Experiments in the Internet age: A modern Bayesian look at the multi-armed bandit

Steven L. Scott

June 21, 2016

Summary

- The economics of running an experiment in the online service economy are different than manufacturing or agriculture.
- Costs have shifted from production to opportunity cost.
- Multi-armed bandits minimize the opportunity cost of running an experiment.
- Explicitly minimizing cost is hard, but a simple Bayesian heuristic known as Thompson sampling produces good outcomes.

Google

Multi-armed bandits

(a) "one armed bandit"

(b) multi-armed

- Sequential experimental design.
- Produce the highest reward under an uncertain payoff distribution.

Google

Outline

Motivation and background

Thompson sampling

When to stop experimenting

Incorporating good ideas from DOE

Google

Classical experiments

Fisher arrives at Rothamstead in 1919

R. A. Fisher

Rothamstead station.

The classical experiment playbook

- Steps:
  - Decide which factors you want to test.
    Think about possible interactions.
  - Create a "design matrix" that will allow you to estimate the effects that are likely to be important.
  - Power analysis or "optimal design" for sample size at each design point (row of the design matrix).
  - Maybe "block" on contextual factors (men / women) to reduce variance.
  - Do the experiment (collect the data).
  - Analyze the results.
- Get it right in the beginning, because you've only got one shot!
- All the probability calcuations are done before seeing any data, because the experiment had to be planned before any data were observed.

Google

Experiments through the ages

- Agricultural
  - Experiments take a long time (seasons).
  - Inputs (land, labor, machinery) are expensive.
- Industrial
  - Time requirements are typically less.
  - Inputs still expensive. (Factory retooling, lost units)
- Service
  - Experiments fast (real time).
  - Inputs are cheap (programmer time).
  - "All" cost is opportunity cost.

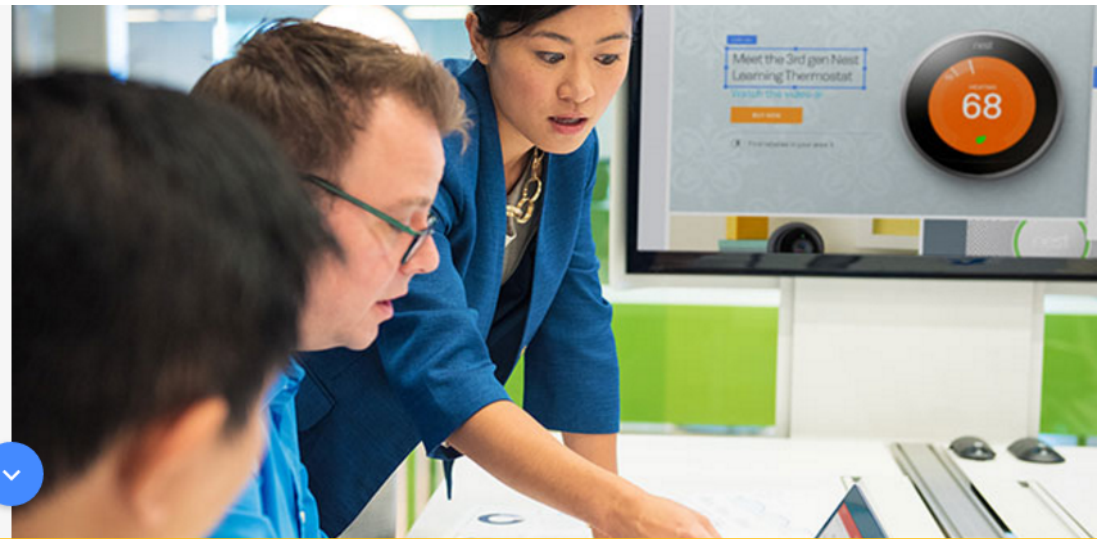
Google

Experiments today: outsource the details

Can be run through online optimization frameworks (e.g. Google Optimize 360).

Test, adapt, personalize.

Discover the most engaging customer experiences with Google Optimize 360 (beta). Test different variations of your site and then tailor it to deliver a personalized experience that works best for each customer and for your business.

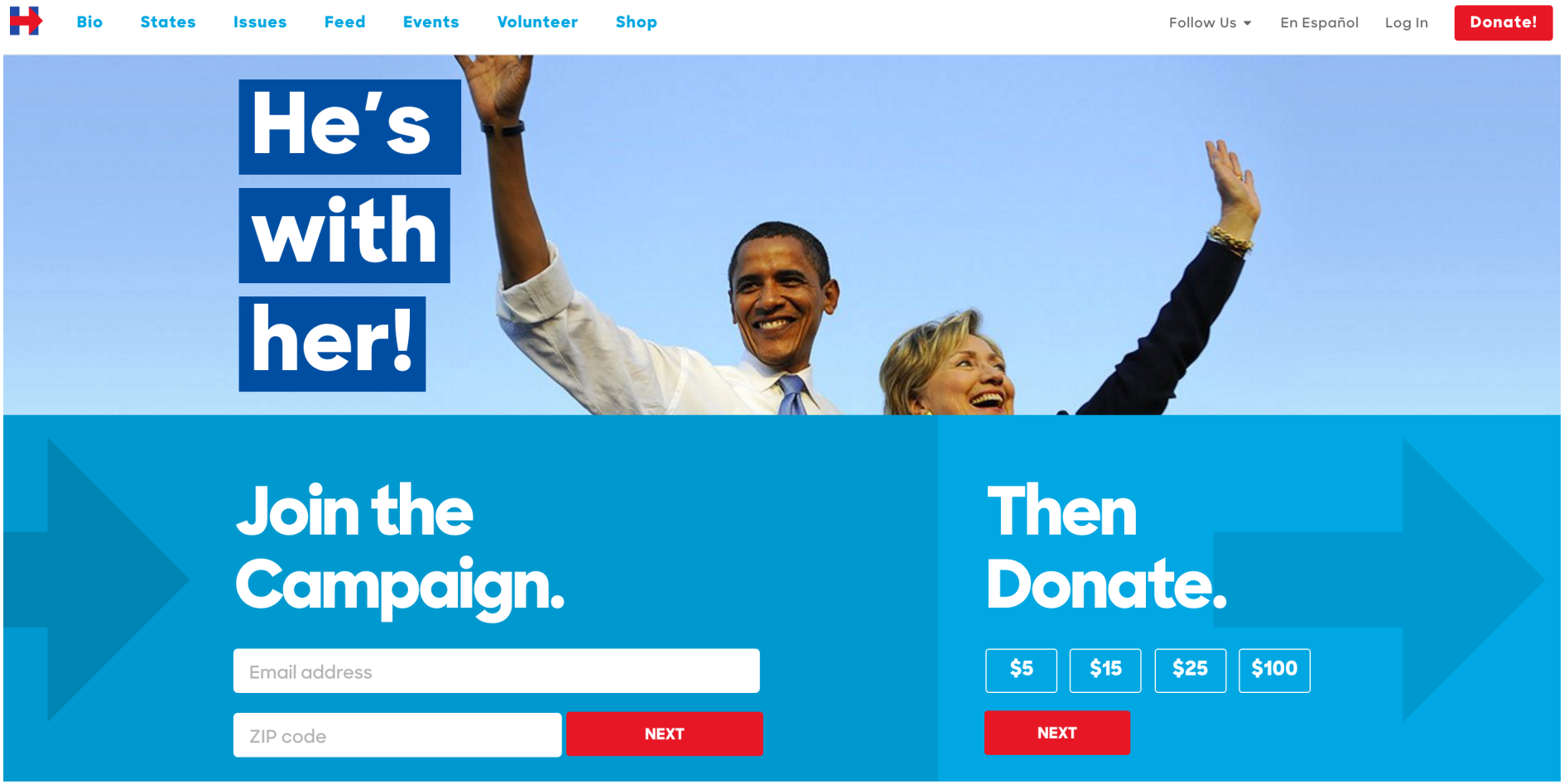


"Optimize 360 has given our team the power to implement tests across our entire site. We have reduced the amount of time it takes to launch a test from a few days to a few minutes."

Erryn Neckel - Director, Marketing Operations, The Motley Fool

Google

Website optimization



Service experiments

The economics are favorable.

- No cost to acquiring experimental units.
- Minimal costs to changing the way to treat them.
- Automated experimental frameworks.

Why not experiment with everything, all the time?

- Some do (esp. big tech companies)
- For many, the opportunity cost is a barrier.

Stats education can do real damage here

From a misguided blogger focused on type-I errors

If you run experiments: the best way to avoid repeated significance testing errors is to not test significance repeatedly.

Decide on a sample size in advance and wait until the experiment is over before you start believing the "chance of beating original" figures that the A/B testing software gives you.

"Peeking" at the data is OK as long as you can restrain yourself from stopping an experiment before it has run its course.

I know this goes against something in human nature, so perhaps the best advice is: no peeking!

- Type I errors cost (effectively) zero. Type II errors are expensive.
- This is terrible advice (he later suggests Bayesian design).

