# Project notes for automated trading algorithm

-Leslie Cuadra

##introduce project scaffold##

In: Pandas Stocks List

Hidden process: Select, filter, and sort methods

Out: Trade execution (token refresh program needed)

##define functions##

Filter\_Sort function:

Get top stocks

Screen

JSON, Pandas

Ameritrade (refresh token script), Yahoo

Trade\_Execution function:

If S < MA, buy if S-MA>

If S<MA, sell if MA-S>

##declare conditions##

Weights, rates and triggers:

Fundamental weights🡪high volume, price range, delta Greeks

Technical indicator weights🡪buy and sell signals

##describe gradient descent of total error##

TE=Target return rate – actual return rate

How to determine if trade was profitable?

Max {(sell-buy)/buy}

Delta Zero Weight control (DzWc)

##set zero change function for technical indicator weights##

If slope(TI) = 0, then

Buy if TI < S

Sell if TI > S

Where S is the Stock and TI is the technical indicator

###elucidate weight adjustment

Find max % ΔL

Where L is the total loss of each S

Gradient Descent

Layout:

Folder

Executable python program

Config

Temp, etc

Executable:

Import modules

Definitions

Expressions

Actions

Stoppage and Maintenance

Framework:

Tabular Data

Select

Execute (1 week token constraint)

Linear algorithmic flow: get\_data\_build\_frame\_execute\_trade

Triggers and weights:

If Δp/ Δt = 0:

Buy:

If S>MA by S/MA>Tb

Sell

If S<MA by S/MA<Ts

Trade Execution Variables: symbol, price, date, quantity, type1(buy or sell), type2(call or put) – 6 vars

TEV contains 6 vars to be cleaned and pruned by pandas data-reader

TEV method added to data frame

Execution function invokes data frame to trade on selected strings

CSV file should be dynamic: Get movers add stocks to it, so does get options, auto updates.

Stock search should aid option search by acting as a second filter for it: only searching for options for stocks that make the target dataframe.

Get\_movers, to csv file 1

Get\_stocks, to csv file 2

Get\_options, to csv file 3

Trade\_execute from final csv file

Need more candidates for testing to put together different components\*

References: Big Thanks To