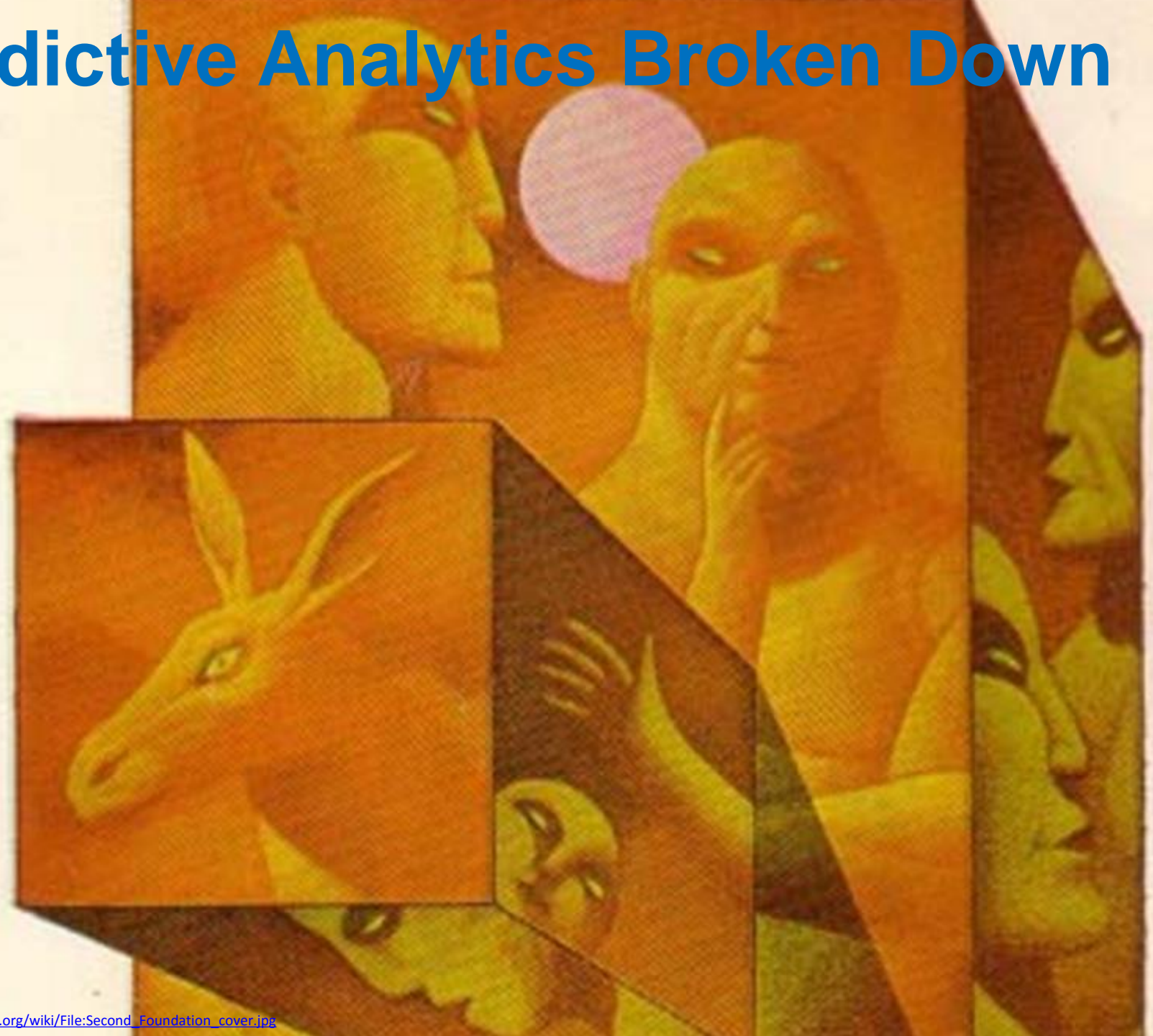


Predictive Analytics Broken Down



Who is this guy?

CEO / Co-Founder Conductrics

www.conductrics.com

matt@conductrics.com

Past: Database Marketing

Education: Artificial Intelligence & Economics



twitter: @mgershoff, @conductrics

Email: matt@conductrics.com



What is Conductrics?

