REPLICATING A GAME THEORETIC MODEL

Mirco Bazzani
FS20
Replication Seminar
Dr. Andrea De Angelis

4/26/2020

Replication of "The Time Efficiency Gain in Sharing and Reuse of Research Data" by Tessa Pronk

Introduction

Reproducing a model from a study which, for itself, wants to calculate the benefits of datasharing seemed more than fitting to me. I've therefore chosen to work on the paper "THE TIME EFFICIENCY GAIN IN SHARING AND REUSE OF RESEARCH DATA" by Tessa Pronk, which produced an estimate of efficiency gains based on Gametheretic modelling.

Even though the replication data was generated for this particular study, I've had to reconcile a DIFFRENT PAPER by the author in order to better understand the mathematic models she used in her R code. I've included some of the formula in chapter 1.1 of this replication paper.

As my main criticism of the original paper lied in its steady state and therefore fixed parameters I've added some random noise to the original code in order to increase it's complexity. This had only little effect on the results, even though most of the basic parameters were either replaced by a random sample or the usage of a random distribution of some form.

I've therefore decided to add a second layer to my reproduction and created a model based on the results on the datasets created in my first replication. This was mainly done to maximize the learning benefits of this replication seminar. My own model was based on the findings of D.V. LANDE ET. AL., which proposed the usage of Markov Chains in order to predict the distribution of information in a social network.

1. GAMETHEORETIC MODEL BY T.PRONK

1.1 – SUMMARY

Tessa Ponk wanted to model the turning point from which the additional expenditure of refurbisching and structuring the data for future users is outweighed by the efficiency gain from researchers not needing to gather the data on their own. She accomplishes this by assuming that the produced papers per year are indicating how efficient the scientific community really is.

In short, the main variables are:

1. How long it takes to *write a paper*
2. How long it takes to *gather data*
3. How long it takes to *prepare* the data for *reproduction*
4. How easy the shared data is *understandable by other* researchers

The original paper assumes that point 1 and 2 are constant, and only the latter vary in their dimensions. I've changed this so that both 1 and 2 are normally distributed values. The parameters found in the original paper were chosen as means, and a standard deviation of 10% around those values. You find a full list of all parameters in Table 1.

COMPARISON OF VARIABLES BETWEEN MY REPLICATION AND THE ORIGINAL PAPER

Parameter	Original Definition	Original Value	Changes in Replication
t_a / t_{pap}	Time-cost to produce a paper (days per year)	47/365	A normal distribution with a sd of 10% around the mean of 47
t_d / t_{dat}	Time-cost to produce a dataset (days per year)	73/365	A normal distribution with a sd of 10% around the mean of 73
t_c / $timeCostShare$	Time-cost to prepare a dataset for sharing	0.1	Defined by the function; either 1 or 15 days
- / $timeCostSearch$	Time-cost for searching an appropriate Dataset (not in original paper)	-	A Uniform distribution between 0.5 and 3 days
t_r / $timeCostKnow$	Time-cost to prepare a dataset to reuse	0.05	Defined by the function; either 1 or 15 days
qx	Decay rate of shared datasets	0.1	A Uniform distribution between 0.1 and 0.2
b	Citation benefit (sharing researcher)	0	-
f	Probability to find an appropriate dataset	0.00001	Not changed, as the parameter is different in paper and original model
Y_s	Proportion of sharing researchers	.5	Defined in the function, not changed

These parameters were then inserted into the base formula of the model, which calculates the pool of shared datasets X_{mb}.

$$X_{mb} = -\big(qx(t_a + t_c + t_d) - Y_{sf}\big) + \frac{\sqrt{(qx(t_a + t_c + t_d) - Y_s)^2 - 4(qx * f(t_a + t_c + t_r + TimeCostSearch)) * (-Y_s)}}{2\big(qx * f(t_a + t_c + t_r + TimeCostSearch)\big)}$$

Furthermore, the model calculated the **Effect Rate** of every published paper by including the possible citations per paper. The formula was split up into 3 steps:

1. The *time spent* writing a paper, defined by:

$$T_{s/ns} = t_a + \frac{t_d}{1 + f * X} + (t_r - \frac{t_r}{1 + f * X}) + t_c$$

2. The number of publications per Researcher P:

$$P_{s/ns} = \frac{1}{T_{s/ns}}$$

3. The actual *Effect Rate*, defined by the publications, multiplied with the citations per paper c and the citation benefit b:

$$E_s P_{s/ns} = P_{s/ns} * c * (1 + b)$$

The last formula can be ignored, as Tessa Pronk sets the Effectrate equal to the published papers. My second model takes this Effect rate and makes further calculations with it, so we will keep this in mind. I've slightly changed the model, so that the Effect Rate varies unifromly around the Papers / Researcher by a factor of 0.9 - 1.1.

