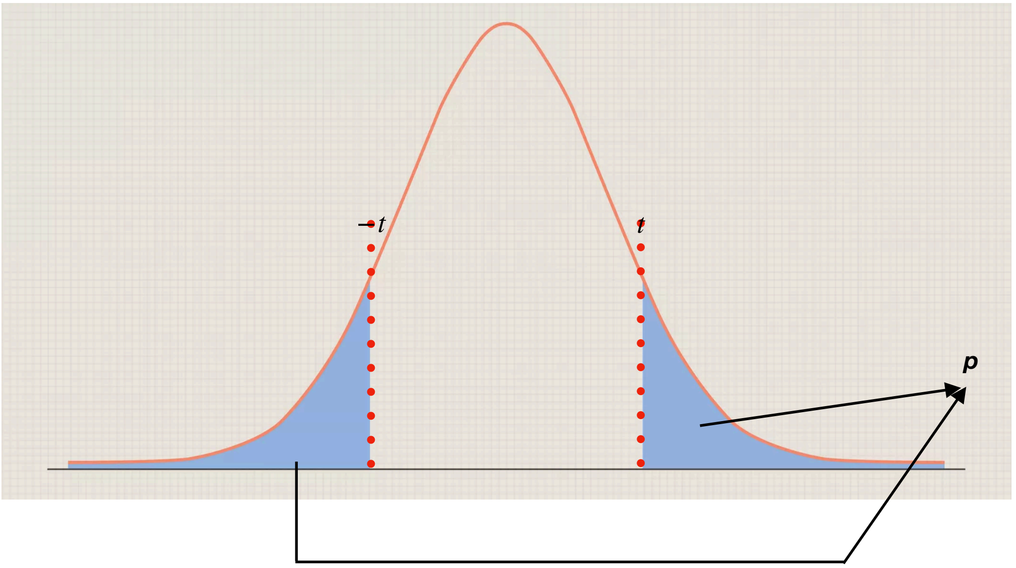
S5 Improve Sensitivity

Correct & Reduce Variance

Shan Huang, HKU

Variance σ

- Almost all the key statistics behind A/B testing are related to σ:
  - $t_{stat} = \dfrac{\Delta}{se(\Delta)} = (m_1 - m_0)/se$
    - $p = Pr(\,|T| \geq t \mid H_0)$
    - Statistical significance
  - CI = $[\Delta - se\,.\,t_{\alpha/2}, \Delta + se\,.\,t_{\alpha/2}]$
  - Type II error & Statistical power
    - Sample Size = $16\sigma^2/\delta^2$
- Smaller σ indicates
  - A greater power
  - Smaller sample for a certain statistical power (80%)



Standard Error

- Compute $se^2 = \text{Var}\ (\Delta) = \text{Var}\ (m_1 - m_0) = \text{Var}\ (\bar{Y}_1 - \bar{Y}_0)$
  - $\text{Var}\ (Y) = \dfrac{1}{n-1} \sum (Y_i - \bar{Y})^2$
  - $\text{Var}\ (\bar{Y}) = \dfrac{1}{n}\text{Var(Y)}$

    Assumption: Observations of units are independent
  - $\text{Var}\ (\Delta) = \text{Var}\ (\bar{Y}_1 - \bar{Y}_0) = \text{Var}\ (\bar{Y}_1) + \text{Var}\ (\bar{Y}_0) = \dfrac{1}{n_1}\text{Var(Y1)} + \dfrac{1}{n_0}\text{Var(Yo)}$
  - $\text{se}\ (\Delta) = \sqrt{Var(\Delta)}$
- $\text{t} = \Delta\ /\ \text{se}$
- $\text{CI} = [\Delta - 1.96.se, \Delta + 1.96.se]$

We want to:

1. **Correctly estimate se.**
2. **Reduce variance σ^2 if we can.**

If se is overestimated (underestimated), will type I or II error happen?

Correlated Units (Observations)

- Students' behaviors in the same class can be correlated.
- Users' behaviors on the same ads can be correlated.
- Behaviors on different pageviews of the same user can be correlated.
- Users' behaviors on the same day can be correlated.
- Behaviors on different searches of the same user can be correlated.

What will you do? A Classic Problem

- OEC is clicks/pageview (pv)
- Two cases:
    1. Randomization Unit = User
    2. Randomization Unit = Page View

Ratio Metrics

1. Randomization Unit = User

- OEC: clicks/pageview
- Behaviors are independent (without network interferences) across users
- Transform to a ratio metric between two averages of user level metrics
  - $m = \frac{\bar{X}_1}{\bar{X}_2}$
  - $\bar{X}_1$: # clicks/# users
  - $\bar{X}_2$: # pvs/# users
- m = # clicks/ # pvs = OEC

Ratio Metrics

Why don't we compute #clicks/#page views for each user and then compare the mean?

	user	group	page_view_cnt	click_cnt
1	1	0	6	1
2	2	1	1	0
3	3	1	6	1
4	4	0	3	2
5	5	1	1	1
6	6	0	0	0
7	7	1	0	0
8	8	0	5	1
9	9	1	7	3
10	10	1	0	0

Ratio Metrics

- $\bar{X}_1, \bar{X}_2$ are two random variables that are jointly bivariate normal in the limit
- m is also a normally distributed random variable.
- By Delta Method:

$$\text{Var (m)} = \frac{1}{\bar{X}_2^{\ 2}} Var(\bar{X}_1) + \frac{\bar{X}_1^{\ 2}}{\bar{X}_2^{\ 4}} Var(\bar{X}_2) - 2\frac{\bar{X}_1}{\bar{X}_2^{\ 3}} Cov(\bar{X}_1, \bar{X}_2)$$

$$Cov(\bar{X}_1, \bar{X}_2) = \frac{1}{n} Cov(X_1, X_2)$$

- Delta Method:
    - A method concerning the approximated probability distribution for a function of an asymptotically normal statistical estimator

Compare Two Ratio Metrics

- Compare m_1, m_0 (ratio metrics of treatment and control groups)
- Compute $se^2 =$ Var (Δ) = Var ($m_1 - m_0$)
  - Var (Δ) = Var ($m_1 - m_0$) = Var (m_1) + Var (m_0)
  - se (Δ) = $\sqrt{Var(\Delta)}$
  - CI=$[\Delta - 1.96.se, \Delta + 1.96.se]$

Variance of Lift (Ratio Metrics)

- Bootstrap is computational expensive.
- Statistical tests are strongly preferred to Bootstraps
- Lift = Δ % = Mean_OEC(treatment)/Mean_OEC(control)

$$\text{Var}\ (\Delta\ \%\ ) = \frac{1}{\bar{Y}_0^{\ 2}} Var(\bar{Y}_1) + \frac{\bar{Y}_1^{\ 2}}{\bar{Y}_0^{\ 4}} Var(\bar{Y}_0)$$

$$\text{se}\ (\Delta\ \%\ ) = \sqrt{Var(\Delta\ \%\ )}$$

$$\text{CI}=[\Delta\% - 1.96.se, \Delta\% + 1.96.se]$$

Class Exercise

Compare z and bootstrap confidence intervals of a lift.

```
lift = 1.1
ctr0=0.5
n0=1000
n1=1000
ctrl = np.random.binomial(30, p=ctr0, size=n0) * 1.0
test = np.random.binomial(30, p=ctr0*lift, size=n1) * 1.0

m1=np.mean(test)
m0=np.mean(ctrl)
lift=m1/m0
var0 = np.var(ctrl,ddof=1)
var1 = np.var(test,ddof=1)
print(lift)
```

Class Exercise

Compute variance of lift

```
var_m0=var0/n0
var_m1=var1/n1
var_lift = (1/m0**2)*var_m1+(m1**2/m0**4)*var_m0
se_lift=np.sqrt(var_lift)
ci = (lift-1.96*se_lift,lift+1.96*se_lift)
```

