5: Big-Data Bayesian Data Science

David Draper

Department of Applied Mathematics and Statistics
University of California, Santa Cruz

Short Course (Day 5)
University of Reading (UK)

© David Draper (all rights reserved)
1 B = 1 byte = 8 binary digits = 2^0 bytes
1 KB (kilobyte) = 1,000 bytes = 2^10 bytes
1 MB (megabyte) = 1,000 KB = 2^20 bytes
1 GB (gigabyte) = 1,000 MB = 2^30 bytes
1 TB (terabyte) = 1,000 GB = 2^40 bytes
1 PB (petabyte) = 1,000 TB = 2^50 bytes
1 EB (exabyte) = 1,000 PB = 2^60 bytes
1 ZB (zettabyte) = 1,000 EB = 2^70 bytes
1 YB (yottabyte) = 1,000 ZB = 2^80 bytes

? 1 humongobyte? 1 gynormobyte? 1 too-f**cking-big-byte?
1 you-gotta-be-kidding-byte? 1 i-have-a-headache-byte?
1 just-kill-me-byte? 1 will-somebody-turn-out-the-lights-
when-they-close-the-door-byte?
A Brief History of “Big Data”

- **(1944)** It’s estimated that American libraries will double in size every 16 years; therefore the Yale University Library in 2040 will have approximately **200,000,000** books, occupying almost 10km of shelving and requiring 6,000 catalog employees.

- **(1961)** A scientist concludes that the number of new academic journals is growing **exponentially** (not linearly), doubling every 15 years.

- **(1986)** It’s estimated that the recording density achieved by Gutenberg (1450) was 500 bytes per cubic inch, 500 times the density of Sumerian clay tablets (4,000 BC); prediction: by 2000, RAM should be storing **1.25 \cdot 10^{11}** bytes per cubic inch.

- **(1997)** The term “**Big Data**” is used in an academic article for the first time; a different article uses the word **petabytes** (1,000,000 Gbytes) for the first time, estimating that the entire world contains a few hundred petabytes worth of information; therefore by **2000** (a) with tape and disk production there will **never** be a future need to throw any information away, and (b) a typical piece of information will **never** be looked at by a human being.
History of “Big Data” (continued)

- **(1998)** The growth rate of traffic on the Internet is estimated at about **100% per year**; at that rate, data traffic will overtake voice traffic around 2002.

- **(1999)** An influential CACM article has a section called **Scientific Visualization of Big Data**:

> “Where megabyte data sets were once considered large, we now find data sets from individual simulations in the **300 GB** range. ... But it is just plain difficult to look at all the numbers.” **Hamming**: “The purpose of computing is insight, not numbers; with Big Data we’re in danger of failing to achieve that purpose.”

- **(2000)** A study found that in 1999 the world produced about **1.5 exabytes** (1,000,000,000,000 GB) of data, about 250 MB for every human on the planet; by 2003 the volume had increased to **5 exabytes/year**, 92% of it stored in disks.

- **(2001)** The defining dimensions of Big Data are identified as the 3Vs: **volume**, **velocity** and **variety**.
(2007) Now the estimate is that in 2006 the world created 161 exabytes of data; between 2006 and 2010 this increased six-fold, to 988 exabytes/year, doubling every 18 months; as of 2012 we were up to 2.8 zettabytes (1 trillion GB) of data generated/year worldwide.

(2008) It was estimated that internet protocol (IP) traffic will reach 0.5 zettabytes/year in 2012 (this prediction was correct), an eightfold increase in 5 years.

(2009) A study finds that in 2008 Americans consumed information for about 1.3 trillion hours, an average of 12 hours/day/person; consumption totaled 3.6 zettabytes (11 trillion words), averaging out to 100,000 words and 34 GB per person per day; this means that you were exposed to about 100 words/minute of your 16 waking hours per day.

(2011) It’s estimated that the world’s information storage capacity grew at a compound annual rate of 25%/year between 1986 and 2007; moreover, in 1986, 99% of storage was analog, but in 2007 94% was digital.
History of “Big Data” (continued)

- **2011** A study finds that (a) in 2005 people in the U.S. had 1,000 minutes of curated Internet and video content for every minute available for consumption and (b) the world produced 14.7 exabytes of new information in 2008, triple the volume in 2003.

- **2015** Experts predict a 4,300% increase in annual data generation by 2020.

**Big Data:** “A collection of data sets so large and complex that it becomes difficult to process using hands-on database management tools or traditional data processing applications.”

**Big Data:** “A cultural, technological, and scholarly phenomenon that rests on the interplay of: (1) **Technology:** maximizing computation power and algorithmic accuracy to gather, analyze, link, and compare large data sets. (2) **Analysis:** drawing on large data sets to identify patterns in order to make economic, social, technical, and legal claims. (3) **Mythology:** the widespread belief that large data sets offer a higher form of intelligence and knowledge that can generate insights that were previously impossible, with the aura of truth, objectivity, and accuracy.”
Here’s a popular frequentist caricature of Bayesian analysis of large data sets:

All you Bayesians ever do is attach dubious priors to our likelihood functions, and with a lot of data this is at best ineffectual and at worst actually harmful: with so much data, your priors don’t matter (ineffectual), and with huge amounts of data, you can’t do your Bayesian calculations in a realistic amount of time (harmful).

So go back to analyzing your tiny little data sets in clinical trials and the social sciences, and leave the important big-data work to the frequentists.

As I’ll now show, this argument is partly wrong and partly right.

Unrelated remark: My case studies today are from private industry, in the field of eCommerce.

If you’re offended by the profit motive, OK, but this doesn’t let you off the “Big-Data” hook: there are many current examples of “Big Data” analyses that are non-industrial (e.g., data from Electronic Medical Records in medicine and health policy).
Some Sobering Truths About “Big Data”

- I’ve put “Big Data” in quotes because, if you think we have a lot of data now, try imagining what it will be like in 20 years.

- **Bayesian analysis** provides a wonderful approach to logically-internally-consistent quantitative conclusions; but our “exact” computational methods (e.g., MCMC) **DO NOT SCALE WELL** with increasing sample size \( n \) (in the absence of sufficient statistics) and/or increasing model complexity (e.g., number \( k \) of predictor variables).

- The machine-learning (ML) guys have methods that (a) produce an answer in tasks such as **point prediction** and (b) do so very quickly and in a way that **DOES** scale with \( n \) and \( k \); they don’t claim that their answer is “best,” but they give the clients an answer quickly; we Bayesians know how to give the clients the **OPTIMAL** answer (conditional on assumptions), but we make them wait days, weeks or months for the results; the clients rightly conclude that we have no practically useful answer for them, and they turn to ML for an answer.

