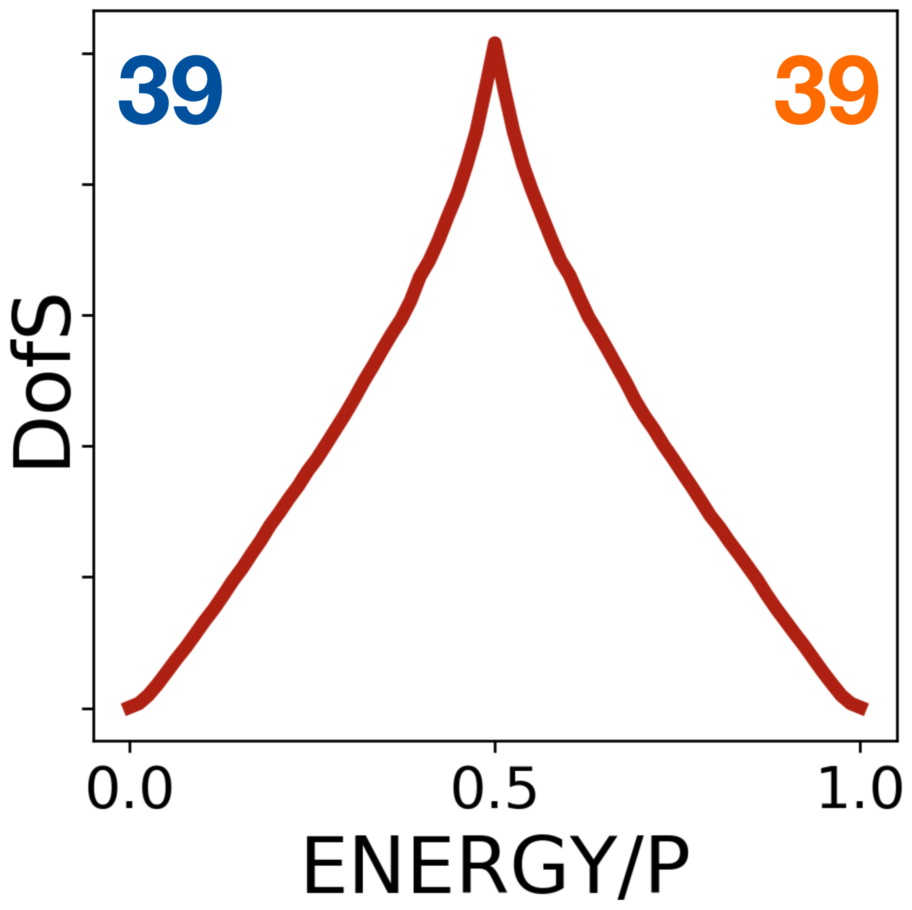
Class Unbalancing



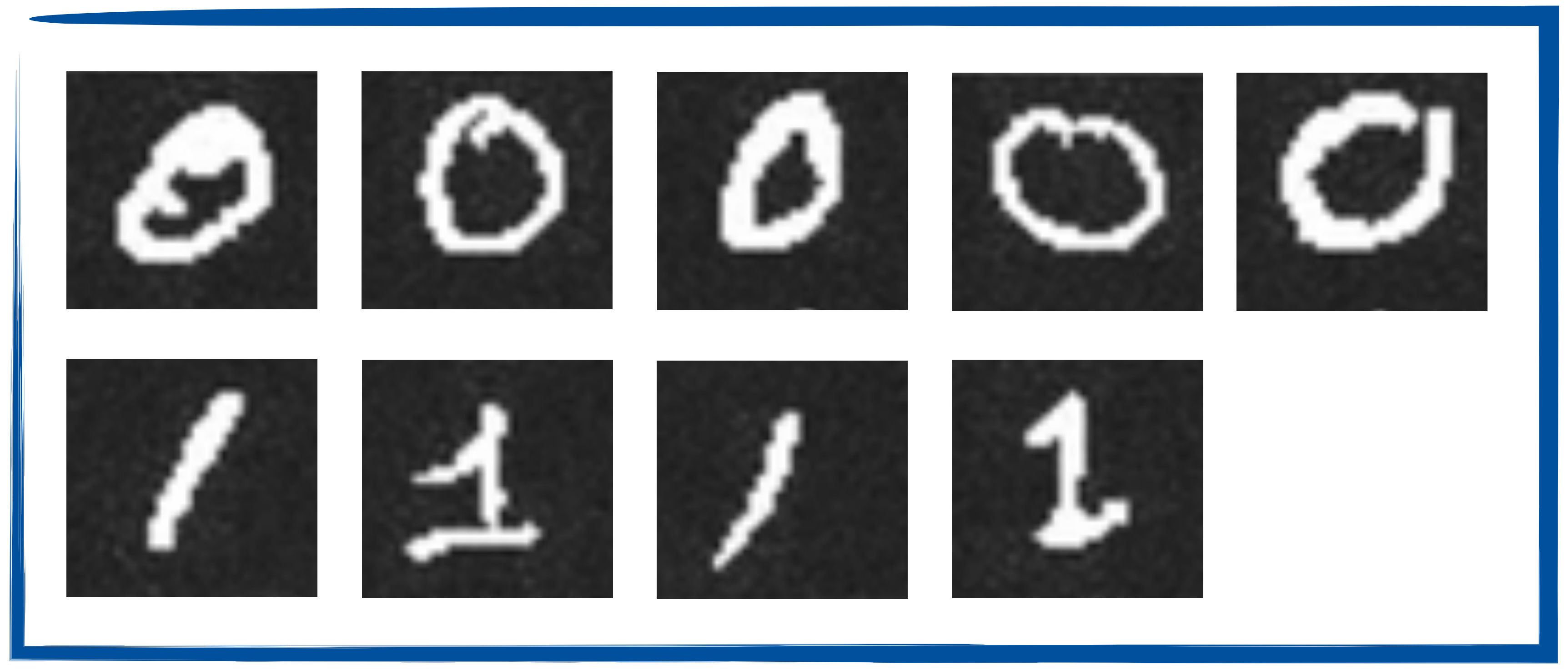
$$P_0 \neq P_1$$

$$\frac{P_1}{P} \neq 0.5$$

$$P_1 = P_0$$



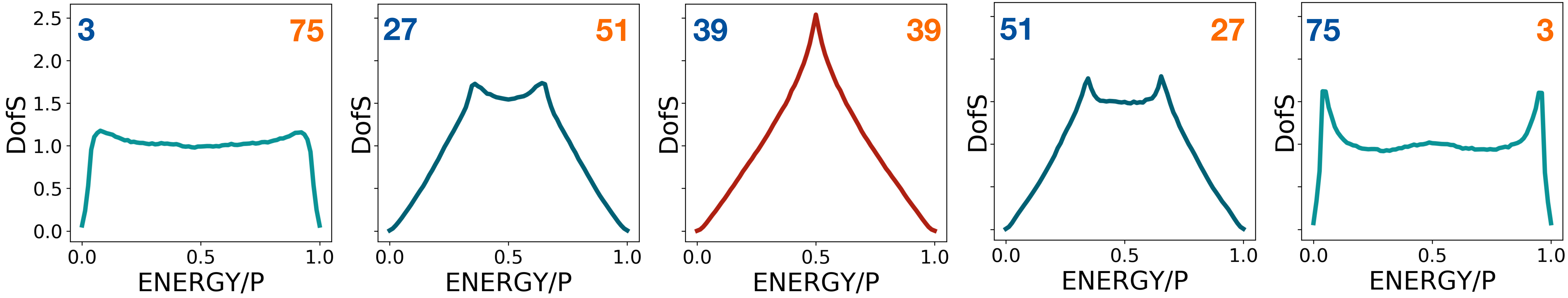
Class Unbalancing



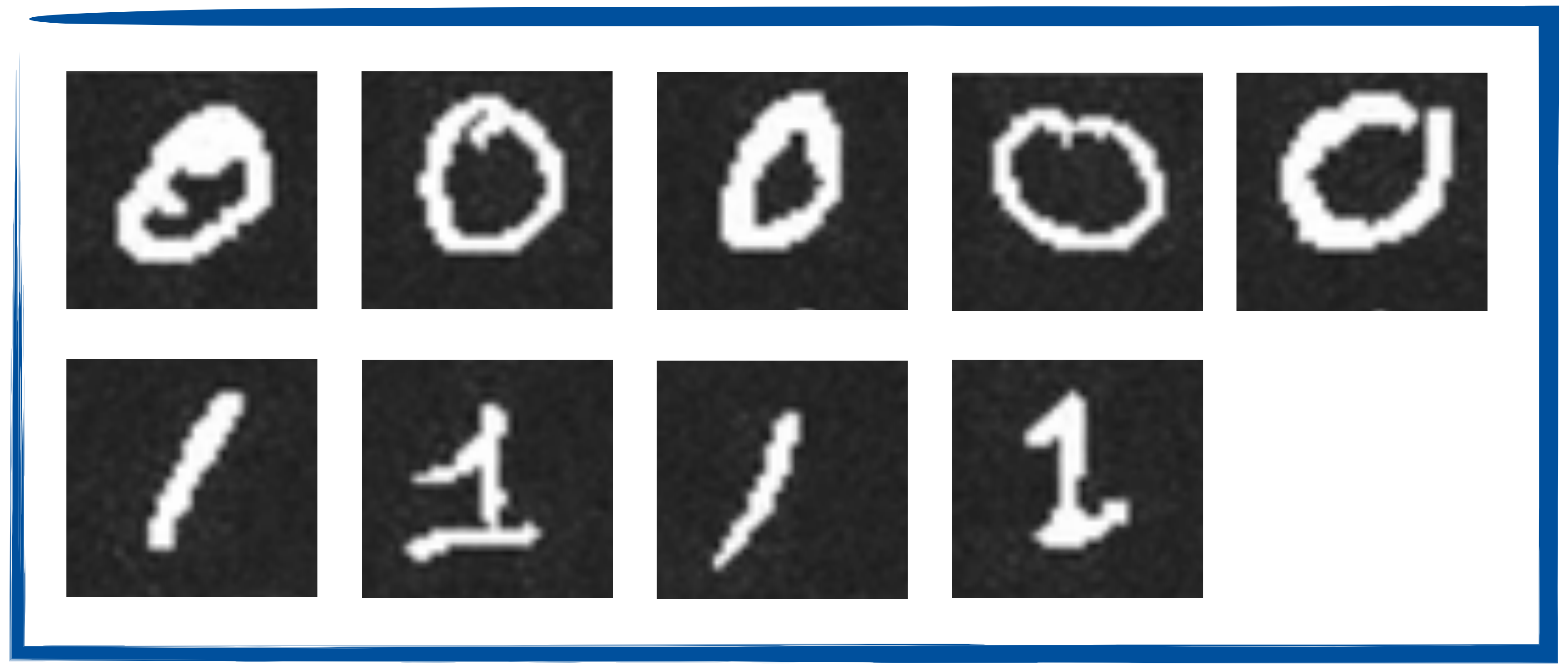
$$P_0 \neq P_1$$

$$\frac{P_1}{P} \neq 0.5$$

$P_1 \ll P_0$ | $P_1 = P_0$ | $P_1 \gg P_0$

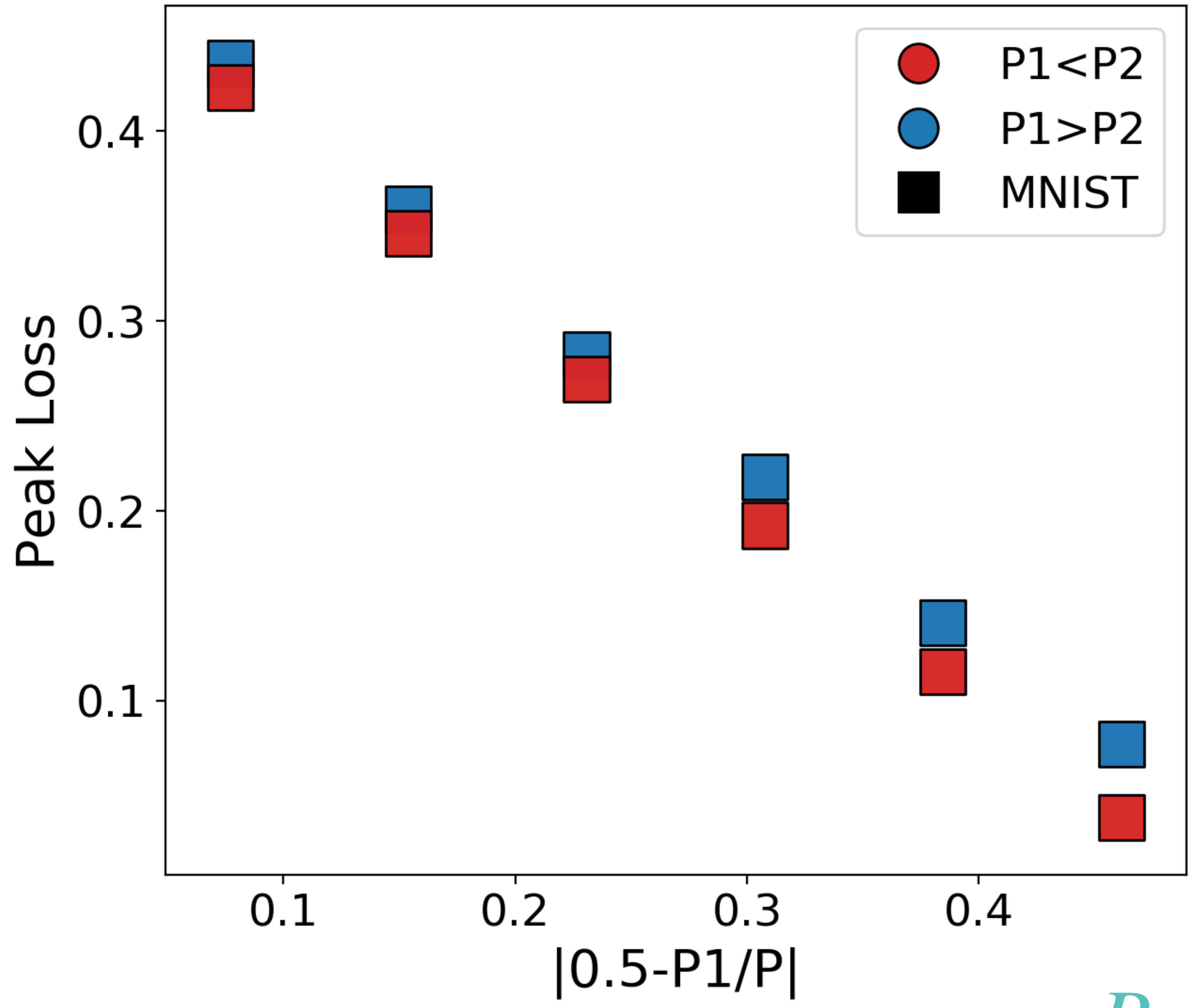


Class Unbalancing

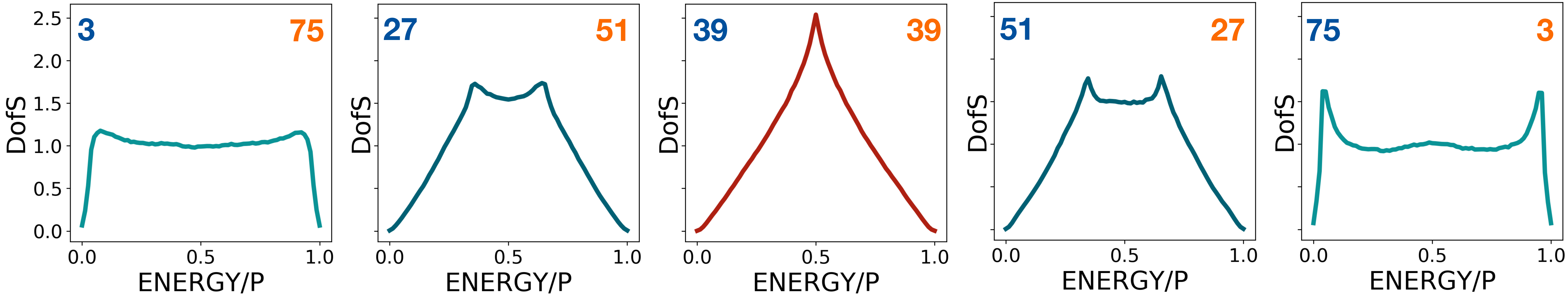


$$P_0 \neq P_1$$

$$\frac{P_1}{P} \neq 0.5$$



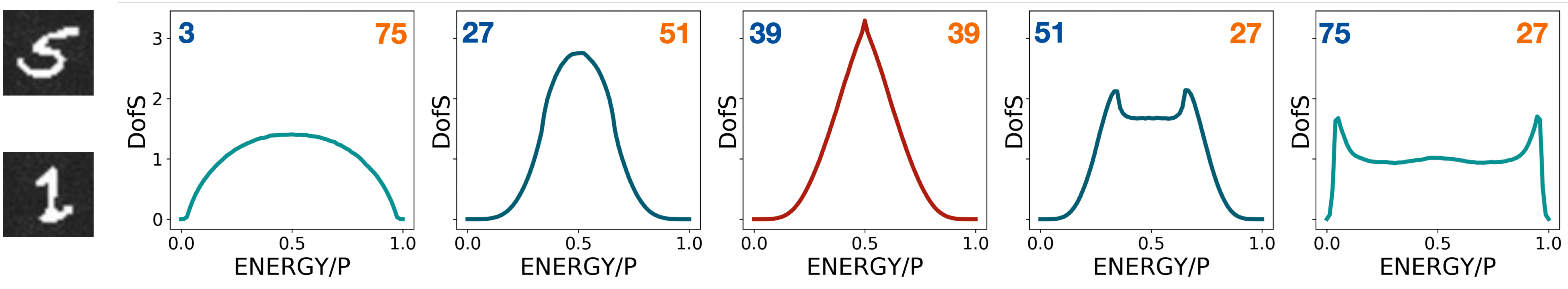
$P_1 \ll P_0$ $\quad P_1 = P_0$ $\quad P_1 \gg P_0$