Google

Outline

Multi-armed bandit: Problem statement

- Rewards y are generated from a probability distribution

$$f_a(y|\theta).$$

- **Goal**: Choose action a to maximize your total reward.
- **Challenge**: You don't know θ, so you need to experiment.
- Trade-off:

  Exploit Take the action your model says is best.
  Explore Do something else, in case your model is wrong.

- Uncertainty about θ is the critical piece.

Example reward distributions

Binomial Rewards are independent 0/1. θ is a vector of distinct success probabilities for each arm.

Logistic Rewards are independent 0/1. θ is a set of logistic regression coefficients for a design matrix. The design matrix may include both experimental and contextual factors.

Restless bandits Rewards accrue in a time series, where the rules are changing. θ_t is a set of logistic regression coefficients modeled as $\theta_t = \theta_{t-1} + \epsilon_t$.

Hierarchical Experiments share structure, perhaps because of between group heterogeneity.

Etc (Continuous rewards, continuous action spaces, your problem here.)

Google

The Thompson heuristic

[Thompson(1933)]

Thompson sampling

Randomly allocate units in proportion to the probability that each arm is "best."

Computing "optimal arm probabilities"

The slow way:

- Given data $\mathbf{y}$, simulate $\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(N)}$ from $p(\theta|\mathbf{y})$.
- For each draw, compute $v_a(\theta)$, the value of action a conditional on θ.
- Let $I_a(\theta)$ be the indicator that action a has the largest value. (Break ties at random).
- Set

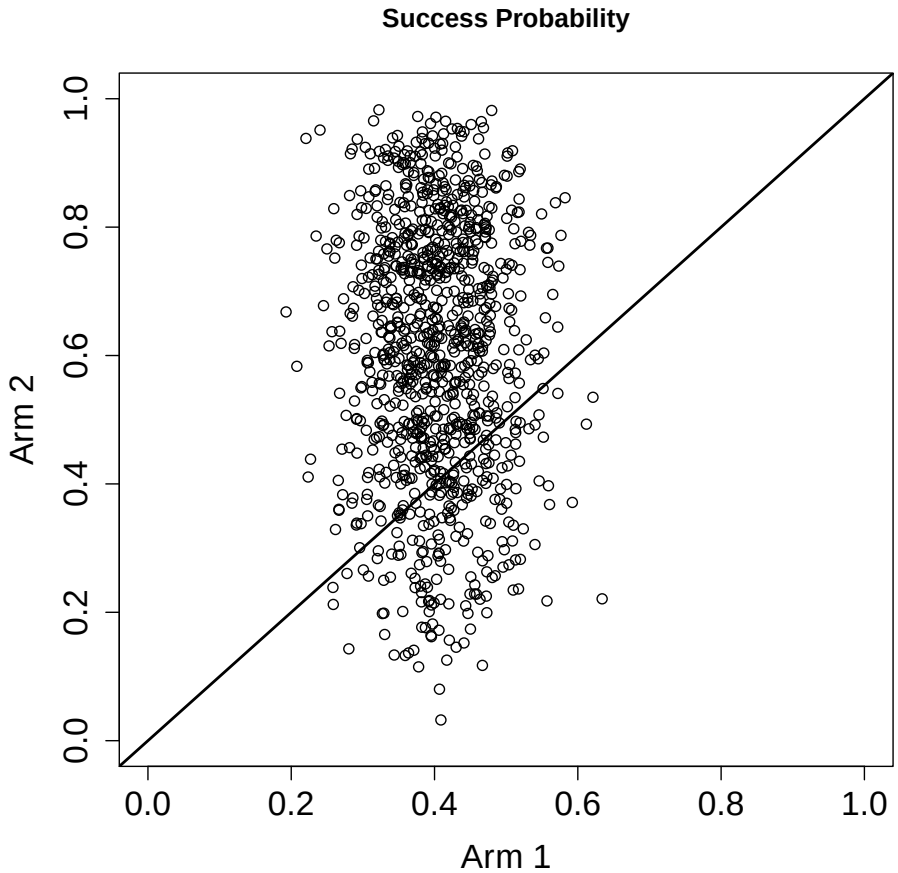
$$w_a = \frac{1}{N} \sum_{g=1}^{N} I_a\left(\theta^{(g)}\right)$$

The fast way:

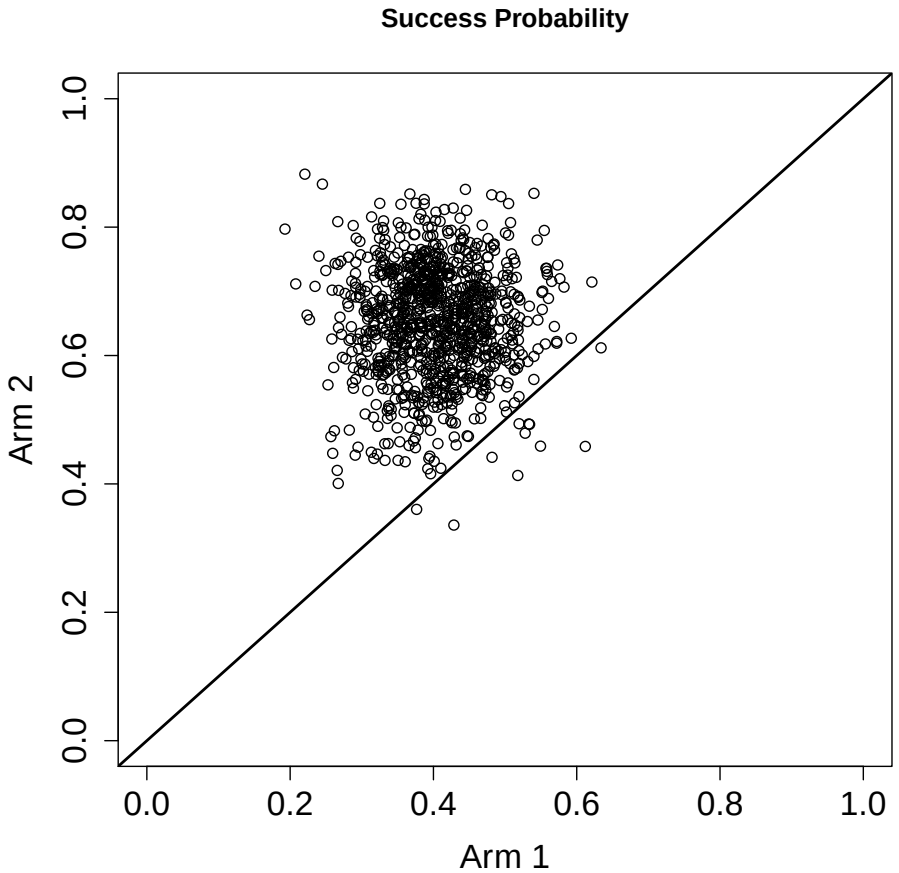
- Draw a single $\theta \sim p(\theta|\mathbf{y})$.
- Choose a to maximize $v_a(\theta)$.

Google

Computing the probability that one arm beats another.



$$X = P(\theta_1 | 20 \text{ successes}, 50 \text{ trials})$$
$$Y = P(\theta_2 | 2 \text{ successes}, 3 \text{ trials})$$
$$P(\theta_1 > \theta_2 | \mathbf{y}) \approx .16$$



$$X = P(\theta_1 | 20 \text{ successes}, 50 \text{ trials})$$
$$Y = P(\theta_2 | 20 \text{ successes}, 30 \text{ trials})$$
$$P(\theta_1 > \theta_2 | \mathbf{y}) \approx .009$$

Google

Some nice features of Thompson sampling

1. Easily understood principle

   Arms attract observations in proportion to their probability of being optimal.

2. Easy to implement
   - Can be applied to a very generally
   - Just need to be able to sample from $p(\theta|\mathbf{y})$
   - Free of arbitrary tuning constants
     - No need to set "exploration fraction" (ϵ-learning)
     - No decay schedule
     - No "discount factor"
   - It can be used with batch updates of the posterior.
3. Nearly optimal performance
   - Obviously suboptimal arms are quickly dropped, increasing reward.
   - Increases the sample size for picking the winner from the "good" arms.
   - Allocations are attracted to performance and uncertainty.