- ML apocryphal anecdote.
For the past two decades the ML guys have been **STEALING** our Bayesian ideas, renaming them (e.g., Chinese restaurant and Indian buffet processes = Dirichlet process/Pólya urn schemes) and collecting higher consulting fees; they’re taking food off of your children’s dinner table; this has to stop.

Here’s how we get our food back:

- Companies such as Amazon (market capitalization US$205 billion, 30% annual growth rate) are filled with people who think that

  \[
  \text{Quantitative analysis} = \text{ML};
  \]

  we **MUST** push back against this vigorously and teach them that

  \[
  \text{Quantitative analysis} = \{\text{Econometrics, Statistics, ML, Optimization (OR), ...}\}. 
  \]

- The **ONLY** thing that mediocre ML guys are moderately good at is point prediction, with (barely) smaller RMSE than the next guy as success for them; we **MUST** emphasize that there is **FAR MORE** than this in good quantitative analysis.
Here’s how we get our food back (continued):

- Even the good ML guys only know how to do predictive inference by bootstrapping their point predictions; in general, ML guys have no idea how to do (a) parametric inference, (b) causal inference, (c) experimental design, (d) time series analysis, (e) decision theory, …; we MUST emphasize to management that whenever the real problem is something other than interval prediction, we are the ONLY guys who know how to even think about the problem correctly.

- BUT we must up our COMPUTING game, by developing fast and highly accurate approximate methods of Bayesian computation that SCALE well with \( n \) and \( k \); until we do this, all we can do is TALK about how we’re better than the ML guys.

The rest of the talk: (a) optimal Bayesian analysis of randomized controlled experiments with 12 million observations in EACH of the \( T \) and \( C \) groups; (b) Bayesian analysis of observational studies with 10 million participants; and (c) time series forecasting with 30 million outcome variables.
The randomized controlled trial (RCT, rebranded \textit{A/B testing} in \textit{eCommerce}) has a long and distinguished history in medicine and agriculture that dates back to the 1920s.

In the early days RCTs were used successfully in settings in which the noise-to-signal ratio, as measured by the between-subject standard deviation divided by the size of the effect the RCT was trying to find, was on the order of 1:1;

- sample sizes in the low hundreds in each of the treatment (T) and control (C) groups sufficed.

Today in \textit{eCommerce}, people face noise-to-signal-ratios of \textbf{100:1} or higher;

- I’ve seen a trial that would need \textbf{420 million} total experimental subjects to find a business-relevant effect with (5%/5%) false positive/negative error rates.
How should data from a large A/B test be optimally analyzed?

- Is optimal analysis possible?
  (Yes)

How can A/B tests be designed for greater efficiency, so that hundreds of millions of subjects are not needed?

A promising alternative to the usual static A/B test, in which sample sizes are fixed at design time, is dynamic, adaptive design and analysis of experiments, in which

- subjects are assigned to treatments sequentially to optimize expected information gain.

The idea is not new — it goes back at least to WR Thompson in 1933 — but it’s not yet been fully exploited in eCommerce, even at cutting-edge companies such as Google and Amazon.
Suppose **You** have an **idea** for **improving** the **Amazon web experience**.

You run an **A/B experiment** — in which **some visitors** (the treatment group (A or T)) get \{the current best Amazon web experience\} + \{your innovation\} and others (the control group (B or C)) get \{the current best Amazon web experience\} — over (say) a 3–week time period.

You choose **Gross Merchandise Volume Bought (GMB)** as the outcome (or response) variable \(y\) of principal interest to you.

(Other outcomes may also be relevant, including the number of Bought Items (BI) and a variable measuring whether a successful sale involved a New or Re-activated Buyer (NoRB).)

It turns out that all of the four basic statistical activities arise in analyzing the results from this A/B experiment:

- **(Description)** What was the mean value \(\bar{y}_T\) of GMB in the treatment group?

  How about \(\bar{y}_C\), the control mean?
The Four Basic Statistical Activities

How much bigger was $\bar{y}_T$ relative to $\bar{y}_C$, as measured (for example) by the lift $\hat{\theta} = \frac{\bar{y}_T - \bar{y}_C}{\bar{y}_C}$?

- **(Inference)** If the experiment were repeated for a much longer period of time than 3 weeks, how likely is it that the 3–week lift value might diminish or disappear, just because of random fluctuations in who shows up at amazon.com?

  In other words, was the apparent change in lift caused by the treatment intervention, or could it easily have been the result of haphazard fluctuations of unknown cause?

- **(Prediction)** If the treatment intervention were accepted as useful, how much extra GMB would result in (say) the second half of 2015?

- **(Decision)** Should this experimental intervention be adopted as part of the new current best Amazon web experience?

Of these four activities, only description involves no uncertainty: we’re not sure about the answers to the inferential, predictive and decision questions above, even after the experiment has been run.
Bayesian Analysis of $A/B$ Test Results

**Case Study:** experiment 5108, initial outcome variable raw GMB; visitors randomized to $T$ or $C$ and followed for 2 weeks.

The treatment intervention consisted of a change to the basic search engine that was supposed to increase the relevance of the search findings.

In this case study I use the freeware statistical analysis environment R.

The first thing to know about raw GMB, even in a multi-week trial, is that most of the data values at the visitor level (i.e., aggregated across 1 or more visits during the 2 weeks) are $0$:

<table>
<thead>
<tr>
<th>Group</th>
<th>n zeros</th>
<th>n nonzeros</th>
<th>n total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treatment</td>
<td>11,100,587</td>
<td>1,133,706</td>
<td>12,234,293</td>
</tr>
<tr>
<td>Control</td>
<td>11,096,065</td>
<td>1,135,435</td>
<td>12,231,500</td>
</tr>
</tbody>
</table>

Each group had about 12 million observations, of which about 90% were 0.
# analysis (1): read in the non-zero treatment data and
# look at it descriptively

setwd( "C:/e-Bay/Lift" )

nonzero.treatment.values <-
  scan( "pgmb-raw-5108-treatment-v1.txt" )

# Read 1133706 items

print( n.nonzero.treatment.values <-
  length( nonzero.treatment.values ) )

# 1133706

nonzero.treatment.values <- sort( nonzero.treatment.values )

print( mean.nonzero.treatment.values <-
  mean( nonzero.treatment.values ) )

# 98.50182
n.zero.treatment.values <- 11100587

print( n.treatment.total <- n.nonzero.treatment.values +
      n.zero.treatment.values )

# 12234293

print( overall.treatment.mean <- ( n.zero.treatment.values * 0 +
      n.nonzero.treatment.values * mean.nonzero.treatment.values ) /
      n.treatment.total )

# 9.127794

So the **mean raw GMB value** in the **treatment group** was **$9.13**.

hist( nonzero.treatment.values, breaks = 100000, probability = T,
      main = '', xlab = 'Nonzero Treatment Values',
      ylab = 'Density' )

The **histogram** on the **next page** offers a way to get information
about the **distributional shape** of the **nonzero raw GMB variable**.
Nonzero treatment raw GMB had an enormously heavy right-hand tail: most of the values were near $0$, but the largest value was $91,417$. 
The most-studied distributional shape is that of the normal, or Gaussian, distribution.