1.2 - RESULTS

Table 2 shows us the definition of each variable that was calculated inside my replication. The highlighted parameters are used for further calculations in my second model.

DEFINITONS OF THE RESULTS OF THE MODEL BY TESSA PRONK

Parameter	Definition
pap_research	Calculated by an exponential distribution with a rate of $t_d + t_a$
pub_research	The inverse of pap_research, calculated by $1/pap_research$
effect_rate	The EffectRate, originally equal to pap_research, in my Paper `` `EFFECT_RATE <- PAP_RESEARCH * RUNIF(10000,.9,1.1)` ``
share	$0/1$ coded variable which defines for each line if the corresponding researcher shares it's work. Random sample based on the ratio of Y_s
use_opp	A vector, calculation the chanche of finding an appropriate Dataset and applying it on every pap_research by a the factor $(1 - (1/(1 + (f * X_m b))))$
benefits	The time saved by using a premade dataset $t_a(useOpp/papResearch)$
reusecost	The time lost by trying to understand a premade dataset $TimeCostKnow(useOpp/papResearch)$
cost	The time costs of sharing a dataset (only for sharing scientists) $TimeCostShare * share$
time	$(t_a + t_d - benefits + cost + reusecost) * (pubResearch/(t_a + t_d)$
Publ	$1/time$
impact	$impact = Publ$

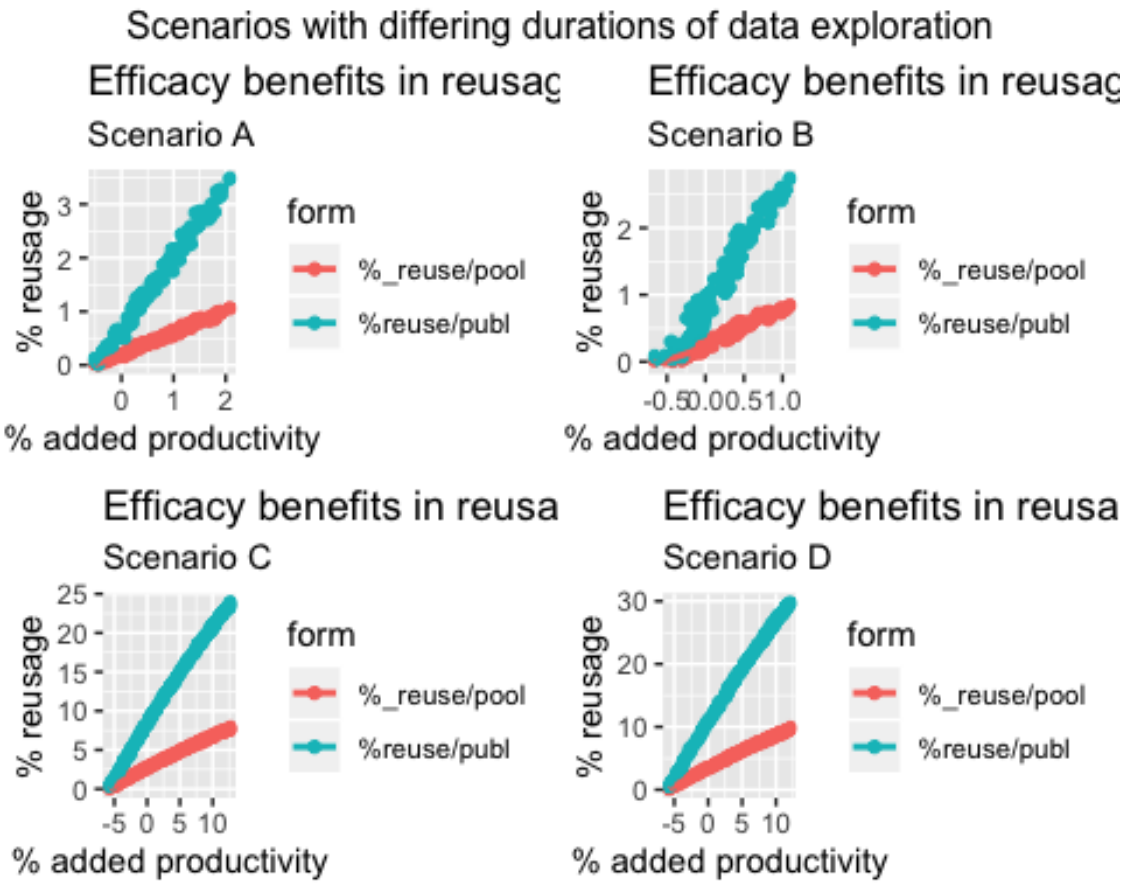
I've chosen not to alter the base model any more than needed and focus on producing a very own model. The plots below are therefore the result of the above-mentioned equations and are based on 4 different scenarios.

