Poisson process and continuous-time Markov chains

Stochastic Processes - Assignment 2

Javier Martínez Llamas
Santiago Raposo
Ricardo Hortelano

1 Non-homogeneous Poisson Process

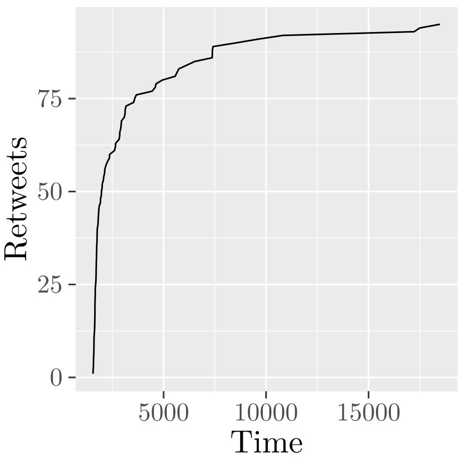
The account selected for the extraction of the tweets is *@realmadrid* since it has a constant number of retweets (RT), not extremely high, allowing us to measure the counting process in a reasonable time space. Although library *rtweet* obtains information about the retweets and their time stamps it is worth mentioning that the results detailed here do not reflect the actual behaviour of the account since retweet count is limited to 100.

1.1 Use descriptive statistics and graphics to explore the retweets data set in terms of the number of retweets by time and the time between retweets. Does it fit the hypothesis of a Poisson process?

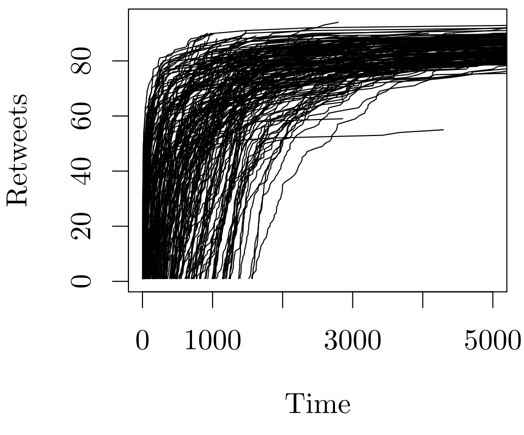
We know that a Poisson process is a counting process related with the Poisson and Exponential distributions. Being a counting process a stochastic process $\mathbf{N} = \{N_t, t \geq 0\}$ satisfying:

- $N_t = 0$
- N_t is integer valued
- For $s < t, N_s \leq N_t$
- For $s < t, N_t - N_s$ represents the number of events that occur in the time intervals (s, t]

When plotting the tweet with maximum retweets (limited) and all the tweets we can see the behaviour of the retweet counting process.



(a) Maximum RT Tweet Distribution



(b) All Tweets Distributions

Each retweet counting process is defined by a time series, measuring the time for the next count (in minutes) since the creation of the tweet. That is, starting at $N_t = 0, t = 0$ the arrival times s increases the count by 1 unit, fulfilling the condition $s < t, N_s \leq N_t$. Moreover, for any time $s < t$ the number of retweets in the interval $(s, t]$ is the difference $N_t - N_s$.

Assuming that, if a user undoes a retweet, when acquiring the dataset the former time of retweet it is no longer recorded, the count can only increase and never decrease. The real model does not hold this assumption since this restriction is imposed by the internal functioning of Twitter and its API.

i.e. The following sequence of times would define a retweet counting process for a tweet with 6 RT. Where the first event occurs after 1.4 minutes.

1.400000 1.616667 1.716667 1.716667 1.900000 2.616667

Knowing that, as stated in Section 6.7 of Dobrow RP *Introduction to Stochastic Processes with R*, a non-homogeneous Poisson process is a counting process $(N_t)_{t\geq 0}$ with intensity function $\lambda(t)$ if

- $N_0 = 0$
- For all $t > 0$, N_t has a Poisson distribution with mean
$$\mathbb{E}[N_t] = \int_0^t \lambda(x)dx$$
- For $0 \leq q < r \leq s < t$, $N_r - N_q$ and $N_t - N_s$ are independent random variables.

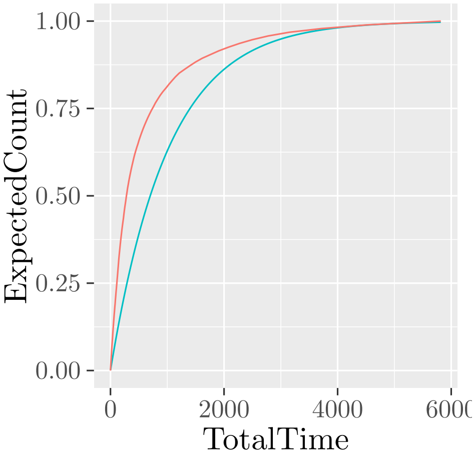
We know that the process already satisfies the conditions of a counting process. In order to find if it fits the hypothesis of a Poisson process, knowing that the times are not homogeneous, we will consider a non-homogeneous Poisson process.

