ST456 Deep Learning

Assessment 1 background material

Set functions

Milan Vojnovic

https://github.com/lse-st456/lectures2022

Permutation invariant functions

- A function $f: \mathbf{R}^{m\times d} \to \mathbf{R}$ is said to be permutation invariant if for every $\boldsymbol{X} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_m)^\top \in \mathbf{R}^{m\times d}$ and $\boldsymbol{X}' = \left(\boldsymbol{x}_{\pi_1}, \dots, \boldsymbol{x}_{\pi_m}\right)^\top$ where π is an arbitrary permutation of $1, \dots, m$, it holds

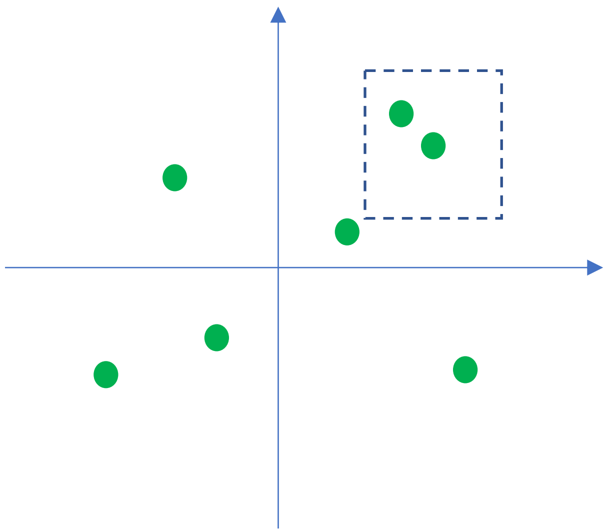
$$f(\boldsymbol{X}') = f(\boldsymbol{X})$$

- In other words, the output of the function does not change (is invariant) by changing the order of values (permuting) of different coordinates of the input

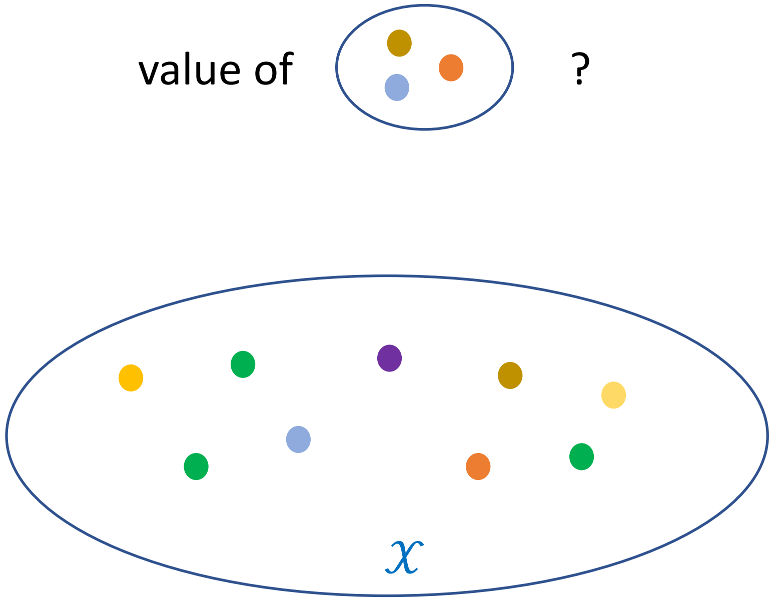
- Examples of permutation invariant functions for $d = 1$:
  - Statistics queries, e.g. count, max, min, mean, median, any quantile value
  - p-norm: $f(\boldsymbol{x}) = \|\boldsymbol{x}\|_p = (|x_1|^p + \cdots + |x_m|^p)^{1/p}$
  - A nonlinear transformation of sum: $f(\boldsymbol{x}) = g(x_1 + \cdots + x_m)$

Set functions

- A function f is said to be a set function if it maps every set $X \subseteq \mathcal{X}$, for some ground set of values $\mathcal{X}$, to a real number $f(X)$
- Set functions are permutation invariant
- Examples of set functions



Range queries



Valuation functions

- Note that each element of a set may be a vector

Sum decomposable functions

- Set functions can be represented by the class of sum-decomposable functions

- A set function f is said to be sum-decomposable via $\mathcal{Z}$ if

$$f(\boldsymbol{X}) = \rho\left(\sum_{\boldsymbol{x} \in \boldsymbol{X}} \phi(\boldsymbol{x})\right) \text{ for all } \boldsymbol{X} \subseteq \mathcal{X}$$

  where $\phi: \mathcal{X} \to \mathcal{Z}$ and $\rho: \mathcal{Z} \to \mathbf{R}$ are some functions

