Experimental Design on Networks: Exploring Nonlinear Social Influence

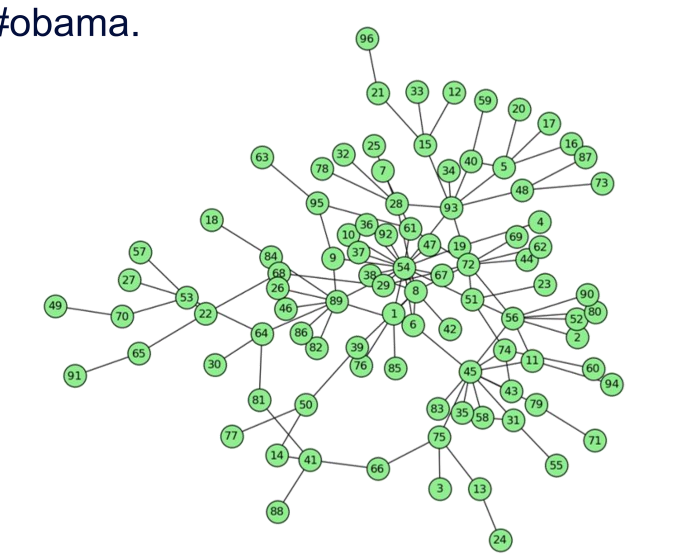
1 Introduction

Businesses run A/B tests every day - a method of comparing the success of two versions of an advertisement by analysing the potential purchasing behaviour of its end users.

On social networks, people influence each other. If your friend saw Advert A and talked about it, that might change how you respond to Advert B - even though you never saw Advert A yourself. This tells us that social influence impacts user response and should be modelled.

Our network

We used a real dataset of 96 X (Twitter) users who mutually retweeted each other using activism-related hashtags such as #fem2, #wethepeople, and #obama.



Rossi & Ahmed (2015)

2 Modelling Social Influence

The more connections a user has, the less any single connection can sway them. Think of it this way:

Given 4 close friends, each friend has strong individual influence

Given 64 acquaintances, each individual has much weaker individual influence

We use the POW-DEG model, which includes a "dampening" parameter, $\lambda \in (0,1]$, to capture exactly this effect. When λ is small, high-degree (popular) users are less easily swayed by any single connection.

The POW-DEG model

$$\underbrace{Y_i}_{\text{Response of unit } i} = \underbrace{\mu}_{\text{Baseline response}} + \underbrace{\tau_{t(i)}}_{\text{Direct treatment effect}} + \underbrace{\gamma_0}_{\text{Network effect of Treatment 0}} \underbrace{\left(\sum_{k=1}^{N} A_{ik}\{u_0\}_k\right)}_{\text{Number of Node } i\text{'s neighbours assigned Treatment 0}}^{\lambda} + \underbrace{\gamma_1}_{\text{Network effect of Treatment 1}} \underbrace{\left(\sum_{k=1}^{N} A_{ik}\{u_1\}_k\right)}_{\text{Number of Node } i\text{'s neighbours assigned Treatment 1}}^{\lambda} + \underbrace{\epsilon_i}_{\text{Random noise } \epsilon_i \sim N(0,\sigma^2)}$$

Neighbour-influence tempering factor: λ

Bui, Steiner & Stevens (2023)

3 Optimising Criteria

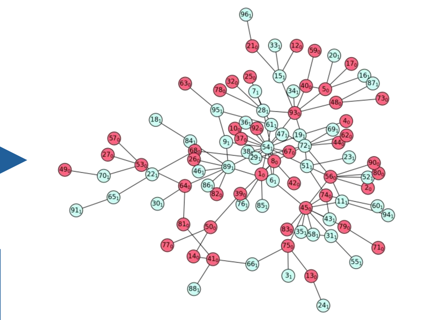
We utilise FIM-based optimality criteria:

- ϕ_1 **-optimality:** minimising the variance of treatment effect contrast estimates
- ϕ_2 **-optimality:** minimising the variance of network effect contrast estimates
- D **-optimality:** minimising the joint uncertainty across all parameters

