A/B Testing: Next Steps

Lachlan Deer

Social Media and Web Analytics, 2024

Learning Goals

- Explain how CUPED decreases variance of estimates in an A/B Test
- Implement a CUPED analysis in R
- Explain why and when one needs to adjust standard errors in A/B Test analysis using linear regression
- Implement standard error adjustments in R
- Define the SUTVA assumption and analyze whether the it is appropriate in a particular setting
- Explain alternative experiment designs that allow unbiased treatment effect estimation when SUTVA would be violated in a standard test design

Where Are We Now?

So far:

- Randomization as a modus operandi to overcome selection effects and omitted variable bias
- Design and analysis of “standard” A/B tests

This lecture: Tweaking the standard design

- Reducing the variance of our estimates
- Correct inference when treatment allocation is at a coarser level than the data we analyse
- How to handle violations of a *hidden* assumption

1/ Variance Reduction with CUPED

What is CUPED?

CUPED: **Controlled-Experiment using Pre-Experiment Data**

- A technique to increase the power of randomized controlled trials in A/B tests.

How does it work?

Let's start with some data...

Testing the Effectivness of a New Recommender

Business questions: Does the new recommender system increase spending?

Test setting: Online Website, recommender system

Unit: A consumer

Treatments: control group, new recommender system

Reponse: spending in the next 14 days

Selection: all consumers who purchased in last 60 days

Assignment: randomly assigned (1/2 each)

Sample size: 2,000 consumers

The Data

```
# A tibble: 6 x 4
     id treatment_status pre_spend post_spend
  <dbl>            <dbl>     <dbl>      <dbl>
1     1                0     133.        97.7
2     2                1     107.        72.5
3     3                0      90.1       88.9
4     4                0      36.4       31.5
5     5                0     151.       162. 
6     6                0      33.6       11.9
```

We also observe consumer behaviour before the test

What We've been Doing So Far

$$spend_i = \beta_0 + \beta_1 Treatment_i + \varepsilon_i$$

```
mod <- lm(post_spend ~ treatment_status,
          data = df)
tidy(mod)
```

```
# A tibble: 2 x 5
  term             estimate std.error statistic p.value
  <chr>               <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)         89.7       1.62     55.3   0
2 treatment_status     4.25      2.28      1.86  0.0626
```

What can we improve?

Our **existing estimator is unbiased**

- Which means it delivers the correct estimate, on average.

Potential **improvement**: we could try to **decrease its variance**.

Decreasing the variance of an estimator is important since it allows us to:

- Detect smaller effects
- Detect the same effect, but with a smaller sample size

In general, an estimator with a smaller variance allows us to run tests with a higher power, i.e. ability to detect smaller effects.

CUPED

Suppose you are running an A/B test and Y is the outcome of interest (revenue in our example)

- The binary variable T indicates whether a single individual has been treated or not

Suppose you have access to **another variable** X at the unit level which is **not affected by the treatment**

- And has known expectation $E[X]$.

Can we use X to reduce the variance of the estimate of the average treatment effect?

CUPED

Define:

$$\hat{Y}^{CUPED} = \bar{Y} - \theta\bar{X} + \theta E[X]$$

This is an **unbiased estimator** for E[Y] since last terms cancel out

CUPED

However the **variance of $\hat{Y}^{CUPED}$ is lower** than Y:

$$Var(\hat{Y}^{CUPED}) = Var(\bar{Y})(1 - \rho^2)$$

where ρ is the correlation between Y and X

$\implies$ higher correlation between Y and X $\rightarrow$ higher variance reduction using CUPED

Estimating the ATE with CUPED

$$\begin{aligned}\widehat{ATE}^{CUPED} &= \hat{Y}^{CUPED}(T=1) - \hat{Y}^{CUPED}(T=0) \\ &= (\bar{Y} - \theta\bar{X} + \theta E[X] | T=1) - (\bar{Y} - \theta\bar{X} + \theta E[X] | T=0) \\ &= (\bar{Y} - \theta\bar{X} | T=1) - (\bar{Y} - \theta\bar{X} | T=0)\end{aligned}$$

Optimal Choice of Pre-Experiment Variable (X)

X should have the following properties:

- **Not affected by the treatment**
- Be as correlated with Y as possible

The authors of the original CUPED paper suggest using **pre-treatment outcome** variables since it gives the most variance reduction in practice.

