Stock Market Prediction Using Difference Equations and Natural Language Processing

Zachary M. Robers

University of North Carolina, Chapel Hill

MATH 383: First Course in Differential Equations

Dr. Roberto Camassa

April 29, 2022

Abstract

The aim of this paper is to discuss the creation of stock market modeling software using difference equations, differential equations, and a simplistic natural language processing model. The research methodology involved expressing common stock market patterns through difference equations, an analysis to find the correlation between a specific stock's history and standard stock patterns, using a second order differential equation to evaluate the volatility of a stock, quantifying the impact of news articles and social media discussions on a stock through an NLP model, and combining these factors to produce a predictive algorithm for the stock's future. While equations and algorithms used in this paper are arbitrarily defined, the idea that difference equations and NLP could be used in unison to predict stock changes in real time is proven to be a legitimate possibility given the success of modeling the Allstate Insurance Company's stock throughout the paper.

Keywords: difference equations, matrix differential equations, natural language processing, stock patterns, stock prediction

Stock Market Prediction Using Difference Equations and Natural Language Processing

The stock market is no longer controlled by a select group of Ivy League graduates who strut to work in personally tailored suits and enter skyscrapers adorned with the names of JP Morgan, BlackRock, Citigroup, and Goldman Sachs. Rather, it is the schoolteacher dressed in pajamas and armed with an investment app that is a part of the fastest-growing sector of the world investment economy.

Over the past two years, retail investors have not shied to exert their influence over global markets. In early 2021, Reddit-inspired amateur investors flocked to scoop up shares of struggling entertainment companies, GameStop and AMC, thus creating the term “meme stock.” The ability of retail investors to congregate and lead a coordinated effort to boost the prices of stocks up to 1,700% of their original value came as a great shock to Wall Street investors who logically bet on the downfall of companies that were destined to fail in an era marked by a raging pandemic.

But along with volatility, retail investors also bring predictability. Since individual investors lack the funding to significantly impact share prices, they must congregate in substantial numbers to sway the market. Such congregation requires communication through a public forum, thus leaving a trail of evidence detailing the retail investors' intentions. This trail of evidence can be efficiently and accurately analyzed by artificial intelligence through a natural language processing model, opening the opportunity for the impact of retail investors on the market to be known before they even place their trades.

Beyond social media interactions, an NLP model focused on market impact can also be utilized to predict the influence of news publications on an individual stock or the entire market. While Wall Street is beyond concentrating on what news publications, whether financial or general, mean for the market, individual investors have the tendency to base their investments on what they read in the news.

Before gauging the impact of news sources or social media communications on the market, it is essential to have a baseline modeling system for a certain company’s stock. Luckily, the graphs of most stocks typically resemble a stock pattern that can be used to loosely predict the stock’s future performance. Additionally, to determine how much of an effect the results of the NLP model will have on a stock, one must know if the stock is generally responsive to changes in the global economic environment. This necessitates the use of a second order differential equation to evaluate the volatility of a stock. Through a combination of the results of the NLP model, a stock's correlation to typical stock patterns, and its volatility index, one can model a stock's future performance to a moderate degree of accuracy.

Modeling Stock Patterns through Difference Equations

Difference Equations

The ever-changing, unpredictable nature of a stock graph can be most easily represented by difference equations as the principal factor in a stock's future price is its price in the present. Additionally, the use of difference equations allows the stock analyzer to easily add the influence of a variety of different coefficients as the stock prediction model advances through time. While differential equations can be a helpful tool when analyzing specific segments of a stock's history, a recursive modeling system, such as difference equations, is necessary for proactive modeling over extended periods of time.