Scenario A, where a Researcher takes only 1 day to share and get to know the data.

Scenario B, where a Researcher takes only 1 day to share, but 15 to get to know the data.

Scenario C, where a Researcher takes 15 days to share, but only 1 to get to know the data.

Scenario D, where a Researcher takes 15 days to share and get to know the data.



The models show that it takes increasingly more people which reuse the data to reach the 0-point of added productivity, for the scenarios *C* and *D*, where preparing the data for sharing takes up to 15 days need up to 10% of data reusage in the population.

What astonished me the most was that changing the duration of data sharing and / or getting used to the data does not alter the resulting plot that much. What alter them the most is a correction-variable Tessa Pronk adds to her function, even though this addition is never further explained.

We can see some effect of this variable in Scenario A and B, where the individual points of the plot are not in a perfectly linear distribution. Further increasing the correction variable only adds to this deviance from linearity. I have no explanation for this effect, and will therefore jump right to my learnings.

1.2 - Learnings

I want to start with the complexity of the paper. As I am not the most patient person, I've started replicating the model way before I've really understood the mathematics behind it This cost me way more time than it should have, as I was not familiar with all of the parameters. But what I've found in the end was that neither of the papers provided by Tessa Pronk fully explain the cumputation she has done for her model - you need to look at the code in order to understand her results and how she got to them.

This not only showed me the importance of replication, but also how intransparent some research papers possibly can be.

Secondly, I was able to find a minor error in Tessa Ponks base code which I first had to fix in order for the model to work. While only being a minor porblem, it showed that even the original authors had issues with the sheer amount of parameters they created during their computation, which resulted in them naming one variable wrong and finally the code leading to an error.

This abundance of variables, combined with my impatience led to errors on my side as well, as many parameters like t_a, t_d and t_r nearly sound and look the same. I therefore mixed up two of them whilst rewriting the code, what led my results to be absolutely nonsensical.

Nevertheless, working on the code and trying to improve it - even though i deleted mos of my changes in the end - thaught me a lot about the workings of R.

2. Markov Chains

2.1 - Summary

As I was not satisfied with my own replication, I've searcher for ways to alter the results in a new model that was not based on a replication. Instead of replicating code this allowed my to tinker on my very own solutions.

As I already had created a sample dataset which included results for the model of Tessa Ponk, I chose to base my model on these results. The parameters that were used for my further computation were $papResearch$, $pupResearch$, $share$, $EffectRate$ and $impact$.

What I wanted to model was the approximate amount of citations per paper that would occur in my sample population, and if sharing the data would have a visible effect like boosting or reducing citations per paper. The method I used was a watered down form of a **Markvo Chain**, where each paper was given an energy level based on its initial **Effect Rate** which randomly increased or decreased during the upcoming 100 days. Based on Tessa Pronks paper, I assumed that the Effect Rate and it's popularity was equal to the number of Citations.