Class Unbalancing

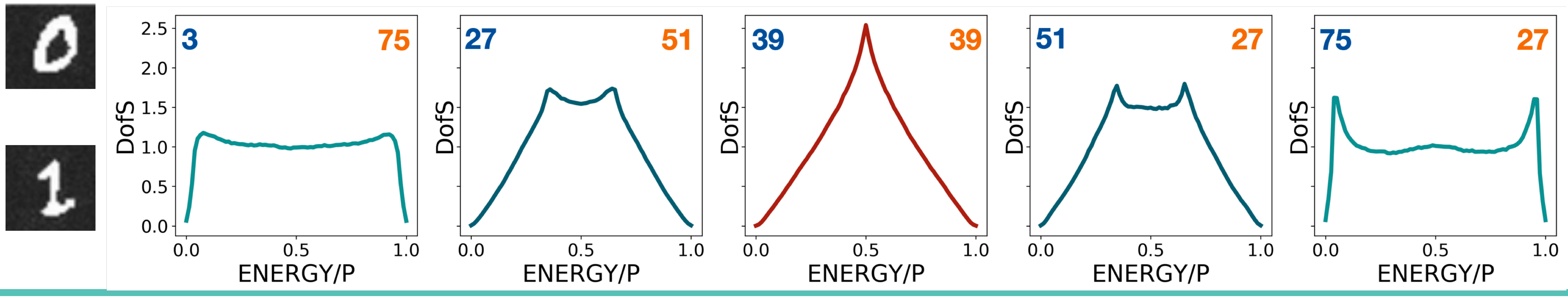
5 vs 1

$P_1 \ll P_5$ $P_1 \gg P_5$



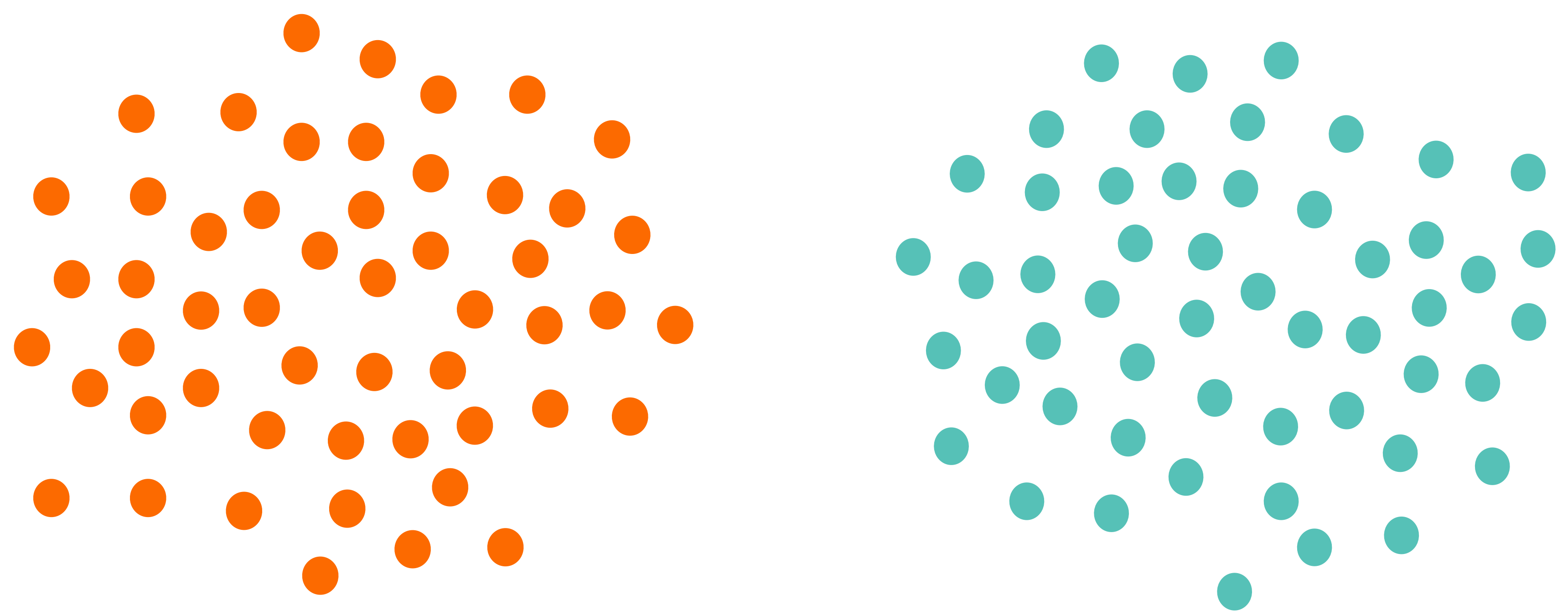
0 vs 1

$P_1 \ll P_0$ $P_1 \gg P_0$

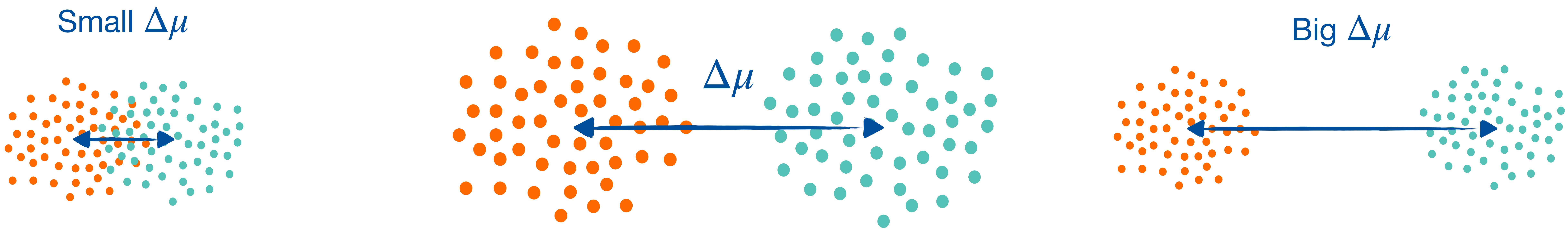


Class Unbalancing

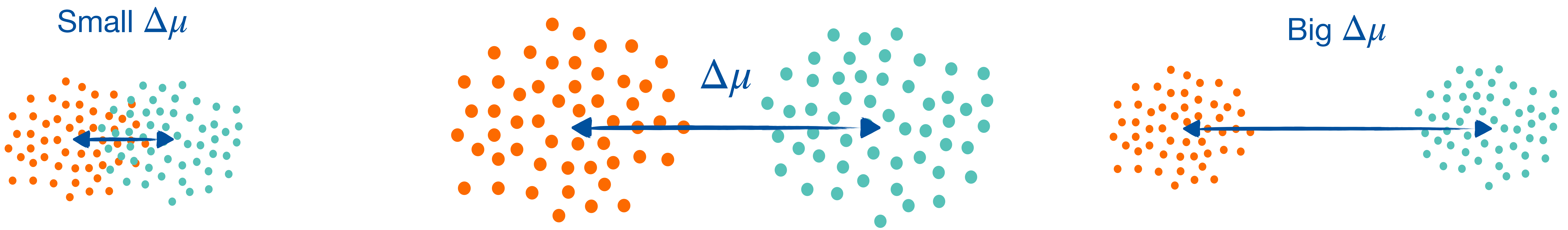
Back to random data

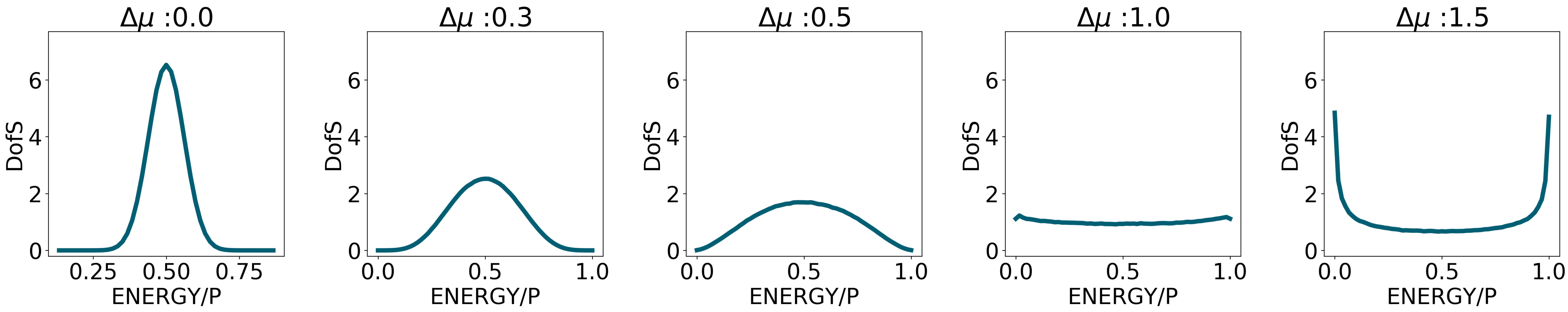


Back to random data

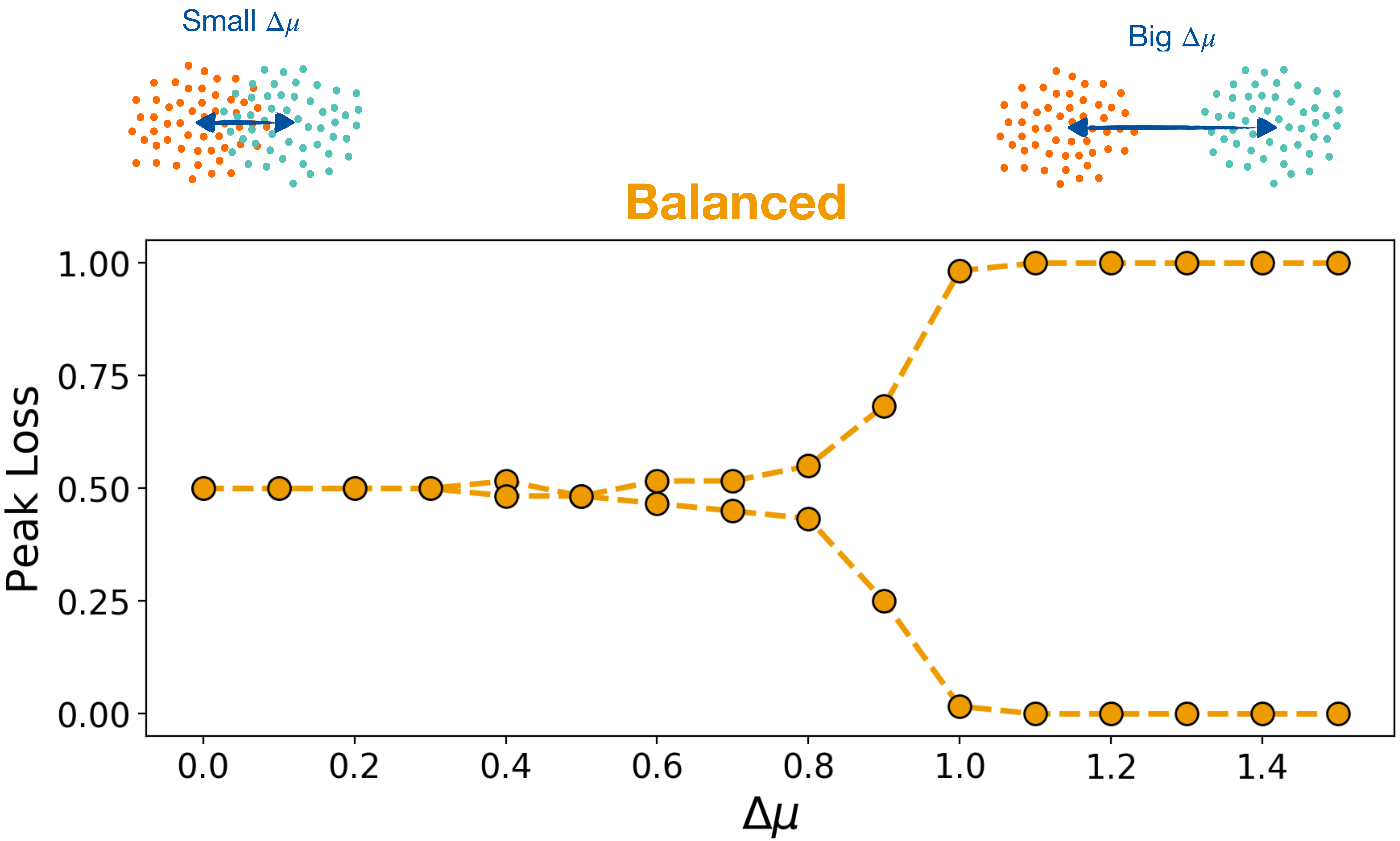


Back to random data

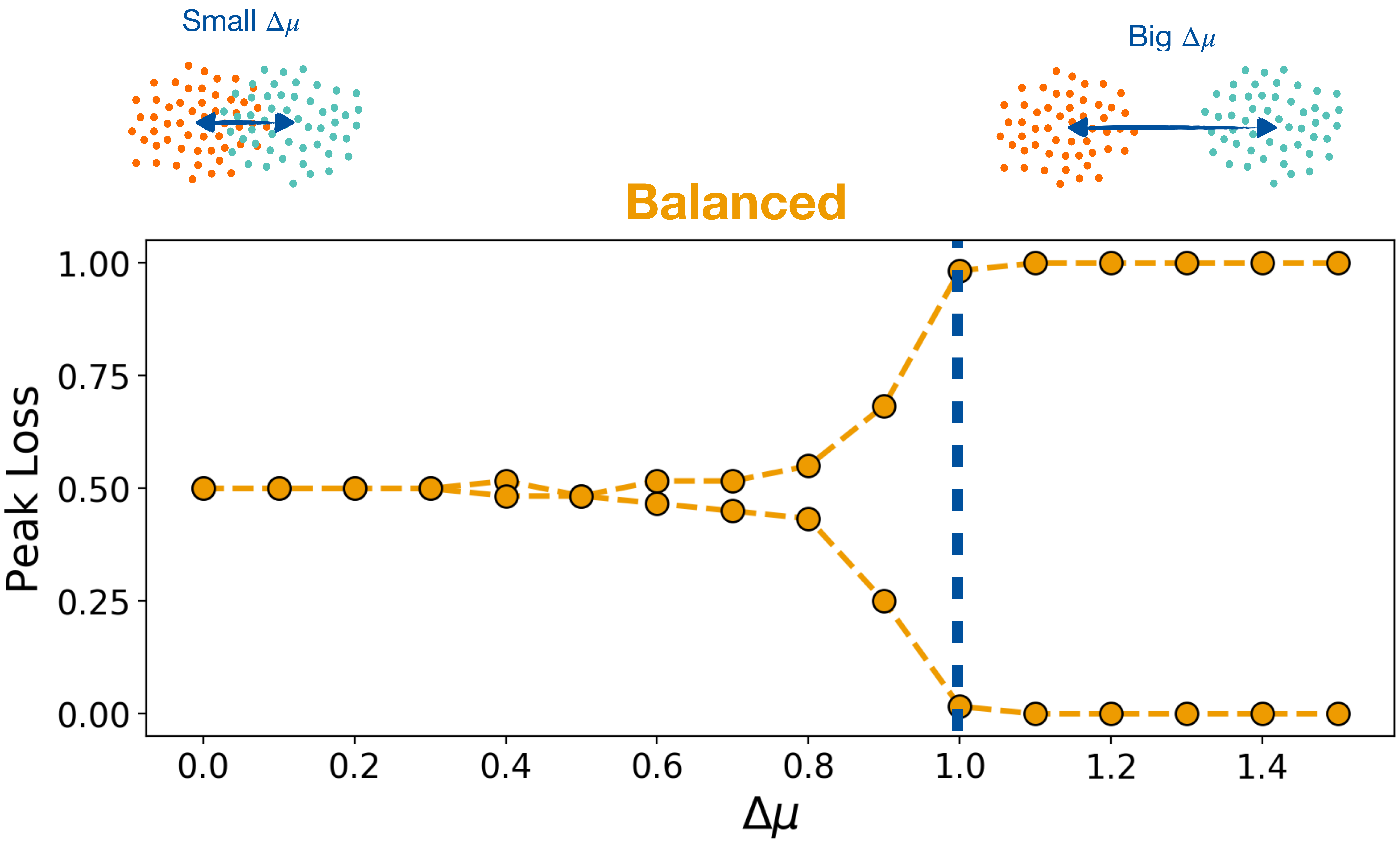




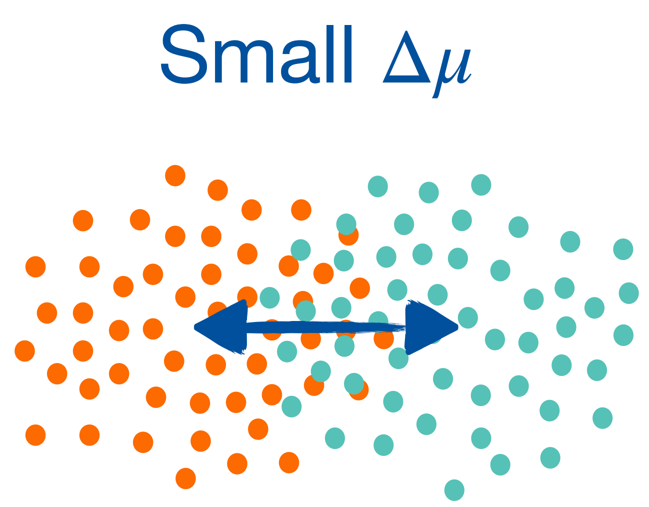
Back to random data

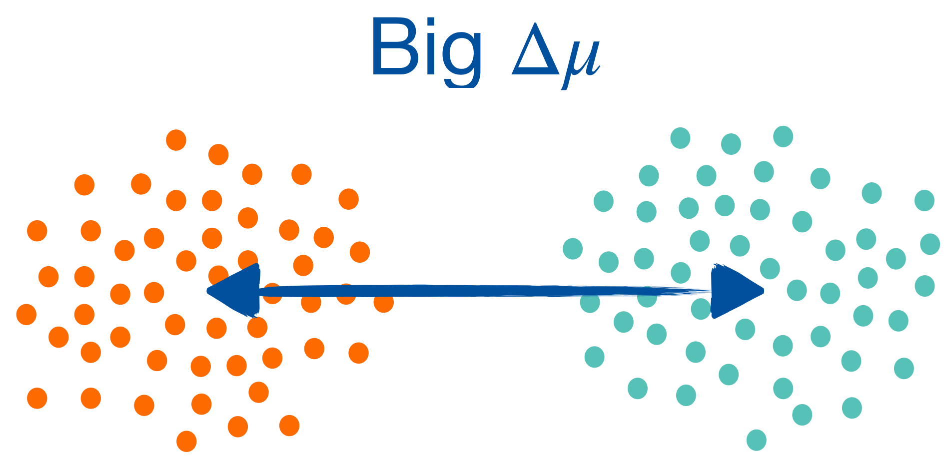


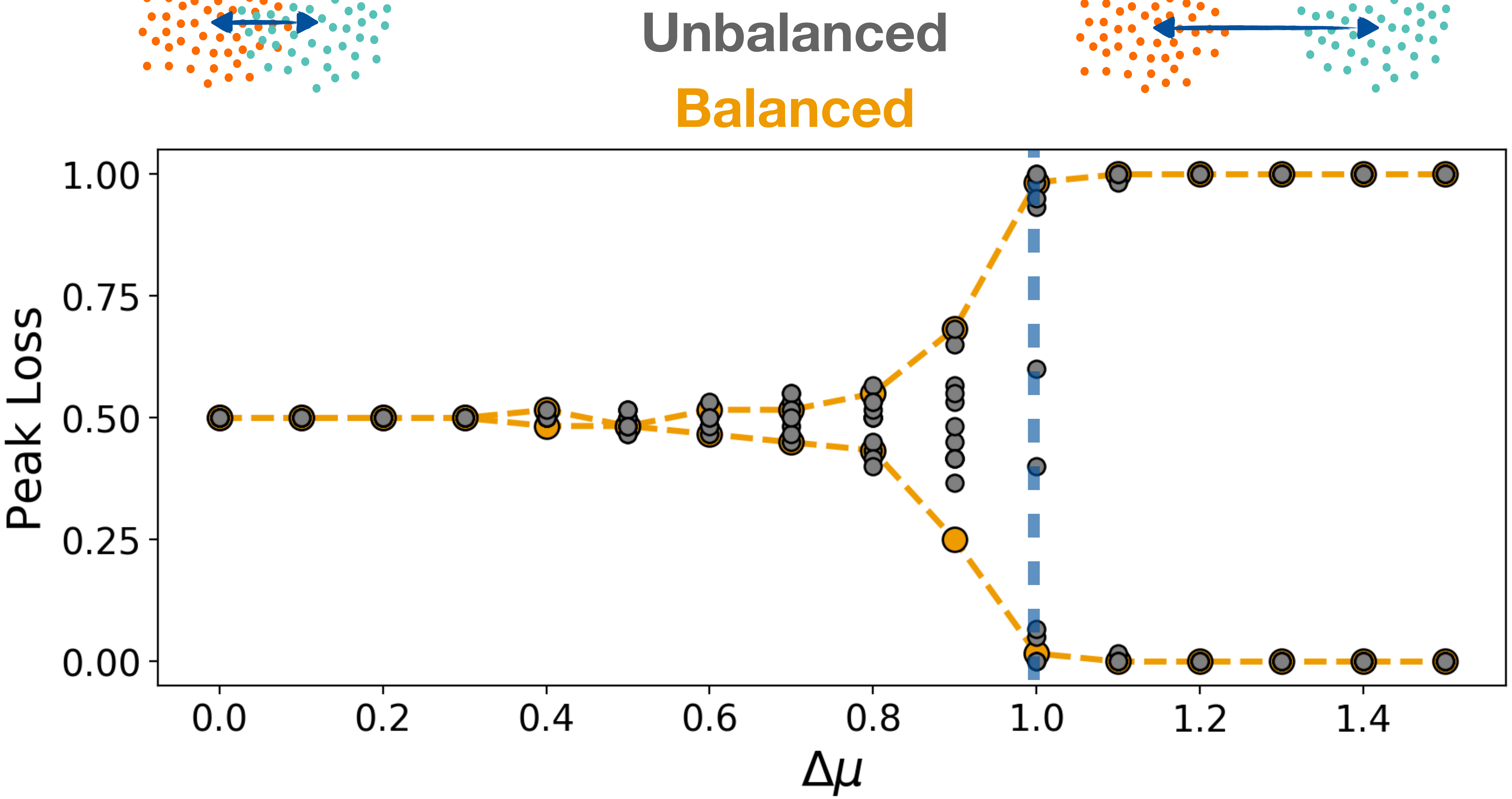
Back to random data



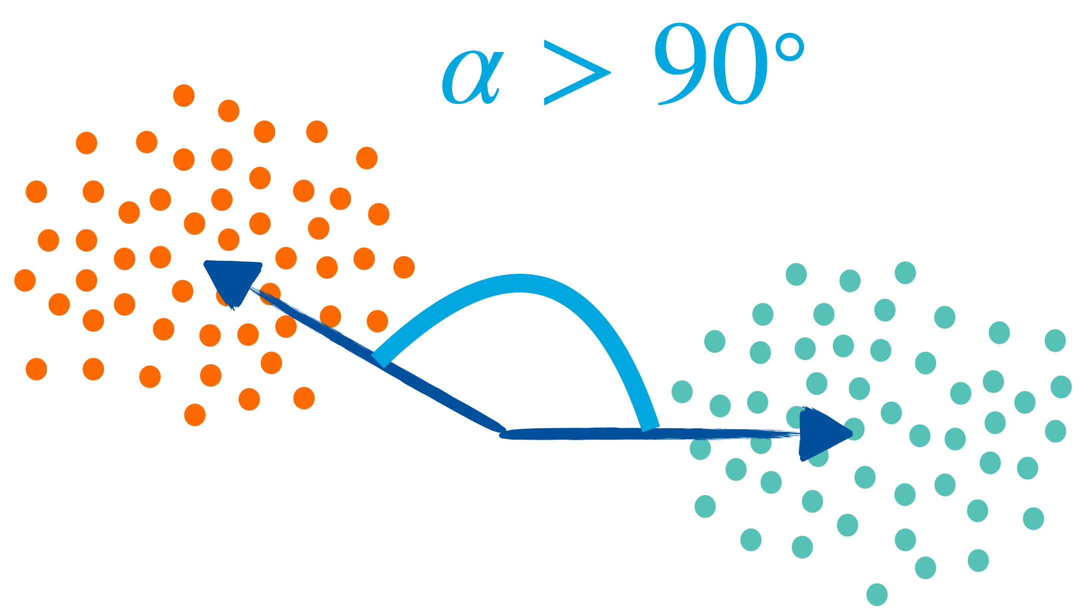
Back to random data

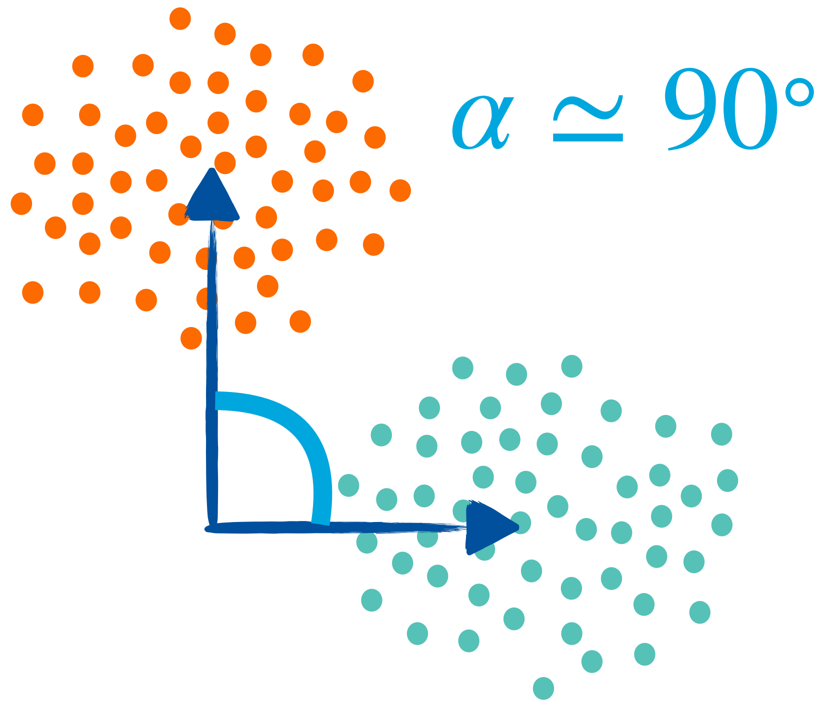


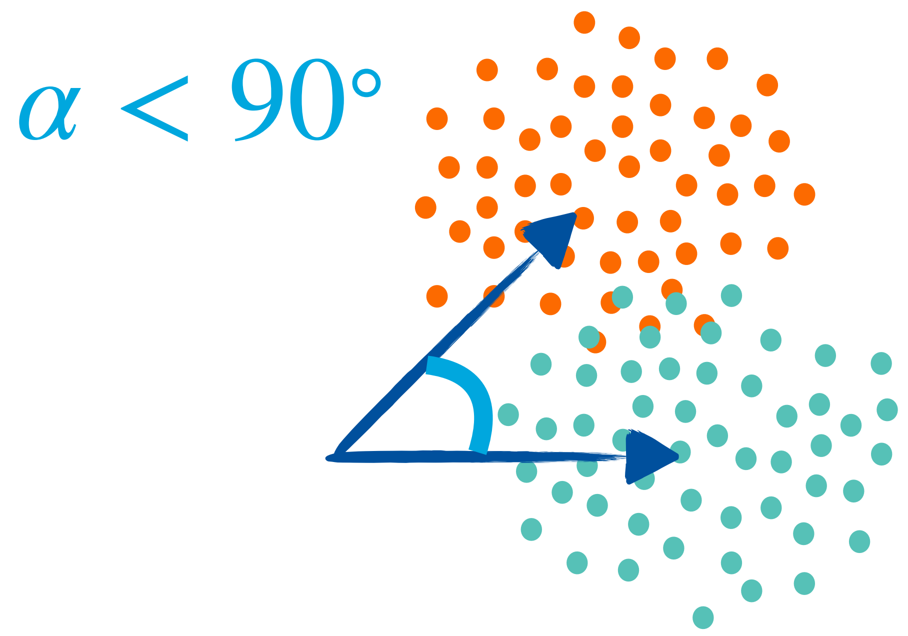




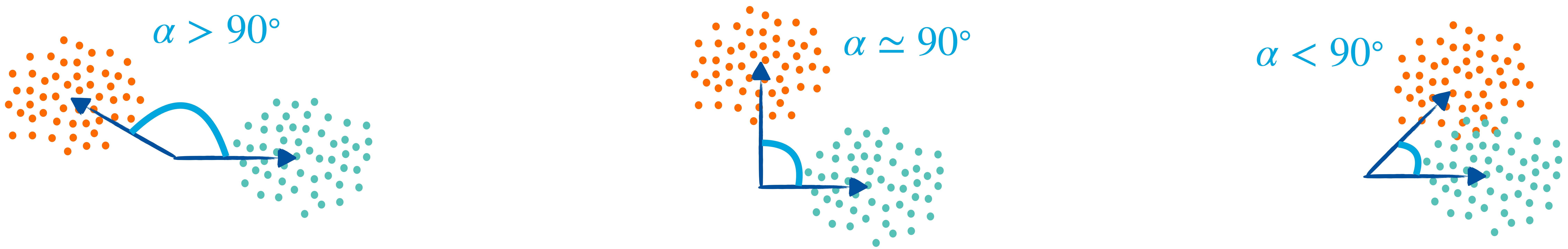
Back to random data



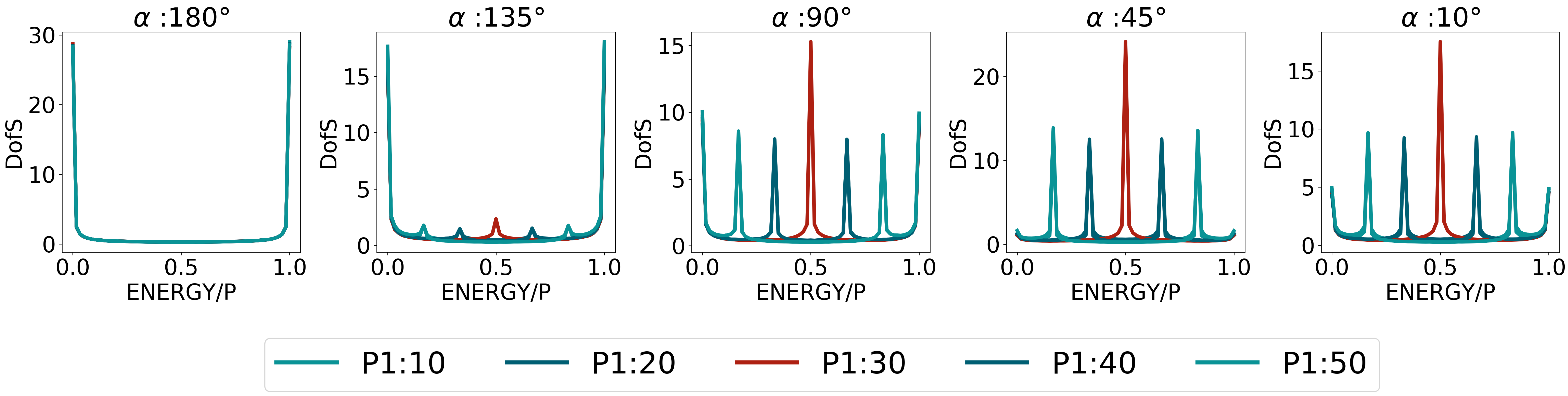




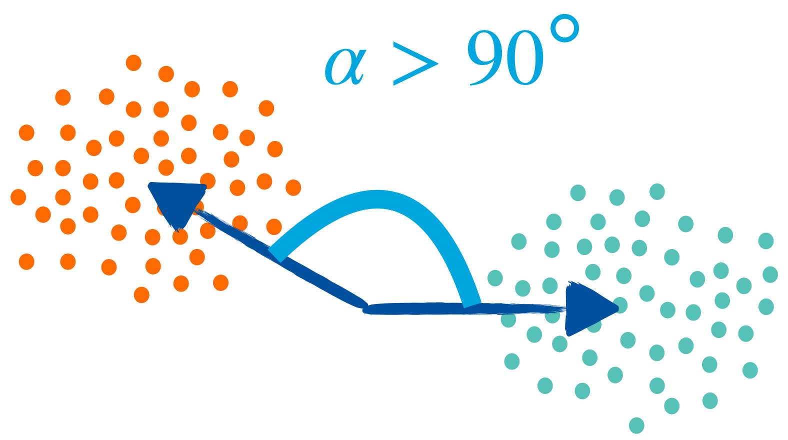
Back to random data

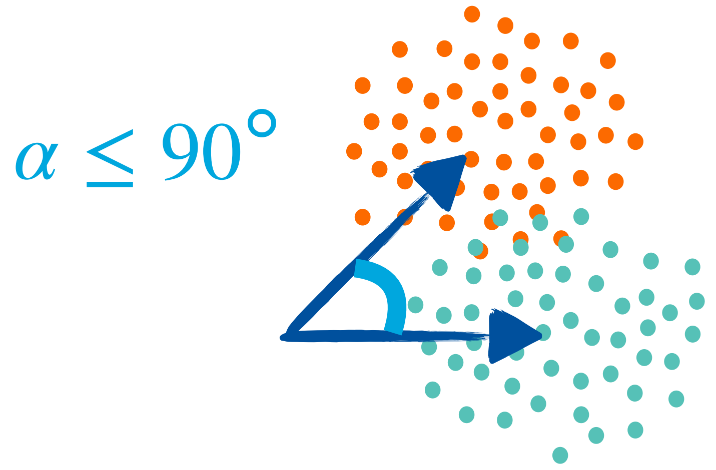


$\Delta\mu \gg 1$

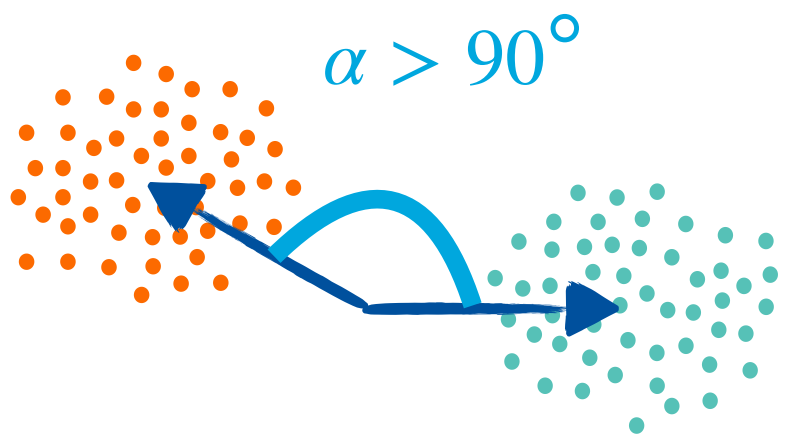