Its skewness (degree of asymmetry) and kurtosis (heaviness of tails) values are both 0.

By contrast, the nonzero treatment raw GMB variable had skewness and kurtosis values of +51.0 and +6,017, respectively.
With **heavily positively skewed variables** that **don’t take on negative values**, it’s **typically helpful** to **look at the histogram** of the **logarithm** of the **variable**.
Here we see an interesting behavior that looks like porcupine quills: individual values along the number line with much higher frequency than that of their neighbors.
The porcupine quills are evidently psychological price-points: people would vastly rather transact at $0.99 and $1.99 than at $1 and $2.

How about the control raw GMB variable?

```
print( mean.nonzero.control.values <-
       mean( nonzero.control.values ) )
```

# 99.14066
The control mean raw GMB value was $9.20, versus $9.13 in treatment.

In this experiment the treatment was actually a bit worse than the control using the raw GMB outcome, by about 82 basis points (0.82%).

The control histograms were similar to those in treatment, but the largest value in the control group was $161,572, leading to skewness and kurtosis values of +106 and +26,661, respectively.
It’s helpful to report lift in two parts: the change in the percentage of 0 values and the change in the nonzero mean.

```r
print( treatment.percent.zero <- n.zero.treatment.values / n.treatment.total )
# 0.9073338

print( control.percent.zero <- n.zero.control.values / n.control.total )
# 0.9071712
```

Percent non-zero overall

<table>
<thead>
<tr>
<th>Group</th>
<th>Zeros</th>
<th>Non-Zero Mean</th>
<th>Overall Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treatment</td>
<td>90.73</td>
<td>$98.50</td>
<td>$9.128</td>
</tr>
<tr>
<td>Control</td>
<td>90.72</td>
<td>$99.14</td>
<td>$9.203</td>
</tr>
</tbody>
</table>

The treatment had 0.02% more zeros than control; the treatment nonzero mean was 0.64% lower than the control nonzero mean; the net result was a lift of −0.82%.
So in this experiment the treatment was worse both ways on raw GMB: it had a slightly higher percentage of zero transactions, and it also had a somewhat lower mean for the nonzero transactions.

One way to build a full stochastic model — for the raw GMB variable, one group at a time — would be to break it up into three parts, or mixture components (this is Bayesian hierarchical/mixture modeling):

- The spike at $0$, which is modeled with the Bernoulli (0/1) distribution with unknown probability $p_0$ of being $0$;

- The psychological price-points, which can be handled with the multinomial distribution with known locations on the $\$\ scale and unknown probabilities $(p_1, \ldots, p_k)$; and

- an appropriate continuous distribution for what’s left over.

The histograms on the next page (for the treatment group) show that what’s left over looks roughly Gaussian on the log scale:
However, careful analysis demonstrates that a mixture of 2 Gaussian distributions on the log scale
(a) fits a lot better than a single Gaussian and
(b) fits the treatment data well (blue dotted curve = 1 Gaussian; red solid curve = mixture of 2 Gaussians):
It looks like we have a good Bayesian (parametric) probability model for the raw GMB data: a mixture of Bernoulli for the spike at 0, multinomial for the psychological price-points, and a mixture of 2 Gaussians on the log scale for what’s left; however,

- Exact Bayesian computations with 12 million observations in each of the treatment and control groups in this model are infeasible, and

- raw GMB appears to have a serious defect, which seems to make it unacceptable as the basis for decisions on whether to launch promising-looking treatment interventions.

There was a hint of this earlier, when I mentioned that the largest treatment value was $91,417 and the largest control value was $161,572 (77% larger than the corresponding treatment maximum).

The basic apparent problem is that

Raw GMB can be extremely sensitive to a small number of very large observations that, arguably, were not causally influenced by offering or withholding the treatment intervention.
Shifting only the 3 largest $C$ values to $T$ drives the estimated lift from $–0.8\%$ to $0\%$, and shifting only the largest 14 observations (out of 12,231,500) from $C$ to $T$ is enough to move the posterior probability that $T$ is better than $C$ from $0.1$ to over $0.95$. 
The sensitivity illustrated on the previous page has led some experimenters to recommend an analysis method that is sometimes called capping (the technical statistical term is Winsorizing) for outcomes such as GMB:

- In \( C \), find the \( 100(1 - \epsilon) \)th GMB percentile, for a value of \( \epsilon \) such as 0.0001 (this number depends on \( n_C \)); call the resulting GMB value \( y_{C,1-\epsilon} \);

- Replace all GMB values in \( C \) that are \( > y_{C,1-\epsilon} \) with \( y_{C,1-\epsilon} \); now define \( \bar{y}_{C,\text{Winsorized}} = \text{the mean of the resulting modified } C \text{ data set} \);

- In \( T \), find the \( 100(1 - \epsilon) \)th GMB percentile, for a value of \( \epsilon \) such as 0.0001 (this number depends on \( n_T \) and is generally chosen to be the same as the \( \epsilon \) in \( C \)); call the resulting GMB value \( y_{T,1-\epsilon} \);

- Replace all GMB values in \( T \) that are \( > y_{T,1-\epsilon} \) with \( y_{T,1-\epsilon} \); now define \( \bar{y}_{T,\text{Winsorized}} = \text{the mean of the resulting modified } T \text{ data set} \);

- Compute \( \hat{\theta}_{\text{Winsorized}} = \frac{\bar{y}_{T,\text{Winsorized}} - \bar{y}_{C,\text{Winsorized}}}{\bar{y}_{C,\text{Winsorized}}} \), and base decisions on \( \hat{\theta}_{\text{Winsorized}} \) instead of on \( \hat{\theta} = \frac{\bar{y}_{T} - \bar{y}_{C}}{\bar{y}_{C}} \).
Capping Is a Bad Idea

This is current “best” practice in some eCommerce companies, but it turns out to be a bad idea, and should immediately be stopped and replaced by $\hat{\theta}$.

Capping has sometimes been justified on the ground of diminished root mean squared error ($RMSE_{RS}$, in repeated sampling) of $\hat{\theta}_{\text{Winsorized}}$ as an estimate of $\theta$, when compared with $\hat{\theta}$, and it’s true that $\hat{\theta}_{\text{Winsorized}}$ does indeed have lower RMSE than $\hat{\theta}$ for this task; however, in general, when $\hat{\gamma}$ is used to estimate $\gamma$,

$$RMSE_{RS}(\hat{\gamma}) = \sqrt{[b_{RS}(\hat{\gamma})]^2 + [SE_{RS}(\hat{\gamma})]^2}, \quad (1)$$

in which $b_{RS}(\hat{\gamma}) = [E_{RS}(\hat{\gamma}) - \gamma]$ is the (repeated-sampling) bias of $\hat{\gamma}$ and $SE_{RS}(\hat{\gamma})$ is its (repeated-sampling) standard error.

