

Configuring Loss/Optimizers

Function Signature

```
def config(  
    optimizer: str = 'gradient descent',  
    loss: str = 'mean square error'  
) -> None:
```

Parameters

- `optimizer` (str, default='gradient descent'): The optimizer to use during training. Currently, `deeprai` is in beta, so the only valid option for optimizer is 'gradient descent'.
- `loss` (str, default='mean square error'): The loss function to use during training. Valid options are 'mean square error', 'categorical cross entropy', and 'mean absolute error'.

Return Value

This function does not return anything. It modifies the `deeprai.models.FeedForward` instance by setting the optimizer and loss function.

Description

The `config` function sets the optimizer and loss function for the `deeprai.models.FeedForward` instance. While it is not necessary to call this function, if called, it will use the default values of 'gradient descent' for optimizer and 'mean square error' for loss function.

Currently, `deeprai` is in beta, so the only valid option for optimizer is 'gradient descent'. The loss parameter sets the loss function to use during training. Valid options are 'mean square error', 'categorical cross entropy', and 'mean absolute error'.

Examples

Here's an example of how to use the `config` function:

```
from deeprai.models import FeedForward

model = FeedForward()
model.add_dense(784)
model.add_dense(128, activation='relu')
model.add_dense(64, activation='relu')
model.add_dense(10, activation='softmax')
model.config(optimizer='gradient descent', loss='categorical cross entropy')
```

This code creates a `FeedForward` model with an input shape of `(784,)`, adds three dense layers with ReLU and softmax activation functions, and sets the optimizer to 'gradient descent' and the loss function to 'categorical cross entropy'.

Revision #2

Created 7 April 2023 07:23:11 by Kieran Carter

Updated 7 April 2023 07:40:56 by Kieran Carter