# MONTE CARLO POWER ANALYSIS FOR SMALL-SYSTEM A/B TRIALS

Michael D. Ekstrand (*Drexel University*), Daniel Kluver (*University of Minnesota*), Karl Higley (*Gwendydd Research*), Bart P. Knijnenburg (*Clemson University*)

I have a great idea for a better recommender system!

I'll do an A/B test with my small platform's users.

Will the experiment correctly conclude my idea is great?

## The Problem

This is **power analysis**: if an effect is real, how likely is the experiment to detect it?

**Standard solutions** exist for simple experiments (*t*-test of normally-distributed means).

Real recommender systems are **not simple experiments:** feedback is **sparse** and **heavily skewed**, among other challenges — closed-form solutions aren't sufficient.

## The Solution?

Monte Carlo power analysis **simulates** the experiments and their analyses to measure power!

1. Generate synthetic experimental data with known conditions / effects.
2. Apply intended analysis to detect known effects.

## Experimental Setting

- ▷ Daily **personalized e-mail newsletters** of selected AP news articles, sent to each subscriber.
- ▷ Goal: evaluate recommender impact on subscriber news engagement by counting article clicks.
- ▷ **Two-way**, **between-subjects** studies.

## Metrics

- ▷ **Newsletter Click-Any Probability** (CAP): probability subscriber clicks at least one article in a newsletter.
- ▷ **Article Click-Through Rate** (CTR): prob. subscriber clicks any given article.

## Requirements

- ▷ Accurate generative model of user response sufficient to compute statistics for varying experiment effects.
- ▷ Model parameterization we can manipulate to realize different product utility and subscriber activity levels (e.g. increase in user propensity to click).
- ▷ Interpretations of these parameters in terms of classic experiment effect sizes (observed log-odds).
- ▷ Simulator that uses this model to measure power to detect the modeled experimental effects.

## Our Solution!

A **hierarchical zero-inflated Poisson model** based on the negative binomial distribution

- ▷ Basic Poisson: draw clicks/newsletter from Poisson(λ)
- ▷ Negative Binomial: λ is Gamma-distributed
- ▷ Our model: fix λ for each *user* (hierarchical model), draw from (zero-inflated) Poisson.

$$\lambda_u \sim \mathrm{Gamma}(\mu, \sigma)$$

$$n_\ell \sim \begin{cases} \mathrm{Poisson}(\lambda_u) & \text{with prob. } \psi \\ n_\ell = 0 & \text{with prob. } 1-\psi \end{cases}$$

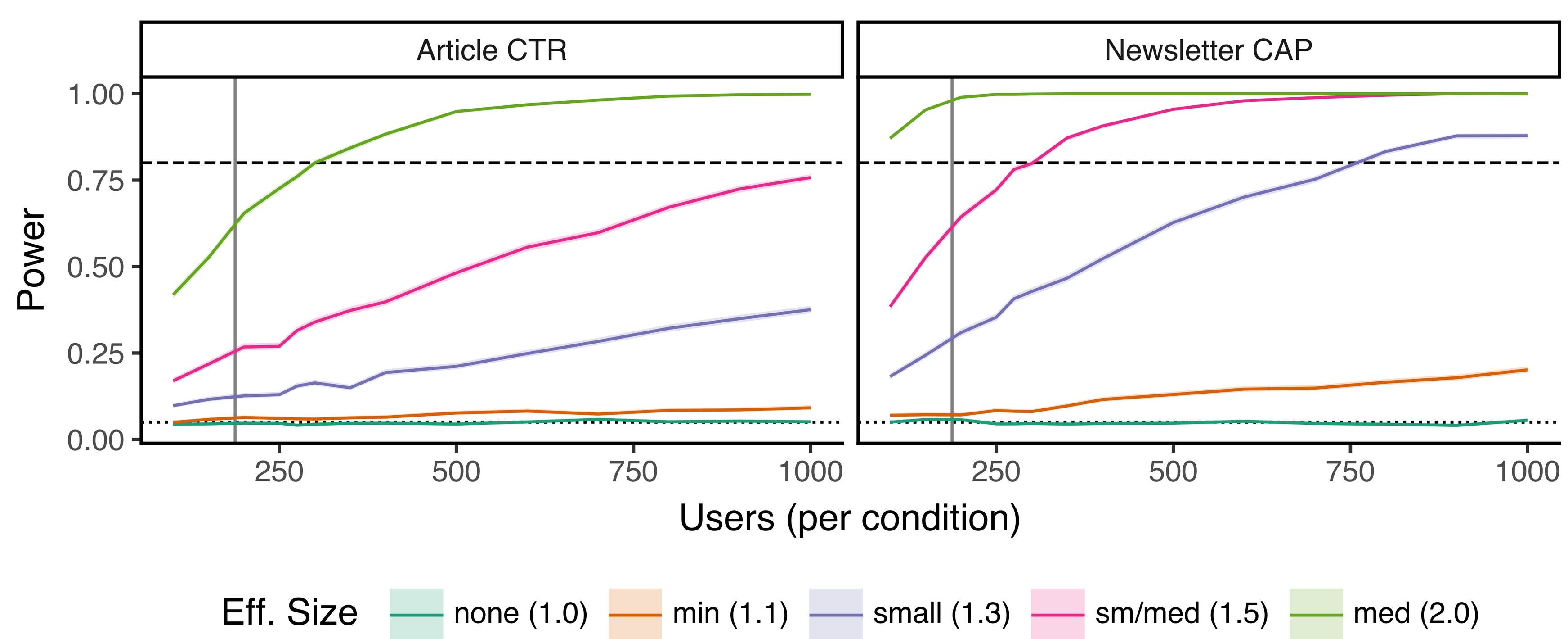
$\lambda_u$ approximately models *per-user "clickiness"* — we can change this to model experimental effects. We assume fixed clickiness variance σ and "look prob." ψ.

**Challenge:** experiment measures *observed* user clicks, but model has latent variable. Solved for relationship between μ and CTR with SymPy.

## Learning Model Parameters

Hamiltonian Monte Carlo Bayesian inference with vague priors to learn free parameters for baseline condition to mimic user behavior in our click logs.

## Output




- ▷ Reliably detecting small effects requires more subscribers than we currently have.
- ▷ Making *usable* power analyses for real-world settings is surprisingly tricky and subtle.
- ▷ Simulation is specific to platform and problem.

Code online if you want to see all the fun details:

`https://mde.one/rspower`