Clustered Standard Errors

Randomization Unit = Page View

- We cannot use ratio metrics of two user-level averages. WHY?
  - The page views of the same user may be assigned to different groups
- Same users' behaviors on different page views can be correlated.
- We apply a more advanced statistical method: regression with clustered standard errors.
- **Analysis level should be consistent with the randomization unit (level)**

treat	pv	uin	# clicks
1	1	1	1
0	2	1	0
1	3	1	3
1	4	1	4
0	5	1	1
0	6	2	2
1	7	2	3
0	8	2	0
0	9	2	0
1	10	2	2
1	11	2	1

Clustered Standard Errors

- Correctly estimate the standard error of a regression parameter in settings:
  - observations can be subdivided into smaller-sized groups ("clusters", e.g., users)
  - & observations are correlated within each group.
- Review: OLS (ordinary least squares) gives the same results with using t-tests

  $y_i = \beta_0 + \beta_1 . T_i + \epsilon_i, \;\; \epsilon_i$ is IID

  $T_i = \{0,1\}$, Control or Treatment

  y_i is the OEC (or other metrics)

  $\mu = \beta_0 + \beta_1 - \beta_0 = \beta_1$

- OLS assumes *Independent and Identically Distributed* observations (y_i)

Clustered Standard Errors

- Two common corrections of standard errors:
  - Assume unequal variances, *heteroscedasticity,* across clusters
  - Assume correlations within clusters (e.g., users)
- Clustered standard errors can correct both at cluster-level:
  - Unequal variances across clusters
  - Correlations within clusters

Know your data before analysis

- If randomization units are smaller than users, observations of units are likely to cluster at the user level.
    - Metrics are correlated within same users
- You need to understand your data to
    - Choose a correct error structure to correctly estimate standard errors.
- You might think your data correlates in more than one way
    - Cluster your data in different ways
    - Cluster your data in multiple dimensions

Covariance Matrix of Error Terms

$$y_i = \beta_0 + \beta_1 \,.\, T_i + \epsilon_i = \hat{y} + \epsilon_i$$

$$\Omega = diag(\Sigma_g)$$

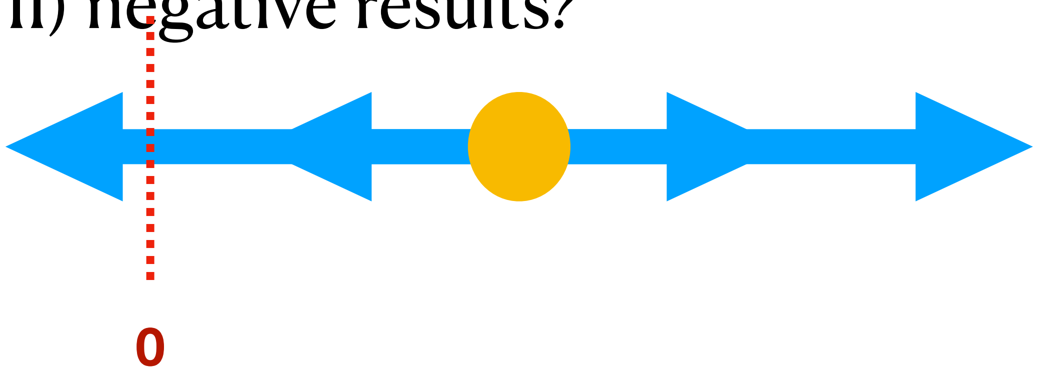
$$= \begin{bmatrix}
\sigma^2_{(11)1} & \cdots & \sigma_{(1N_1)1} & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & & \vdots & & \vdots \\
\sigma^2_{(N_11)1} & & \sigma_{(N_1N_1)1} & 0 & & 0 & & 0 & & 0 \\
0 & \cdots & 0 & \sigma^2_{(11)2} & \cdots & \sigma_{(1N_2)2} & & \vdots & & \vdots \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & & & & \\
0 & \cdots & 0 & \sigma_{(N_21)2} & \cdots & \sigma^2_{(N_2N_2)2} & & & & \\
 & & & & & & \ddots & \vdots & & \vdots \\
 & & & & & & & \sigma^2_{(11)g} & \cdots & \sigma_{(1N_g)g} \\
 & & & & & & & \vdots & \ddots & \vdots \\
 & & & & & & & \sigma_{(N_g1)g} & \cdots & \sigma^2_{(N_gN_g)g}
\end{bmatrix}$$

Clustered Standard Errors

- Clustered SE would increase se but do not change point estimate $\hat{\beta}$

$$t_{stat} = \frac{\hat{\beta} - \beta}{se(\hat{\beta})} = \frac{\hat{\beta} - 0}{se(\hat{\beta})} = \frac{\hat{\beta}}{se(\hat{\beta})}$$

    - t would be closer to 0
    - p-value would be larger, resulting in a less significant difference.
    - Confidence intervals would be wider.
- Without clustered SE, the $se(\hat{\beta})$ would be underestimated,
    - Lead to false positive (Type I) or (Type II) negative results?
    - Conclude wrong effects.



Correlated Units (Observations)

- Observations of units can be correlated
  - e.g., OECs are clicks/pageview (pv)
  - Same users' clicking behavior on different pageviews can be correlated.
- Two cases:
  1. Randomization Unit = User

     Ratio Metrics of Two user-level averages

  2. Randomization Unit = Page View

     Clustered Standard Errors

Class Exercise

- User data, 'exp_data_cluster.csv'
- Calculate & Compare the treatment effects with OLS with and without clustered standard errors at ad-level
    - Users behaviors can be correlated within the same ads

```
result = model.fit(cov_type='cluster', cov_kwds
= {'groups': data.adid})

print('Result with cluster')

print(result.summary2())
```

Class Exercise

```
import pandas as pd

import numpy as np, statsmodels.stats.api as sms

import statsmodels.api as sm

import statsmodels.formula.api as smf

infile = 'exp_data_cluster.csv'

data = pd.read_csv(infile)


model = smf.ols(formula='if_click ~ expid',
data=data)
```

Class Exercise

```python
import pandas as pd
import numpy as np, statsmodels.stats.api as sms
import statsmodels.api as sm
import statsmodels.formula.api as smf
infile = 'exp_data_cluster.csv'
data = pd.read_csv(infile)

# set the functional form of the regression
model = smf.ols(formula='if_click ~ expid', data=data)
```

Class Exercise

```
# OLS with Clustered SE
result = model.fit(cov_type='cluster', cov_kwds =
{'groups': data.adid})
print('Result with cluster')
print(result.summary2())

# OLS without Clustered SE
result = model.fit()
print('Result without cluster')
print(result.summary2())
```

We want to:

1. Correctly estimate variance.
2. Reduce variance σ^2 if we can.

Improve Sensitivity (Power)

1. Reduce Variance
    - Transform Metrics (dummies, log, capping)
    - Paired Design (interleaving, test algorithms)
2. Increase Sample Size
    - More granular randomization units
    - Pooled Control Group (Increase No & Large Control Group)
3. Increase Effect Size (δ) (OECs)
    - Trigger Experiments

