nn.softmax

```

Copy fnsoftmax(tensor:@Tensor, axis:usize)->Tensor;

```

Applies the Softmax function to an n-dimensional input Tensor rescaling them so that the elements of the n-dimensional output Tensor lie in the range [0,1] and sum to 1.

! softmax (x i) = e x i ∑ j = 1 n e x j \text{softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}

Args

- tensor
- (@Tensor
-) - The input tensor.
- axis
- (usize
-) - The axis along which to compute the softmax.
-

Returns

A Tensor of fixed point numbers with the same shape than the input Tensor.

Type Constraints

Constrain input and output types to fixed point tensors.

Examples

```

Copy usecore::array::{ArrayTrait,SpanTrait};

useorion::operators::tensor::{TensorTrait,Tensor,FP8x23Tensor}; useorion::operators::nn::{NNTrait,FP8x23NN};
useorion::numbers::{FP8x23,FixedTrait};

fnsoftmax_example()->Tensor { lettensor=TensorTrait::::new( shape:array![2,2].span(), data:array![ FixedTrait::new(0,false),
FixedTrait::new(1,false), FixedTrait::new(2,false), FixedTrait::new(3,false), ] .span(), );

returnNNTrait::softmax(@tensor,1); }

                [[2255697,6132911],[2255697,6132911]] // The fixed point representation of // [[0.2689,
                0.7311],[0.2689, 0.7311]]

```

Previous nn.sigmoid Next nn.softmax_zero

Last updated3 months ago