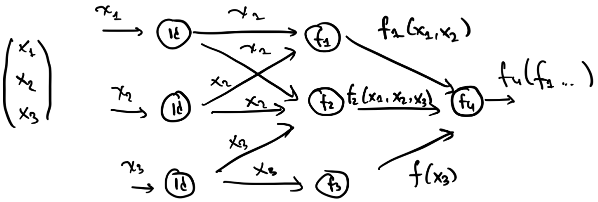
Quiver representations of neural networks

Rubén Ballester Bautista
Topological Machine Learning @ UB
November 29, 2022

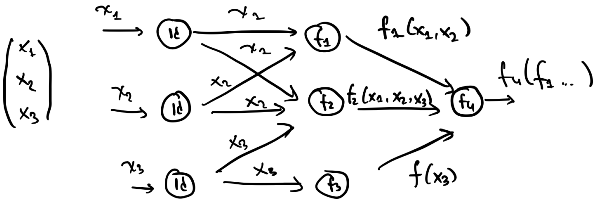
Neural networks formalism

- Neural networks are defined as computational graphs, i.e. functions $f : \mathbb{R}^n \times \Theta \to \mathbb{R}^m$ given by the composition of multiple **smaller** functions determined by the vertices of the graph.



Neural networks formalism

- Neural networks are defined as computational graphs, i.e. functions $f : \mathbb{R}^n \times \Theta \to \mathbb{R}^m$ given by the composition of multiple **smaller** functions determined by the vertices of the graph.



- Computational graphs are **hard-to-study** mathematical structures. We would like to have some easier, algebraic representations of neural networks.

Motivation to use quivers

- **Quivers** generalise the concept of graphs.

Motivation to use quivers

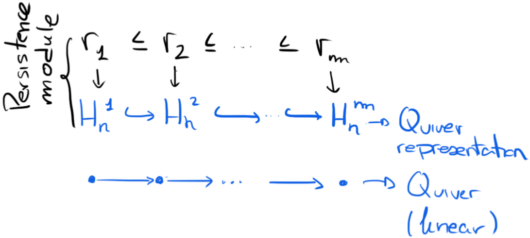
UNIVERSITAT DE BARCELONA

- **Quivers** generalise the concept of graphs.
- The field of **representation theory** has strong results for quiver representations.

Motivation to use quivers

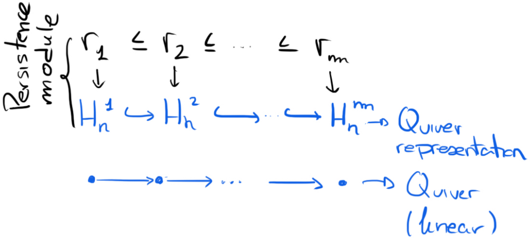
UNIVERSITAT DE BARCELONA

- **Quivers** generalise the concept of graphs.
- The field of **representation theory** has strong results for quiver representations.
- Decomposition of persistence modules into **persistence barcodes** is proved by quiver representation theory!



Motivation to use quivers

- **Quivers** generalise the concept of graphs.
- The field of **representation theory** has strong results for quiver representations.
- Decomposition of persistence modules into **persistence barcodes** is proved by quiver representation theory!



- Can neural networks benefit from quiver theory?

Basics of quiver representation theory

Quiver theory for neural networks

Quiver theory for data representations

About the talk

- We will explore mainly the paper **"The Representation Theory of Neural Networks"**[a], by Marco Antonio Armenta and Pierre-Marc Jodoin.

[a] Marco Armenta and Pierre-Marc Jodoin. "The representation theory of neural networks". In: *Mathematics* 9.24 (2021), p. 3216.

About the talk

- We will explore mainly the paper **"The Representation Theory of Neural Networks"**[a], by Marco Antonio Armenta and Pierre-Marc Jodoin.
- Several papers have been developed studying neural networks theoretically:
  - **Neural teleportation**, by Armenta et al. Study of neural network *isomorphisms* and consequences during training.
  - **Double Framed Moduli Spaces of Quiver Representations**, by Armenta et al. Definition of the neural network category. Further advances in neural network representation theory.