The model works as follows:

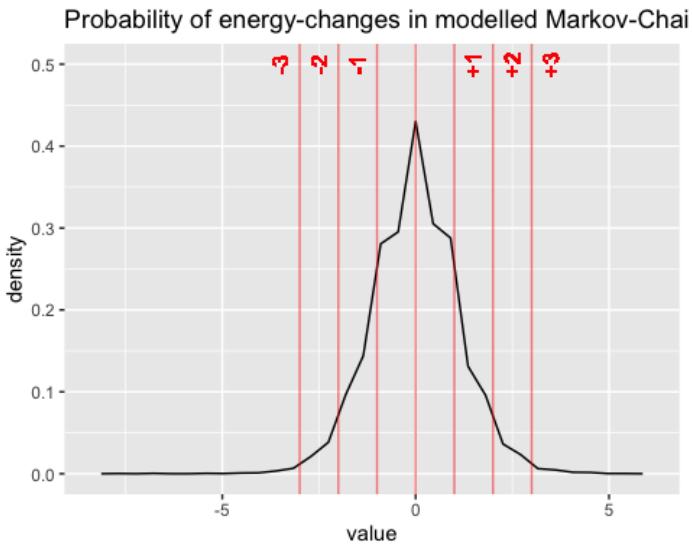
- Each case p_i in the population of $n = 10'000$ gets assigned an *Enegry Level* E. This Enegy Level is equal to its Effect Rate.

$$E = effectRate$$

- I then started a random walk of $j = 100$ steps. In each step, a randomly chosen amount gets added or substracted to the Energy Level of each case. If the Energy Level of a Variable fell below 0, it was reset to 0. The changes of the Energy Level were based on a t-Distribution with a mean of 0 and a sd of 10. This leaves us with the following density for each incrementation:

Energy change	Density
+3	.01
+2	.06
+1	.25
+/- 0	.42
-1	.25
-2	.06
-3	.01

The corresponding plot is seen right below.



- Furthermore, for every step, the Energy Level decreases by a uniformly distributed parameter dr, that fluctuates between 0.05 and 0.15. This **Decay Rate** assumes that the novelty of a scientific paper decreases over time.
- As I want to take into account the **"virality"** of popular papers (and to counterreact the decay rate), popular papers that were cited over 15 times get an energy boost of $E_i >= 15 \Rightarrow (E_i + 1.2)$.

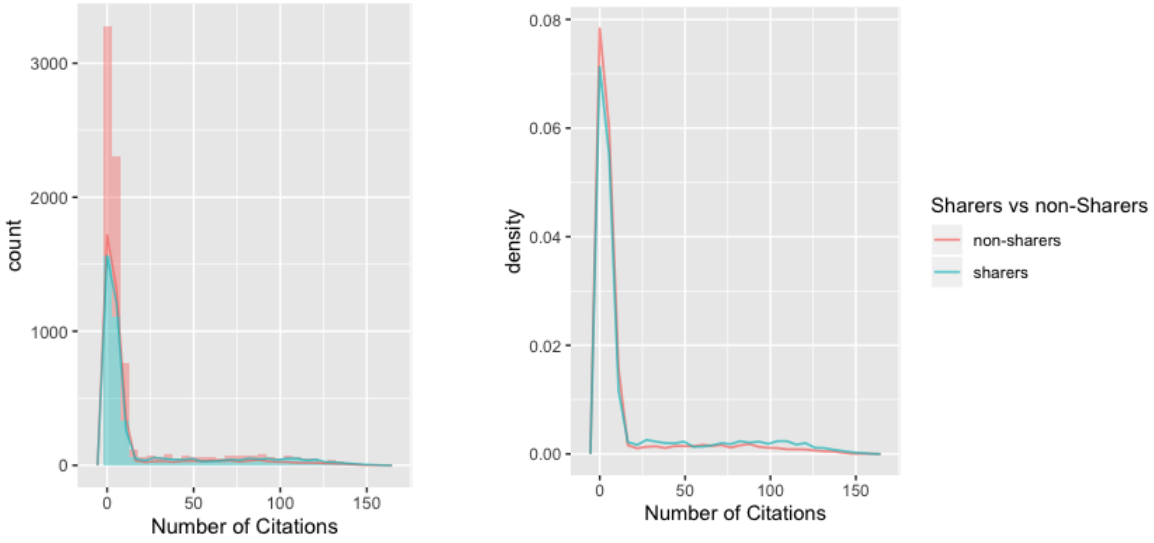
- And lastly, I assumed that the distribution of the **replication data for viral papers** adds to their total citation count. This is why $E_i >= 10 \Rightarrow (E_i + effect)$ for every sharing researcher. To geth the variable $effect$, I normalized the Effect Rate for every Researcher to a number between 0 and 1 using the formula

$$effect = \frac{effectRate_i - effectRate_{min}}{effectRate_{max} - effectRate_{min}}$$

Each computation is done for each of the n cases, and repeated j times.