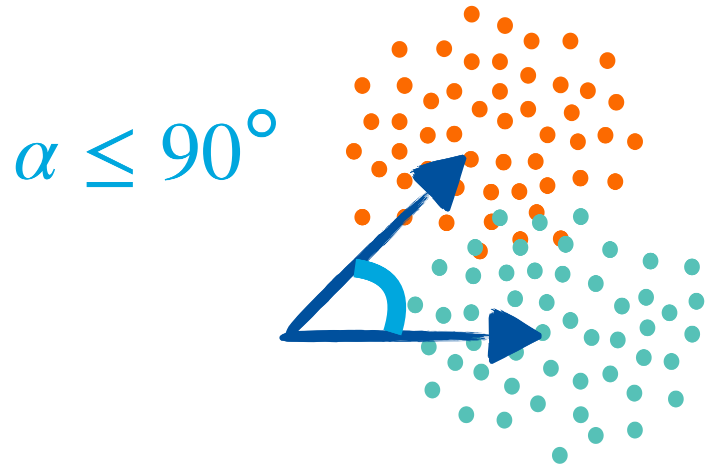
Back to random data

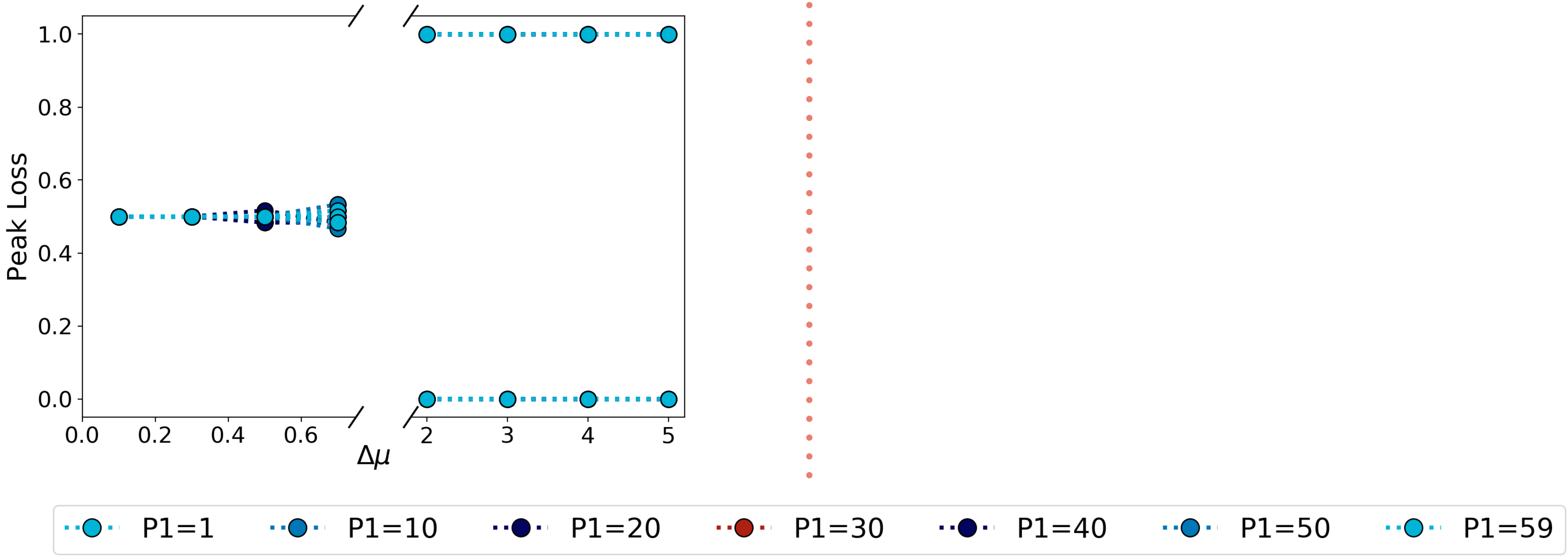




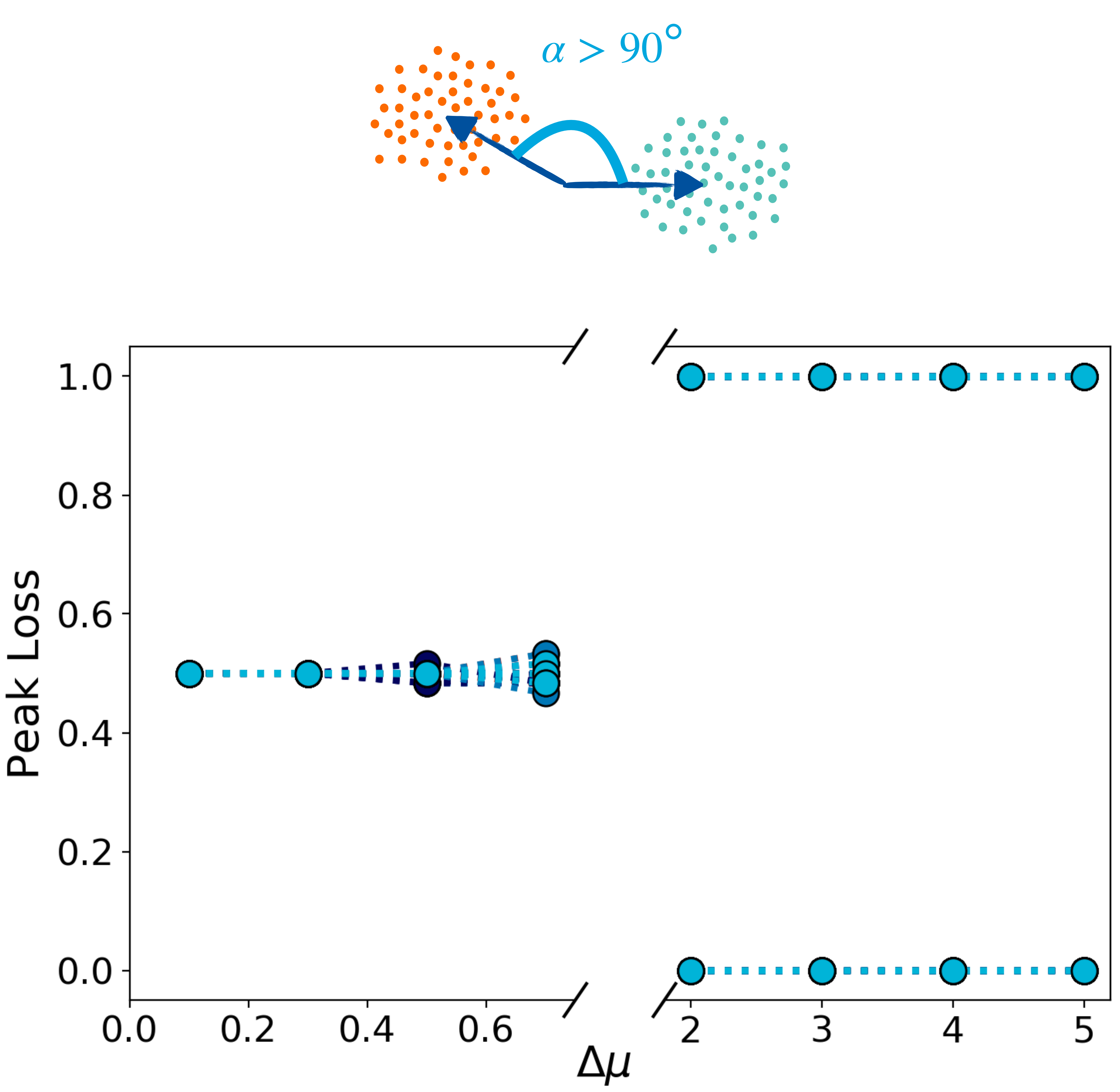
Back to random data

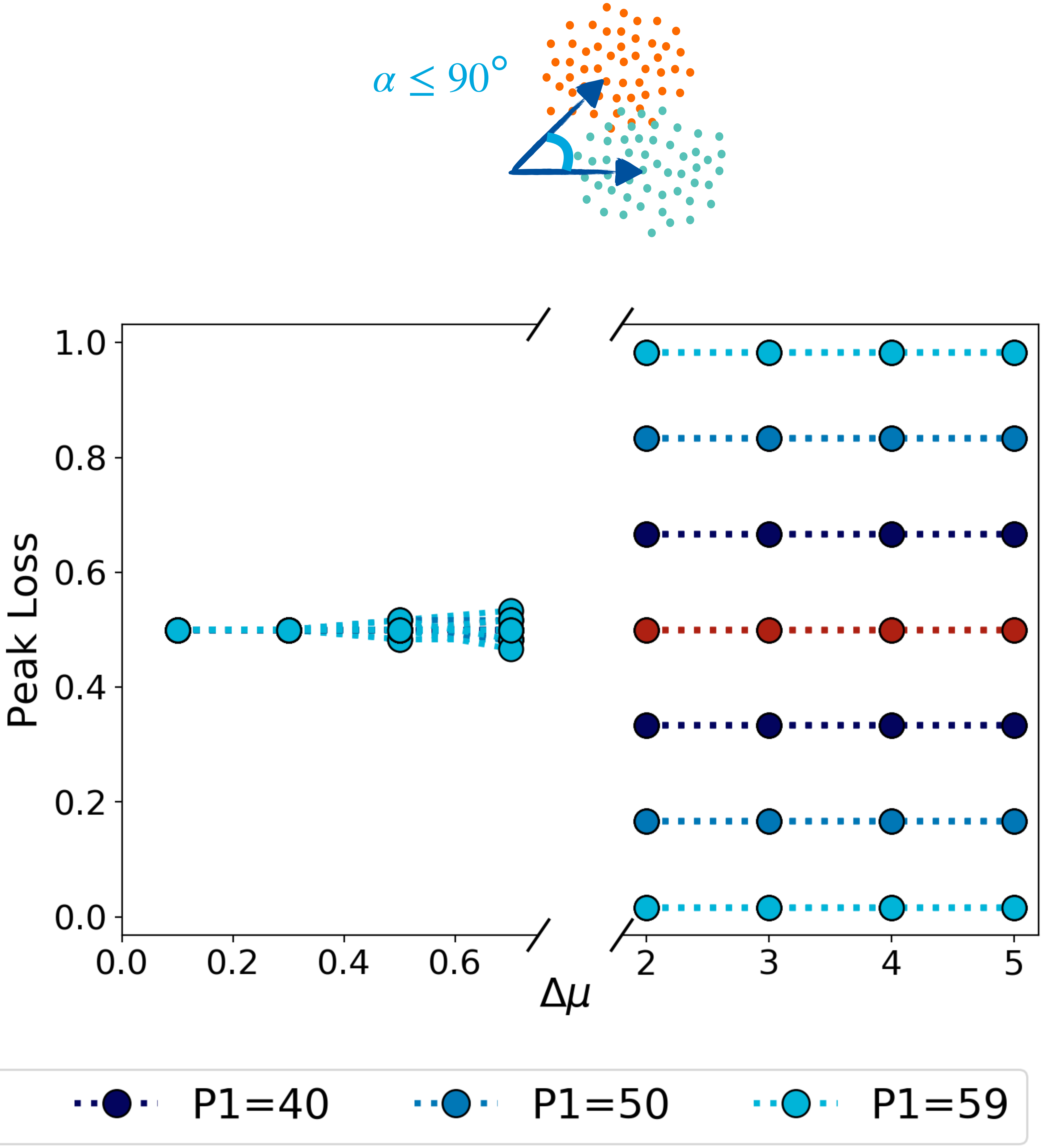




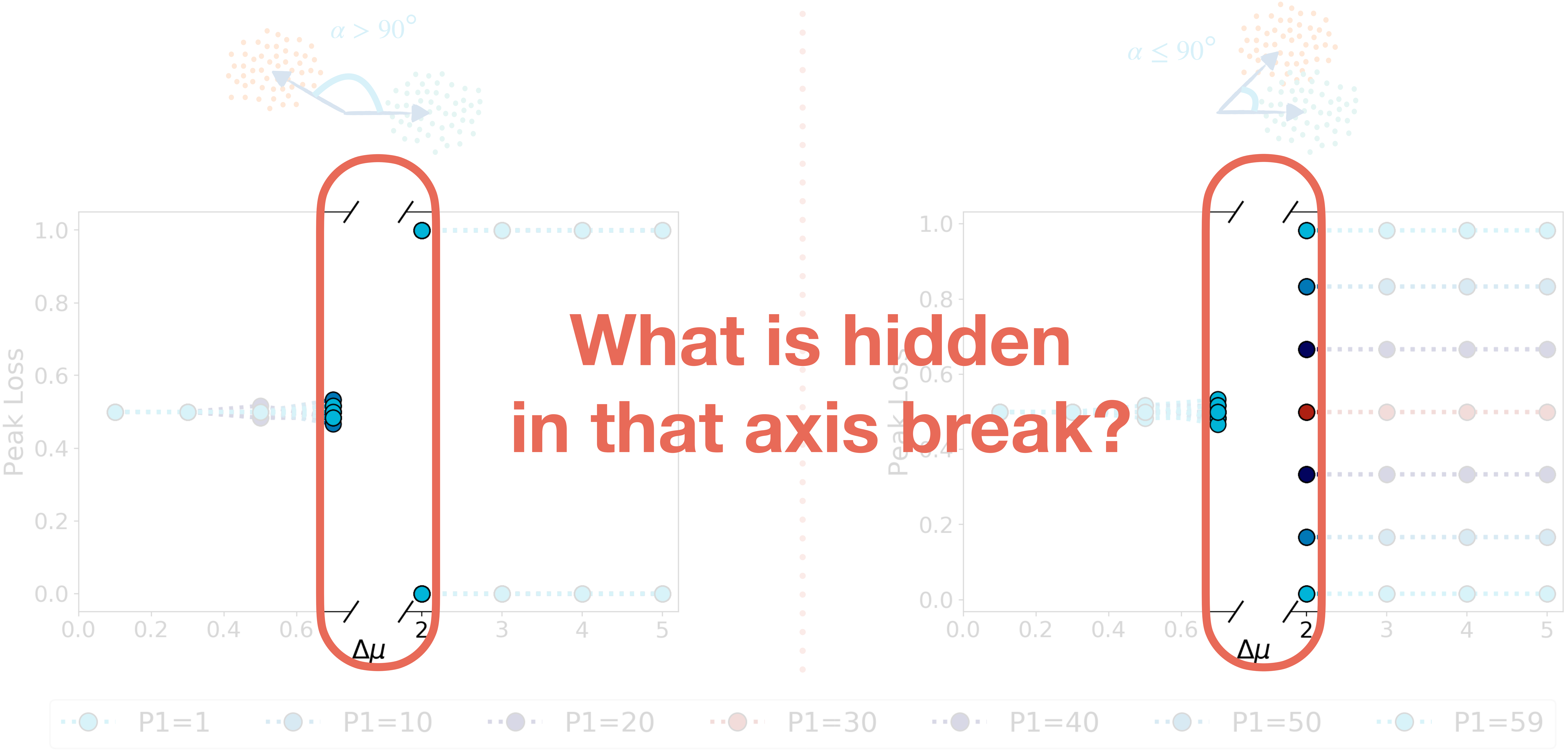


Back to random data



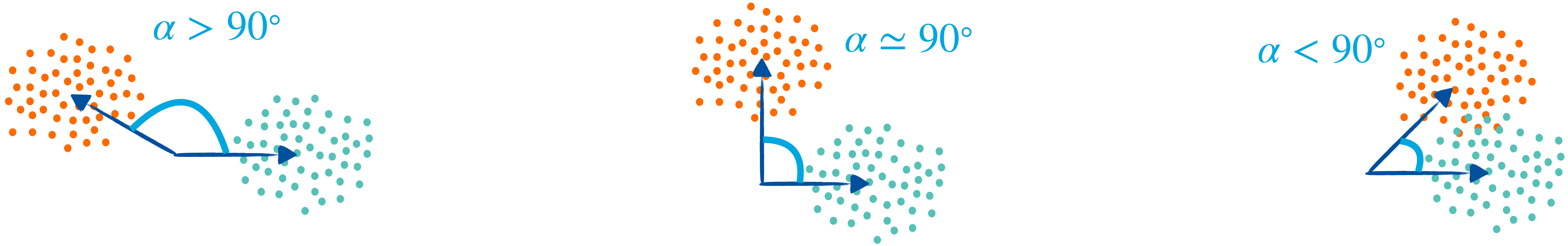


Back to random data

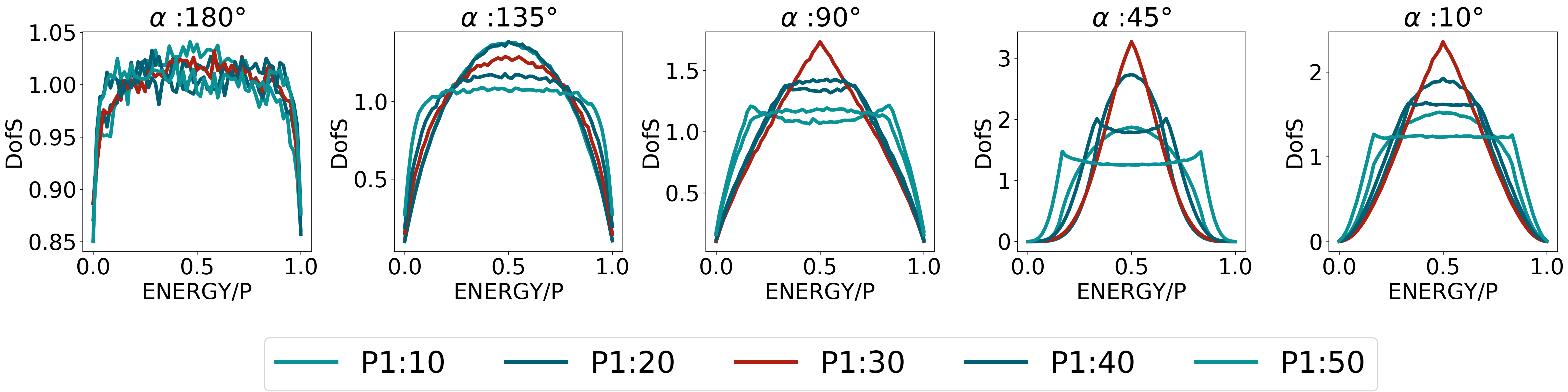


What is hidden in that axis break?

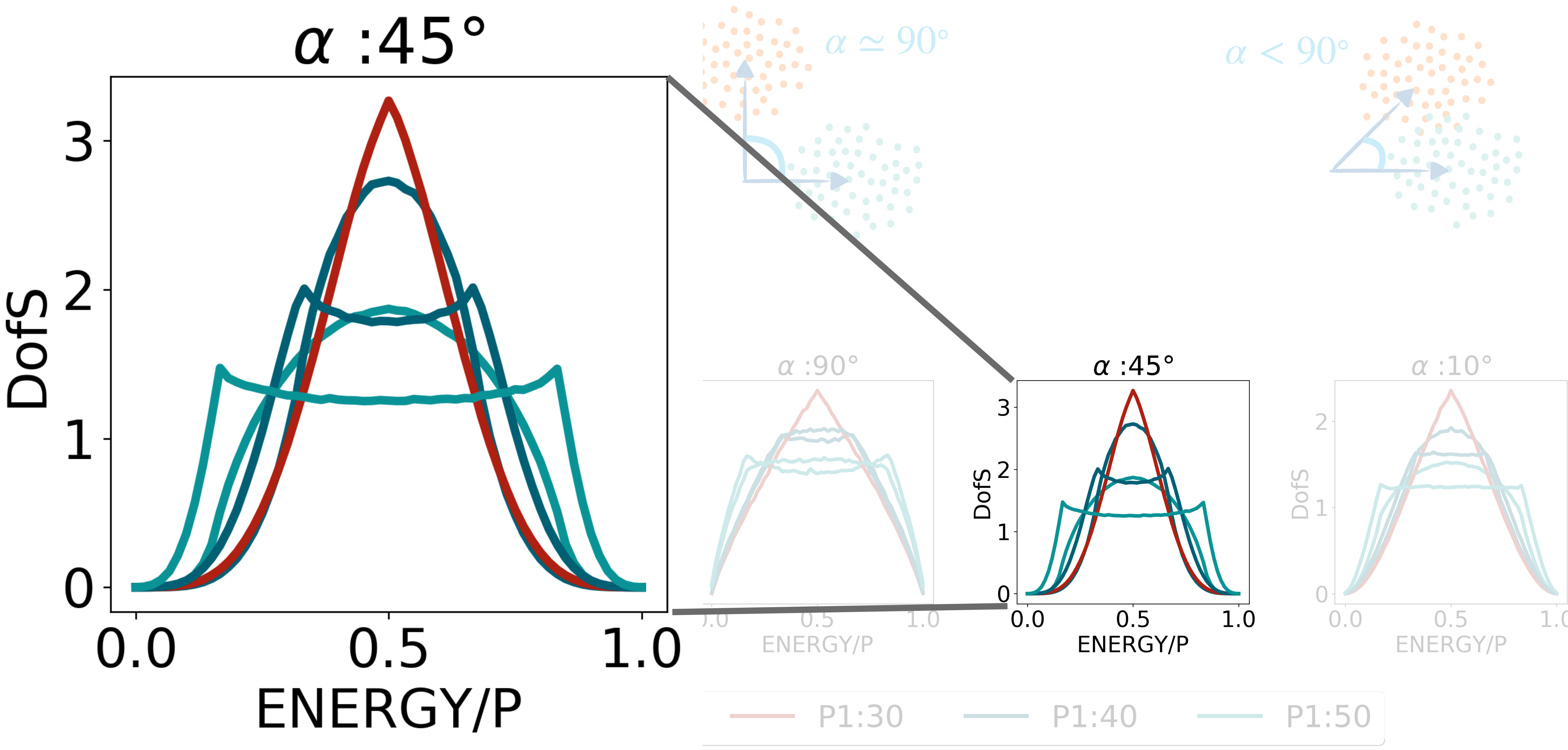
Back to random data



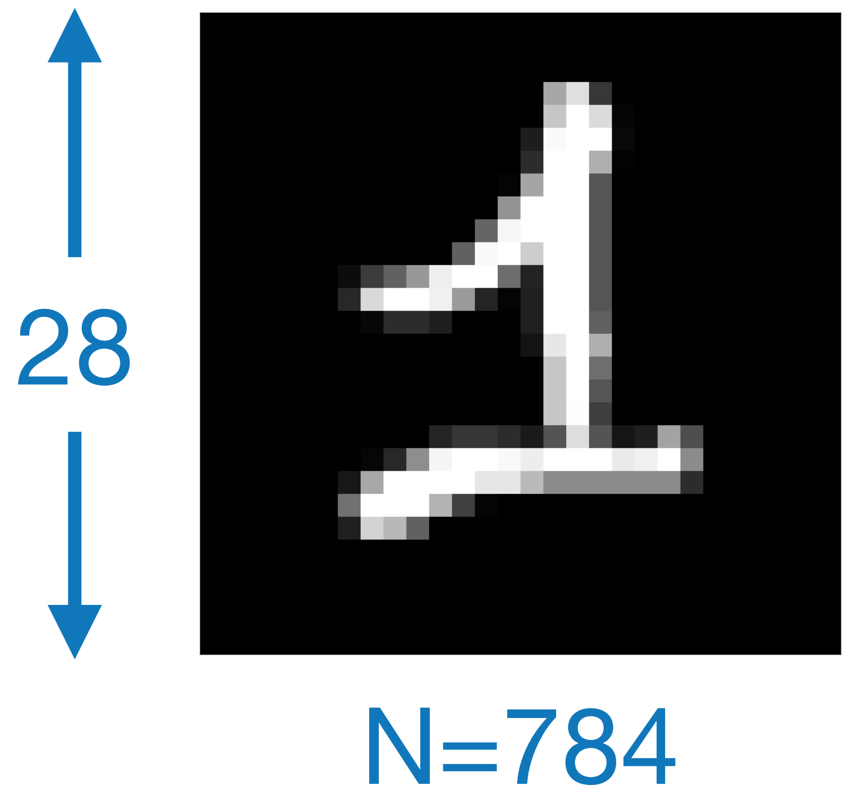
$\Delta\mu \simeq 1$



Back to random data

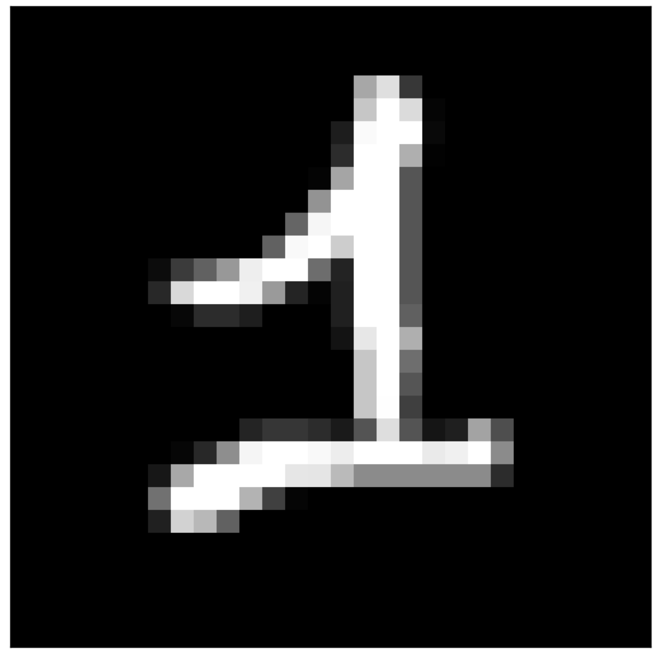


Back back to *realistic* data



Back back to *realistic* data

28

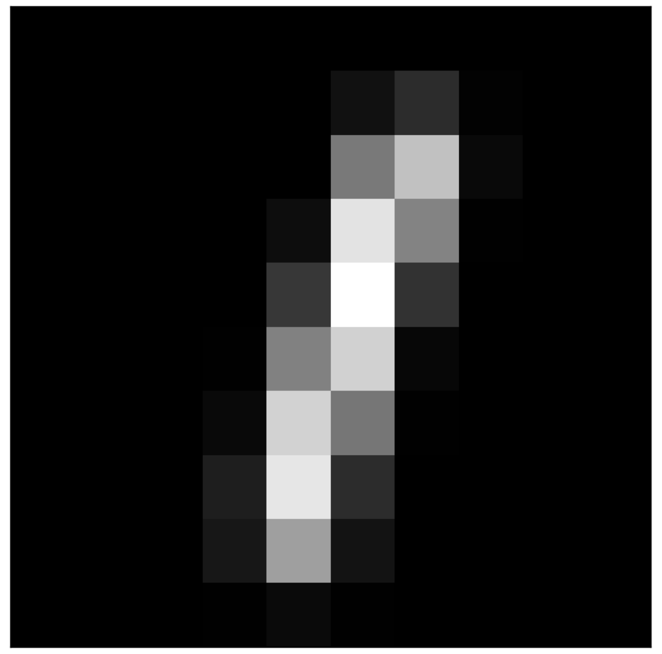


N=784

Reduce Dimensions

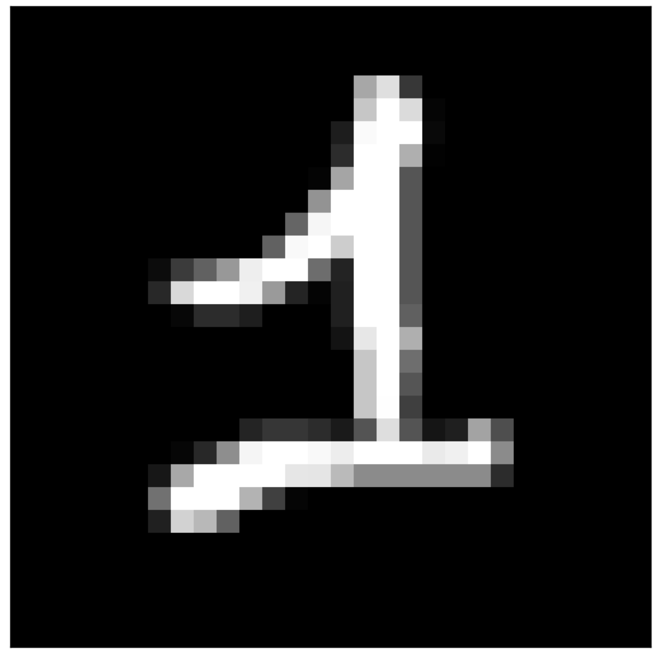
N=100

10



Back back to *realistic* data

28

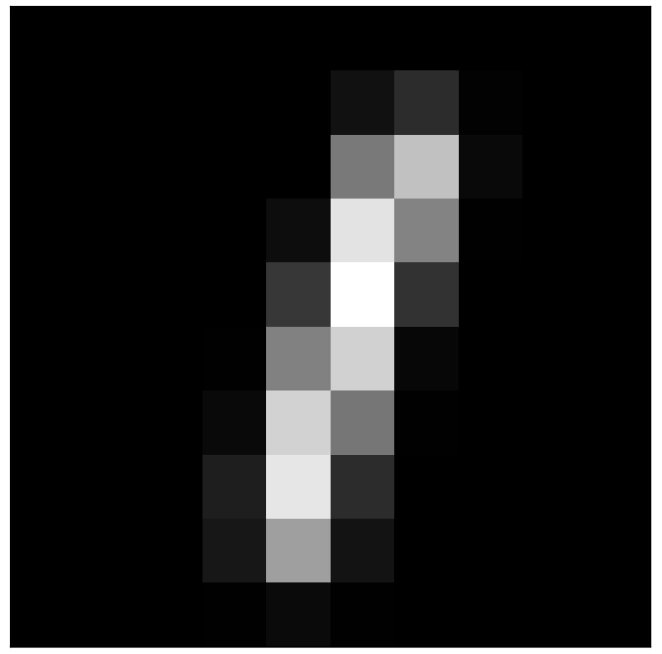


N=784

Reduce Dimensions

N=100

10



Gaussian Clones

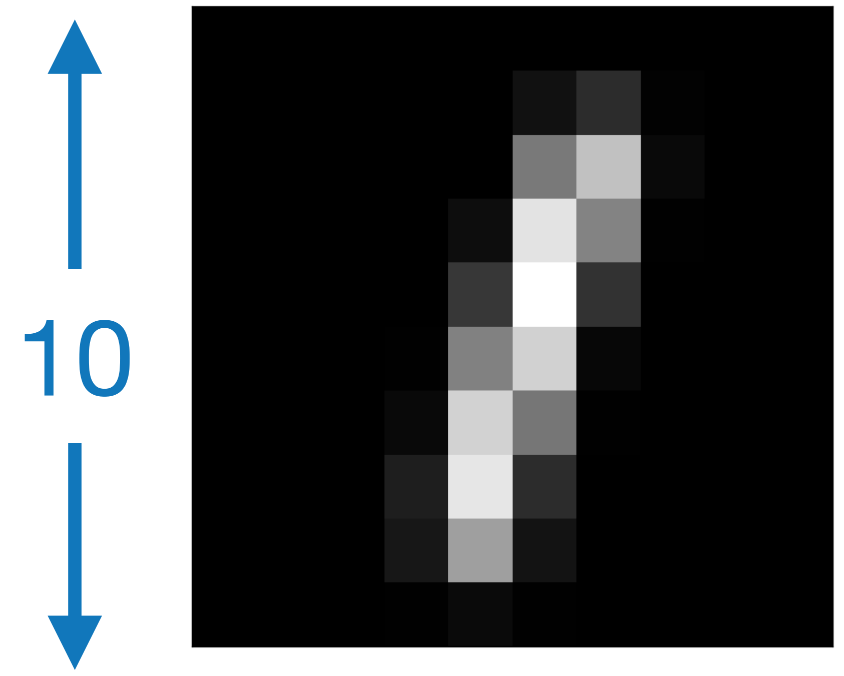
$$N(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T\, \Sigma^{-1}\, (x-\mu)\right)$$

