Carnegie Mellon University Africa

COURSE 18-785: DATA, INFERENCE & APPLIED MACHINE LEARNING

ASSIGNMENT 1

Nchofon Tagha Ghogomu

ntaghagh

September 2, 2024

LIBRARIES:

The following libraries were used.

- Math
- Pandas[1]
- Numpy[2]
- Pyplot from Matplotlib [3]
- Matplotlib.ticker from MultipleLocator - for defining axis spacing

Programming Language:

- Python

INTRODUCTION

This assignment was made of ten practical questions that depicted problems commonly encountered in the real world. We had to analyse the question, reason out a solution, come out with an algorithm that solves the problem and implement this algorithm to obtain a solution. Some of the mathematical concepts explored in this assignment include:

- Simple interest,
- Compound interest,
- Rate of change (growth and decay rates)
- Mean, Maximum value, and Minimum Value.

Python was the programming language used and Jupyter notebook was the programming environment.

SOLUTIONS

Question 1:

1) Goal:

Calculate number of folds that will make height exceed Mount Everest at 8,848 m.

2) Reasoning:

Each fold makes the paper's height double the current height. Thickness was converted to meters. Using a while loop, the 'number_of_folds' count was incremented by 1 unit at every stage that the thickness of the paper was doubled. When the thickness('paper') surpasses 8848, the number of folds value is returned.

3) Result/Output:

Number of folds:
24

4) Observation:

Though the thinness of the paper is swimmingly very small (0.001m in this case), just 24 folds can make it thicker that 8848m: the power of exponential growth.

Question 2:

1) Goal:

Calculate time for volume to decrease to less than half.

2) Reasoning:

The equation that simulates this rate is given. The initial volume is given a unit value for convenience and with the help of a while loop, the current volume value is calculated, and the time units is increased by on. When the current volume is less than half of v0, the value of t is returned.

3) Result/Output:

Time to decrease to less than half:
7 units

4) Observation:

It takes 7t units to for a reservoir emptying at this rate to get to half of its initial volume.

Question 3:

1) Goal:

Calculating compound interest in 1, 2, 3, to 5 years for a $100 capital and 5%yearly interest rate. Note: Value rounded up to the nearest $ as instructed.

2) Reasoning:

For every new year, the interest is calculated on the initial value of the previous year plus the interest made for that year. Using a while loop, for every year, we calculate the profit and add it to that year's capital. The interest calculation for the subsequent year will be based on the revenue (principal plus profit) of the current year.

3) Result/Output:

Yearly revenue for 1st to 5th year respectively:
105
110
116
122
128

4) Observation:

In five years, the investor will make a profit of $28. The longer one stays, the more cumulative profit one acquires.

Question 4:

1) Goal:

Calculate monthly payment if depth is paid off in one, two or three years.

2) Reasoning:

The is a monthly interest rate of 1%. In this case, we calculate the compound interest for one year (12 months), two years (24 months), and three years (36 months). To obtain the total monthly payoffs, we divide the yearly payoff by the number of months. This was accomplished with the help of for loops.

3) Result/Output:

The monthly pay off respectively in 1, 2, and 3 years:
1878
1192
1137

4) Observation:

Though the interest on payoff increases (at this rate) with the number of years, this payoff is distributed over several months causing monthly payment to decrease slightly.

Question 5:

1) Goal:

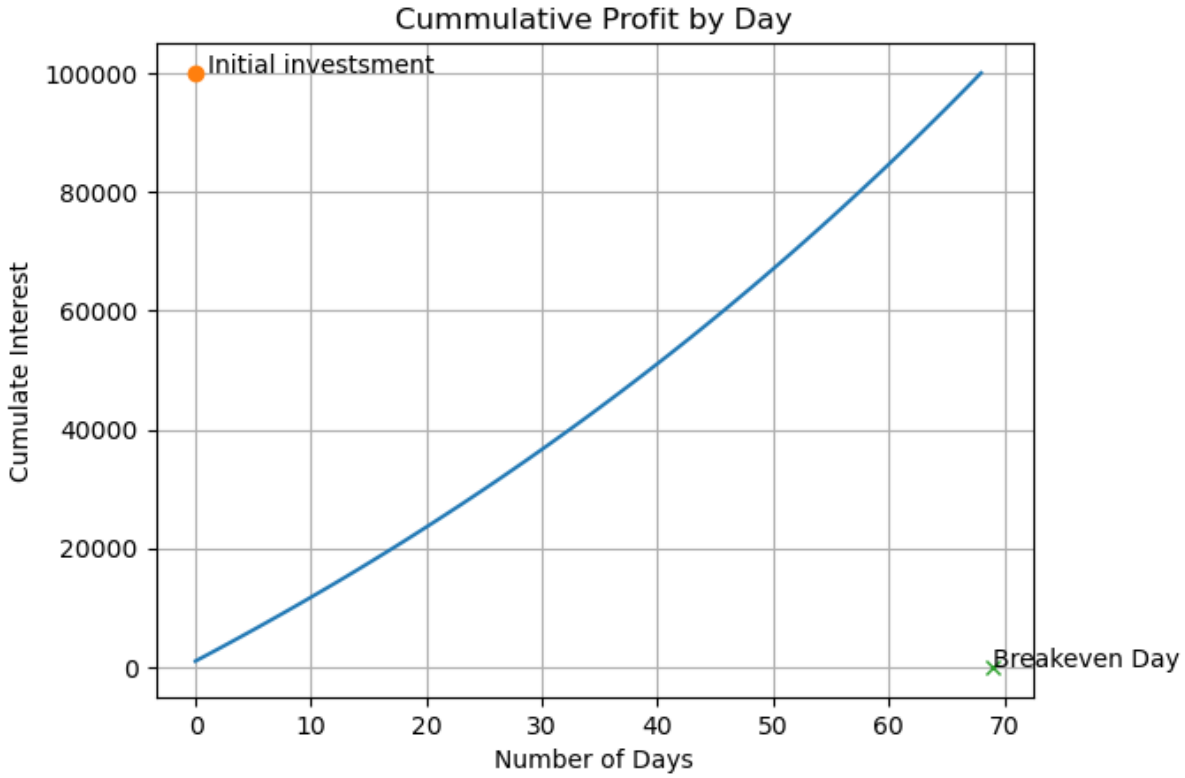
Calculate number of days to repay initial investment and plot a graph of cumulated profits indicating initial investment and break-even day.

2) Reasoning

The profit per customer is given and the daily growth rate of customers is given. Based on this, we can calculate the daily customer growth and deduce the daily profit made by the costumers. Daily profit was appended to a list named 'interest' and was later plot against days. Note: Customers were rounded up to the nearest whole number.

3) Result/Output:

Number of days:
69



4) Observation:

The cumulative profit/customer growth is a convex function, and it takes 69 days to break even when customers are rounded to the next whole number and 70 when costumers are not rounded to a whole number.

Question 6:

1) Goal:

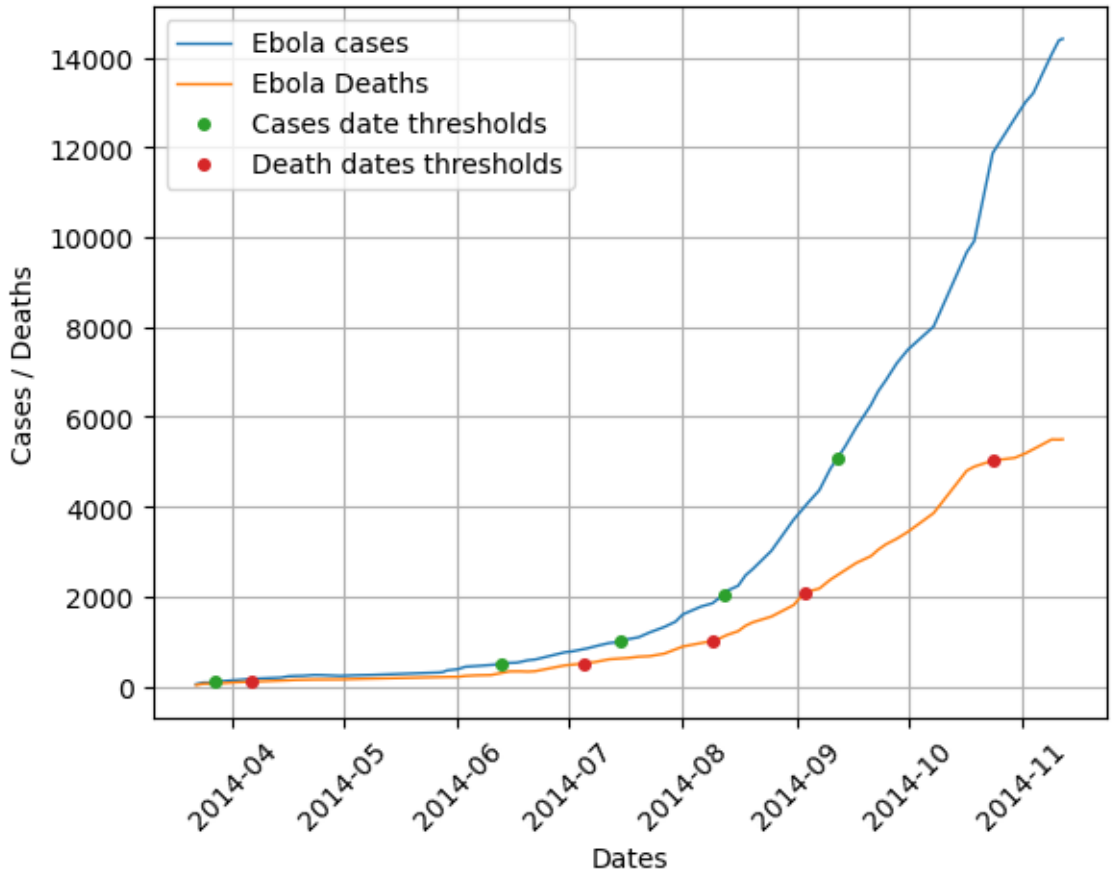
Perform linear interpolation on data to fill missing values, plot new data with interpolation, and find dates when deaths and cases exceeded the given threshold points.

2) Reasoning:

The date range is readjusted to uncover missing/omitted dates and with the help of linear interpolation, the missing value for the corresponding dates is filled. This new data frame is plotted with the new interpolated data.

3) Result/Output:

Death Toll Dates:
2014-04-06
2014-07-05
2014-08-09
2014-09-03
2014-10-24
Case Toll Dates:
2014-03-27
2014-06-13
2014-07-15
2014-08-12
2014-09-12



5) Observation:

Overall, the number of cases and deaths increase exponentially within the observed time frame. There was a slight difference between number of deaths and cases for the first three months, and later diverged. On average, it took the number of cases longer to cross the set threshold values.

Question 7:

1) Goal:

Calculate average growth rate (in %) of Ebola deaths and cases

2) Reasoning:

The daily growth rate of both cases and deaths in percentages are computed but looping through everyday and calculating cumulative increase. In the end, the average of these values is extracted.

3) Result/Output:

Average death growth rate (in %) per day (in 5 dp):

2.33061

Average Cases growth rate (in %) per day (in 5 dp):

2.50652

4) Observation:

The average growth rate of both cases and deaths don't differ widely with death rates being greater.

Question 8:

1) Goal:

Plot deaths against cases and compute the ratio of deaths to cases.