Google

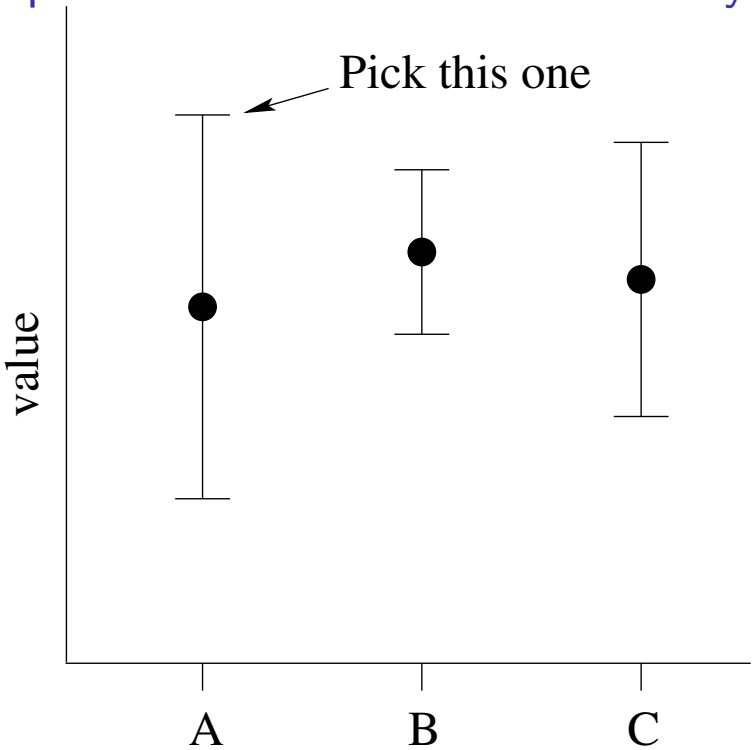
Other methods

Several other well known methods and heuristics [Sutton and Barto(1998)]

- Equal allocation
- Greedy
- $\epsilon-$greedy
- $\epsilon-$decreasing
- Softmax learning
- Upper confidence bound
- Gittins index
- Dynamic programming

Upper confidence bound (UCB)

Optimism in the face of uncertainty.



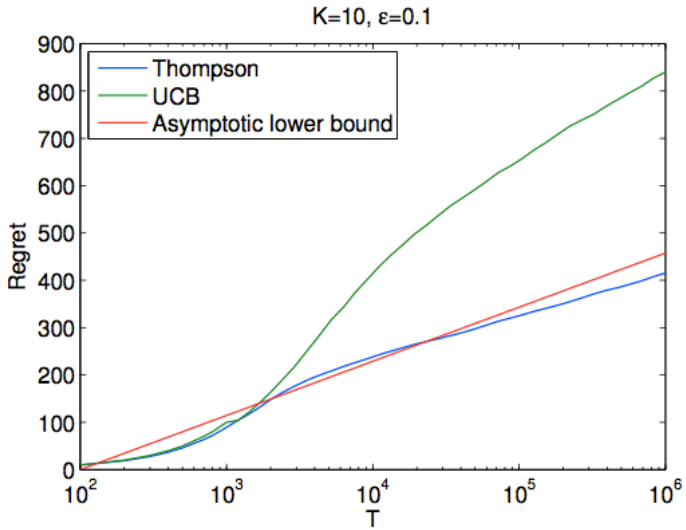
- Pick the arm with the highest upper confidence bound.
- Not the usual bounds from normal approximations: extra factors of $\log n$ in the SE numerator.
- [Auer *et al.*(2002)] showed UCB satisfies optimal rate of exploration [Lai and Robbins(1985)].

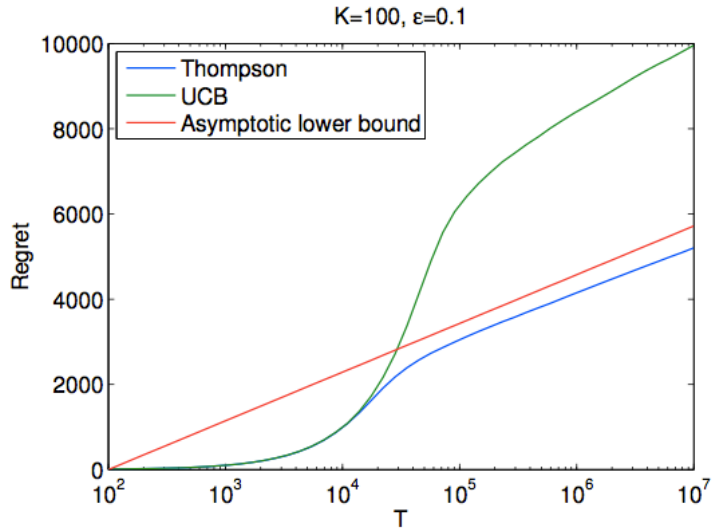
- UCB can be effective, but some skill is needed to find the "right" confidence set.
- Replacing confidence sets with posterior distributions finds the "right" set automatically. [Russo and Van Roy(2014)]

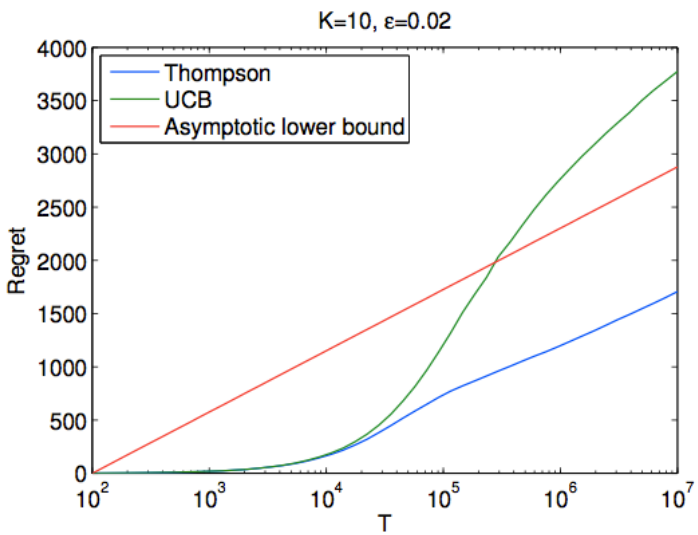
Google

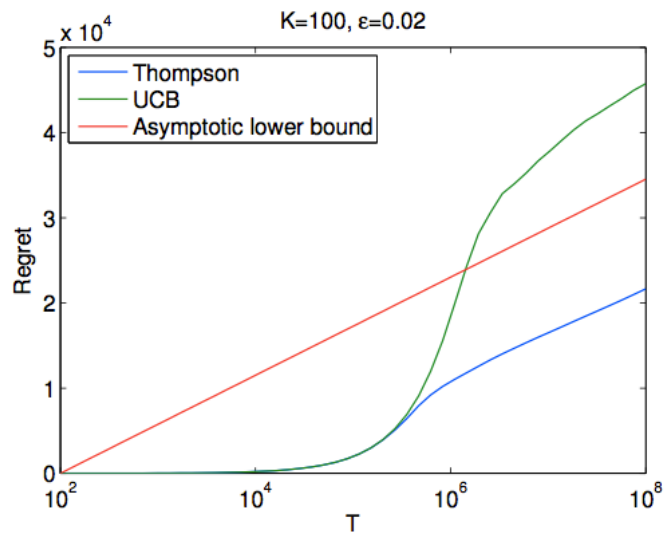
Thompson vs UCB

Results from [Chapelle and Li(2011)]









A timeline of recent literature

- [Scott(2010)] and [Chapelle and Li(2011)] present empirical results suggesting Thompson sampling is a good idea.
- [May *et al.*(2012)] prove asymptotic convergence.
  - All arms are visited "infinitely often."
  - Almost all time spent on optimal arm.
- Several authors provide finite time regret bounds in the case of independent Bernoulli arms, including [Kaufmann *et al.*(2012)], [Bubeck and Liu(2013a), Bubeck and Liu(2013b)] and others.
- [Russo and Van Roy(2014)] is the current state of the art. Regret bounds for nearly arbitrary reward distributions.

How bad is equal allocation?