Back back to *realistic* data

N=784

Reduce Dimensions

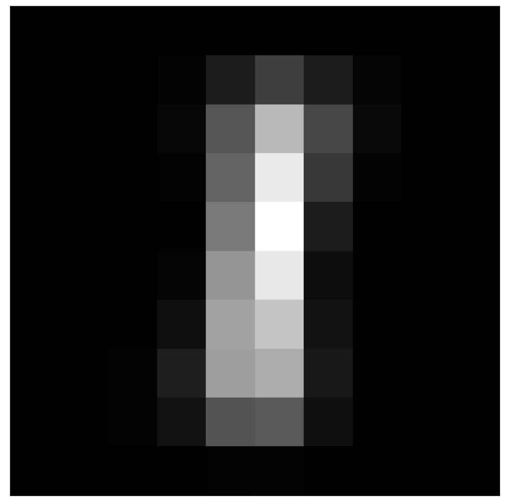
N=100

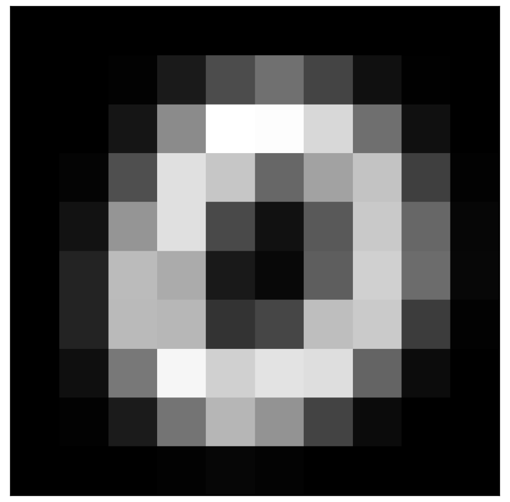


Gaussian Clones

$$N(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)$$

Mean - μ



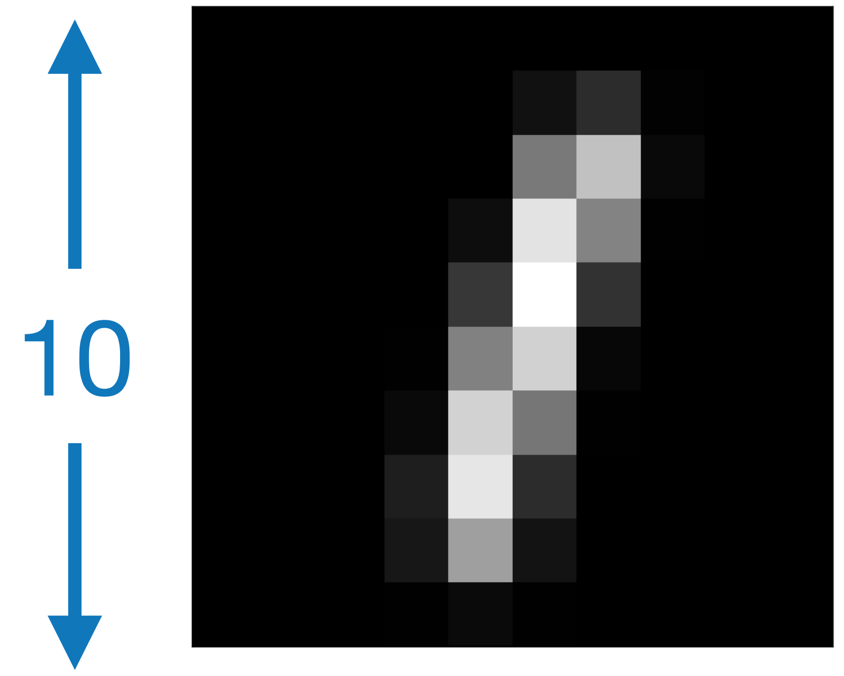


Back back to *realistic* data

N=784

Reduce Dimensions

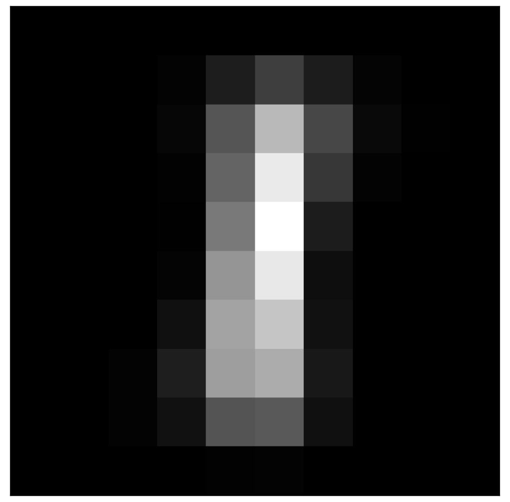
N=100

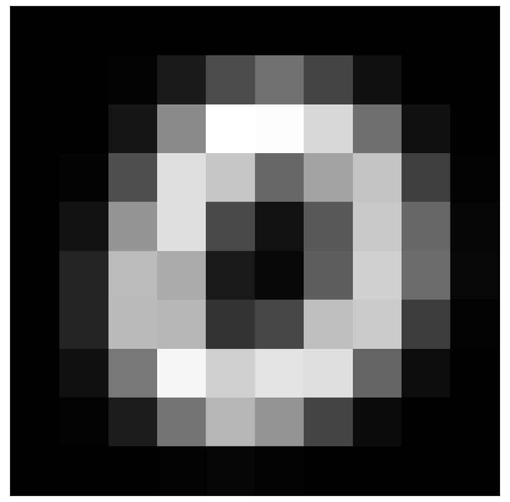


Gaussian Clones

$$N(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T \, \Sigma^{-1} \, (x-\mu)\right)$$

Mean - μ



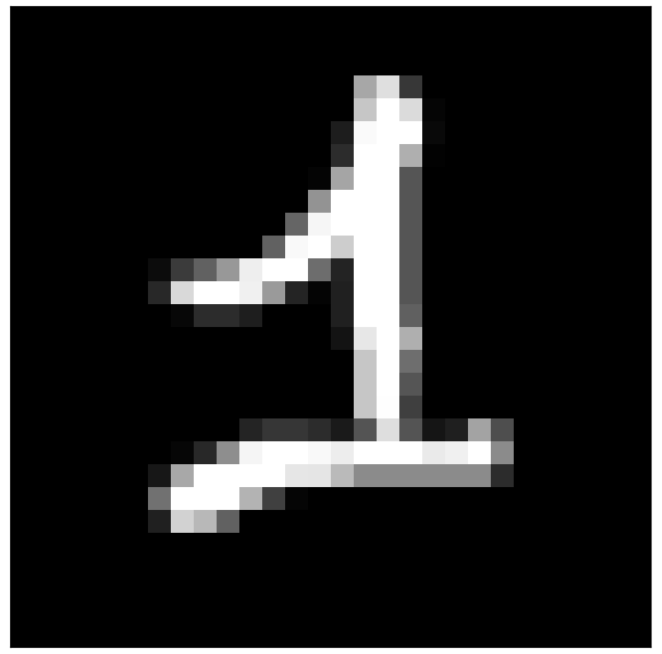


+

Covariance - Σ

Back back to *realistic* data

28

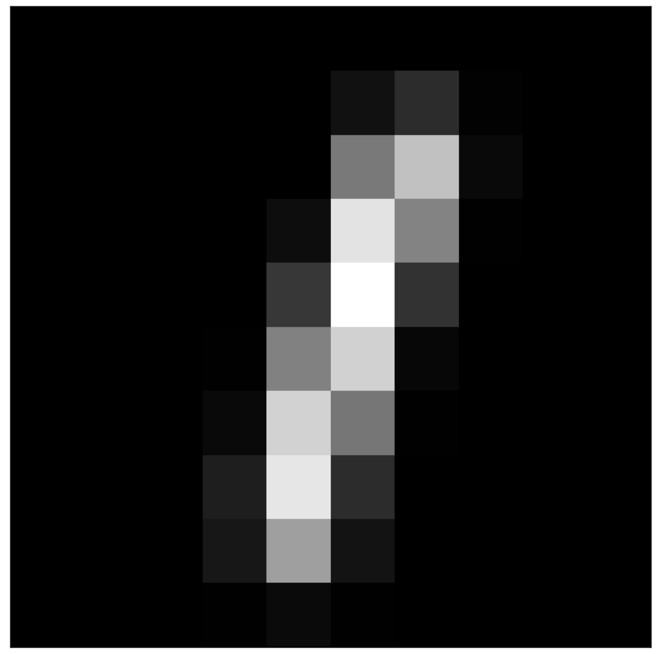


N=784

Reduce Dimensions

N=100

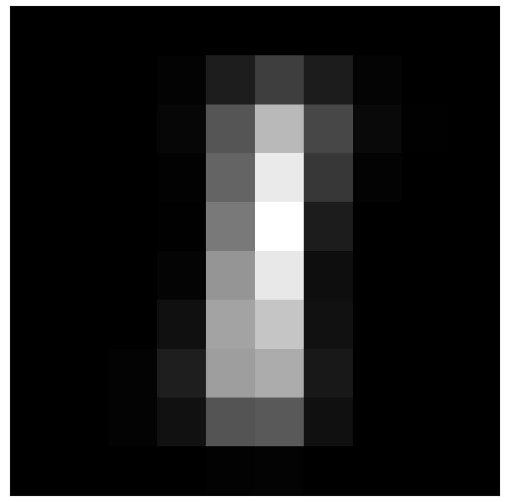
10

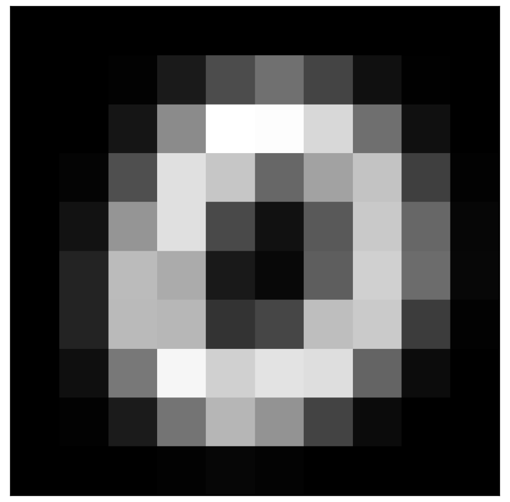


Gaussian Clones

$$N(x,\mu,\Sigma)=\frac{1}{\sqrt{(2\pi)^k\det\Sigma}}\exp\left(-\frac{1}{2}(x-\mu)^T\,\Sigma^{-1}\,(x-\mu)\right)$$

Mean - μ

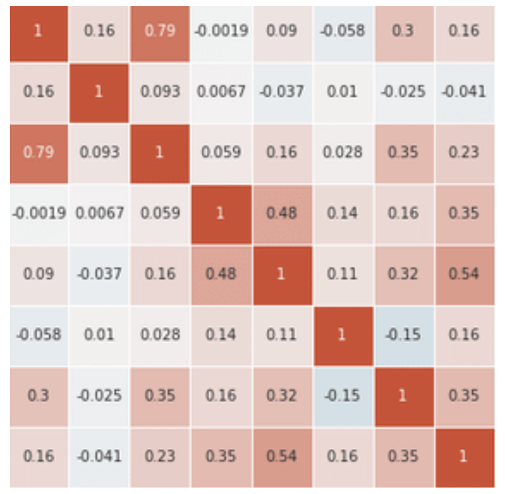


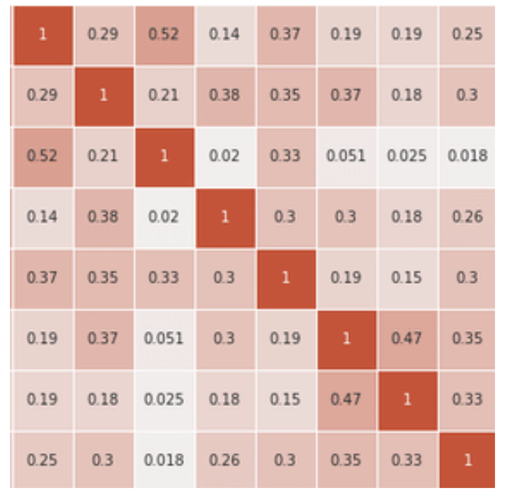


+

Covariance - Σ

GM

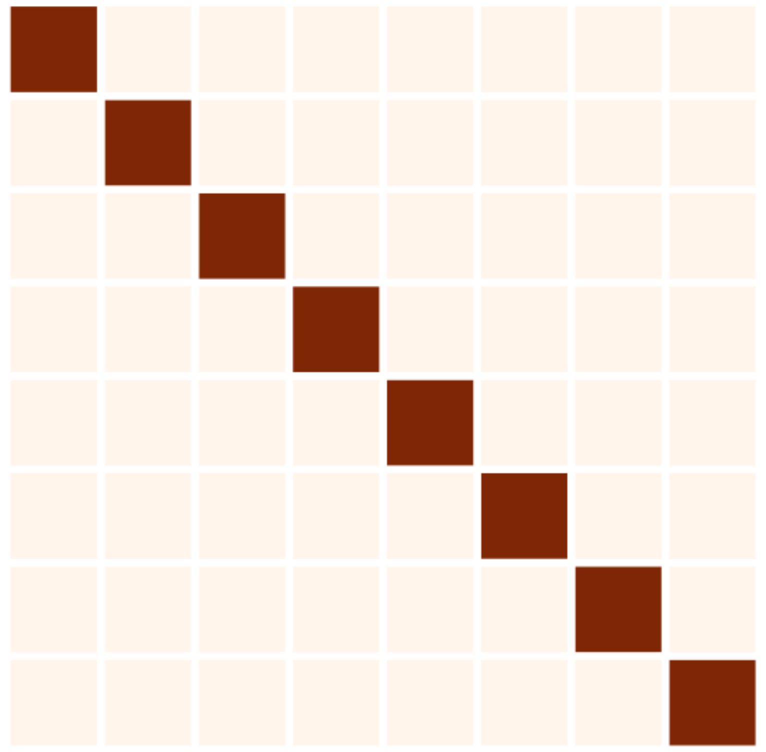




2ISO

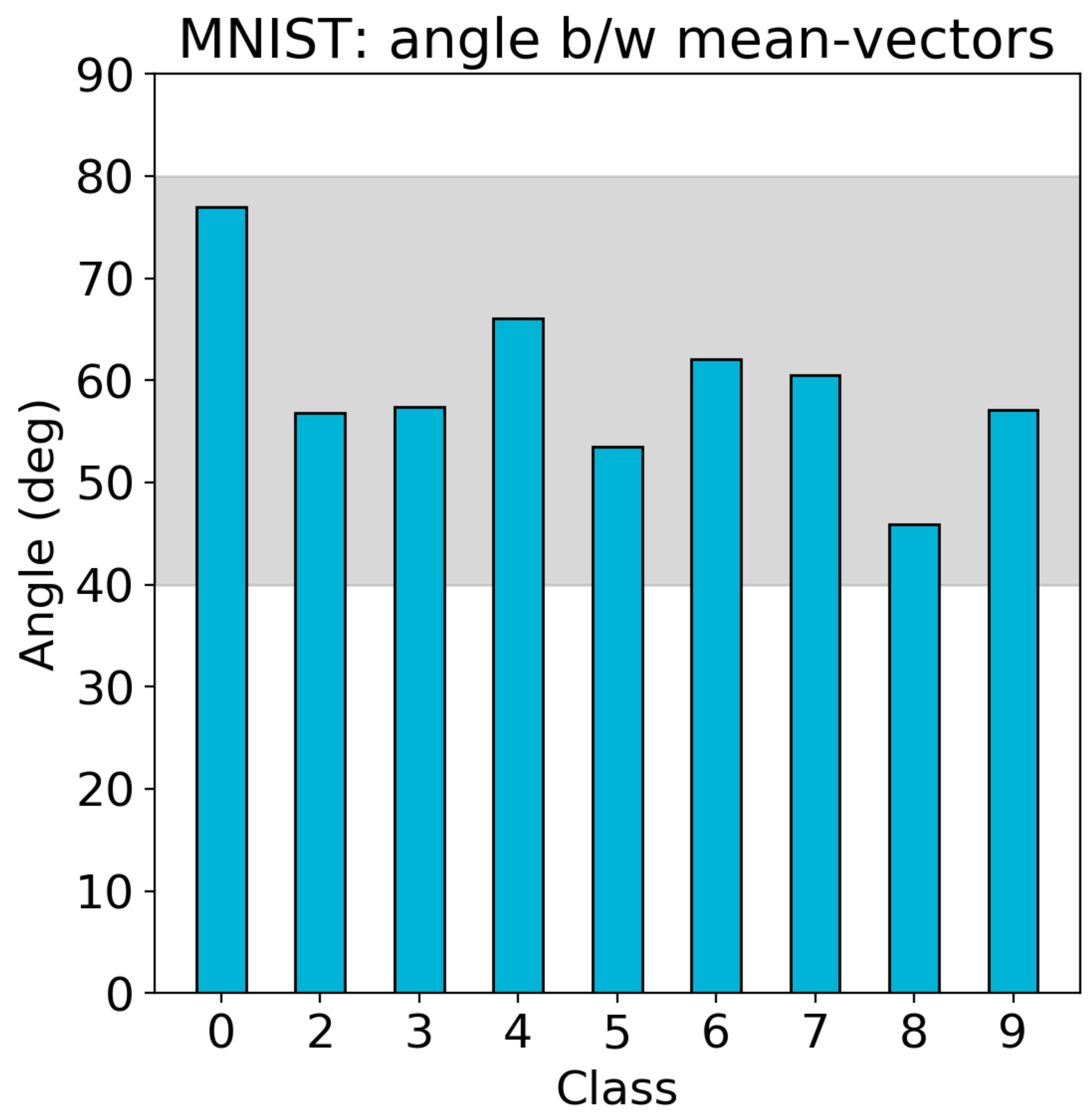
diagonal matrix

$v=\sqrt{v_1\cdot v_2}$



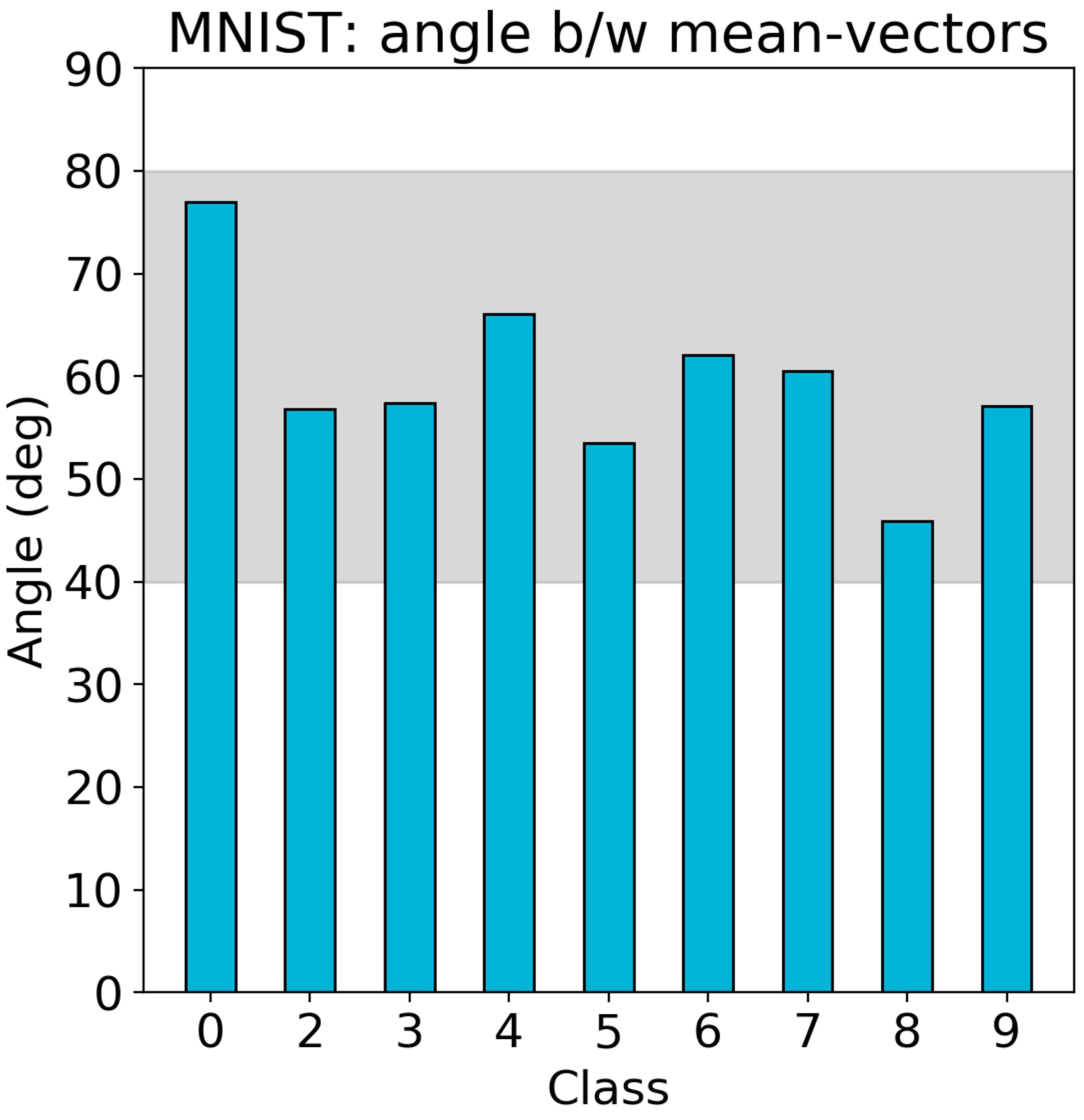
Back back to *realistic* data

1 vs ✱

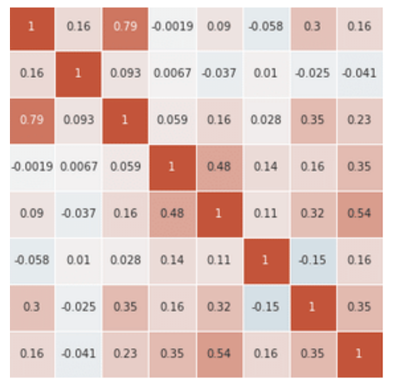


Back back to *realistic* data

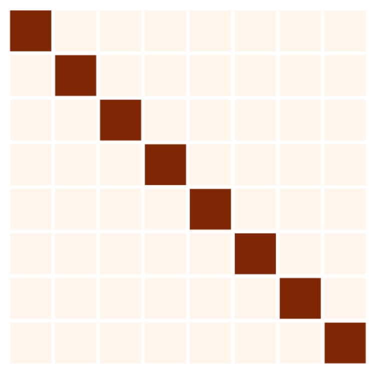
1 vs ✱

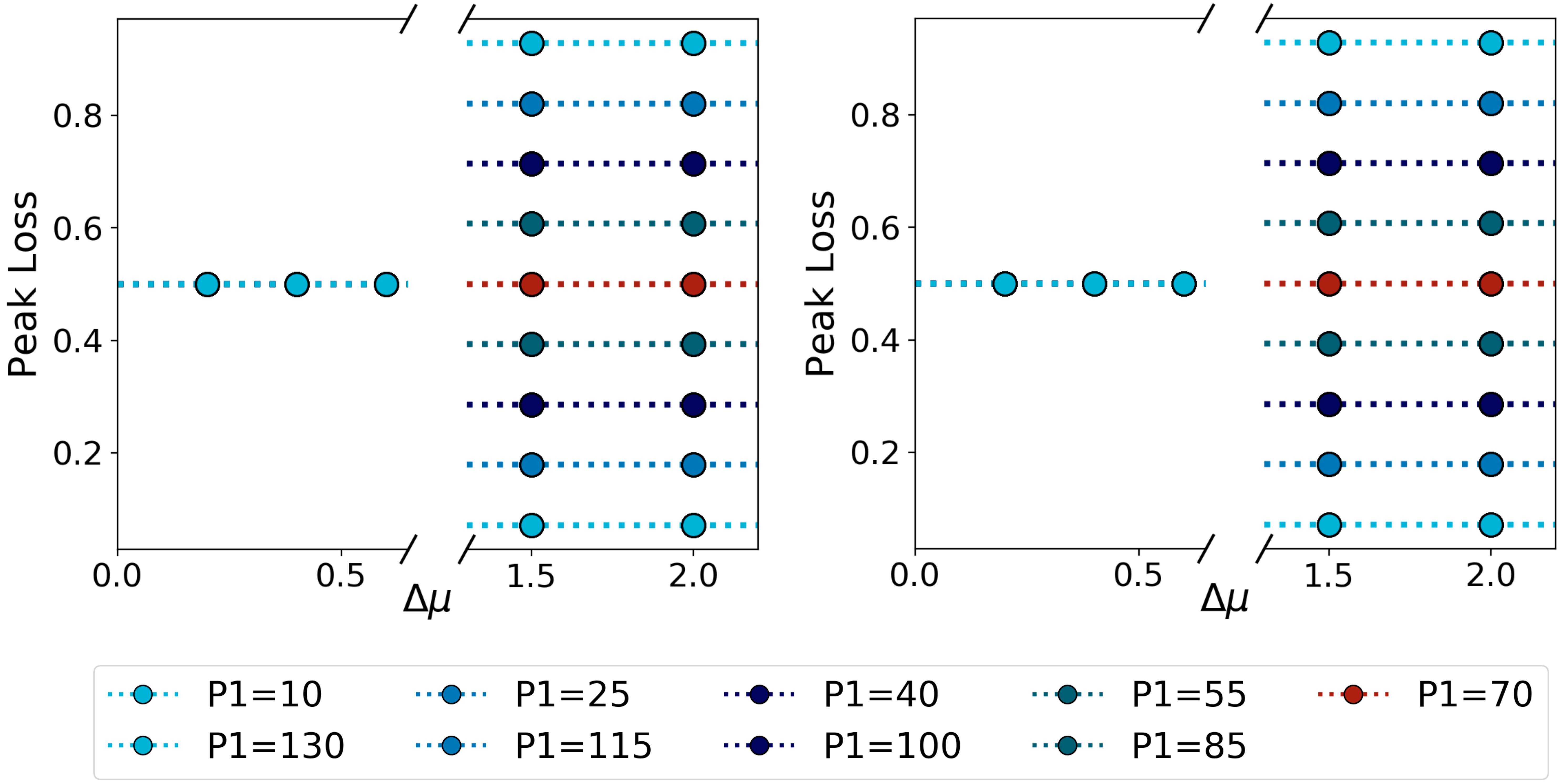


GM



2ISO



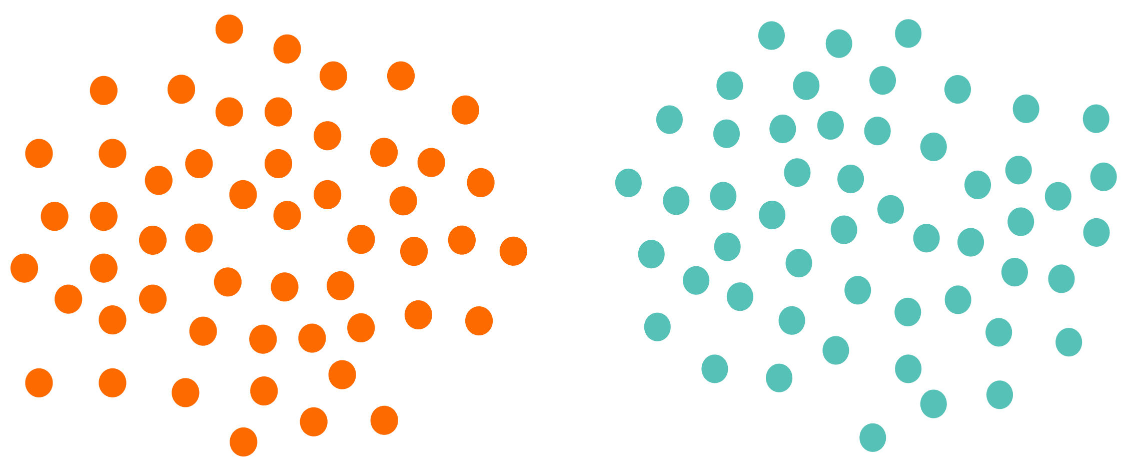


Are there other parameters to control the peaks?