Reduce σ^2

Reduce population variance of metrics

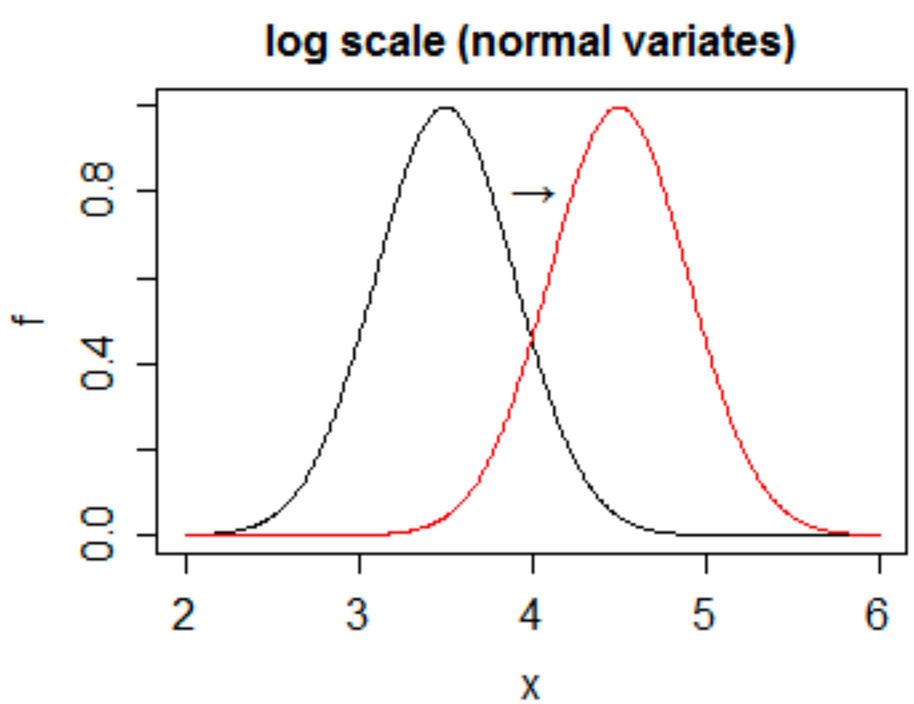
Create an metric (OEC) with a smaller variance:

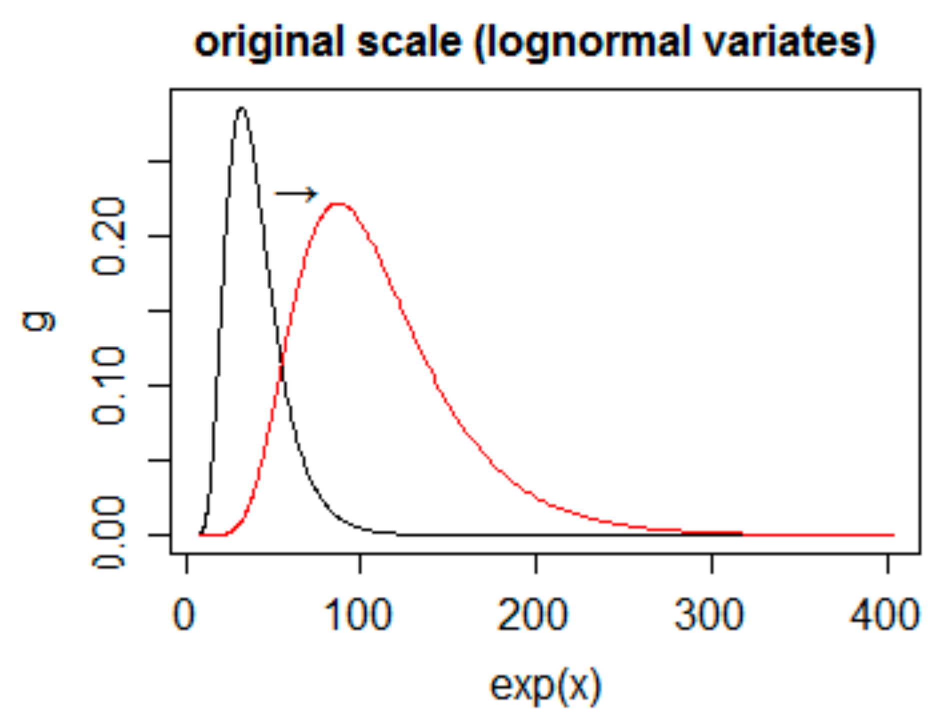
- σ^2 (# searches) > σ^2 (# searchers)
- σ^2 (purchase amount) > σ^2 (dummy(purchase))
- σ^2 (# clicks) > σ^2 (dummy(click))
- σ^2 (# messages between friends) > σ^2 (dummy(message))
    - Strong tie measure between two users: d(message) in the last month

Reduce σ^2

Transform a metric

- Capping
- Binarization
    - Netflix uses binary metrics to indicate whether users stream more than x hours (heavy users)
- Log tranform
    - Log transform heavy long-tailed metrics (e.g., sales)
    - Transform skewed data to a normal distribution
    - Hypothesis Testing: $\log(y_1) > \log(y_2) \Leftrightarrow y_1 > y_2$





Log Transform of OECs

- Log(x) = ln(x)
- Log transform the Treatment Effects:
  - $\delta_{log} = E(log(Y_i(1)) - log(Y_i(0)) = E(log\frac{Y_i(1)}{Y_i(0)}) = log(E(\frac{Y_i(1)}{Y_i(0)}))$
  - E(lift) = $e^{\delta_{log}} = E(\frac{Y_i(1)}{Y_i(0)})$
  - For example:
    - $\Delta_{log} = 0.7$
    - Lift = $\frac{m_1}{m2} = e^{0.7}$ = 2.014

Improve Sensitivity (Power)

1. Reduce Variance
    - Transform Metrics (dummies, log, capping)
    - Paired Design (interleaving, test algorithms)
2. Increase Sample Size
    - More granular randomization units
    - Pooled Control Group (Increase No & Large Control Group)
3. Increase Effect Size (δ) (OECs)
    - Trigger Experiments

