**Note 1**

* In this last lesson, we will continue our analysis of the dealer model used in the previous two lessons in order to calculate the dynamical data. In particular we are interested computing the time correlation functions of the price returns.

**Note 2**

* As usual, we start by importing the necessary libraries.

**Note 3**

* Here we import many of the helper functions introduced in previous lessons to analyze our data.
* We have a function to calculate the logarithmic return, a function to normalize any time series data to have zero-mean and unit variance, and a function to compute the time-correlation function.

**Note 4**

* Let us briefly recall the situation we are modeling.
* We consider a market composed of two dealers buying and selling stock from each other.
* Their mid-prices, the average of their bid and ask prices, evolve according to a 1D random-walk in price space.
* At each step, this price can move up or down some fixed amount with equal probability.
* In order to incorporate the trend-following behavior, an additional memory or drift term is added to the price change at each step.
* The equation for the price dynamics is given in Eq.(M1-M2).
* By tuning the parameter d, we can model "trend-followers" that like to bet with the market, or "contrarians", that bet against the market.
* The current trend is calculated by a running-average of the previous M price increases.
* Finally, when the random-walks of the two dealers take them to a configuration in which the bid price of one matches the ask prices of the other, the random walk ends and a transaction takes place. This condition is as shown in Eq.(M4).
* The average prices of the dealers at this point in time define the Market-price of the transaction.
* Finally, we analyze all price returns over a time τ, in terms of the logarithmic difference, as given in Eq. (M5). This is essentially the relative return, if one were to buy at time t and sell at time t+τ.

**Note 5**

* Here, we have the code corresponding to model 2, which we introduced in the previous lesson in order to recover the correct power-law behavior for the price returns.
* We have not made any modifications.

**Note 6**

* Let us begin where we left of in the previous lesson, by comparing the price dynamics for different values of d, for both contrarians, trend-followers, and random-walkers.
* Here, we setup 5 simulations, for d values of -2, -1.25, 0, 1.25, and 2.
* The running average is computed using the previous 10 ticks, M=10.
* The other parameters are the same as in previous simulations.
* Since running the simulations is not suitable for a live demonstration, as it takes almost 30 minutes to complete, we have saved the trajectory data for the prices in the file 'model2\_M10\_5d.txt'.
* We can load the data from the file to continue with our analysis.
* To simplify the code, we import all the data as a single pandas DataFrame.
* This contains the price data for all five d values for the 5000 ticks.
* In addition, we have compute the price returns over one tick as a separate dataframe.

**Note 7**

* If we attempt to plot all the data on the same graph, we will only be able to see the evolution of the trend-followers for d=2d=2. For this case, we have incredibly large changes in the price, reminiscent of a stock bubble with exponential growth.
* Let us ignore this data for now, and focus on the other four data sets.

**Note 8**

* Here we have plotted the time evolution of the market price and price return without the data for d=2.
* As we had already shown in the previous lesson, contrarians exhibit very stable prices, whereas trend-followers show very unstable price-changes.
* All simulations were run using the same random number sequence for the steps in the random-walk, this is the reason why the trajectories seem so similar. We did this so that you could have a better understanding for the role of d.
* The price returns seem to show some small scale oscillations, but without computing the time-correlation function it is impossible to be sure.
* Let us do this now.

**Note 9**

* As expected, we find non-zero correlations over a time-scale of tens of ticks. This is the time-scale set by M, the number of ticks to use when calculating the running-average of the previous time-increments.
* We also see a very clear difference between trend-followers and contrarians. With the former showing only positive correlations, whereas the latter show a negative correlation.
* This is intuitively what we expected, since price changes for trend-followers are likely to follow the current trend, whereas contrarians will likely go against it.
* Also, the correlations for d=0, where there is no memory-term, are exactly zero.
* Now the question we must ask ourselves is the following: What if any are the time correlations seen in real stocks?

**Note 10**

* The main problem when computing the time-correlations in real stock data is in finding the necessary high-frequency data.
* Before, when looking at the distribution of price increments, we could do with the closing price after each trading data.
* Now, we are interested in having either tick data or minute-by-minute data.
* Fortunately, after a quick google, search we were able to find such data.
* Here we will use the per tick and per minute price of Apple stock on the day of March 01 2017.
* You will be able to download the corresponding text files from the course website.
* We import the data using pandas, and printout the first five lines in each file.
* The tick data contains over 15000 transactions for this single day. As you can see, in a single second you can have several transactions.
* To prepare for our analysis, here we also calculate the price return over one unit of time, either one tick and one minute, and add it as an additional column to the dataframe.

**Note 11**

* Here we plot the stock price as a function of tick time and clock time.
* The two are in agreement, as they should be, but it is clear that there is not a simple one-to-one correspondence between tick-tick and clock-time.

**Note 12**

* Here we plot the price return as a function of time.
* Note that the top shows the price return over one tick, whereas the bottom shows the price return over one minute.
* There is a clear difference between the two data sets. In particular, the top data shows almost discrete returns.
* We are seeing the finite resolution of the price data. We only have the prices up to a cent, and over one tick, the changes in price are typically of this order. Hence the discrete nature of the returns.
* At first glance, there seems to be no correlation whatsoever in this data.
* Let us calculate the time correlation function to be sure.

**Note 13**

* We now plot the time correlation function of the price returns, for the one-tick returns at the top, and the one minute returns at the bottom.
* Although the tick data seems to show a very small correlation at short times, it is impossible to be sure, as the noise in the data is of similar magnitude.
* For the minute data this seems to also be the case.
* Recall that here we have only used one day of trading data. To obtain better statistics we should average over several days of data.

**Note 14**

* We note that a more detailed analysis, with millions of points for several years of data, in contrast to the ten thousand or so we have used, has been reported by Gopikrishnan and collaborators.
* They show non-zero correlations in the one minute returns, with a decay time of approximately 4 minutes.
* However, the fact that these correlations have such a short-range means it is impossible to predict future prices from previous values.

**Note 15**

* We have shown how a simple-model stochastic model, built borrowing concepts from statistical physics, can reproduce many behaviors seen in real-world stock markets.
* While we considered the simplest possible version, with only two dealers and constant and equal trend-following characteristics, you can easily remove these restrictions. You can try to simulate for hundreds of dealers, with non-constant d values. The main results still hold, it just becomes more complicated to analyze as the number of parameters increases.
* However, it should be clear that no such model will ever be able to accurately and consistently describe all the features of such complicated systems as stock markets.
* For the dealer model we presented, one can recover the non-trivial power law decay of the price returns, but only for a specific set of parameter values. If the parameters of the model are changed, then the nature of the distribution can also change.
* While this type of modeling can help you understand complex real-world systems, you should be very careful when trying to make precise quantitative predictions based on them.