



# ENGIN 604 INTRODUCCIÓN A PYTHON PARA LAS FINANZAS

## DATOS BURSÁTILES UTILIZANDO LA API DE YAHOO FINANCE

**Profesor:** *Gabriel E. Cabrera*

**Ayudante:** *Alex Den Braber*



Las API (Interfaz de Programación de Aplicaciones) permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. Una API muy usada es la de Yahoo Finance, que permite al usuario utilizar y descargar datos bursátiles con distinta frecuencia. En python existe la librería `yfinance` que permite acceder a dicha API. Para instalar `yfinance`:

```
!pip install yfinance
```

Vistar <https://pypi.org/project/yfinance/> para ver la documentación de la librería.

## 1. API Yahoo Finance

1. Descargué la información bursátil de GameStop (GME) desde 2015-01-01 hasta 2021-03-01 con frecuencia diaria.

```
import pandas as pd # importa pandas
import numpy as np # import numpy
import yfinance as yf # import ytfinance

# ticker
asset = "GME"

# descarga la información OHCL del GameStop
gme_ohcl_daily = yf.download(asset, start = "2015-01-01", end = "2021-03-01")

## [*****100%*****] 1 of 1 completed
```

2. Realice una breve estadística descriptiva que incluya: el total de observaciones, promedio, desviación estandar, mínimo, máximo, percentil 25, 50 y 75.
3. Extraiga del índice el año, mes y día.
4. Seleccione el precio al cierre (Close).

## 2. Cambios a través del tiempo

1. Utilizando una función lambda (anónima) genere la diferencia del precio al cierre:

$$\Delta_t = p_t - p_{t-1}$$

Donde  $p_t$  es el precio al cierre en  $t$  y  $p_{t-1}$  el precio al cierre en  $p_{t-1}$  (rezago).

2. Utilizando una función lambda (anónima) genere el primer rezago de  $p_t$ .

3. Divida la variable creada en (1) por la variable creada en (2).
4. Utilize el método `pct_change()` y compárelo con la variable creada en (3).

### 3. Visualización Básica de Series de Tiempo

1. Grafique el precio al cierre de GameStop.

#### 3.1. Análisis Técnico: Media Movil

1. Utilizando el precio al cierre de GameStop (GME) genere:

- a. El precio mínimo movil a 20 días.
- b. La media movil a 20 días (corto plazo).
- c. El precio máximo movil a 20 días.

Elimine los `NAs` y grafique las variables creadas desde 2020-01-01 hasta 2020-09-08.

2. Al DataFrame creado en (1), genere:

- a. La media movil a 252 días (largo plazo).
- b. Una variable que sea igua a 1 si media movil a 20 días > media movil a 252 días, -1 caso contrario.

Elimine los `NAs` y grafique las variables creadas desde 2020-01-01 hasta 2020-09-08.

3. Filtre los datos OHCL de GameStop entre 2020-01-01 y 2020-09-08. Luego utilizando la librería `plotly` grafique un *Candle-Stick* (ver <https://plotly.com/python/candlestick-charts/>).

```
# importa plotly
import plotly.graph_objects as go

# filtra los datos OHCL
part_i = (gme_ohcl_daily.index >= "2020-01-01") # primer filtro
part_ii = (gme_ohcl_daily.index <= "2020-09-08") # segundo filtro
df_plotly = gme_ohcl_daily[part_i & part_ii]

# grafica candlestick
fig_plotly = go.Figure(data=[go.Candlestick(x=df_plotly.index,
                                             open=df_plotly['Open'],
                                             high=df_plotly['High'],
                                             low=df_plotly['Low'],
                                             close=df_plotly['Close'])])

# se visualiza
fig_plotly.show()
```

### 4. Descargar Múltiples índices

1. Utilizando la librería `yfinance`, descargue con frecuencia mensual la información OHCL desde 2000-01-01 hasta 2021-03-01 de los siguientes índices bursátiles: Facebook (FB), Amazon (AMZN), Apple (AAPL), Netflix (NFLX) y Google (GOOG).
2. Grafique la evolución de los precios para todos los índices.
3. Realice un *reshape* al DataFrame generado en (1) de manera que las columnas queden en las filas.
4. Devuelva el DataFrame anterior a su forma original.