RCTs and Large Experiments (Big Data)

Jared Edgerton

Why Experiments for Big Social Data

Experiments help answer causal questions when observational data are ambiguous.

Big data changes the *scale* and *failure modes*:

- Massive n (small effects become detectable)
- Logging pipelines (measurement + missingness)
- Interference and spillovers (networked settings)
- Multiple testing and adaptive decisions

Experiments as Data Pipelines

An experiment is not just assignment:

- Define unit of randomization
- Assign treatment (and store assignment!)
- Collect outcomes (often via logs)
- Validate measurement
- Analyze under a pre-specified estimand
- Document the entire workflow

Units of Randomization

Common units in big data experiments:

- User / account
- Session / visit
- Content item
- Geographic cluster (ZIP, county)
- Classroom / school / workplace

Choice affects:

- interference risk
- statistical power
- implementation complexity

A/B Tests Are RCTs

A/B tests are randomized experiments:

- random assignment to A vs B
- outcomes measured through platform instrumentation
- analysis focuses on *average treatment effects*

Key question: what is the estimand?

- ITT vs per-protocol
- short-run vs long-run outcomes

Assignment Must Be Stored

If you cannot reproduce assignment, you cannot reproduce the experiment.

Best practice:

- log assignment at the moment of randomization
- keep a stable unit ID (user_id, device_id, cluster_id)
- prevent re-randomization (sticky assignment)

Blocking / Stratification

Goal: reduce variance and improve balance.

Block on variables strongly related to outcomes:

- baseline activity
- geography
- cohort / signup time
- platform (iOS vs Android)

Analysis: estimate treatment effect within blocks, then aggregate.

Cluster Randomization

Cluster randomization is common when:

- interference is likely
- implementation is easiest at cluster level
- treatment operates through shared environment

Examples:

- schools, neighborhoods, workplaces
- network clusters or communities

Inference: use cluster-robust standard errors (or randomization inference).

Interference and Spillovers

SUTVA often fails in big social systems:

- network effects
- information diffusion
- shared resources (platform congestion)

Design strategies:

- cluster or saturation designs
- buffer zones
- measure exposure (treated neighbors)

Power and Practical Significance

With huge n, tiny effects can be statistically significant.

Always report:

- effect size in meaningful units
- uncertainty intervals
- minimum detectable effect (MDE)
- cost-benefit or welfare interpretation (when relevant)

Multiple Testing

Big experiments often generate:

- many outcomes
- many subgroups
- many variants

Controls:

- preregistration / analysis plans
- correction (FDR, family-wise)
- holdout outcomes or holdout time windows
- hierarchical modeling for many estimates

Randomization Checks and Placebos

Before trusting results:

- balance checks on pre-treatment covariates
- A/A tests (two control groups)
- placebo outcomes (should not move)
- instrumentation checks (event definitions stable?)

Noncompliance and Attrition

Common issues:

- users do not receive treatment (delivery failure)
- users do not engage with treatment
- attrition / missing outcomes

Estimands:

- ITT: effect of assignment (default)
- TOT/LATE: effect among compliers (requires assumptions)

Logging and Measurement Error

Outcomes are often *measured* through event logs:

- changes in logging definitions create fake effects
- missing events look like behavior changes
- bots / fraud can contaminate outcomes

Your measurement pipeline is part of the design.

Querying Big Experimental Data

Large experiments typically store raw logs in:

- databases (SQL)
- data lakes (Parquet)
- feature stores

Common workflow:

- raw events -> cleaned table -> analysis-ready dataset
- careful joins on IDs + timestamps

Minimal Analysis: Difference in Means

The simplest estimator: $\hat{\tau} = \bar{Y}_1 - \bar{Y}_0$

Regression adjustment improves precision $Y_i = \alpha + \tau D_i + \gamma^T X_i + \varepsilon_i$

Example: Assignment (Python)

```python
import numpy as np
import pandas as pd

np.random.seed(123)

n = 100000
df = pd.DataFrame({
    'user_id': np.arange(n),
    'baseline_activity': np.random.gamma(shape=2.0, scale=1.0, size=n)
})

# Block by baseline activity deciles
df['block'] = pd.qcut(df['baseline_activity'], 10, labels=False)

# Randomize within blocks (conceptual; production systems store assignment!)
df['d'] = df.groupby('block')['user_id'].transform(
    lambda s: (np.random.rand(len(s)) < 0.5).astype(int)
)
```

Example: Analysis (R)

```r
library(dplyr)
library(estimatr)

set.seed(123)

# df has columns: y, d, block, baseline_activity, cluster_id (optional)

ate_simple <- df %>%
  group_by(d) %>%
  summarize(y_bar = mean(y, na.rm = TRUE))

# Regression adjustment with robust SE (classic)
fit <- lm_robust(y ~ d + baseline_activity + factor(block),
                 data = df,
                 se_type = 'HC2')
summary(fit)
```

Cluster-Robust Inference (R)

```r
library(estimatr)

# Cluster-robust SE (e.g., school_id, county_id, network_cluster)
fit_cl <- lm_robust(y ~ d + baseline_activity,
                    data = df,
                    clusters = cluster_id)
summary(fit_cl)
```

Heterogeneous Treatment Effects (HTE)

Big data encourages subgroup analysis, but:

- subgroups can be numerous
- estimates can be noisy or misleading

Safer approaches:

- pre-specified subgroups
- partial pooling / hierarchical models
- cross-fitting / honest estimation

Reproducibility for Large Experiments

Minimum artifacts:

- assignment table (or reproducible assignment seed + hashing rule)
- raw data snapshot (or immutable query + date)
- analysis-ready dataset build script
- analysis script + outputs
- environment lockfile (renv/requirements)
- logging of parameters and versions

Discussion

- What is the unit of randomization in your research?
- Where would interference likely occur?
- Which outcomes are log-defined and fragile?
- How would you prevent multiple-testing disasters?
- What does it mean for an effect to be 'real' at scale?