1. Cloud-based Adaptive Testing and Decision Engine

2. API-Based Testing, Targeting and Optimization

- *REST API: Compatibility with CMS systems and other platforms*
- *Native Programming Wrappers (iOS, PHP, jQuery, Node, etc.)*
- *New JavaScript API for super fast decisions at scale*

3. “WAX” Framework for point-and-click style customers

- *Client-side, tag-based, “skip IT” style implementation*

4. Browser UI

- *Admin Console*
 - *Reporting*
- twitter: @mgershoff, @conductrics
Email: matt@conductrics.com



What does Conductrics do?

1. Experimentation

- *AB and Multivariate Testing*
- *Adaptive / Bandit Testing*

2. Personalization

- *Targeting with Business Logic*
- *Targeting via machine learning*

twitter: @mgershoff, @conductrics

Email: matt@conductrics.com



Promise of Predictive Analytics

The Promises:

- Help make predictions about the future
- Predictions about customer:
 - Preferences
 - Intent



Benefits of Predictive Analytics

The Benefits:

- Provide customers with right set of experiences
- Eliminate marketing waste



Why care how it works?

- Better consumer of predictive analytics tools
- How to get the most out of it predictive analytics
- Help ensure you understand its limitations

Scope of talk?



All Predictive Analytics
Uses

Transactional
System



Two Requirements for Personalization

1. Data

2. Logic

Data: 'Sensing' the World



Types of Data

Observable

- **Return Customer**
- **Weekend/Weekday**
- **Mobile/DeskTop**
- **Browser Type**
- **User Age**
- **Geo/Census**
- **Weather**
- **Tenure/RFM Score**

Intervention

- **Lottery Game**
- **Price**
- **Sales Offers**
- **Shipping Type**
- **Layout/UX**
- **Which Products**
- **Suggested Quantities**

Two Requirements for Personalization

1. Data

2. Logic



Requirements for Personalization

Decision logic links

Observations to Actions



How to come up with the Logic?



IF [Customer]

THEN

[Experiences?]

How to come up with the Logic?



IF [Customer]

THEN

[Experiences?]

...in way that Optimizes Performance

Example: Veikkaus

The screenshot shows the top navigation bar of the Veikkaus website. On the left is the Veikkaus logo. In the center are input fields for 'Tunnus' (username) and 'Salasana' (password), followed by a green 'Kirjaudu' (login) button with the text 'Unohtuiko tunnus?' (Forgot password?). To the right is an orange 'Rekisteröidy' (register) button, a '18+' age restriction icon, and the text 'Avoinna 24h' (Open 24h). On the far right are links for 'Asiakaspalvelu' (customer service), 'Chat', 'Points', and 'Asiakasedut' (customer benefits).

Below the navigation bar is a dark banner with a 'Kampanja' (campaign) dropdown menu. The main banner features a colorful illustration of five cartoon birds (purple, green, red, blue, and yellow) perched on a ledge. The text 'Nokkapokka' is written in large, stylized green letters. To the right of the birds, the text 'Nokkapokka 60 000€' is displayed in large white font. Below this, a white box contains the game's name 'Nokkapokka' and its price '1-4€'. The text describes the game as a pyramid game where the goal is to find a pair of birds to win a larger sum. At the bottom of the box are two buttons: a green 'Pela' (play) button with a right arrow and a white 'Kokeile maksutta' (try for free) button.

VEIKKAUS Tunnus Salasana **Kirjaudu** Unohtuiko tunnus? **Rekisteröidy** **18+** Avoinna 24h Asiakaspalvelu Chat Points Asiakasedut

Kampanja ▼ Loton potti nousee uuteen ennätykseen – potissa 14,1 miljoonaa euroa ▼

Nokkapokka

60 000€

Nokkapokka **1-4€**

Kansansanontakin sen tietää: Parempi pulu pivossa kuin kymmenen pyramidissa. Tai ei se sanonta ihan niin mene, mutta tästä on kyse riemastuttavassa Nokkapokka-nettiarvassa. Mitä useammalle linnulle löydät parin, sitä suuremmaksi voittosummasi kasvaa.

