

# On the Responsible use of Pseudo-Random Number Generators in Scientific Research

Charlie Rahal

LCDS and Nuffield College, University of Oxford

Xi'an Jiaotong, 2024

LEVERHULME  
CENTRE  
FOR  
**DEMOGRAPHIC**  
**SCIENCE**



# What is a PRNG?

**ChatGPT**

A pseudo-random number generator (PRNG) is a deterministic algorithm creating seemingly random number sequences using a specified seed. Unlike true random generators relying on unpredictable physical processes, PRNGs offer repeatability. This trait is beneficial in research, ensuring consistent results with the same seed, essential for reproducibility in various applications.



# Pseudo-Random Number Generation

An invisible source of uncertainty in the scientific record: PRNGs!

# Pseudo-Random Number Generation

An invisible source of uncertainty in the scientific record: PRNGs!

- **Q:** Has anyone ran a program twice, with different results?

# Pseudo-Random Number Generation

An invisible source of uncertainty in the scientific record: PRNGs!

- **Q:** Has anyone ran a program twice, with different results?
- **Q:** Has anyone here ever set a '**seed**'? Which seed?

# Pseudo-Random Number Generation

An invisible source of uncertainty in the scientific record: PRNGs!

- **Q:** Has anyone ran a program twice, with different results?
- **Q:** Has anyone here ever set a ‘seed’? Which seed?
- Maybe to eliminate variation in algorithms with PRNGs?
  - This seems to be the current ‘best practice’ advice.

# Pseudo-Random Number Generation

An invisible source of uncertainty in the scientific record: PRNGs!

- **Q:** Has anyone ran a program twice, with different results?
- **Q:** Has anyone here ever set a ‘seed’? Which seed?
- Maybe to eliminate variation in algorithms with PRNGs?
  - This seems to be the current ‘best practice’ advice.
- We argue this is the **opposite of what we want!**
  - We propose you pre-specify **multiple** (complex) seeds.

# Pseudo-Random Number Generation

An invisible source of uncertainty in the scientific record: PRNGs!

- **Q:** Has anyone ran a program twice, with different results?
- **Q:** Has anyone here ever set a ‘seed’? Which seed?
- Maybe to eliminate variation in algorithms with PRNGs?
  - This seems to be the current ‘best practice’ advice.
- We argue this is the **opposite of what we want!**
  - We propose you pre-specify **multiple** (complex) seeds.
  - Or, alternatively, use a subset of the seeds **we** provide.

# Pseudo-Random Number Generation (Cont.)

- We want need to assess possible variation independent of arbitrary seed choice.

# Pseudo-Random Number Generation (Cont.)

- We want need to assess possible variation independent of arbitrary seed choice.
- An extremely important and scarcely researched problem.

# Pseudo-Random Number Generation (Cont.)

- We ~~want~~ need to assess possible variation independent of arbitrary seed choice.
- An extremely important and scarcely researched problem.
- Pseudo-Random Number Generators occur **everywhere!**

# Pseudo-Random Number Generation (Cont.)

- We ~~want~~ need to assess possible variation independent of arbitrary seed choice.
- An extremely important and scarcely researched problem.
- Pseudo-Random Number Generators occur **everywhere!**
- The variation in estimand can be **huge** (as we'll show).

# Pseudo-Random Number Generation (Cont.)

- We ~~want~~ need to assess possible variation independent of arbitrary seed choice.
- An extremely important and scarcely researched problem.
- Pseudo-Random Number Generators occur **everywhere!**
- The variation in estimand can be **huge** (as we'll show).
- We bring attention to this through multiple types of replications.
  - Simulations, machine learning, NLP, and inferential research.
- Seed variation at times: i.) can be eliminated, ii.) is not a concern, iii.) cannot be eliminated (requires examination).

