

1 Replication Equation

1.1 Solve the ODE

$$\frac{dr}{dt} = u(1 + z(1 - \frac{r(t)}{n})) - cr(t) \quad (1)$$

$$= u(1 + z) - r(t)(c + \frac{uz}{n}) \quad (2)$$

This is a linear ordinary differential equation and we know that $r(t)$ differentiates roughly into itself. So let's try a solution of the form ae^{bt} .

$$r(t) = ae^{bt} + d \quad (3)$$

$$\frac{dr}{dt} = bae^{bt} \quad (4)$$

$$= b(r(t) - d) \quad (5)$$

$$= br(t) - bd \quad (6)$$

$$\implies b = -(c + \frac{uz}{n}) \quad (7)$$

$$bd = -u(1 + z) \quad (8)$$

$$\implies d = \frac{u(1 + z)}{c + \frac{uz}{n}} \quad (9)$$

$$\implies r(t) = ae^{-t(c + \frac{uz}{n})} + \frac{un(1 + z)}{cn + uz} \quad (10)$$

Initial conditions give us

$$r(0) = a + \frac{un(1 + z)}{cn + uz} \quad (11)$$

$$a = r(0) - \frac{un(1 + z)}{cn + uz} \quad (12)$$

$$\implies r(t) = (r(0) - \frac{un(1 + z)}{cn + uz})e^{-t(c + \frac{uz}{n})} + \frac{un(1 + z)}{cn + uz} \quad (13)$$

1.2 Differentiate the Solution

Just in case.

$$r(t) = (r(0) - \frac{un(1+z)}{cn+uz})e^{-t(c+\frac{uz}{n})} + \frac{un(1+z)}{cn+uz} \quad (14)$$

$$\frac{dr}{dt} = -(c + \frac{uz}{n})(r(0) - \frac{un(1+z)}{cn+uz})e^{-t(c+\frac{uz}{n})} \quad (15)$$

$$= -(c + \frac{uz}{n})(r(t) - \frac{un(1+z)}{cn+uz}) \quad (16)$$

$$= u(1+z) - r(t)(c + \frac{uz}{n}) \quad (17)$$

2 Popularity and Query Distributions

2.1 Start from Popularity Distribution

We have a torrent popularity distribution, $P(T)$, tells us the probability that a torrent is owned by a given number of nodes:

$$P(T = t) = (a-1)t^{-a}, a = 1.63 \quad (18)$$

The CCD of this distribution will tell us the probability that a torrent is owned by x or more nodes:

$$CCD(T = t) = t^{1-a} \quad (19)$$

Leaving the model and using the empirical data we can say how many node-torrent pairings we observed where the torrents were owned by x or more nodes:

```
nodes = [1 2 3 ... 34391]
frequency = [397121 202599 107501 ... 1]
i = <index of x in nodes, i.e. nodes[i] = x>
total = nodes*frequency'
satisfiable = nodes[x:end]*frequency[x:end]'
percent = satisfiable / total
```

To do the same calculation for the model we need solve:

$$\int_x^{\inf} (a-1)t^{-a} * total * t dt = total * (a-1) \int_x^{\inf} t^{1-a} dt \quad (20)$$

I'm pretty certain this is just inf for $a < 2$.

In order to determine what proportion of the queries per hour would be satisfiable we need to know how they're distributed over the torrents. If we assume that query distribution matches torrent distribution then we have $P(Q = q) = (a-1)q^{-a}$, the probability that a torrent has q queries performed for it. If we knew the number of torrents that existed per hour, we could use $P(Q = q) * \sum(\text{torrents})$ to give us the number of torrents that have q queries.

Multiply by q to get the total number of queries performed at this rate. We can similarly use the $CCD(Q=q)$ to know how many torrents or queries were performed at a rate of q or more per hour.

We know how many queries there are per hour (6,395,604), so we know that:

$$\int_1^{\infty} (a-1)q^{-a} * \sum(\text{torrents}) * q dq = 6395604 \quad (21)$$

Unfortunately this means the number of torrents has to be infinitely close to 0 in order to balance the infinite integral.

This is as close as I can get to working out a query success per hour figure.

2.2 Start from Query Distribution

Let's assume that the query distribution over torrents follows a power law: $P(S = s) = (b-1)s^{-b}$, the probability that a query is part of a group of s queries for a torrent. e.g. pick a query at random, what's the probability that it is one of s queries for a torrent. We can work out the $CCD(S = s)$ as before, and, if we multiply up by the total number of queries per hour we can calculate the number of queries per hour that constitute query rates for torrents of s per hour or higher:

$$6395604 * s^{1-b} \quad (22)$$

But what's b ? We don't have empirical data to fit to. Instead, we assume that the probability that a query is part of a query rate, s , is the same as the probability that a torrent is owned by s nodes, i.e. $a = b$. What can we use this for? We can work out the proportion of queries that we can satisfy, but it's the same as the number of torrents that we can discover because we're using the same exponent and the scale doesn't matter.

If we change the total query volume per hour but keep the number of torrents and the number of nodes the same then the query rate to torrents must increase. If this happens the query distribution is no longer following the popularity distribution, and so $a \neq b$. When we change the total query volume we are implicitly increasing the number of torrents (and so nodes must be joining more torrents than they were).