$RMSE$ is an acceptable (frequentist) criterion to use when choosing among estimators, but only when the bias of the $RMSE$-minimizing estimator is low; otherwise (in A/B experimentation) You get a distorted view of lift.
I’ll now give an example in which $\hat{\theta}_{\text{Winsorized}}$ is biased by more than $-82\%$, making it completely unacceptable as the basis of good decision-making.

**Case Study:** Treatment $T$: a marketing email in the Business and Industrial category; Control $C$: no such email.

**Design:** controlled trial (A/B test), with 256,721 representative users randomized (128,349 to $T$, 128,372 to $C$) and followed for 7 days, starting on 14 July 2014.

**Outcome of interest:** Gross Merchandise Bought (GMB), i.e., total $\$$ generated by buyers in the 7-day period.

\[
\theta = \text{lift} \left( \frac{\mu_T - \mu_C}{\mu_C} \right) \text{ that would be observed if all users in the population } \mathcal{P} \text{ of interest (future users) were to counterfactually either receive the email (resulting population mean } = \mu_T \text{) or not receive it (population mean } = \mu_C \text{): here } k = 1.
\]
Top panel: Inference with $\hat{\theta}$ is unbiased, and noisy because the sample sizes are small.

Bottom panel: Inference with $\hat{\theta}_{\text{Winsorized}}$ appears to be less noisy, because extreme observations have been Winsorized, but is enormously biased on the low side.
It’s Possible to Run Too Many “Small” Experiments

As an outcome variable, GMB is a nightmare: in typical experiments, its noise-to-signal ratio is on the order of 15 to 1 (this is exceptionally noisy, meaning that You’ll need a lot of data to find small-but-still-business-relevant improvements).

Typical 4–week sample sizes in treatment (T) and control (C) groups in some eCommerce companies are about 12 million.

<table>
<thead>
<tr>
<th>false positive rate</th>
<th>false negative rate</th>
<th>true lift</th>
<th>sample size required in each of T and C groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>5%</td>
<td>20%</td>
<td>1.0%</td>
<td>12,365,114</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>1.0</td>
<td>12,174,946</td>
</tr>
<tr>
<td>15</td>
<td>50</td>
<td>0.5</td>
<td>19,334,495</td>
</tr>
<tr>
<td>10</td>
<td>20</td>
<td>1.0</td>
<td>20,285,382</td>
</tr>
<tr>
<td>10</td>
<td>50</td>
<td>0.5</td>
<td>29,562,739</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>1.0</td>
<td>29,562,739</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>1.0</td>
<td>38,537,313</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>0.5</td>
<td>48,699,782</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>0.5</td>
<td>154,149,252</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>0.5</td>
<td>221,307,004</td>
</tr>
</tbody>
</table>
Either Option [I] or Option [II]

Along many dimensions that many eCommerce companies want to explore experimentally, most of the big incremental, at-the-margin lifts ($\geq 1\%$) have already been found (this doesn't preclude finding future big lifts from bold, non-incremental, big-think treatment interventions).

Conclusion from the table on the previous page:

- With GMB as the outcome of principal interest, to find 0.5% lifts with decently low false positive and false negative rates, current sample sizes in each of $T$ and $C$ would have to go up multiplicatively by a factor of 15 or more.

The total number of users available for experimentation per year is essentially fixed; if you currently perform $N$ too-small experiments per year, this would mean performing $\frac{N}{15}$ right-sized experiments in the future.

There are two main static-design options.

Option [I]: Run fewer, larger, better-designed experiments; or

Option [II]: Concentrate on performance variables that have lower noise-to-signal ratios.
Option II: Concentrate on performance variables that have lower noise-to-signal ratios.

These will typically be process items (such as number of clicks in “desirable” places inside your web-page tree) rather than outcomes (such as GMB or Bought Items (BI)).

For this to work, you need to pick process items that are strongly related to (correlated with) your desired outcomes (if the T intervention drives more red-haired people to your site, this will probably not make GMB go up).

Advantage of focusing on process items with good process-outcome links:

- Most users give you process information more often than outcome information (e.g., many days, most users don’t buy anything at all, but many users click around inside your web-page tree on many days).

Disadvantage of focusing on such process items:

- The lower the correlation between a process item and your desired outcomes, the less relevant any process improvements you find will be.
(Static-Design) Recommendation: Run some experiments under each of Options I and II.

The founder of statistical experimental design was Sir Ronald Aylmer Fisher FRS OBE (1890–1962), who did his initial pioneering work on design in 1926.

“No aphorism is more frequently repeated ... than that we must ask Nature few questions, or ideally, one question at a time. The writer is convinced that this view is wholly mistaken. Nature, he suggests, will best respond to a logically and carefully thought out questionnaire; indeed if we ask her a single question, she will often refuse to answer until some other topic has been discussed.” (Fisher, 1935)

It’s unwise to ignore Fisher’s advice on the topic of experimental design.

And yet that’s exactly what many eCommerce companies are doing now:
Another Recommendation

People run each experiment in a vacuum with respect to all other experiments, which is like asking Nature only a single question at a time.

Not only is this one-at-a-time approach inefficient (you obtain less information per experimented-upon user than you should);

It’s also flatly wrong in situations in which the combined effect of two $T$ interventions is anything other than the sum of their separate effects (this is the problem of interactions, and it’s currently being completely ignored at some companies).

Another recommendation:

The right way to “ask Nature a questionnaire” is with experiments in which multiple factors are varied simultaneously; everybody should do this.

For example,

- factor $A = \{10$ different possible improvements to your current best search engine$\}$;
- factor $B = \{6$ different versions of a discount offer$\}$; and
- factor $C = \{3$ different ways to re-structure the Amazon server farm to try to deliver web pages faster$\}$.
Interactions; Fractional-Factorial Designs

Note that factors $A$, $B$ and $C$ explore rather independent (orthogonal) directions in improvement space, but may still interact with each other in their effects on Amazon users.

As part of finding the very best combination of treatments, you need to be able to estimate the sizes of those interactions.

$(10 \cdot 6 \cdot 3)$ is 180 different $T$ groups simultaneously, if all possibilities need to be explored.

