Database Systems

You have been hired to be a data analyst at an equity investment fund. You will need to use your data analysis skills to look at market data and come up with an investment strategy.

One way to approach this problem is by looking at the historical performance of some equities, identifying a group of high performers, and trying to identify common factors that contributed to their success.

If these factors can be identified, you can build an investment strategy by selecting other equities to purchase in the future which exhibit similar characteristics to your high-performing group.

HW4:

1. Download the data. Create a new database for this assignment.
   Create tables for the data (in your new database), and load all of the .csv data into them.

2. Calculate all annual returns for each symbol for each year and sort them by performance. (not daily - just yearly is good enough).

   Annual Return Formula: (good for one year only - multi-year is different).

   $$R = (end\_price / start\_price) - 1 \qquad (\%)$$

   Example: Start at $100, end at $115:

   $$R = (\$115 / \$100) - 1 = (1.15) - 1 = 0.15 = 15\%$$

3. Select a group of 30-60 or so companies with very good annual returns. You may want the top group, or you may want to eliminate a few from the very top - if you are suspicious that they might be anomalies, and that the company cannot consistently repeat these very high returns.

   Save these high-performing companies into a table for use in HW5.

   > Hint: CREATE TABLE awesome_performers AS
   >   SELECT ... FROM ... WHERE ... ;

   > Hint II: You may want to use OFFSET if you don't want to take the top 30.
   >   I.e., to eliminate the top 3 and take numbers 4 through 34:
   >   SELECT ...   LIMIT 30 OFFSET 3;

   You can use more than 30 companies if you feel it gives you better results, but use at least 30.

   Submit the SQL script that you used to create the database and the tables.
   ***Submit the SQL script that you used to calculate the annual returns and save
   your high-performers to a new table of their own.***

HW5: (the biggest one)

1. For the group of companies you discovered in HW4, we want to look at their fundamentals in the year that they produced great returns.

   Identify one or two factors in their fundamentals that most of the high-performers have in common. We want factors which you believe could have contributed to why these companies performed well.

   To help identify why these companies had good returns, answer the following questions: (turn in SQL queries that answer the following questions).

   Do the high-performers have any of the following in common?

   High net worth - (total assets - total expenses)
   Or: (even better) high net worth *growth* year-over-year.
   High net income growth year-over-year?
   High revenue growth year-over-year?
   High earnings-per-share?
   High earnings-per-share (eps) *growth*?
   Low price-to-earnings ratio ?
   (this is stock price relative to eps - i.e.: pe ratio = share price / eps)
   Amount of liquid cash in the bank vs. total liabilities?

   Hints: In class, we will go over how to get year-over-year changes using LAG() in a window function, using EXTRACT() to get years from dates, making temporary tables, performing JOINs ON more than one column at a time, and other things that may help.

   ALSO: you may find that we do not have enough data in this small data set to answer all your questions. If you run into difficulties here - try going back to the end of HW4, and "widening the funnel at the top". I.e., allowing more candidates into your high-performers group, to increase the chances that we have enough data on the high-performers to make a conclusion.

   If you don't like these questions, you may look for any characteristic that you like, but clearly identify what you have chosen and why.

   If you choose to answer different questions than those above, or want to answer some combination of the above and some of your own, then *turn in at least six queries* of equivalent or greater difficulty as those listed above.

   It's ok to ask questions and get *negative* answers - i.e., conclude that some characteristic did *not* contribute to your high-performing results (because the high-performers do not share this in common). But you need to find at least one or two characteristics that the high-performers *do* have in common.

2. Once you have identified some factors that you believe contributed to high performance, go back to the original list of all the companies (*not* just the highest returns), and find the top 25-30 companies that have the most similar fundamental factors to your high-performers for the year 2016.

   These will be your potential candidates for investment.

3. From this list of potential candidates, select ten companies to invest in:
    - Make sure there are no more than 2 companies in the same sector.
        We want diversification, and selecting ten companies in the same sector fails to achieve that.
    - Try to base your final selection on some reason you find in the data.
        Identify why you made your final selections (sector diversification, profits, etc.).

Extra Credit:
    From your potential candidates list (from Q #2, above), check the correlation between
        each of your potential candidates, and try to choose companies from the list that have
        the lowest correlation with each other.

        Note: this is not the *minimum* correlation value - as two stocks may be highly
            negatively correlated. We want the correlation values which are <u>closest to zero</u>.

Submit one SQL script which includes the queries that you used to make your selections.
Submit a list of your 10 investment choices, and a (very) breif explanation
        of why you chose them (data-driven reasons).

HW6:

1. Export and back-up your investment database using the pg_dump.

    pg_dump will dump an entire database with the '-d' flag, or individual tables with the '-t' flag:

    Example: pg_dump -U postgres -d investments > backup.sql
    Example: pg_dump -U postgres -t high_performers -t candidates -t final_choices > backup.sql

    This will create a backup of all your schema, tables, and the data that lives in them.
    You can compress, encrypt, and store this data off-site - to be restored easily if needed.
    You can re-create the entire database by simply running the backup file as a regular SQL script:

        psql -U postgres -f backup.sql

2. Create a VIEW of your portfolio - which will be a query of the current status, company name,
    most recent price, and other summary information, so you and your boss can quickly monitor
    your investments.
        (we are presuming that new data is being added to the *prices* table daily).

3. Export the results of that VIEW to .csv so you can email it to your boss, and so your
    non-programmer colleagues can track your portfolio status by viewing it in a spreadsheet.

    Hint: use the following flag when you run your script: -tAF,
    Example: psql -U postgres -tAF, -f your_script.sql > output_file.csv

4. Look up the last price of 2017 of each of the equities you chose, and determine the percent return
    you (your company) would have made on each of them if you had invested in your portfolio
    on Dec 30, 2016.

        Submit the SQL code you used to create your VIEW, and the pg_dump and psql commands
            that you used to export the data.
        Submit the annual return for 2017 of each of your 10 choices, as well as the total return
            of your portfolio.