# The General Premise

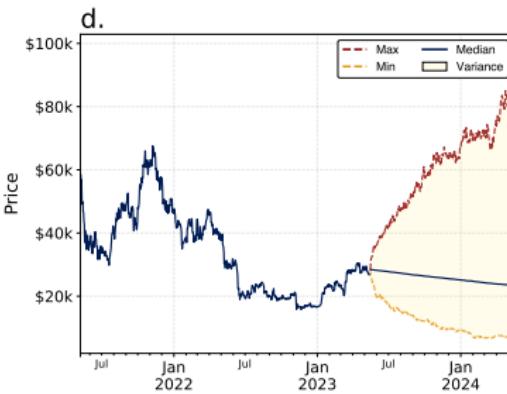
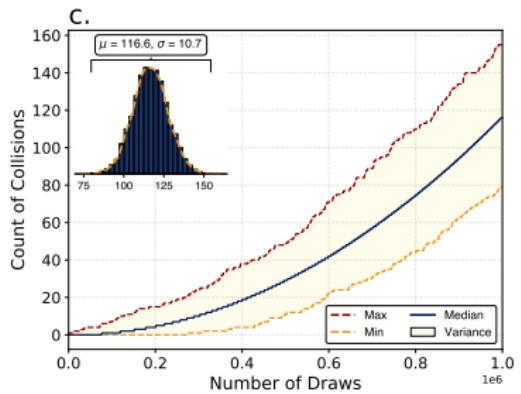
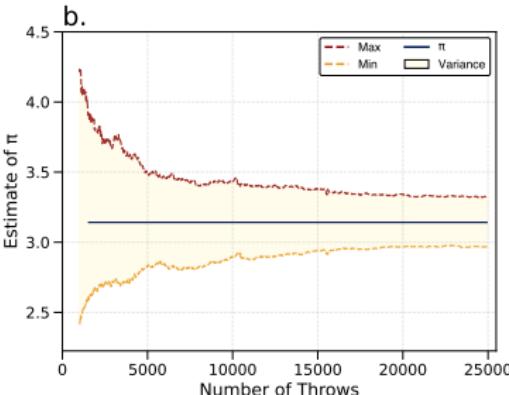
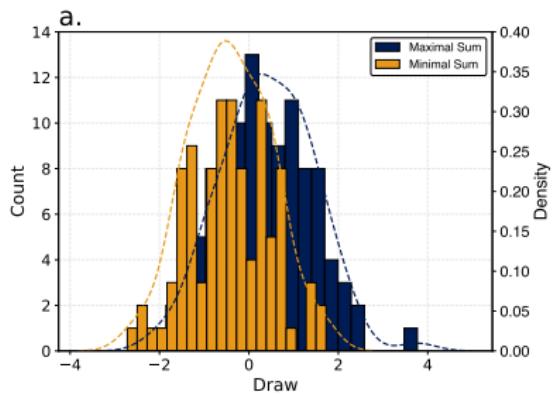
- **Problem Statement:** By setting **one** seed, we ignore the variation of our estimand as a function of how PRNGs were generated: computationally un-intensive, but scientifically dour.

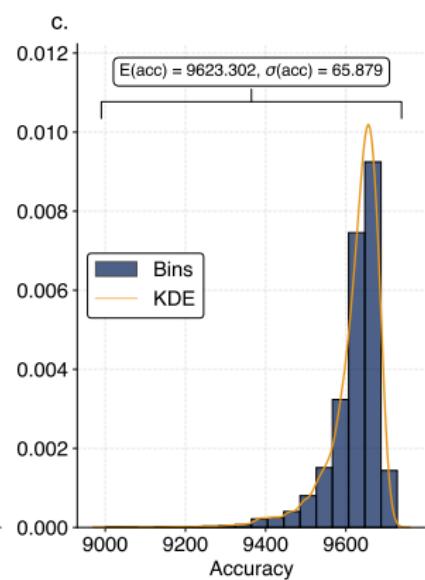
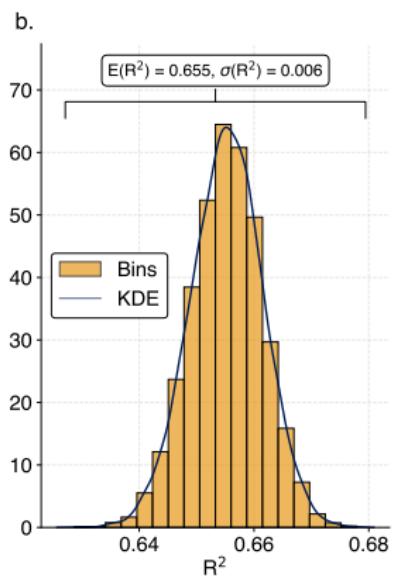
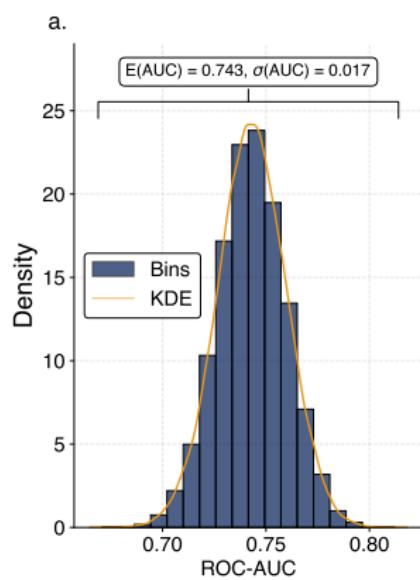
# The General Premise

