

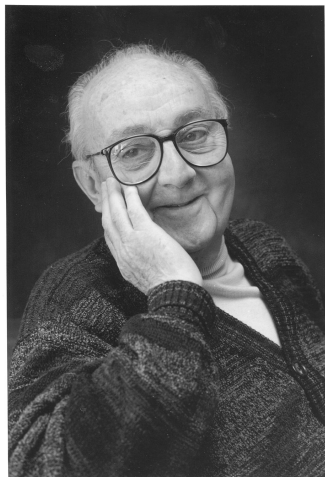
Probabilistic programming with Edward

John Reid

Biostatistics Unit,
School of Clinical Medicine,
Cambridge University

April 25, 2017

George E.P. Box (1919 - 2013)

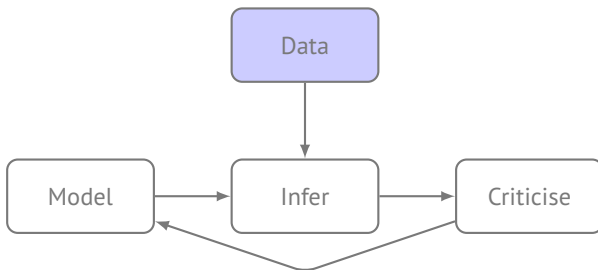


An iterative process for science:

1. Build a model of the science
2. Infer the model given data
3. Criticize the model given data

G. E. P. Box and Hunter, 1962; G. E. P. Box and Hunter, 1965; G. E. P. Box and Hill, 1967;
George E. P. Box, 1976; George E. P. Box, 1980

Box's Loop



Edward is a library designed around this loop.
(George E. P. Box, 1976; George E. P. Box, 1980)

Edward is a probabilistic programming language, designed for fast experimentation and research (Tran et al., 2017).

Modelling

- ▶ Composable Turing-complete language of random variables.
- ▶ Examples: Graphical models, neural networks, probabilistic programs.
- ▶ Many data types, tensor vectorization, broadcasting, 3rd party support.

Inference

- ▶ Composable language for hybrids, message passing, data subsampling.
- ▶ Examples: Black box VI, Hamiltonian MC, stochastic gradient MCMC, generative adversarial networks.
- ▶ Infrastructure to develop your own algorithms.

Criticism

- ▶ Examples: Scoring rules, hypothesis tests, predictive checks.

Built on TensorFlow (features distributed computing, GPUs, autodiff).

DATA

```
x_data = np.array([0, 1, 0, 0, 0, 0, 0, 0, 0, 1])
```

MODEL

```
p = Beta(a=1.0, b=1.0)
```

```
x = Bernoulli(p=tf.ones(10) * p)
```

VARIATIONAL DISTRIBUTION

```
qp_a = tf.nn.softplus(tf.Variable(tf.random_normal([])))
```

```
qp_b = tf.nn.softplus(tf.Variable(tf.random_normal([])))
```

```
qp = Beta(a=qp_a, b=qp_b)
```

INFERENCE

```
inference = ed.KLqp({p: qp}, data={x: x_data})
```

```
inference.run(n_iter=500)
```

CRITICISM

```
x_post = ed.copy(x, {p : qp})
```

```
T = lambda xs, zs: tf.reduce_mean(xs[x_post])
```

```
ed.ppc(T, data={x_post: x_data})
```