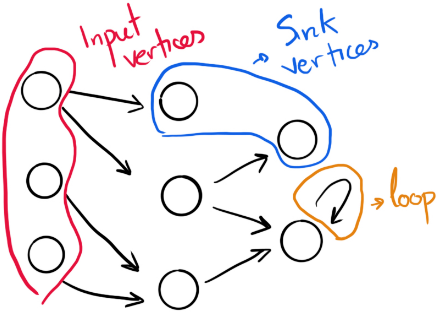
[a] Marco Armenta and Pierre-Marc Jodoin. "The representation theory of neural networks". In: *Mathematics* 9.24 (2021), p. 3216.

Basics of quiver representation theory

Quivers

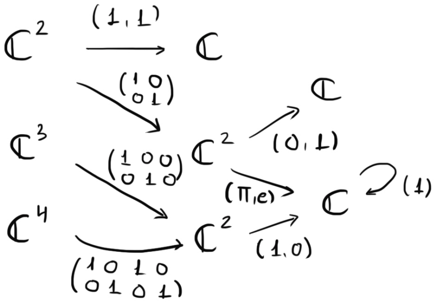
UNIVERSITAT DE BARCELONA

- A **quiver** Q is given by a tuple (V, E, s, t) where (V, E) is an oriented graph with a set of vertices V and a set of oriented edges E and maps $s, t\colon E \to V$ that send $\epsilon \in E$ to their source and target vertices $s(\epsilon) \in V$ and $t(\epsilon) \in V$ respectively.



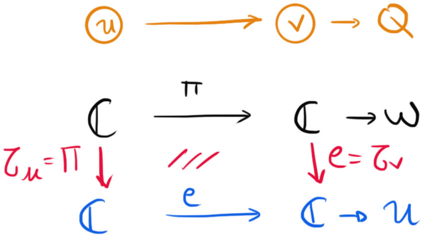
Quiver representations

- A **quiver representation** of Q is given by a pair of sets $W := ((W_v)_{v \in V}, (W_e)_{e \in E})$ where the W_v's are vector spaces and the W_e 's are linear maps such that $W_e \colon W_{s(e)} \to W_{t(e)}$ for every $e \in E$.



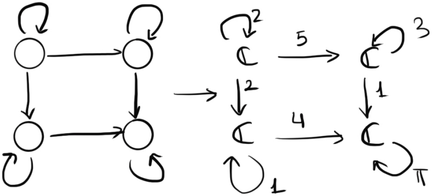
Representation morphisms

- Let Q be a quiver and let W and U be two representations of Q. A **morphism of representations** $\tau\colon W \to U$ is a set of linear maps $\tau = (\tau_v)_{v \in V}$ indexed by the vertices of Q, $\tau_v\colon W_v \to U_v$, such that $\tau_{t(e)} W_e = U_e \tau_{s(e)}$ for every $e \in E$. It is an **isomorphism** if τ_v is an isomorphism for all $v \in V(Q)$.



Thin representation of a quiver

- A **thin** representation of a quiver Q is a quiver representation W such that $W_v = \mathbb{C}$ for all $v \in V$.



Actions and change of basis group

UNIVERSITAT DE BARCELONA

- Let G be a group and let X be a set. We say that there is an **action** of G on X if there exists a map $\cdot\colon G \times X \to X$ such that $e \cdot x = x$ and $a \cdot (b \cdot x) = (ab) \cdot x$.

Actions and change of basis group

UNIVERSITAT DE BARCELONA

- Let G be a group and let X be a set. We say that there is an **action** of G on X if there exists a map $\cdot\colon G \times X \to X$ such that $e \cdot x = x$ and $a \cdot (b \cdot x) = (ab) \cdot x$.
- The **change of basis group** of thin representations over a quiver Q is $G = \prod_{v \in V} \mathbb{C}^*$ where $\tau\sigma = (\tau_1\sigma_1, \ldots, \tau_n\sigma_n)$, $n = |V(Q)|$.