That’s a lot of $T$ groups, but fractional-factorial experimental design technology (which permits you to experiment with a well-chosen subset of the 180 groups and still get the information You want — has been available since the 1950s to help structure these designs efficiently, and the analysis of variance (which is a good method to analyze fractional-factorial designs) has been around since the 1920s to help analyze the results from these experiments.

The one-at-a-time approach was best practice around the year 1915.

It would be good for eCommerce to at least get up to 1950s-era speed in experimentation.
Suppose you have dozens or hundreds of variations on a basic theme (e.g., search-engine strategies, or the 180 different groups above). You want to know which ones are best, and you don’t care that much about full causal understanding of why they’re best.

Then you can use an experimentation method called sequential multi-arm bandit,

in which (e.g.)

(a) you get a little bit of information about how well variations 1 to $k_1$ do (e.g., $k_1 = 10$), and you immediately drop the worst $k_2$ of them (e.g., $k_2 = 5$);

(b) now get a little bit of information about how well variations ($k_1 + 1$) to $2k_1$ do, and again drop the worst $k_2$ of the ones you’ve looked at so far;

(c) repeat (b), except that from time to time at random you bring back some of the rejected variations to see if your rejection of them was hasty.

Recommendation: eCommerce should do more of this.
The sequential multi-arm bandit idea dates back to 1933, but improvements to the basic plan are still at the research forefront. Other versions of Bayesian sequential optimal experimental design exist, and should also be tried (e.g., the Google approach): search on google analytics multi-armed bandits and go to the top page to see how Google currently does this (there are better sequential designs than theirs: use Bayesian decision theory).

- Some eCommerce companies have a long-standing problem: they don’t really know who some of their users are until they buy something.
  - This cripples your ability to do sensible longitudinal data analysis:
    - To stratify on important variables at design time, to improve accuracy of $A/B$ tests; and
    - To move eCommerce into the era of “personalized medicine,” in which users get targeted treatments that are known to work in the recent past on similar users.
Q: From an information-processing point of view, can (static) A/B tests be analyzed optimally, even with sample sizes in the tens of millions?

A: Yes.

To see how, first look at a tiny case study, then go big.

**Case Study (1970s Version):** Captopril, a new type of anti-hypertension drug, was developed in the mid-1970s.

- Nothing was known about captopril’s effects prior to the first experiment on it (MacGregor et al., 1979; I’ve changed a few of the details for ease of exposition): 24 representative hypertensive people, randomized (12 to C [placebo], 12 to T [captopril]; SD = standard deviation; outcome variable = systolic blood pressure [mmHg] at the end of the trial).

<table>
<thead>
<tr>
<th>group</th>
<th>sample size</th>
<th>sample mean</th>
<th>sample SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>12</td>
<td>185.3</td>
<td>17.1</td>
</tr>
<tr>
<td>T</td>
<td>12</td>
<td>166.8</td>
<td>14.9</td>
</tr>
</tbody>
</table>
Summary: sample sizes \((n_C, n_T) = (12, 12)\); sample means \((\bar{y}_C, \bar{y}_T) = (185.3, 166.8)\); sample SDs \((s_C, s_T) = (17.1, 14.9)\).

Intuitive estimated lift \(\hat{\theta} = \frac{\bar{y}_T - \bar{y}_C}{\bar{y}_C} = \frac{166.8 - 185.3}{185.3} \approx -0.0998 = -10.0\%\).

We estimate that captopril causes a 10% reduction in systolic blood pressure (sounds like a big win), but how much uncertainty is associated with this estimate, in generalizing inferentially from the patients in the experiment to \(P = \{\text{all hypertensive patients}\}\)?

We need to finish the model specification to answer this question.

- \(p(\theta|B)\) — the “prior” distribution for \(\theta\) (given \(B\)).

Since nothing was known about captopril prior to this experiment, the external-information distribution should contain essentially no information.

In other words, from an entropy point of view it should be close to uniform, so take \(p(\theta|B) \propto 1\) (this is a diffuse or flat prior).
• \( p(D|\theta \mathcal{B}) \) — the “sampling” distribution for \( D \) given \( \theta \) and \( \mathcal{B} \):

Off-the-shelf specification for this is as follows — let \( \{y_{iC}\}_{i=1}^{nc} \) and \( \{y_{jT}\}_{j=1}^{nt} \) be the \( C \) and \( T \) outcome values, respectively; then

\[
(y_{iC}|\mu_C, \sigma_C^2 \mathcal{B} \mathcal{G}) \overset{\text{IID}}{\sim} N(\mu_C, \sigma_C^2)
\]

\[
(y_{jT}|\mu_T, \sigma_T^2 \mathcal{B} \mathcal{G}) \overset{\text{IID}}{\sim} N(\mu_T, \sigma_T^2),
\]

in which \( \mathcal{G} = \) assumption of Gaussian sampling distributions in \( C \) and \( T \).

Fact: With this sampling distribution, the induced likelihood distribution for \( \theta \) is

\[
\ell(\theta|D \mathcal{B} \mathcal{G}) \overset{\cdot}{=} \text{Normal with mean } \hat{\theta} \text{ and SD } \\
\sqrt{\frac{\bar{y}_T^2 s_C^2}{\bar{y}_C^4 n_C} + \frac{s_T^2}{\bar{y}_C^2 n_T}},
\]

and, with the prior distribution \( p(\theta|\mathcal{B}) \propto 1 \), the resulting posterior distribution is

\[
(\theta|D \mathcal{B} \mathcal{G}) \overset{\cdot}{=} N \left( \hat{\theta}, \sqrt{\frac{\bar{y}_T^2 s_C^2}{\bar{y}_C^4 n_C} + \frac{s_T^2}{\bar{y}_C^2 n_T}} \right) \overset{\cdot}{=} N(-0.0998, 0.0334^2).
\]
The signal-to-noise ratio here is \( \frac{|\text{posterior mean of } \theta|}{\text{posterior SD of } \theta} \approx \frac{0.0998}{0.0334} \approx 2.99 \), and the posterior probability \( p(\theta < 0 | D B G) \) that captopril would be beneficial, on average, if administered to the population of \{all hypertensive patients similar to those in this study\} — given the data set \( D \), the background information \( B \), and the Gaussian sampling-distribution assumption \( G \) — is about 0.999.
Of course we don’t want $p(\theta < 0|D\mathcal{B}\mathcal{G})$, because $\mathcal{G}$ is not part of the known-to-be-true background information $\mathcal{B}$; we want $p(\theta < 0|D\mathcal{B})$.

Definition (Draper, 2015): Given $(\theta, D, \mathcal{B})$ from $\mathcal{C} = (\text{problem context}, \text{data-gathering protocol}),$

a Bayesian model specification $[p(\theta|\mathcal{B}), p(D|\theta\mathcal{B})]$ is optimal if it includes only assumptions rendered true by the structure of $\mathcal{C}$.