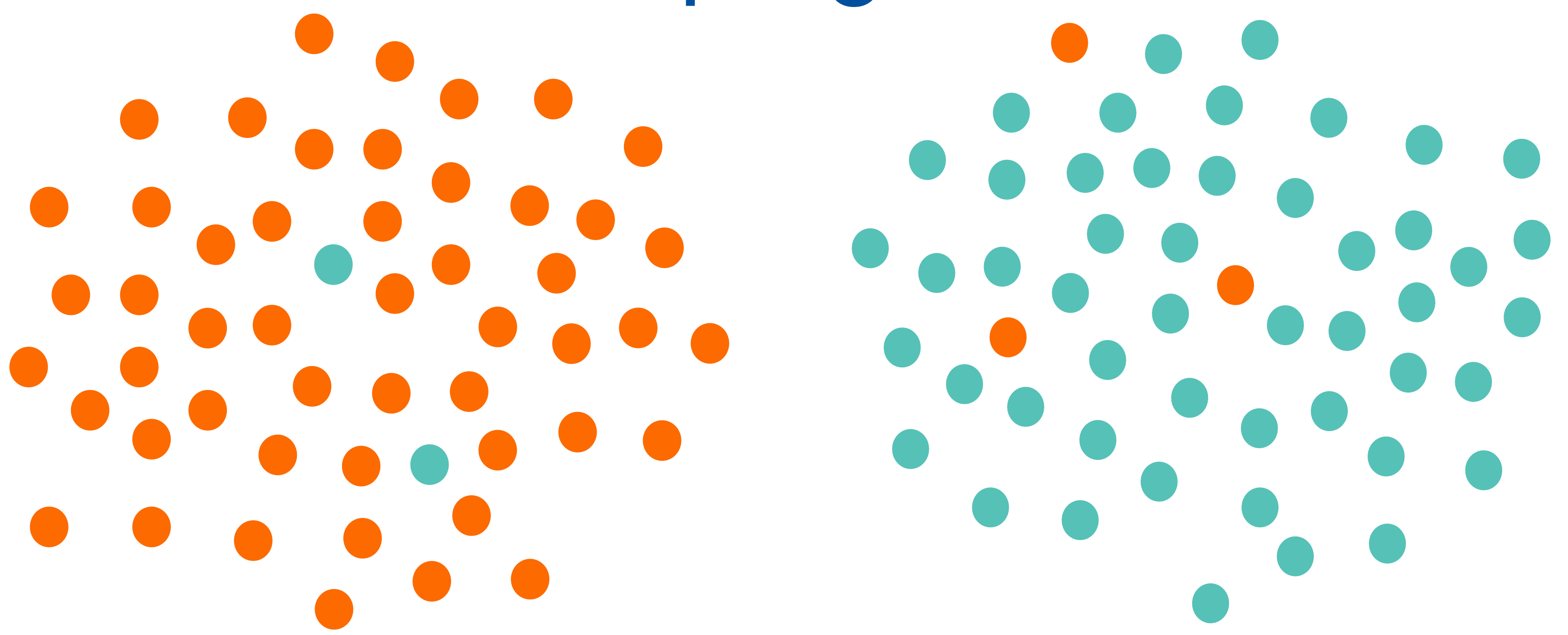
Are there other parameters to control the peaks?

MISLABELLING

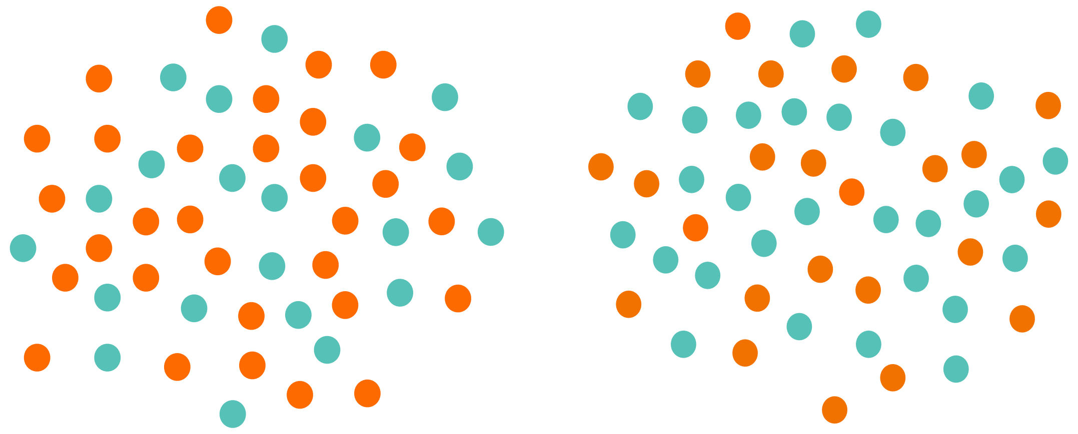
f = 0



f = 5

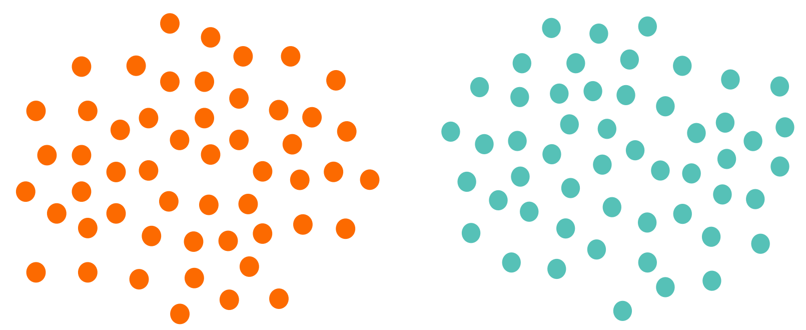


f = P/2

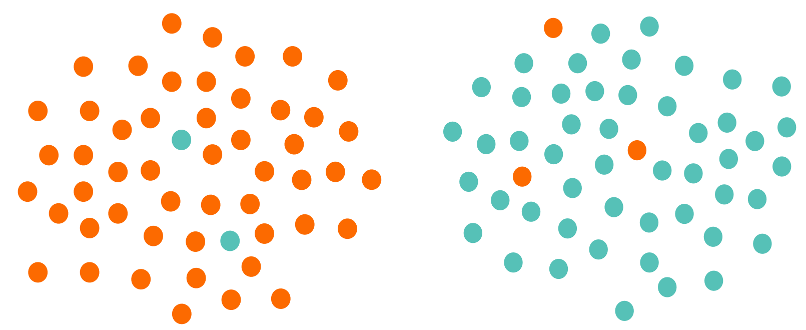


Are there other parameters to control the peaks?

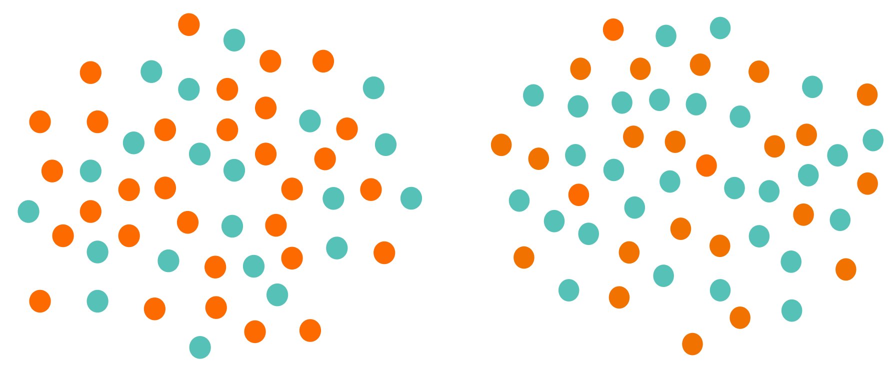
f = 0

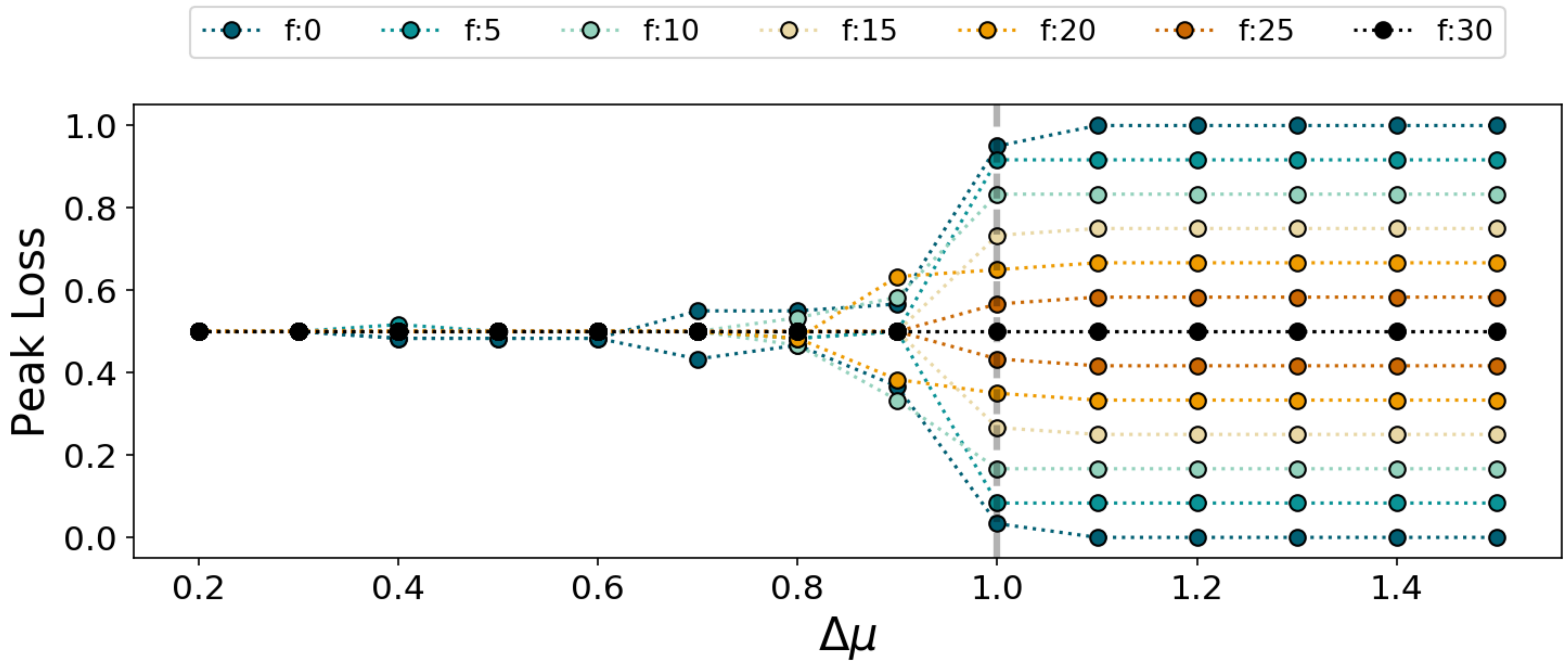


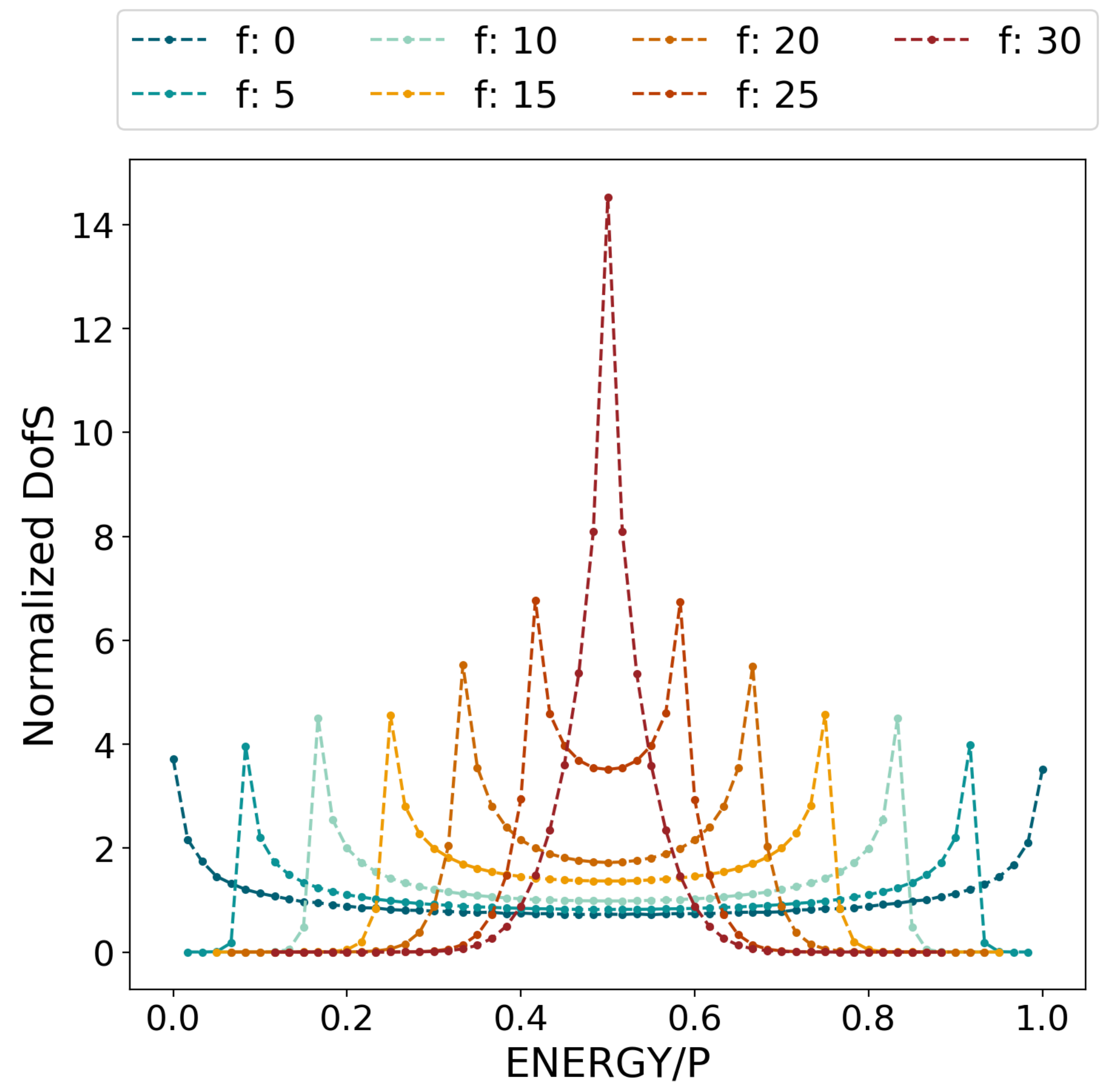
f = 5



f = P/2







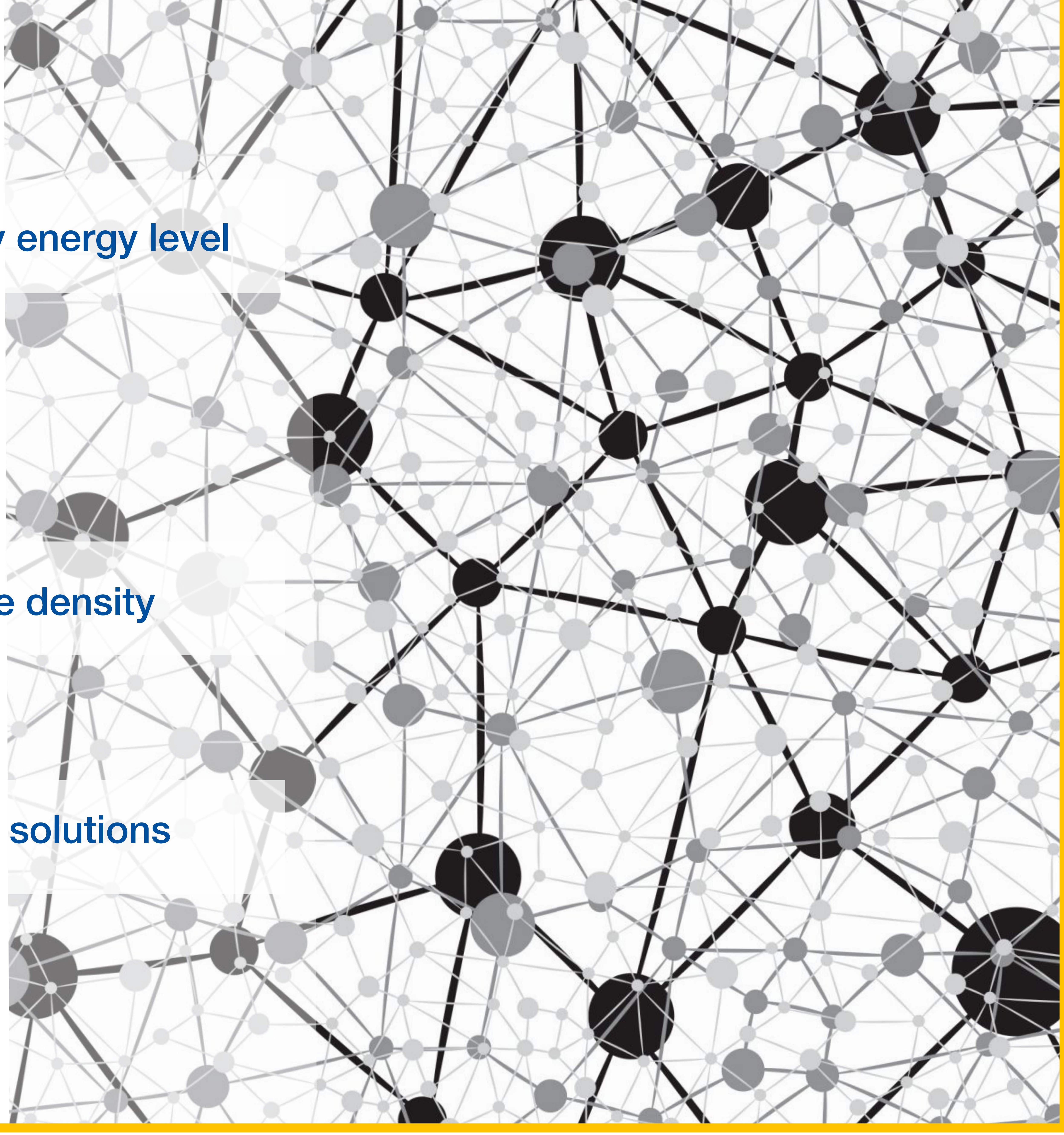
Conclusions

- Thermodynamic characterisation of the system at any energy level

- Possibility to study real learning problems

- Input-output correlation structure directly impacts the density of states of learning problems

- Insight into the learning process due to a large pull of solutions and higher energy states

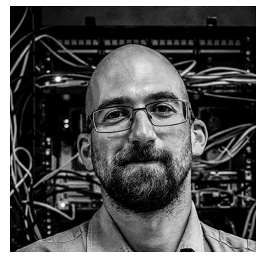


Conclusions

- Thermodynamic characterisation of the system at any energy level
- Possibility to study real learning problems
- Input-output correlation structure directly impacts the density of states of learning problems
- Insight into the learning process due to a large pull of solutions and higher energy states

Statistical and Biological Physics group
University of Trento - TIFPA

Raffaello Potestio
University of Trento - TIFPA



Roberto Menichetti
University of Trento - TIFPA

Alessandro Ingrosso
International Centre of Theoretical Physics

UNIVERSITÀ DI TRENTO

Istituto Nazionale di Fisica Nucleare
TIFPA
Trento Institute for Fundamental Physics and Applications

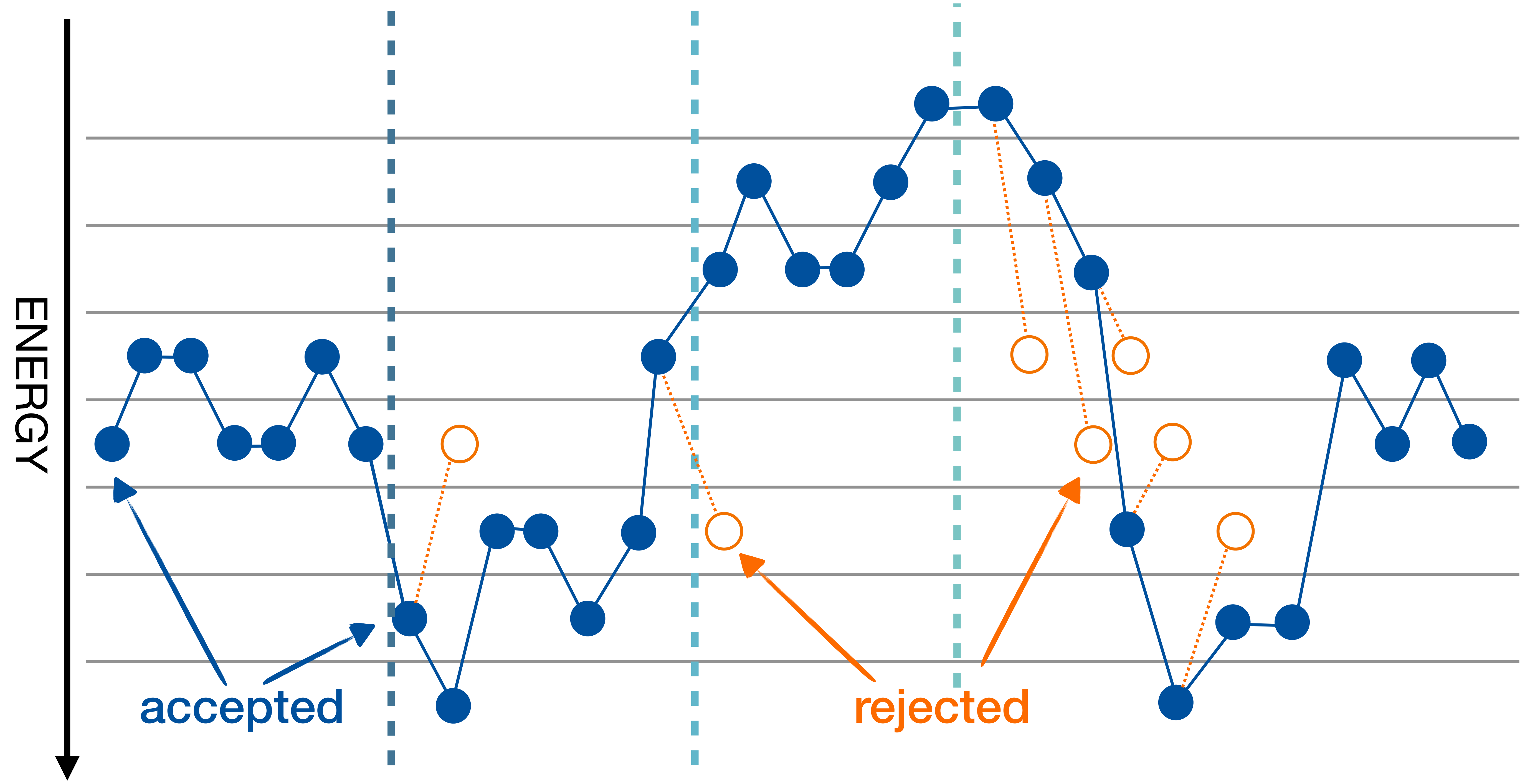
FARE RICERCA IN ITALIA

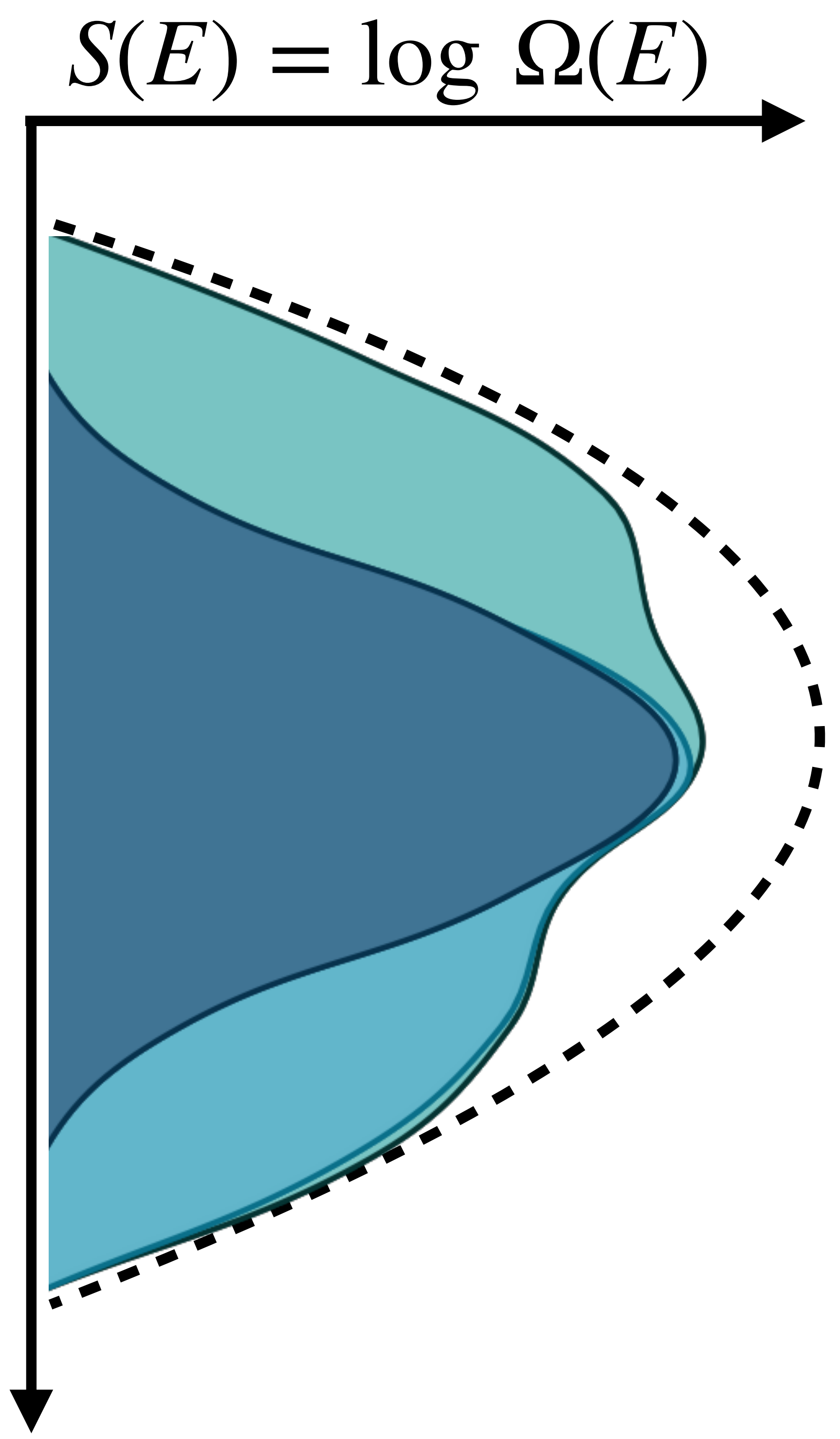
The Abdus Salam International Centre for Theoretical Physics

