**Note 1**

* In the present lesson, we will introduce a simple stochastic model that can reproduce many of the phenomenological properties observed in real-world financial markets
* The model can be easily understood as a generalized random walk.

**Note 2**

* As usual, we begin by importing the necessary numerical and graphics libraries.

**Note 3**

* We define the helper functions introduced in the previous lesson to analyze the stock returns.

**Note 4**

* Here we will work with a simple stochastic model developed by Japanese researchers and published recently in Physical Review E, one of the main journals for statistical physics.
* Any stock market can be understood as a collection of agents or dealers buying and selling some stock.
* As a minimal model we consider a market with only two dealers, which we call 1 and 2, which are trading some stock.
* Each dealer has a maximum price that he is willing to pay to buy new stock, and a minimum price he is willing to accept to sell stock he already owns.
* We refer to these as the bid and ask prices, respectively.
* Since the dealers are in this to make money, the ask price should always be greater than the bid price.
* To characterize the position of the dealers, we introduce an additional price, called the mid-price, which is just the average of the bid and ask prices.
* For simplicity, we will assume that the difference between ask and bid prices, which is called the 'spread', is a constant equal for all dealers.
* Thus all we need to know the configuration of the market are the mid-prices p1 and p2 at each moment in time.
* We have not yet specified the dynamics of the dealer's prices, but we know that whenever the 'ask' price of one dealer is equal or lower than the 'bid' price of the other, they should perform a transaction.
* In mathematical terms, this is expressed as Eq.(K1). If the absolute value of the difference in mid-prices is equal or larger to L, then a transaction will take place.
* In this model, we consider only one type of stock, and we do not attempt to model the amount of stock available.
* Whenever a transaction takes place, we assume that a unit of stock is transferred from one dealer to the other, although the amount possessed by each of them is infinite.
* What we are interested in here is not the amount of stock that can change hands, but the price at which they are trading it.

**Note 5**

* Let us now discuss the dynamics of this dealer model.
* The mid-price is assumed to follow a 1D random-walk in 'price' space.
* Previously, you learned that a random walk can describe the diffusional motion of particles. At each step, the drunken man has two options, he can either take a step to the left, or to the right.
* In this case, we can consider the random step to be caused by the influence of the outside world on the dealer. Everything he sees and learns can change his opinion of how much he thinks that the stock is worth.
* The update rule for the prices is given in Eq. (K2), which you will recognize as a simple 1D random walk process. At each step the stock can move up or down by a fixed amount with equal probabilities.
* While the dealers' prices are evolving independently, the moment that the transaction condition (Eq. (K1)) is satisfied they will execute a trade and exchange one unit of stock.
* The price of the trade, which by definition will be the Market price P of the stock, is given by the average of the mid-prices.
* After the exchange, the dealers are free to resume their respective random walks, but now the starting price is the Market price of the transaction that just took place.
* Finally, it is convenient to define an additional unit of time, called the 'tick' time. The tick time measures the number of transactions. Every time a transaction takes place, we update the tick time by one.
* In contrast, the 'real' time measures the total number of steps taken during the random walks.
* In the figure, we have drawn a schematic representation of the dynamics given by the model.
* At time ti, the bid price of dealer 1 matches the ask price of dealer 2. A transaction takes place, in which dealer 1 buys one unit of stock from dealer 2, at a market price Pn.
* At this point, we set the 'tick' time to be n.
* After the transaction, the two dealers are in agreement regarding the price of the stock. Thus, their mid-price matches the Market price Pn.
* The dealers resume their independent random walks until time ti+3, when dealer 2 buys from dealer 1 at a Market price of Pn+1.

**Note 6**

* To make the link to a random walk process even more evident.
* Let us make a change in variables, from the mid-prices p1 and p2, to the price difference D and A, as defined in Eqs.(K4-K5).
* The two 1D random walks can then be transformed into a single 2D random walk in this (A,D) space, with the dynamics defined in Eq.(K6-K7).
* In this 2D space, the "particle" can move along the A axis left or right a distance c Delta p, with probability 1/4, or it can move along the D axis up or down a distance 2 c Delta p, with probability 1/4. It cannot stay in the same position.
* In contrast to the examples you saw previously, this random walk is constrained by the transaction condition.
* Whenever the price difference is greater than L, the random walk ends, and a transaction takes place.
* This means that we have two absorbing boundaries at the top and bottom of our domain, for D equals plus or minus L. When the "particle" gets to this point the random walk has to end.
* We will try to visualize this with a simple simulation next.

