UNIVERSITY OF OXFORD

On the Limitations of Representing Functions on Sets

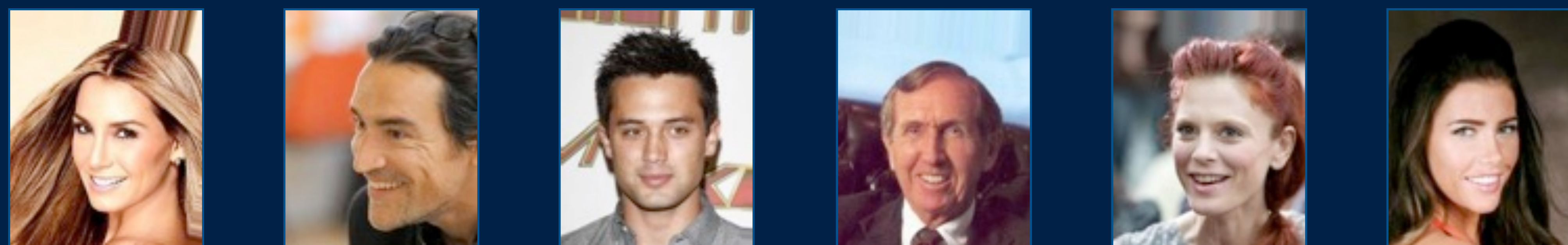
Edward Wagstaff*, Fabian Fuchs*, Martin Engelcke*
Ingmar Posner, Michael Osborne

Machine Learning Research Group

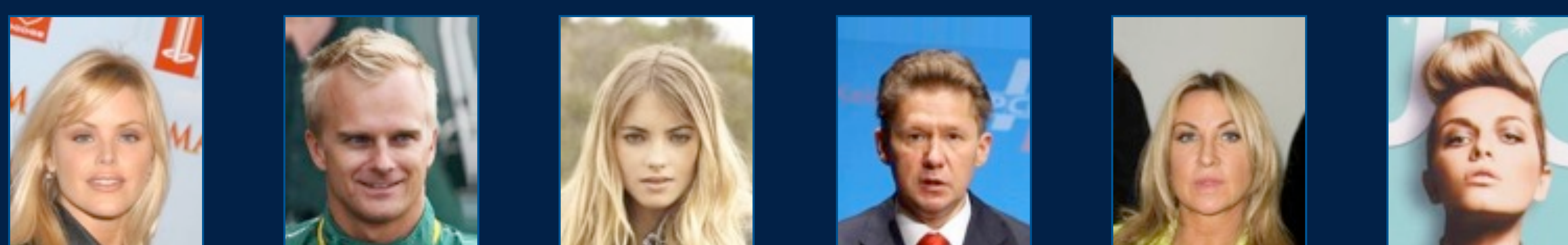
APPLIED ARTIFICIAL INTELLIGENCE LAB
OXFORD ROBOTICS INSTITUTE

*Equal contribution

Examples for Permutation Invariant Problems: Detecting Common Attributes



→ Smiling

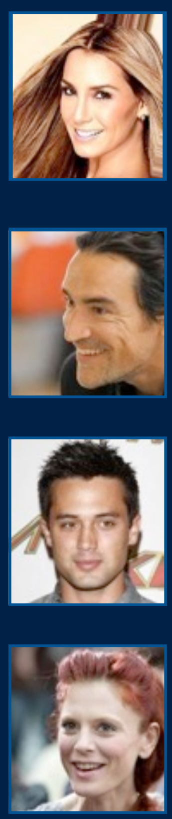


→ Blond Hair

CelebA Dataset, Liu et al.