→ **Pela** **Kokeile maksutta**

Anna palautetta

twitter: @mgershoff

Example: Lottery Games

Show high price games ...



A screenshot of a lottery game advertisement. The main text 'ONNENRIVIT' is written in large, yellow, 3D-style letters. Below the text is a cartoon illustration of a woman with orange hair and a red, spotted headpiece, with her hand raised. At the bottom, there is a purple bar containing the price '2,50 €-7 €', an information icon 'i', and the prize amount '200 000 €'. A blue arrow points to the price information.

2,50 €-7 € i 200 000 €

Example: Lottery Games

Or show the low price games



Example: Lottery Games

To keep it simple just look at:

- New or Repeat Player
- Weekday or Weekend

How to come up with the Logic?



**IF [Repeat and/or
Weekend]**

THEN [High/Low Price?]

...in order to be most profitable

How its Done

1 Learn how Repeat and Weekend customers predict low price games



How its Done

1 Learn how Repeat and Weekend customers predict low price games



2 Learn how Repeat and Weekend customers predict high price games



How its Done

1 Learn how Repeat and Weekend customers predict low price games



2 Learn how Repeat and Weekend customers predict high price games



3 Then compare for each customer
(Choose the one with the highest value)

Predictive Analytics Methods

- **Deep Learning Nets**
- **Decision Trees**
- **Gaussian Process (is a Bayesian method)**
- **Support Vector Machines**
- **KNN - actually kinda like segmentation**
- **Naive Bayes (is NOT a Bayesian method)**
- **Logistic Regression**



<http://conductrics.com/data-science-resources/>

<http://conductrics.com/data-science-resources-2>

**We are going to use
Linear Regression**

Why Linear Regression?

Benefits:

- 1. Has nice Statistical Properties**
- 2. Easy(ish) to interpret**
- 3. In practice, often all you need**

What is Linear Regression

A model of relationships in this form:

What is Linear Regression

A model of relationships in this form:

Prediction = **Base** + **B1*****Attribute1** ... + **Bj*****Attributej**

What is Linear Regression

A model of relationships in this form:

Prediction = **Base** + **B1*****Attribute1** ... + **Bj*****Attributej**

Just Add up all of the customer 'attributes' by the impact (**B**) of the Feature

What is Linear Regression

We will learn two models, one for each game:

What is Linear Regression

We will learn two models, one for each game:

$$\text{Game High} = \text{Base}_H + W_H * \text{Weekend} + R_H * \text{Return}$$

What is Linear Regression

We will learn two models, one for each game:

$$\text{Game High} = \text{Base}_H + W_H * \text{Weekend} + R_H * \text{Return}$$

$$\text{Game Low} = \text{Base}_L + W_L * \text{Weekend} + R_L * \text{Return}$$

Linear Regression + Sequential Learning

Benefits of Sequential Learning

1. Don't have to wait to collect the data

Benefits of Sequential Learning

1. Don't have to wait to collect the data

2. Constantly updating you can use it real time

Benefits of Sequential Learning

- 1. Don't have to wait to collect the data**
- 2. Constantly updating you can use it real time**
- 3. Scalable—any real production PA is almost certainly going to use the method**

Benefits of Sequential Learning

- 1. Don't have to wait to collect the data**
- 2. Constantly updating you can use it real time**
- 3. Scalable—any real production PA is almost certainly going to use the method**
- 4. The computations are simple to understand**

We Can Do It!



twitter: @mgershoff

The Sequential Algorithm in words

- 1) Observe the data for a single customer
- 2) Using the current parameter values to make a prediction
- 3) See how far off your predicted value was from the actual value
- 4) Use how far off you prediction was to update your parameter values
- 5) Adjust how much you update by something like $O(1/n)$
– sort of like an average
- 6) Repeat

The Sequential Algorithm

$$\text{Adjustment} = (\text{Predicted} - \text{Actual}) * 1/\text{sqrt}(n)$$



The Difference (Error) of the actual value and the predicted result

$$\text{Parameter}_{\text{New}} := \text{Parameter}_{\text{old}} - \text{Adjustment}$$