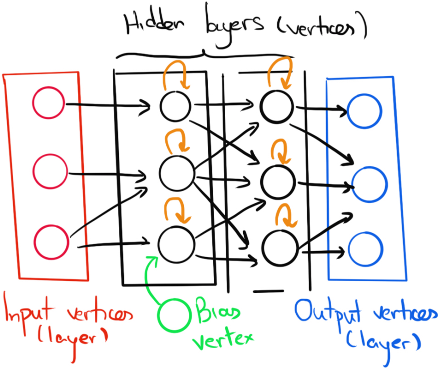
Actions and change of basis group

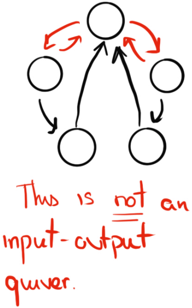
- Let G be a group and let X be a set. We say that there is an **action** of G on X if there exists a map $\cdot : G \times X \to X$ such that $e \cdot x = x$ and $a \cdot (b \cdot x) = (ab) \cdot x$.
- The **change of basis group** of thin representations over a quiver Q is $G = \prod_{v \in V} \mathbb{C}^*$ where $\tau\sigma = (\tau_1\sigma_1, \dots, \tau_n\sigma_n)$, $n = |V(Q)|$.
- G will act on the set X of thin representations of Q.

Quiver theory for neural networks

Input-output quivers

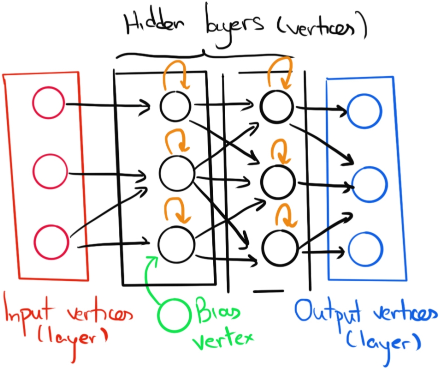
- An **input-output** quiver Q is a quiver where a subset of d source vertices of Q are called **input** vertices.

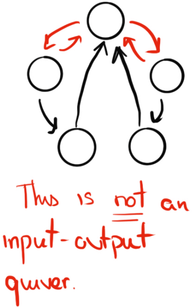




Input-output quivers

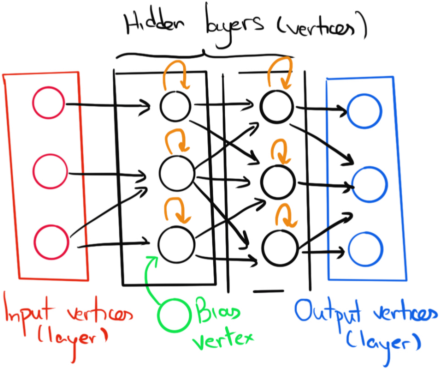
- An **input-output** quiver Q is a quiver where a subset of d source vertices of Q are called **input** vertices.
- Non-input source vertices are called **bias**.
- Sinks are called **output** vertices.

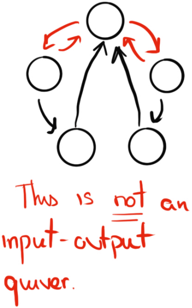




Input-output quivers

- An **input-output** quiver Q is a quiver where a subset of d source vertices of Q are called **input** vertices.
- Non-input source vertices are called **bias**.
- Sinks are called **output** vertices.
- It is **arranged by layers** if it is arranged by layers in the known, natural way.





Network quivers

- A **network quiver** Q is an input-output quiver arranged by layers such that:
  1. There are no loops on source vertices (input and bias) nor sink (output) vertices.
  2. There is exactly one loop on each hidden vertex.

Delooped quivers

- The **delooped** quiver $\mathring{Q}$ of Q is the quiver obtained by removing all loops of Q and it is denoted by $\mathring{Q} = (\mathring{V}, \mathring{E}, \mathring{s}, \mathring{t})$.

Delooped quivers

UNIVERSITAT DE BARCELONA

- The **delooped** quiver $\mathring{Q}$ of Q is the quiver obtained by removing all loops of Q and it is denoted by $\mathring{Q} = (\mathring{V}, \mathring{E}, \mathring{s}, \mathring{t})$.
- The weights of a neural network define a thin quiver representation of the delooped quiver $\mathring{Q}$ of its network quiver Q.