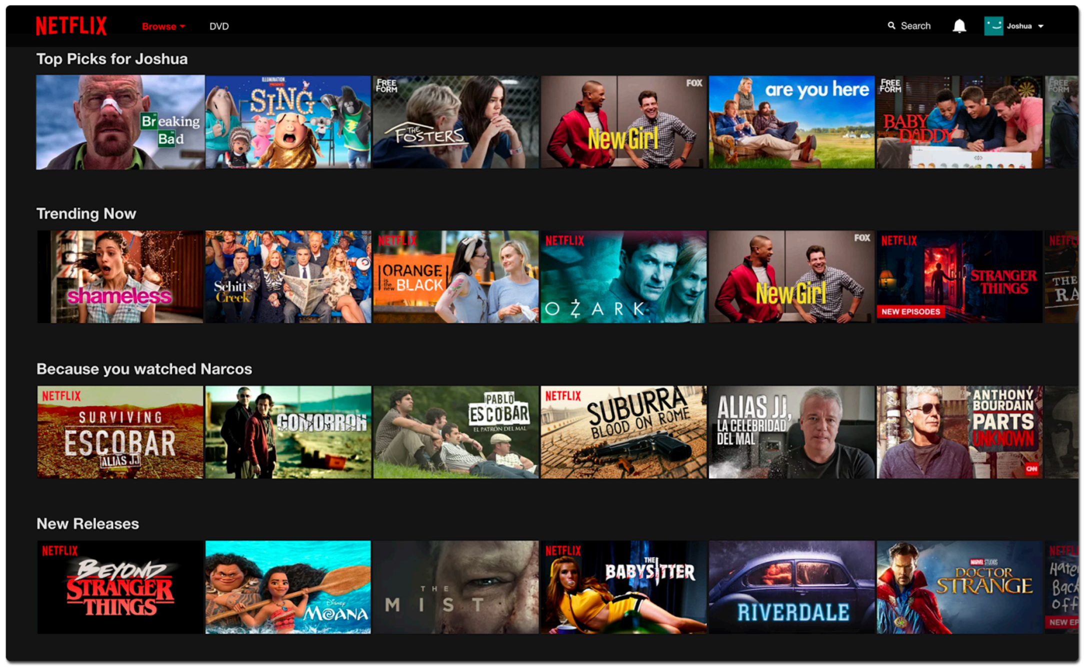
Reduce SE

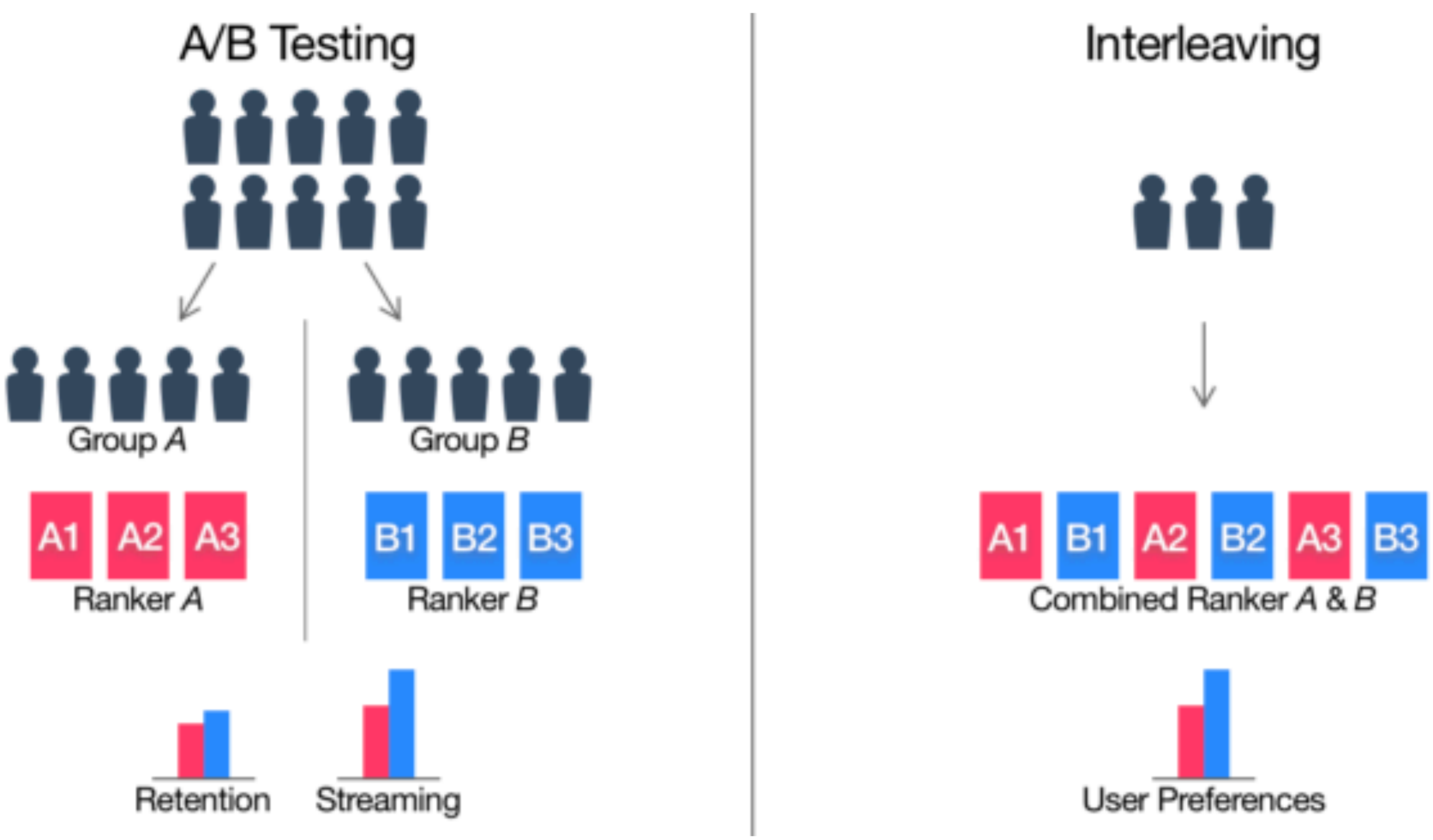
- Paired Design
    - Show the same user both Treatment and Control Conditions in a paired design.
        - To remove between user variability
        - To achieve smaller σ^2
    - e.g., Interleaving design
        - A popular method for evaluating ranked lists.

Class Exercise

- Compare the se, t stat, and p value between interleaving and traditional AB testing.
- Compute $se^2 = \text{Var}(\Delta) = \text{Var}(m_1 - m_0) = \text{Var}(\bar{Y}_1 - \bar{Y}_0)$
  - $\text{Var}(Y) = \dfrac{1}{n-1}\sum (Y_i - \bar{Y})^2$
  - $\text{Var}(\bar{Y}) = \dfrac{1}{n}\text{Var(Y)}$
  - $\text{Var}(\Delta) = \text{Var}(\bar{Y}_1 - \bar{Y}_0) = \text{Var}(\bar{Y}_1) + \text{Var}(\bar{Y}_0)$
  - $\text{se}(\Delta) = \sqrt{Var(\Delta)}$
- $\text{CI} = [\Delta - 1.96.se, \Delta + 1.96.se]$

Interleaving Design at Netflix





- Test two algorithms for recommending movies.
- Users hardly notice the treatments.
- Compare % hours users viewed movies recommended by A and B.
- $\Delta_i = y_i(B) - y_i(A)$
- Test $\delta_i \neq 0$ with t-tests

Interleaving Difference? AB Testing

UIN (i)	Y0_i	Y1_i	A_i-B_i
111	3	1	-2
112	4	4	0
113	2	0	-2
114	0	0	0
115	1	0	-1
116	0	1	1
117	2	1	-1
118	8	7	-1
119	5	5	0
120	1	1	0
121	10	11	1
122	12	10	-2
123	1	0	-1
124	3	2	-1
		Delta=	
		Variance =	
		SE =	
		t =	
		p value =	

UIN	VARIANT	Y_i
111	1	1
112	0	4
113	1	0
114	0	0
115	1	0
116	1	1
117	0	2
118	0	8
119	1	5
120	0	1
121	1	11
122	0	12
123	0	1
124	1	2
	Delta=	
	Variance =	
	SE =	
	t =	
	p value =	

What will you do?

If A and B recommend the same results:
A1B1… A1 and B1 are the same movies (docs).

Solution: always select the doc with the highest rank among the ones different from those already recommended.