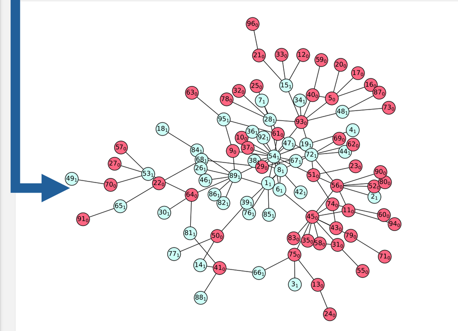
Together with the information matrix for the nonlinear normally-distributed model, defined as

$$\underbrace{I}_{\text{Information matrix}} = \sum_i \underbrace{\sigma^{-2}}_{\text{SD of random noise}} [\nabla\eta(\xi_i,\beta)]\,[\nabla\eta(\xi_i,\beta)]^{\mathrm{T}}$$

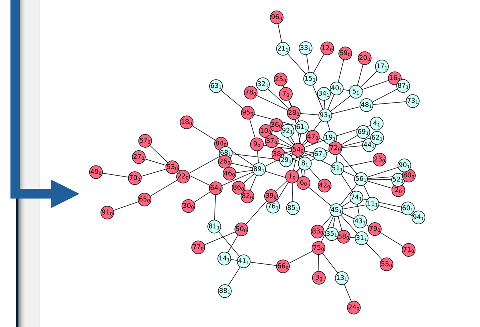
Mean function of the model: $y_i = \eta(\xi_i,\beta) + \epsilon_i$



ϕ_1-optimal design (minimising $Var(\widehat{\tau_0 - \tau_1})$)



ϕ_2-optimal design (minimising $Var(\widehat{\gamma_0 - \gamma_1})$)



D-optimal design (maximising $\det(I)$)

Dealing with Nonlinearity

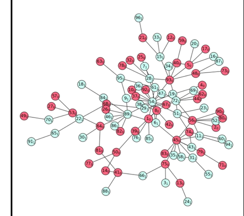
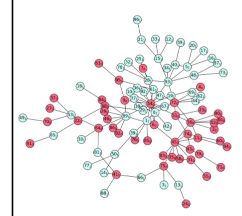
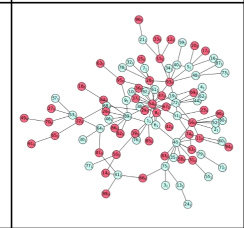
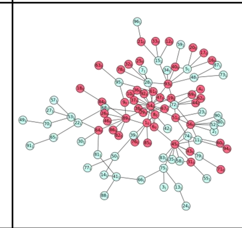
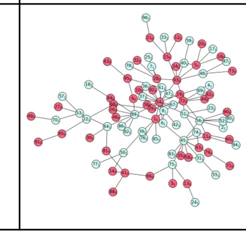
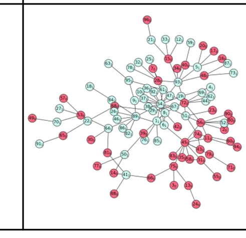
Given the parameter-dependence of the information matrix, we use Monte Carlo approximation to find the average of a chosen optimality criterion over multiple draws from a prior distribution of the parameters:

$$\Phi(\xi) = \mathbb{E}_{\beta\sim p(\beta)}[\varphi(\beta,\xi)] = \int \varphi(\beta,\xi)\,p(\beta)\,d\beta.$$
$$\approx \frac{1}{N}\sum_{j=1}^{N} \varphi(\xi \mid \beta_j)$$

where β_j is one of N draws from a chosen prior distribution, $p(\beta)$, of the parameter(s), β.

4 Point Prior Table

As an alternative to pseudo-Bayesian methods, we may use point priors to find locally optimal designs. The table below shows ϕ_1 - and ϕ_2 - optimal designs, and their respective ϕ_2 - and ϕ_1 - efficiencies for $\lambda = 0.7$, $\gamma_0 \in \{0.3, 0.6\}$ and $\gamma_0 \in \{0.4, 0.7\}$ on the X/Twitter network.

$(\lambda,\gamma_0,\gamma_1)$	ϕ_1 - optimal design	ϕ_2 - efficiency	ϕ_2 - optimal design	ϕ_1 - efficiency
$\lambda = 0.5$ $\gamma_0 = 0.3$ $\gamma_1 = 0.4$		0.57		0.98
$\lambda = 0.5$ $\gamma_0 = 0.3$ $\gamma_1 = 0.7$		0.53		0.98
$\lambda = 0.5$ $\gamma_0 = 0.6$ $\gamma_1 = 0.4$		0.48		0.99

Genetic Algorithm to find Optimal Designs

1) **Initialise:** Sample n candidate designs $\xi_i \in \{0,1\}^N$ from Bernoulli(1/2).