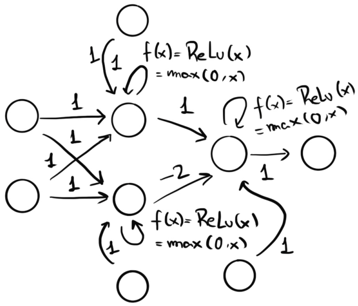
Delooped quivers

UNIVERSITAT DE BARCELONA

- The **delooped** quiver $\mathring{Q}$ of Q is the quiver obtained by removing all loops of Q and it is denoted by $\mathring{Q} = (\mathring{V}, \mathring{E}, \mathring{s}, \mathring{t})$.
- The weights of a neural network define a thin quiver representation of the delooped quiver $\mathring{Q}$ of its network quiver Q.
- Together with **activation functions** $f : \mathbb{C} \to \mathbb{C}$ we can obtain the neural network function $f(x, \theta)$.

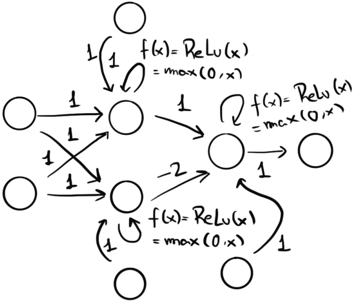
Neural networks

- A **neural network** over a network quiver Q is a pair (W, f) where W is a thin representation of the delooped quiver $\mathring{Q}$ and $f = (f_v)_{v \in V}$ are activation functions, assigned to the loops of Q.



Neural networks

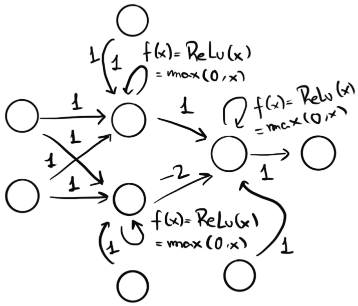
- A **neural network** over a network quiver Q is a pair (W, f) where W is a thin representation of the delooped quiver $\mathring{Q}$ and $f = (f_v)_{v \in V}$ are activation functions, assigned to the loops of Q.
- The weights of the neural network (W, f) are the complex numbers defining the maps W_e for all $e \in E$.



Neural networks

- A **neural network** over a network quiver Q is a pair (W, f) where W is a thin representation of the delooped quiver $\mathring{Q}$ and $f = (f_v)_{v \in V}$ are activation functions, assigned to the loops of Q.
- The weights of the neural network (W, f) are the complex numbers defining the maps W_e for all $e \in E$.

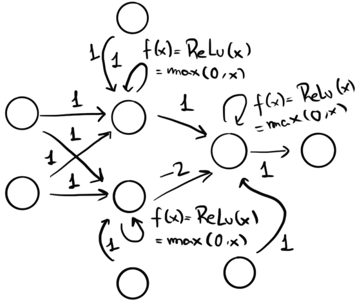
- The **activation output** of $v \in V(Q)$ in x, $a(W, f)_v(x)$, is:



Neural networks

- A **neural network** over a network quiver Q is a pair (W, f) where W is a thin representation of the delooped quiver $\mathring{Q}$ and $f = (f_v)_{v \in V}$ are activation functions, assigned to the loops of Q.
- The weights of the neural network (W, f) are the complex numbers defining the maps W_e for all $e \in E$.

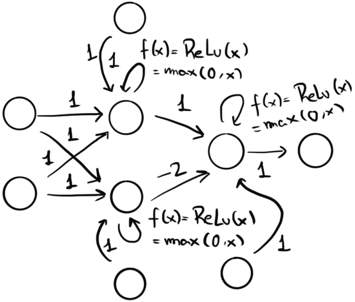
- The **activation output** of $v \in V(Q)$ in x, $a(W, f)_v(x)$, is:
  - If v input, $a(W, f)_v(x) = x_v$.



Neural networks

- A **neural network** over a network quiver Q is a pair (W, f) where W is a thin representation of the delooped quiver $\mathring{Q}$ and $f = (f_v)_{v \in V}$ are activation functions, assigned to the loops of Q.
- The weights of the neural network (W, f) are the complex numbers defining the maps W_e for all $e \in E$.

- The **activation output** of $v \in V(Q)$ in x, $a(W, f)_v(x)$, is:
  - If v input, $a(W, f)_v(x) = x_v$.
  - If v bias, $a(W, f)_v(x) = 1$.