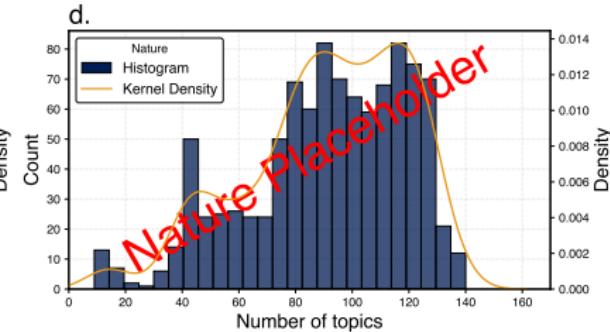
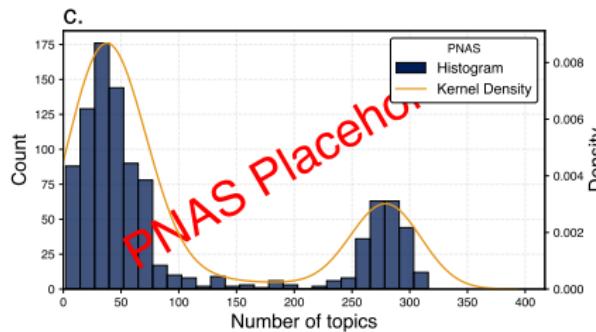
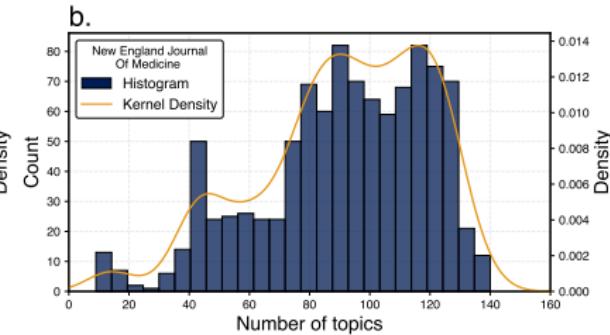
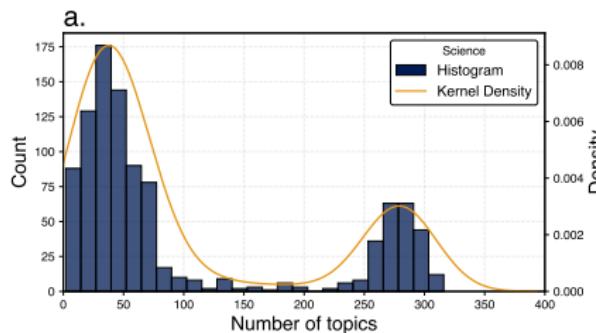
- **Problem Statement:** By setting **one** seed, we ignore the variation of our estimand as a function of how PRNGs were generated: computationally un-intensive, but scientifically dour.
- **Solution:** Visualize the outcome space of a **large number** (10k? 100k?) of seeds simultaneously. This is computationally intensive, but scientifically faithful.

