

Random walk on the discrete line

Author: Thierry Monteil

Average jump

The thirst street is made of a juxtaposition of bars labelled $0, 1, 2, \dots$, starting from downtown (bar number 0). A bunch of drunkards is drinking in bar 0. At midnight, all of them are kicked out of the bar. Every minute, each drunkard either stays at its position with probability $1/2$, or jumps in the next bar with probability $1/2$, they are so drunk that the jumps are independent, and the drunkards do not interact with each other. We want to understand how the drunkards get distributed when n is large.

Write a function `drunkard_jumps(n)` which samples a list of n jumps (with values 0 or 1).

Hint: you can have a look at the `randint` function.

Write a function `drunkard_positions(n)` which returns a list of n consecutive positions of such a drunkard, starting at position 0.

Write a function `plot_drunkard_positions(n)` which plots the sequence of positions as a function $\{0, \dots, n-1\} \rightarrow \mathbb{N}$.

Hint: you can have a look at the `enumerate`, `point2d` and `line2d` functions. Starting from an empty `Graphics()` object and adding of several graphics objects to it results in the superposition of those graphics.

Extend this to a function `plot_drunkard_positions(n, d)` which samples the sequences of d drunkards on the same graphic and observe the result.

Hint: You can have a look at the `rainbow` function to get a sequence of different colors, and use the `color` option in the graphics objects.

Write a function `drunkard_average_speeds(n)` which returns a list of n consecutive average speeds, that is, for each $i \leq n-1$, the average number of bars visited from 0 to i per minute.

Write a function `plot_drunkard_average_speeds(n, d)` that plots the consecutive average speeds of d drunkards up to n minutes.

Hint: if you think that this function is very close to `plot_drunkard_positions(n, d)`, you can try to refactor your code by writing a single `plot_samples(function, n, d)`, to be applied to the `drunkard_positions` and `drunkard_average_speeds` functions.

Formulate a conjecture on the long-term behaviour of the average speed of the drunkards.

Scattering

To avoid accidents caused by dangerous cars, an enlightener lifeguard is following the group of drunkards at constant speed $1/2$ bar per minute with a light.

Write a function `plot_enlightener_view(n, d)` that plots the trajectories of d drunkards from the point of view of the enlightener lifeguard.

In order to save energy, the enlightener lifeguard wants to know how far should her light illuminate in order to enlighten most drunkards. Of course, after n minutes, a distance of $n/2$ bars is enough to ensure that all drunkards are illuminated, whatever the probabilities. But she wants to save more energy, while still illuminating most drunkards.

Write a function `guess_scattering(n, d)` to help you guess the best α in $[0, 1]$ such that a light illuminating n^α after n minutes illuminates most drunkards.

Hint: the logarithm helps to discover polynomial and exponential rates.

Write a function `plot_scattered_enlightener_view(n, d)` that adds to the enlightener lifeguard's view, some constant (positive and negative) multiples of n^α .

Formulate a conjecture on the second order long-term behaviour of the trajectories of the drunkards.

Distribution

After n minutes, the drunkards fall asleep, and the lamp illuminates them. The enlightener lifeguard takes a picture of the "heap" of sleeping drunkards. The zoom is such that the picture boundaries corresponds to the enlightened part of the street.

Write a function `asleep_drunkard_distribution(n, d)` that plots the distribution of d asleep drunkards on the picture after n steps.

Formulate a conjecture on the limit shape of the heap of asleep drunkards (d might depend on n , do not forget the zoom).

Write an animation that shows the convergence to some limit curve when n tends to infinity.

An exhaustive search

To find the exact limit distribution, an army of docile soldiers is hired, so that each possible drunkard walk is simulated by one soldier.

How many soldiers do we need to simulate each possible trajectory with n steps ? After n steps, **how many** soldiers are located in front of the bar k ?

Write a function `exact_soldier_distribution(n)` that plots the distribution of the soldiers at time n .

Universality

The day after, the drunkards got sick of alcohol and decide to take some ecstasy. Hence, at each second, they are jumping by a step s_i with probability p_i , where:

- $s_i \in \mathbb{R}$ ($0 \leq i < k$)
- $p_i \in [0, 1]$ ($0 \leq i < k$)
- $\sum_{i=0}^{k-1} p_i = 1$

Try to extend and illustrate the previous laws to this setting.

Hint: you can have a look at the `random` function.

Conclusion

Let us discover the following laws of probabilities:

- the [law of large numbers](#)
- the [central limit theorem](#) and the [normal distribution](#)
- its relation with the [binomial coefficients](#)