USING PYTHON TO SOLVE REAL WORLD WORK PROBLEMS

Joao Paulo Campos Maia Pinto

DATA ANALYTICS BOOTCAMP Final Project

EXECUTIVE SUMMARY

The objective of this project is to demonstrate how I am able to apply the advanced techniques and skills learned throughout this course in developing scripts and solutions that help me solve a range of different problems at work.

I am grouping 4 different projects that collect and transform data inputs into deliverables to clients and impact the daily routines, immensely decreasing the time for performing certain tasks and greatly reducing risks of human error when sending information to clients.

It is also worth noting that this allows a better preparation for business growth as increase in volumes traded have marginal impact on the performance of the solutions.

PROJECT SCOPE

NI 24-101 Compliant Contact to Clients

OVERVIEW

1. Project Background and Description

The NI 24-101 is a norm established by the Ontario Securities Commission that regulates institutional trade matching and settlement and demands that participants have their trades matched in CDS by noon (12pm) the day prior to the settlement date. In order to achieve that, the Middle Office must search for all trades that are still unmatched in the morning and reach out to the clients requesting them to advise their custodians to match our instructions in CDS. The idea of the project is to create a python script that will investigate the trades that require action and will automatically group them by client and prepare an email with the trade details and client's email address in order to advise customers in a timely manner. This will reduce the time of execution of this task astonishingly and will diminish risk of not capturing all trades. It should also decrease the number of errors of sending incorrect information or even more important, sending the information to the wrong counterparty.

2. Project Scope

- Part 1. Reading and filtering data :
  - Read a csv file that is extracted using the system Arrow with filters that are currently in place.
  - Filter the dataframe's memo1 field to exclude etfs and other trades not covered by this team, removing items that start with "REF#", "REF #", "NO CLIENT", "YIELD" and "ETF TRADE"
  - Filter the dataframe's security field to exclude securities not covered by this team, removing items with code equals to "CA99997Z1099"
  - Filter the dataframe's date field to exclude today's trades
  - Cast column Client Account as string

- Part 2. Merging data with account information:
  - Read csv file generated daily that contains all institutional accounts information
  - Cast column containing account number as string
  - Left merge data with accounts dataframes on account number columns
  - Sort by status, client name and security
  - Move client name to first column
  - Save excel spreadsheet with information

- Part 3. Getting list of clients that need to be contacted:
  - Filter merged DF status column for unmatched trades only
  - Get list of unique clients from DF
  - For loop through client's DF to filter by client, convert table to html and append it a new list
  - Create dataframe with list of unique clients and html table
  - Read csv file containing contact information by client
  - Left merge the two DFs on client name

- Part 4. Preparing emails and finish task:
  - Dispatch outlook application
  - For loop through unique clients merged DF creating new email for each client
  - Dispatch excel with created spreadsheet

3. High-Level Requirements

- Permission to access Arrow
- Permission to access and save network I: drive
- Ability to run python scripts

4. Deliverables

- Email to each client that has unmatched trades
- Spreadsheet with all unmatched and DKed trades