CS 7646 Summer 2016 - Quiz 1

May 31, 2016

Student Name _____

GT ID# _____

1 Stock Data

Adjusted close alters the historical price of a stock to eliminate discontinuities caused by stock splits and dividend payments. (b) and (d)

1. Circle *all* of the statements below that are *true* of the "adjusted close" stock price discussed in class. Assume there is a historical time series of data that begins at some time in the past and ends today.

   (a) *Adjusted Close* and *Close* will always be equal at the earliest date of the time series.

   (b) *Adjusted Close* and *Close* will always be equal at the latest date of the time series (today).

   (c) *Adjusted Close* creates discontinuities in the price that must be considered when backtesting a trading strategy.

   (d) At a minimum, *Adjusted Close* will usually account for stock splits and dividends paid to shareholders.

2. You have obtained a source of historical stock data that contains some gaps in the data for certain stocks. The "best practices" way to fill those gaps if you want to make time series predictions is:

   (a) Don't fill the gaps. Leave them as `NaN`.

   (b) Fill gaps backward (repeating the price after the gap), then forward (repeating the price before the gap).

   (c) Fill gaps forward, then backward.

   (d) Interpolate between the prices on either side of the gap.

2 Pandas

Unfilled gaps will "infect" math operations with NaN. Interpolation or filling backward first will "see the future". (c)

3. You have two `DataFrame` objects. `dfIBM` contains ten years of IBM data, only from days that IBM was traded. `dfSPY` contains one month of market data for all days the market was open. Which join type will result in a dataframe containing one month of data with rows for all days the market was open, and IBM data as `NaN` on days it did not trade? Pay attention to the order of the dataframes given!

   (a) `dfIBM.join(dfSPY, how="left")`

   (b) `dfIBM.join(dfSPY, how="right")`

   (c) `dfIBM.join(dfSPY, how="inner")`

   (d) `dfIBM.join(dfSPY, how="outer")`

Here, dfIBM is treated as the "left" DataFrame. A right join will keep all dates/rows in dfSPY. Rows without a match in dfIBM will be NaN. (b)

4. Native Python code should be preferred to operations in Pandas or Numpy whenever possible, because the native Python code will run much faster.

   (a) True

   (b) False

False. Python is an interpreted language. Pandas and Numpy are written in C and Fortran and are much faster.

3 Numpy

5. Numpy defaults to inner (dot) or outer (cross) product for multiplication, so you must use the operator `.*` to force element-wise multiplication when desired.

   (a) True
   (b) False

   False. The dot-operators like .* are a Matlab convention not followed in Numpy. Operations are element-wise by default.

6. What will be the result of the expression: `a[0:4] = 6`, assuming `a` is a one-dimensional numpy array containing at least four elements?

   (a) An error will be raised, because you can not use an array slice on the left-hand side of an assignment.
   (b) An error will be raised, because the left side of the assignment is size 4 and the right side is size 1, so there is a size mismatch.
   (c) Elements 0 and 4 of `a` will be set to 6.
   (d) Elements 0, 1, 2, and 3 of `a` will be set to 6.

   Assigning to a slice is a key feature of Numpy. Broadcasting (size matching) is also. Range is [0:4), so (d) is correct.

4 Time Series Statistics

7. When you wish to make stock price predictions, it is better to use a single global statistic to approximate the mean or standard deviation of the entire time series, because this way you are incorporating all of the available data.

   (a) True
   (b) False

   False. A global statistic that incorporates data from years ago is not useful when predicting the next five minutes. You would want a rolling statistic.

8. You have obtained a series of *daily* portfolio returns for a 90-day period, and wish to calculate the Sharpe Ratio of the portfolio. Which formulation would be most appropriate to use? Assume the base Sharpe Ratio equation is: $SR = \frac{R_p - R_f}{\sigma_p}$

   (a) SR
   (b) $\sqrt{252} \times SR$
   (c) $\sqrt{90} \times SR$
   (d) $\sqrt{52} \times SR$

   We always annualize Sharpe Ratio for fair comparisons among different time periods of analysis. With daily portfolio returns and 252 market days in a year, (b) is right.

   Remember: the number of days does not matter (90 is a red herring). Only the sampling frequency matters!

5 Vectorization

9. Which of the following code snippets is *not* equivalent (same result) to the other three? (Assume `a` is a 2-D numpy array. Assume `b` is a numpy array of same dimensions as `a`, but set to all zero values.)

   (a)
   ```
   for i in range(1,a.shape[0]):
       for j in range(0,a.shape[1]):
           b[i,j] = a[i-1,j]
   ```
   (b)
   ```
   for i in range(1,a.shape[0]):
       b[i] = a[i-1]
   ```
   (c)
   ```
   b[:-1] = a[1:]
   ```
   (d)
   ```
   b[1:]  = a[:-1]
   ```

   (a), (b), and (d) all shift the rows of a in one direction.

   (c) shifts in the opposite direction.

10. The statement `a[a<4]=0` can be used in numpy to return an array exactly like `a` except with all elements with value less than 4 set to 0 instead.

    (a) True
    (b) False

    True. Creating a boolean array and using it as a "mask" array to select certain elements, and then using that slice as the target of assignment is a powerful feature of Numpy.