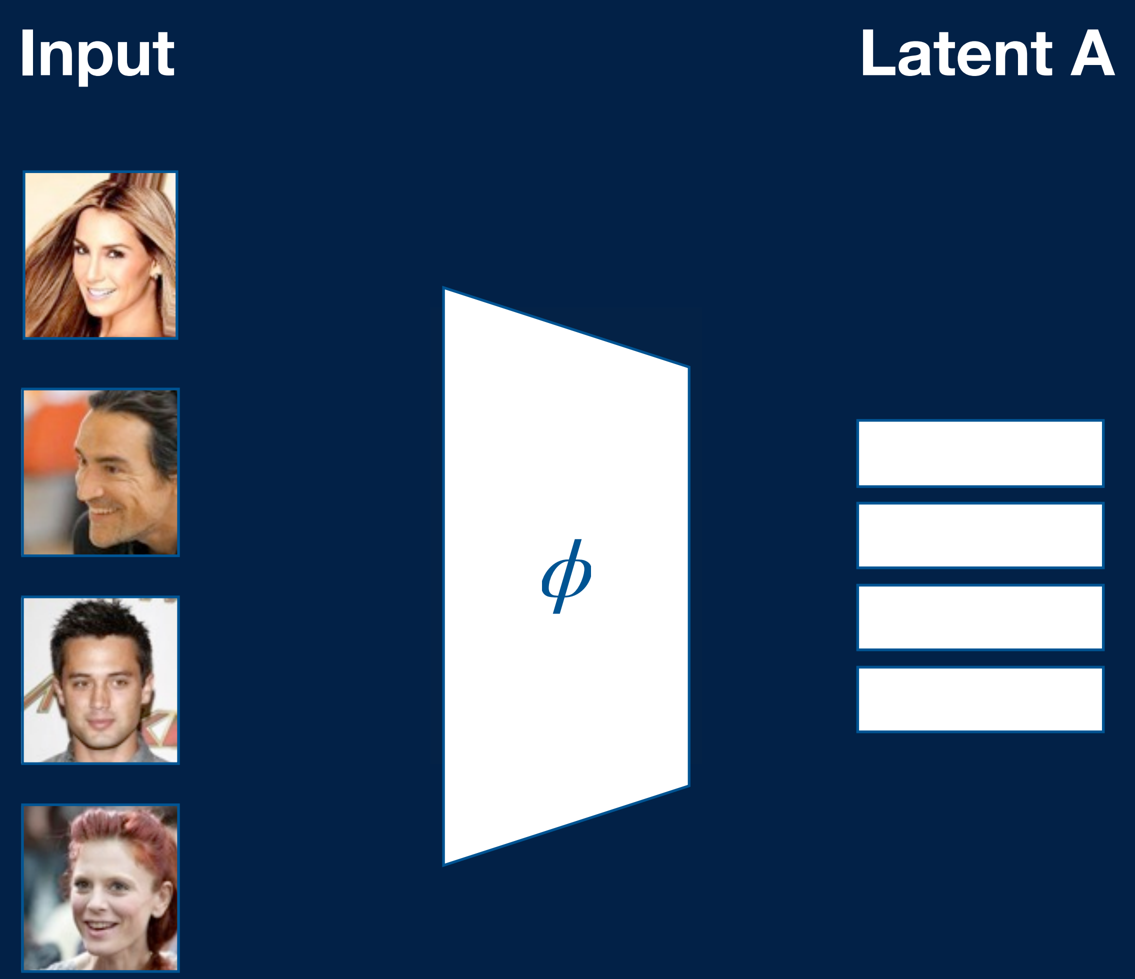
The deep sets architecture

Input

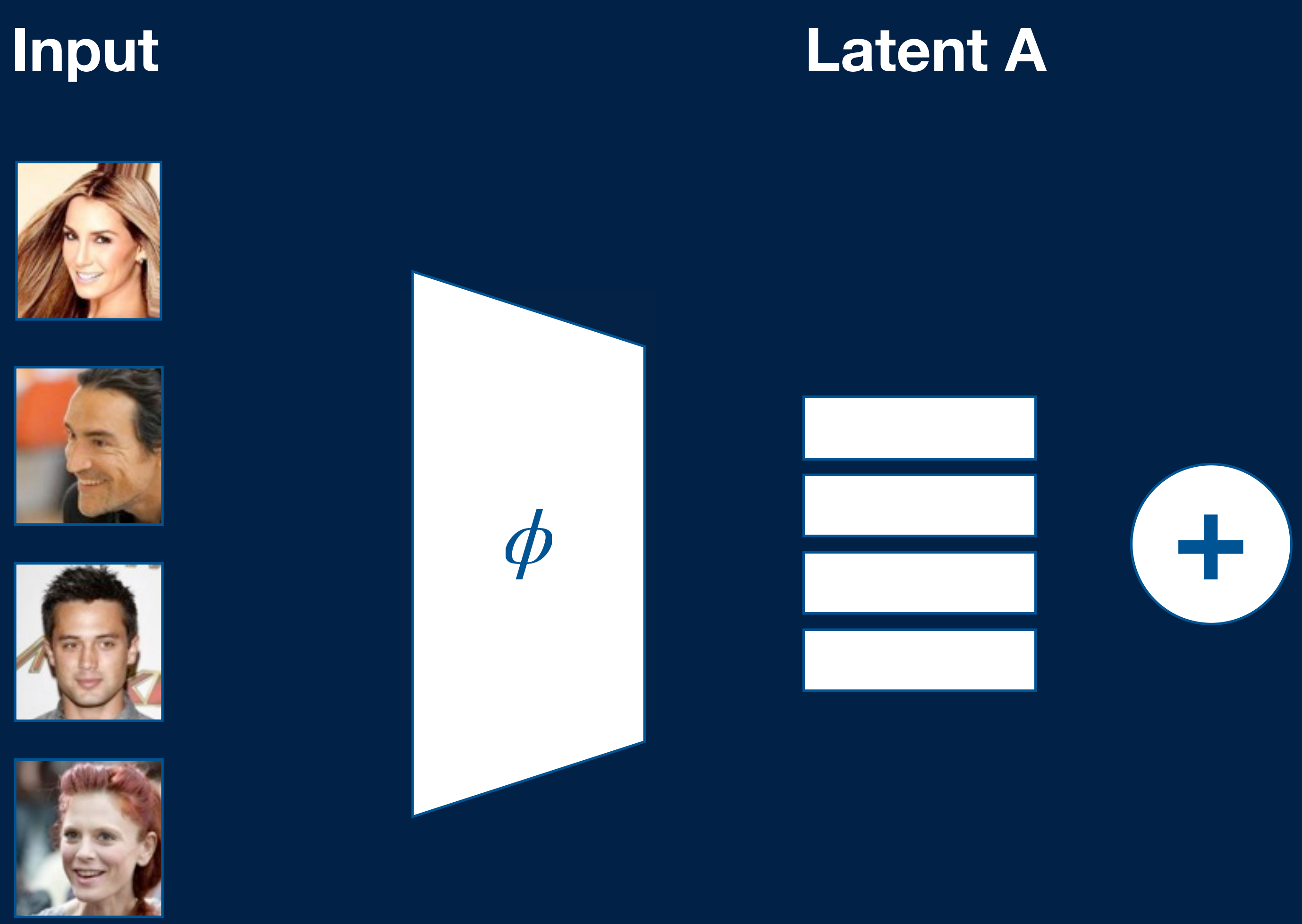


The deep sets architecture

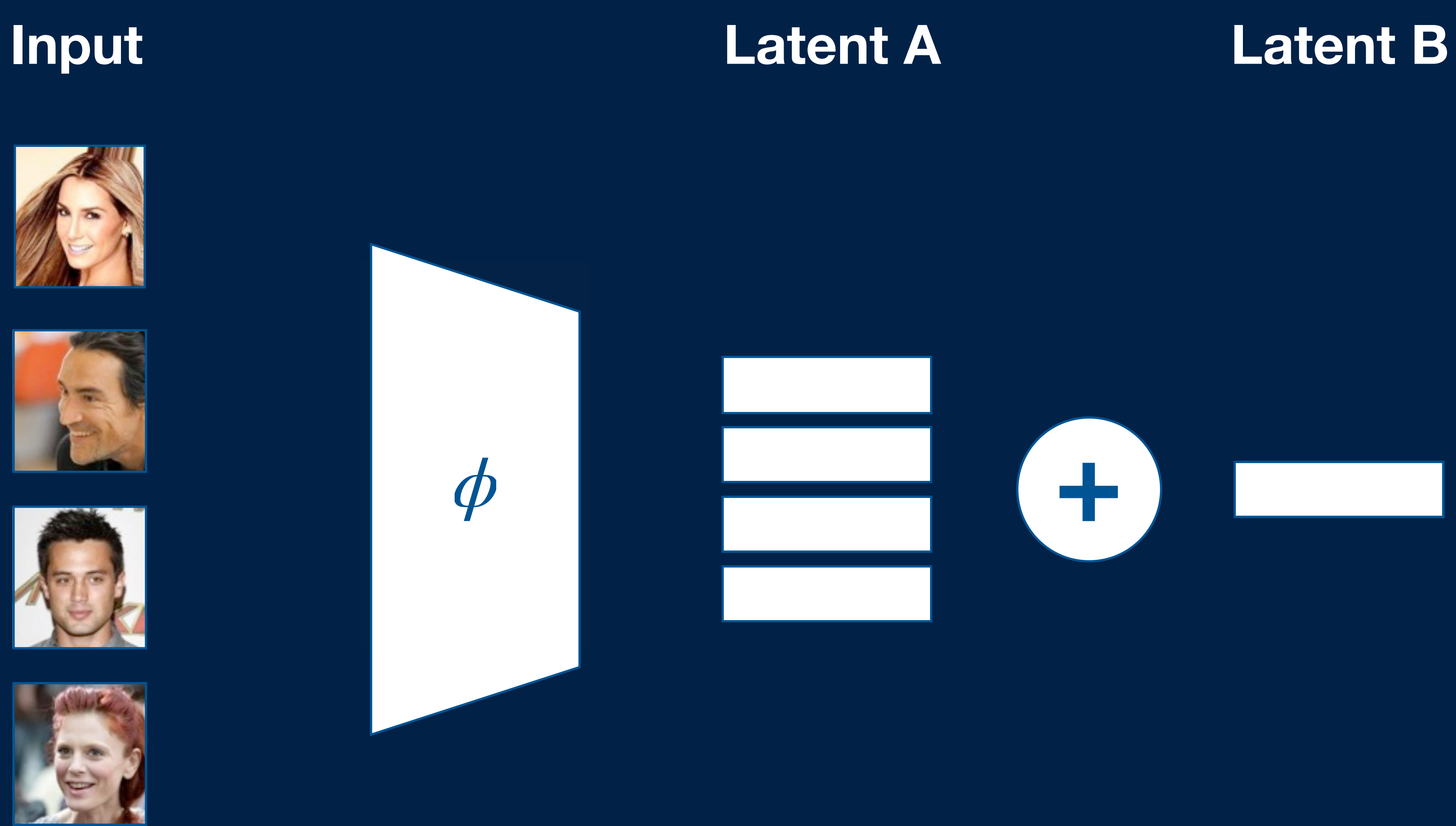
The deep sets architecture



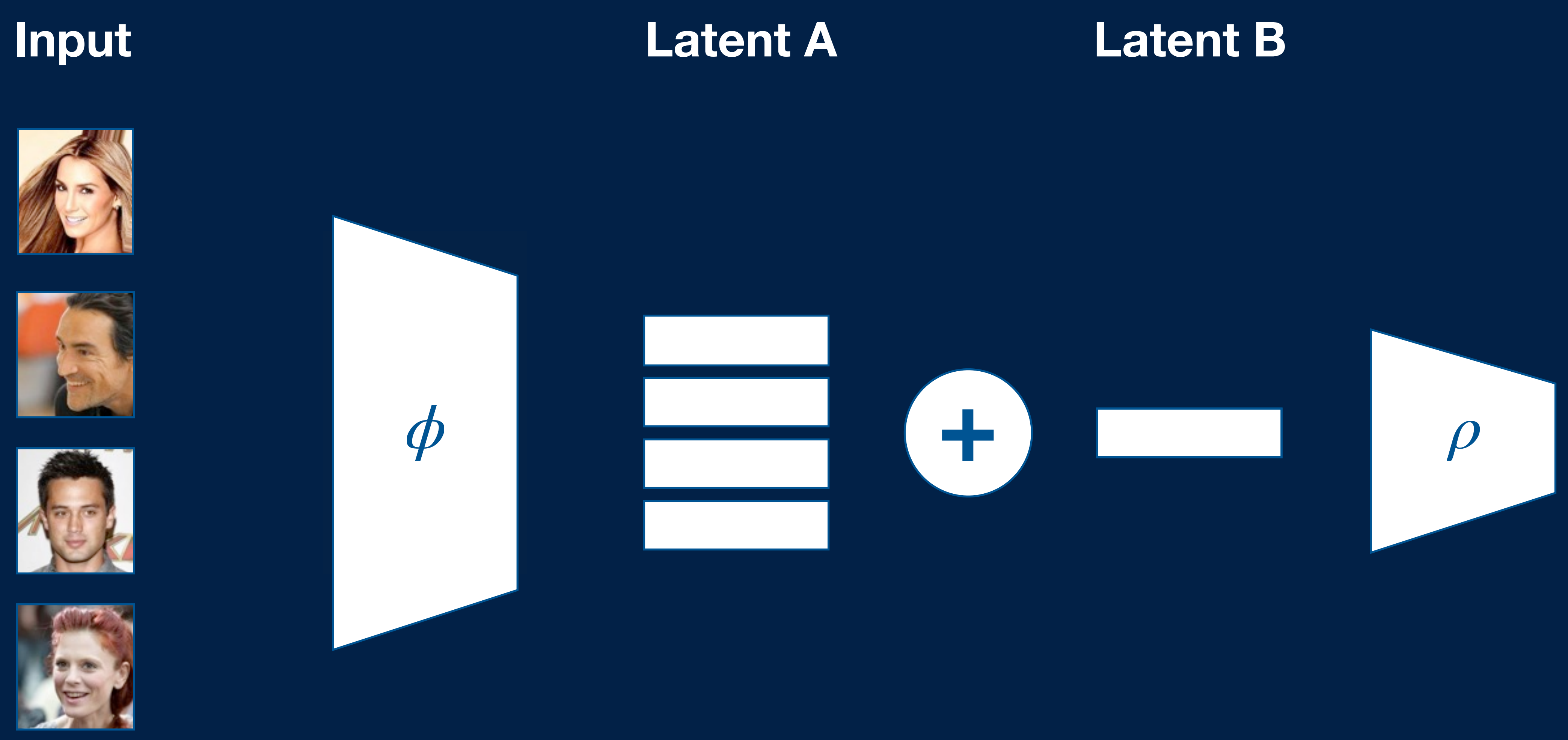
The deep sets architecture



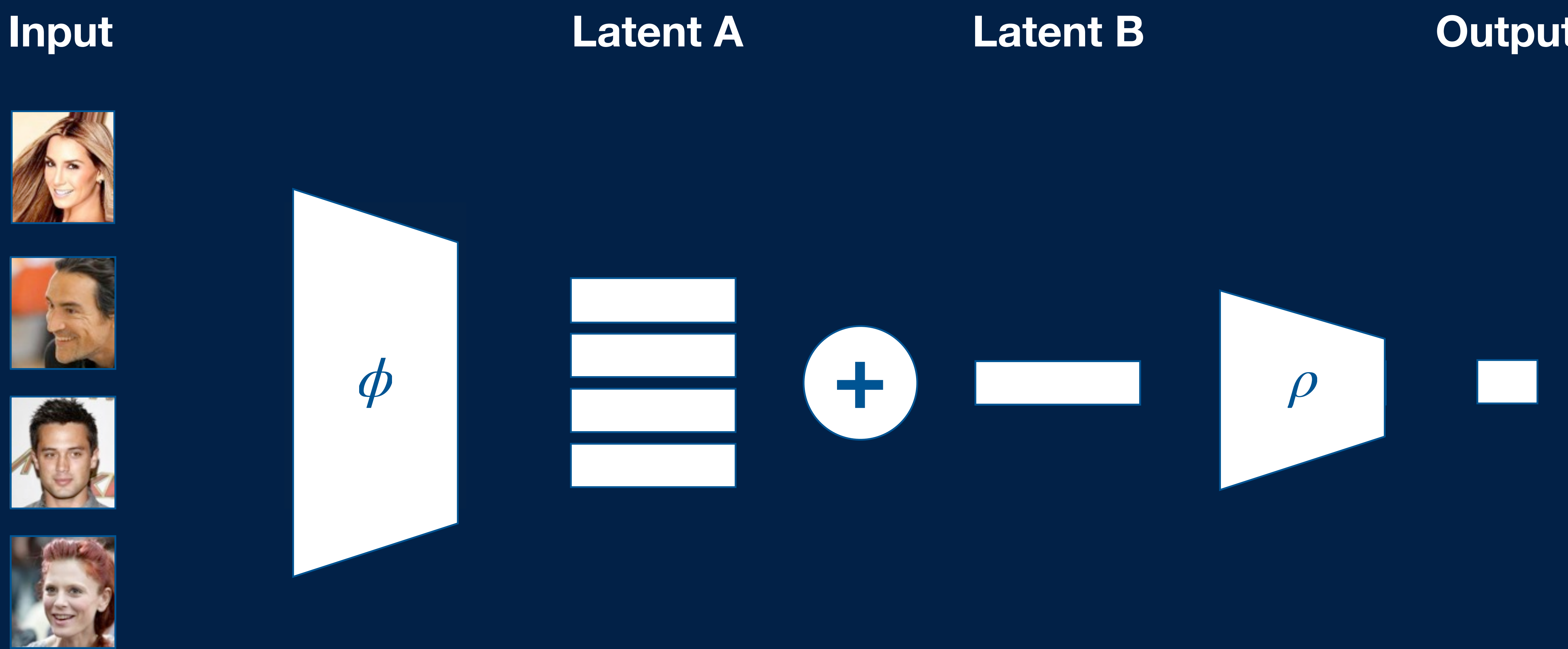
The deep sets architecture

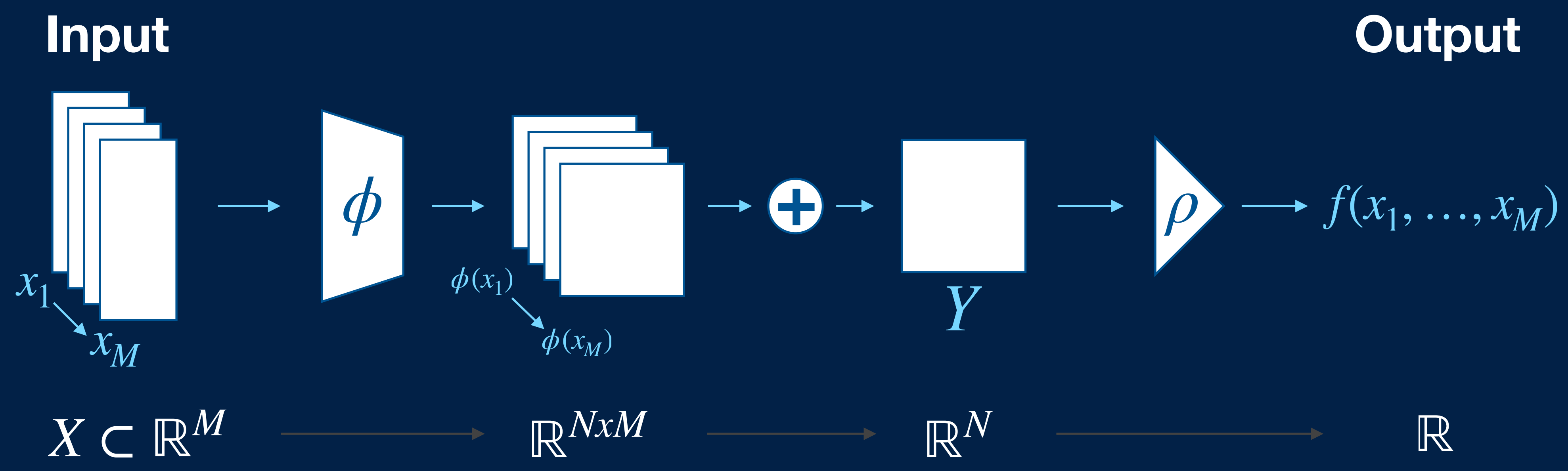


The deep sets architecture