2.2 – Results

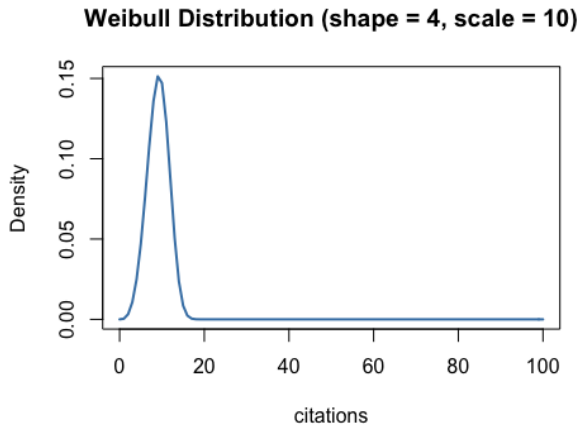
The results of this Random Walk are seen below:



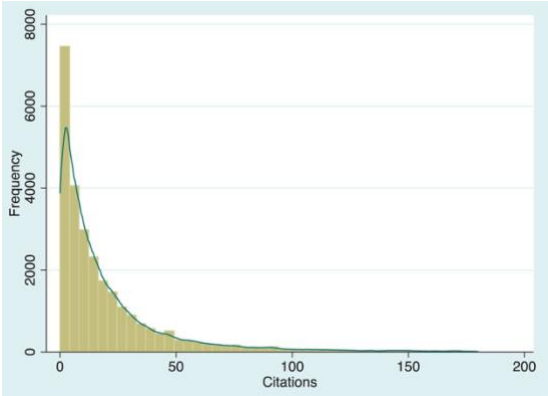
Even though the sharers got an additional bonus for virality, the frequency plots nearly overlap. This seems to show that the random increase mapped by the t-distribution has the stronger effect on the population.

Although I would not call my model scientifically accurate by any means, it does show the characteristics of a Weibull-Distribution that - *according to the paper this model is based on* - occurs in the real world, too. But I have to add that the original paper was based on interactions on twitter,

and was built around completely different assumptions concerning the changes in energy levels per step.



What I found to be the most interesting point was that my model showed a remarkable resemblance to the actual distribution of citations found in academia shown for example by THIS PAPER.



Histogram of citations in legal papers

2.3 - LEARNING

Working on my own model was something I have never done to this extent before. I would have liked to spend more time on it in order to remove the "playfullness" and transform it into real academic work, but for the time being I am really fond of the learning effect I've had this far.

Form creating markdown documents to plotting up to modelling, the replication seminar has surely thaught me a lot and showed me that the potential of R far exceeds what I've learned this until this point.

