

Timeseries README

Varun Varanasi

March 14, 2022

The goal of this document is to summarize the datatypes and functionality of the TimeSeriesPlot.py script. This script assumes that it is in the same working directory as the Data folder which includes specially named files required for the functions to work. Data in the data folder is updated according to the the DownloadCSV.py script which is run hourly by a cronjob.

Functions

ts_plot

This function creates a timeseries plot given a financial index, a start date, and variables to plot.

Inputs:

- **data_name:** Name of financial index you would like to plot (see Available Datasets)
- **start_date:** Start date of the timeseries (syntax note: YYYY-MM-DD format)
- **variables:** Select which variables from the dataset you would like to plot (syntax note: put all variables in a list)
- **fill:** Select 1 to create a fill plot (timeseries value duplicated over x-axis and filled in to better visualize change)

Outputs: A series of python plots detailing the timeseries at the specified conditions and variables

ts_decomp

This function decomposes a given financial index into its trend, seasonal, and noise components by both multiplicative and additive methods.

Inputs:

- **data_name:** Name of financial index you would like to plot (see Available Datasets)
- **start_date:** Start date of the timeseries (syntax note: YYYY-MM-DD format)
- **variables:** Select which variables from the dataset you would like to plot (syntax note: put all variables in a list)
- **period:** Determines what the period of the timeseries analysis is

Outputs: A series of python plots detailing the timeseries decomposition at the specified conditions and variables.

ADH_test

The ADH Test is a test of stationarity of a timeseries. This function runs the ADH test on given financial index, start date, and variables. Generally, the rejection criteria is at a p-value less than 0.05

- **data_name:** Name of financial index you would like to plot (see Available Datasets)
- **start_date:** Start date of the timeseries (syntax note: YYYY-MM-DD format)

- **variables:** Select which variables from the dataset you would like to plot (syntax note: put all variables in a list)

Outputs: Variable name, ADF statistic, associated p-value and critical values.

seasonality_test

The `seasonality_test` function creates a plot detailing the seasonality of the timeseries given. Expected output will highlight peaks at regular intervals corresponding with the seasonality of the data.

- **data_name:** Name of financial index you would like to plot (see Available Datasets)
- **start_date:** Start date of the timeseries (syntax note: YYYY-MM-DD format)
- **variables:** Select which variables from the dataset you would like to plot (syntax note: put all variables in a list)

Outputs: Python plot showing an autocorrelation plot for the given timeseries.

interpolate_data

This function adds granularity at a specified interval to the provided timeseries.

- **data_name:** Name of financial index you would like to plot (see Available Datasets)
- **start_date:** Start date of the timeseries (syntax note: YYYY-MM-DD format)
- **variables:** Select which variables from the dataset you would like to plot (syntax note: put all variables in a list)
- **intervals:** Select which interval of granularity you would like to interpolate (30S: 30 second intervals, 1H: 1 hour intervals, etc.)
- **methods:** Select between a "Linear" or a "Seasonal" interpolation (seasonal interpolation is a linear interpolation with random noise corresponding to seasonal variation of the timeseries)
- **period:** Select the period for the timeseries seasonality analysis
- **csv:** Select 1 to save a .csv file of your interpolated data in the "Data" folder

Outputs: Saves .csv of interpolated data in "Data" folder and produces a plot of the interpolated timeseries

Available Datasets

VIX

- VIX is a measure of S&P 500 volatility 30 days from the date in question
- The VIX dataset updated daily and labeled as **VIX_History_DATE_OF_UPDATE.csv** in the Data folder
- The dataset contains **date, OPEN, HIGH, LOW, CLOSE** for each day

VIX9D

- VIX9D is a measure of S&P 500 volatility 9 days from the date in question
- The VIX9D dataset updated daily and labeled as **VIX9D_History_DATE_OF_UPDATE.csv** in the Data folder
- The dataset contains **date, OPEN, HIGH, LOW, CLOSE** for each day

VVIX

- VVIX is a VIX index on the VIX itself
- The VVIX dataset is updated daily and labeled as **VVIX_History_DATE_OF_UPDATE.csv** in the Data folder
- The data set contains **date, VVIX**

VXX

- VXX is a short-term future derivative of the VIX
- The VXX dataset is updated daily and labeled as **VXX_History_DATE_OF_UPDATE.csv** in the Data folder
- The data set contains **Date, Open, High, Low, Close, Adj Close, Volume**

VIX Futures

As of now this data set has not been implemented into the TimeSeriesPlot.py script.