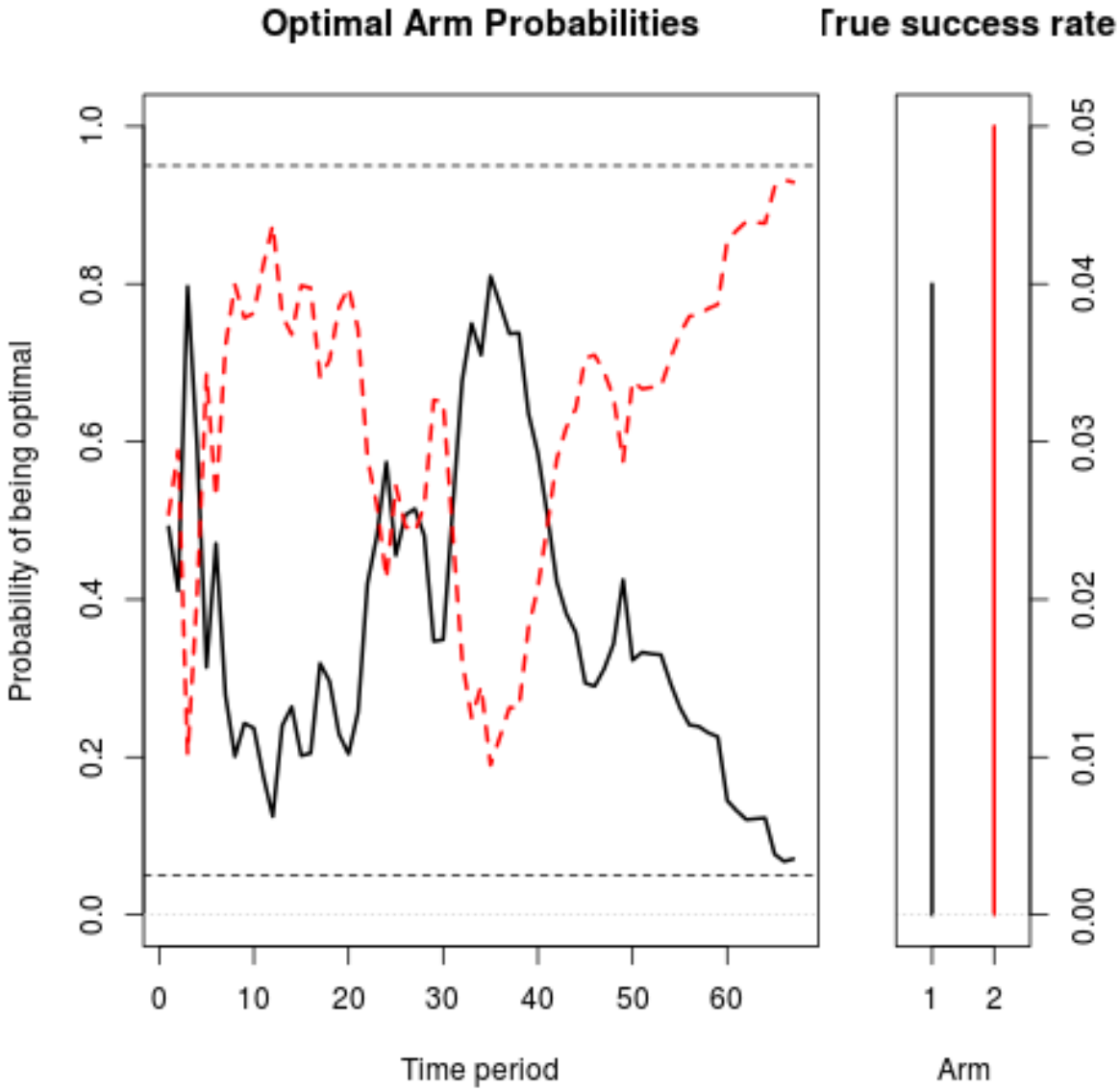
- Consider two arms: $\theta_1 = .04$, and $\theta_2 = .05$.
- Plan a classical experiment to detect this change with 95% power at 5% significance.

```
> power.prop.test(p1 = .04, p2 = .05, power = .95)
              n = 11165.99
NOTE: n is number in *each* group
```

- We need over 22,000 observations.
- Regret is $11,165 \times .01 = 111$ lost conversions.
- At 100 visits per day, the experiment will take over 220 days.

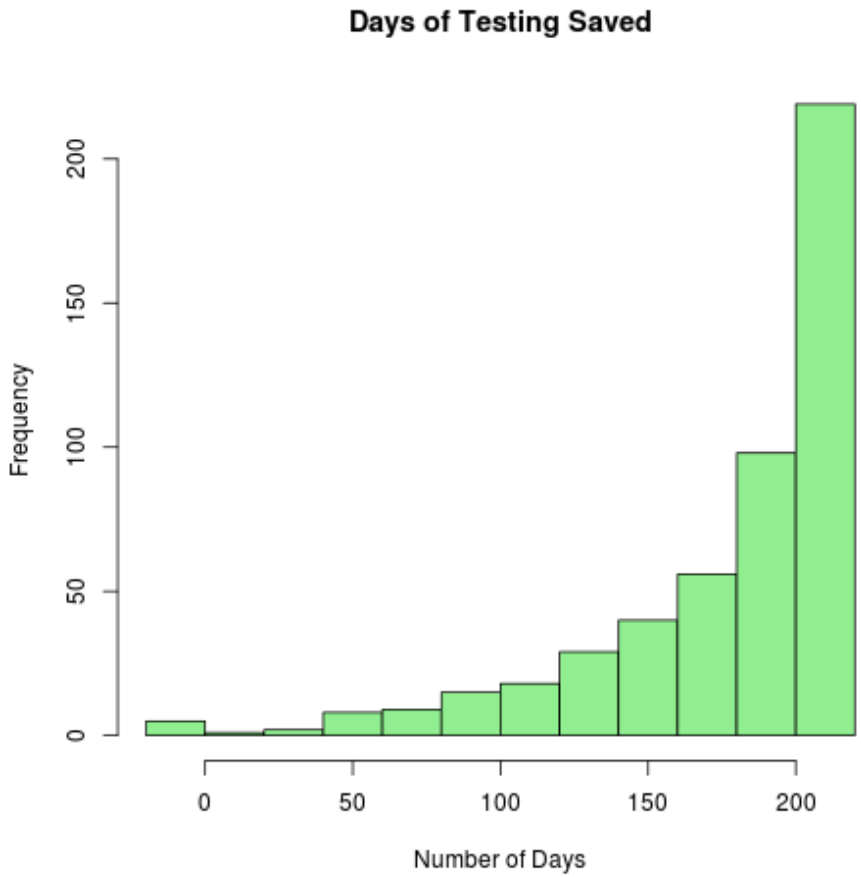
Two-armed experiment

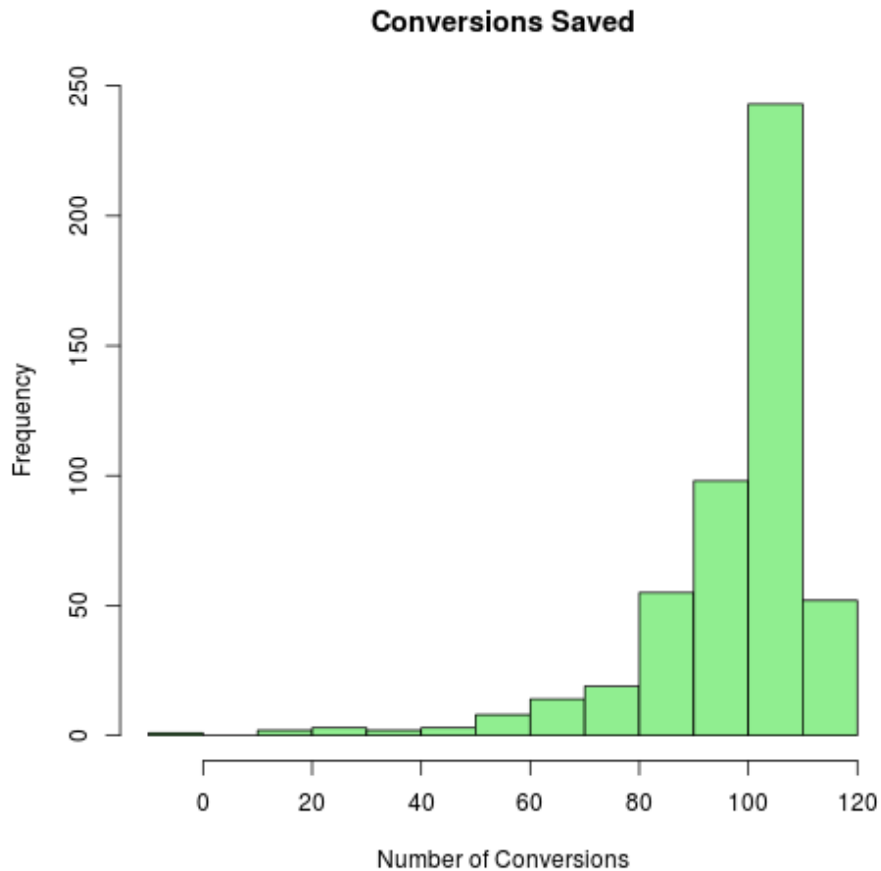
Bandit shown 100 visits per "day"



Two armed experiment

Savings vs equal allocation in terms of time and conversions





Source: https://support.google.com/analytics/answer/2844870?hl=en

Bandits' advantage grows with experiment size

Now consider 6 arms (formerly the limit of GA Content Experiments).

- Compare the original arm to the "best" competitor.
- Bonferroni correction says divide significance level by 5.

