Programming with Estimates

James Bornholt
University of Washington
Dr. Kathryn McKinley
Dr. Matthias Felleisen
Dr. Shan Lu
Dr. Steve Blackburn
Dr. Michael Carbin
Dr. Matthew Might
Dr. Jason Mars
Dr. Mary Hall
Dr. John Davis
Dr. Shan Shan Huang

Dr. Milind Kulkarni
Dr. Armando Solar-Lezama
Dr. Lingjia Tang
James Bornholt
Dr. Ben Wiedermann
Dr. Sam Blackshear
Dr. Alvin Cheung
Dr. Ravi Chugh
Dr. Cindy Rubio-Gonzalez
Dr. Jean Yang
Programming with Estimates

James Bornholt
University of Washington

with:
Todd Mytkowicz
Kathryn S. McKinley
Microsoft Research

Na Meng
Virginia Tech

Adrian Sampson
Cornell

Dan Grossman
Luis Ceze
University of Washington
59 mph
<table>
<thead>
<tr>
<th>Place</th>
<th>Country</th>
<th>Name</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>JAM</td>
<td>Usain Bolt</td>
<td>9.63</td>
</tr>
<tr>
<td>2</td>
<td>JAM</td>
<td>Yohan Blake</td>
<td>9.75</td>
</tr>
<tr>
<td>3</td>
<td>USA</td>
<td>Justin Gatlin</td>
<td>9.79</td>
</tr>
<tr>
<td>4</td>
<td>USA</td>
<td>Tyson Gay</td>
<td>9.80</td>
</tr>
<tr>
<td>5</td>
<td>USA</td>
<td>Ryan Bailey</td>
<td>9.88</td>
</tr>
<tr>
<td>6</td>
<td>NED</td>
<td>Churandy Martina</td>
<td>9.94</td>
</tr>
<tr>
<td>7</td>
<td>TRI</td>
<td>Richard Thompson</td>
<td>9.98</td>
</tr>
<tr>
<td>8</td>
<td>JAM</td>
<td>Asafa Powell</td>
<td>11.99</td>
</tr>
<tr>
<td>Rank</td>
<td>Country</td>
<td>Name</td>
<td>Time</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>-------------------</td>
<td>-------</td>
</tr>
<tr>
<td>1</td>
<td>AUS</td>
<td>James BORNHOLT</td>
<td>3.79s</td>
</tr>
<tr>
<td>1</td>
<td>JAM</td>
<td>USAIN BOLT</td>
<td>9.63s</td>
</tr>
<tr>
<td>2</td>
<td>JAM</td>
<td>YOHAN BLAKE</td>
<td>9.75s</td>
</tr>
<tr>
<td>3</td>
<td>USA</td>
<td>JUSTIN GATLIN</td>
<td>9.79s</td>
</tr>
<tr>
<td>4</td>
<td>USA</td>
<td>TYSON GAY</td>
<td>9.80s</td>
</tr>
<tr>
<td>5</td>
<td>USA</td>
<td>RYAN BAILEY</td>
<td>9.88s</td>
</tr>
<tr>
<td>6</td>
<td>NED</td>
<td>CHURANDY MARTINA</td>
<td>9.94s</td>
</tr>
<tr>
<td>7</td>
<td>TRI</td>
<td>RICHARD THOMPSON</td>
<td>9.98s</td>
</tr>
<tr>
<td>8</td>
<td>JAM</td>
<td>ASAFA POWELL</td>
<td>11.99s</td>
</tr>
</tbody>
</table>
GeoPoint Prev = Get();
Sleep(5);
GeoPoint Curr = Get();
double Dist = Distance(Prev, Curr);
double Speed = Dist / 5;
Uncertain<T>
An abstraction for programming with estimates
Automates complex statistics!

Uncertain<GeoPoint> Prev = Get();
Sleep(5);
Uncertain<GeoPoint> Curr = Get();
Uncertain<double> Dist = Distance(Prev, Curr);
Uncertain<double> Speed = Dist / 5;

86% more accurate!

5 mph
In the beginning...

We want to make programming with sensors easier.

We already have the cute name: Uncertain<T>.

Nah, this will never work.
Probabilistic programming

Programming language support for **probabilistic modeling**

Let’s build a recommendation system!

\[
P[\text{Likes PL}|\text{Paper 1}] = \ldots \\
P[\text{Likes PL}|\text{Paper 2}] = \ldots \\
P[\text{Paper 3}|\text{Likes PL}] = \ldots \\
\]

\[
\text{paper1} = \ldots \\
\text{likesPL} = f(\text{paper1}, \ldots) \\
\ldots
\]
Applications

public class GeoCoordinate {
    public double Latitude;
    public double Longitude;
    public double HorizontalAccuracy;
}

106 apps that use GPS
1 app that reads HorizontalAccuracy
Uncertain<T>

Programs are graphical models

A = Get()
B = Get()
C = Get()
D = A / B
E = D - C

So long as the data sources are updated, we can get distributions for your code for free!
Uncertain<T>

Conditionals in your code are great places to do inference

```csharp
if (Speed > 4)
    Alert("You rock!");
```

Compute the probability that the condition is true, and use a hypothesis test to decide the branch.

Borrow from medical science to provide fully automated inference.
Uncertain<T> for machine learning

Neural networks give only a single output

```
if (Sobel(p) > 0.1)
    EdgeFound();
```

36% false positives!
Uncertain<T> for machine learning

Neural networks give only a single output

```cpp
if (Sobel(p) > 0.1)
    EdgeFound();
```

36% false positives!
Uncertain\(<T>\) for machine learning

Neural networks give only a single output

if (Sobel(p) > 0.1)
    EdgeFound();

36% false positives!
Uncertain<$T>$ for machine learning

Neural networks give only a single output

if (Sobel(p) > 0.1)
    EdgeFound();

36% false positives!
Done is better than good

We really need another case study for the paper.

I’ve been taking a machine learning class this semester...

MATLAB → CSV → C#
Done is better than good

We really need another case study for the paper.

I’ve been taking a machine learning class this semester...

MATLAB → CSV → Python → CSV → C#

R

CSV
Done is better than good

We really need another case study for the paper.

I’ve been taking a machine learning class this semester...

MATLAB → CSV → Python → CSV → C# → CSV → R → CSV → Python

Precision/Recall (%)

<table>
<thead>
<tr>
<th>Threshold</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.5</td>
<td>0.6</td>
</tr>
<tr>
<td>0.5</td>
<td>0.7</td>
<td>0.8</td>
</tr>
<tr>
<td>0.6</td>
<td>0.9</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Uncertain (T) → Precision = Recall

Conditional threshold

Naive Precision

Naive Recall
Neural networks make great examples

Optimizing Synthesis with Metasketches

James Bornholt  Emina Torlak  Dan Grossman  Luis Ceze
University of Washington, USA
{bornholt, emina, djg, luisceze}@cs.washington.edu

Figure 10. The sketch for a neural network is an SSA-form implementation of its evaluation function, with holes for each weight. In this example, the input nodes are the grayscale values of each pixel in the input image, and the output is a binary classification.
PLSE @ University of Washington