Computing CUPED Estimate

1. Estimate $\hat{\theta}$ by regressing Y on X
2. Compute $\hat{Y}^{CUPED} = \bar{Y} - \hat{\theta}X$
3. Compute the difference of $\hat{Y}^{CUPED}$ between treatment and control groups

CUPED in Action: Estimating θ

```r
theta <-
    tidy(lm(post_spend ~ pre_spend, data = df)) %>%
    filter(term=="pre_spend") %>%
    select(estimate) %>%
    purrr::pluck('estimate')

print(theta)
```

```
[1] 0.8393084
```

```r
#alternative:
#cov(df$post_spend, df$pre_spend) / var(df$pre_spend)
```

CUPED in Action: Computing $\widehat{Y}_1^{CUPED}$

```
df <-
    df %>%
    mutate(cuped_spend = post_spend -
                theta*(pre_spend)
          )
```

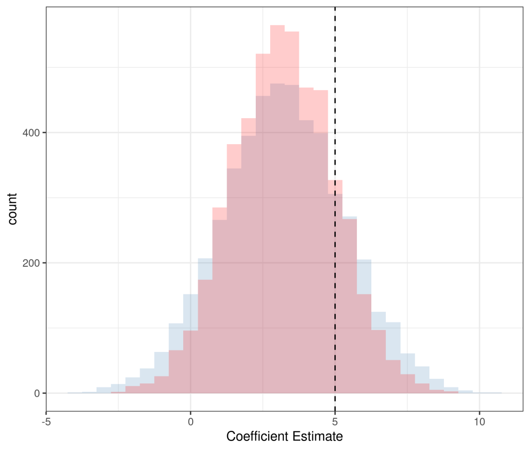
CUPED in Action: Estimate the ATE

```r
mod_cuped <- lm(cuped_spend ~ treatment_status,
                data = df)
tidy(mod_cuped)
```

```
# A tibble: 2 x 5
  term             estimate std.error statistic   p.value
  <chr>               <dbl>     <dbl>     <dbl>     <dbl>
1 (Intercept)          5.31      1.24      4.30 0.0000180
2 treatment_status     5.55      1.74      3.19 0.00144
```

CUPED Performance

Comparison of CUPED vs “standard” estimate over 5000 simulated datasets from the same DGP



Summary

- **CUPED aims to decrease the variance of the ATE** by leveraging additional consumer data that is unaffected by the experiment
- **CUPED transforms the outcome variable**, then we use our **conventional toolkit** to analyse the transformed data
- CUPED decreases variance **by using the additional data to make differences between groups “clearer”**

2/ Clustered Standard Errors

A Problem We Need to Solve

Unit of **treatment assignment differs from the unit of observation**

- Example 1: treat all customers in a certain region while observing outcomes at the customer level,
- Example 2: treat all articles of a certain brand, while observing outcomes at the article level.

Usually this happens because of practical constraints with how we can randomize

Implication: Treatment effects are “not independent” across observations

- Example 1: Customer in a region is treated, also other customers in the same region will be treated
- Example 2: If one article of a brand not treated, neither are any of the others

In our **inference** we have to **take this dependence into account**

Example: Customer Order Data and Recommenders Redux

Business questions: Does showing a carousel of related articles at checkout to incentivize customers to add other articles to their basket?

Test setting: Online Website, carousel introduction

Unit: A consumer

Treatments: control group, adding a carousel after adding an item to cart

Reponse: spending in the next 28 days

Selection: all consumers who purchased in last 60 days

Assignment: Display carousel to **consumers** at random

Sample size: 2,000 consumers

Load the Data

```
# A tibble: 6 x 3
   user treatment_status revenue
  <dbl>            <dbl>   <dbl>
1     1                1   192.
2     2                1    91.3
3     3                1    45.6
4     4                1   101.
5     5                0    88.2
6     6                0    15
```

Question: Do we see the same consumers make more than one purchase?

Question: If so, why might this be a problem?

Estimate the ATE the 'usual' way

```
tidy(lm(log(revenue) ~ treatment_status, data = recommender))
```

```
# A tibble: 2 x 5
  term             estimate std.error statistic p.value
  <chr>               <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)        4.38      0.0224    195.    0
2 treatment_status   0.0642    0.0311      2.07  0.0390
```

Question: What assumptions have we made about the distribution of the error term when we compute the standard error this way?