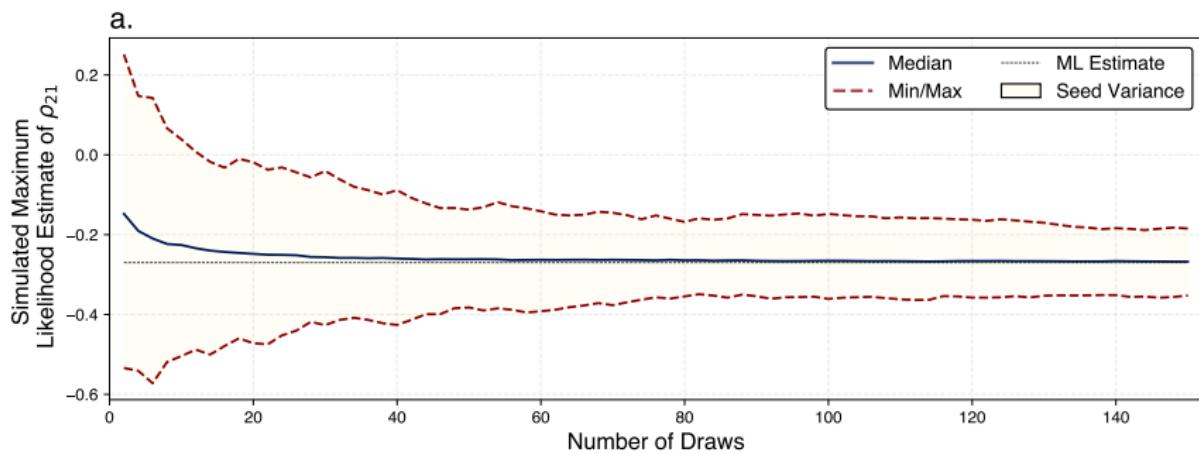
# The General Premise

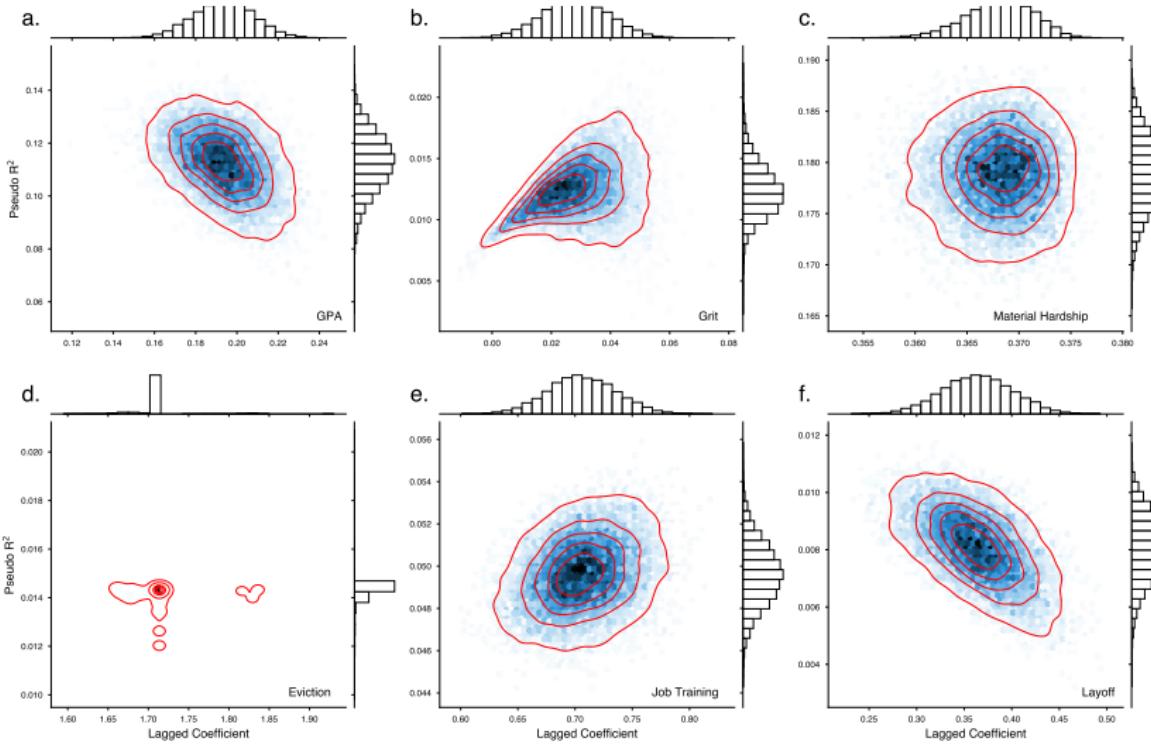
- **Problem Statement:** By setting **one** seed, we ignore the variation of our estimand as a function of how PRNGs were generated: computationally un-intensive, but scientifically dour.
- **Solution:** Visualize the outcome space of a **large number** (10k? 100k?) of seeds simultaneously. This is computationally intensive, but scientifically faithful.
- **Replication:** Similar to specification curves and p-hacking, we can consider whether an original result is in the tail/IQR of the distribution of possible outcome space.











# Concluding Thoughts

- Matt Salganik commented: ‘That sounds great, but expensive!’
  - Expense no-longer prohibitive: everything here ran locally.
- Does anyone have ideas for other types of seed variability?
- Can this be corrective? Index historical seed variability?
- Prospectively: we make available a list of seeds (replication materials), encouraging their use as a pre-specified set.
- **TLDR:** when variation can’t be eliminated, should be visualised.