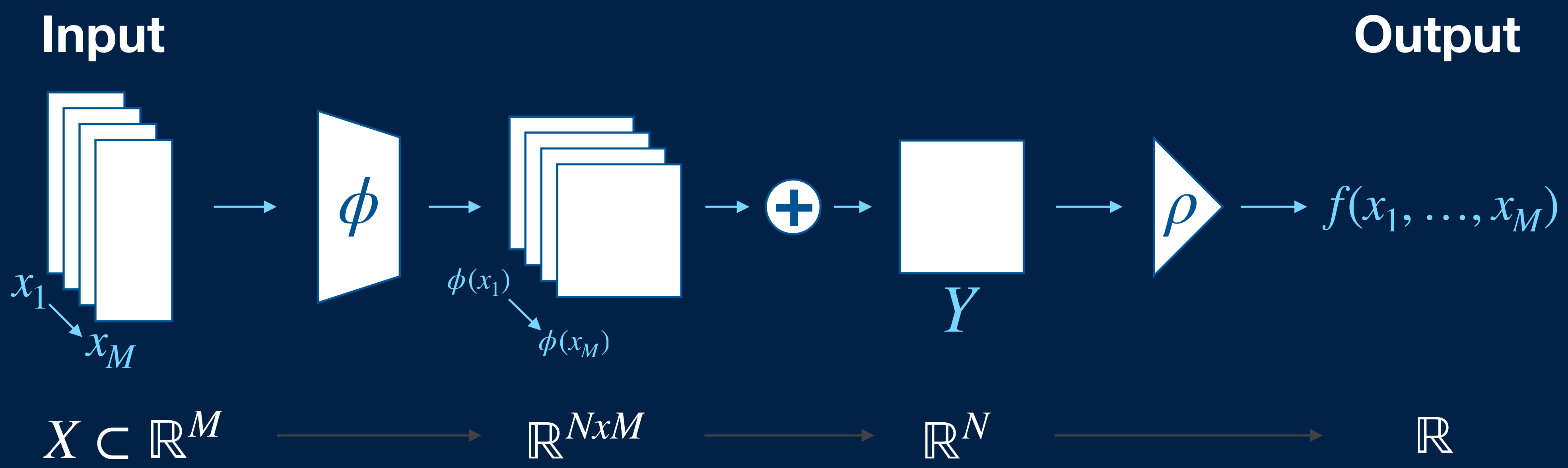


The deep sets architecture

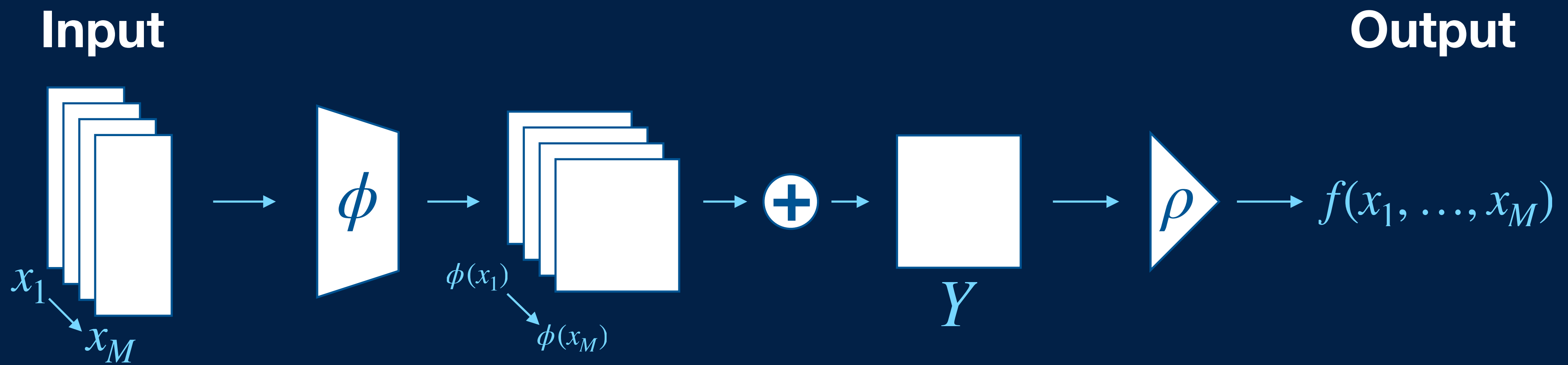




Input
Output
ϕ
ρ
x_1
x_M
$\phi(x_1)$
$\phi(x_M)$
Y
$f(x_1, \ldots, x_M)$
$X \subset \mathbb{R}^M$
$\mathbb{R}^{NxM}$
$\mathbb{R}^N$
$\mathbb{R}$



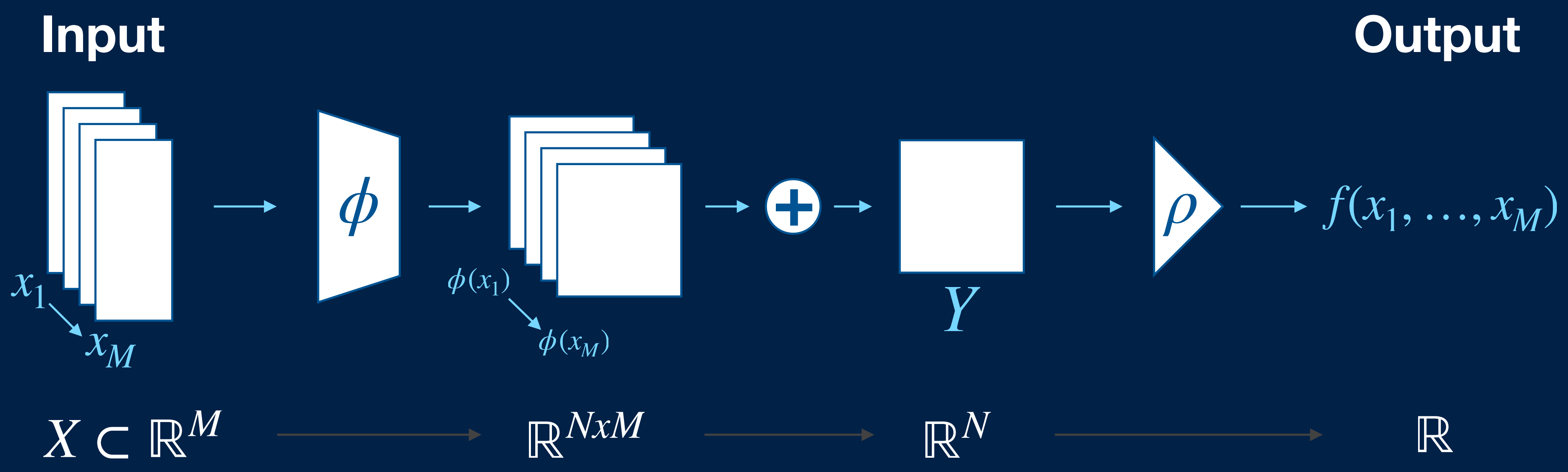
Theorem 1 (Zaheer et al.): *This architecture can successfully model any permutation invariant function, even for latent dimension N=1.*