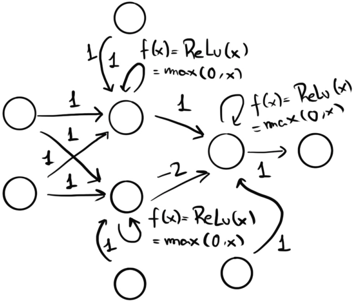


Neural networks

Universitat de Barcelona

- A **neural network** over a network quiver Q is a pair (W, f) where W is a thin representation of the delooped quiver $\mathring{Q}$ and $f = (f_v)_{v \in V}$ are activation functions, assigned to the loops of Q.
- The weights of the neural network (W, f) are the complex numbers defining the maps W_e for all $e \in E$.

- The **activation output** of $v \in V(Q)$ in x, $a(W, f)_v(x)$, is:
  - If v input, $a(W, f)_v(x) = x_v$.
  - If v bias, $a(W, f)_v(x) = 1$.
  - If v hidden vertex, $a(W, f)v(x) = f\left(\sum_{\alpha \in \zeta_v} W_\alpha a(W, f)_{s(\alpha)}(x)\right)$.

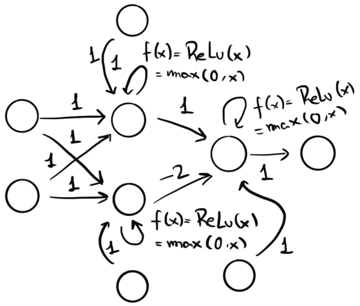


Neural networks

UNIVERSITAT DE BARCELONA

- A **neural network** over a network quiver Q is a pair (W, f) where W is a thin representation of the delooped quiver $\mathring{Q}$ and $f = (f_v)_{v \in V}$ are activation functions, assigned to the loops of Q.
- The weights of the neural network (W, f) are the complex numbers defining the maps W_e for all $e \in E$.

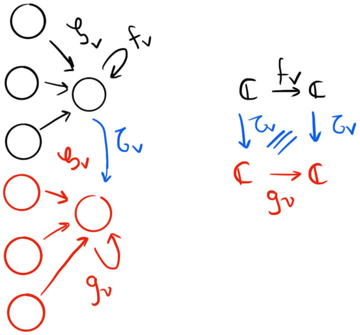
- The **activation output** of $v \in V(Q)$ in x, $a(W, f)_v(x)$, is:
  - If v input, $a(W, f)_v(x) = x_v$.
  - If v bias, $a(W, f)_v(x) = 1$.
  - If v hidden vertex, $a(W, f)v(x) = f\left(\sum_{\alpha \in \zeta_v} W_\alpha a(W, f)_{s(\alpha)}(x)\right)$.
  - If v output vertex, $a(W, f)v(x) = \sum_{\alpha \in \zeta_v} W_\alpha a(W, f)_{s(\alpha)}(x)$.



Neural network morphisms

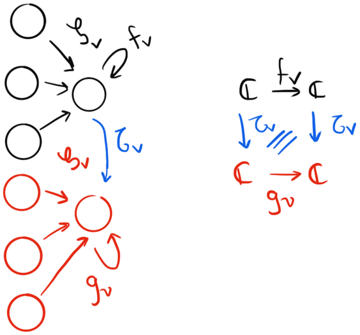
Universitat de Barcelona

- A(n) **(iso)morphism of neural networks** over the same quiver Q $\tau\colon (W, f) \to (V, g)$ is:



Neural network morphisms

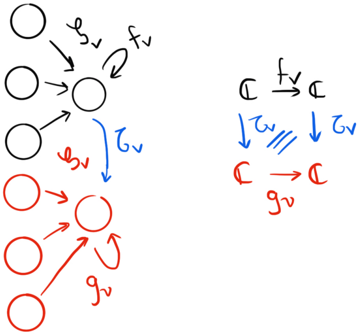
- A(n) **(iso)morphism of neural networks** over the same quiver Q $\tau\colon (W, f) \to (V, g)$ is:
  1. A(n) (iso)morphism of thin quiver representations $\tau\colon W \to V$,