2) **Evaluate & Rank:** Score each design by the chosen criterion

3) **Improve:** Apply coordinate exchange to the top n/2 designs, greedily updating each unit's treatment to maximise the criterion.

4) **Mutate:** Every G-th generation, randomly reassign treatments in select offspring to preserve diversity.

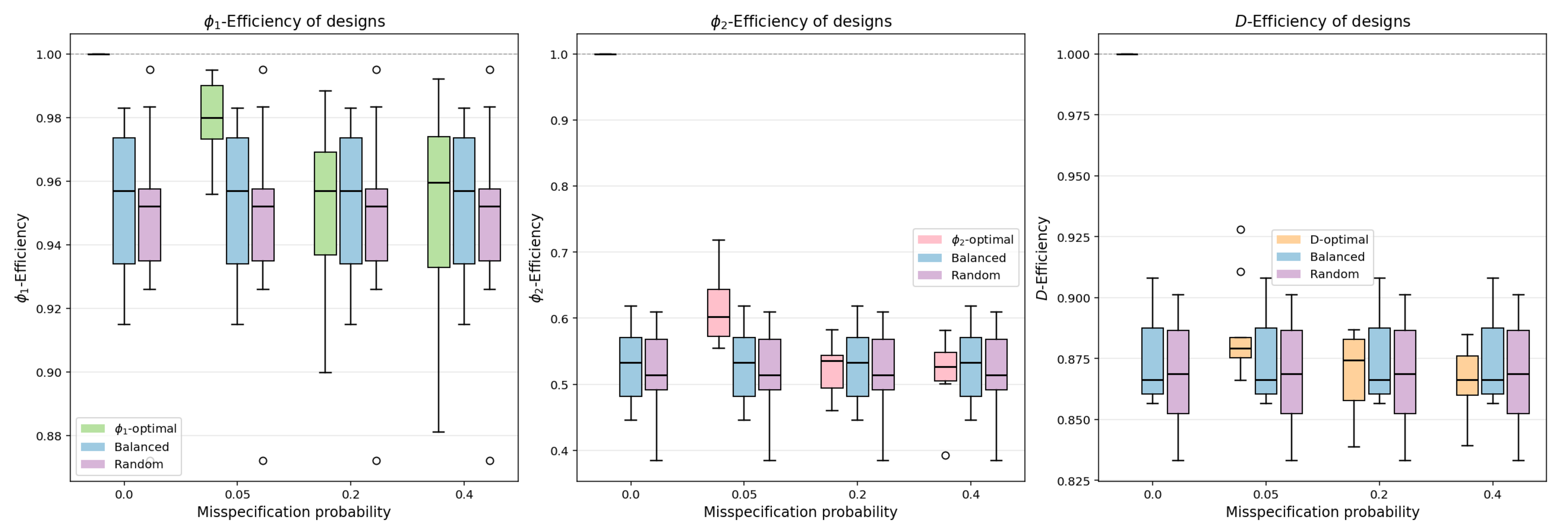
5) **Terminate:** Repeat Steps 2-5 for T generations; return best design.

5 What if the network is misspecified?

We misspecify the network by randomly deleting true edges or inserting false edges according to a chosen probability.

Efficiency of pseudo-Bayesian ϕ_1 -, ϕ_2 -, and D-optimal designs, benchmarked against balanced and random designs, under increasingly perturbed versions of the X/Twitter network. For each misspecification probability, edges are independently added or removed with that probability. Each boxplot summarises efficiencies across 10 independently misspecified networks; for each, a design is optimised on the misspecified network and evaluated against the true network, with efficiency defined relative to the corresponding optimal design on the true network at p = 0.