L1 (A): d1, d2, d3, d4	A1 B2 A3 B3
L2 (B): d1, d2, d4, d3	d1 d2 d3 d4

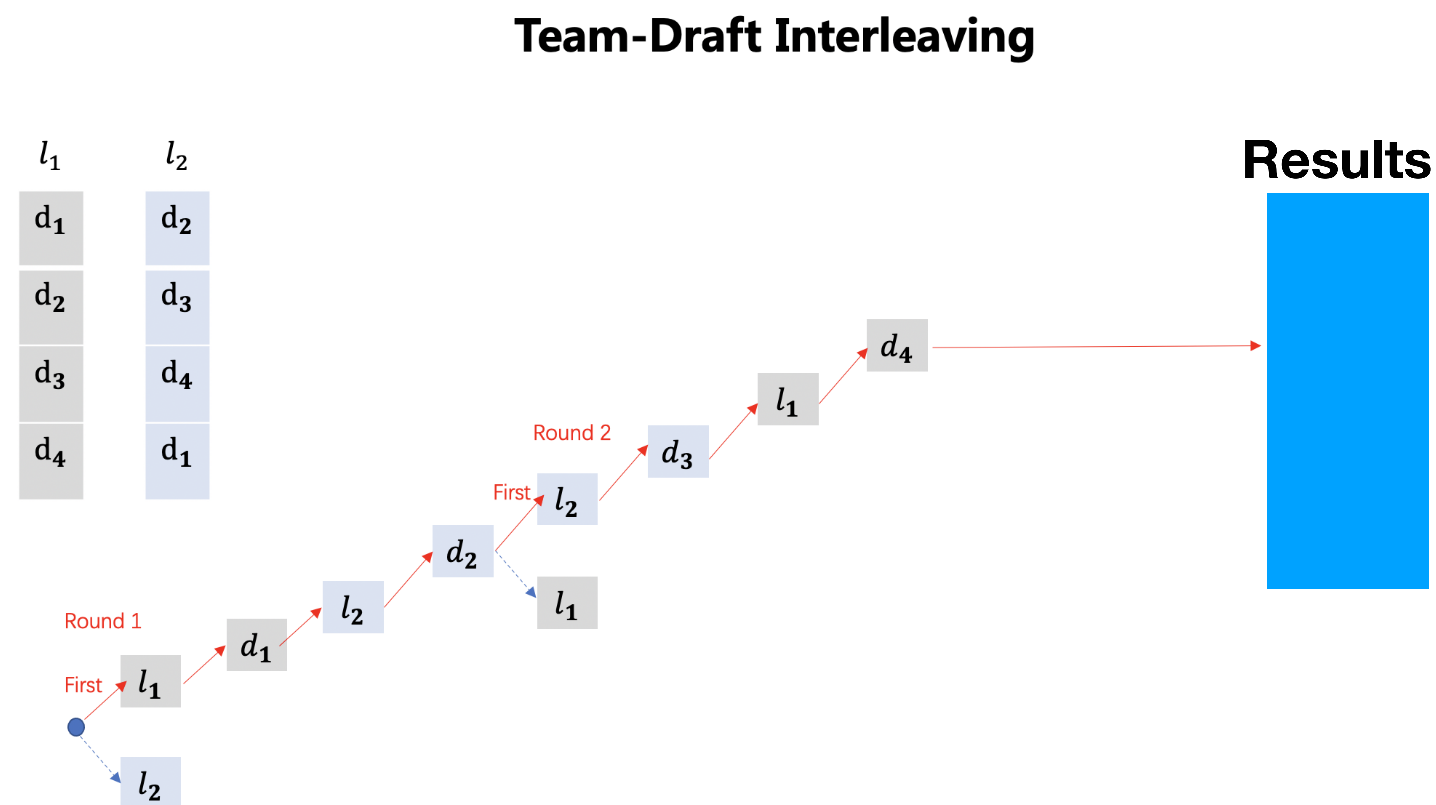
Is there any bias ?

If we always have A recommends first…

Team-Draft Interleaving

Procedure:

1. Unit 1:
   A. Randomly select A or B as the first ranker (if it is A) —> A1B1
   B. Randomly select A or B as the first ranker (if it is B) —> B2A2
   C. Until the doc n-1 for Unit 1
2. Unit N: follow the same procedure as Unit 1



Any threats to internal validity?
Randomized the bias among docs

Reduce se (Δ)

Randomize at a more granular unit.

- Var (Δ) = Var $(\bar{Y}_1 - \bar{Y}_0)$ = Var $(\bar{Y}_1)$ + Var $(\bar{Y}_0)$
- se (Δ) = $\sqrt{Var(\Delta)}$
- Var $(\bar{Y})$ = $\frac{1}{n}$Var(Y)
- Var (Δ) = Var $(\bar{Y}_1 - \bar{Y}_0)$ = Var $(\bar{Y}_1)$ + Var $(\bar{Y}_0)$= $\frac{1}{n_1}$Var(Y1)+ $\frac{1}{n_0}$Var(Y0)

Reduce sample mean variance by increasing n

Improve Sensitivity (Power)

1. Reduce Variance
   - Transform Metrics (dummies, log, capping)
   - Paired Design (interleaving, test algorithms)
2. Increase Sample Size
   - More granular randomization units
   - Pooled Control Group (Increase No & Large Control Group)
3. Increase Effect Size (δ) (OECs)
   - Trigger Experiments

More Granular Randomization Units

- Randomization unit
  - pages instead of user
    - Be aware of the disadvantages of randomizing units smaller than users.
    - Inconsistent UI
    - Correlated observations

Reduce se (Δ)

Increase Sample Size for Control Groups

- $\text{Var}(\Delta) = \text{Var}(\bar{Y}_1 - \bar{Y}_0) = \text{Var}(\bar{Y}_1) + \text{Var}(\bar{Y}_0)$
- $\text{se}(\Delta) = \sqrt{Var(\Delta)}$
- $\text{Var}(\Delta) = \text{Var}(\bar{Y}_1 - \bar{Y}_0) = \text{Var}(\bar{Y}_1) + \text{Var}(\bar{Y}_0) = \frac{1}{n_1}\text{Var}(Y_1) + \frac{1}{n_0}\text{Var}(Y_0)$

- Reduce Var ($\bar{Y}_0$) by increasing n_0 of the control group (existing features).
- Why don't we use all the rest traffic for the control group,
  - Other experiments also need the traffic to treat with new features.
  - A very large n_0 cannot save a tiny n_1

Improve Sensitivity (Power)

1. Reduce Variance
   - Transform Metrics (dummies, log, capping)
   - Paired Design (interleaving, test algorithms)
2. Increase Sample Size
   - More granular randomization units
   - Pooled Control Group (Increase No & Large Control Group)
3. Increase Effect Size (δ) (OECs)
   - Trigger Experiments

Reduce se (Δ) 2

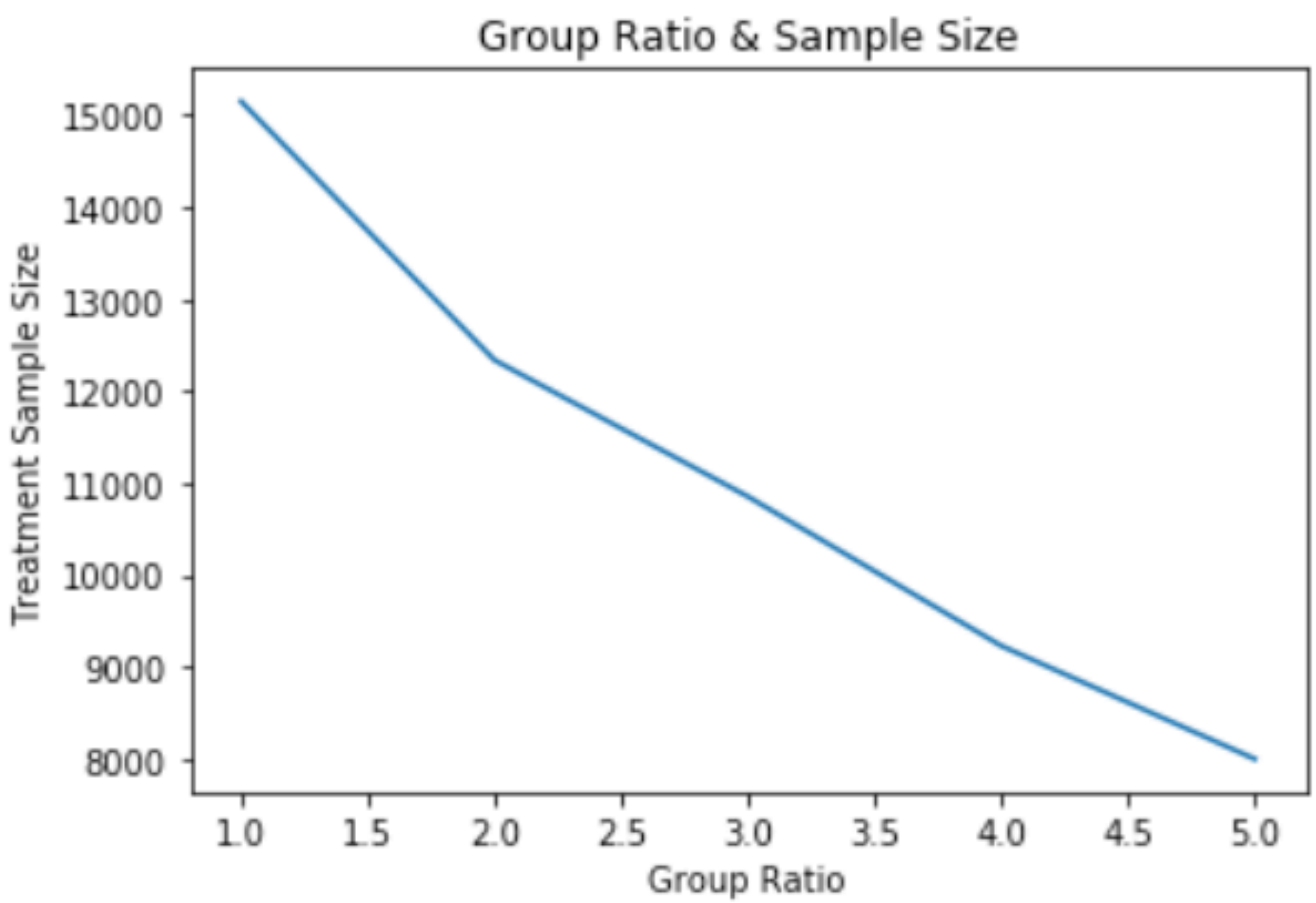
- Pool Control Groups
  - Consider pooling the separate controls to form a large, shared Control group.
  - Compare each Treatment with this shared Control group to increase the power for all the experiments
  - The Control group should be exactly the same for all the experiments.
  - *However*, equal variants lead to faster normality convergence.

