

---

**panndas**

**Charles Frye**

**Mar 19, 2022**



# CONTENTS

<b>1</b>	<b>Reference</b>	<b>1</b>
<b>2</b>	<b>Installation</b>	<b>5</b>
<b>3</b>	<b>Usage</b>	<b>7</b>
	<b>Python Module Index</b>	<b>9</b>
	<b>Index</b>	<b>11</b>



## REFERENCE

- [\*panndas.nn\*](#)

## 1.1 panndas.nn

**class** panndas.nn.**AdditiveSkip**(*block*)

A Module that applies an additive “skip” connection around the provided Module.

**forward**(*xs*)

Minimally, define this method to define a Module.

**show**()

Displays the Module.

**class** panndas.nn.**AlphaDropout**(*p*, *alpha=0.0*)

A Module that multiplies its inputs by the weights\_df and adds the bias\_series.

Input ‘tensors’ can be at most 2-D here: feature (rows) and batch/sequence (columns).

The weights dataframe should have the input feature space as its column index and the output feature space as its row index.

**forward**(*xs*)

Minimally, define this method to define a Module.

**show**()

Displays the Module.

**class** panndas.nn.**Dropout**(*p*)

**class** panndas.nn.**Identity**

A Module that returns its inputs unaltered.

**forward**(*xs*)

Minimally, define this method to define a Module.

**class** panndas.nn.**LayerMaxNorm**

Normalize across the feature dimension with respect to the infinity norm.

**forward**(*xs*)

Minimally, define this method to define a Module.

**class** panndas.nn.**Linear**(*weights\_df*, *bias\_series=-1*)

A Module that multiplies its inputs by the weights\_df and adds the bias\_series.

Input ‘tensors’ can be at most 2-D here: feature (rows) and batch/sequence (columns).

The weights dataframe should have the input feature space as its column index and the output feature space as its row index.

**forward**(*xs*)

Minimally, define this method to define a Module.

**show**()

Displays the Module.

**class** panndas.nn.**LinearAttention**(*queries\_df, keys\_df, values\_df*)

The most basic version of an attention layer.

**forward**(*xs*)

Combines queries, keys, and values linearly.

**class** panndas.nn.**Mish**

Applies the Mish function, element-wise.

For details, see [Mish: A Self-Regularized Non-Monotonic Neural Activation Function](#).

**forward**(*xs*)

Applies the Mish function, element-wise.

**class** panndas.nn.**Module**

An object that is callable via its .forward method.

**abstract forward**(*xs*)

Minimally, define this method to define a Module.

**show**()

Displays the Module.

**class** panndas.nn.**ReLU**

0/1 ReLU-iable.

**forward**(*xs*)

Minimally, define this method to define a Module.

**class** panndas.nn.**Sequential**(*modules*)

A Module that applies an iterable of Modules sequentially.

**forward**(*xs*)

Minimally, define this method to define a Module.

**show**()

Displays the Module.

**class** panndas.nn.**Sigmoid**

Applies the sigmoid function, element-wise.

**forward**(*xs*)

Applies the sigmoid function, element-wise.

**class** panndas.nn.**Softmax**

Applies softmax function, column-wise.

**forward**(*xs*)

Applies softmax function, column-wise.

**class** panndas.nn.**SoftmaxAttention**(*queries\_df, keys\_df, values\_df*)

The best-known version of an attention layer.

**forward**(*xs*)

Uses a softmax over the sequence dim to select which values to attend to.

**class** panndas.nn.**Softplus**

Applies the softplus function, element-wise.

**forward**(*xs*)

Applies the softplus function, element-wise.





## INSTALLATION

```
$ pip install pandas
```



---

CHAPTER  
**THREE**

---

**USAGE**

Simply don't.



## PYTHON MODULE INDEX

### p

`pandas.nn`, [1](#)



## A

AdditiveSkip (*class in panndas.nn*), 1  
AlphaDropout (*class in panndas.nn*), 1

## D

Dropout (*class in panndas.nn*), 1

## F

forward() (*panndas.nn.AdditiveSkip method*), 1  
forward() (*panndas.nn.AlphaDropout method*), 1  
forward() (*panndas.nn.Identity method*), 1  
forward() (*panndas.nn.LayerMaxNorm method*), 1  
forward() (*panndas.nn.Linear method*), 2  
forward() (*panndas.nn.LinearAttention method*), 2  
forward() (*panndas.nn.Mish method*), 2  
forward() (*panndas.nn.Module method*), 2  
forward() (*panndas.nn.ReLU method*), 2  
forward() (*panndas.nn.Sequential method*), 2  
forward() (*panndas.nn.Sigmoid method*), 2  
forward() (*panndas.nn.Softmax method*), 2  
forward() (*panndas.nn.SoftmaxAttention method*), 2  
forward() (*panndas.nn.Softplus method*), 3

## I

Identity (*class in panndas.nn*), 1

## L

LayerMaxNorm (*class in panndas.nn*), 1  
Linear (*class in panndas.nn*), 1  
LinearAttention (*class in panndas.nn*), 2

## M

Mish (*class in panndas.nn*), 2  
module  
    panndas.nn, 1  
Module (*class in panndas.nn*), 2

## P

panndas.nn  
    module, 1

## R

ReLU (*class in panndas.nn*), 2

## S

Sequential (*class in panndas.nn*), 2  
show() (*panndas.nn.AdditiveSkip method*), 1  
show() (*panndas.nn.AlphaDropout method*), 1  
show() (*panndas.nn.Linear method*), 2  
show() (*panndas.nn.Module method*), 2  
show() (*panndas.nn.Sequential method*), 2  
Sigmoid (*class in panndas.nn*), 2  
Softmax (*class in panndas.nn*), 2  
SoftmaxAttention (*class in panndas.nn*), 2  
Softplus (*class in panndas.nn*), 3