tf.zeros

zeros(  
    shape,  
    dtype=tf.float32,  
    name=None  
)

Defined in [tensorflow/python/ops/array\_ops.py](https://www.github.com/tensorflow/tensorflow/blob/r1.4/tensorflow/python/ops/array_ops.py).

See the guide: [Constants, Sequences, and Random Values > Constant Value Tensors](https://www.tensorflow.org/api_guides/python/constant_op#Constant_Value_Tensors)

Creates a tensor with all elements set to zero.

This operation returns a tensor of type dtype with shape shape and all elements set to zero.

For example:

tf.zeros([3, 4], tf.int32)  # [[0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0]]

Args:

* **shape**: A list of integers, a tuple of integers, or a 1-D Tensor of type int32.
* **dtype**: The type of an element in the resulting Tensor.
* **name**: A name for the operation (optional).

Returns:

A Tensor with all elements set to zero.

tf.zeros\_like

zeros\_like(  
    tensor,  
    dtype=None,  
    name=None,  
    optimize=True  
)

Defined in [tensorflow/python/ops/array\_ops.py](https://www.github.com/tensorflow/tensorflow/blob/r1.4/tensorflow/python/ops/array_ops.py).

See the guide: [Constants, Sequences, and Random Values > Constant Value Tensors](https://www.tensorflow.org/api_guides/python/constant_op#Constant_Value_Tensors)

Creates a tensor with all elements set to zero.

Given a single tensor (tensor), this operation returns a tensor of the same type and shape as tensor with all elements set to zero. Optionally, you can use dtype to specify a new type for the returned tensor.

For example:

tensor = tf.constant([[1, 2, 3], [4, 5, 6]])  
tf.zeros\_like(tensor)  # [[0, 0, 0], [0, 0, 0]]

Args:

* **tensor**: A Tensor.
* **dtype**: A type for the returned Tensor. Must be float32, float64, int8, int16, int32, int64, uint8, complex64, or complex128.
* **name**: A name for the operation (optional).
* **optimize**: if true, attempt to statically determine the shape of 'tensor' and encode it as a constant.

Returns:

A Tensor with all elements set to zero.

tf.ones

ones(  
    shape,  
    dtype=tf.float32,  
    name=None  
)

Defined in [tensorflow/python/ops/array\_ops.py](https://www.github.com/tensorflow/tensorflow/blob/r1.4/tensorflow/python/ops/array_ops.py).

See the guide: [Constants, Sequences, and Random Values > Constant Value Tensors](https://www.tensorflow.org/api_guides/python/constant_op#Constant_Value_Tensors)

Creates a tensor with all elements set to 1.

This operation returns a tensor of type dtype with shape shape and all elements set to 1.

For example:

tf.ones([2, 3], tf.int32)  # [[1, 1, 1], [1, 1, 1]]

Args:

* **shape**: A list of integers, a tuple of integers, or a 1-D Tensor of type int32.
* **dtype**: The type of an element in the resulting Tensor.
* **name**: A name for the operation (optional).

Returns:

A Tensor with all elements set to 1.

tf.ones\_like

ones\_like(  
    tensor,  
    dtype=None,  
    name=None,  
    optimize=True  
)

Defined in [tensorflow/python/ops/array\_ops.py](https://www.github.com/tensorflow/tensorflow/blob/r1.4/tensorflow/python/ops/array_ops.py).

See the guide: [Constants, Sequences, and Random Values > Constant Value Tensors](https://www.tensorflow.org/api_guides/python/constant_op#Constant_Value_Tensors)

Creates a tensor with all elements set to 1.

Given a single tensor (tensor), this operation returns a tensor of the same type and shape as tensor with all elements set to 1. Optionally, you can specify a new type (dtype) for the returned tensor.

For example:

tensor = tf.constant([[1, 2, 3], [4, 5, 6]])  
tf.ones\_like(tensor)  # [[1, 1, 1], [1, 1, 1]]

Args:

* **tensor**: A Tensor.
* **dtype**: A type for the returned Tensor. Must be float32, float64, int8, int16, int32, int64, uint8, complex64, complex128 or bool.
* **name**: A name for the operation (optional).
* **optimize**: if true, attempt to statically determine the shape of 'tensor' and encode it as a constant.

Returns:

A Tensor with all elements set to 1.

tf.fill

fill(  
    dims,  
    value,  
    name=None  
)

Defined in tensorflow/python/ops/gen\_array\_ops.py.

See the guide: [Constants, Sequences, and Random Values > Constant Value Tensors](https://www.tensorflow.org/api_guides/python/constant_op#Constant_Value_Tensors)

Creates a tensor filled with a scalar value.

This operation creates a tensor of shape dims and fills it with value.

For example:

# Output tensor has shape [2, 3].  
fill([2, 3], 9) ==> [[9, 9, 9]  
                     [9, 9, 9]]

Args:

* **dims**: A Tensor of type int32. 1-D. Represents the shape of the output tensor.
* **value**: A Tensor. 0-D (scalar). Value to fill the returned tensor.
* **name**: A name for the operation (optional).

Returns:

A Tensor. Has the same type as value.

numpy compatibility

Equivalent to np.full

tf.constant

constant(  
    value,  
    dtype=None,  
    shape=None,  
    name='Const',  
    verify\_shape=False  
)

Defined in [tensorflow/python/framework/constant\_op.py](https://www.github.com/tensorflow/tensorflow/blob/r1.4/tensorflow/python/framework/constant_op.py).

See the guide: [Constants, Sequences, and Random Values > Constant Value Tensors](https://www.tensorflow.org/api_guides/python/constant_op#Constant_Value_Tensors)

Creates a constant tensor.

The resulting tensor is populated with values of type dtype, as specified by arguments value and (optionally) shape (see examples below).

The argument value can be a constant value, or a list of values of type dtype. If value is a list, then the length of the list must be less than or equal to the number of elements implied by the shape argument (if specified). In the case where the list length is less than the number of elements specified by shape, the last element in the list will be used to fill the remaining entries.

The argument shape is optional. If present, it specifies the dimensions of the resulting tensor. If not present, the shape of value is used.

If the argument dtype is not specified, then the type is inferred from the type of value.

For example:

# Constant 1-D Tensor populated with value list.  
tensor = tf.constant([1, 2, 3, 4, 5, 6, 7]) => [1 2 3 4 5 6 7]  
  
# Constant 2-D tensor populated with scalar value -1.  
tensor = tf.constant(-1.0, shape=[2, 3]) => [[-1. -1. -1.]  
                                             [-1. -1. -1.]]

Args:

* **value**: A constant value (or list) of output type dtype.
* **dtype**: The type of the elements of the resulting tensor.
* **shape**: Optional dimensions of resulting tensor.
* **name**: Optional name for the tensor.
* **verify\_shape**: Boolean that enables verification of a shape of values.

Returns:

A Constant Tensor.

Raises:

* **TypeError**: if shape is incorrectly specified or unsupported.