tf.contrib.bayesflow.stochastic_tensor.StochasticTensor

Class `StochasticTensor`

Inherits From: `BaseStochasticTensor`

Defined in `tensorflow/contrib/bayesflow/python/ops/stochastic_tensor_impl.py`.

See the guide: BayesFlow Stochastic Tensors (contrib) > Stochastic Tensor Classes

StochasticTensor is a BaseStochasticTensor backed by a distribution.

Properties

`distribution`

`dtype`

`graph`

`name`

`value_type`

Methods

`__init__`

```
__init__(
    dist,
    name='StochasticTensor',
    dist_value_type=None,
    loss_fn=sge.score_function
)
```

Construct a `StochasticTensor`.

`StochasticTensor` is backed by the `dist` distribution and its `value` method will return the same value each time it is called. What `value` is returned is controlled by the `dist_value_type` (defaults to `SampleValue`).

Some distributions' sample functions are not differentiable (e.g. a sample from a discrete distribution like a Bernoulli) and

so to differentiate wrt parameters upstream of the sample requires a gradient estimator like the score function estimator. This is accomplished by passing a differentiable `loss_fn` to the `StochasticTensor`, which defaults to a function whose derivative is the score function estimator. Calling `stochastic_graph.surrogate_loss(final_losses)` will call `loss()` on every `StochasticTensor` upstream of final losses.

`loss()` will return None for `StochasticTensor`s backed by reparameterized distributions; it will also return None if the value type is `MeanValueType` or if `loss_fn=None`.

Args:

- `dist`: an instance of `Distribution`.
- `name`: a name for this `StochasticTensor` and its ops.
- `dist_value_type`: a `_StochasticValueType`, which will determine what the `value` of this `StochasticTensor` will be. If not provided, the value type set with the `value_type` context manager will be used.
- `loss_fn`: callable that takes `(st, st.value(), influenced_loss)`, where `st` is this `StochasticTensor`, and returns a `Tensor` loss. By default, `loss_fn` is the `score_function`, or more precisely, the integral of the score function, such that when the gradient is taken, the score function results. See the `stochastic_gradient_estimators` module for additional loss functions and baselines.

Raises:

- `TypeError`: if `dist` is not an instance of `Distribution`.
- `TypeError`: if `loss_fn` is not `callable`.

entropy

```
entropy(name='entropy')
```

loss

```
loss(
    final_loss,
    name='Loss'
)
```

mean

```
mean(name='mean')
```

value

```
value(name='value')
```

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | **Privacy**