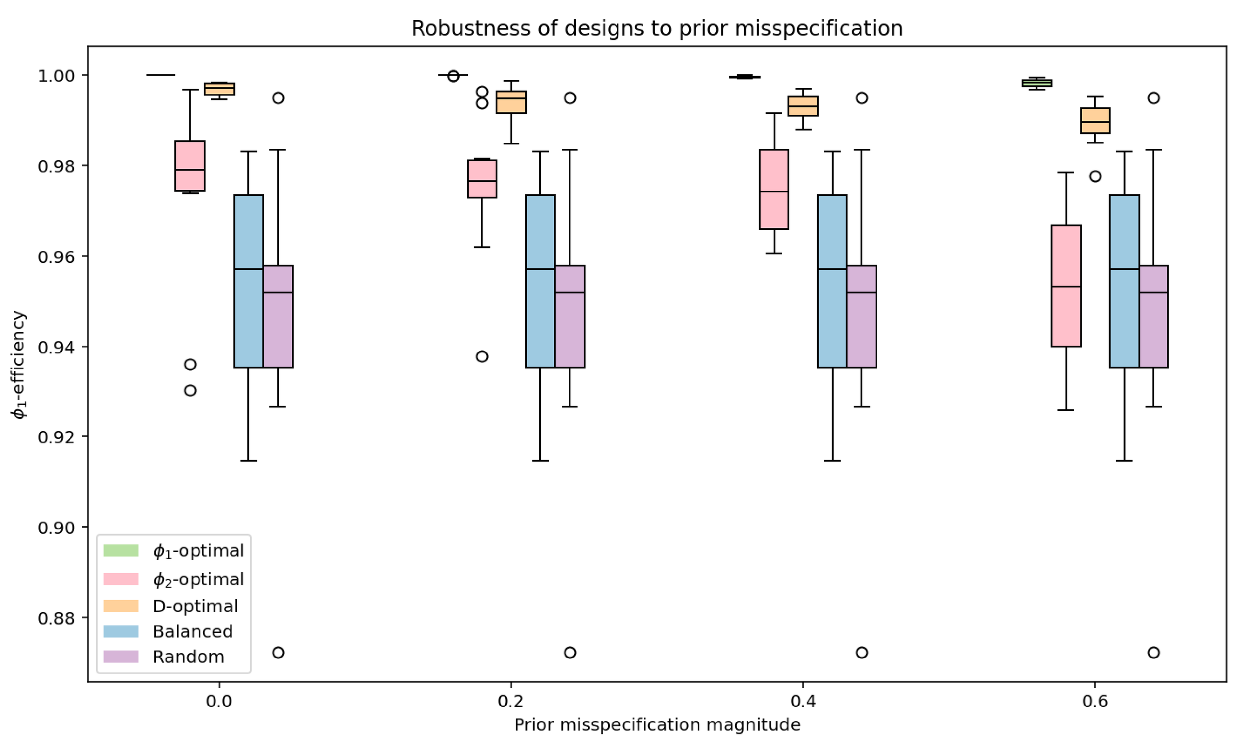
The following priors are sampled from to calculate the pseudo-Bayesian optimality criteria $p(\lambda) \sim N(0.5, 0.1^2)$, $p(\gamma_0) \sim N(0.3, 0.25^2)$, and $p(\gamma_1) \sim N(-0.4, 0.25^2)$



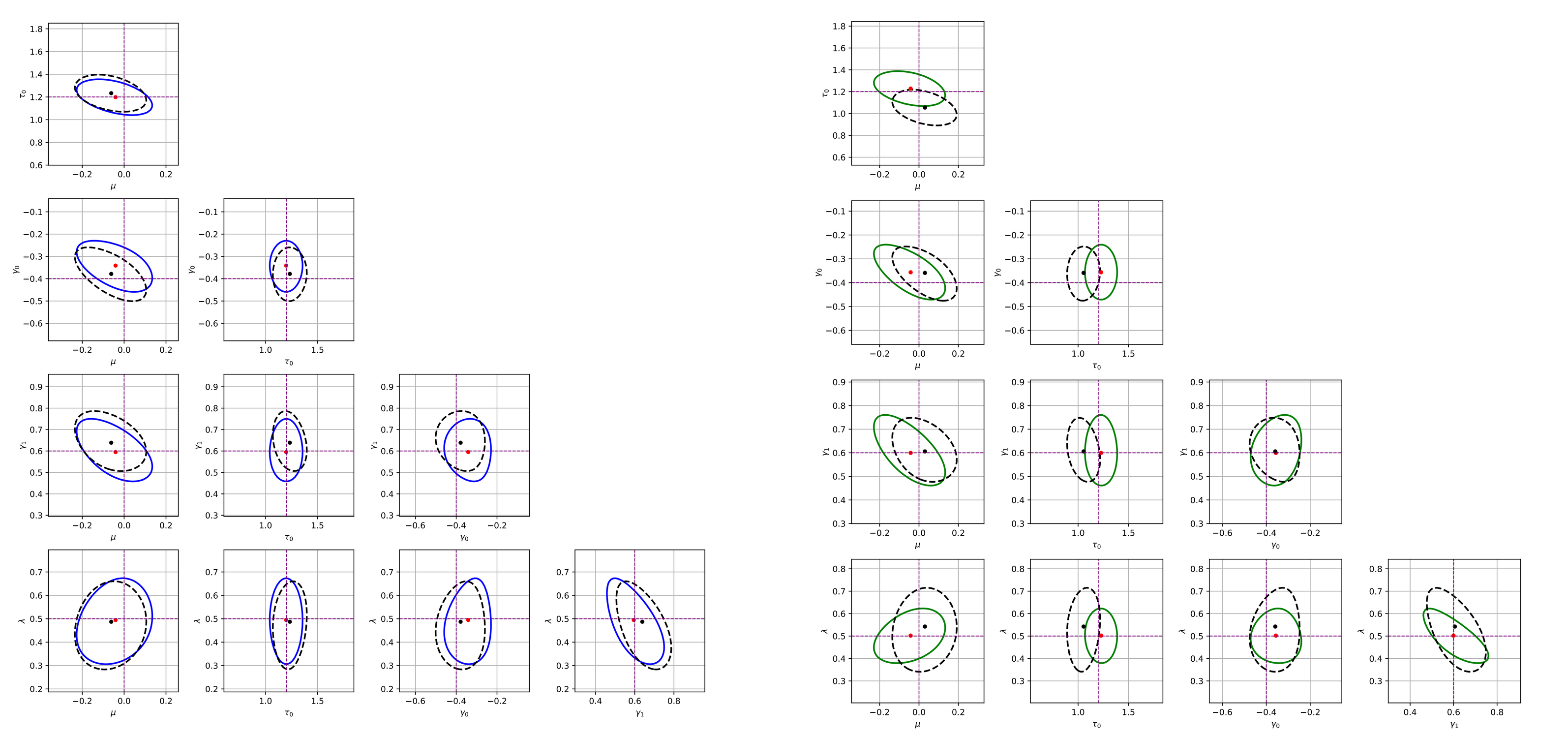
6 What if our prior knowledge is incorrect?