Current Value

How it is done: No data yet, high cost game

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
0	0	0					

Observe New Customer on Weekend

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
0	0	0	0	1	1.00		

Observe New Customer on Weekend

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
0	0	0	0	1	1.00		

$$\text{Prediction} = \text{Base}_H + W_H * \text{Weekend} + R_H * \text{Return}$$

Plug in values

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
0	0	0	0	1	1.00	0	-1.00

$$0 = 0 + 0 * 0 + 0 * 1$$

Update Base and Weekend Impact Score

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
0	0	0	0	1	1.00	0	-1.00



Updated		
Base	R	W
1	0	1

Observe New Customer Weekday

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
1	0	1	0	0	2.00		

$$\text{Prediction} = \text{Base}_H + W_H * \text{Weekend} + R_H * \text{Return}$$

Plug in values

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
1	0	1	0	0	2.00	1.00	-1.00

$$1 = 1 + 0*0 + 1*0$$

Update Just the Base Impact Score

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
1	0	1	0	0	2.00	1	-1.00



Updated		
Base	R	W
1.5	0	1

Observe Return Customer on Weekday

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
1.5	0	1	1	0	3.00		

$$\text{Prediction} = \text{Base}_H + W_H * \text{Weekend} + R_H * \text{Return}$$

Plug in values

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
1.5	0	1	1	0	3.00	1.50	-1.50

$$3 = 1.5 + 0 * 1 + 1 * 0$$

Update the Base and Return Impact Score

Hidden			What We Know				
Base	R	W	Return	WkEnd	Sales	Predict	Error
1.5	0	1	1	0	3.00	1.50	-1.50