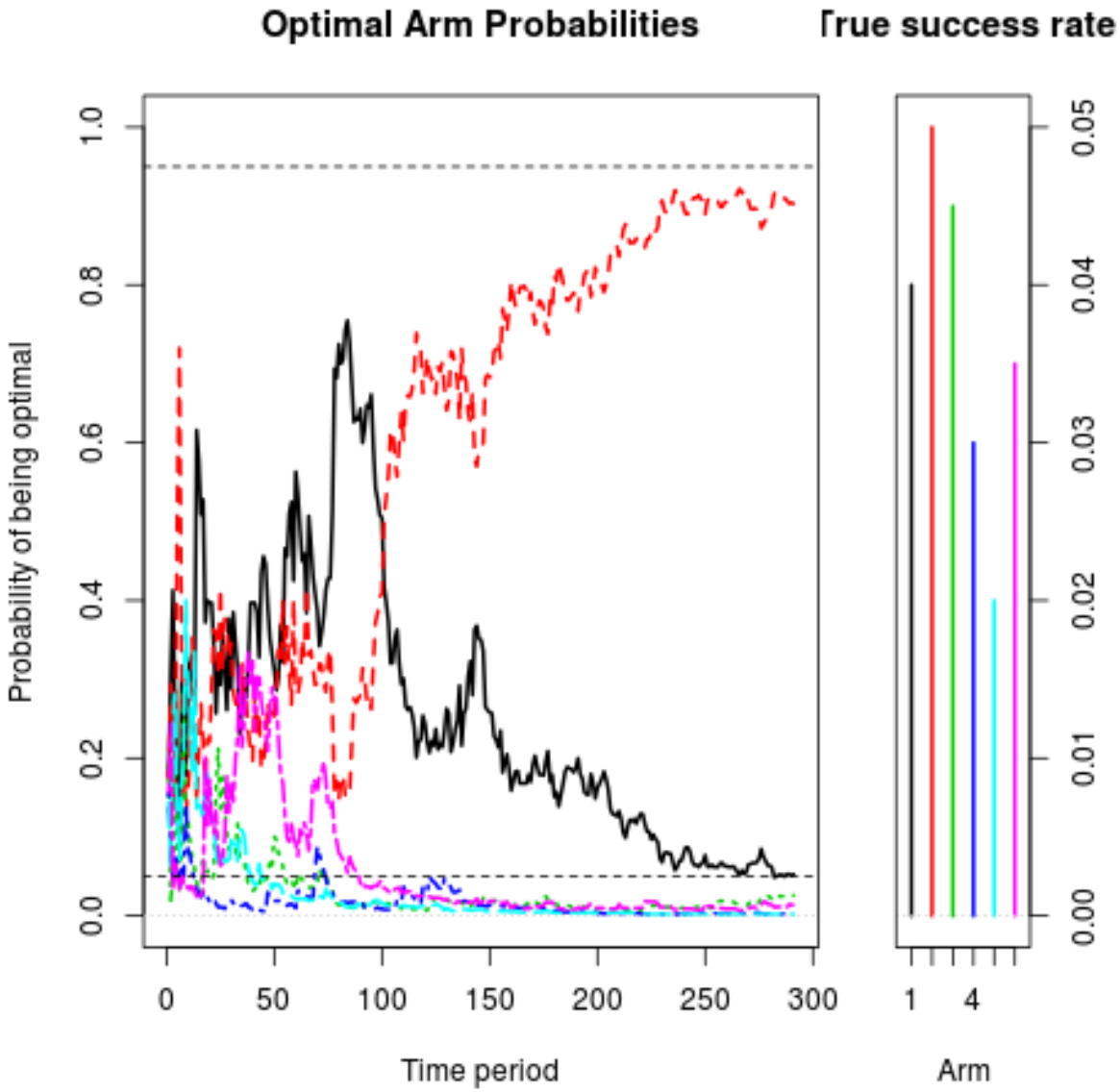
```
> power.prop.test(p1 = .04, p2 = .05, power = .95,
                  sig.level=.01)
             n = 15307.8
NOTE: n is number in *each* group
```

- In theory we only need this sample size in the largest arm, but we don't know ahead of time which arm that will be.
- Experiment needs 91848 observations.
- At 100 per day that is 2.5 years.