To examine how designs hold up when the prior for λ is misspecified, we shift its mean by $\delta \in \{0, 0.2, 0.4, 0.6\}$, assuming $p(\lambda) \sim N(0.5 + \delta, 0.1^2)$ with $\gamma_0 = 0.3$, $\gamma_1 = -0.4$ fixed.

Each boxplot summarises ϕ_1 -efficiencies across 20 Monte Carlo draws from the misspecified prior, relative to the ϕ_1 -optimal design at $\delta = 0$. The ϕ_1 -optimal design collapses to a line near 1 by construction.



7 Pairwise Parameter Estimate Confidence Regions



Pairwise profile log likelihood confidence regions for the parameters of the POW-DEG model under a ϕ_1**-optimal design** and a ϕ_2 **-optimal design**, compared with such confidence regions constructed using the **balanced design** $u_1 = \{0,1,0,1,\ldots,0,1\}$, illustrated by black-dashed ellipses. True values are $\boldsymbol{\beta_0} = (\mu = 0, \tau_0 = 0.5, \gamma_0 = -0.4, \gamma_1 = 0.6, \lambda = 0.5)$ and $\sigma^2 = 0.1$. The **red** (**black**) dots indicate the NLS parameter calculated from the simulated data under optimal (balanced) design. **Purple dashed lines** indicate the true parameter values. The simulation is performed by generating a single dataset for balanced designs and optimal designs, assuming that $\sigma^2 = 0.1$ using the POW-DEG model with the retweet network defined

8 Open questions!

- How else could I model the misinterpretation of the prior?
- The POW-DEG model captures the network effects of direct neighbours. What would happen if we extended to the effects of second-order neighbours - would the design change substantially?
- Would it be more realistic to allow λ to vary between units?

9 References

Bui, T., Steiner, S.H. and Stevens, N.T. (2023) 'General Additive Network Effect Models', The New England Journal of Statistics in Data Science, 1(3), pp. 342–360. Available at: https://doi.org/10.51387/23-NEJSDS29.

Parker, B.M., Gilmour, S.G. and Schormans, J. (04 2017) 'Optimal Design of Experiments on Connected Units with Application to Social Networks', Journal of the Royal Statistical Society Series C: Applied Statistics, 66(3), pp. 455–480. Available at: https://doi.org/10.1111/rssc.12170.

Rossi, R.A. and Ahmed, N.K. (2015) 'The Network Data Repository with Interactive Graph Analytics and Visualization', in AAAI. Available at: https://networkrepository.com.

Daniel Marshall (daniel.marshall@brunel.ac.uk) | Supervisor: Dr Ben Parker