When analyzing time frequencies between retweets we notice that the ratio decreases over time (this behaviour can be observed in the (1a) plot). Meaning that if it were to be a non-homogeneous Poisson process, for all $t > 0$, N_t it would follow a Poisson distribution with $\mathbb{E}[N_t] = \lambda = \int_t^0 \lambda(x)dx$ where $\lambda_t < \lambda_{t+1}$ for all t (difference greater over time), assuming unknown intensity function.

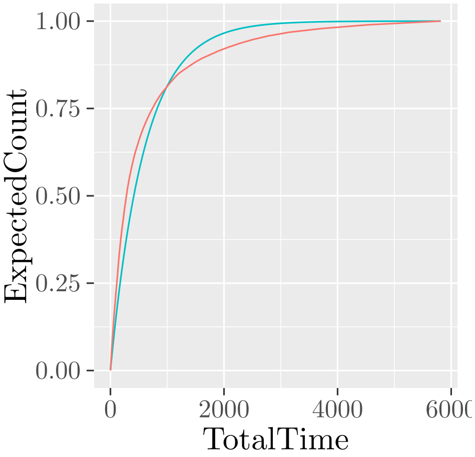
1.2 Assuming you can model the number of retweets by time as a non-homogeneous Poisson process with intensity function $\lambda(t) = \theta e^{-\theta t}$, $t > 0$, graphically explore possible values of θ and choose the one that best fit your data. Explain all the considerations you make.

In order to find the best θ possible, we need to simplify the problem. The first thing we perform is to find the mean poisson process of the Real Madrid twitter account. We can treat this mean poisson process as the expected retweet poisson process of a random tweet written in the Real Madrid account.

Because not all the tweets have the same amount of retweets we cannot calculate directly the mean value for a given time. In order to compute such value, first, all tweets times are interpolated to have the same amount of points (the number of points is equal to the number of retweets of the max RT tweet). Later, the values can simply be summed and divided by the total number of tweets. Finally, this mean line is scaled to a range $[0, 1]$ for easy computation and visualization along with the θ. Once we have this mean line, we can plot it along with an arbitrary value of theta and compare visually.



(a) Mean RT time of tweets (Red) and Simulated values with arbitrary θ (Blue)



(b) Mean RT time of tweets (Red) and Simulated values with optimal (Blue)θ

Once this two lines are plotted jointly, it is easily to see that an optimal value for θ would be the one that minimize the area between the two lines. That is exactly we did in the right plot. We write a function that automatically minimize this area and give us a value of $\theta = 0.001683839$ with an area between lines of 173.1745

1.3 Write an R function to simulate data from a non-homogeneous Poisson process during a given time interval. Describe the inputs and outputs of your function.

As stated in section 11.5.1 of Ross, Sheldon *Introduction To Probability Models*, given n events of a non-homogeneous Poisson process by time T (maximum time) the n event times are independent with a common density function

$$f(s) = \frac{\lambda(s)}{\mathbb{E}[N_T]}, \;\; 0 < s < T, \quad \mathbb{E}[N_T] = \int_0^T \lambda(s) ds$$

By simulating N_T, the number of events by time T, and then simulating N_T random variables from the previous density function we can generate a NHPP.

However, this procedure requires on the inverted distribution of $f(s)$ and may be too complex to obtain. Since this is a general function with the aim of simulating any non-homogeneous process we will opt for a rejection method.

That is, we reject or accept values of uniform $(0, T)$ random variables. Defining a limiting $\overline{\lambda}$, a bound on $\lambda(s), 0 \leq s \leq T$, then

$$\frac{T\lambda(s)}{\mathbb{E}[N_T]} \leq \frac{\overline{\lambda}T}{\mathbb{E}[N_T]}$$

Therefore, generating random numbers U_1 and U_2 we accept U_1T if

$$U_2 \leq \frac{\lambda(U_1T)}{\overline{\lambda}}$$

This $\overline{\lambda}$ is an arbitrary parameter based on the non-homogeneous Poisson process to be modelled. For the retweet counting process, knowing that the rate at which a retweet occurs decreases over time (most RT occur right after the publication), in order to mimic that behaviour, $\overline{\lambda}$ needs to be a low value as to reject most of the values at the end of the process.

Translated into the following R function

```
simulateNHPP <- function(intensity_function , time, lambda_bound) {
        X <- numeric(0)

        for ( t in 1:time){
                u <- runif(2)
                accept <- u[2] <= intensity_function(time*u[1]) / lambda_bound
                if (accept)
                        X <- c(X, time * u[1])
        }

        return(sort(X))
}
```

The simulation function takes 3 parameters, *intensity_function*, *time* and *lambda_bound*. Representing the intensity function of the process to be simulated, a maximum time t to simulate data in the interval $(0, t)$ and an arbitrary $\overline{\lambda}$ to adjust the simulation.

It returns a list containing all arrival times, from the starting time $t = 0$, each one representing one retweet.

