

## tf.contrib.training.rejection\_sample

```
rejection_sample(  
    tensors,  
    accept_prob_fn,  
    batch_size,  
    queue_threads=1,  
    enqueue_many=False,  
    prebatch_capacity=16,  
    prebatch_threads=1,  
    runtime_checks=False,  
    name=None  
)
```

Defined in [tensorflow/contrib/training/python/training/sampling\\_ops.py](#).

See the guide: [Training \(contrib\) > Online data resampling](#)

Stochastically creates batches by rejection sampling.

Each list of non-batched tensors is evaluated by `accept_prob_fn`, to produce a scalar tensor between 0 and 1. This tensor corresponds to the probability of being accepted. When `batch_size` tensor groups have been accepted, the batch queue will return a mini-batch.

#### Args:

- `tensors`: List of tensors for data. All tensors are either one item or a batch, according to `enqueue_many`.
- `accept_prob_fn`: A python lambda that takes a non-batch tensor from each item in `tensors`, and produces a scalar tensor.
- `batch_size`: Size of batch to be returned.
- `queue_threads`: The number of threads for the queue that will hold the final batch.
- `enqueue_many`: Bool. If true, interpret input tensors as having a batch dimension.
- `prebatch_capacity`: Capacity for the large queue that is used to convert batched tensors to single examples.
- `prebatch_threads`: Number of threads for the large queue that is used to convert batched tensors to single examples.
- `runtime_checks`: Bool. If true, insert runtime checks on the output of `accept_prob_fn`. Using `True` might have a performance impact.
- `name`: Optional prefix for ops created by this function.

#### Raises:

- `ValueError`: `enqueue_many` is True and labels doesn't have a batch dimension, or if `enqueue_many` is False and labels isn't a scalar.
- `ValueError`: `enqueue_many` is True, and batch dimension on data and labels don't match.
- `ValueError`: if a zero initial probability class has a nonzero target probability.

## Returns:

A list of tensors of the same length as `tensors`, with batch dimension `batch_size`.

Example: # Get tensor for a single data and label example. data, label = data\_provider.Get(['data', 'label'])

# Get stratified batch according to data tensor. accept\_prob\_fn = lambda x: (tf.tanh(x[0]) + 1) / 2 data\_batch = tf.contrib.training.rejection\_sample([data, label], accept\_prob\_fn, 16)

# Run batch through network. ...

---

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](#)

English

[Terms](#) | [Privacy](#)