$X \subset \mathbb{R}^M \longrightarrow \mathbb{R}^{NxM} \longrightarrow \mathbb{R}^N \longrightarrow \mathbb{R}$

Theorem 1 (Zaheer et al.): *This architecture can successfully model any permutation invariant function, even for latent dimension N=1.*

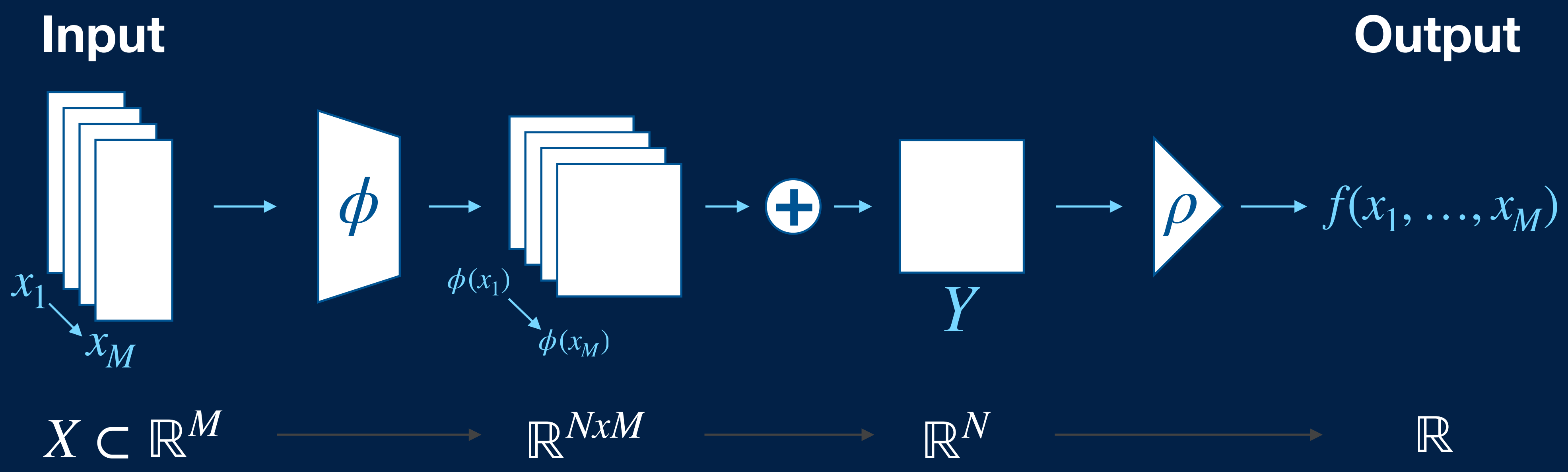
Proof



Theorem 1 (Zaheer et al.): *This architecture can successfully model any permutation invariant function, even for latent dimension N=1.*

