Introduction to probabilistic programming (with PyMC3)

A. Richards

02.08.2017

1. Introduction
2. PyMC3
3. Warm-up
4. PMF
5. Cool down

Probabilistic programming

A probabilistic programming language makes it easy to:

1. write out complex probability models
2. And subsequently solve these models automatically.

Generally this is accomplished by:

1. Random variables are handled as a primitive
2. Inference is handled behind the scenes
3. Memory and processor management is abstracted away

The pros and the cons

Why you might want to use probabilistic programming

1. Customization - We can create models that have built-in hypothesis tests
2. Propagation of uncertainty - There is a degree of belief associated prediction and estimation
3. Intuition - The models are essentially 'white-box' which provides insight into our data

Why you might NOT want use out probabilistic programming

1. Deep dive - Many of the online examples will assume a fairly deep understanding of statistics
2. Overhead - Computational overhead might make it difficult to be production ready
3. Sometimes simple is enough - The ability to customize models in almost a plug-n-play manner has to come with some cost.

```
pip install --process-dependency-links git+https://github.com/pymc-devs/
     pymc3
```

woot

woot

coin flip example

woot

Recommenders

woot

Recommenders

-

woot