tf.contrib.distributions.percentile

```
percentile(
    x,
    q,
    axis=None,
    interpolation=None,
    keep_dims=False,
    validate_args=False,
    name=None
)
```

Defined in `tensorflow/contrib/distributions/python/ops/sample_stats.py`.

Compute the `q`-th percentile of `x`.

Given a vector `x`, the `q`-th percentile of `x` is the value `q / 100` of the way from the minimum to the maximum in a sorted copy of `x`.

The values and distances of the two nearest neighbors as well as the `interpolation` parameter will determine the percentile if the normalized ranking does not match the location of `q` exactly.

This function is the same as the median if `q = 50`, the same as the minimum if `q = 0` and the same as the maximum if `q = 100`.

```
# Get 30th percentile with default ('nearest') interpolation.
x = [1., 2., 3., 4.]
percentile(x, q=30.)
==> 2.0

# Get 30th percentile with 'lower' interpolation
x = [1., 2., 3., 4.]
percentile(x, q=30., interpolation='lower')
==> 1.0

# Get 100th percentile (maximum).  By default, this is computed over every dim
x = [[1., 2.]
     [3., 4.]]
percentile(x, q=100.)
==> 4.0

# Treat the leading dim as indexing samples, and find the 100th quantile (max)
# over all such samples.
x = [[1., 2.]
     [3., 4.]]
percentile(x, q=100., axis=[0])
==> [3., 4.]
```

Compare to `numpy.percentile`.

Args:

- `x`: Floating point `N-D` `Tensor` with `N > 0`. If `axis` is not `None`, `x` must have statically known number of dimensions.

- `q`: Scalar `Tensor` in `[0, 100]`. The percentile.
- `axis`: Optional `0-D` or `1-D` integer `Tensor` with constant values. The axis that hold independent samples over which to return the desired percentile. If `None` (the default), treat every dimension as a sample dimension, returning a scalar.
- `interpolation`: {"lower", "higher", "nearest"}. Default: "nearest" This optional parameter specifies the interpolation method to use when the desired quantile lies between two data points `i < j`:
  - lower: `i`.
  - higher: `j`.
  - nearest: `i` or `j`, whichever is nearest.
- `keep_dims`: Python `bool`. If `True`, the last dimension is kept with size 1 If `False`, the last dimension is removed from the output shape.
- `validate_args`: Whether to add runtime checks of argument validity. If False, and arguments are incorrect, correct behavior is not guaranteed.
- `name`: A Python string name to give this `Op`. Default is "percentile"

Returns:

A `(N - len(axis))` dimensional `Tensor` of same dtype as `x`, or, if `axis` is `None`, a scalar.

Raises:

- `ValueError`: If argument 'interpolation' is not an allowed type.

Last updated November 2, 2017.