Neural network morphisms

UNIVERSITAT DE BARCELONA

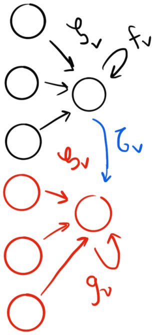
- A(n) **(iso)morphism of neural networks** over the same quiver Q $\tau\colon (W, f) \to (V, g)$ is:
  1. A(n) (iso)morphism of thin quiver representations $\tau\colon W \to V$,
  2. such that $\tau_v = 1$ for v not hidden vertex,

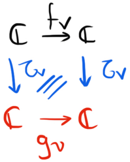


Neural network morphisms

UNIVERSITAT DE BARCELONA

- A(n) **(iso)morphism of neural networks** over the same quiver Q $\tau\colon (W,f) \to (V,g)$ is:
  1. A(n) (iso)morphism of thin quiver representations $\tau\colon W \to V$,
  2. such that $\tau_v = 1$ for v not hidden vertex,
  3. and such that $\forall v$ hidden vertex $\tau_v \circ f_v = g_v \circ \tau_v$.





Hidden quiver and group of change of basis

UNIVERSITAT DE BARCELONA

- The **hidden quiver** of Q denoted by $\bar{Q}$ is given by $\mathring{Q}$ but without the source and sink vertices.

Hidden quiver and group of change of basis

UNIVERSITAT DE BARCELONA

- The **hidden quiver** of Q denoted by $\bar{Q}$ is given by $\mathring{Q}$ but without the source and sink vertices.
- The **group of change of basis** for neural networks is $\bar{G} = \prod_{v \in \bar{V}} \mathbb{C}^*$.

Hidden quiver and group of change of basis

- The **hidden quiver** of Q denoted by $\bar{Q}$ is given by $\mathring{Q}$ but without the source and sink vertices.
- The **group of change of basis** for neural networks is $\bar{G} = \prod_{v \in \bar{V}} \mathbb{C}^*$.
- An element $\bar{\tau}$ induces an element $\tau \in G$ group of change of basis of thin representations $\mathring{Q}$ by setting $\tau_v = 1$ for all v not hidden vertices.

Hidden quiver and group of change of basis

UNIVERSITAT DE BARCELONA

- The **hidden quiver** of Q denoted by $\bar{Q}$ is given by $\mathring{Q}$ but without the source and sink vertices.
- The **group of change of basis** for neural networks is $\bar{G} = \prod_{v \in \bar{V}} \mathbb{C}^*$.
- An element $\bar{\tau}$ induces an element $\tau \in G$ group of change of basis of thin representations $\mathring{Q}$ by setting $\tau_v = 1$ for all v not hidden vertices.
- We define the action of $\tau \in \bar{G}$ in a neural network (W, f) as
  - $(\tau \cdot W)_e = W_e \frac{\tau_{t(e)}}{\tau_{s(e)}}$.
  - $(\tau \cdot f)_v(x) = \tau_v f\left(\frac{x}{\tau_v}\right)$.

Hidden quiver and group of change of basis

UNIVERSITAT DE BARCELONA

- The **hidden quiver** of Q denoted by $\bar{Q}$ is given by $\mathring{Q}$ but without the source and sink vertices.
- The **group of change of basis** for neural networks is $\bar{G} = \prod_{v \in \bar{V}} \mathbb{C}^*$.
- An element $\bar{\tau}$ induces an element $\tau \in G$ group of change of basis of thin representations $\mathring{Q}$ by setting $\tau_v = 1$ for all v not hidden vertices.
- We define the action of $\tau \in \bar{G}$ in a neural network (W, f) as
  - $(\tau \cdot W)_e = W_e \frac{\tau_{t(e)}}{\tau_{s(e)}}$.
  - $(\tau \cdot f)_v(x) = \tau_v f\left(\frac{x}{\tau_v}\right)$.
- **Theorem**: If $\tau \colon (W, f) \to (V, g)$ is an isomorphism of neural networks then their network functions $\psi(W, f) = \psi(V, g)$ coincide.