Sample Ratio: 1 vs. 3

- Sample Size for the Treatment Group
  - 6400 (1:1) vs 4300 (1:3)
- Total Sample Size
  - 12800 vs 17200
- Control Group - Existing system with no change
- Treatment Group - Risky new feature
- We want to decrease the risks for user experience by minimizing the sample size for the Treatment Group with the same power (80%).

Power Analysis for Unequal Sample Sizes

- Increase the sample size for the control group will reduce that of the treatment group, for the statistical power = 0.8 & given δ, σ
- However, a very large n_0 cannot "save" a very small n_1.
  - se(m1) cannot be too large



Half the size of the treatment group will require about three times the size of the control group.

If you get a certain amount of traffic for your experiment, how should we split the sample between Treatment and Control to achieve the largest power?

Class Exercise

- Same Setting:
  - We expect a new feature to increase at least 5% purchase rate. The purchase rate among the triggered users is 50%. - effect size, variance,
- How should we split the traffic - 15000 users (i.e., find the best ratio between the sample sizes of the control and treatment) to achieve the largest statistical power?

Class Exercise

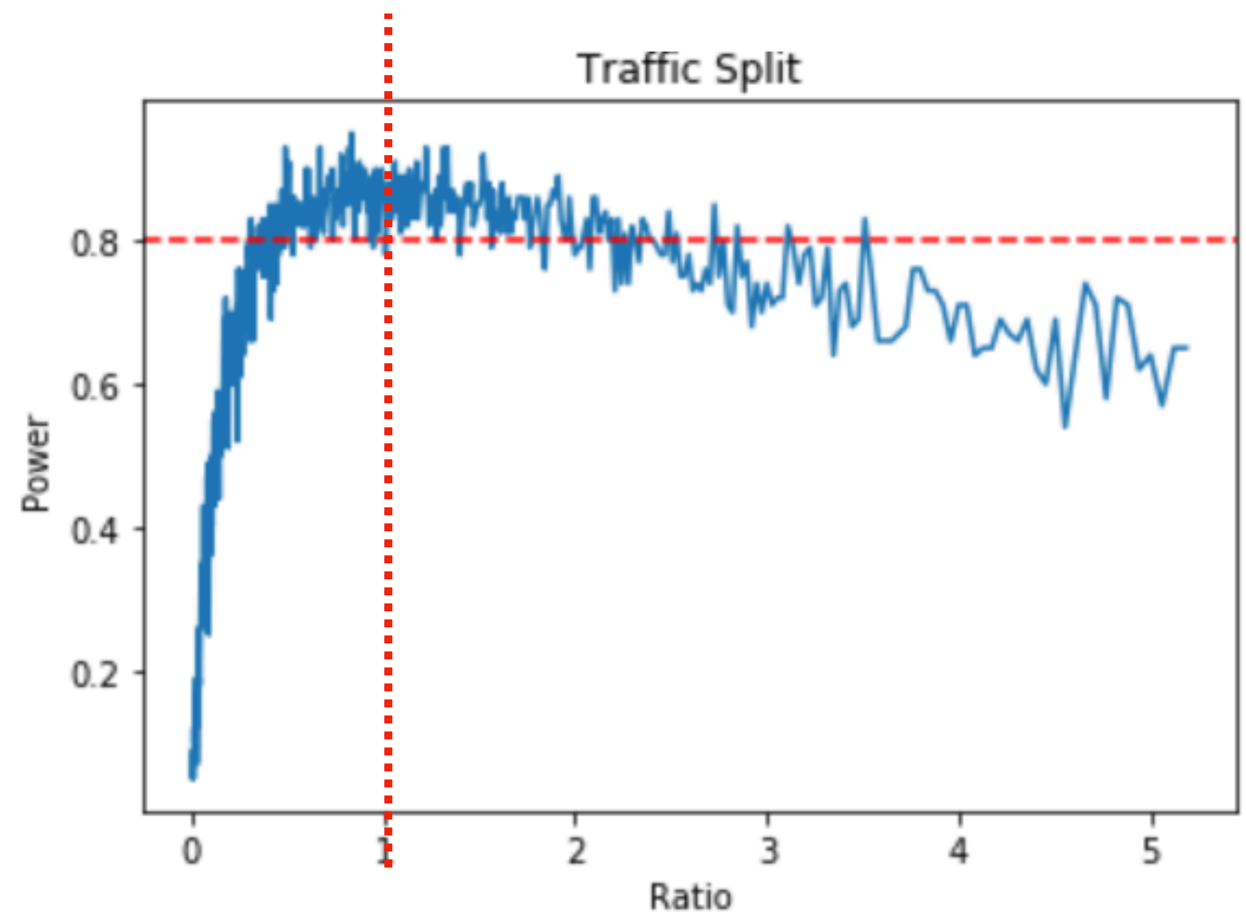
```
import numpy as np, statsmodels.stats.api as sms
import matplotlib.pyplot as plt
import pandas as pd
# Find Statistical Power for different traffic split
lift = 1.05
p0 = 0.5
power=[]
n=100
m=500
for s in range(500):
    k =2500+25*(s+1)-100
    ci=[]
    for i in range(n):
        ctrl = np.random.binomial(1, p0, 15000-k)
        test = np.random.binomial(1, p0*lift, k)
        cm = sms.CompareMeans(sms.DescrStatsW(test), sms.DescrStatsW(ctrl))
        a,b = cm.tconfint_diff(alpha=0.05, alternative='two-sided', usevar='pooled')
        ci.append((a,b))
    t2=sum((x[0]<=0 and x[1]>=0) for x in ci)/n
    pw=1-t2
# Ration of Sample Sizes between Treatment and Control
    r=(15000-k)/k
    power.append((r,pw))
```

Class Exercise

```
l_y=[x[1] for x in power]
s_x=[x[0] for x in power]

plt.plot(s_x,l_y)
plt.title('Traffic Split')
plt.xlabel('Ratio')
plt.ylabel('Power')
plt.axhline(y=0.8, color='r', linestyle='-
```

Equal Split Results in the Greatest Power



Improve Sensitivity (Power)