- Function f is said to be continuously sum-decomposable via $\mathcal{Z}$ if it is sum-decomposable with ϕ and ρ being some continuous functions

- We will refer to $\mathcal{Z}$ as a latent space and dimension of $\mathcal{Z}$ as latent dimension

Set and sum-decomposable functions

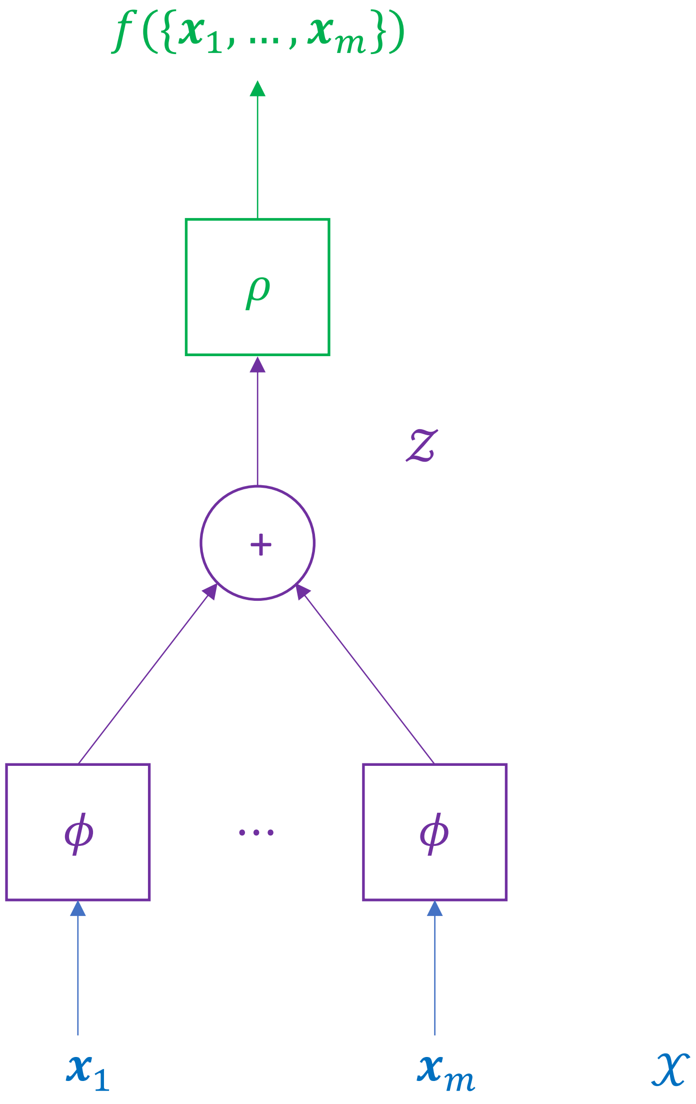
- Thm 1: Any set function f defined on subsets of a countable set $\mathcal{X}$ is permutation invariant if, and only if, it is sum-decomposable via $\mathbf{R}$

- Thm 2: Any continuous function $f: \mathbf{R}^m \to \mathbf{R}$ is permutation invariant if, and only if, it is continuously sum-decomposable via $\mathbf{R}^m$

- Thm 3: Any continuous function $f: \mathbf{R}^{\leq m} \to \mathbf{R}$ is permutation invariant if, and only if, it is continuously decomposable via $\mathbf{R}^m$

Note: $\mathbf{R}^{\leq m}$ denotes the set of real vectors of dimension $\leq m$

Learning set functions



- Functions ϕ and ρ are neural networks
- For example, we may take
  - ϕ to be a feedforward neural network
  - ρ to be a feedforward neural network
- We refer to the entire network as a (ϕ, ρ)-sum-decomposition network

Permutation equivariant functions

- Let $f: \mathbf{R}^{m\times d} \to \mathbf{R}^{m\times d'}$, and for every $\boldsymbol{X} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_m)^\top \in \mathbf{R}^{m\times d}$, we write