Implications and applications

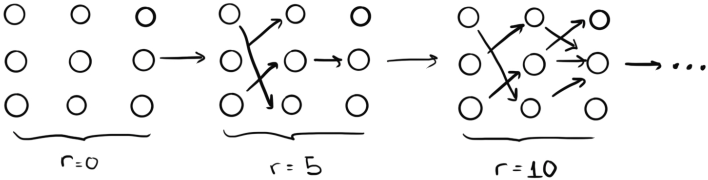
- There are an **infinite** number of neural networks with the same network function, independently of the architecture and the activation functions.

Implications and applications

- There are an **infinite** number of neural networks with the same network function, independently of the architecture and the activation functions.
- ReLU neural networks are **positive scale invariant**, i.e. ,multiplying weights by positive factors in an appropiate way do not change the network function nor the *architecture* (**infinite minima** in loss functions).

Implications and applications

- There are an **infinite** number of neural networks with the same network function, independently of the architecture and the activation functions.
- ReLU neural networks are **positive scale invariant**, i.e. ,multiplying weights by positive factors in an appropiate way do not change the network function nor the *architecture* (**infinite minima** in loss functions).
- Use of **persistent path homology** directly on the quiver representation of neural networks.



UNIVERSITAT DE BARCELONA

Quiver theory for data representations

Data representations (I)

- A **labeled dataset** is given by a finite set $D = \{(x_i, t_i)\} \subseteq \mathbb{C}^d \times \mathbb{C}^o$.

Data representations (I)

- A **labeled dataset** is given by a finite set $D = \{(x_i, t_i)\} \subseteq \mathbb{C}^d \times \mathbb{C}^o$.
- The **data quiver representation** of a neural network $\mathcal{N}$ over a quiver Q and example $x \in D$ is the thin representation of $\mathring{Q}$

$$\left(W_x^f\right)_e = \begin{cases} W_e x_{s(e)}, & \text{if } s(e) \text{ is an input vertex} \\ W_e, & \text{if } s(e) \text{ is an bias vertex} \\ W_e \frac{a(W,f)_{s(e)}(x)}{\sum_{\beta \in \zeta_{s(e)}} W_\beta \cdot a(W,f)_{s(\beta)}(x)}, & \text{if } s(e) \text{ is a hidden vertex,} \end{cases}$$

where $\zeta_{s(e)}$ is the set of oriented edges of Q with target $s(e)$.

Data representations (I)

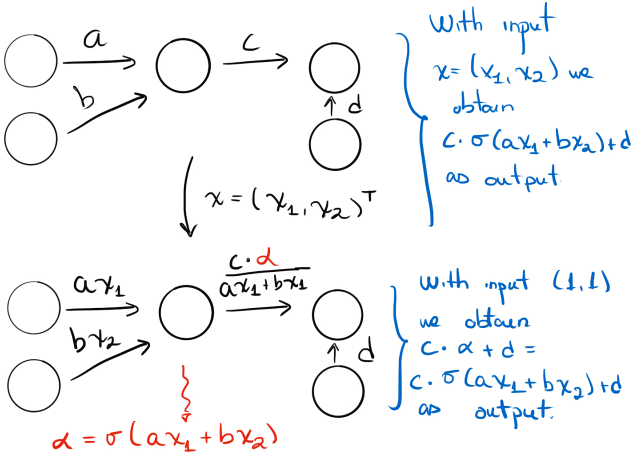
UNIVERSITAT DE BARCELONA

- A **labeled dataset** is given by a finite set $D = \{(x_i, t_i)\} \subseteq \mathbb{C}^d \times \mathbb{C}^o$.
- The **data quiver representation** of a neural network $\mathcal{N}$ over a quiver Q and example $x \in D$ is the thin representation of $\mathring{Q}$

$$\left(W_x^f\right)_e = \begin{cases} W_e x_{s(e)}, & \text{if } s(e) \text{ is an input vertex} \\ W_e, & \text{if } s(e) \text{ is an bias vertex} \\ W_e \frac{a(W,f)_{s(e)}(x)}{\sum_{\beta \in \zeta_{s(e)}} W_\beta \cdot a(W,f)_{s(\beta)}(x)}, & \text{if } s(e) \text{ is a hidden vertex,} \end{cases}$$

where $\zeta_{s(e)}$ is the set of oriented edges of Q with target $s(e)$.
- The neural network defined by W_x^f and identity activation functions satisfy $\Psi(W_x^f, 1)(1^d) = \Psi(W, f)(x)$ (**linear rectification!**).