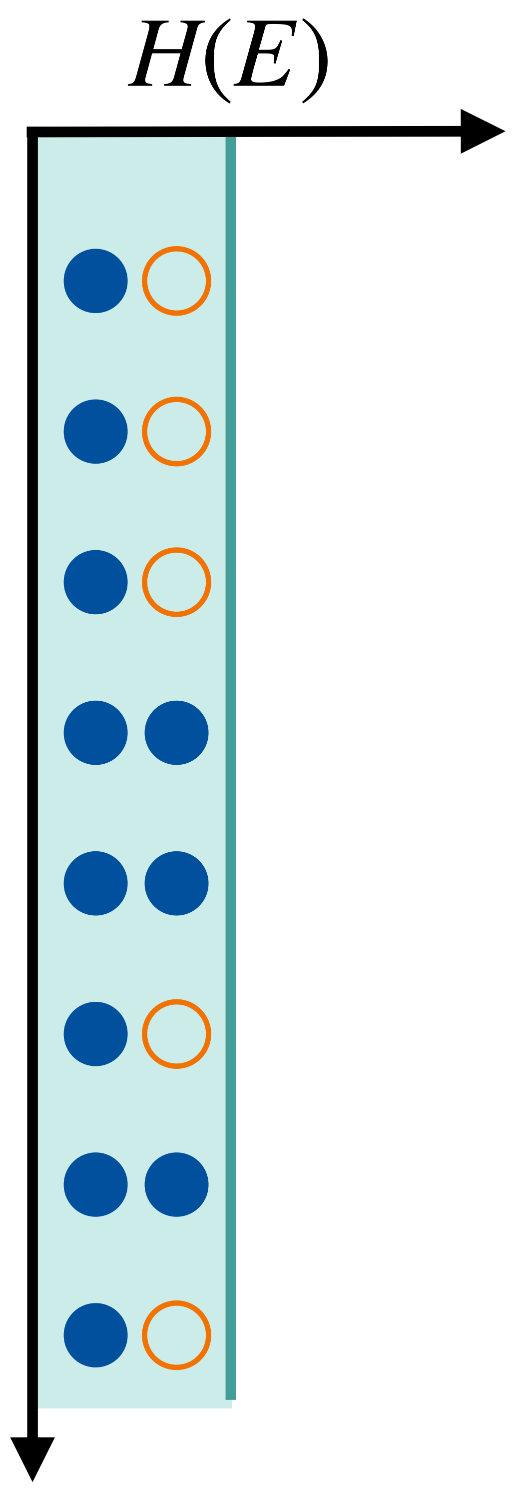
FONDAZIONE CASSA RURALE DI TRENTO

margherita.mele@unitn.it

Wang-Landau algorithm







Proposal

$$g(w, \tilde{w})$$

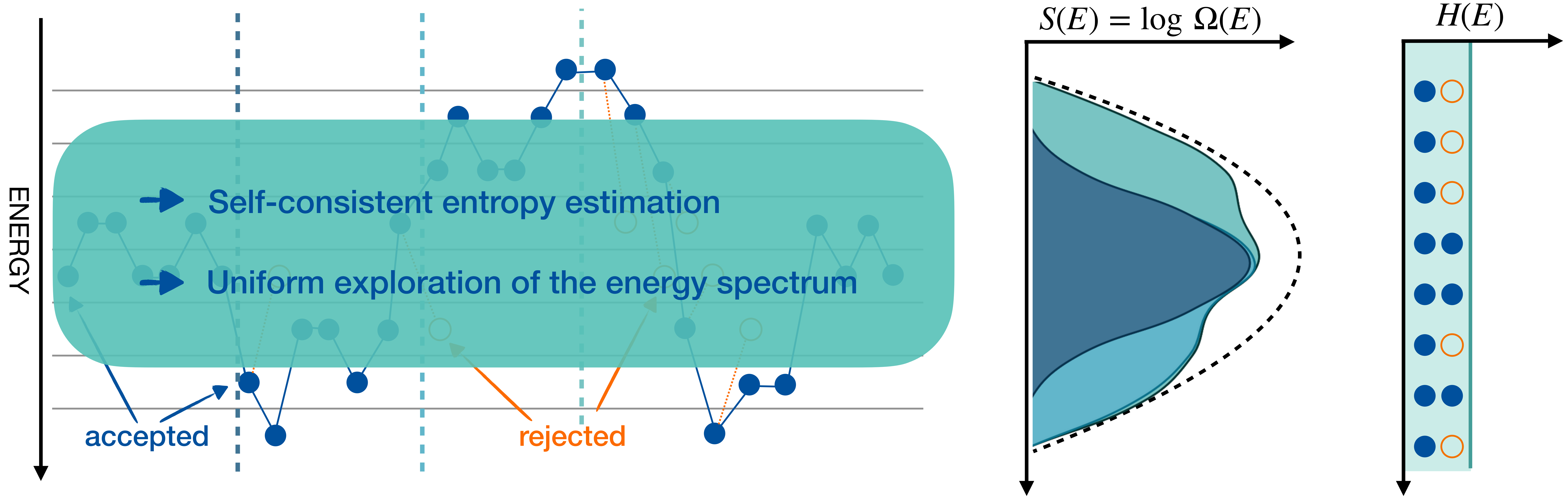
Acceptance

$$A(w,\tilde{w}) = \min\left\{1,\ e^{S_w - S_{\tilde{w}}}\frac{g(\tilde{w},w)}{g(w,\tilde{w})}\right\}$$

Updates

$$S_w \leftarrow S_w + f \qquad H_w \leftarrow H_w + 1$$

Wang-Landau algorithm



Proposal

$$g(w,\tilde{w})$$

Acceptance

$$A(w,\tilde{w}) = \min\left\{1,\ e^{S_w - S_{\tilde{w}}}\frac{g(\tilde{w},w)}{g(w,\tilde{w})}\right\}$$

Updates

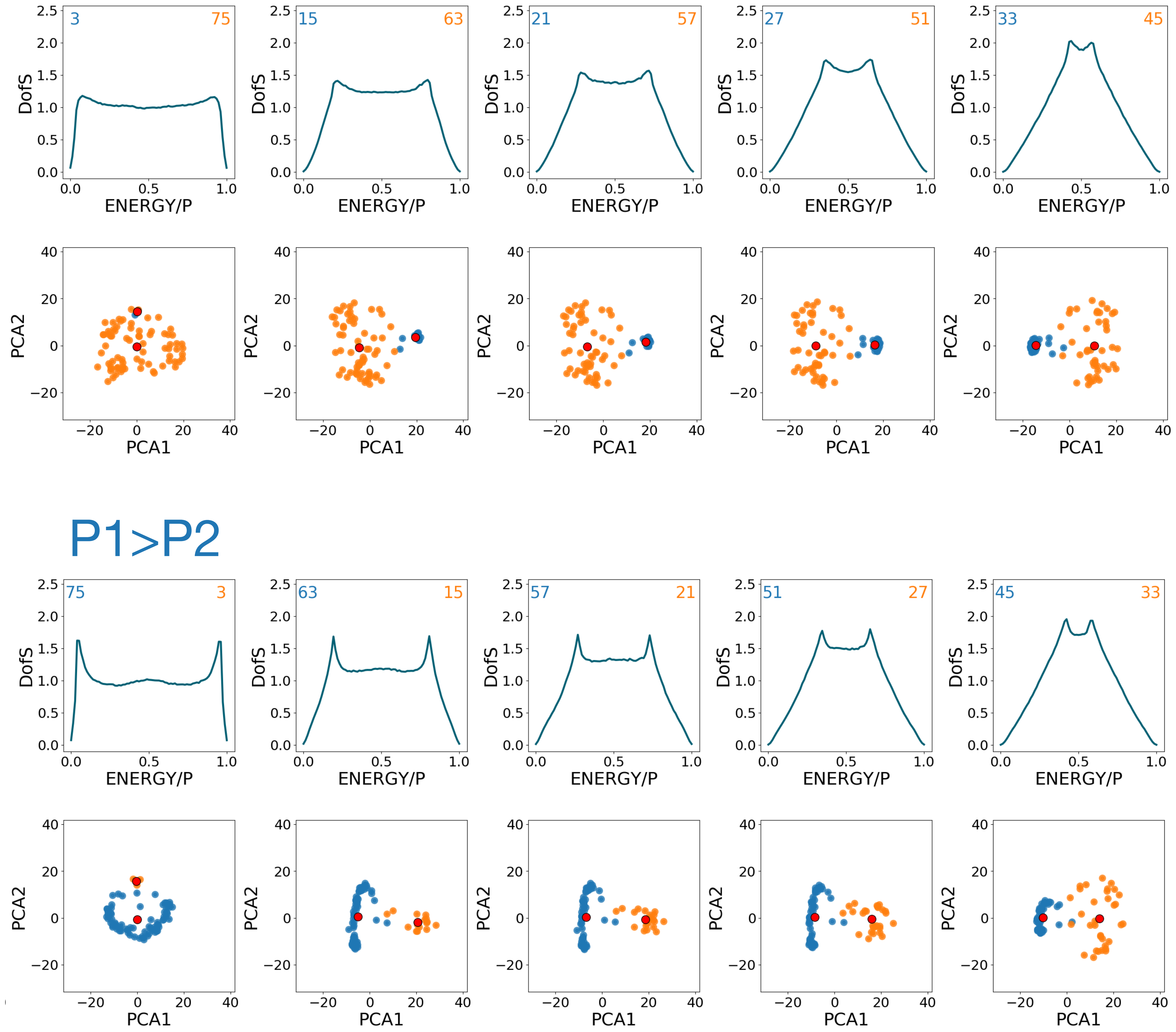
$$S_w \leftarrow S_w + f \qquad H_w \leftarrow H_w + 1$$

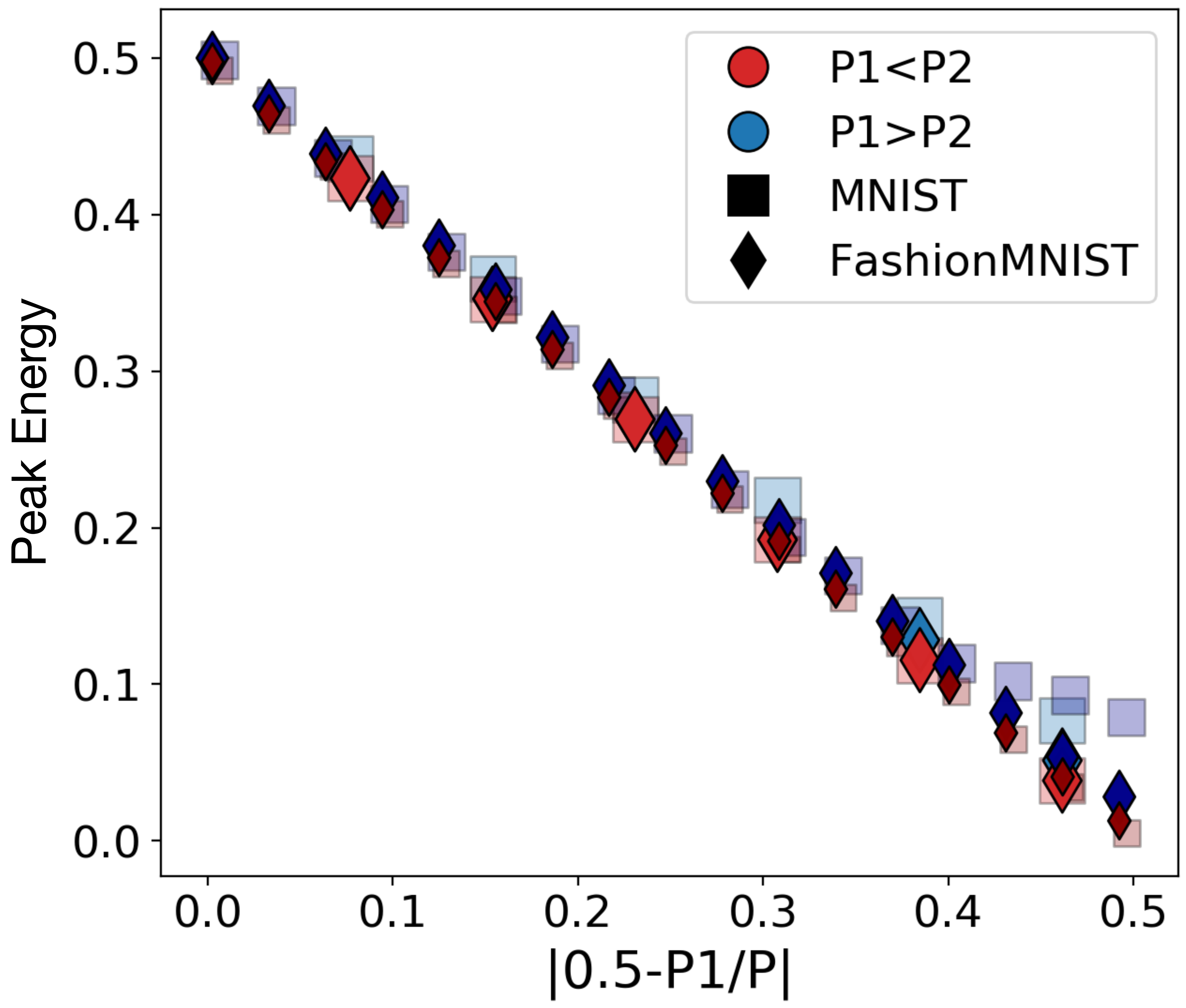
MNIST & FashionMNIST

MNIST (P/N=0.1)

P1<P2

P1>P2



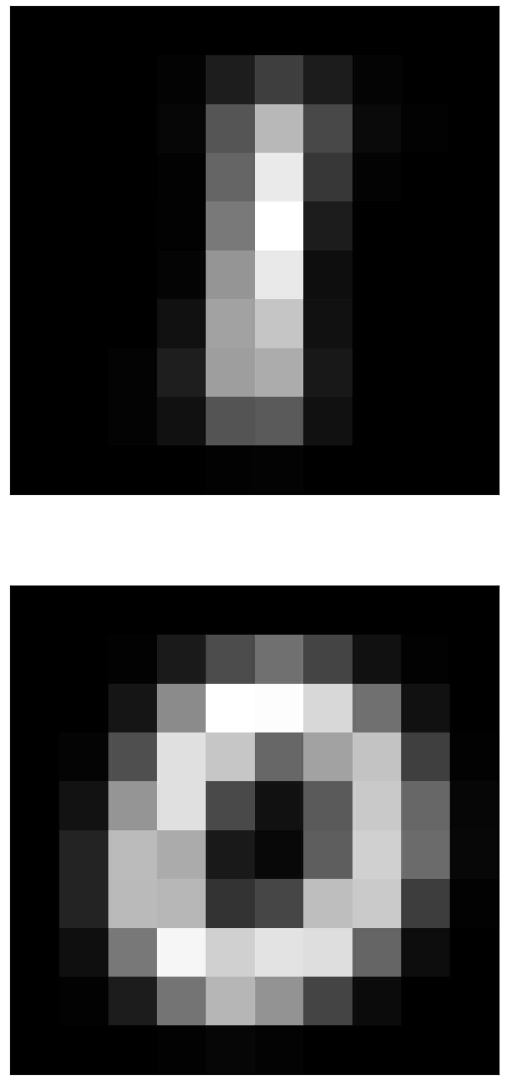


Back back to *realistic* data

Gaussian Clones

$$N(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)$$

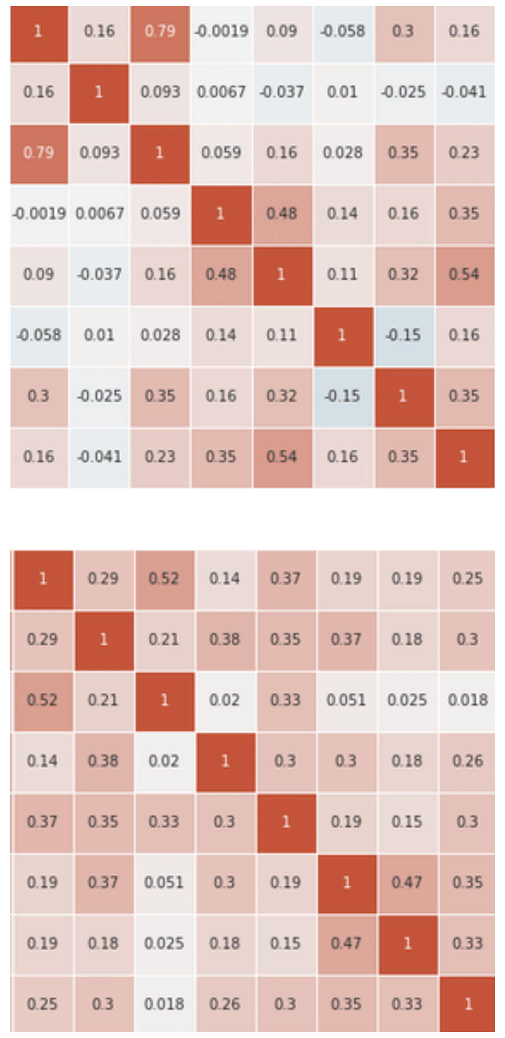
Mean - μ



+

Covariance - Σ

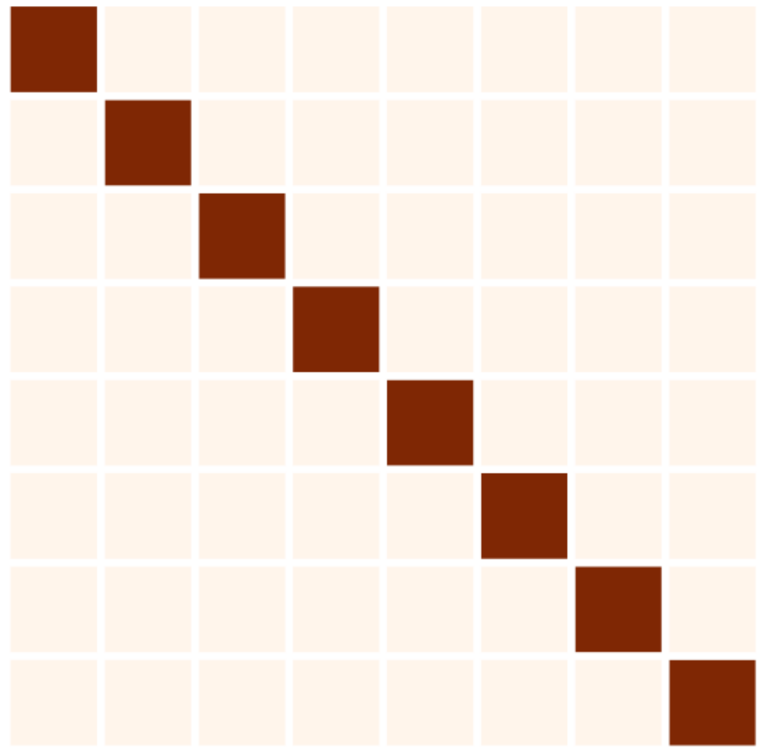
GM



2ISO

diagonal matrix

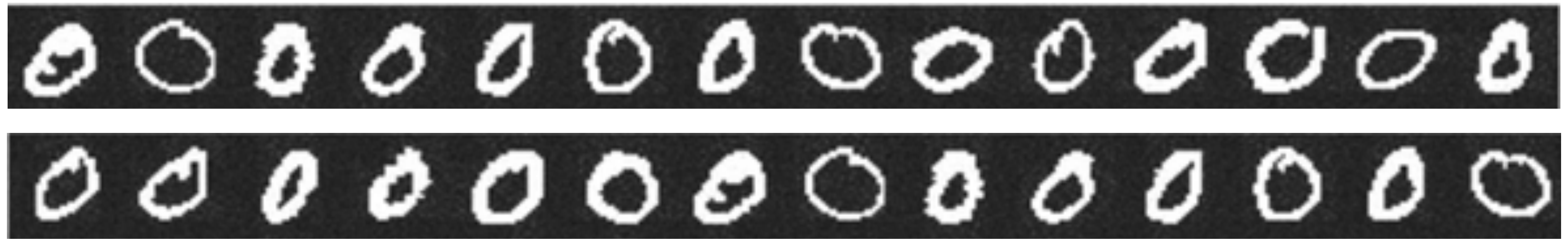
$$v = \sqrt{v_1 \cdot v_2}$$

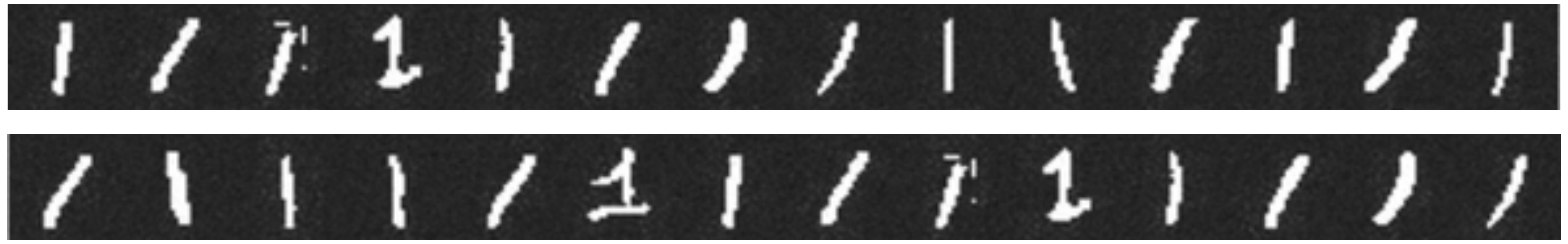


Gaussian Clones

Multivariate Normal Distribution

$$N(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T \, \Sigma^{-1} \, (x-\mu)\right)$$



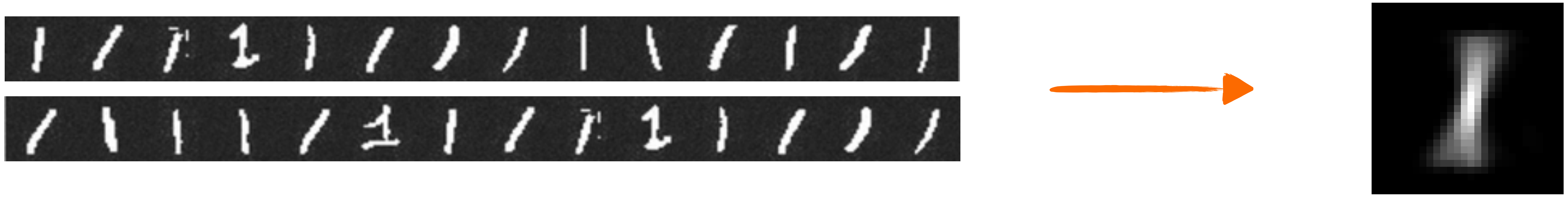


Gaussian Clones

Multivariate Normal Distribution

$$N(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T \, \Sigma^{-1} \, (x-\mu)\right)$$

Mean - μ



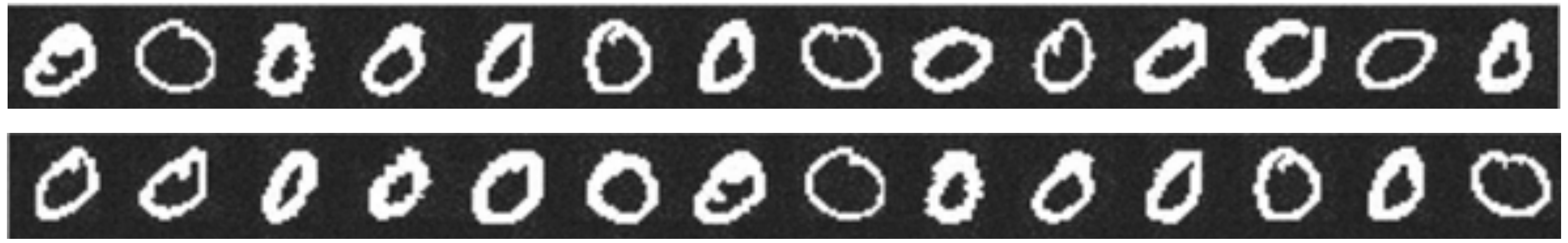
Gaussian Clones

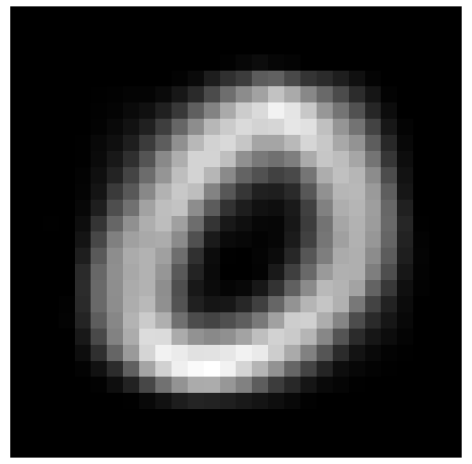
Multivariate Normal Distribution

$$N(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T \, \Sigma^{-1} \, (x-\mu)\right)$$

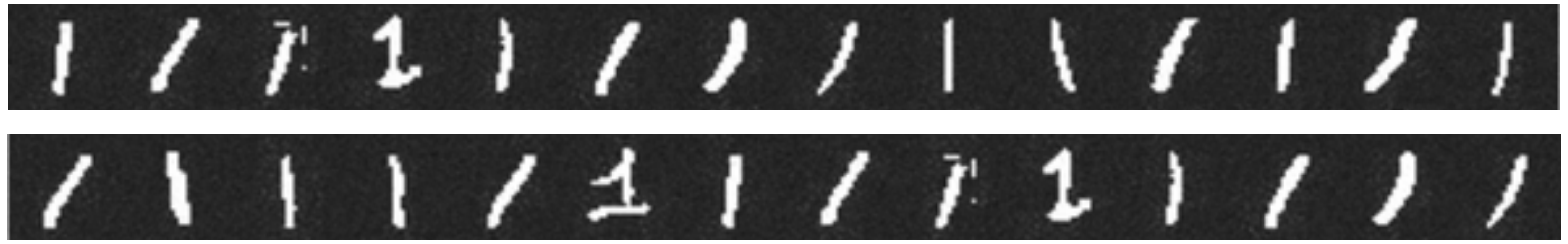
Mean - μ

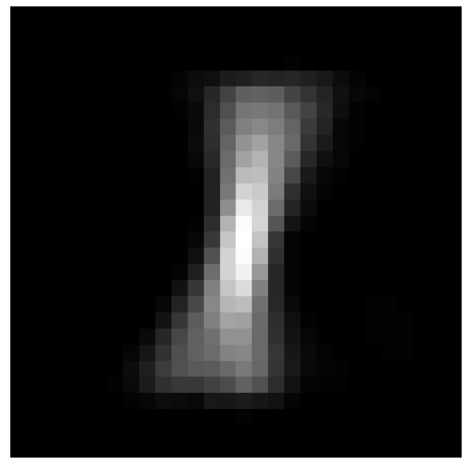
Covariance - Σ





1	0.16	0.79	-0.0019	0.09	-0.058	0.3	0.16
0.16	1	0.093	0.0067	-0.037	0.01	-0.025	-0.041
0.79	0.093	1	0.059	0.16	0.028	0.35	0.23
-0.0019	0.0067	0.059	1	0.48	0.14	0.16	0.35
0.09	-0.037	0.16	0.48	1	0.11	0.32	0.54
-0.058	0.01	0.028	0.14	0.11	1	-0.15	0.16
0.3	-0.025	0.35	0.16	0.32	-0.15	1	0.35
0.16	-0.041	0.23	0.35	0.54	0.16	0.35	1