The code I've used for my own model is pasted below, and can be found in the shared project file under the name "markov_model_bazzani.R"

```
SET.SEED(1234)

SHORT_LIST <- RATE_LIST %>%
  MUTATE(IMPACT_R = ROUND(EFFECT_RATE)) %>%
  SELECT(SHARE,IMPACT_R)

EFFECT <- RATE_LIST %>%
  MUTATE(EFFECT = 1.5*(((RATE_LIST$EFFECT_RATE - MIN(RATE_LIST$EFFECT_RATE)) /
                                                (MAX(RATE_LIST$EFFECT_RATE) - MIN(RA
TE_LIST$EFFECT_RATE))))) %>%
  SELECT(EFFECT)

SY <- SHORT_LIST %>% FILTER(SHARE == 1)
SN <- SHORT_LIST %>% FILTER(SHARE == 0)

SLY <- AS.TIBBLE(SY)
SLN <- AS.TIBBLE(SN)
I <- 1
MAX <- 100
DR <- RUNIF(10000,0.05,0.15)

FOR (I IN  1:MAX) {

  DAYS <- PASTE("DAY", 1:MAX, SEP = "")
  #Q <- MATRIX(DAYS,1,MAX)
  #T <- AS.CHARACTER(Q[1,I])

  SLN <- PRINT(SLN %>% ADD_COLUMN(!!(DAYS[I]) := 0))

  MUTATE(SLN, DAY1 = IMPACT_R)

  IF (I >= 1){

    SLN[2+I] <- SLN[(2+I)-1] + ROUND(RT(N = 10000, DF = 10), 1)
    SLN[2+I] <- REPLACE(SLN[2+I], SLN[2+I] < 0, 0)
    SLN[2+I] <- SLN[2+I] + IFELSE(SLN[[2+I]] >= 15, 1.2, 0)
    SLN[2+I] <- SLN[2+I] - DR

  }

  PRINT(SLN)

}
```

```
FOR (I IN  1:MAX) {

  DAYS <- PASTE("DAY", 1:MAX, SEP = "")
  #Q <- MATRIX(DAYS,1,MAX)
  #T <- AS.CHARACTER(Q[1,I])

  SLY <- PRINT(SLY %>% ADD_COLUMN(!!(DAYS[I]) := 0))

  MUTATE(SLY, DAY1 = IMPACT_R)

  IF (I >= 1){

    SLY[2+I] <- SLY[(2+I)-1] + ROUND(RT(N = 10000, DF = 10),1)
    SLY[2+I] <- REPLACE(SLY[2+I], SLY[2+I] < 0, 0)
    SLY[2+I] <- SLY[2+I] + IFELSE(SLY[[2+I]] >= 15, 1.1, 0)
    SLY[2+I] <- SLY[2+I] + IFELSE(SLY[[2+I]] >= 10, AS.NUMERIC(SAMPLE_N(EFFECT,1)
), 0)
    SLY[2+I] <- SLY[2+I] - DR

  }

  PRINT(SLY)

}


# COMBINING THE RESULTS OF SLN AND SLY ----------------------------------
SLN[102] <- REPLACE(SLN[102], SLN[102] < 0, 0)
SLY[102] <- REPLACE(SLY[102], SLY[102] < 0, 0)

TBL1 <- HEAD(SELECT(SLN, DAY100), 4000)
TBL2 <- HEAD(SELECT(SLY, DAY100), 4000)
TBL2 <- RENAME(TBL2, DAY100_2 = DAY100)
JOIN <- BIND_COLS(TBL1, TBL2)
JOIN <- GATHER(JOIN, DAY100, DAY100_2, KEY= "SOURCE", VALUE = "CASES")
JOIN <- JOIN %>% MUTATE(CASES = ROUND(CASES))
JOIN <- AS_TIBBLE(JOIN)

# CREATING PLOTS ----------------------------------


JOIN %>% GGPLOT(AES(X = CASES, FILL = SOURCE)) +
  GEOM_HISTOGRAM(ALPHA = .4, BINWIDTH = 5) +
  GEOM_FREQPOLY(MAPPING = AES(COLOR = SOURCE), SIZE = .5, ALPHA = .7) +
  LABS(X = "NUMBER OF CITATIONS", FILL = "SHARERS VS NON-SHARERS",
       COLOUR = "SHARERS VS NON-SHARERS") +
  SCALE_FILL_DISCRETE(NAME = "SHARERS VS NON-SHARERS", LABELS = C("NON-SHARERS",
```

```
"SHARERS")) +
  SCALE_COLOR_DISCRETE(NAME = "SHARERS VS NON-SHARERS", LABELS = C("NON-SHARERS",
 "SHARERS"))



JOIN %>% GGPLOT(AES(X = CASES, Y = ..DENSITY.., COLOR = SOURCE)) +
  GEOM_FREQPOLY(ALPHA = .7, SIZE = .6) +
  LABS(X = "NUMBER OF CITATIONS") +
  SCALE_COLOR_DISCRETE(NAME = "SHARERS VS NON-SHARERS", LABELS = C("NON-SHARERS",
 "SHARERS"))

SUMMARY(JOIN)
```