1.4 Simulate data using the previous function trying to mimic the real retweets data set. Compare the distribution of arrival times for the real and simulated data set. Comment on the differences.

Using the previously implemented function and having estimated the optimal value for θ we model our account retweet process. Knowing that the maximum number of retweets is limited by Twitter's API to 100, the following $\overline{\lambda} = 0.01 = \frac{1}{100}$ has been established. This $\overline{\lambda}$ would vary depending on these maximum retweets.

When running the algorithm we obtain a sequence of arrival times in the interval $(0, T)$ with the following frequency histogram

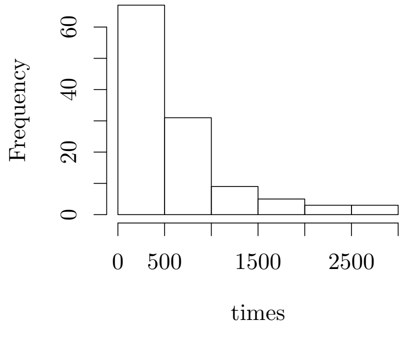
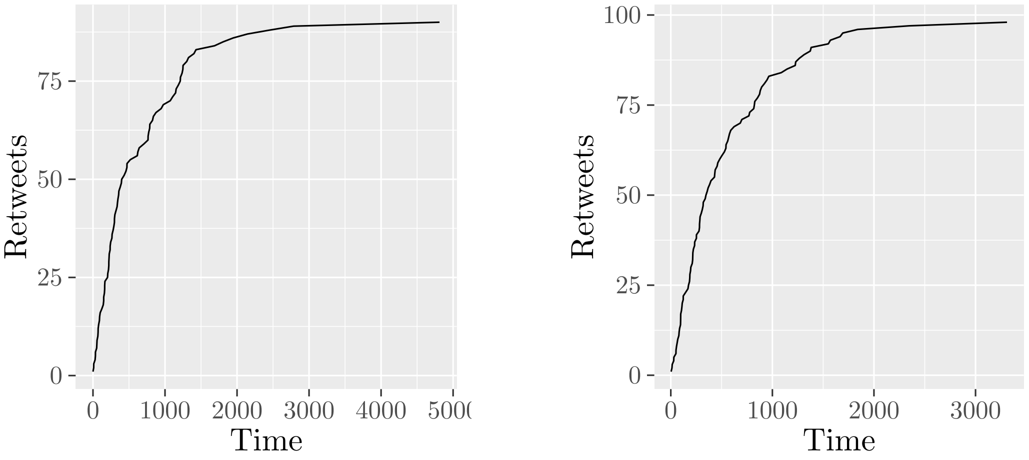


Figure 3: Frequencies of Simulated Process

As expected from the original retweet counts the frequency of a user retweeting is higher after the publication of the tweet.



(a) First Simulation, $\overline{\lambda} = 0.01$

(b) Second Simulation, $\overline{\lambda} = 0.01$

The simulated process behaves similarly to the account being studied. However, this does not represent the behaviour of all tweets, as the potential impact of the tweet and the interest it may generate influences the frequency of times.

1.5 Using the intensity function of part b), compute the probability that a tweet from this user will get more than 10 retweets in the first hour, and the expected number of retweets of a tweet after 24 hours.

The expected number of retweets are given by

$$\mathbb{E}[N_t] = \int_0^t \lambda(x)dx \quad \Longrightarrow \quad \mathbb{E}[N_t] = \int_0^t \theta e^{-\theta x} dx$$

where the intensity function is $\lambda(t) = \theta e^{-\theta t},\ t > 0$.

However, when calculating the expected values for the previously defined $\theta = 0.001683839$ we get that $\mathbb{E}[N_t] \in (0,\ 1],\ \forall t > 0$. Since the mean retweet process was previously scaled, in order to revert this behaviour, we will use the resulting probability to obtain the following adjusted expectation for N_t while using the same $\overline{\lambda} = 0.01$ as in the simulation as the scaling factor of the expected retweets.

$$\mathbb{E}'[N_t] = \frac{1}{\overline{\lambda}} \times \mathbb{E}[N_t] = \frac{1}{\overline{\lambda}} \int_0^t \theta e^{-\theta x} dx$$

Therefore, the expected number of retweets after 24 hours (1440 minutes) would be

$$\mathbb{E}'[N_{1440}] = \frac{1}{\overline{\lambda}} \times \mathbb{E}[N_{1440}] = \frac{1}{0.01} \times 0.9114979 \sim 91\ retweets$$

Similarly, the probability of N_t being greater than k is

$$P(N_t \geq k) = 1 - \sum_{t=0}^{k} \frac{\mathbb{E}'[N_t]^k e^{-\mathbb{E}'[N_t]}}{k!}$$

then

$$P(N_{60} \geq 10) = 1 - \sum_{t=0}^{10} \frac{\mathbb{E}'[N_{60}]^k e^{-\mathbb{E}'[N_{60}]}}{k!} = 0.3682237$$