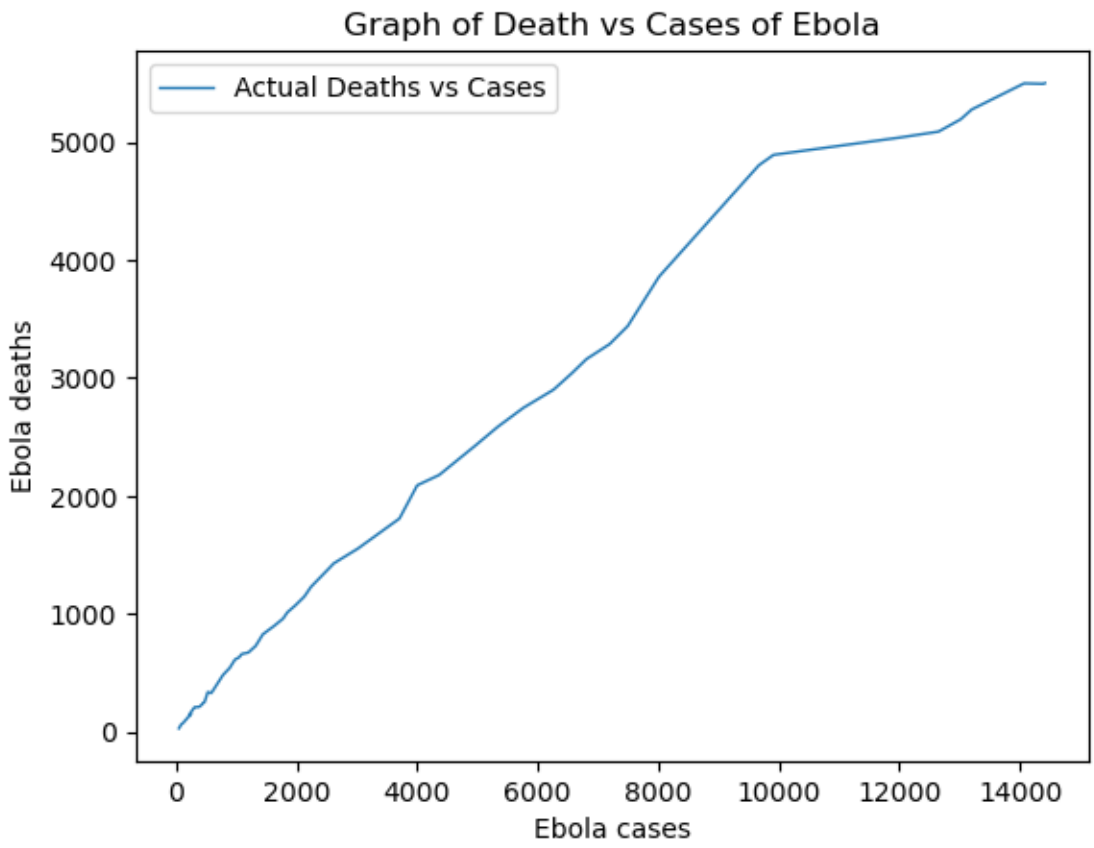
2) Reasoning:

For every row, we dived the number of deaths by the number of cases and get the average of this value.

3) Result/Output:

Ratio of deaths to cases (5 decimal places):

0.55855



4) Observation:

The ratio estimates to; for every death, there are 2 cases. Beyond, the scope, a fitted line was plotted and the gradient of the line varied slightly from this calculated ratio.

Question 9:

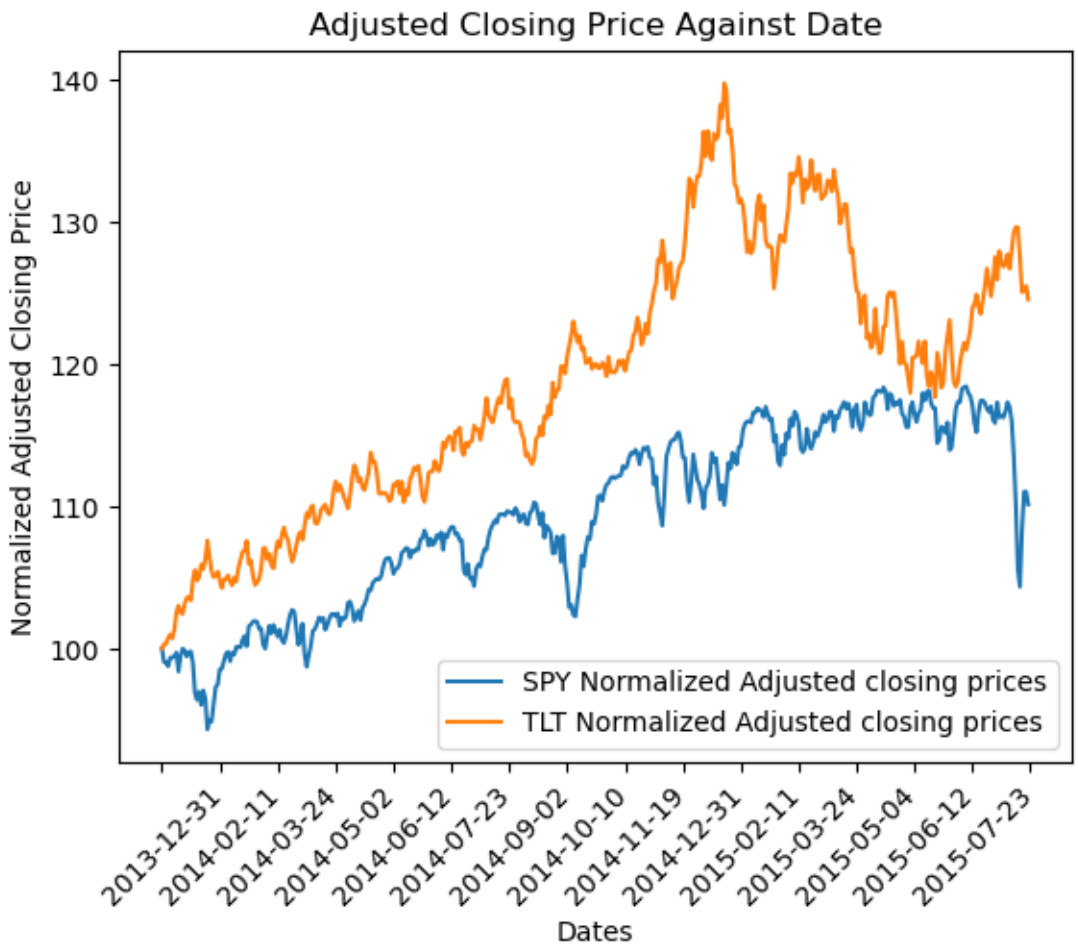
1) Goal:

Plot two time series of SPY and TLY of Adjusted closing prices and make them comparable starting from $100

2) Reasoning:

To make them comparable about $100, both data were normalised. For every value of Adjusted closing price, it was divided by the initial price and multiplied by 100. This help us determine the degree to which individual prices vary from initial price.

3) Result/Output:



4) Observation:

The increase in closing price tends to be steeper for TLT than for SPY within the observed period.

Note: This data was not interpolated, and some dates were missing. This was done this way strictly following the directives of the question.

Question 10:

1) Goal:

Calculate mean, max and mean for each of the two ETFs.

2) Reasoning:

Using the formular given, and the inbuilt function pct_change(), we calculate the daily returns and get the mean, max, and min for both ETFs.

3) Result/Output:

SPY Mean, Max, and Min respectively(to 5 dp):
0.00026

0.03839
-0.04211
TLT Mean, Max, and Min respectively (to 5 dp): zxr
0.00056
0.02647
-0.02432

4) Observation

For the SPY STF, the mean daily return is at 0.00026 with a maximum value of 0.03 and a minimum value of -0.04. On the other hand, for TLT, the mean daily value seats at 0.00056, with a maximum value of 0.03, and a minimum value of -0.02

References

[1] "pandas documentation — pandas 2.2.2 documentation." Accessed: Sep. 02, 2024. [Online]. Available: https://pandas.pydata.org/pandas-docs/stable/index.html
[2] "NumPy -." Accessed: Sep. 02, 2024. [Online]. Available: https://numpy.org/
[3] "Matplotlib documentation — Matplotlib 3.9.2 documentation." Accessed: Sep. 02, 2024. [Online]. Available: https://matplotlib.org/stable/