Fact: One way to achieve optimal Bayesian model specification is via Bayesian non-parametric (BNP) methods, which place prior distributions on cumulative distribution functions (CDFs).

Fact: With little loss of generality, an optimal Bayesian model specification for $\{y_i\}_{i=1}^{n_c}$ and $\{y_j\}_{j=1}^{n_T}$ in the current Case Study involves Dirichlet-process (DP) priors, as follows:

$$
(F_C|\mathcal{B}) \sim DP(\alpha, F_{0C})
$$

$$(y_{iC}|F_C\mathcal{B}) \overset{\text{IID}}{\sim} F_C$$

(5)

and similarly for $\{y_{jT}\}_{j=1}^{n_T}$, where $F_C$ is the CDF of the outcome values in the population of (patients, users) similar to those in experiment.
Fact: With no information about \( F_C \) external to \( D \), the optimal BNP analysis is based on the DP posterior

\[
(F_C|D \mathcal{B}) \sim DP \left(n_C, \hat{F}_{n_C}\right),
\]

(6)

where \( \hat{F}_{n_C} \) is the empirical CDF based on \( \{y_i\}_{i=1}^{n_C} \).

Definition: Given a real-valued data set \( y = (y_1, \ldots, y_n) \), the (frequentist) bootstrap distribution of the sample mean \( \bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i \) may be approximated by

(a) choosing a sample of values \( y_i^* \) at random with replacement from the \( y \) vector and computing \( \bar{y}^* = \frac{1}{n} \sum_{i=1}^{n} y_i^* \), and

(b) repeating (a) \( M \) times (for large positive integer \( M \geq 100,000 \)) and making a histogram or kernel density trace of the values \( (\bar{y}_1^*, \ldots, \bar{y}_M^*) \).

Fact (Draper 2015): The posterior distribution \( p(\mu_C|D \mathcal{B}) \) induced by \( DP \left(n_C, \hat{F}_{n_C}\right) \) distribution may be sampled from accurately and quickly by (frequentist) bootstrapping the sample mean and interpreting the resulting distribution as a good approximation to \( p(\mu_C|D \mathcal{B}) \).
Summary of Conclusions

fact: (a) bootstrap is 30 times faster than standard DP sampling algorithm (stick-breaking), and (b) bootstrap is embarrassingly parallelizable

- captopril: bnp analysis coincides with gaussian-assumption analysis, because clt has kicked in even with only 12 obs per group, because skewness and kurtosis values in C and T are both so close to 0

- gold-standard analysis in some eCommerce companies: hope that captopril gaussian-assumption analysis is ‘close to optimal’; no proof that this hope is justified

fact: gmb has hideously non-gaussian skewness and kurtosis values

fact: but the gaussian-assumption analysis is still approximately optimal, provided that the C and T sample sizes n.C and n.T are large enough for the Central Limit Theorem (CLT) to save us
### Case Study Details

<table>
<thead>
<tr>
<th></th>
<th>number of zero group values</th>
<th>number of nonzero group values</th>
<th>total number of values</th>
<th>proportion of zero values</th>
<th>nonzero mean</th>
<th>total mean</th>
<th>nonzero SD</th>
<th>total SD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>treatment</strong></td>
<td>90,006</td>
<td>38,343</td>
<td>128,349</td>
<td>0.7013</td>
<td>3,618.0</td>
<td>1080.9</td>
<td>33,096</td>
<td></td>
</tr>
<tr>
<td><strong>control</strong></td>
<td>89,863</td>
<td>38,509</td>
<td>128,372</td>
<td>0.7000</td>
<td>3,387.5</td>
<td>1016.2</td>
<td>36,485</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>all values skewness</th>
<th>all values kurtosis</th>
<th>non-zero values skewness</th>
<th>non-zero values kurtosis</th>
<th>noise-to-signal ratio</th>
<th>noise-to-signal ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>treatment</strong></td>
<td>205.9</td>
<td>52,887.9</td>
<td>112.8</td>
<td>15,861.1</td>
<td>30.62</td>
<td>16.72</td>
</tr>
<tr>
<td><strong>control</strong></td>
<td>289.1</td>
<td>92,750.5</td>
<td>158.7</td>
<td>27,902.6</td>
<td>35.90</td>
<td>19.65</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>nonzero values min</th>
<th>nonzero values max</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>treatment</strong></td>
<td>0.09</td>
<td>9,381,532</td>
</tr>
<tr>
<td><strong>control</strong></td>
<td>0.09</td>
<td>12,018,199</td>
</tr>
</tbody>
</table>

- **lift estimate** = +0.0636 = + 6.36%
- **sd of lift estimate** = 0.1400 = +14.00%

p( \theta > 0 \mid data, background information ): gaussian 0.675 optimal 0.696
Example of Setting Where CLT is Not Good Enough

Let \( \eta = P(\theta > 0 | \text{data, background information}) \) in a segment (subset, stratum) of users that comprises 10\% of traffic in the Case Study (12,837 observations in each of T and C).

Here Gaussian analysis (current best practice in some of eCommerce) produces an estimate of \( \eta \) that’s too low by about 5\%, in relation to an optimal analysis; this underestimation gets worse with decreasing segment sample size.
Posterior Distribution of Lift (Uncapped)

theta
Density

-0.4 -0.2 0.0 0.2 0.4 0.6 0.8
0.0 0.5 1.0 1.5 2.0 2.5 3.0

theta
library( doParallel )
n.processors <- makeCluster( 1024 )
registerDoParallel( n.processors )

parallel.mean.bootstrap <- function( y, M, n, p.hat.0 ) {
  foreach( i = 1:M, .inorder = F, .multicombine = T, .combine = 'c' ) %dopar% {
    sum( sample( y, n - rbinom( 1, n, p.hat.0 ), replace = T ) ) / n
  }
}

seed <- 1
set.seed( seed )
M.b <- 100000

system.time(
)
(eCommerce, not captopril) case study: n.C and n.T are just barely big enough for gaussian-assumption analysis to be decent

fact: when clt has not yet kicked in, gaussian-assumption analysis will be conservative in the right tail (positive lift) and liberal in the left tail

conservative in the right tail means that the gaussian-assumption analysis might say $p(\theta > 0|DBG) = 0.88$ when really the optimal analysis concludes that $p(\theta > 0|DB) = 0.97$

this conservatism can be noticeable if n.C and n.T are quite small and the outcome variable is quite skewed and kurtotic
Design: Identify $n = (n_C + n_T)$ users representative of

$P = \{\text{all future users relevant to this experiment}\}$

(You have to specify relevant).

Randomize $n_C$ of these users to $C$ (current best environment without the $T$ intervention) and $n_T$ to $T$ (identical to $C$ but with the $T$ intervention).

