

tensor.shrink

...

Copy fnshrink(self:@Tensor, bias:Option, lambd:Option)->Tensor

...

Shrinks the input tensor element-wise to the output tensor with the same datatype and shape based on the following formula: If $x < -\text{lambd}$: $y = x + \text{bias}$; If $x > \text{lambd}$: $y = x - \text{bias}$; Otherwise: $y = 0$.

Args

- self
- (@Tensor
-) - The input tensor to be shrunk.
- bias
- (Option
-) - The bias value added to or subtracted from input tensor values.
- lambd
- (Option
-) - The lambd value defining the shrink condition.
-

Returns

A newTensor of the same datatype and shape as the input tensor with shrunk values.

Type Constraints

Constrain input and output types to fixed point numbers.

Examples

...

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

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

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

letbias=Option::Some(FixedTrait::new(1,false)) letlambd=Option::Some(FixedTrait::new(1,false))

returntensor.shrink(tensor, bias, lambd); }

[-8388608,0,0,8388608] // The fixed point representation of [-1,0,0,1]

...

[Previous tensor.reduce_min](#) [Next tensor.reduce_mean](#)

Last updated3 months ago