Data representations (II)



Applications

- Isomorphism classes $\left[W_x^f\right] = \{\tau \cdot W_x^f : \tau \in \bar{G}\}$ induce **bad-behaved** Lie algebras.

Applications

UNIVERSITAT DE BARCELONA

- Isomorphism classes $\left[W_x^f\right] = \{\tau \cdot W_x^f : \tau \in \bar{G}\}$ induce **bad-behaved** Lie algebras.
- Under some technical transformations, we get what is called the **moduli space** of data of a neural network $\mathcal{N}$: **a smooth algebraic variety** coming from the previous isomorphism classes.

Applications

- Isomorphism classes $[W_x^f] = \{\tau \cdot W_x^f : \tau \in \bar{G}\}$ induce **bad-behaved** Lie algebras.
- Under some technical transformations, we get what is called the **moduli space** of data of a neural network $\mathcal{N}$: **a smooth algebraic variety** coming from the previous isomorphism classes.

Two possible outcomes:

Applications

- Isomorphism classes $\left[W_x^f\right] = \{\tau \cdot W_x^f : \tau \in \bar{G}\}$ induce **bad-behaved** Lie algebras.
- Under some technical transformations, we get what is called the **moduli space** of data of a neural network $\mathcal{N}$: **a smooth algebraic variety** coming from the previous isomorphism classes.

Two possible outcomes:

1. Theoretical and practical study of the adjacency matrices W_x^f for a dataset D and different neural networks. Do these **linear transformations** give us information about the general **robustness** of a neural network? Should we study the **topology of the set of linear transformations** induced by the neural network?

Applications

UNIVERSITAT DE BARCELONA

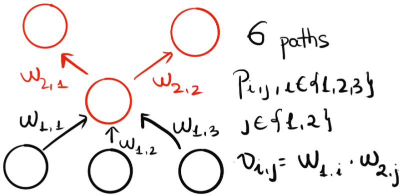
- Isomorphism classes $\left[W_x^f\right] = \{\tau \cdot W_x^f : \tau \in \bar{G}\}$ induce **bad-behaved** Lie algebras.
- Under some technical transformations, we get what is called the **moduli space** of data of a neural network $\mathcal{N}$: **a smooth algebraic variety** coming from the previous isomorphism classes.

Two possible outcomes:

1. Theoretical and practical study of the adjacency matrices W_x^f for a dataset D and different neural networks. Do these **linear transformations** give us information about the general **robustness** of a neural network? Should we study the **topology of the set of linear transformations** induced by the neural network?
2. **Manifold-hypothesis**: It can be proved that the data manifold $\mathcal{M}$ in the input space of a neural network (W, f) forms a **sub-manifold** of the moduli space that parameterise the image of the neural network.

Generalisation using moduli spaces

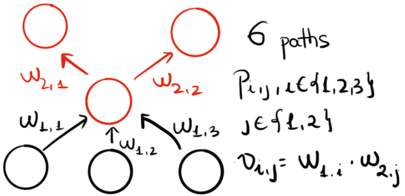
- The dimension of the moduli space is equal to the number of **basis paths** in ReLU networks, that yields, **provably**, an **upper bound** of the generalisation error.



Generalisation using moduli spaces

UNIVERSITAT DE BARCELONA

- The dimension of the moduli space is equal to the number of **basis paths** in ReLU networks, that yields, **provably**, an **upper bound** of the generalisation error.



- Can we use these moduli spaces to further amplify our knowledge on generalisation bounds?

Conclusions

UNIVERSITAT DE BARCELONA

- Quiver theory can be used effectively to model most neural networks.

Conclusions

Universitat de Barcelona

- Quiver theory can be used effectively to model most neural networks.
- Quiver theory allows us to use new persistent theory models to study neural networks (persistent path homology).

Conclusions

UNIVERSITAT DE BARCELONA

- Quiver theory can be used effectively to model most neural networks.
- Quiver theory allows us to use new persistent theory models to study neural networks (persistent path homology).
- Quiver theory allows us to give data representations depending on the structure of the neural network (moduli spaces).