Power Laws

The Frequency Distribution of Web Page Requests,
Does it follow Zipf law?

1 Introduction

The Poisson, the Binomial and many of the common distributions, are such that for large values of the random variable the distribution decays exponentially. Power law distributions decay polynomially for large values of the random variable. That is, the distribution decays polynomially as $x^{-\gamma}$ for $\gamma > 1$. This means that in a power law distribution, rare events are not so rare (thick tail).

A discrete power law distribution with coefficient ? ¿ 1 is a distribution of the form

$$P(X = k) = Ck^{-\gamma} \ ; \ \ k = 1, 2,$$

This equation describes the behavior of the random variable X for sufficiently large values of X. The distribution for small values of X may deviate from the expression above.

To detect a power law In a log-log plot (log of probability vs log of X) we should see a line with a slope determined by the coefficient γ.

In Web applications, the distribution of web pages ranked by their popularity (frequency of requests in a large set of web pages) is believed to be a power law distribution known as Zipfs law.(G.K Zipf, Human Behavior and the Principle of Least Effort. Addison-Wesley. Cambridge, MA 1949). First count events (i.e., how often each web page is requested). Then list the pages in order of their frequency of occurrence (f) and assign rank=1 (r) to the page with the highest frequency. This allows us to explore the relationship between the frequency of the web page f and its position in the list, known as its rank r. Zipfs law says that $f = Cr^{-1}$ for r=1,......,n with n being the total number of pages and where r=the rank. This implies that f.r=C for some C (e.g. the 50th most popular page should appear with three times the frequency of the 150th). This is a power law with $\gamma = 1$. The letter f is the equivalent of P(X=k) in the formula above and the letter r is the equivalent of k.

A preliminary first impression of whether Zipf's law holds would be to multiply f.r and see if you get a constant. Don't expect to get a constant for very small values of r (i.e, for r=1,2, or 3....). Getting a constant for most of the multiplications, is a good sign. But don't expect all the products to equal the constant. Look for an overall pattern. Then you can take the log of the frequency and the log of the rank, plot them and see if it follows a straight line. Determine the slope of that line. It should be -1. If it is not -1, then Zipfs law doesnt hold.

2 A fixed user community

Are web requests from a fixed user community distributed according to Zipf's law too? Put another way, does $P(X = r)$, the probability of a request for the rth ranked page follow Zipf's law? As we saw above, Zipf's law states that the relative probability of a request for the r'th ranked page is inversely proportional to r.

$$P(X = r) = \frac{c}{r}, \quad 1 \leq r \leq n$$

where n is the total number of Web pages and the constant c is determined from the normalization requirement, $\sum P_y(i) = 1$. Thus,

$$c = \frac{1}{\sum_{i=1}^{n} \frac{1}{r}} = \frac{1}{H_n} \simeq \frac{1}{ln(n) + E}$$

, where H_n is the partial sum of a harmonic series; that is: $H_n = \sum_{i=1}^{n}(\frac{1}{r})$ and $E = 0.577$ is the Euler constant.

The assumption made about the distribution of the popular web pages has a lot of implications for web cache replacement algorithms.

A web cache –also called a proxy server– is a network entity that satisfies HTTP requests on the behalf of an origin server. The Web cache has its own disk storage and keeps in this storage copies of recently requested jobs. A user's browser can be configured so that all of the user's HTTP requests are first directed to the Web cache. Once a browser is configured, each browser request for an object is first directed to the Web cache. As an example, suppose a browser is requesting the object `http://www.stat.ucla.edu/courses`. Here is what happens:

1.- The browser establishes a TCP connection to the Web cache and sends an HTTP request for the object to the Web cache.

2.- The Web cache checks to see if it has a copy of the object stored locally. If it does, the Web cache forwards the object within an HTTP response message to the client browser.

3.- If the Web cache does not have the object, the Web cache opens a TCP connection to the origin server, that is, to `www.stat.ucla.edu`. The Web cache then sends an HTTP request for the object into the TCP connection. After receiving this request, the origin server sends the object within an HTTP response to the Web cache.

4.- When the Web cache receives the object, it stores a copy in its local storage and forwards a copy, within an HTTP response message, to the client browser (over the existing TCP connection between the client browser and the Web cache).

A Web cache can substantially reduce the response time for a client and Web traffic in the Internet as a whole. For more details on web caches, you can read Kurose and Ross (2003).

If we assume Zip's law for the page request distribution, and we assume that the Web page requests are independent and the cache can hold only m web pages regardless of the size of each Web page; furthermore if we adopt a removal policy called "least frequently used", which always keeps the m most popular pages, then the hit ratio $h(m)$ –the probability that a request can find its page in cache–is given by

$$h(m) = \sum_{i=1}^{m} P(X = r) \simeq cH_m = \frac{H_m}{H_n} \simeq \frac{ln(m) + E}{ln(n) + E}$$

, which means the hit ratio increases logarithmically as a function of cache size. This result is consistent with previously observed behavior of web cache found by some authors (Trivedi, 2002)

How can we determine whether all the above assumptions hold? By looking at data. In this activity, we will only check one assumption: do requests for web pages follow zipf law?