**Note 7**

* To begin we define a dictionary which groups all the simulation parameters, the spread L, the constants c and dp that determine the random step size, and the time step dt.
* Then we define a function which performs the random walk for a single transaction.
* That is, given an initial mid-price for dealers one and two, it will solve for the two independent random walks defined in Eq.(K2).
* The random walks end the moment that the price difference exceeds the spread L.
* To aid in the visualization, we save the trajectory data not for the prices, but for the difference and average, defined in Eq.(K4-K5), which should describe a 2D random walk.

**Note 8**

* Here we perform three independent simulations of one transaction each, starting from the same initial configuration (same prices).
* As you can see from the plots of the trajectory data, the particle is undergoing a random walk, with random displacements up/down/left/right.
* However, we can easily see that the vertical steps are longer than the horizontal steps, just as predicted from Eq.(K6-K7)
* We have drawn the absorbing boundaries for D=±L. When the particle reaches any of these lines, the difference in the mid-prices is larger than the spread and a transaction occurs.
* This signals the end of the random-walk and the start of a transaction.
* The horizontal position at which the particles touch the boundary gives the change in the market price of the stock, and determines the new market price.
* In the three examples shown here, Walk 0 and Walk 1 end at the top boundary, which means dealer 1 buys from dealer 2; and Walk 2 ends with dealer 1 selling to dealer 2.
* Also, for Walk 0, the Market price at the end of the walk is lower than at the start, whereas for Walks 1 and 2 it is higher.
* Finally, we notice that the number of steps required for the particle to reach the boundary can be vary large. Here, we needed almost 10000 steps for Walk 1 to finish, compared with just 1600 for Walk 2.

**Note 9**

* We want to run a simulation of our model stock market over many transactions.
* For this, we modify the previous code to repeatedly perform many random walks, updating the market price, and resetting the mid-prices of the dealers after each one.
* We only require one additional parameter numt, which is the number of ticks, transactions, or random walks to perform.
* We define arrays of size numt for the market price and the tick time. The former for the market price of the transactions, the latter for the time at which the transactions took place.
* The main code can be written as two nested while loops, one for the number of transactions, the other for the random-walk of a single transaction.
* At the end of a random walk, we perform a transaction by setting the new market price to be the average of the mid-prices and we also save the current time.
* Because the simulation takes around 4 minutes, we will not evaluate it now. We have previously saved the data to a file and will use this to analyze the trajectories.

**Note 10**

* We start by loading our previously saved simulation data.
* This data contains the time stamp for each transaction, together with the stock price.
* The time values are not interesting in themselves, what is more interesting is the time intervals between subsequent trades, which we save in an array 'dprice'.
* Likewise, the stock price in itself does not tell us so much. Instead we look at the logarithmic difference of the price between two trades G1(t).
* This return can be easily computed using the functions defined in the previous lesson. For convenience we also normalize the data to have unit variance.
* The three data sets, the stock price, the price return, and the transaction interval are plotted here.
* Looking at the stock price, we see a quantity that is fluctuating in time in a manner reminiscent of real stocks. We see small scale fluctuations together with large scale price changes.
* For the price return, we see evidence of a stochastic process that is clearly non-Gaussian in nature, as we obtain values which are more than 6 standard deviations away from the mean.

**Note 11**

* To evaluate the mode we need to perform a more quantitative analysis.
* For this, we calculate the probability distribution functions of the absolute price return and the transaction interval.
* We plot the data on a semi-logarithmic scale, and see that both distributions fall on a straight line (at least near the tails).
* This means that the distribution is exponential, and thus defined by a Poisson process.
* [RY] How does this compare to real data? We saw that the price return is actually a power law, what about the transaction interval?
* As you say, the price return is not correctly described, and neither is the transaction interval.
* In real markets, the transactions show very clear long-scale patterns (corresponding to times when the markets open in Tokyo, London, and New York) and show clustered and sparse regions at short time scales.
* This cannot be described by the simple Poisson process we have here.
* However, I think the fact that we have such a simple model, essentially a 2D random walk, that can reproduce some non-trivial behavior similar to that of real markets is quite remarkable.
* In fact, we can easily improve the model to fix the short-comings you have pointed out.
* In the next lesson I will present one solution to recover the power-law behavior of the price returns you have mentioned.