

tf.contrib.data.batch_and_drop_remainder

```
batch_and_drop_remainder(batch_size)
```

Defined in [tensorflow/contrib/data/python/ops/batching.py](#).

A batching transformation that omits the final small batch (if present).

Like [tf.data.Dataset.batch](#), this transformation combines consecutive elements of this dataset into batches. However, if the batch size does not evenly divide the input dataset size, this transformation will drop the final smaller element.

The following example illustrates the difference between this transformation and [Dataset.batch\(\)](#):

```
dataset = tf.data.Dataset.range(200)
batched = dataset.apply(tf.contrib.data.batch_and_drop_remainder(128))
print(batched.output_shapes) # ==> "(128,)" (the batch dimension is known)
```

By contrast, [dataset.batch\(128\)](#) would yield a two-element dataset with shapes [\(128,\)](#) and [\(72,\)](#), so the batch dimension would not be statically known.

Args:

- batch_size**: A [tf.int64](#) scalar [tf.Tensor](#), representing the number of consecutive elements of this dataset to combine in a single batch.

Returns:

A [Dataset](#) transformation function, which can be passed to [tf.data.Dataset.apply](#)

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated November 2, 2017.

Stay Connected

[Blog](#)

[GitHub](#)

[Twitter](#)

Support

[Issue Tracker](#)

[Release Notes](#)

[Stack Overflow](#)