Proof

Assume that neural networks Φ and ρ are universal function approximators



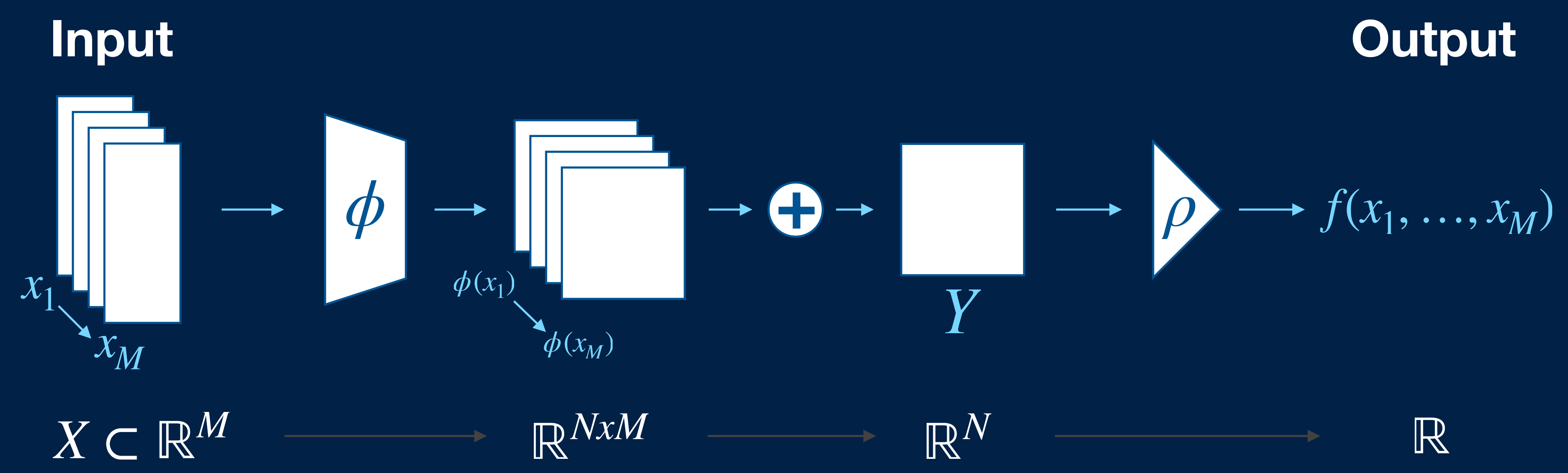
Theorem 1 (Zaheer et al.): *This architecture can successfully model any permutation invariant function, even for latent dimension N=1.*

Proof

Assume that neural networks Φ and ρ are universal function approximators

&

Find a Φ such that mapping from input set X to latent representation Y is injective



Theorem 1 (Zaheer et al.): *This architecture can successfully model any permutation invariant function, even for latent dimension N=1.*

Proof

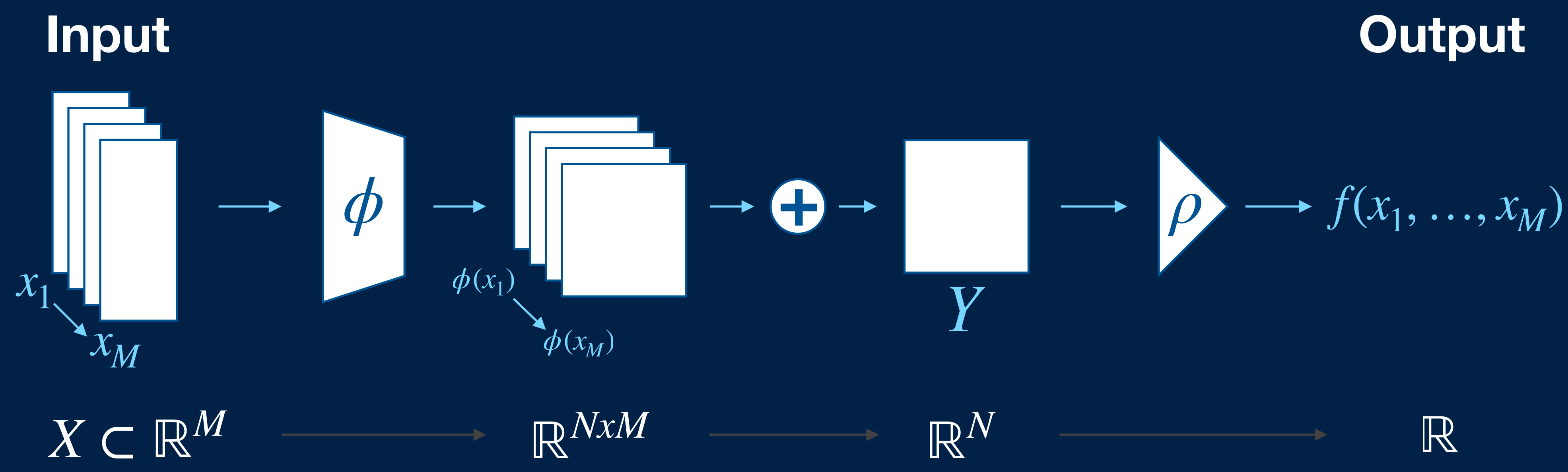
Assume that neural networks Φ and ρ are universal function approximators

&

Find a Φ such that mapping from input set X to latent representation Y is injective

→

Everything can be modelled □



Theorem 1 (Zaheer et al.): *This architecture can successfully model any permutation invariant function, even for latent dimension N=1.*

Proof

Assume that neural networks Φ and ρ are universal function approximators

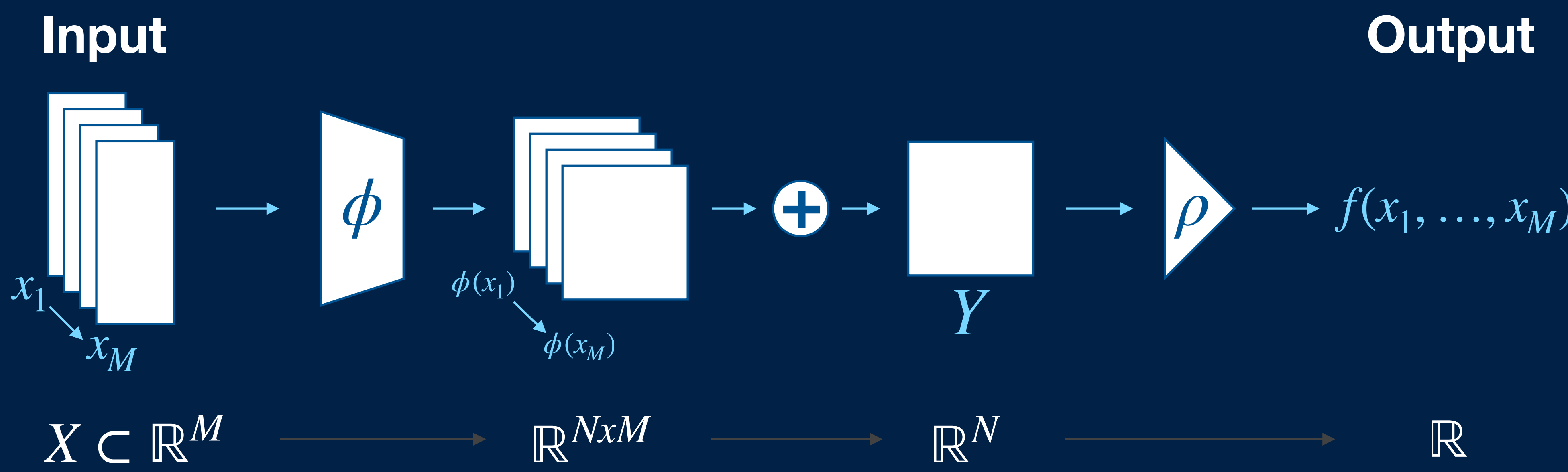
&

Find a Φ such that mapping from input set X to latent representation Y is injective

→

Everything can be modelled □

define $c(x) : \mathbb{Q} \to \mathbb{N}$



Theorem 1 (Zaheer et al.): *This architecture can successfully model any permutation invariant function, even for latent dimension N=1.*

Proof

Assume that neural networks Φ and ρ are universal function approximators

&

Find a Φ such that mapping from input set X to latent representation Y is injective