Google

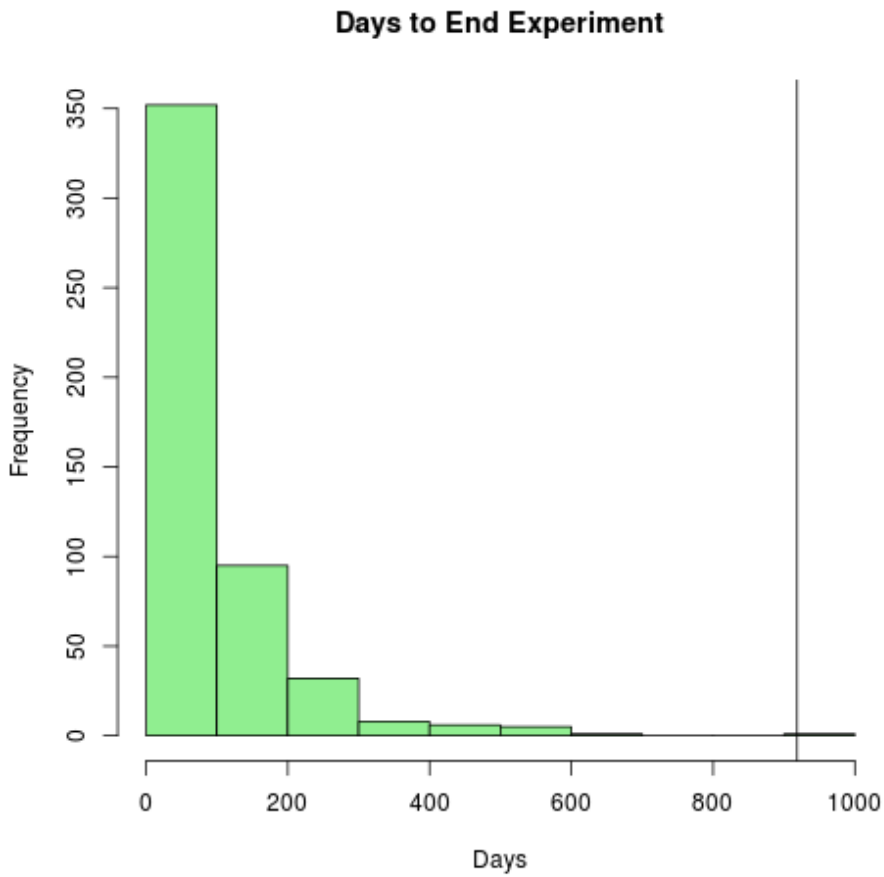
6-arm experiment

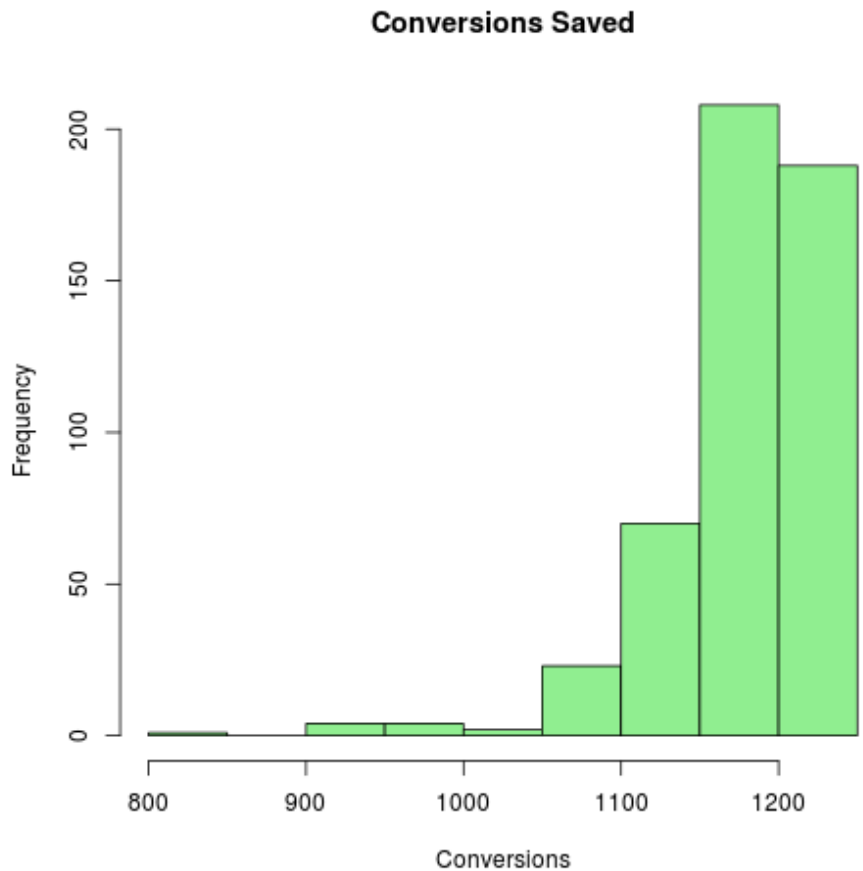
Still 100 observations per day



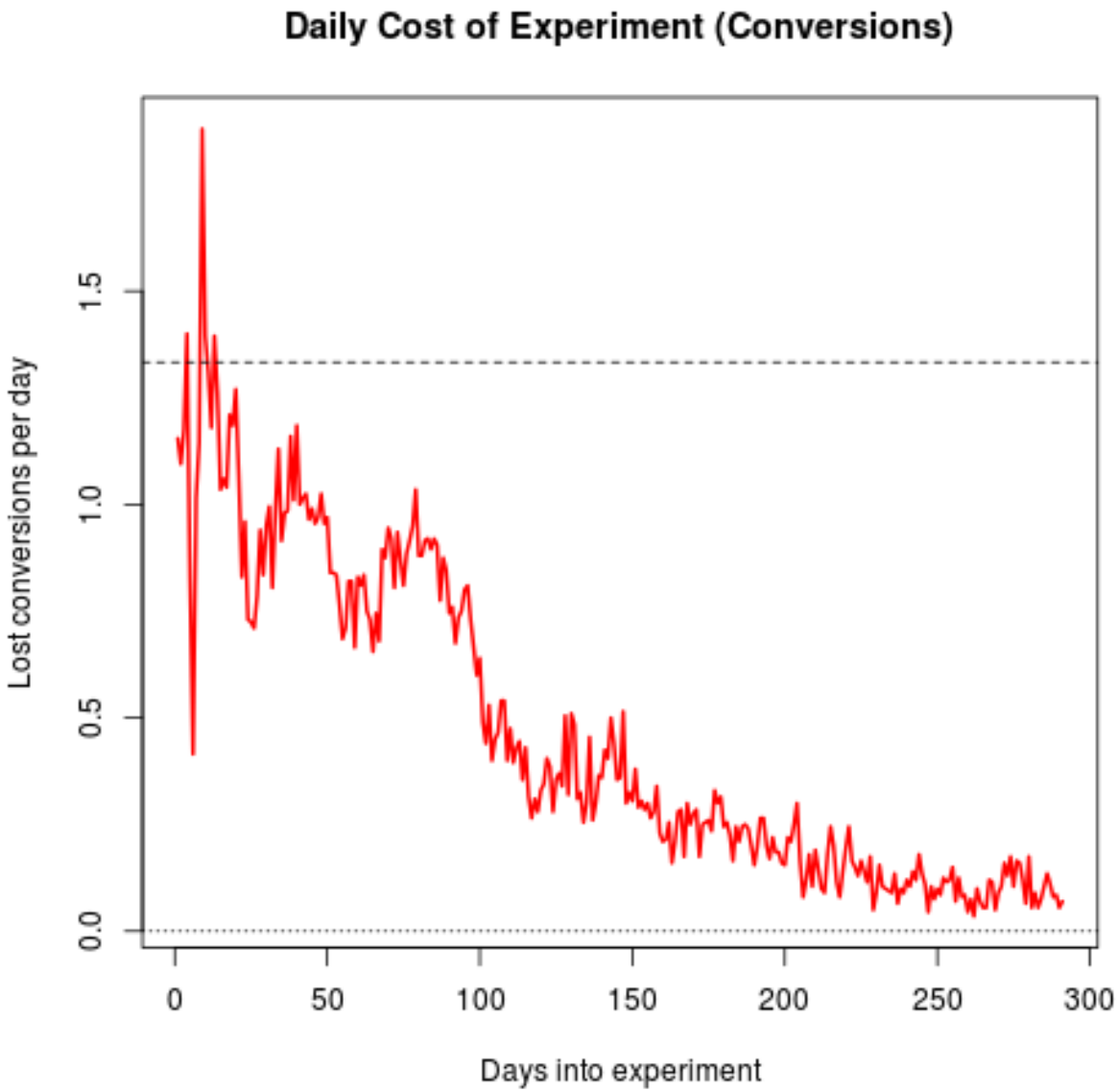
Huge savings vs equal allocation

Partly due to ending early, and partly due to lower cost per day.





Daily cost diminishes as inferior arms are downweighted

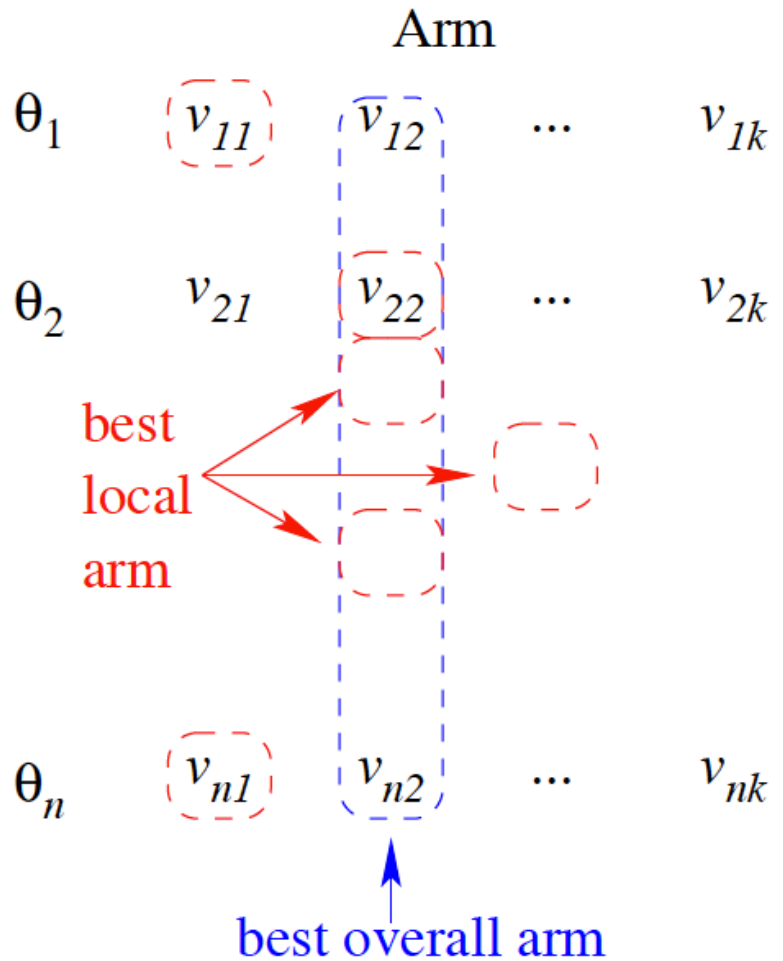


Outline

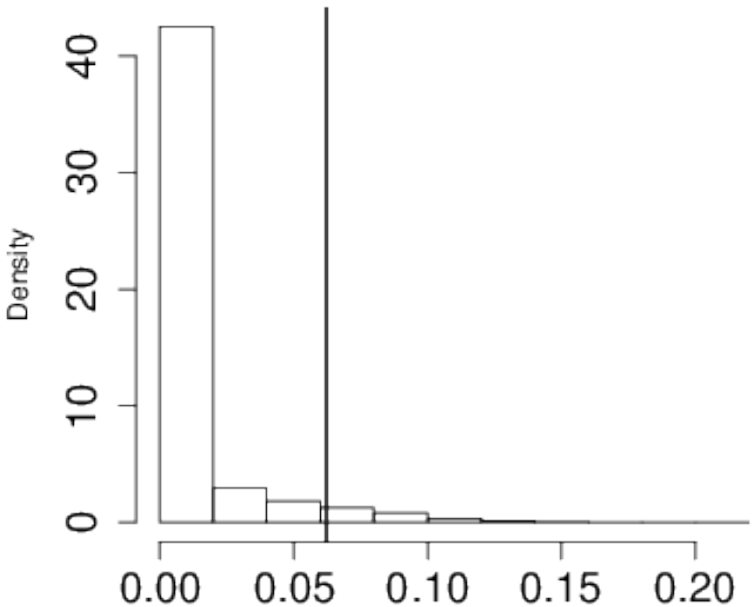
Value remaining

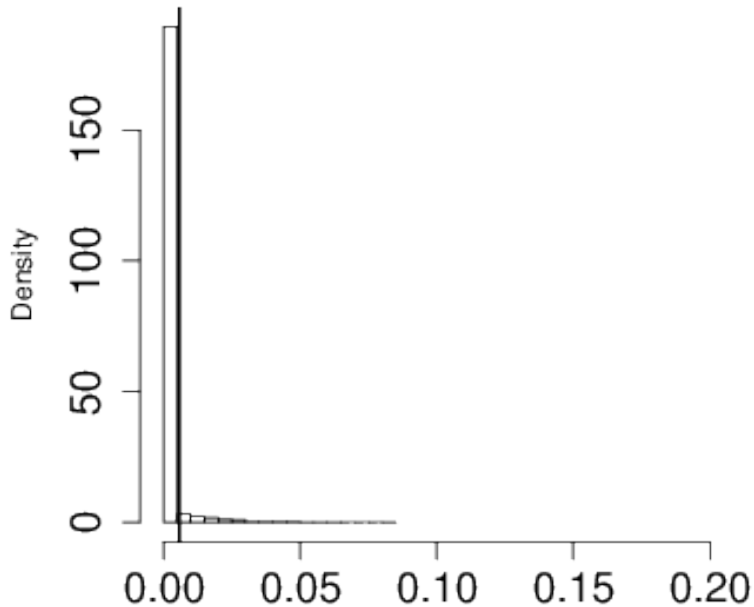
A better name than "per-play regret."

- Industrial experiments happen because people want to improve something.
- You can stop experimenting when you've squeezed all the value out of the experiment.
- Let v_{i*} = value in draw i of best overall arm.
- Let v_i^* = value of best arm in draw i.
- $v_i^* - v_{i*}$ is a draw of the value remaining in the experiment.



"Potential value remaining" shrinks as n grows.





successes	30	5	20
trials	100	50	80
w_{at}	0.76	0.00	0.24

successes	120	20	80
trials	400	200	320
w_{at}	0.93	0.00	0.07

Google

Features of PVR

- Nice
  - If a subset of arms tie, the experiment can still end.
    This can easily happen in multi-factor experiments.
  - If any single optimal arm probability exceeds .95 then PVR $= 0$.
    (Assuming the .95 quantile)
  - Units are meaningful to the experimenter.
    Allows the experiment to end based on "practical significance."