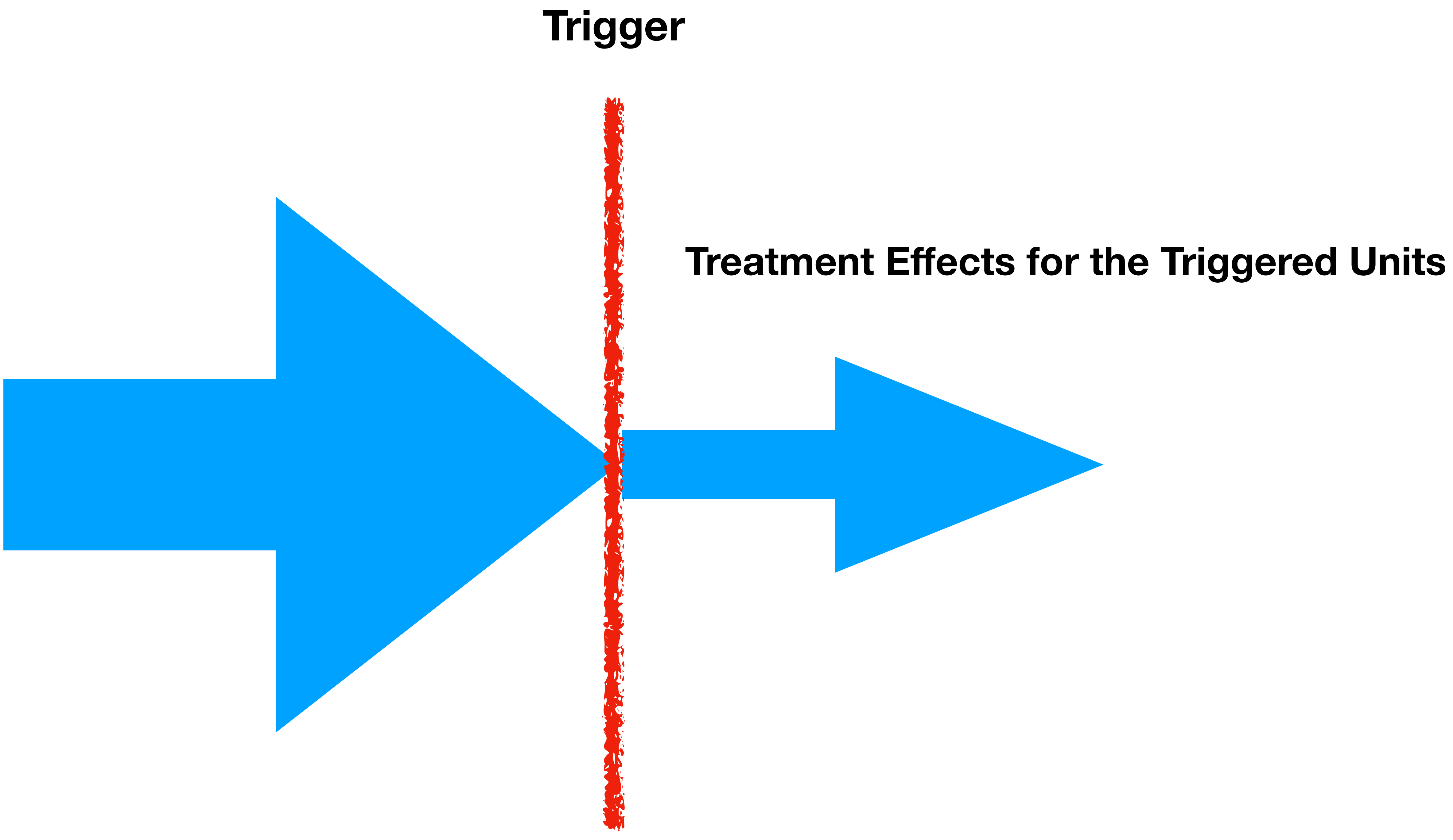
1. Reduce Variance
   - Transform Metrics (dummies, log, capping)
   - Paired Design (interleaving, test algorithms)
2. Increase Sample Size
   - More granular randomization units
   - Pooled Control Group (Increase No & Large Control Group)
3. Increase Effect Size (δ) (OECs)
   - Trigger Experiments

Triggering Experiments

Triggering Experiment

- If the experiment only impacts some users, filter out the noises: Units not impacted by the treatments.
  - e.g., Recommender systems for the restaurants nearby for users who have never searched for restaurants.
  - e.g., Features of adding search history for users who never started a search.
  - e.g., Users never updated WeChat during the experiment.
  - e.g., Features only for new users.
  - e.g., Feature only for active users.

Triggering Experiment



A Numerical Example

- Please calculate σ^2 for the triggering and non-triggering experiment.
  - You test a new feature for the checkout process.

A. The e-commerce site with 5% purchase rate. The conversion (purchase) event is a Bernoulli trial with $p = 0.05$.

B. Assume the experiment was triggered by the users who started the checkout process. Assume that 10% of users initiate checkout, so that given the 5% purchase rate, half of them complete checkout.

Review: A Numerical Example

A. The e-commerce site with 5% purchase rate. The conversion event is a Bernoulli trial with p = 0.05 .

$$\sigma^2 = p(1-p) = 0.05 * 0.95 = 0.0475$$

$$\delta = 5\% * 5\ \% = 0.25\ \%$$

Sample size = 16*0.0475/0.25%^2=121,600

A Numerical Example

B. Assume the experiment was triggered by the users who started the checkout process. Assume that 10% of users initiate checkout, so that given the 5% purchase rate, half of them complete checkout. What changes?

p (purchase|checkout) = 5%/10% = 50%

$\sigma^2 = p(1-p) = 0.5 * 0.5 = 0.25$ Increase, WHY?

$\delta = 5\% * 50\% = 2.5\%$ Increase, WHY?

Sample size = $16 * 0.25/2.5\%^2$ = 6,400 Decrease, WHY

A Numerical Example

- No effects for those not impacted by the treatment but in the Treatment Group.
- Treatment effects would be tiny without triggering the experiment.

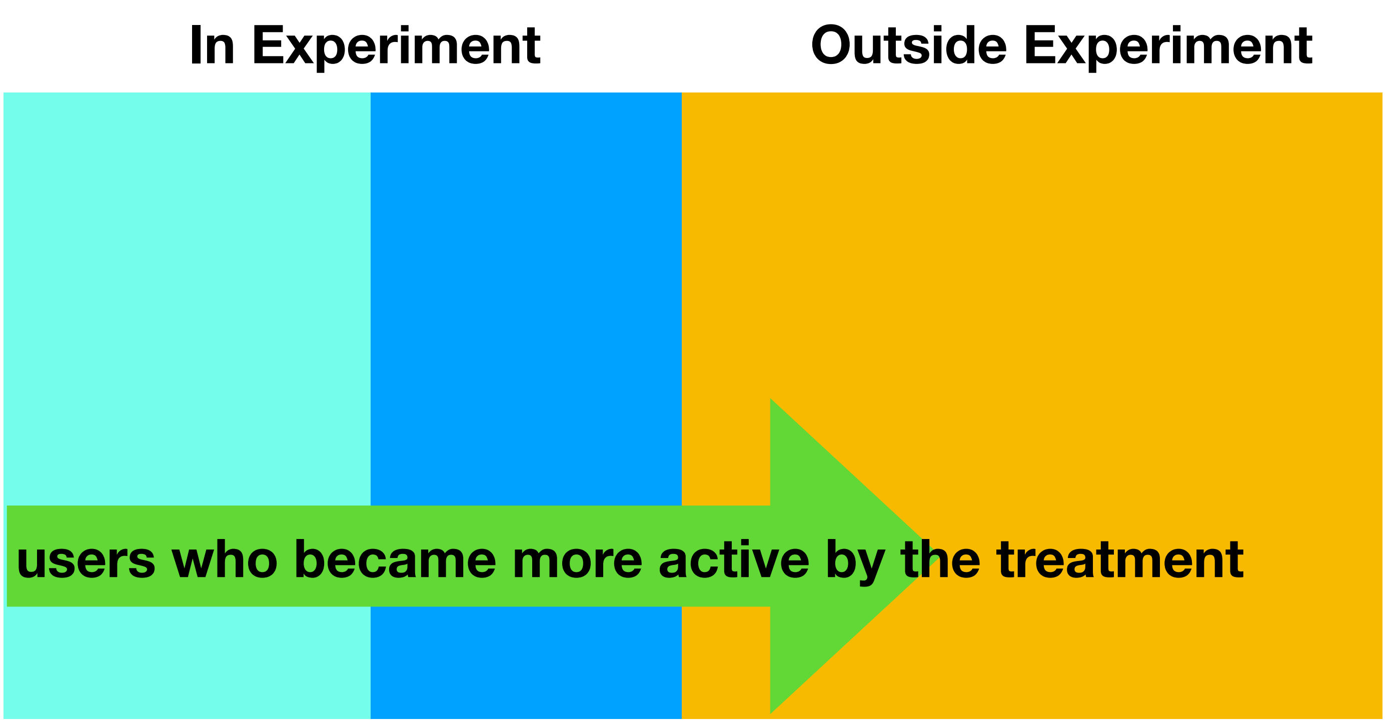
UIN	Treatment Group	Control Group
1	0	0
2	1	0
3	0	0
4	0	0
5	0	0
6	0	0
7	0	0
8	0	0
9	0	0
10	0	0
11	0	0
12	0	0
13	0	0
14	0	0
15	0	0
16	0	0
17	0	0