Updated		
Base	R	W
1.75	0.75	1

Online Regression

After 200 Iterations

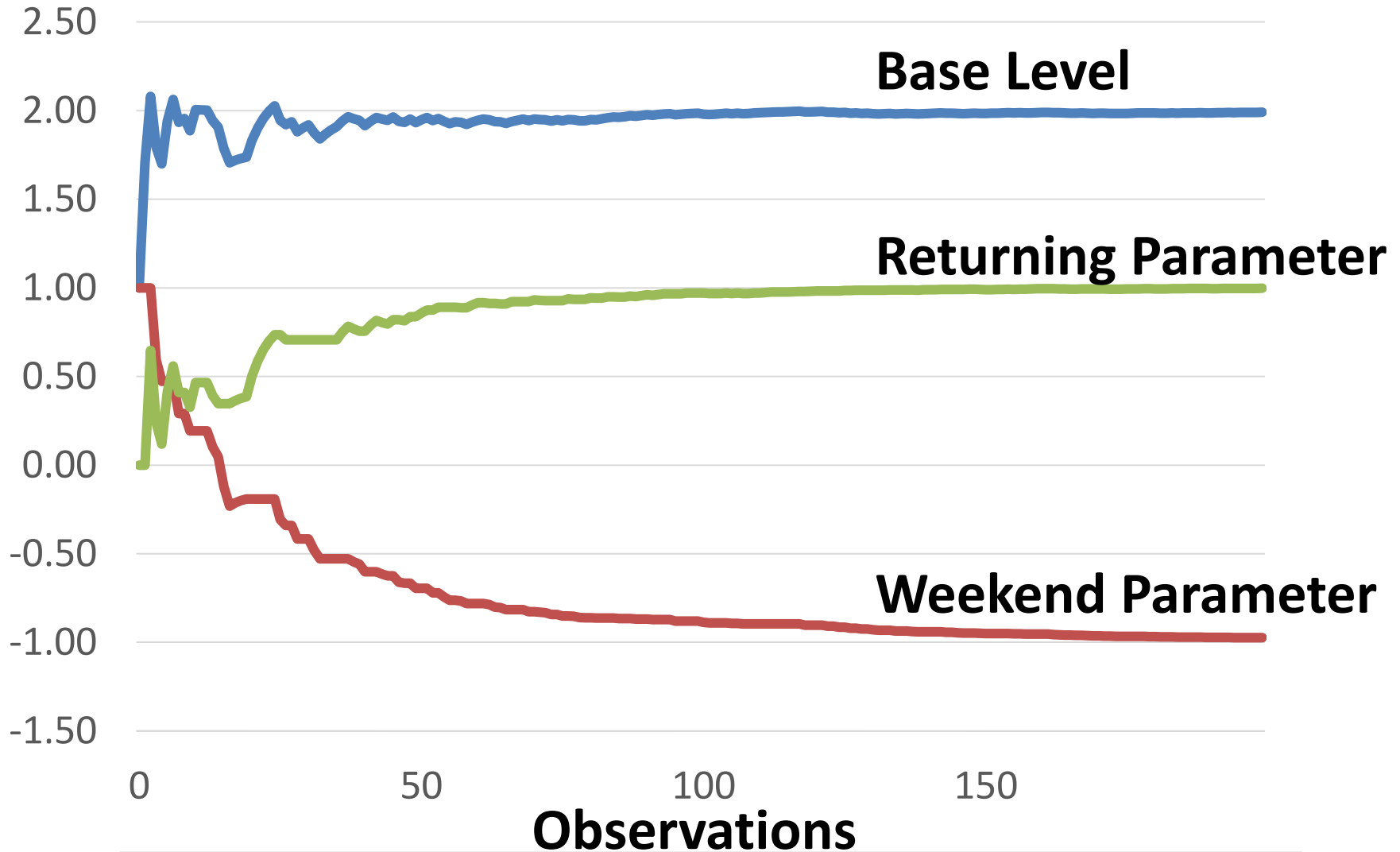
Base	R	W
2.0	1.0	-1.0

High Price Model Results

$\text{Sales} = 2.0 + 1.0 * \text{Return} - 1.0 * \text{Weekend}$

Online Results: 200 Iterations

Parameter Value



Back to Our Task

Model: High Price Game

$$\text{High} = 2.0 + 1.0 * \text{Return} - 1.0 * \text{Weekend}$$



Back to Our Task

Model Low Price Game

$$\text{Low} = 1.0 + 1.0 * \text{Return} + 0.5 * \text{Weekend}$$



Tabular Targeting Logic

Returning	Weekend	High Price	Low Price	Selection
N	N	2.0	1.0	High
Y	N	3.0	2.0	High
N	Y	1.0	1.5	Low
Y	Y	2.0	2.5	Low

Targeting Logic as Rule

IF [Weekend]
THEN [Low]
Else [High]

Targeting Logic as Rule

Expressing the logic as a set of succinct rules is generally a hard problem

**IF [Weekend]
THEN [Low]
Else [High]**

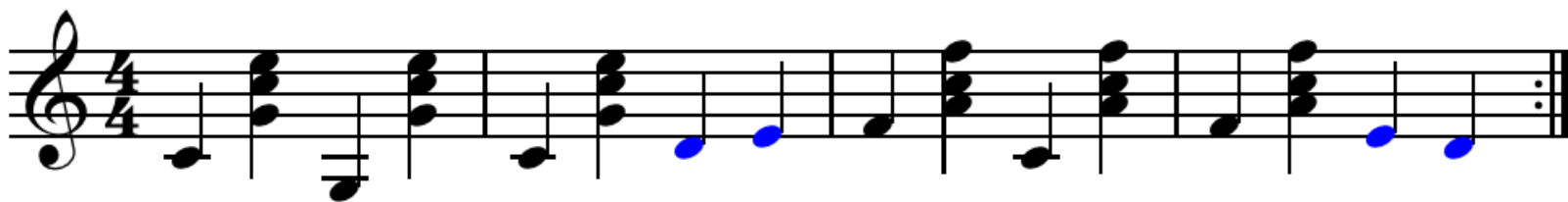
How to evaluate our targeting logic?

How do we measure our Targeting?