“Default” Standard Errors

- By default, R assumes **homoskedastic standard errors**:

$$Var(\varepsilon_i|X_i) = \sigma^2$$

and between any two observations:

$$Cov(\varepsilon_i, \varepsilon_j|X_i) = 0$$

In our setting:

- Variance of the error term is the same across consumers
- Covariance of error term is the same across consumers is zero
- **Covariance of error term between multiple purchases of the same consumer is zero**

Relaxing Homoskedasticity: Heteroskedasticity

Let's weaken these assumptions step by step:

- ~~Variance of the error term is the same across consumers~~
- **Variance of the error term is different across consumers**
- Covariance of error term is the same across consumers is zero
- Covariance of error term between multiple purchases of the same consumer is zero

$$Var(\varepsilon_i|X_i) = \sigma_i^2$$

Different assumption on $Var(\varepsilon_i|X_i) \implies$ different formula to compute standard error

- We'll skip the math (hurrah!)

Heteroskedasticity Robust Standard Errors

```
tidy(lm_robust(log(revenue) ~ treatment_status,
               data = recommender,
               se_type = "HC1"), conf.int = FALSE
     )
```

```
              term   estimate  std.error  statistic    p.value   df
1      (Intercept) 4.38103323 0.02310626 189.603753 0.00000000 2548
2 treatment_status 0.06422808 0.03117015   2.060564 0.03944597 2548
       outcome
1 log(revenue)
2 log(revenue)
```

Question: Do we see much of a difference in this case?

Relaxing Homoskedasticity: Clustering

- Let's weaken these assumptions step by step:
  - ~~Variance of the error term is the same across consumers~~
  - Variance of the error term is different across consumers
  - Covariance of error term is the same across consumers is zero
  - ~~Covariance of error term between multiple purchases of the same consumer is zero~~
  - **Covariance of error term between multiple purchases of the same consumer is non-zero**

For any two observations of the **same consumer**, g:

$$Cov(\varepsilon_{ig}, \varepsilon_{jg} | X_g) = \rho_g \sigma_{ig} \sigma_{jg}$$

Different assumption $\implies$ different standard error!

Cluster Robust Standard Errors

```
tidy(lm_robust(log(revenue) ~ treatment_status,
               data = recommender,
               cluster = user), conf.int = FALSE
     )
```

```
             term   estimate  std.error  statistic    p.value        df
1     (Intercept) 4.38103323 0.02545544 172.105997 0.00000000  844.4227
2 treatment_status 0.06422808 0.03469982   1.850963 0.06434326 1750.6688
       outcome
1 log(revenue)
2 log(revenue)
```

Question: Is there a difference now?

Summary

- If you **assign treatment at a higher level than your unit of observation**, you **need to correct the standard errors** in your analysis
- You should **cluster your standard errors** at the **level at which the treatment was allocated**
  - In our example: the consumer
- Cluster-robust standard errors are larger than the usual standard errors only if there is dependence across observations.
  - If observations are only mildly correlated across clusters, then cluster-robust standard errors will be similar to homoskedastic ones.

3/ The SUTVA assumption

SUTVA

So far we have made an (implicit) assumption:

”We know what the treatment is”

More precisely:

1. The **potential outcomes** for each unit **do not vary with the treatment assigned to other units** (no interference)
2. For each unit, there are **no different versions** of each **treatment** level (no hidden variation of treatments)

This is known as the **Stable Unit Treatment Value** (SUTVA) assumption

Why Might SUTVA Fail in Online Experiments?

The Stories team at Instagram tries to understand the effect of a new product feature

- e.g., a new emoji reaction

on user engagement, measured by the time on the app.

A simple randomization strategy at the user level assigns half of the population into the treatment group and the other half in the control.

Question: How does SUTVA fail here?

Why Might SUTVA Fail in Online Experiments?

Answer:

Users are connected on the platform, the control group increases (or decreases) the time spent on the app as their treated friends increase (or decreases) the engagement.

- The original assignment strategy does not work as expected because of user interference, a clear violation of the SUTVA assumption.

Is it a BIG deal?

In short, **yes!**

- Empirical studies and simulations show that the bias from interference ranges from 1/3 to the same size as the treatment effect
- It may mess with the direction of the treatment, e.g., turns positive effect into negative; vice versa.

