On the Limitations of Representing Functions on Sets

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

*Equal contribution
Examples for Permutation Invariant Problems: Detecting Common Attributes

- Smiling
- Blond Hair

CelebA Dataset, Liu et al.
The deep sets architecture
The deep sets architecture
The deep sets architecture
The deep sets architecture

Input

Latent A

ϕ
The deep sets architecture

Input

Latent A

Latent B

\(\phi\)
The deep sets architecture

Input

Latent A

Latent B

$\phi$

$\rho$
The deep sets architecture

Input

Latent A

Latent B

Output
\[
\begin{align*}
X \subset \mathbb{R}^M &\xrightarrow{\phi} \mathbb{R}^{N \times M} &\xrightarrow{+} \mathbb{R}^N &\xrightarrow{\rho} f(x_1, \ldots, x_M)
\end{align*}
\]
Theorem 1 (Zaheer et al.): This architecture can successfully model any permutation invariant function, even for latent dimension $N=1$.  

\[ X \subset \mathbb{R}^M \rightarrow \mathbb{R}^{N \times M} \rightarrow \mathbb{R}^N \rightarrow \mathbb{R} \]
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 $\Phi$ and $\rho$ are universal function approximators.
Input $x_1, \ldots, x_M$ $\xrightarrow{\phi} \phi(x_1), \ldots, \phi(x_M)$ $\xrightarrow{+} Y$ $\xrightarrow{\rho} f(x_1, \ldots, x_M)$

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 $\Phi$ and $\rho$ are universal function approximators.

Find a $\Phi$ 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 $\Phi$ and $\rho$ are universal function approximators

Find a $\Phi$ such that mapping from input set $X$ to latent representation $Y$ is injective

Everything can be modelled
\[ X \subset \mathbb{R}^M \rightarrow \mathbb{R}^{N \times M} \rightarrow \mathbb{R}^N \rightarrow \mathbb{R} \]

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

\textbf{Proof:}

Assume that neural networks \( \Phi \) and \( \rho \) are universal function approximators.

Find a \( \Phi \) such that mapping from input set \( X \) to latent representation \( Y \) is injective.

Everything can be modelled.

Define \( \phi(x) : \mathbb{Q} \rightarrow \mathbb{N} \)
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 $\Phi$ and $\rho$ are universal function approximators

Find a $\Phi$ 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

A Continuous Function on $\mathbb{Q}$

We need to take real numbers into account!
Input

\[ X \subset \mathbb{R}^M \]

\[ x_1, x_M \]

\[ \phi \]

\[ \phi(x_1), \phi(x_M) \]

\[ Y \]

\[ \rho \]

\[ f(x_1, \ldots, x_M) \]

Output

\[ \mathbb{R}^M \rightarrow \mathbb{R}^{N \times M} \rightarrow \mathbb{R}^N \rightarrow \mathbb{R} \]
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 $\text{max}(X)$. 
**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 $\text{max}(X)$. We show that, in order to represent $\text{max}(X)$, $\Phi(X) = \sum_{x} \phi(x)$ 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 $\text{max}(X)$. We show that, in order to represent $\text{max}(X)$, $\Phi(X) = \sum_x \phi(x)$ 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