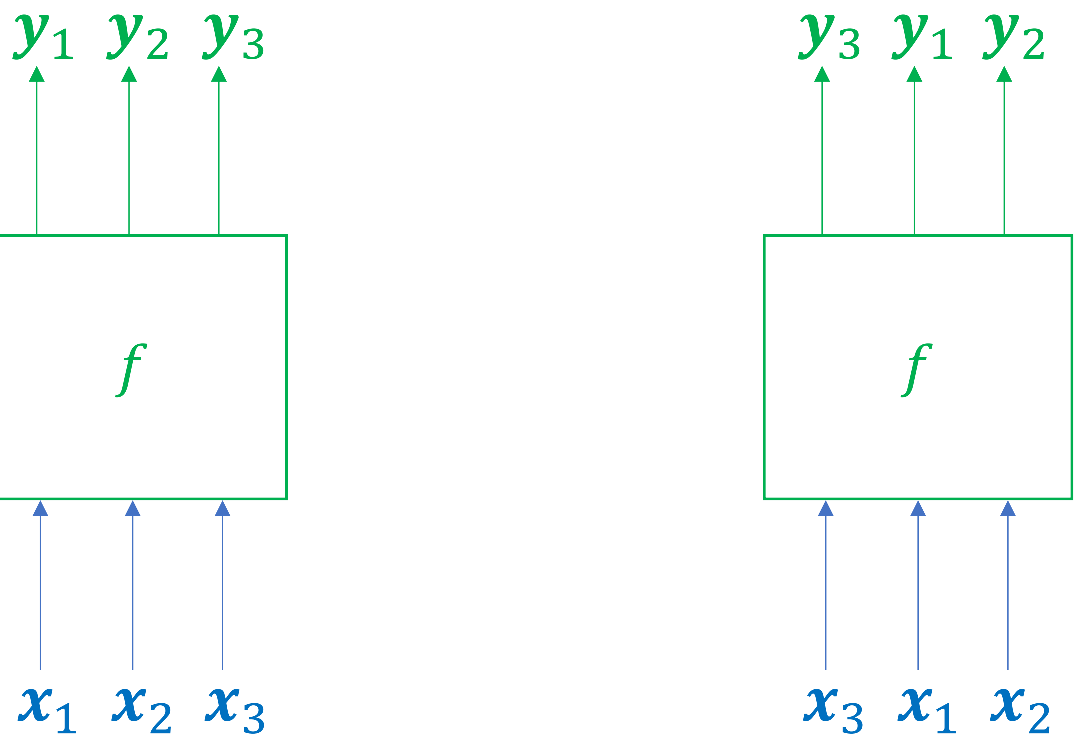
$$f(\boldsymbol{X}) = \left(f_1(\boldsymbol{X}), \dots, f_m(\boldsymbol{X})\right)^\top \in \mathbf{R}^{m\times d'}$$

- Function f is said to be permutation-equivariant if for every $\boldsymbol{X} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_m)^\top$ and $\boldsymbol{X}' = \left(\boldsymbol{x}_{\pi_1}, \dots, \boldsymbol{x}_{\pi_m}\right)^\top$ where π is an arbitrary permutation of $1, \dots, m$, it holds

$$f(\boldsymbol{X}') = \left(f_{\pi_1}(\boldsymbol{X}), \dots, f_{\pi_m}(\boldsymbol{X})\right)^\top$$

- In other words, changing the order of inputs $\boldsymbol{x}_1, \dots, \boldsymbol{x}_m$ to function f according to an arbitrary permutation, changes the output of f only in changing the order of the outputs according to the same permutation

Illustration



Examples

- Feature selection: input is a feature vector, the output is a vector with 0 or 1 valued coordinates, with $y_i = 1$ if feature i is selected, and $y_i = 0$ otherwise

- Choice functions: input is a set of items, the output is a vector with 0 or 1 valued coordinates, with $y_i = 1$ if item i is chosen, and $y_i = 0$ otherwise
    - For example, exactly one item is chosen

- Ranking functions: input is set of feature vectors $\boldsymbol{x}_1, \dots, \boldsymbol{x}_m$ representing some items, and the output is a ranking of items with y_i denoting the rank of item i

Affine equivariant transformations

- Let $f: \mathbf{R}^m \to \mathbf{R}^m$ be an affine transformation, i.e.

$$f(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{c} \text{ for some } \boldsymbol{W} \in \mathbf{R}^{m\times m} \text{ and } \boldsymbol{c} \in \mathbf{R}^m$$

- Then, f is a permutation equivariant if, and only if,

$$f(\boldsymbol{x}) = a\,\boldsymbol{x} + b\left(\frac{1}{m}\sum_{i=1}^{m} x_i\right)\mathbf{1} + c\mathbf{1}$$

for some real scalar parameters a, b, and c (Exercise: show this)

- In general, an affine transformation $f: \mathbf{R}^{m\times d} \to \mathbf{R}^{m\times d'}$ is permutation equivariant if, and only if,

$$f(\boldsymbol{X}) = \boldsymbol{X}\boldsymbol{A} + \frac{1}{m}\mathbf{1}\mathbf{1}^\top\boldsymbol{X}\boldsymbol{B} + \mathbf{1}\boldsymbol{c}^\top$$

where $\boldsymbol{A} \in \mathbf{R}^{d\times d'}$, $\boldsymbol{B} \in \mathbf{R}^{d\times d'}$, and $\boldsymbol{c} \in \mathbf{R}^{d'}$ are parameters

Permutation equivariant neural networks

- An equivariant neural network f is a network with equivariant layers, i.e.