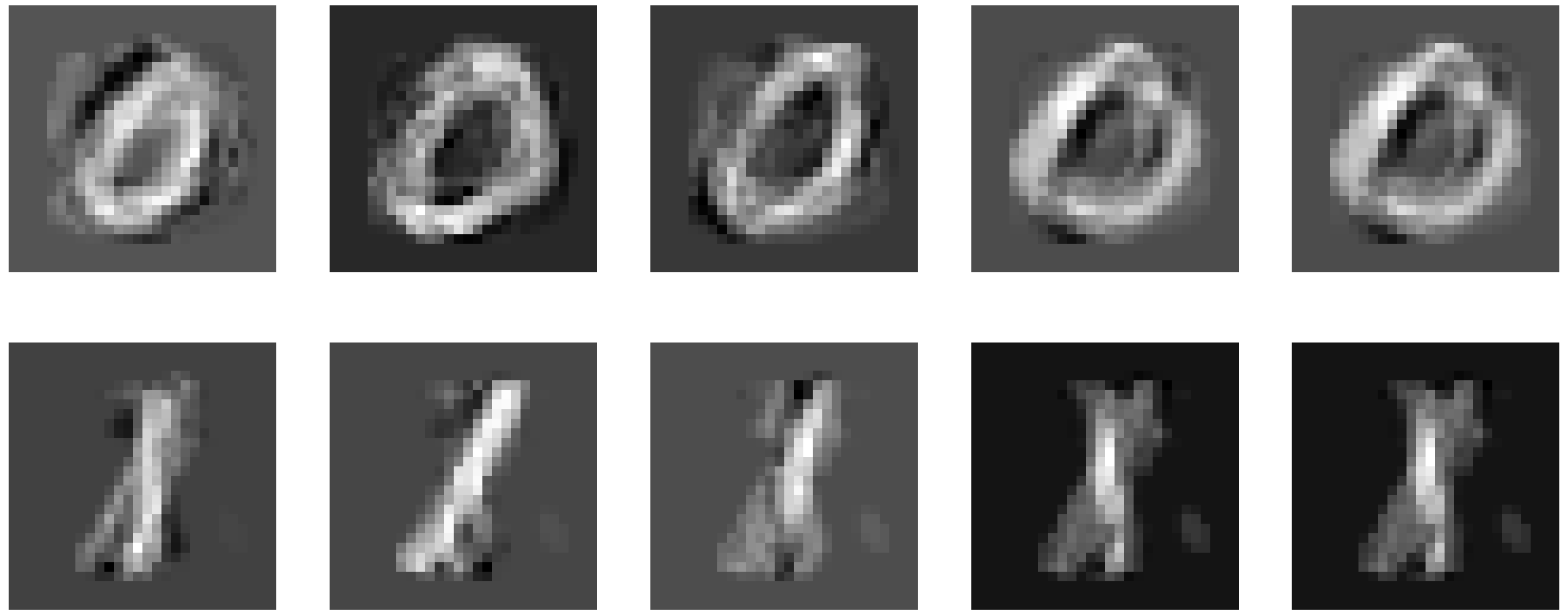


1	0.02	0.33	0.051	0.025	0.018	0.0074	0.086
0.02	1	0.3	0.3	0.18	0.26	0.3	0.27
0.33	0.3	1	0.19	0.15	0.3	0.19	0.33
0.051	0.3	0.19	1	0.47	0.35	0.88	0.53
0.025	0.18	0.15	0.47	1	0.33	0.33	0.75
0.018	0.26	0.3	0.35	0.33	1	0.28	0.76
0.0074	0.3	0.19	0.88	0.33	0.28	1	0.4
0.086	0.27	0.33	0.53	0.75	0.76	0.4	1

Gaussian Clones

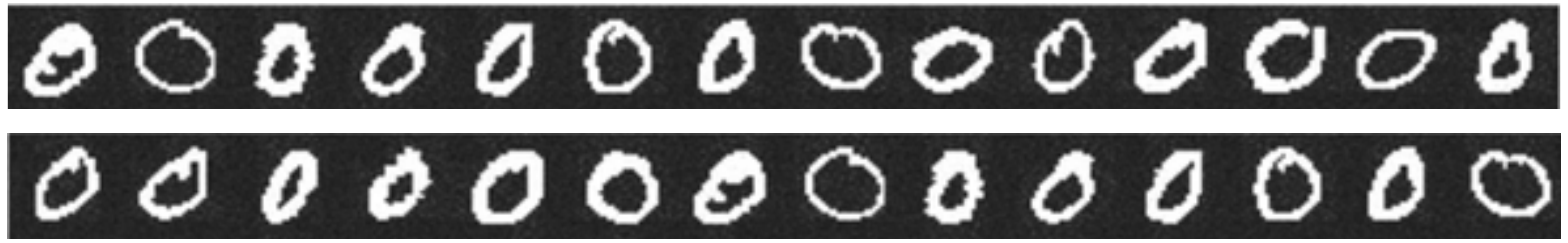
Multivariate Normal Distribution

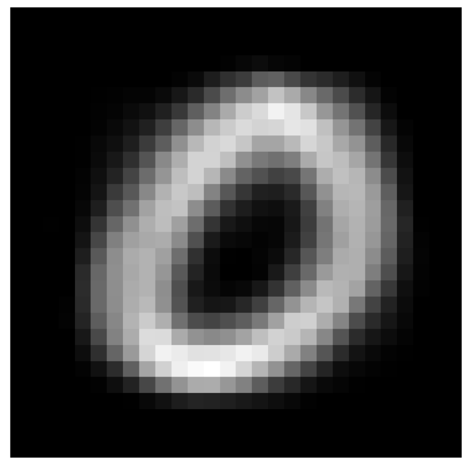
$$N(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T \, \Sigma^{-1} \, (x-\mu)\right)$$



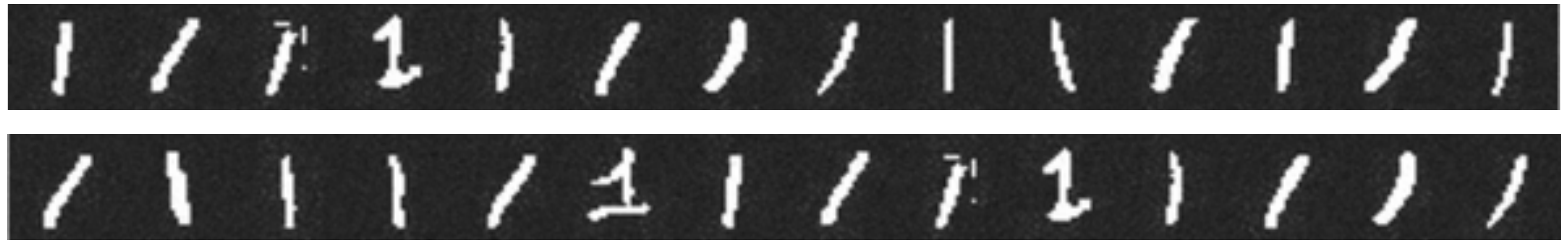
Mean - μ

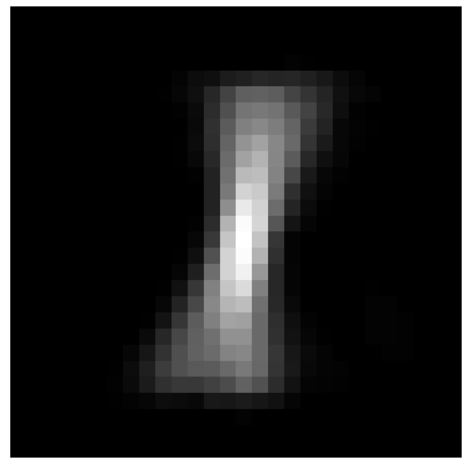
Covariance - Σ





1	0.16	0.79	-0.0019	0.09	-0.058	0.3	0.16
0.16	1	0.093	0.0067	-0.037	0.01	-0.025	-0.041
0.79	0.093	1	0.059	0.16	0.028	0.35	0.23
-0.0019	0.0067	0.059	1	0.48	0.14	0.16	0.35
0.09	-0.037	0.16	0.48	1	0.11	0.32	0.54
-0.058	0.01	0.028	0.14	0.11	1	-0.15	0.16
0.3	-0.025	0.35	0.16	0.32	-0.15	1	0.35
0.16	-0.041	0.23	0.35	0.54	0.16	0.35	1





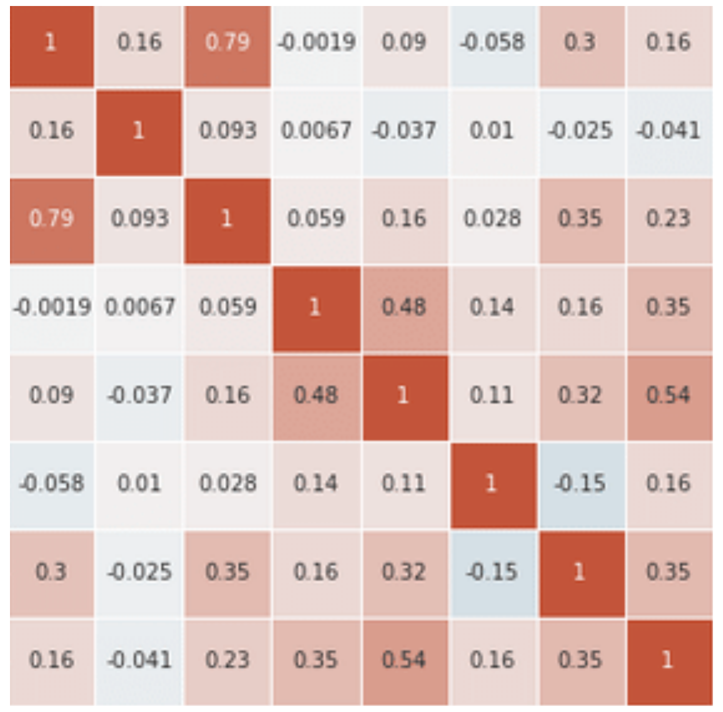
1	0.02	0.33	0.051	0.025	0.018	0.0074	0.086
0.02	1	0.3	0.3	0.18	0.26	0.3	0.27
0.33	0.3	1	0.19	0.15	0.3	0.19	0.33
0.051	0.3	0.19	1	0.47	0.35	0.88	0.53
0.025	0.18	0.15	0.47	1	0.33	0.33	0.75
0.018	0.26	0.3	0.35	0.33	1	0.28	0.76
0.0074	0.3	0.19	0.88	0.33	0.28	1	0.4
0.086	0.27	0.33	0.53	0.75	0.76	0.4	1

Gaussian Clones

Multivariate Normal Distribution

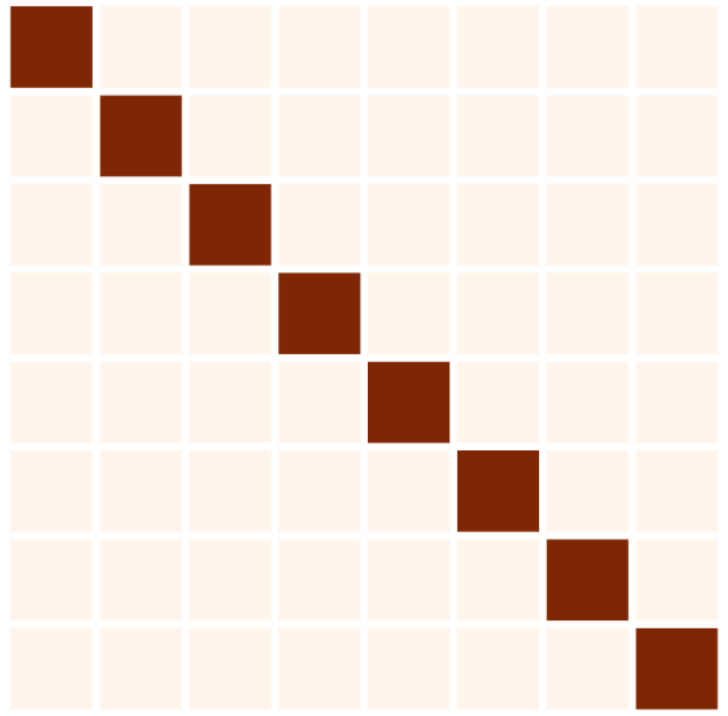
$$N(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^k \det \Sigma}} \exp\left(-\frac{1}{2}(x-\mu)^T \, \Sigma^{-1} \, (x-\mu) \right)$$

GM clone



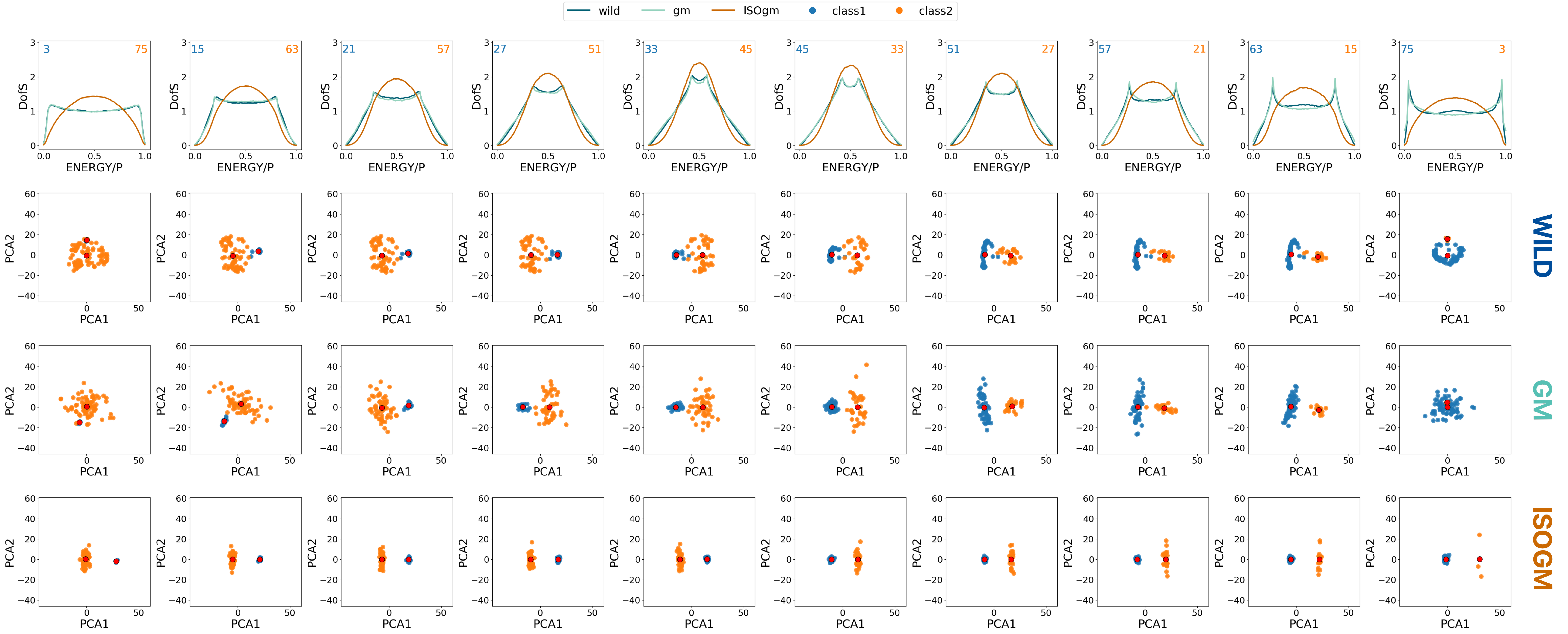
- **Mean vector**
- **Covariance matrix**

ISOGM clone

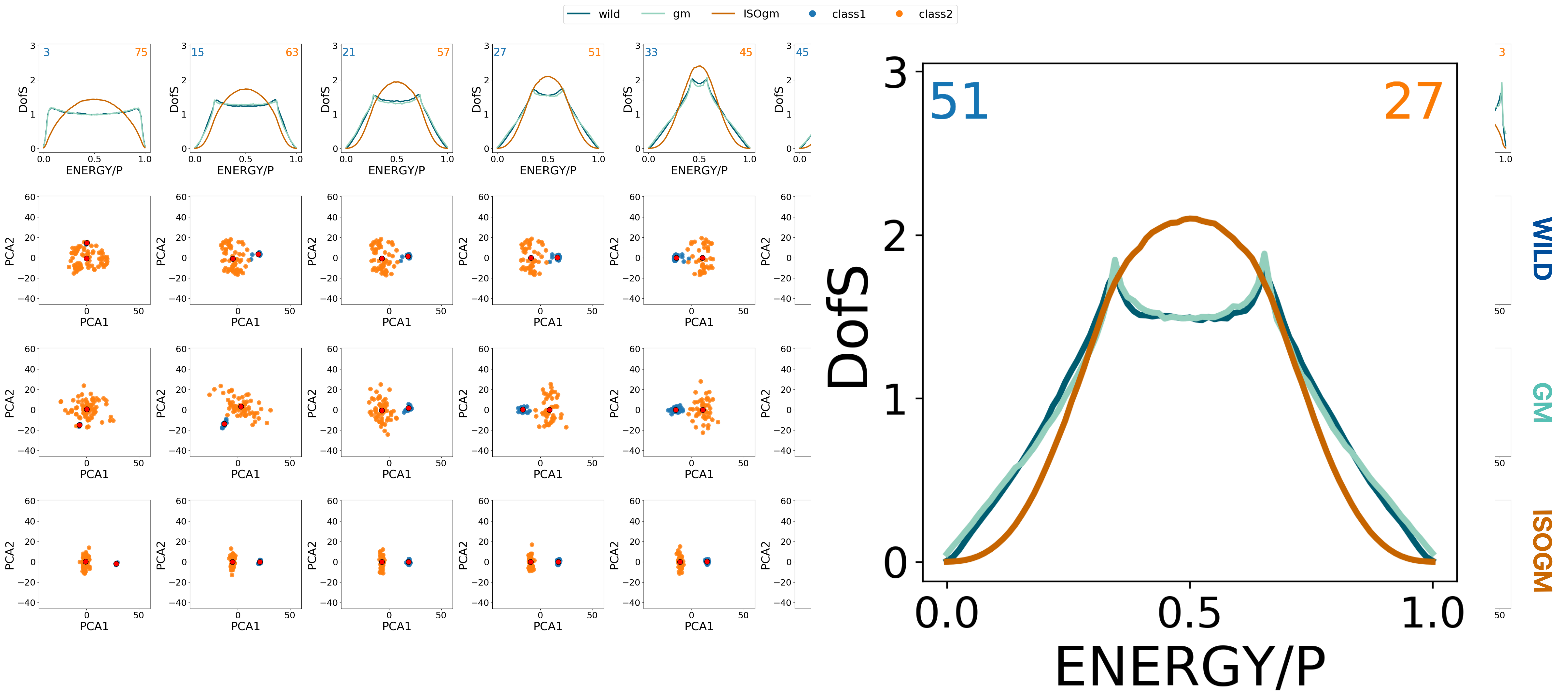


- **Mean vector**
- **Variance : diagonal matrix**

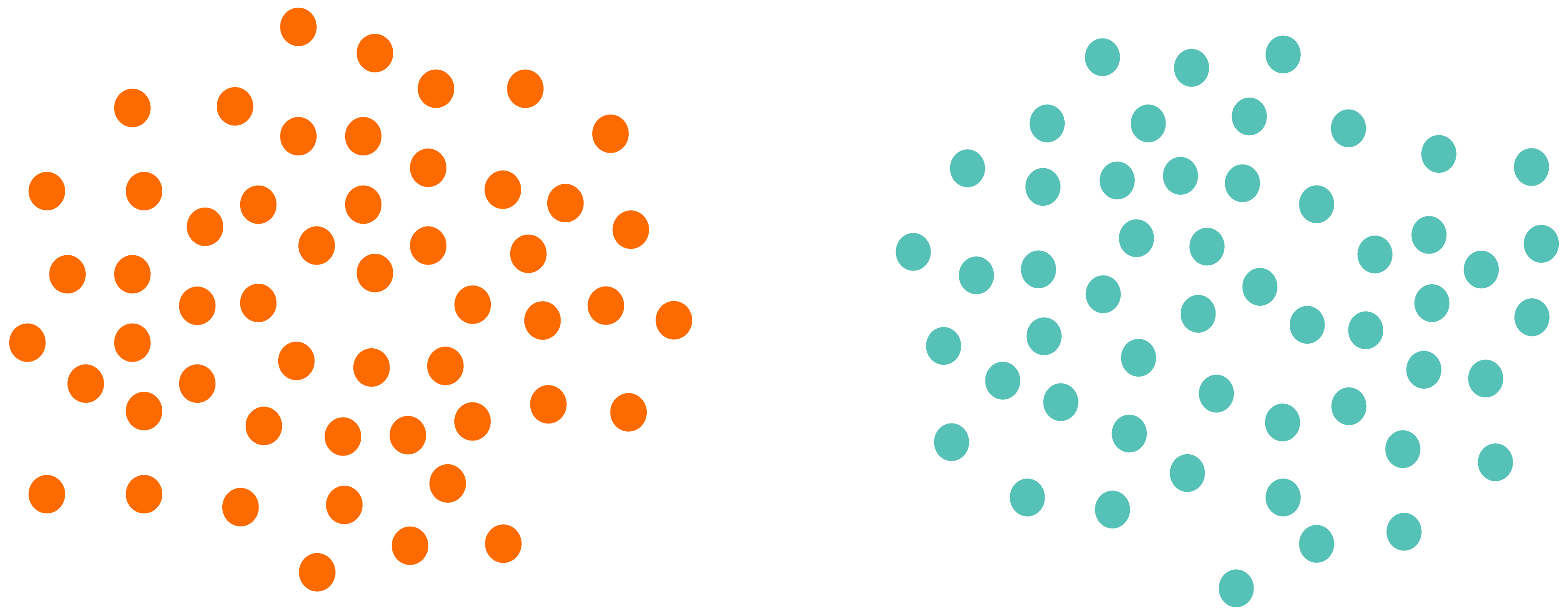
Gaussian Clones - MNIST (P/N=0.1)



Gaussian Clones - MNIST (P/N=0.1)