Notice that some authors, i.e., Breslau et al. (1999) have found that the distribtuion might be Zipf-like, with the function being

$$P(X = r) = \frac{c}{r^{\gamma}}, \quad 1 \leq r \leq n$$

where $0 \leq \gamma \leq 1$.

3 Activity

In this activity, we ask the question that we investigated above with a particular data set. Are web requests from the msnbc community distributed according to Zipf's law. To find out, we will use the msnbc web server logs. So the data on which this activity is based comes from Internet Information Server (IIS) logs for msnbc.com and news-related portions of msn.com for the entire day of September 26, 1999 (Pacific Standard Time). Each sequence in the dataset corresponds to page views of a user during that twenty-four hours period. The actual logs were processed to obtain the sequence of visits for each user.

user 1: 1 1
user 2: 2
user 3: 3 2 2 4 2 2 2 3 3
user 4: 5
user 5: 1

Each event in the sequence corresponds to a user's request for a page. Requests are not recorded at the finest level of detail–that is, at the level of URL, but rather, they are recorded at the levels of page category (as determined by a site administrator). The categories are frontpage, news, tech, local, opinion, on-air, misc, weather, msn-news, health, living, business, msn-sports, sports, summary, bbs, travel. Visit also www.msnbc.com to get a feeling for what these page categories mean.

To acquaint yourself with the raw data, go to **Data URL**
http://kdd.ics.uci.edu/databases/msnbc/msnbc.data.html

The total number of users is $n = 989818$. All together, they did a total of 5,688,612 page hits.

Here are the R commands you will need to do the second problem in homework 1 with R I am showing you what the commands are and the output. I also show how the first lines of the data look like. The column names are

Column[1]: user number

Column[19]: depth of browsing=number of unique pages visited=sum of columns 2-18

Columns[2:18]: each column corresponds to a page in the msnbc data set, e.g. column 2 Is for frontpage, column 2 for news, etc. See the site of the raw data set The variable is 1 if that user visited the page, and 0 if not. The pages follow this order

frontpage news tech local opinion on-air misc weather msn-news health living business msn-sports sports summary bbs travel

```
    >zipfmsnbc=matrix(scan(file="http://www.stat.ucla.edu/~jsanchez/oid03/datasets/msnbc
Read 18806542 items
> zipfmsnbc[1:5,]
     [,1]   [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14] [,15]
[1,]    1    1    0    0    0     0    0    0    0     0     0     0     0     0     0
[2,]    2    0    1    0    0     0    0    0    0     0     0     0     0     0     0
[3,]    3    0    1    1    1     0    0    0    0     0     0     0     0     0     0
[4,]    4    0    0    0    0     1    0    0    0     0     0     0     0     0     0
[5,]    5    1    0    0    0     0    0    0    0     0     0     0     0     0     0
```

Notice how this reflects the fact that users 1-5 visited the pages with a 1 (from row 2-18)

Notice also that the information in this file does not tell us how many times a visitor clicked on a page. We only know whether s/he ever clicked on it, not how many times.

To obtain the frequencies of visitors that visited that web page you will need the following commands

```
 popularity=matrix(rep(0,17),nrow=1)
 for(i in 2:18){
        popularity[i-1]=sum(zipfmsnbc[,i])
         }
 popularity
```

With these numbers, you can go on to rank the pages, and do the analysis you need to do to determine whether zipf or other kind of discrete power law is followed by the ranking of pages by the number of visitors that clicked on them.

Once you rank the pages, do a table that shows the page name and the frequency and rank, from lowest to highest rank.

To do the regression to determine the slope of the log-log relation, you can do

```
sorted=sort(popularity,decreasing=T)  #sort from most frequent to least frequent
rank=seq(1,17)
regfit=lm(log(sorted[1,])~log(rank))  #notice we need logs
regfit        #   this gives you the slope
```

To do the plot of the frequencies versus rank, you can do it with R with the following commands

```
freq=(sorted/989818)
rank=seq(1:17)
plot(rank,freq)
```

Alternatively, you can do it with by hand with the numbers you got for the frequencies.

Question 1

Then one by one, tabulate each of the 17 page categories, for example..

```
tabulate frontpage
```

Then construct a table that contains the number of users that hit each page (the frequency), in decreasing order of importance of the page and the proportion of users that hit each page (the relative frequency); that is, the most popular page goes first, the next is the next most popular, and so on. For example, if the frontpage is the one most accessed, that is $r = 1$. And so on.

Do a graph of this table. What is the shape of this graph?

Question 2:

Does the distribution follow Zipf's law? Use all the approaches described in the theory part to determine that. All the output you obtained from R, once you organize it and put it nice for presentation should help you answer this.

References

[1] Trivedi, Kishor S. (2002). Probability and Statistics with Reliability, Queueing and Computer Science Applications. Wiley and Sons.

[2] Breslau L, Cao P, Fan L, Phillips G, Shenker S. (1999). Web Cachin and Zipf-Like Distributions: Evidence and Implications. IEEE Infocom Proceedings, Vol. XX, No Y, 1999, page 126-134.

[3] Kurose, J. F., Ross K W. Computer Networking. A Top-Down Approach Featuring the Internet. Addison Wesley.

Appendix A.

There is a Stata version of the data file that you can access from Stata. The data set name is msnbc.dta and is in the same location as the text file we used above.