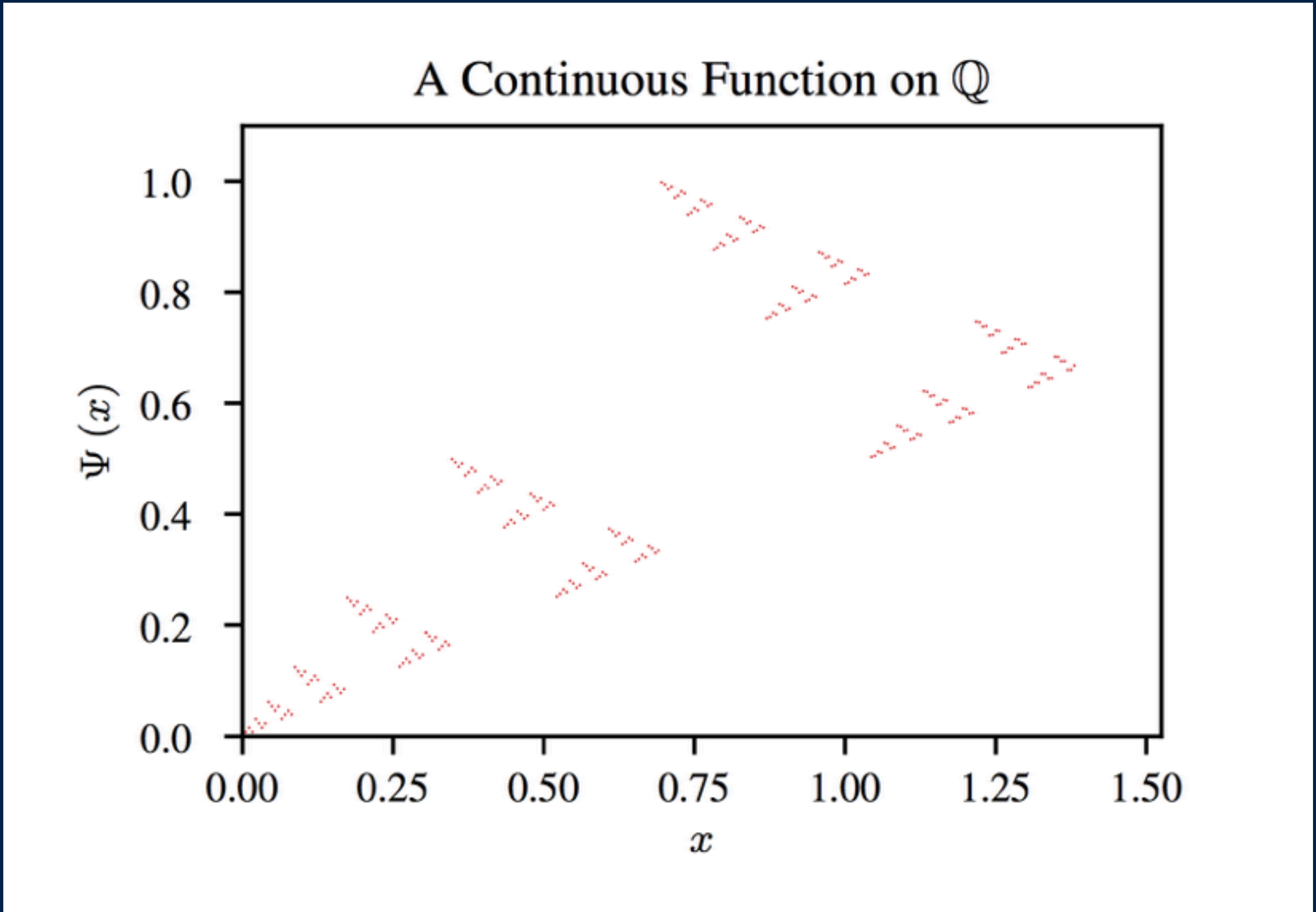
→

Everything can be modelled □

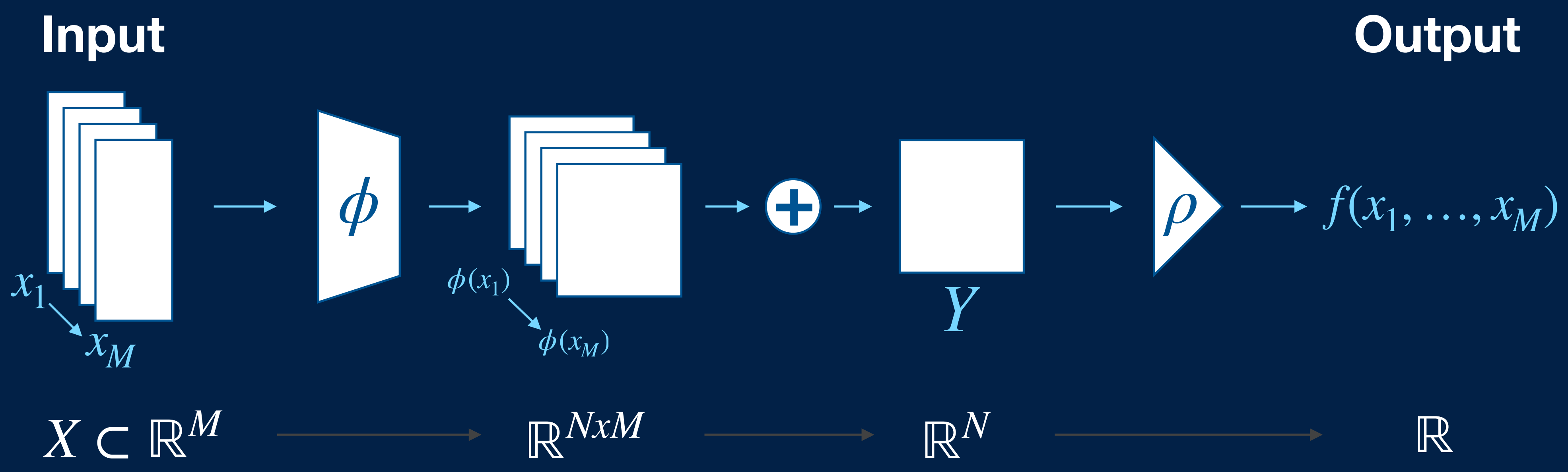
define $c(x) : \mathbb{Q} \rightarrow \mathbb{N}$