**A/B
TEST!!!**

Need a Baseline

Guitar



Random ain't too bad

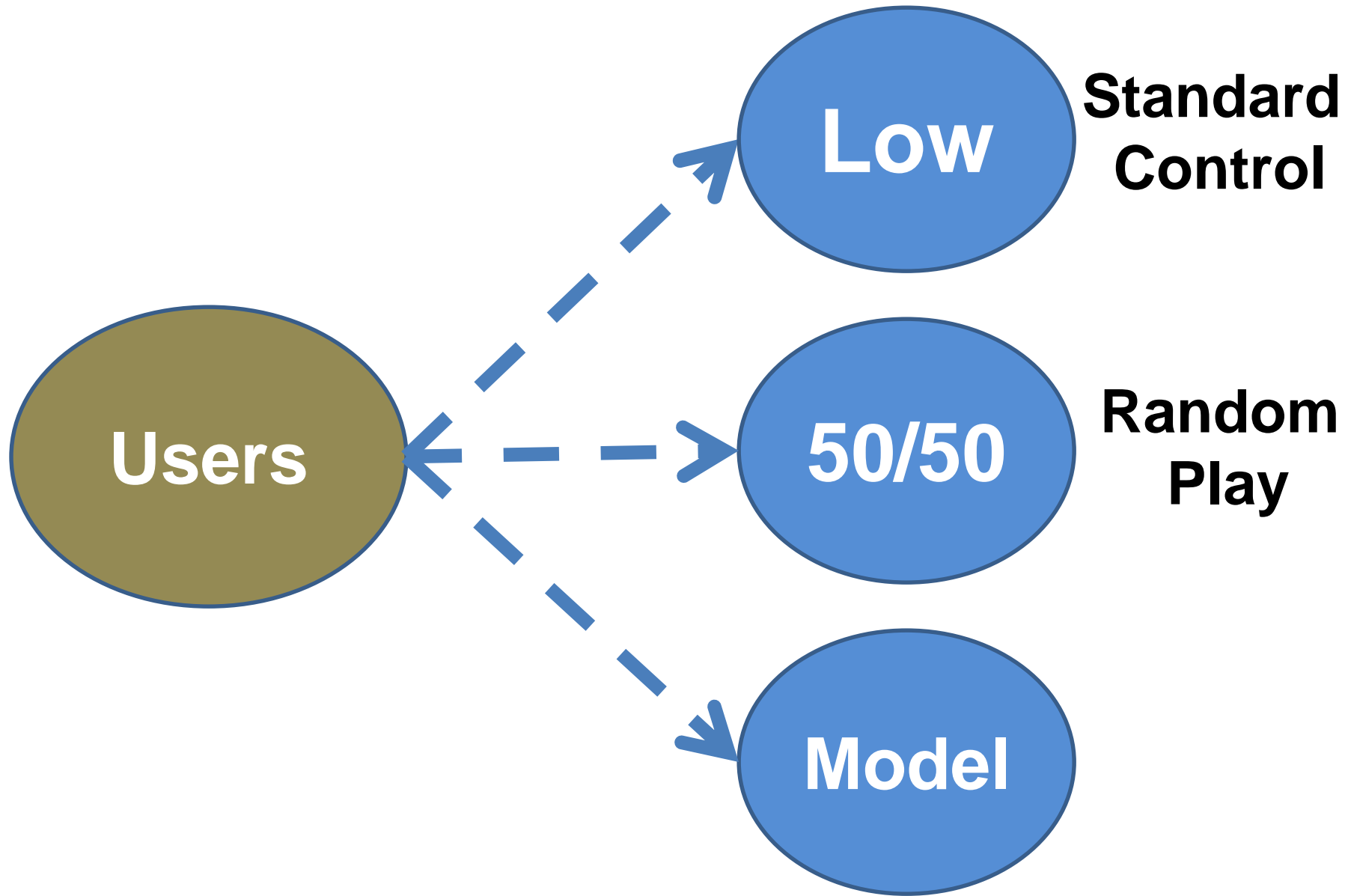
Random Selection as Control

Good:

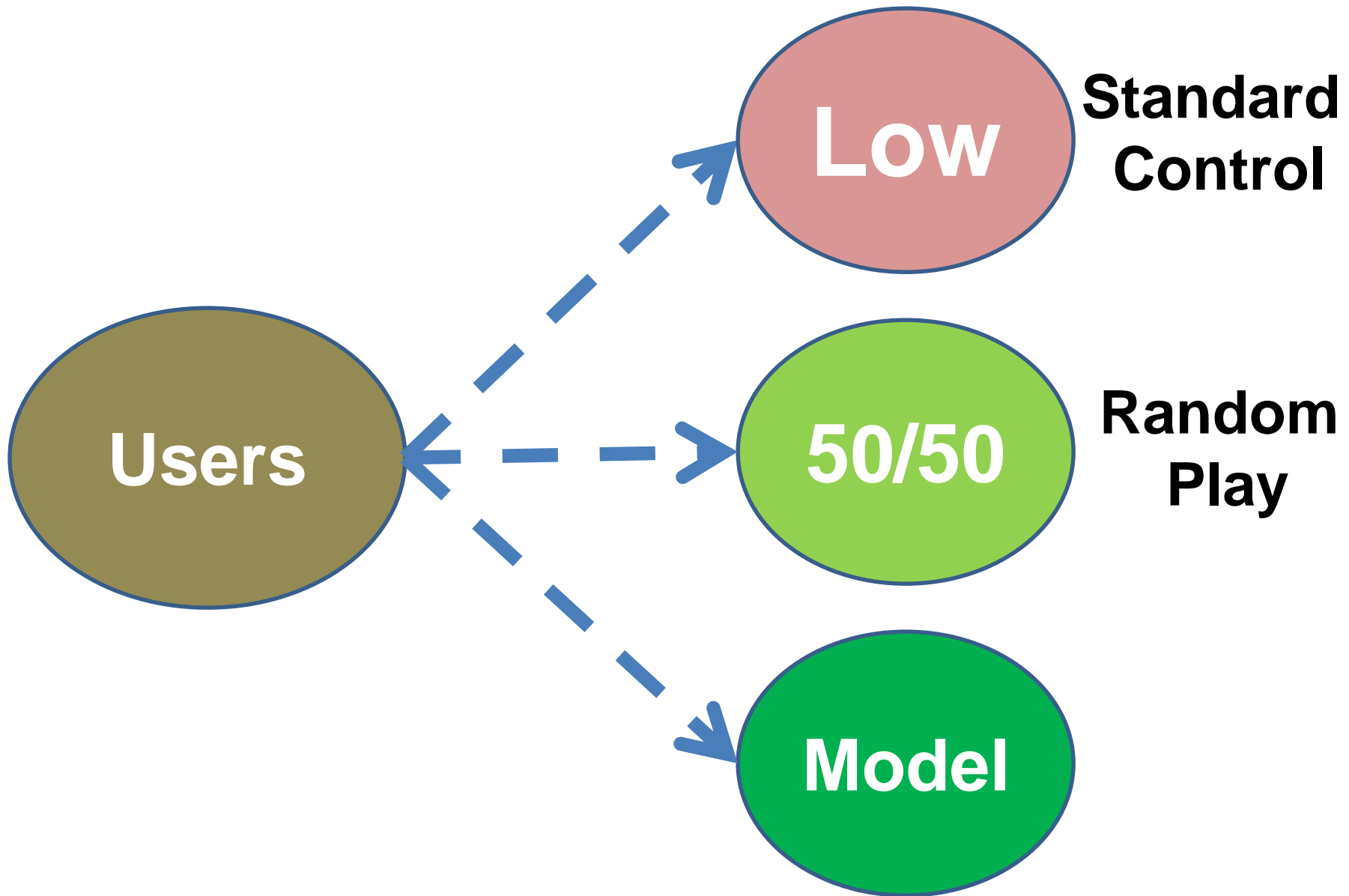
1. A baseline we can always use
2. Selecting randomly is often good policy when we don't have any additional information

Bad: Often hard to beat (which is **Good**)