Example 1: Intentional Partial Exposure

- Run the experiment on a segment of the population, e.g, :
  - Only expose users in some zip codes.
  - Make changes only to heavy users (> 3 visits)
  - Make changes only to inactive users (<1 visits)
- The conditions are well-defined based on the data before the experiment start and
  - The trigger is not the one impacted by the Treatment. WHY?

Example 1: Intentional Partial Exposure

- The trigger is not the one that the Treatment can impact. WHY?
    - Treatment: a feature to improve user engagement
    - Trigger:
        - A: Inactive users: visited less than 1 time during the last day
        - B: Inactive users: visited less than 1 time during the week before the experiment
    - Bias: Exclude users who become active by the Treatment



Example 2: Conditional Exposure

- Suppose the treatment is for users who reach a portion of your products, such as using a feature.
  - A change to a checkout
  - A change to the unsubscribe screens
  - A change to the way the restaurants recommendations displayed in the search results.

Example 3: Coverage Change

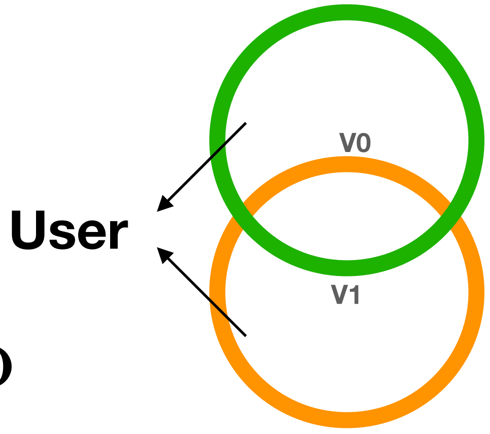
- Suppose that your site offers free shipping to users with more than $35 in their shopping cart.
- Treatment: offer free shipping to users with more than $25 in their shopping cart
- What is the Trigger Condition?
  - Users impacted: [25,35)

Example 3: Coverage Change

- Suppose that your site offers free shipping to users with more than $35 in their shopping cart.
- Treatment: offer free shipping to users with at least $25 in their cart except if they returned any item in the last month.
- Trigger: [25,35) without return in the last month

Example 4: Counterfactual Triggering for Machine learning Models

- Existing: Model V0 to make product recommendations for a user
- Treatment: Model V1
- They are many overlaps in the results between V0 & V1.
- What is the Trigger ?
  - Products recommended by V1 are different from those by V0
  - You must generate the counterfactuals for a user: recommended products by Model V0
  - Include users in the experiments only when:
    - Products recommended by V1 are different from V0



Performance Impact of Counterfactual Logging

- To log the counterfactual Model, both Control and Treatment will execute each other's code.
- However, this may slow the process and make an impact on the performance.
- Run an A/A'/B test
  - A: Control with Model V0
  - A': Control with Model V0 but with a counterfactual logging Model V1
  - B: Treatment with Model V1 but with a counterfactual logging Model V0
- If A' is significantly different from A, this shows an impact of counterfactual logging.
- Consider this impact when evaluating the Treatment Effects.
  - If OEC(A) > OEC(A'), are the treatment effects through comparing A' and B over/under estimated?

Will the Treatment Effects in the Triggered Population Over/Under Estimate the Overall Treatment Effects?

Treatment Effects in the Triggered Population = Treatment Effects on Triggered Population (subset of the population)

Overall Treatment Effect

- Overall Treatment Effect
  - Smaller/Larger than Triggering Treatment Effects?
  - Diluted Impact
- If you improve the revenue by 3% in the triggered population (10% of the population),
  - A: Overall Treatment Effect = 3/10 = 0.3% ?
  - B: Overall Treatment Effect = 0 - 3%?

Example 1

- If you change the check-out process, the revenue increases by 3%.
- What will the overall treatment effects be?
  - You improve both triggered and overall revenue by 3%, and there is no need to dilute it.
- The users excluded from the experiment contribute 0 to OEC.
  - e.g., 90% of users excluded from the experiment contrite 0 to the revenue.
  - The triggered 10% of users contribute X to the revenue.
  - Treatment Effects on Triggered = 3% X/X = 3%
  - Overall Treatment Effects = (3%X+0)/(X+0) = 3%

	Treatment	Control
	0	0
	0	0
	0	0
	0	0
	0	0
	0.2	1
	0	0
Triggered Sample	0	0
	0	0
	1	0.16
Overall-Mean	0.12	0.116
Trigger-Mean	0.24	0.232
Overall-Lift	1.03448276	
Trigger-Lift	1.03448276	

Example 2

- If the change was made to low spenders,
  - low spenders spend 10% of an average user (X).
  - There are 10% low spenders.
- What will the overall treatment effects be?
  - Treatment Effects (lift) = 3%*10% X * 10%N/(100%X*N)=0.03%
- The users excluded from the experiments contrite a lot of more than those involved in the experiment.

Class Exercise

- Assume you changed the algorithm that recommends restaurants in Google search
- This new algorithm improves the clickthrough rate of the recommended restaurants by 10%.
- Restaurant searches contribute 1% of the total searches.
- The clickthrough rate of restaurants is 50% of the average clickthrough rate.
- Calculate the Overall Treatment Effects:

    A. OEC = Click rate for the restaurants' searches

Class Exercise

- OEC = Click rate for the restaurants' searches
  - Treatment Effects on Triggered = 10% X/X = 10%
  - Overall Treatment Effects = (10%X+0)/(X+0) =10%

Class Exercise

- Calculate the Overall Treatment Effects:
    A. OEC = Click rate for the restaurants' searches
    B. OEC = Click rate for the total searches
- Treatment Effects (lift) = 10% *50%*X*1%*N/ 100%X*N=0.05%

Experimentation on Tiny User Segment

- If you improve OEC on a very tiny user segment
  - The lift could be the same for the overall population.
  - Even if the treatment effects are large, the overall treatment effects can be very small.
  - This small change may not matter much for a start-up but can still benefit a mature product.
- The experience gained from the triggered experiment on a tiny user segment may be generalizable to significant features.
  - The algorithms that recommend restaurants may apply to the recommendations of others.