

# Advanced Deep Learning

## Lab 1: Symmetry in Deep Learning Representations

Cong-Vinh Dang

January 2026

### Question 1

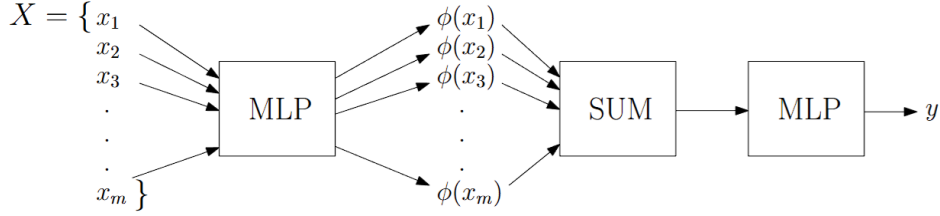


Figure 1: The DeepSets model.

The target function in this task is to compute the sum of the elements of a set,

$$f(X) = \sum_{x \in X} x. \quad (1)$$

This function is invariant to permutations of the input elements, meaning that the output depends only on the multiset of values and not on their ordering.

As illustrated in Figure 1, the DeepSets architecture follows this structure by applying the same transformation to each element of the set, aggregating the results through a summation, and then applying a final linear mapping. This leads to the general form

$$f(X) = \rho \left( \sum_{x \in X} \phi(x) \right). \quad (2)$$

For the task of summing integers, an optimal solution is obtained when each element contributes its original value before aggregation, that is  $\phi(x) = x$ , and when the final mapping simply returns the aggregated result, that is  $\rho(s) = s$ . In the implemented model, the element-wise mapping includes a ReLU activation in the first fully-connected layer, which is defined as

$$\text{ReLU}(z) = \max(0, z).$$

Since the input digits are non-negative, this activation does not modify the signal in this setting.

This can be made explicit by considering a simple choice of parameters. If the linear transformation in the element-wise MLP uses a unit weight and zero bias, then

$$\phi(x) = \text{ReLU}(1 \cdot x + 0) = \text{ReLU}(x) = x,$$

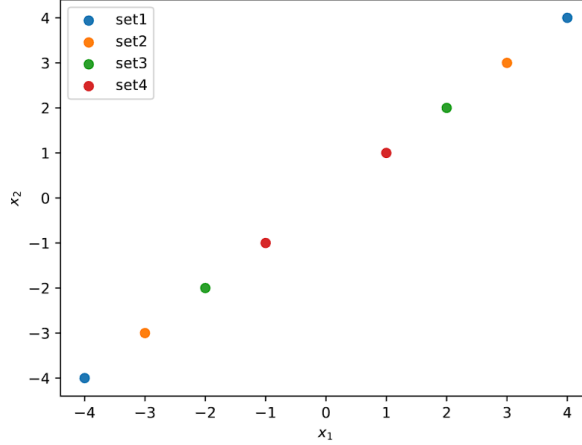


Figure 2: A synthetic example consisting of four sets.

so each input value is passed through unchanged. After aggregation, we obtain

$$\sum_{x \in X} \phi(x) = \sum_{x \in X} x, \quad (3)$$

which already corresponds to the desired output. The final MLP  $\rho$  can therefore be chosen as a linear identity mapping, with weight 1 and bias 0, so that

$$\rho(s) = s.$$

In conclusion, the optimal parameters correspond to identity mappings at both stages: the element-wise MLP preserves each input value under ReLU for non-negative inputs, and the final MLP leaves the aggregated sum unchanged. With these parameters, the DeepSets model exactly implements the target sum function.

## Question 2

In the example shown in Figure 2, each set contains two two-dimensional vectors that are symmetric with respect to the origin. As a consequence, the sum of the two vectors in every set is equal to  $[0, 0]$ . After applying the element-wise mapping  $\phi$  and aggregating the results, the representation of a set can be written as

$$\sum_{x \in X} \phi(x) = \phi(x_1) + \phi(x_2). \quad (4)$$

Because the two elements in each set are symmetric, the aggregated value produced by the summation operator is identical for all four sets, even though their internal structures are different. Since the final mapping  $\rho$  only receives this aggregated result as input, it does not have enough information to distinguish between the sets.

Therefore, the DeepSets model produces the same representation for all four sets in this example and cannot learn different representations despite the differences in their internal structure.