```
import statistics as stats
from pandas import *
from pandas_datareader import data
from numpy import *
from matplotlib.pyplot import *
set_option ("display.max_rows", 20000)
set_option ("display.max_columns", 1000)
set_option ("display.width", 1000)

num_periods = 10   # number of days over which to average
K = 2 / (num_periods + 1)
ema_p = 0
ema_values = []

start_date = "2019-01-01"
end_date = "2020-03-05"
google_data = data.DataReader( "GOOG", "yahoo", start_date, end_date )

for close_price in google_data ["Adj Close"]:
    if ema_p == 0:
        ema_p = close_price
    else:
        ema_p = (close_price - ema_p) * K + ema_p

    ema_values.append(ema_p)

google_data = google_data.assign (Exponential20DayMovingAverage = Series
(ema_values, index=google_data.index)).ffill (axis=0)

ema_difference = google_data ["Exponential20DayMovingAverage"] - google_data ["Adj
Close"]
google_data = google_data.assign (Difference = Series (ema_difference,
index=google_data.index)).ffill (axis=0)
ema = google_data ['Exponential20DayMovingAverage']

time_period = 10   # number of days over which to average
history = []
sma_values = []

for close_price in google_data ["Adj Close"]:
    history.append(close_price)
    if len (history) <= time_period-1:  # because we only average over last
"time_period" prices
        sma_values.append(None)
    elif len(history) == time_period:
        x = stats.mean(history)
        sma_values.append(x)
        history.clear()

google_data = google_data.assign (Simple20DayMovingAverage = Series (sma_values,
```

```
index=google_data.index)).ffill (axis=0)

google_data ["daily_difference"] = google_data ["Adj Close"].diff()
google_data ["signal"] = 0.0
google_data ["signal"] = where(google_data ["daily_difference"] > 0, 1.0, 0.0)

def trading_support_resistance(data, bin_width=20):
    data["sup_tolerance"] = Series( zeros( len( data ) ) )
    data["res_tolerance"] = Series( zeros( len( data ) ) )
    data["sup_count"] = Series( zeros( len( data ) ) )
    data["res_count"] = Series( zeros( len( data ) ) )
    data["sup"] = Series( zeros( len( data ) ) )
    data["res"] = Series( zeros( len( data ) ) )
    data["positions"] = Series( zeros( len( data ) ) )
    data["signal"] = Series( zeros( len( data ) ) )
    in_support = 0
    in_resistance = 0

    for x in range ( (bin_width - 1) + bin_width, len( data ) ):
        data_section = data[x - bin_width:x + 1]
        support_level = min( data_section["Exponential20DayMovingAverage"] )
        resistance_level = max( data_section["Exponential20DayMovingAverage"] )
        range_level = resistance_level - support_level
        data["res"][x] = resistance_level
        data["sup"][x] = support_level
        data["sup_tolerance"][x] = support_level + 0.2 * range_level
        data["res_tolerance"][x] = resistance_level - 0.2 * range_level

        if data["Exponential20DayMovingAverage"][x] >= data["res_tolerance"][x]
and\
                                    data["Exponential20DayMovingAverage"] [x] <=
data["res"][x]:
            in_resistance += 1
            data["res_count"][x] = in_resistance
        elif data ["Exponential20DayMovingAverage"] [x] <= data ["sup_tolerance"]
[x] and\
                                    data ["Exponential20DayMovingAverage"] [x] >=
data ["sup"] [x]:
            in_support += 1
            data["sup_count"] [x] = in_support
        else:
            in_support = 0
            in_resistance = 0

        if in_resistance > 2:
            data["signal"][x] = 1
        elif in_support > 2:
            data["signal"][x] = 0
        else:
            data["signal"][x] = data["signal"][x-1]
```

```
    data["positions"] = data["signal"].diff()

trading_support_resistance(google_data)
print (google_data)

fig = figure()
ax1 = fig.add_subplot (111, ylabel="Google price in $")
google_data ["Adj Close"].plot (ax=ax1, color="g", lw=2., legend=True)
google_data ["Simple20DayMovingAverage"].plot (ax=ax1, color="r", lw=2.,
legend=True)
google_data ["Exponential20DayMovingAverage"].plot (ax=ax1, color="b", lw=2.,
legend=True)
google_data ["Difference"].plot (ax=ax1, color="y", lw=2., legend=True)
ax1.plot( google_data.loc[google_data.positions == 1.0].index,
google_data["Exponential20DayMovingAverage"][google_data.positions == 1.0], "^",
markersize=7, color="k", label="buy" )
ax1.plot( google_data.loc[google_data.positions ==
-1.0].index,google_data["Exponential20DayMovingAverage"][google_data.positions ==
-1.0], "v", markersize=7, color="k", label="sell" )
legend ()
show ()

initial_capital = float (1000.0)
positions = pandas.DataFrame(index=google_data.index).fillna(0.0)
portfolio = pandas.DataFrame(index=google_data.index).fillna(0.0)
positions ["GOOG"] = google_data ["signal"].fillna (0.0)
portfolio ["positions"] = (positions.multiply(google_data
["Exponential20DayMovingAverage"], axis=0)).fillna (0.0)
portfolio ["cash"] = initial_capital - (positions.diff().multiply(google_data
["Exponential20DayMovingAverage"], axis=0)).cumsum()
portfolio ["total"] = portfolio ["positions"] + portfolio ["cash"]

figure_1 = figure ()
ax1 = figure_1.add_subplot (111, ylabel="Total")
google_data ["Exponential20DayMovingAverage"].plot (ax=ax1, color="g", lw=4.)
portfolio ["positions"].plot (ax=ax1, color="b", lw=4.)
portfolio ["cash"].plot (ax=ax1, color="r", lw=4.)
portfolio ["total"].plot (ax=ax1, color="y", lw=4.)
print (portfolio["total"])
legend ()
show ()
```