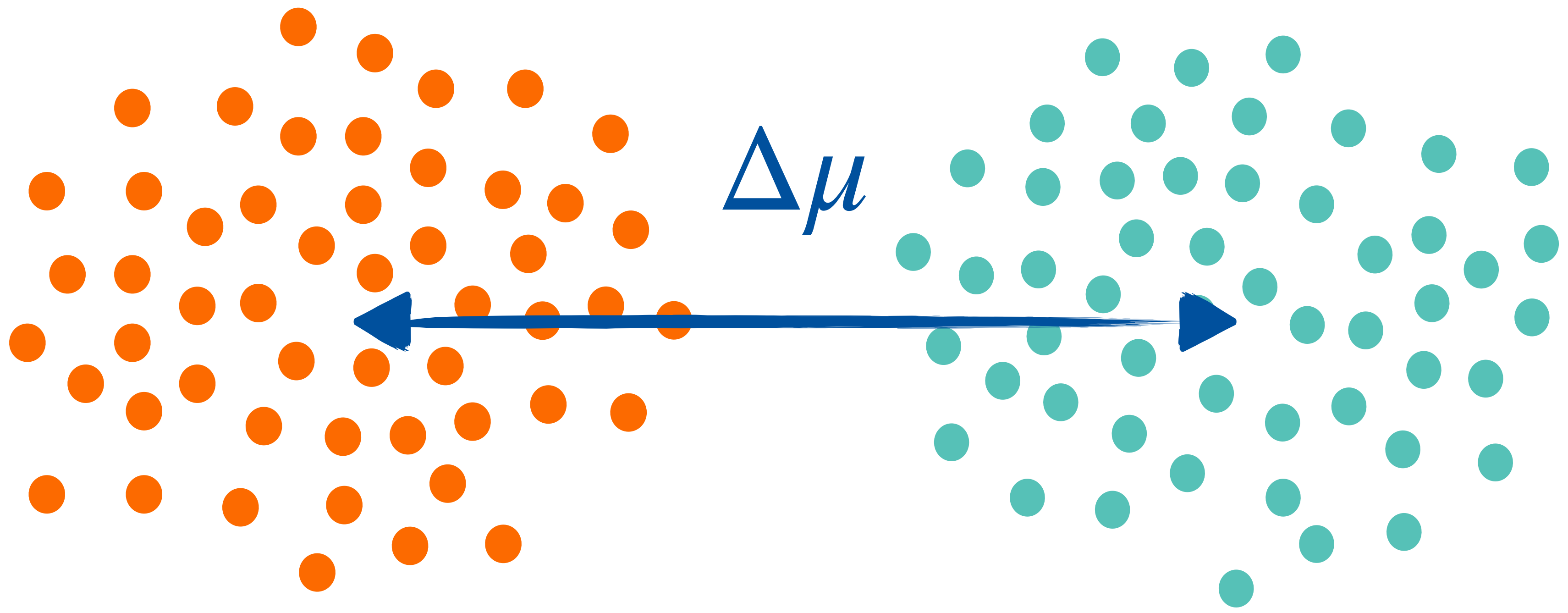


Silico dataset



Silico dataset

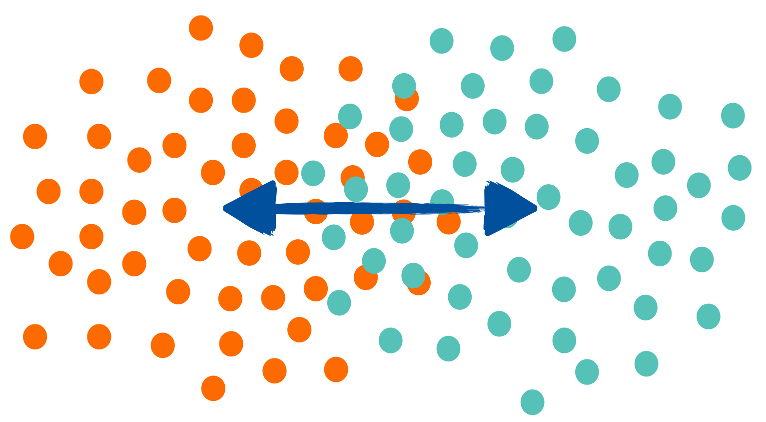
Class separation

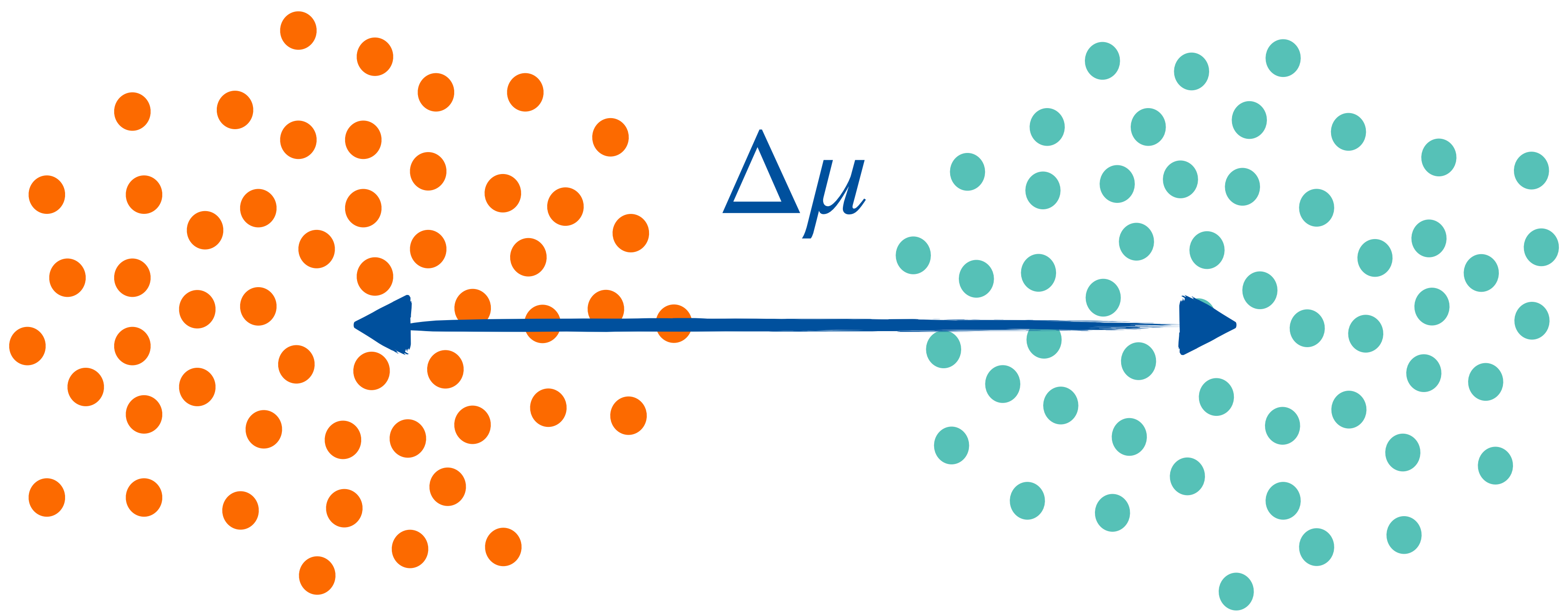


Silico dataset

Class separation

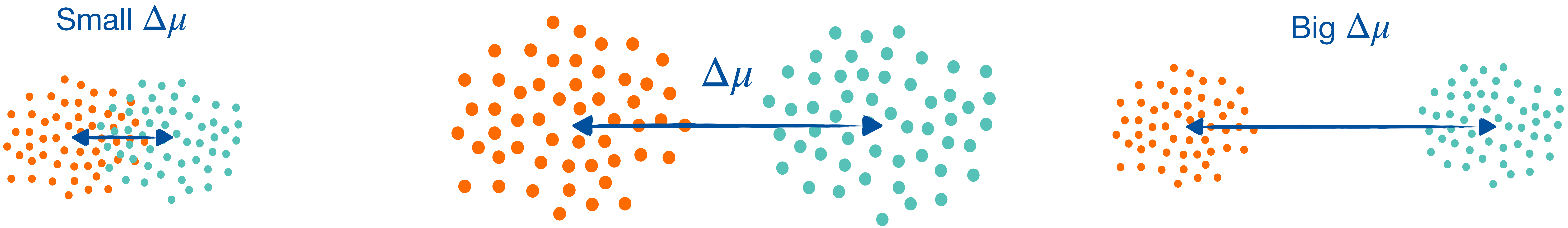
Small $\Delta\mu$





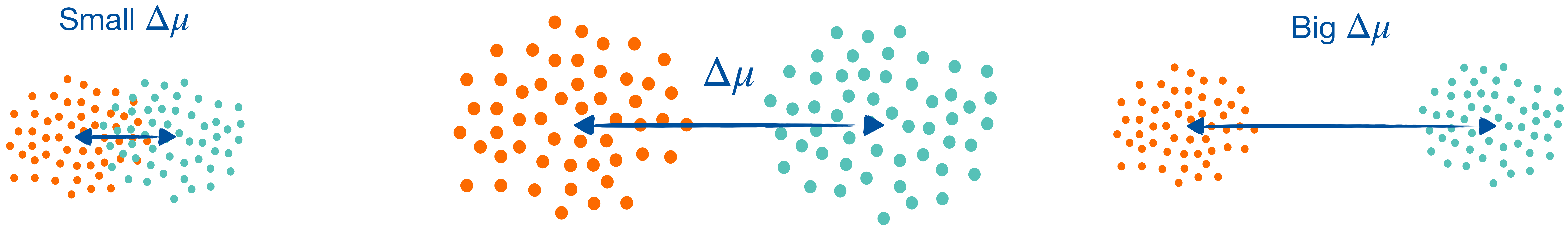
Silico dataset

Class separation

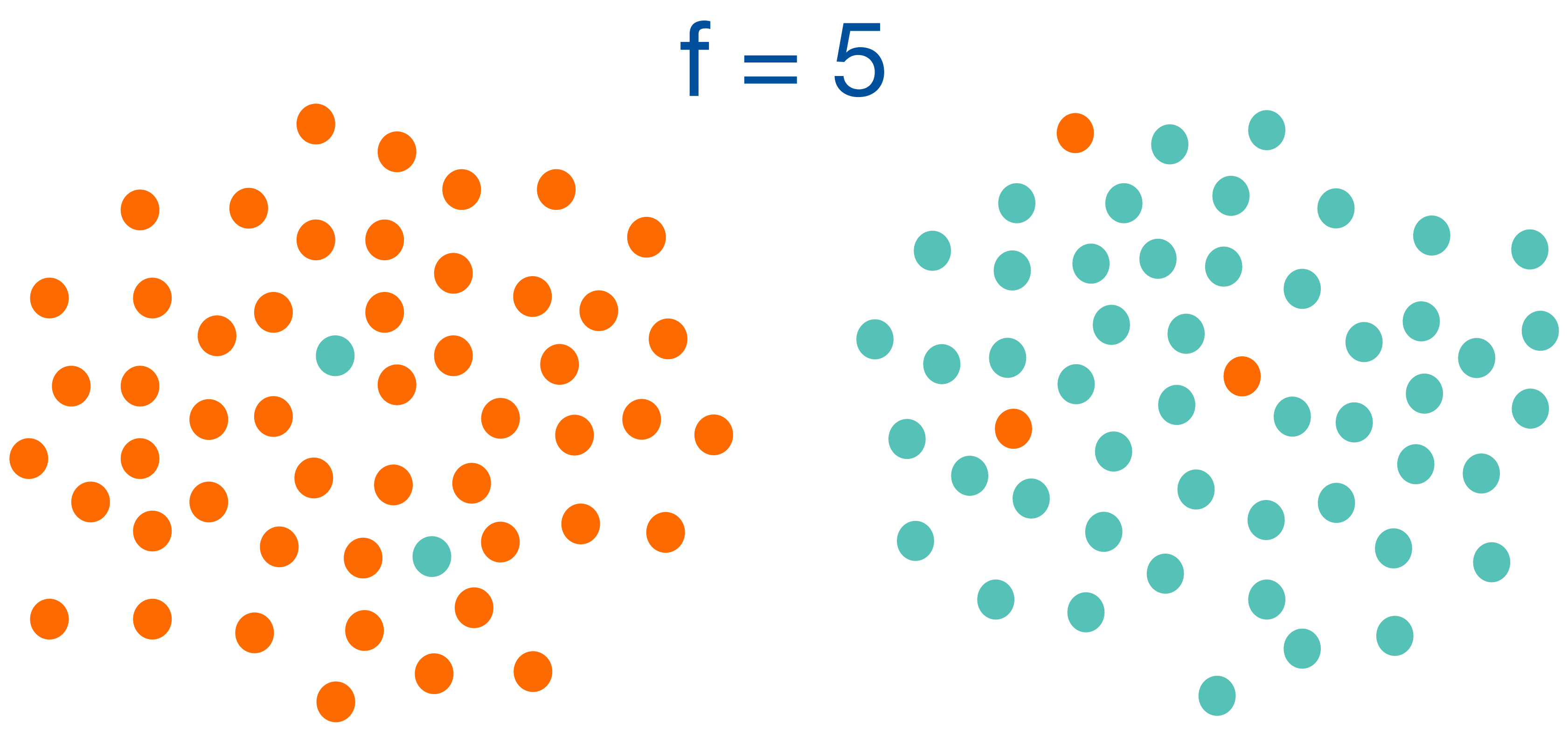


Silico dataset

Class separation

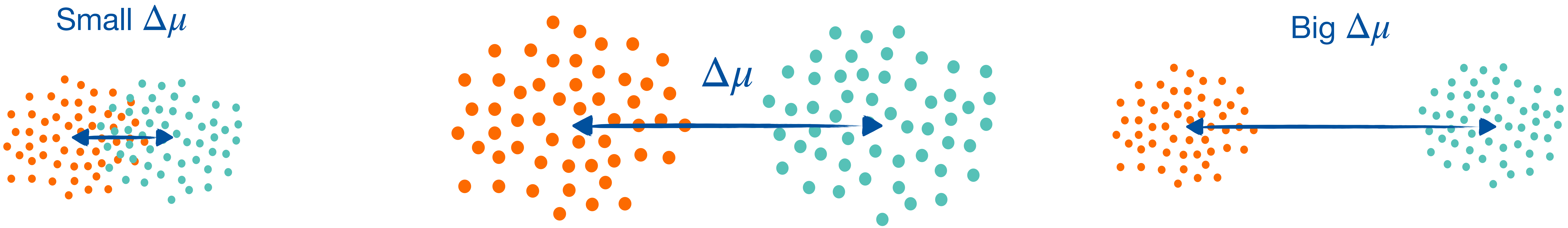


Misclassification

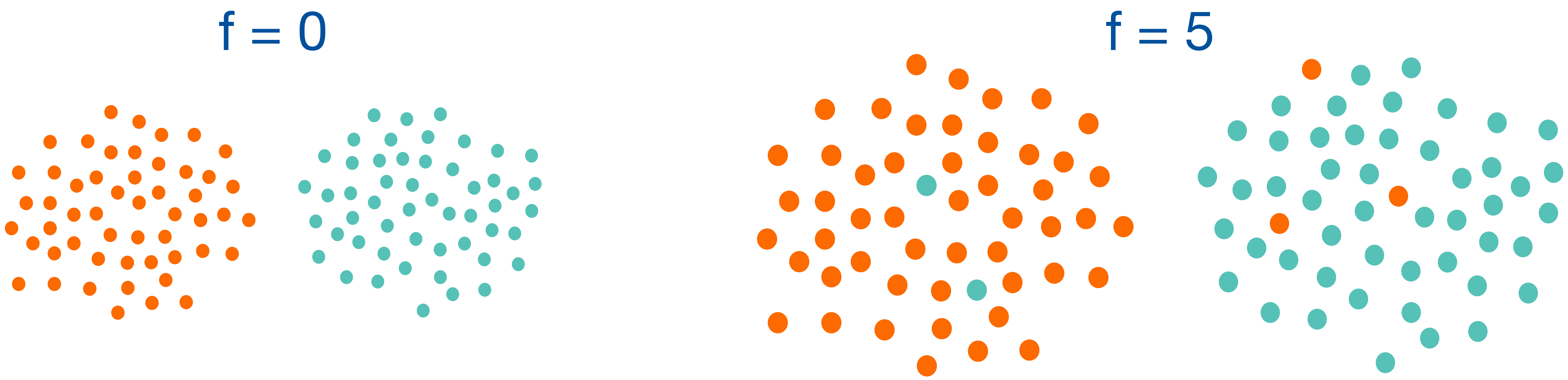


Silico dataset

Class separation

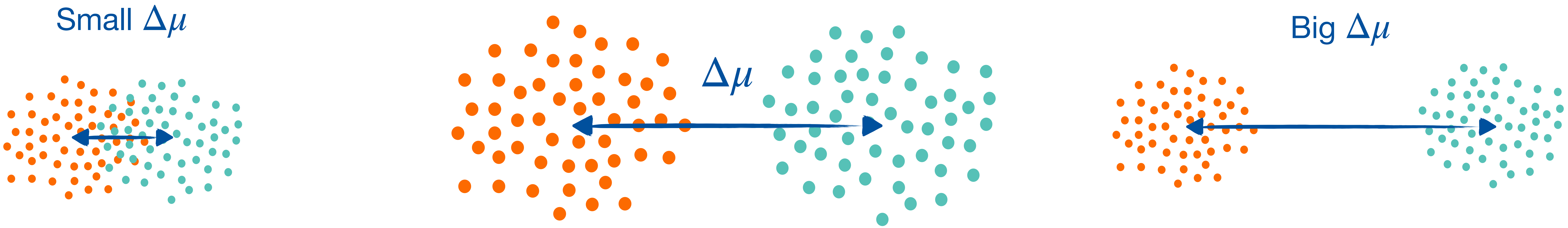


Misclassification

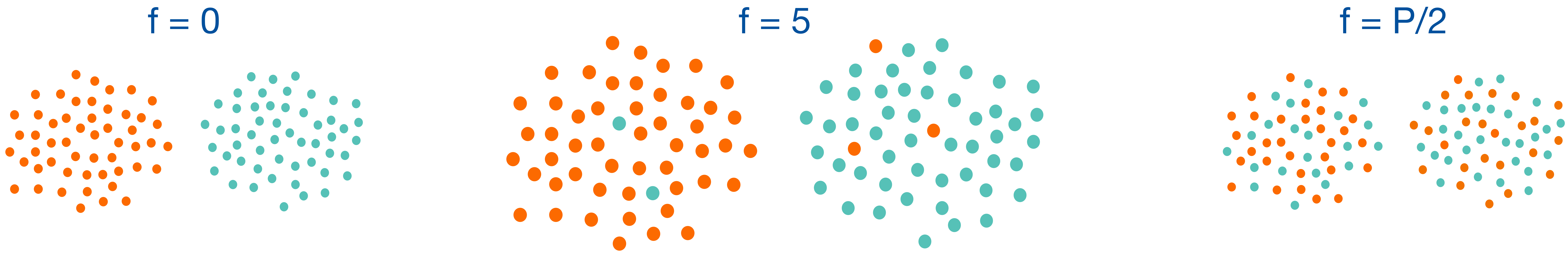


Silico dataset

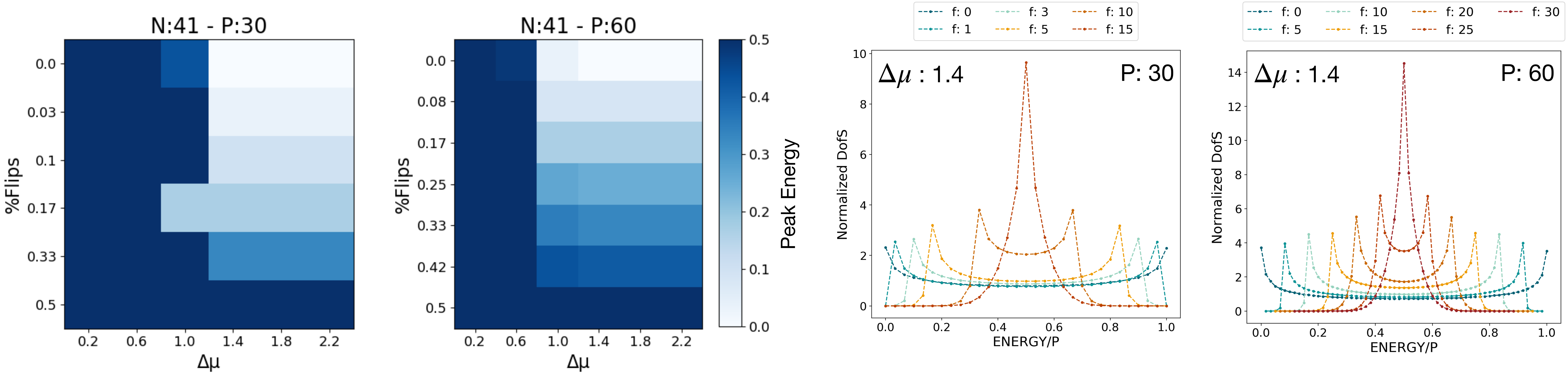
Class separation



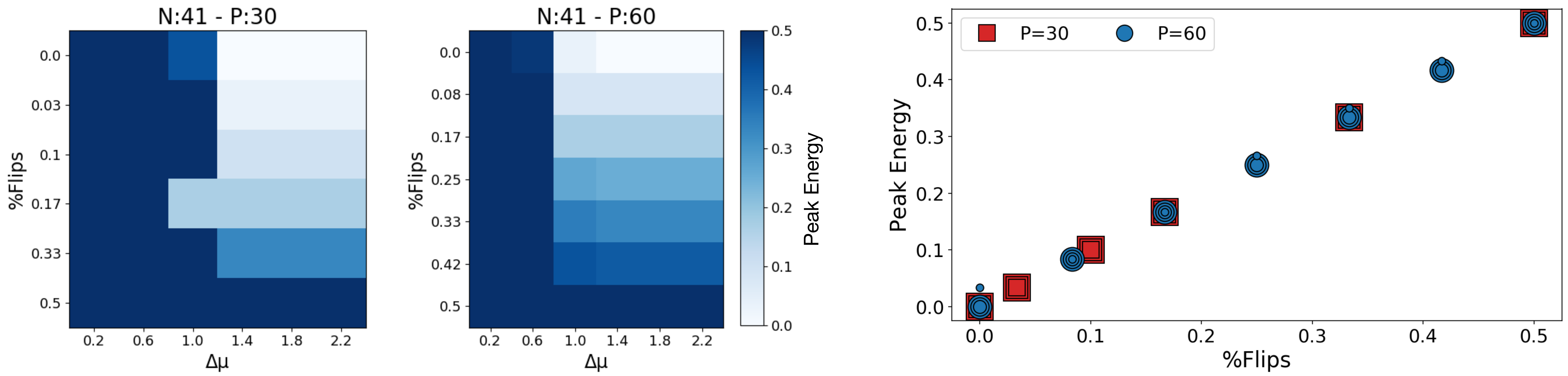
Misclassification



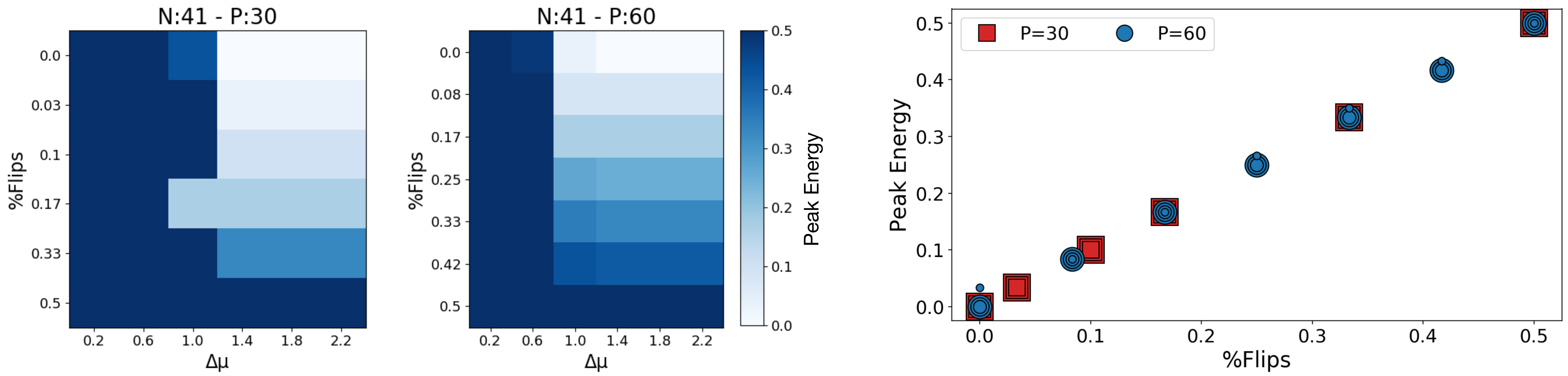
Silico dataset



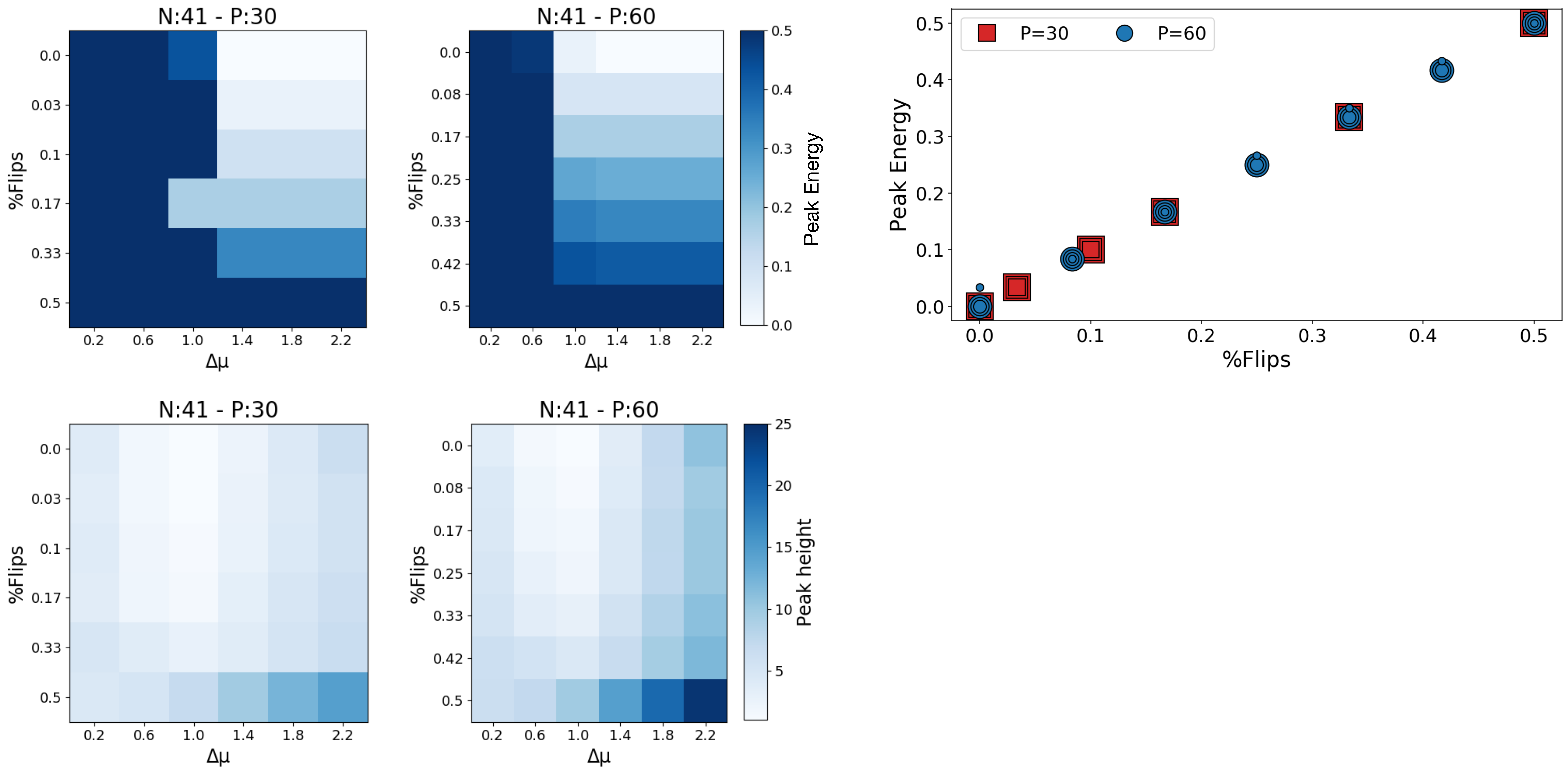
Silico dataset



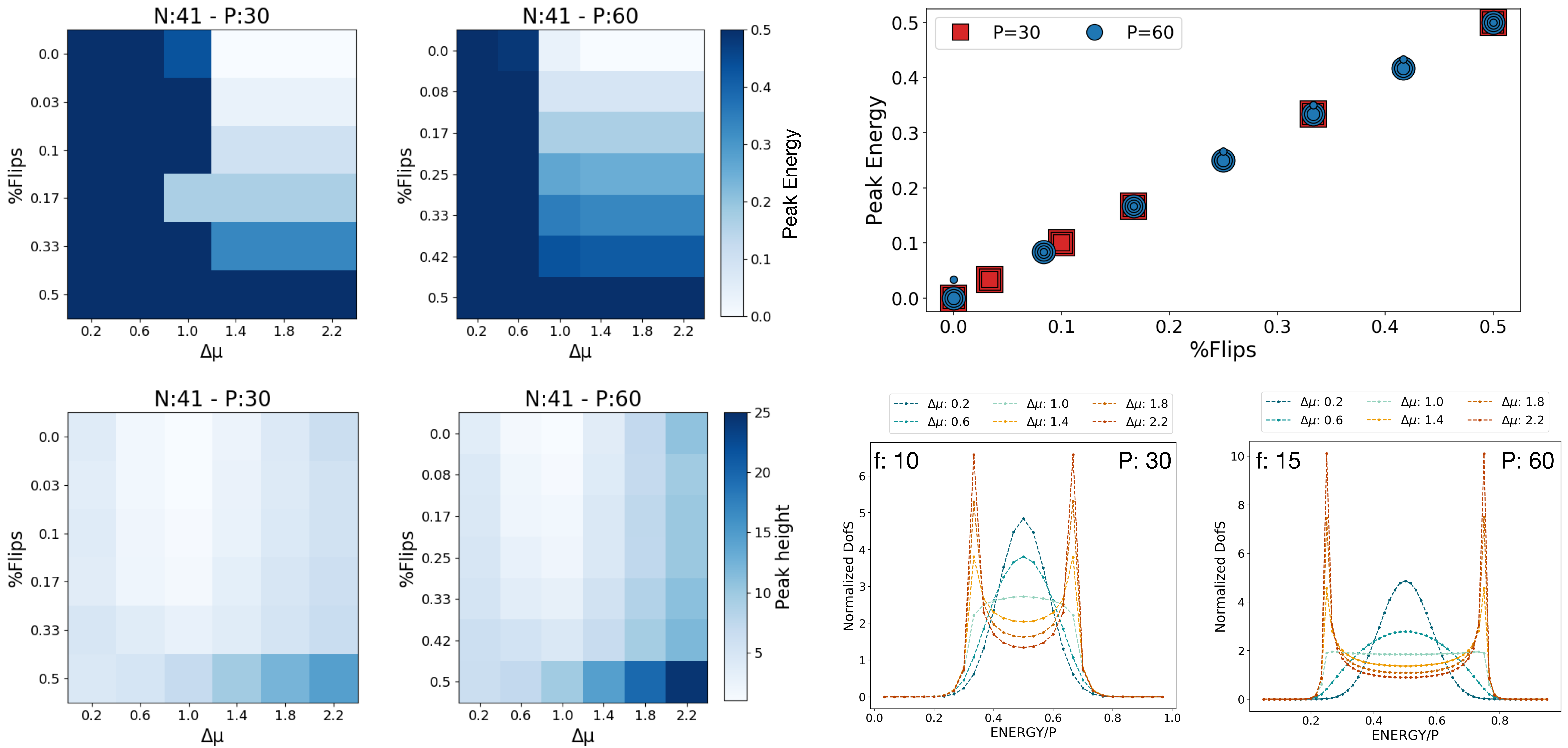
Silico dataset



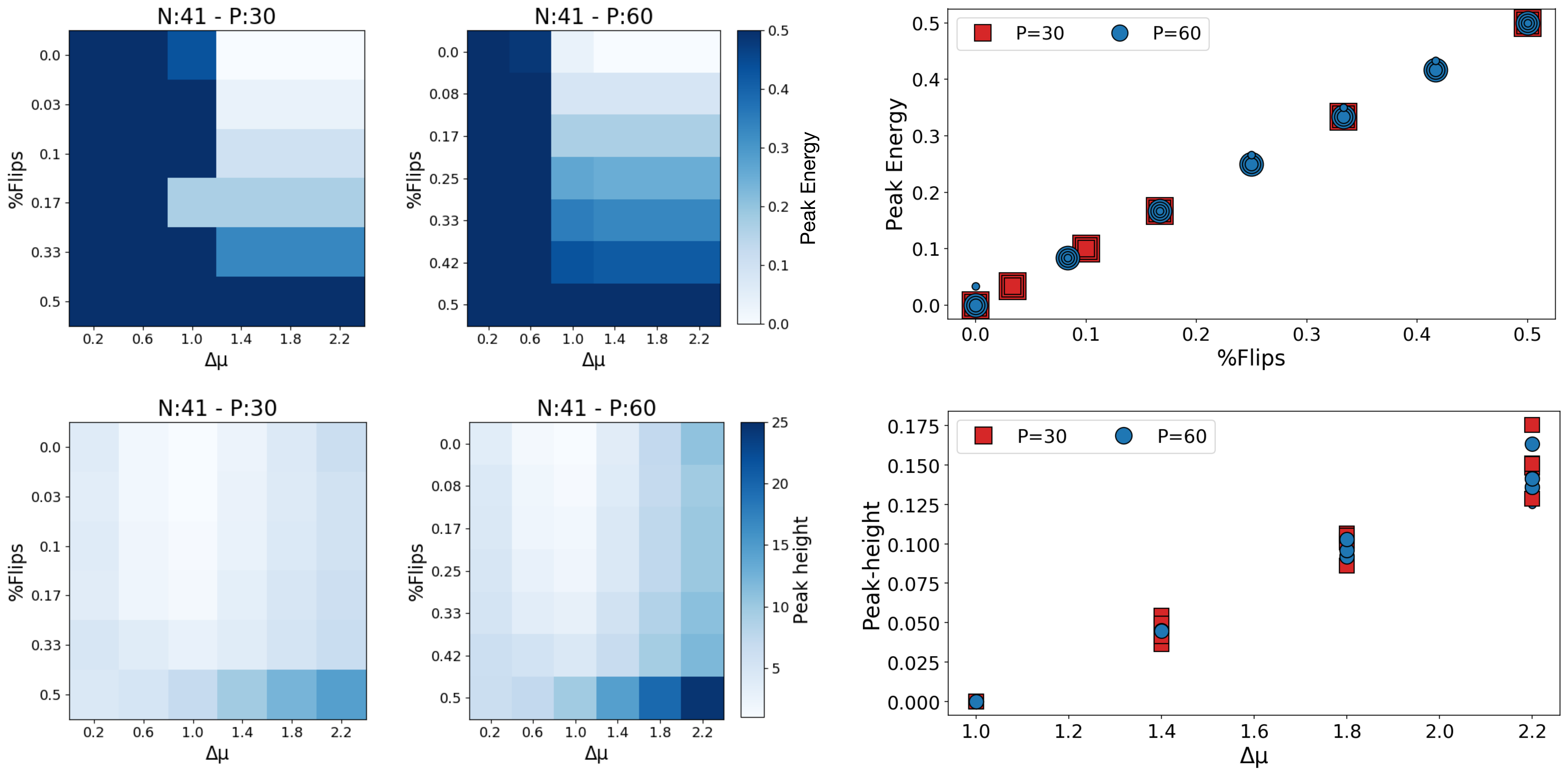
Silico dataset



Silico dataset



Silico dataset



Convergence time