(This is a completely-randomized experiment; better designs exist, but that’s another talk.)

Data summaries: sample means $(\bar{y}_C, \bar{y}_T)$, sample SDs $(s_C, s_T)$ for an outcome $y$ such as GMB.

Inferential target: population lift $\theta = \frac{\mu_T - \mu_C}{\mu_C}$, in which $\mu_C$ ($\mu_T$) is the population mean of $y$ under the $C$ ($T$) condition.

Algorithm (Gaussian approximation): (extremely fast, but may underestimate the posterior probability that the $T$ intervention is beneficial, especially in segments with small sample sizes)
\[ \hat{\theta} = \frac{\bar{y}_T - \bar{y}_C}{\bar{y}_C}, \quad \hat{SD}(\hat{\theta}) = \sqrt{\frac{\bar{y}_T^2 s_C^2}{\bar{y}_C^4 n_C} + \frac{s_T^2}{\bar{y}_C^2 n_T}}. \]

\[ p(\theta > 0|DBG) = 1 - \Phi \left[ \frac{-\hat{\theta}}{\hat{SD}(\hat{\theta})} \right], \quad (7) \]

in which \( \Phi(\cdot) \) is the standard normal CDF.

inferential suggestion (not yet a proper decision algorithm): consider launching the \( T \) if \( p(\theta > 0|DBG) > c \), where conventional (not necessarily in any sense optimal) values of \( c \) include 0.9, 0.95, and 0.99

this logic may be applied not only to the entire data set but also to smaller segments defined by covariates (features) (e.g., separately for male and female users)

arriving at many such inferential suggestions —

\{entire data set, segment 1, segment 2, \ldots, segment \( S \}\)

— (for large \( S \)) creates a multiplicity problem that’s best solved with Bayesian decision theory (another talk)
R code to implement this approximate algorithm:

\[
\text{lift.estimate} <- \frac{(\bar{y}_T - \bar{y}_C)}{\bar{y}_C}
\]

\[
\text{SD.lift.estimate} <- \sqrt{\frac{\bar{y}_T^2 \cdot s^2}{\bar{y}_C^4 \cdot n_C} + \frac{s_T^2}{\bar{y}_C^2 \cdot n_T}}
\]

\[
\text{gaussian.posterior.probability.of.improvement} <- 1 - \text{pnorm}\left(\frac{0 - \text{lift.estimate}}{\text{SE.lift.estimate}}\right)
\]

Even with \((n_C, n_T)\) each on the order of 10–100 million, this code takes less than 1 second to run on a laptop with one decent core and decent RAM.

Approximate validity of Gaussian algorithm depends on \((n_C, n_T)\) and the sample skewness and kurtosis values in each of \(C\) and \(T\).

\((*)\) Unfavorable conditions for this algorithm: \{small sample size, large skewness, large kurtosis\} in either or both groups.

In a future white paper (published to the experimentation wiki) I’ll quantify \((*)\)
Optimal Analysis Algorithm

Algorithm (optimal analysis): (accurate assessment of the posterior probability that the $T$ intervention is beneficial, but may be slow; however, the bootstrap is embarrassingly parallelizable)

to make a valid draw $\mu_T^*$ from the posterior distribution $p(\mu_T | y^T B)$ induced by the $DP \left( n, \hat{F}_T \right)$ posterior on $F_T$,

(a) choose a random sample $(y_{1T}^*, \ldots, y_{nT}^*)$ of size $n_T$ with replacement from the data vector $y^T$, and

(b) compute $\mu_T^* = \frac{1}{n_T} \sum_{\ell=1}^{n_T} y_{\ell T}^*$;

now repeat this $M_b$ times (for large $M_b$) and use a histogram or kernel density trace of the resulting $\mu_T^*$ draws to approximate $p(\mu_T | y^T B)$.

this reasoning obviously applies in parallel to obtain the corresponding posterior $p(\mu_C | y^C B)$ for the control-group population mean, and then to simulate from $p(\theta | y B)$, where $y = (y^C, y^T)$. You just

(a) bind the columns $(\mu_{C1}^*, \ldots, \mu_{CM_b}^*)$ and $(\mu_{T1}^*, \ldots, \mu_{TM_b}^*)$ together to make a matrix with $M_b$ rows and 2 columns,
(b) calculate \( \theta^*_m = \frac{\mu^*_m - \mu^*_c}{\mu^*_c} \) in row \( m = 1, \ldots, M_b \) of this matrix, and

\[
\begin{array}{llllllll}
\text{(c) use a histogram or kernel density trace of the resulting } M_b \theta^* \text{ draws to approximate } p(\theta | DB).
\end{array}
\]

<table>
<thead>
<tr>
<th>Mb</th>
<th>Elapsed Time (Sec) With</th>
<th>Bootstrap Distribution of ( \mu.T.star )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>8 Threads</td>
<td>24 Threads</td>
</tr>
<tr>
<td>10,000</td>
<td>104.82</td>
<td>65.67</td>
</tr>
<tr>
<td>100,000</td>
<td>1049.81</td>
<td>694.97</td>
</tr>
<tr>
<td>100,000</td>
<td>114.64</td>
<td>---</td>
</tr>
<tr>
<td>100,000</td>
<td>1076.14</td>
<td>---</td>
</tr>
</tbody>
</table>
Sometimes you can’t run randomized controlled trials in eCommerce.

Example: you release a new version of your mobile app every 4–6 months, but you allow users to choose when to pull it (rather than pushing it to everyone at the same time)

Q: Is the new app a disaster? (Want answer to this as fast as possible after release)

Users in the “treatment” group (early adopters of the new release) and the “control” group (people who initially continue to use the old release) are not assigned to T and C at random: the early adopters choose when to early-adopt, and they’re systematically different from the later-adopters (this is called selection bias)

typically the early-adopters are enthusiastic buyers

if you just look at monetary outcomes among T and C users (say) 4 weeks after release, the new release will look (much) better than it really is, because of selection bias
there will be millions of users in each of T and C, but this does not save you:

taking more measurements with a systematically biased data-gathering process just perpetuates the bias (unlike non-systematic noise, which you can damp down by averaging over many users)

there are many ways to attempt to estimate the size of the selection bias and adjust for it: standardization, regression, propensity scores, ...

with a Ph.D. student who’s interning in eCommerce, i’m currently working on large-scale time-series methods (based on dynamic linear models) for solving this problem

the idea is to let each user’s past buying behavior help you estimate what her/his buying would have been if she/he did/didn’t early adopt (plus adjusting for lots of other things too)

this method works well (paper coming soon)
Define the following terms:

- foreach; %dopar%;
- Hadoop;
- MapReduce;
- Scala.

Write a program that validly and quickly makes 1,000,000 monitoring draws, from a posterior distribution in \( k > 100 \) dimensions, without a for loop.