- PVR is not a hypothesis test
  - PVR makes no attempt to control "type I error rate."
  - If two (or more) arms tie, then no attempt is made to favor the original.
  - If switching costs are relevant, they need to be baked in.

Google

Outline

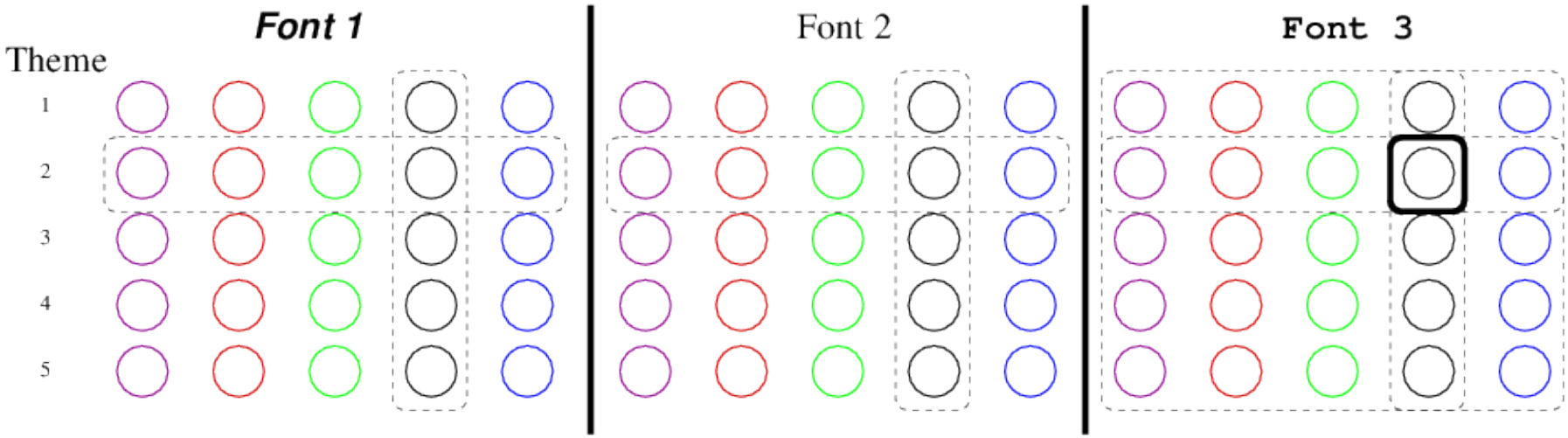
Bringing in good ideas from DOE

- Experiments in the real world involve multiple factors.
- The "1-way" layout is a bad idea because the number of "arms" explodes with multiple experimental factors.
- Classic DOE uses fractional factorial experiments to control the combinatorial explosion.
- The fractional factorial idea remains really powerful in a sequential world.

Google

The power of fractional factorial experiments

A single observation provides partial information on several configurations



- 3 fonts $\times$ 5 themes $\times$ 5 colors $=$ 75 configurations
- A single run of configuration (Font=3, Color=4, Theme=2) ... gives partial information on all configurations with
  - Color=4, or
  - Theme=2, or
  - Font=3
- Each observation teaches us *something* about 43 configurations.

Google

Fractional factorial experiments

The classic way of doing a fractional factorial experiment:

- Imagine that you will be analyzing the results of your experiment with a linear regression model.
- Choose a set of interactions that you're willing to live without.
- Allocate observations so that the remaining coefficients are "estimated as accurately as possible."
  - X-optimal design, with $X = D, G, A$, etc.
  - Heavily dependent on Gaussian linear models (so that mean and variance are distinct, and variance only depends on X.).
- Usually restrict to a small subset of potential design point (rows in the design matrix).

Fractional factorial bandits

How to do it with bandits:

Thompson sampling with logistic regression.

- This is really "full factorial" randomization but "fractional factorial modeling."
- Easy enough to fractionate the design matrix if there is reason to do so (e.g. eliminate problematic rows).

How to take advantage of "free" replications

Regression model with dummy variables

Int	F_2	F_3	C_2	C_3	C_4	C_5	T_2	T_3	T_4	T_5
1	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	1	0	1	0	0	0	1	0	0	0
1	0	0	1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0	0	0	0
1	0	1	0	1	0	0	0	0	0	0
1	0	1	0	1	0	0	0	1	0	0
1	0	0	0	1	0	0	0	1	0	0
1	0	1	0	0	1	0	0	0	0	0
1	0	1	0	0	0	0	0	0	1	0
1	0	0	0	0	1	0	0	0	1	0
1	0	0	0	0	1	0	0	0	0	0
					⋮					

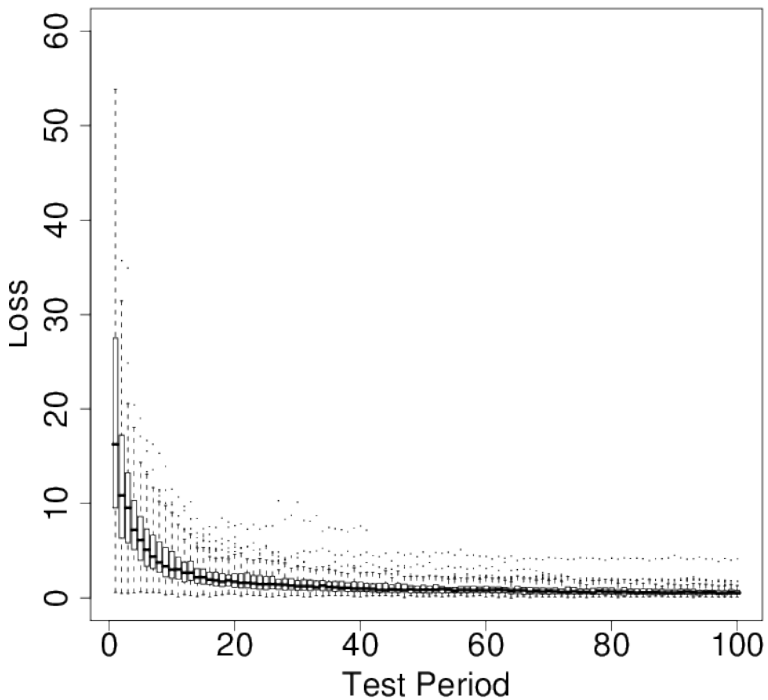
(75 rows, 11 columns)

Interactions

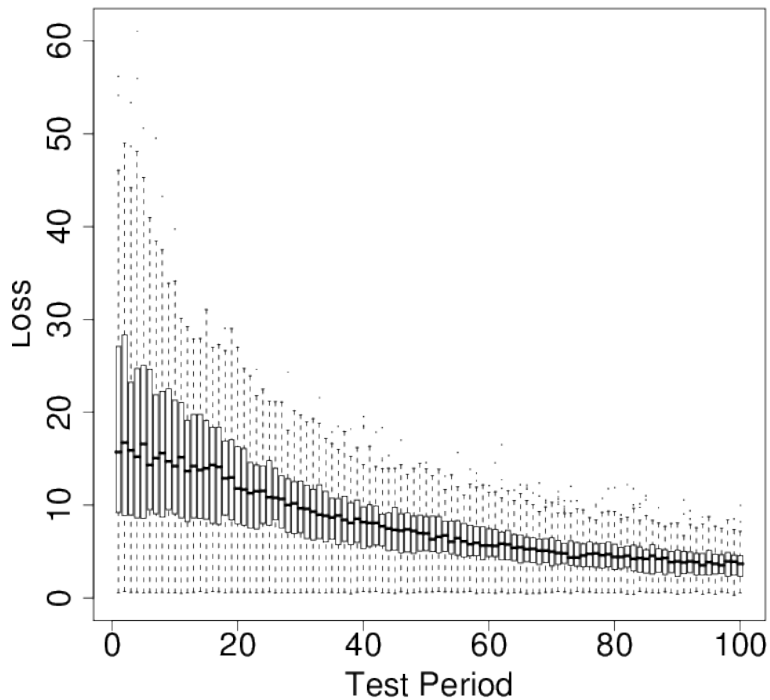
Interaction	# coef	Interpretation
Font:Theme	8	**Helvetica bold** has a greater impact with Theme 1 than Theme 2.
Font:Color	8	Orange increases CTR, but only with *Palatino italic*
Color:Theme	16	⋮
Font:Color:Theme	32	The incremental benefit of Orange with *Palatino italic* is muted in Theme 3, but magnified in Theme 4
Main Effects	2 + 4 + 4	
Intercept	1	
Total	75	

High order interactions tend to be **Small**, **Noisy**, and **Confusing**.

The FF bandit has many fewer parameters to learn, so it learns much faster, and wastes fewer impressions.