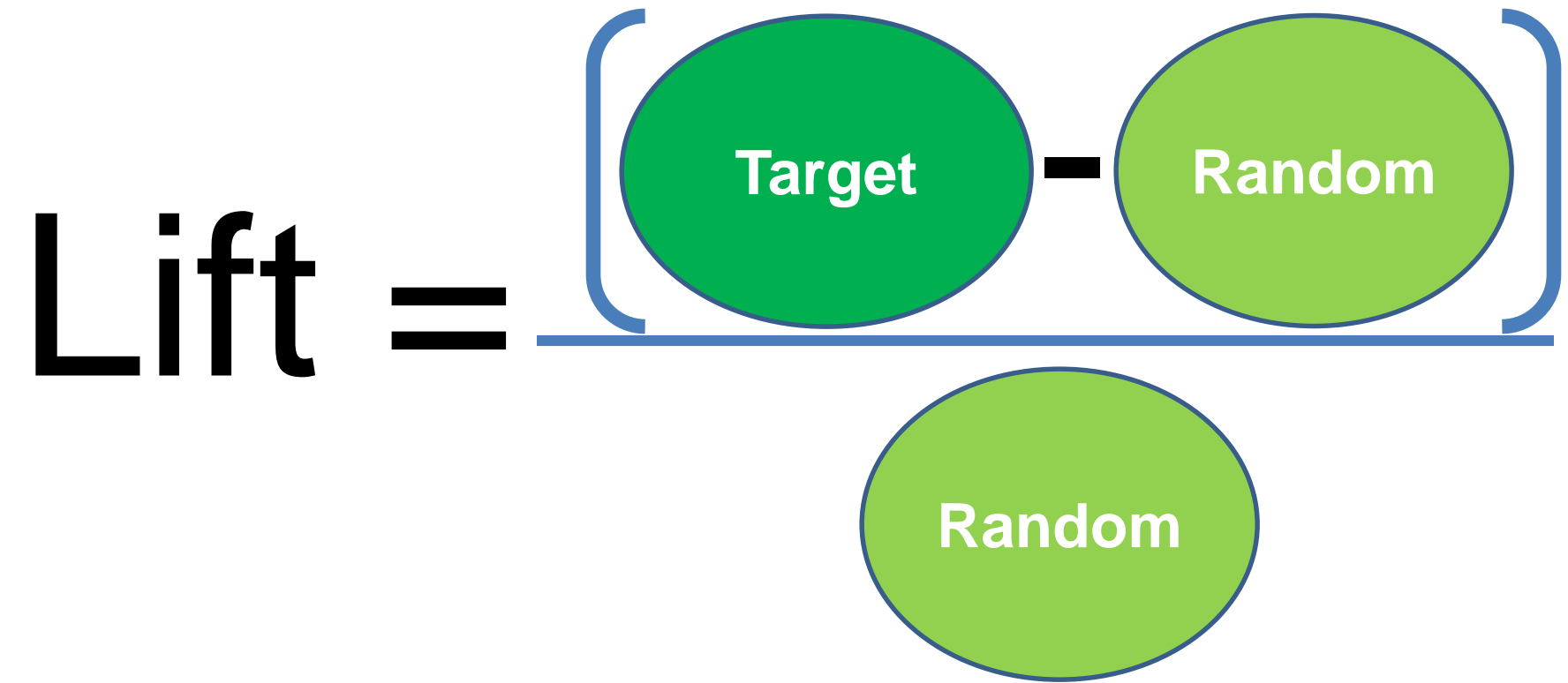
Model Evaluation: AB Style



Compare the Results of Each



Results: Model ROI/Lift



Beware the Faustian Bargain!



Targeting = Complexity

Customer data needs to be 'accurate' AND available at decision time

We need to create AND manage experiences and content

Need to create AND manage our decision logic

Difficult to know what state system is in. Before just one state, now many.

Targeting and the ROI of Complexity

$$\frac{\textit{Value(Complex System)} - \textit{Value(Simple)}}{\textit{Value(Simple)}}$$

What about Segmentation?

Segmentation Doesn't Scale

Just 20 Customer Features

**1 Million
Combinations**



power(2,10) assuming all non mutually exclusive

Image Source: http://en.wikipedia.org/wiki/File:Carl_Sagan_-_1980.jpg

@mgershoff

Wake up. We are Done!



Twitter:mgershoff

Email:matt.gershoff@conductrics.com
