tf.contrib.kfac.fisher_blocks.ConvKFCBasicFB

Contents

Class `ConvKFCBasicFB`

Inherits From: `KroneckerProductFB`

Defined in `tensorflow/contrib/kfac/python/ops/fisher_blocks.py`.

FisherBlock for 2D convolutional layers using the basic KFC approx.

See https://arxiv.org/abs/1602.01407 for details.

Methods

`__init__`

```
__init__(
    layer_collection,
    params,
    inputs,
    outputs,
    strides,
    padding
)
```

Creates a ConvKFCBasicFB block.

Args:

- `layer_collection`: The collection of all layers in the K-FAC approximate Fisher information matrix to which this FisherBlock belongs.
- `params`: The parameters (Tensor or tuple of Tensors) of this layer. If kernel alone, a Tensor of shape [kernel_height, kernel_width, in_channels, out_channels]. If kernel and bias, a tuple of 2 elements containing the previous and a Tensor of shape [out_channels].
- `inputs`: A Tensor of shape [batch_size, height, width, in_channels]. Input activations to this layer.
- `outputs`: A Tensor of shape [batch_size, height, width, out_channels]. Output pre-activations from this layer.
- `strides`: The stride size in this layer (1-D Tensor of length 4).
- `padding`: The padding in this layer (1-D of Tensor length 4).

`full_fisher_block`

```
full_fisher_block()
```

Explicitly constructs the full Fisher block.

Used for testing purposes. (In general, the result may be very large.)

Returns:

The full Fisher block.

`instantiate_factors`

```
instantiate_factors(
    grads_list,
    damping
)
```

`multiply`

```
multiply(vector)
```

`multiply_inverse`

```
multiply_inverse(vector)
```

`tensors_to_compute_grads`

```
tensors_to_compute_grads()
```

Last updated November 2, 2017.