Your hardware consists of 10,000 cores and 20,000 threads, and 1 TB of fast RAM; write a valid and efficient program that fits a series of regression models to the following data set:

- 1,000,000,000 rows, 1,000 outcome variables (some binary, some quantitative on \( \mathbb{R}^+ \), some quantitative on \( \mathbb{R} \)) and 10,000 predictors (some binary, some quantitative on \( \mathbb{R}^+ \), some quantitative on \( \mathbb{R} \)), including the possibility of important two-way interactions; validate your model and construct well-calibrated predictive intervals for a new set of 1,000,000 rows.

Do all of this better than the machine-learning guys.
At company $X$, you have

(b) $B$, your current best web experience (e.g., users browse, looking at things they might buy), and

(a) $A$, a proposed modification to (b) (e.g., make the pictures of the items for sale a bit bigger).

(ecommerce Q:) Is (a) better than (b) at generating revenue?

(Statistical Reformulation:) If you cause the replacement of $B$ by $A$, will the effect on future company $X$ users constitute a net revenue gain?

In more detail:

— $P = \text{company } X \text{ users in the time window } T_1 = (4 \text{ weeks from now, 8 weeks from now}).$

— $\mu_B = \text{mean aggregate revenue} \text{ across } P \text{ in interval } T_1 \text{ if web experience } B \text{ continues.}$
Conventional Static A/B Testing

- In more **detail:**
  - $\mathcal{P} =$ company X users in the **time window**
    \[ T_1 = (4 \text{ weeks from now, 8 weeks from now}). \]
  - $\mu_B =$ **mean aggregate revenue** across $\mathcal{P}$ in interval $T$ if web experience $B$ continues.
  - $\mu_A =$ mean aggregate revenue across $\mathcal{P}$ in interval $T_1$ if web experience $(A + B)$ instead occurs.
  - Relative effect caused by $(A + B)$ versus $B =$ **lift** =
    \[ \theta = \frac{\mu_A - \mu_B}{\mu_B}. \]

- Conventional **static A/B testing approach** to estimating $\theta$:
  - Let $n_A = n_B = n$; **randomly** choose $(2n)$ company X users; **randomize** $n$ of them to $A$ and $n$ to $B$; let $\bar{y}_A$ and $\bar{y}_B$ be the **observed mean aggregate revenue** in groups $A$ and $B$, respectively, in the **time window**
    \[ T_0 = (\text{now, 4 weeks from now}); \]
Conventional Static $A/B$ Testing: Error # 1

- Conventional **static $A/B$ testing approach** to estimating $\theta$:
  - Let $n_A = n_B = n$; **randomly** choose $(2n)$ company $X$ users; **randomize** $n$ of them to $A$ and $n$ to $B$; let $\bar{y}_A$ and $\bar{y}_B$ be the **observed mean aggregate revenue** in groups $A$ and $B$, respectively, in the **time window** $T_0 = (\text{now, 4 weeks from now})$;

- **Estimated lift** is (posterior mean) $\hat{\theta} = \frac{\bar{y}_A - \bar{y}_B}{\bar{y}_B}$; **posterior SD** is $SD(\hat{\theta} \mid \text{data}) = \sqrt{\frac{\bar{y}_A^2 s_B^2}{\bar{y}_B^4 n} + \frac{s_A^2}{\bar{y}_B^2 n}} \ (s_A = \text{SD of } y \text{ in group } A)$; hope $n$ is big enough for **Central Limit Theorem** (CLT) to yield approximately **Gaussian** posterior for $\theta$; if so $p(\theta > 0 \mid \text{data}) = 1 - \Phi \left[ \frac{\hat{\theta}}{SD(\hat{\theta} \mid \text{data})} \right]$; implement $A$ if $p(\theta > 0 \mid \text{data})$ is “big enough.”

- **Error # 1**: This uses **inference** to make a **business decision**; **Bayesian decision-theoretic** reformulation leads to a completely different (and better) action rule.
Error # 2: Outcome $y$ has **noise-to-signal ratio** of $\frac{s_y}{\bar{y}} \approx 30–100$ (!); no point in looking for **tiny lifts** with inadequate sample sizes:

<table>
<thead>
<tr>
<th>false positive rate</th>
<th>false negative rate</th>
<th>true lift</th>
<th>sample size required in each of A and B groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>5%</td>
<td>20%</td>
<td>1.0%</td>
<td>12,365,114</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>1.0</td>
<td>12,174,946</td>
</tr>
<tr>
<td>15</td>
<td>50</td>
<td>0.5</td>
<td>19,334,495</td>
</tr>
<tr>
<td>10</td>
<td>20</td>
<td>1.0</td>
<td>20,285,382</td>
</tr>
<tr>
<td>10</td>
<td>50</td>
<td>0.5</td>
<td>29,562,739</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>1.0</td>
<td>29,562,739</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>1.0</td>
<td>38,537,313</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>0.5</td>
<td>48,699,782</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>0.5</td>
<td>154,149,252</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>0.5</td>
<td>221,307,004</td>
</tr>
</tbody>
</table>
Error # 3: Get clever: try to incentivize A/B testing employees by linking their bonuses to how much lift they seem to have found (maybe OK so far); but (not OK) allow them to keep “throwing wet spaghetti against the wall” by permitting them to run the same A/B test repeatedly with tiny variations until they get “statistical significance”; initial false positive rate of 5% rises to 50% and company X is managing noise; proof: aggregate all the lift “found” in year $N$, get 70%; actual company-wide lift in year $N$ was only 7%.

Error # 4: Concerned over heaviness of right tail of outcome $y$, you decide to ‘cap’ (Winsorize) $y$ by replacing all values bigger than (say) the 99.9th percentile with the 99.9th percentile, before doing the previous estimated-lift analysis; now your outcome variable has a much smaller noise-to-signal ratio, and you feel a lot better; but this introduces negative bias into your lift estimate of up to 85% (!); don’t do this.
What to Do Instead

- If you can, find **process (intermediate-outcome) variables** upon which to experiment that (a) have **smaller noise-to-signal ratios** and (b) are not only correlated with revenue but legitimately on the **causal path** for revenue improvement.
- Experiment with **many different factors at once**, in a **fractional-factorial design**, and estimate **2-way interactions** along with main effects.
- **Sequential adaptive** (not static) Bayesian design and analysis of experiments (e.g., **multi-armed bandits**) can reduce necessary sample sizes by up to **90%**.
- If you can, use **longitudinal (time-series) information** on each user to (dramatically) **sharpen causal understanding**.
- When the CLT is **not relevant**, use **Bayesian non-parametric analyses** to get at the underlying CDFs in the A and B groups; the **frequentist bootstrap** can dramatically speed up the computations.