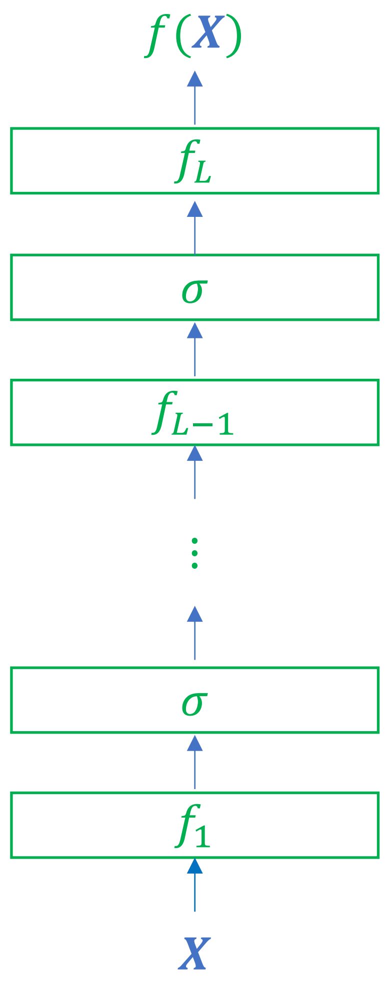
$$f(\boldsymbol{X}) = f_L \circ \sigma \circ \cdots \circ \sigma \circ f_1(\boldsymbol{X})$$

where σ is some activation function, and $f_1, \dots, f_L$ are equivariant affine transformations

$$f_l(\boldsymbol{X}) = \boldsymbol{X}\boldsymbol{A}_l + \frac{1}{m}\mathbf{1}\mathbf{1}^\top \boldsymbol{X}\boldsymbol{B}_l + \mathbf{1}\boldsymbol{c}_l^\top$$

where $\boldsymbol{A}_l \in \mathbf{R}^{d_{l-1} \times d_l}$, $\boldsymbol{B}_l \in \mathbf{R}^{d_{l-1} \times d_l}$, and $\boldsymbol{c}_l \in \mathbf{R}^{d_l}$ are parameters

- Note: $\sigma \circ f_l(\boldsymbol{X})$ means applying σ to each element of the matrix $f_l(\boldsymbol{X})$



References

- N. Sego and Y. Lipman. On universal equivariant set networks. In ICLR 2020
- E. Wagstaff, F. Fuchs, M. Engelcke, I. Posner, and M. A. Osborne, On the limitations of representing functions on sets, ICML 2019
- M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. Salakhutdinov, and A. J. Smola, Deep sets, NeurIPS 2017

Solution for the exercise

- We have $f(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{c}$, where $\boldsymbol{W} \in \mathbf{R}^{m \times m}$ and $\boldsymbol{c} \in \mathbf{R}^m$ are parameters
- Let $\boldsymbol{x}$ and $\boldsymbol{x}'$ be equal except for swapping the values of elements i and j, for some arbitrarily fixed $i, j \in \{1, \dots, m\}$ such that $i \neq j$
- Let $\boldsymbol{y} = f(\boldsymbol{x})$ and $\boldsymbol{y}' = f(\boldsymbol{x}')$
- The following three facts follow from permutation equivariance:
  - For every $l \in \{1, \dots, m\} \setminus \{i, j\}$, it holds $y_l = y'_l$ which is equivalent to $w_{l,i}x_i + w_{l,j}x_j = w_{l,i}x_j + w_{l,j}x_i$, from which it follows that all non-diagonal elements of $\boldsymbol{W}$ must be identical, say of value β
  - It holds $y_i = y'_j$, i.e. $w_{i,i}x_i + w_{i,j}x_j = w_{j,j}x_i + w_{j,i}x_j$, from which it follows that all diagonal elements of $\boldsymbol{W}$ must be identical, say of value α
  - All the elements of $\boldsymbol{c}$ are identical, say of value c
- It follows that

$$f(\boldsymbol{x}) = (\alpha - \beta)\boldsymbol{x} + \beta\left(\sum_{i=1}^{m} x_i\right)\mathbf{1} + c\mathbf{1}$$

which corresponds to the claim by using the reparametrization $a = \alpha - \beta$ and $b = \beta m$