Fractional factorial (probit) bandit

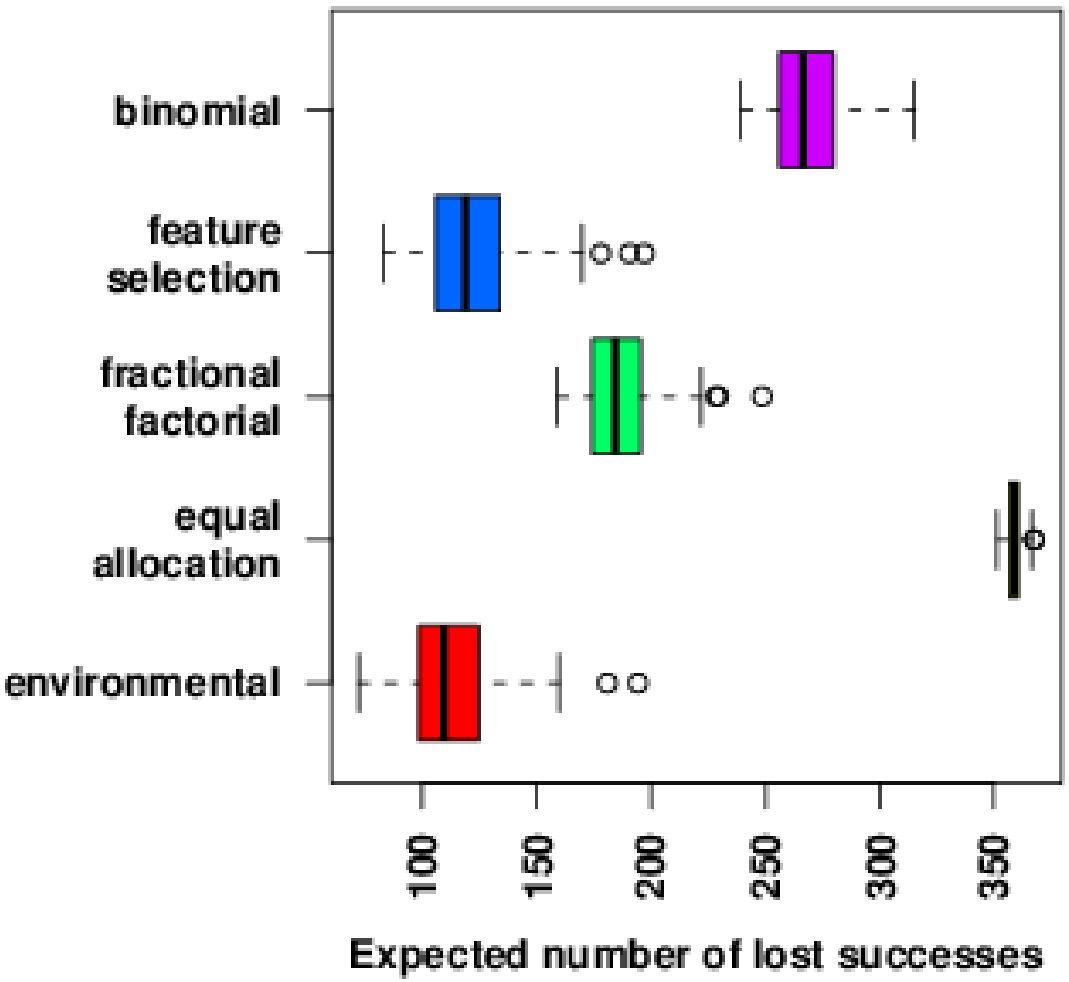


Binomial bandit

100 impressions per update cycle.

Controlling for context

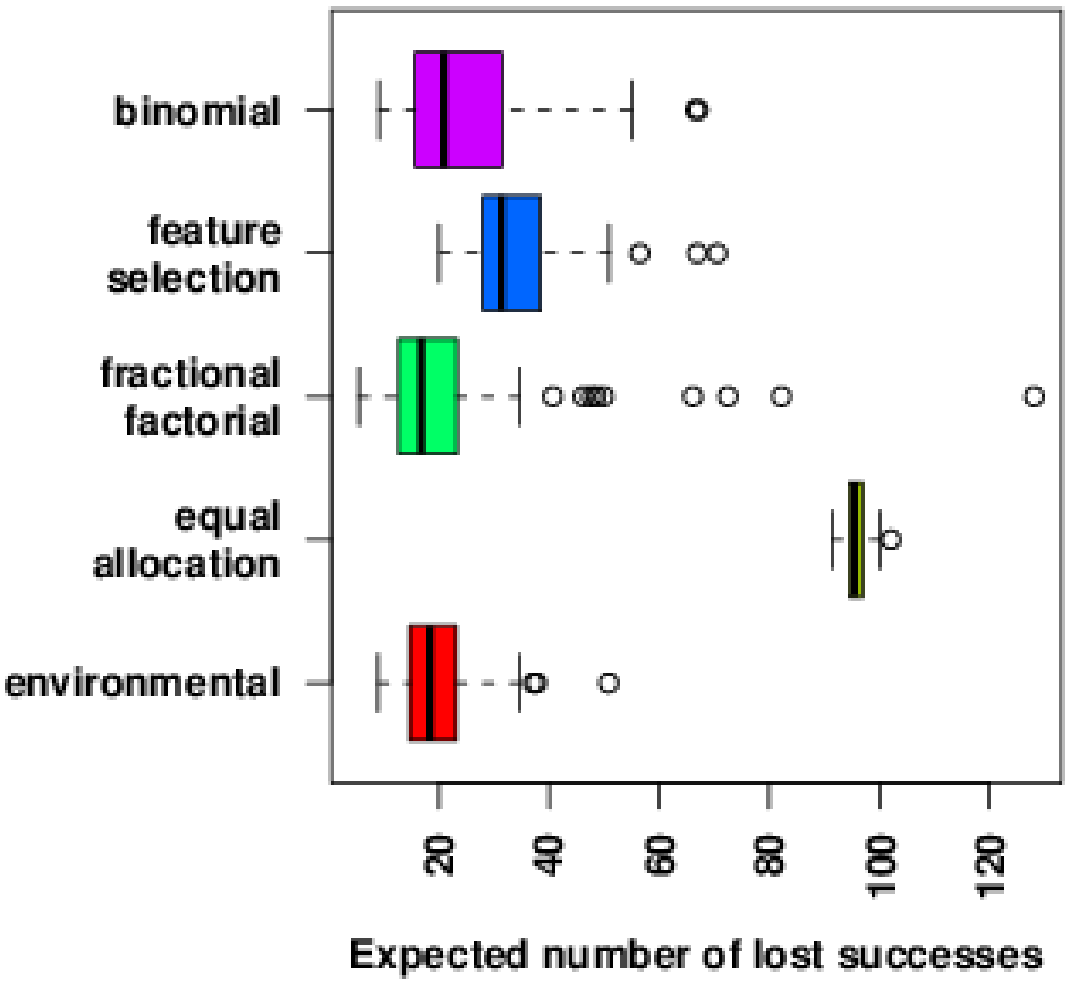
In a simulation with important interactions between experimental and contextual factors



Interactions with environment confound binomial and FF bandits.

Controlling for context

In a simulation with an important environmental effect, but no interactions



Lower variance for environmental bandit.

Conclusion

- The economics of the modern service economy are different than those of agriculture and industry.
- The world improves rapidly by experimenting with everything all the time, but experiments have to be cost effective.
- The cost of experiments can be reduced by
  - Fractional factorial designs that test several factors at once.
  - Sequential methods that down-weight clearly underperforming arms.
- Thompson sampling is an effective, robust method of running a sequential experiment that lets you focus on getting the model right.

References I

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002).
Finite-time analysis of the multiarmed bandit problem.
Machine Learning **47**, 235–256.

Bubeck, S. and Liu, C.-Y. (2013a).
A note on the Bayesian regret of Thompson sampling with an arbitrary prior.
arXiv preprint arXiv:1304.5758.

Bubeck, S. and Liu, C.-Y. (2013b).
Prior-free and prior-dependent regret bounds for Thompson sampling.
In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, eds., *Advances in Neural Information Processing Systems 26*, 638–646. Curran Associates, Inc.

Chapelle, O. and Li, L. (2011).
An empirical evaluation of Thompson sampling.
In *Neural Information Processing Systems (NIPS)*.

Kaufmann, E., Korda, N., and Munos, R. (2012).
Thompson sampling: An asymptotically optimal finite-time analysis.
In *Algorithmic Learning Theory*, 199–213. Springer.

Lai, T.-L. and Robbins, H. (1985).
Asymptotically efficient adaptive allocation rules.
Advances in Applied Mathematics **6**, 4–22.

May, B. C., Korda, N., Lee, A., and Leslie, D. S. (2012).
Optimistic bayesian sampling in contextual-bandit problems.
The Journal of Machine Learning Research **13**, 2069–2106.

Google

References II

Russo, D. and Van Roy, B. (2014).
Learning to optimize via posterior sampling.
Mathematics of Operations Research .

Scott, S. L. (2010).
A modern Bayesian look at the multi-armed bandit.
Applied Stochastic Models in Business and Industry **26**, 639–658.
(with discussion).

Sutton, R. S. and Barto, A. G. (1998).
Reinforcement Learning: an introduction.
MIT Press.

Thompson, W. R. (1933).
On the likelihood that one unknown probability exceeds another in view of the evidence of two samples.
Biometrika **25**, 285–294.