Solutions to SUTVA Violations

Common solutions used in large (tech) companies

1. Coarser Levels of Randomization
2. Ego-Cluster Randomization
3. Switchback designs

We'll briefly talk about each one

Coarsening the Level of Randomization I

A ride-sharing company (e.g., Lyft) wants to check if a new matching algorithm improves User Retention.

Question: Can we randomly assign riders into the treated/non-treated conditions and compare the group means?

Coarsening the Level of Randomization II

Coarsening the Level of Randomization III

A tradeoff is present:

- Coarser granularity $\rightarrow$ fewer units
- Larger variance
- Less statistical power

Rule of Thumb: Aggregate data up to the granularity level that each unit won't interact with each other and we will still have a sufficient number of observations.

Real World examples:

- Ridesharing Marketplaces, Lyft
- Netflix

Ego-Cluster Randomization I

LinkedIn wants to know how a new introducing new reactions to posts impact engagement metrics on the platform

Question: Can they run a standard A/B test? Why or why not?

Ego-Cluster Randomization II

Ego-Cluster Randomization III

Solution: Ego-cluster randomization, which treats a focal person ("ego") and her immediate connections ("alter") as a cluster, then randomizes the treatment assignment at the cluster level.

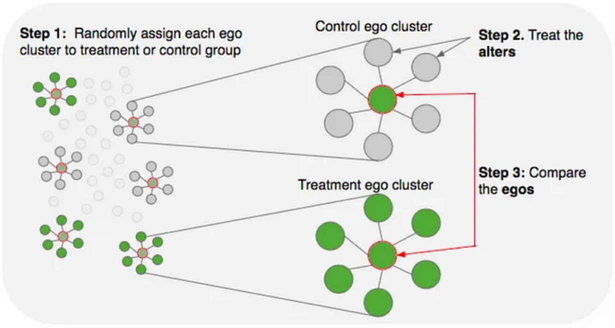


Figure 1: High level diagram of the method

Switchback Designs I

UberEats wants to test out how dynamic pricing (i.e., extra charge for rush hours) would affect customer experience, measured by User Retention.

Question: Why won't A/B tests at the user level work?

Switchback Designs II

Switchback Designs III

Solution: Chooses a higher level of analysis ...

... and randomize the treatment at distinct geography and time window

- Is called a Switchback design.
- The design toggles the treatment on and off at the distinct geography-time level and checks the changes in the outcome variables

Switchback designs IV

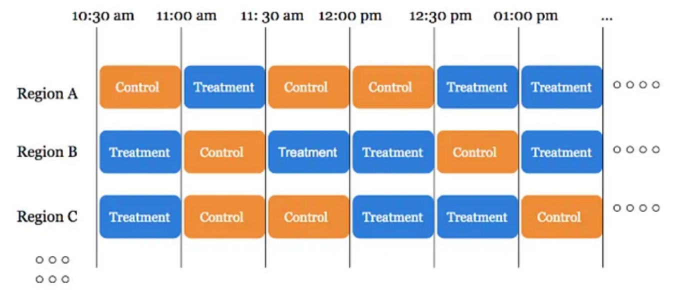


Illustration of time-market unit randomization

Switchback design assumes dependence within the clusters but independence among clusters.

4/ Recap

Summary

- CUPED decreases the variance of A/B test estimates by leveraging pre-treatment data that is unaffected by the experiment
- Robustifying standard errors at the unit of treatment prevents incorrect statistical inference
- Alternative A/B testing designs offer ways around violations of the SUTVA assumption

Acknowledgements

I have borrowed and re-mixed material from the following:

- Matteo Courthoud's "Clustered Standard Errors in A/B Tests" and "Understanding CUPED"
- Chi Huang's "What is SUTVA and What to Do When It's Violated in Practice"

License & Citation

Suggested Citation:

```
@misc{smwa2025_abtest,
      title={"Social Media and Web Analytics: A/B Tests - Next Steps"},
      author={Lachlan Deer},
      year={2025},
      url = "https://tisem-digital-marketing.github.io/2025-smwa"
}
```

This course adheres to the principles of the Open Science Community of Tilburg University. This initiative advocates for transparency and accessibility in research and teaching to all levels of society and thus creating more accountability and impact.