then define $\phi(x) = 2^{c(x)}$

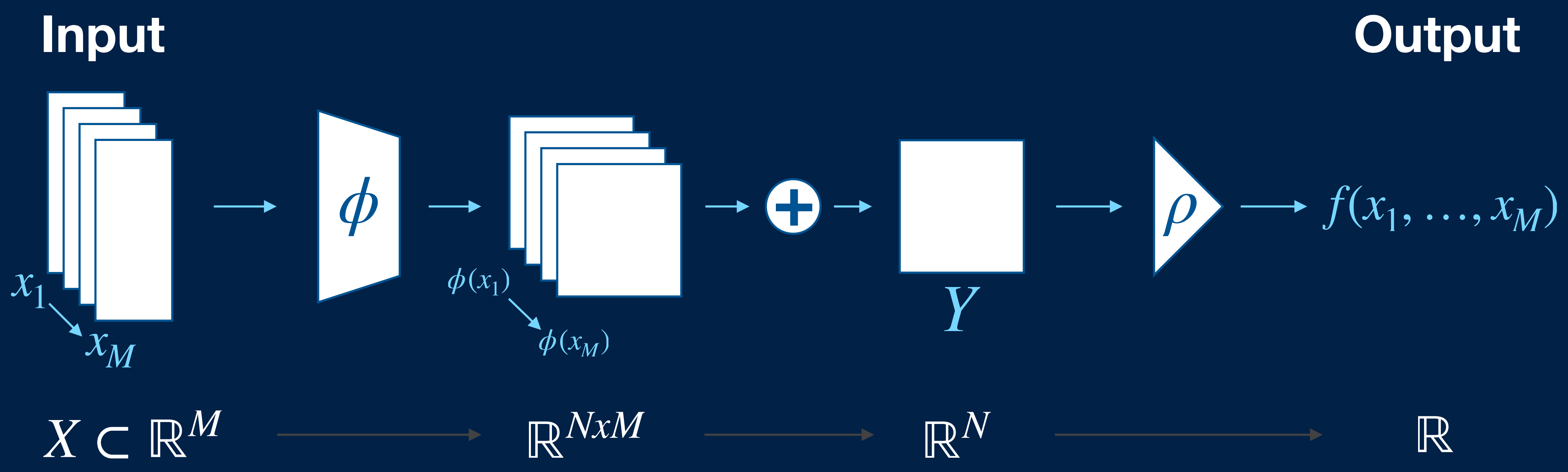
Role of Continuity



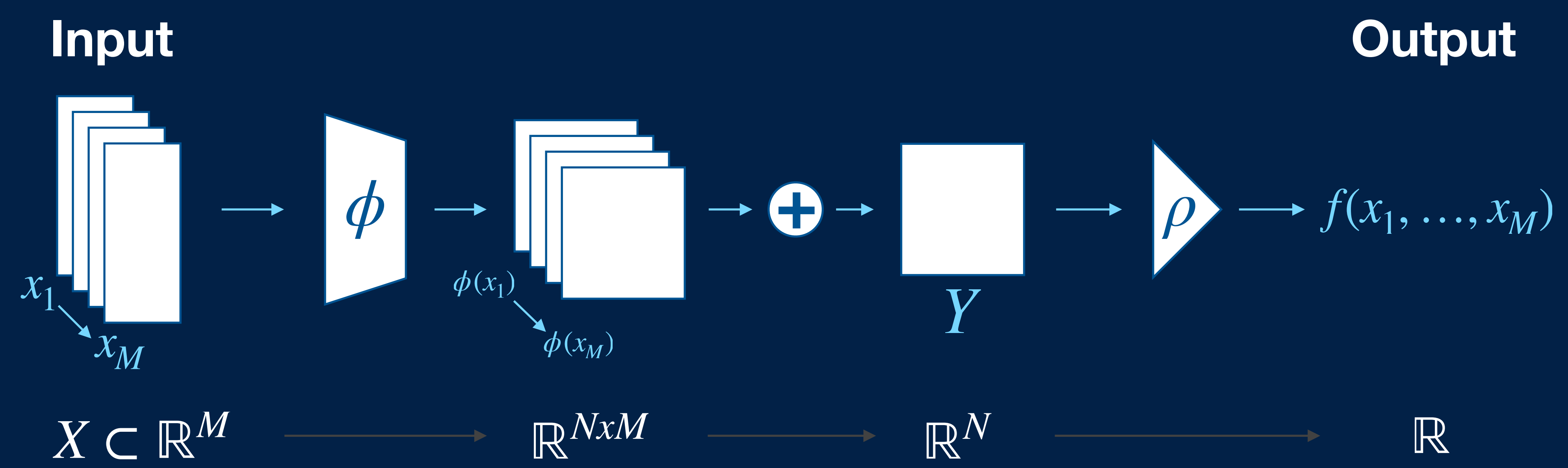
→ We need to take real numbers into account!



Input
Output
ϕ
ρ
x_1
x_M
$\phi(x_1)$
$\phi(x_M)$
Y
$f(x_1, \ldots, x_M)$
$X \subset \mathbb{R}^M$
$\mathbb{R}^{NxM}$
$\mathbb{R}^N$
$\mathbb{R}$

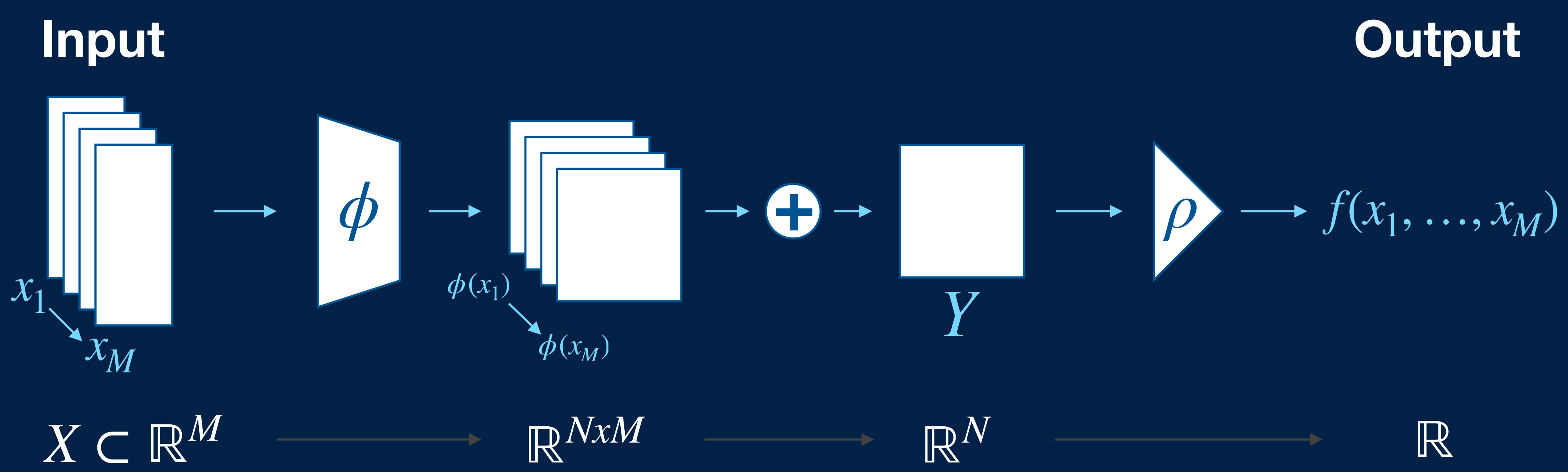


Theorem 2: *If we want to model all permutation invariant functions, it is sufficient and necessary that the latent dimension N is at least as large as the maximum input set size M.*



Theorem 2: *If we want to model all permutation invariant functions, it is sufficient and necessary that the latent dimension N is at least as large as the maximum input set size M.*

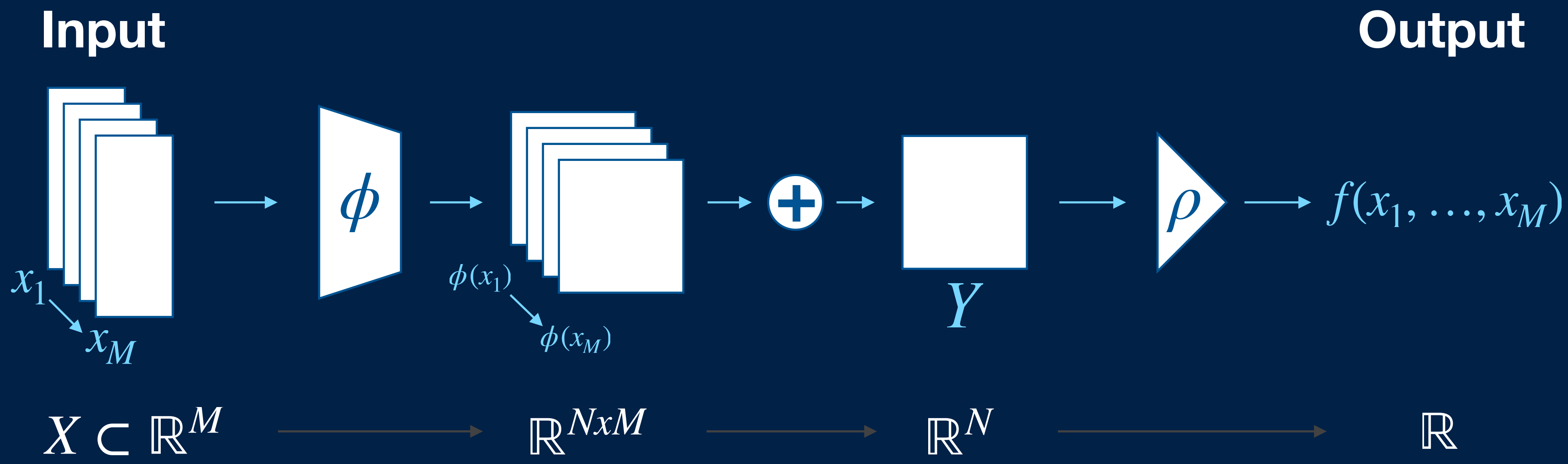
Sketch of Proof for Necessity



Theorem 2: *If we want to model all permutation invariant functions, it is sufficient and necessary that the latent dimension N is at least as large as the maximum input set size M.*

Sketch of Proof for Necessity

To prove necessity, we only need one function which can't be decomposed with *N<M*. We pick *max(X)*.



