tensor.compress

```
Copy fncompress(self:@Tensor, condition:Tensor, axis:Option)->Tensor;
```

Selects slices from an input tensor along a given axis where condition evaluates to True for each axis index. In case axis is not provided, input is flattened before elements are selected.

Args

- self
- (@Tensor
-) - The input tensor.
- condition
- (Tensor
-) - Rank 1 tensor of booleans to indicate which slices or data elements to be selected. Its length can be less than the input length along the axis or the flattened input size if axis is not specified. In such cases data slices or elements exceeding the condition length are discarded.
- axis
- (Option
-) - (Optional) Axis along which to take slices. If not specified, input is flattened before elements being selected. Negative value means counting dimensions from the back. Accepted range is [-r, r-1] where r = rank(input).
-

Panics

- Panics if condition rank is not equal to 1.
-

Returns

A newTensor .

Example

```
Copy usearray::{ArrayTrait,SpanTrait}; useorion::operators::tensor::{TensorTrait,Tensor,U32Tensor};

fncompress_example()->Tensor { lettensor=TensorTrait::::new( shape:array![3,2].span(), data:array![[1,2], [3,4], [5,6]].span(), ); letcondition=TensorTrait::::new( shape:array![3].span(), data:array![0,1,1].span(), );

returntensor.compress( condition:condition, axis:Option::Some((0)), ); }

                [[3,4], [5,6]]
```

Last updated1 month ago