Theorem 2: *If we want to model all permutation invariant functions, it is sufficient and necessary that the latent dimension N is at least as large as the maximum input set size M.*

Sketch of Proof for Necessity

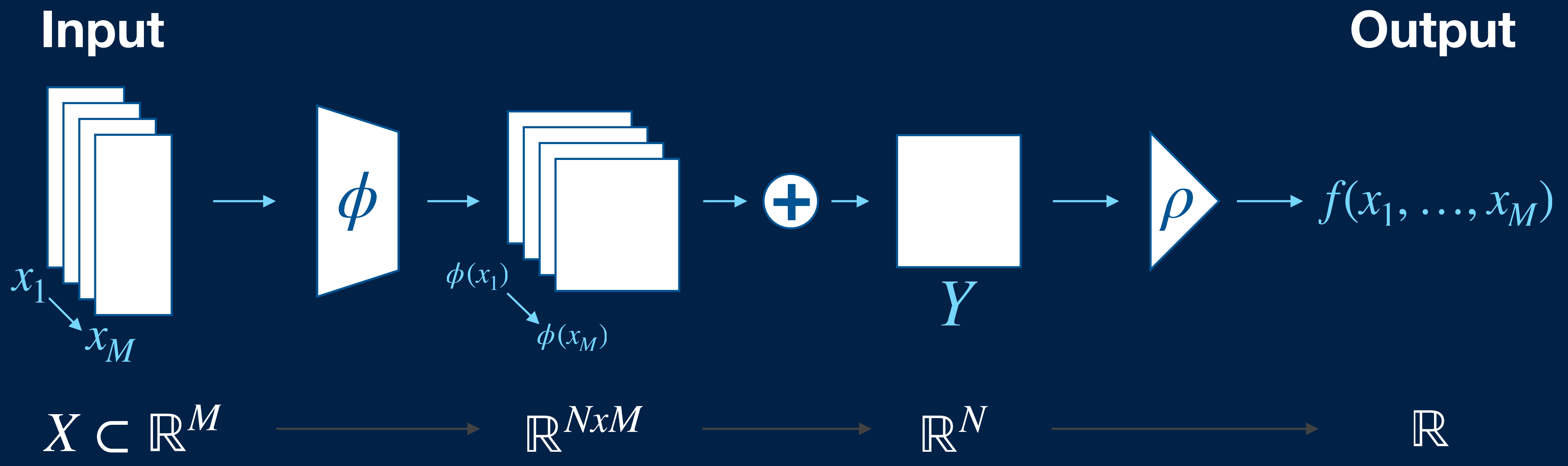
To prove necessity, we only need one function which can't be decomposed with *N<M*. We pick *max(X)*.

→

We show that, in order to represent *max(X)*,

$$\Phi(X) = \sum_x \phi(x)$$

needs to be injective



Theorem 2: *If we want to model all permutation invariant functions, it is sufficient and necessary that the latent dimension N is at least as large as the maximum input set size M.*

Sketch of Proof for Necessity

To prove necessity, we only need one function which can't be decomposed with *N<M*. We pick *max(X)*.

We show that, in order to represent *max(X)*,

$$\Phi(X) = \sum_x \phi(x)$$

needs to be injective

This is not possible with *N<M* □

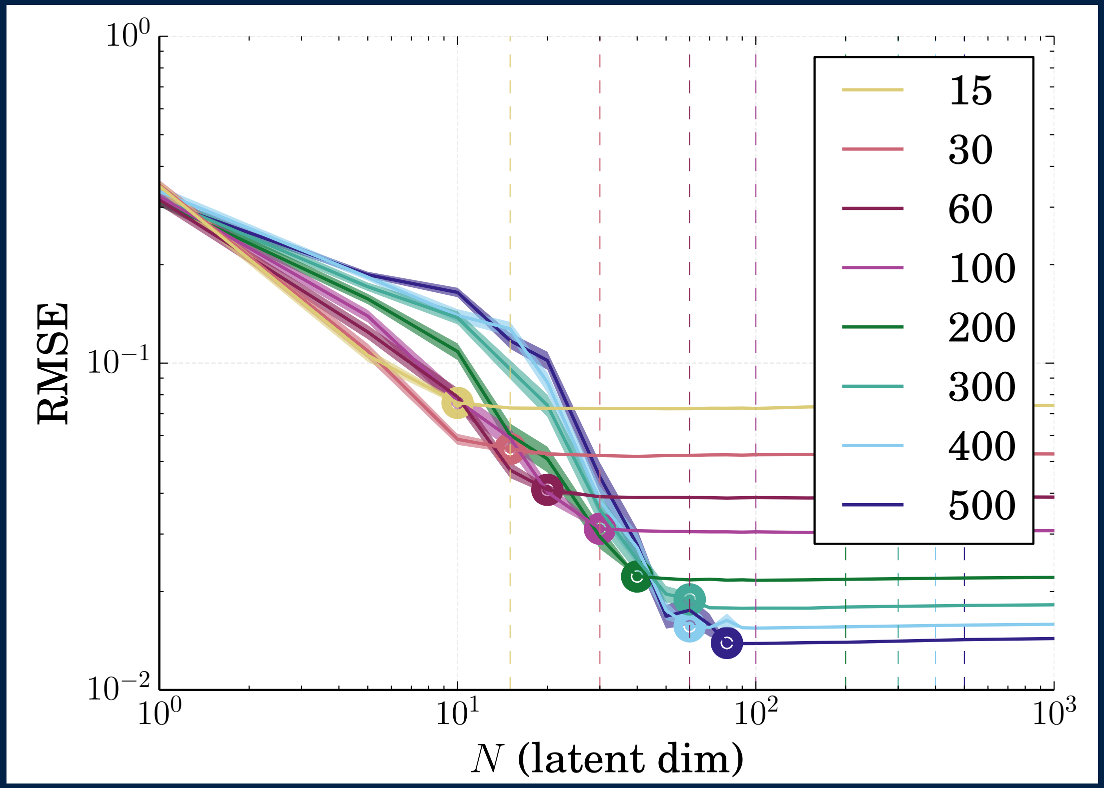
Illustrative Example: Regressing to the Median

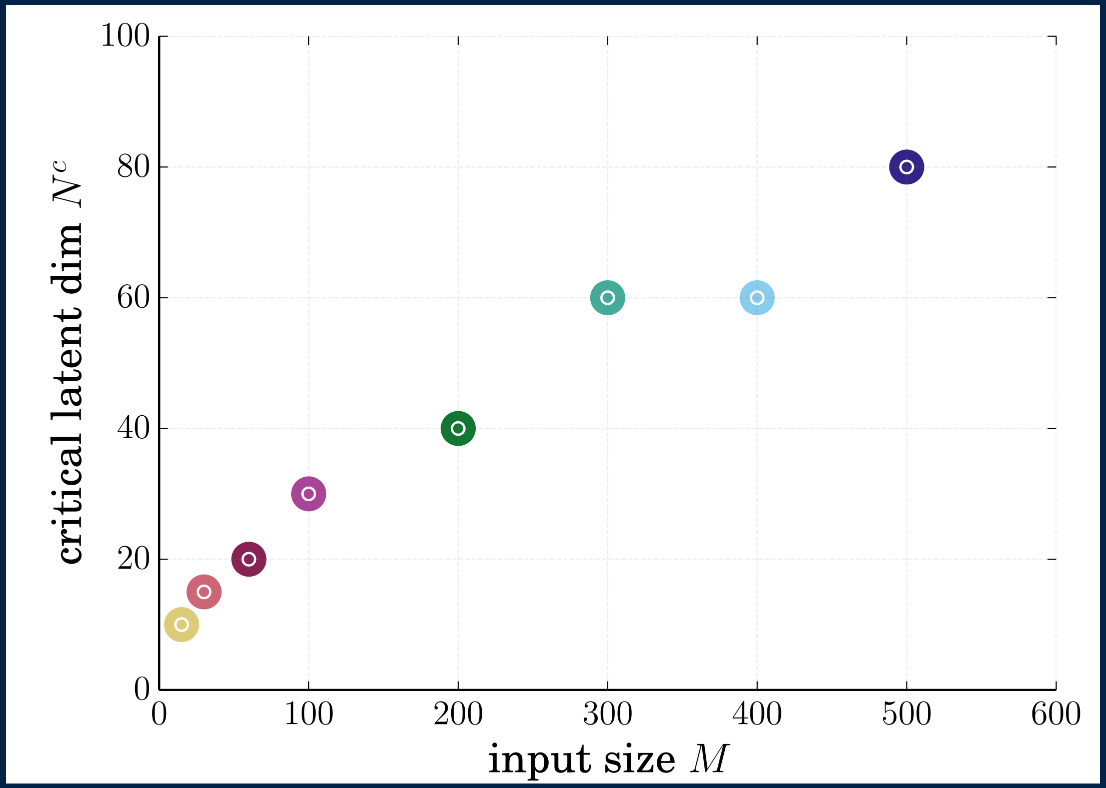
$$\{0.1,\ 0.6,\ -0.32,\ 1.61,\ 0.5,\ 0.67,\ 0.3\}$$

Illustrative Example: Regressing to the Median

$$\{0.1,\ \ 0.6,\ \ -0.32,\ \ 1.61,\ \ 0.5,\ \ 0.67,\ \ 0.3\}$$

Illustrative Example: Regressing to the Median





Thank You