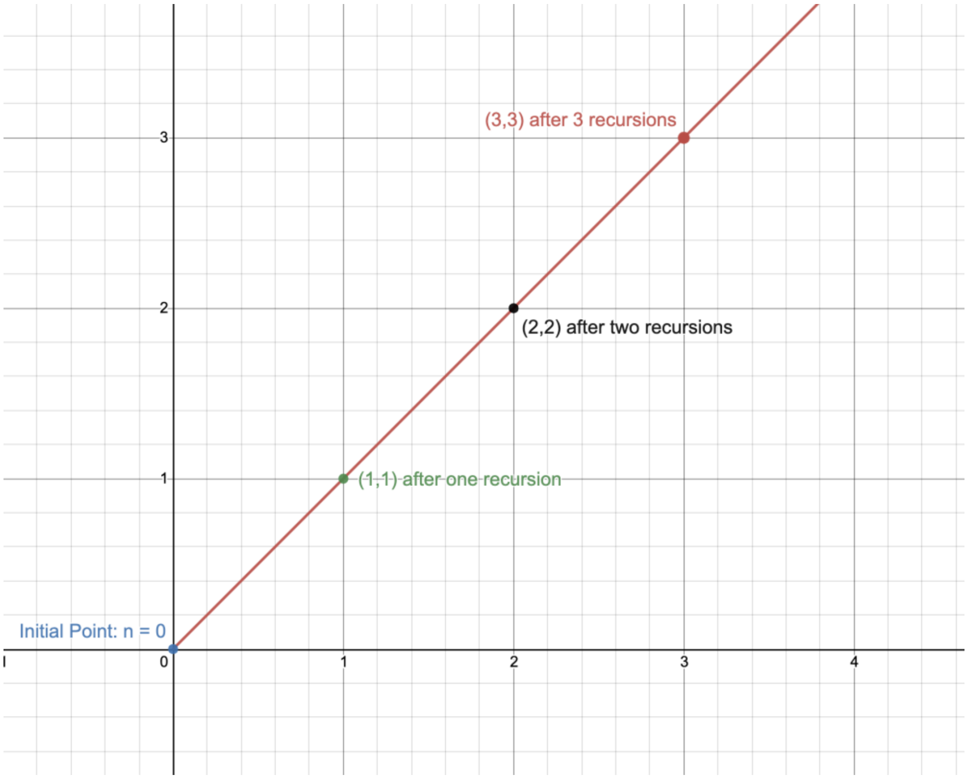
Difference equations can be used to model all types of functions as an alternative to continuous differential equations. The general first order difference equation can be written as:

$$y_{n+1} = f(n, y_n)$$

As demonstrated by this equation, difference equations are recursive and depend on the previous value and the location within the series of values.

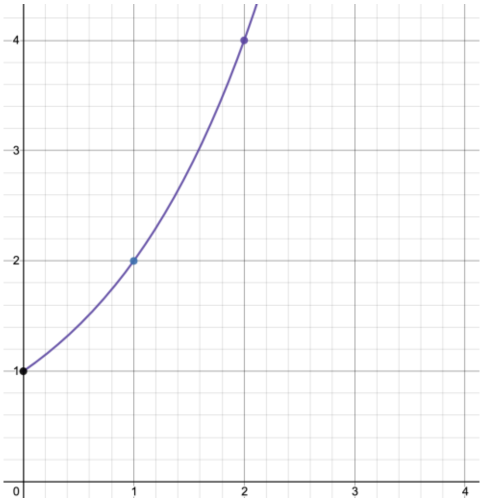
The above equation can be simplified to model linear growth. In the case of linear growth, a constant is simply added to the previous value in the series to demonstrate how the function changes at a constant rate:

$$y_{n+1} = y_n + C_1 , C_1 = 1 \, .$$



The example above demonstrates how a difference equation can be used to model the basic function y = x. Difference equations for linear modeling will be the basis of the stock prediction algorithms used in this report. There are many other difference equations that are essential to mathematical understanding and useful for more complex modeling. Difference equations can also be used to model exponential growth:

$$y_{n+1} = C_1 y_n , C_1 = 2, y_0 = 1$$



In an analogous way to logistic differential equations, logistic difference equations can model population growth with bounds. For illustrative purposes, take the hypothetical example of a beaver population with a carrying capacity of 100 and a growth rate of 3. This information can be plugged into the standard Logisitic difference equation:

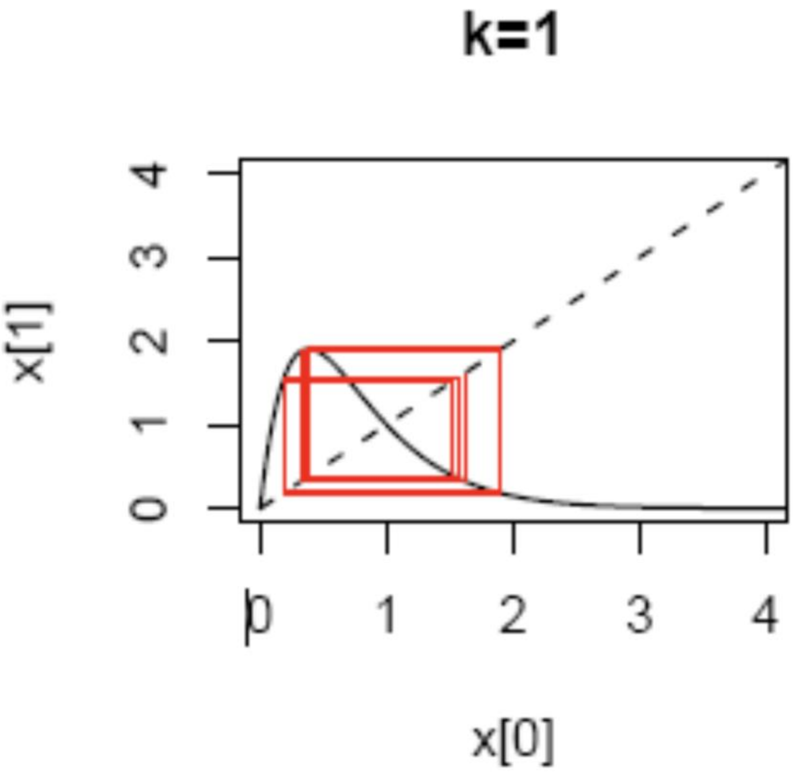
$$l_{n+1} = rl_n (1 - l_n)$$

where r is the growth factor as prescribed by the initial condition and l_n denotes the current recursive value (y_n) divided by the carrying capacity (K). The given situation involving the beaver population can be given by:

$$y_{n+1} = 3y_n (1 - (y_n /100))$$ [5]

The equilibrium solutions can be found in an analogous way to the logistic differential equation by solving for when the next value is equal to the current value. In doing this, one will find two equilibrium solutions for all difference equations. For the beaver example, the two equilibrium solutions are $y_n = 100$ and $y_n = 0$. Another interesting property of logistic difference equations of type [5] is that they can be modeled using a combination of the line y = x and the parabola

$y = rx(1-x)$.



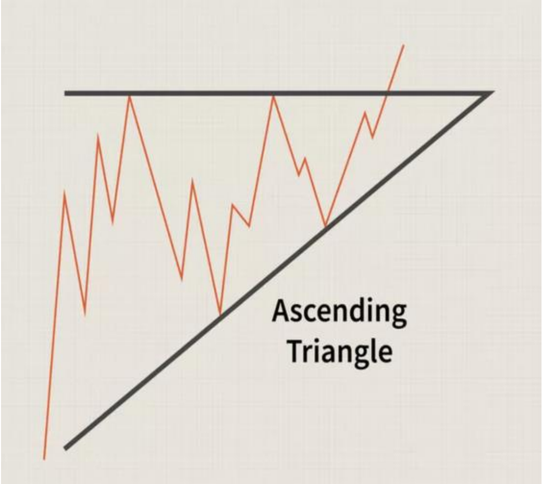
Oftentimes, using the modeling technique demonstrated in the image above can result in unpredictable patterns in which y values oscillate wildly. Such situations are described as chaotic and have been the subject of much recent mathematical research.

The Common Stock Patterns

To later judge the impact of articles and social media discussions on stocks, one must first establish a baseline pattern that can be adapted based on the results of the natural language processing algorithm which will be later discussed. While there are an infinite number of stock patterns, for the purpose of this predictive algorithm, the focus will be on 4 common patterns adapted from CMC Markets. Below, each pattern is accompanied by a piecewise difference equation. Since patterns can occur over various time spans, the difference equation models will be broken up into 10 sections of 10 iterations of the equation. 8 of the 10 sections are for pattern detection and the last 2 are for the continuation of the pattern. The length of the sections in time can later be determined using the graph of a particular stock.

1. The Ascending Triangle

Features: The pattern must include 2 or more relatively equal stock highs, along with 2 or more increasing stock lows mixed in. After exhibiting this pattern, stocks typically trend upwards with a slope near two.



Model:

$$\begin{array}{ll}
y_{n+1} = y_n + 4K & 1 \leq n \leq 10 \\
y_{n+1} = y_n + 4K & 10 < n \leq 20 \\
y_{n+1} = y_n - 3K & 20 < n \leq 30 \\
y_{n+1} = y_n - 3K & 30 < n \leq 40 \\
y_{n+1} = y_n + 3K & 40 < n \leq 50 \\
y_{n+1} = y_n + 3K & 50 < n \leq 60 \\
y_{n+1} = y_n - 2K & 60 < n \leq 70 \\
y_{n+1} = y_n - 2K & 70 < n \leq 80 \\
y_{n+1} = y_n + 2K & 80 < n \leq 90 \\
y_{n+1} = y_n + 2K & 90 < n \leq 100
\end{array}$$

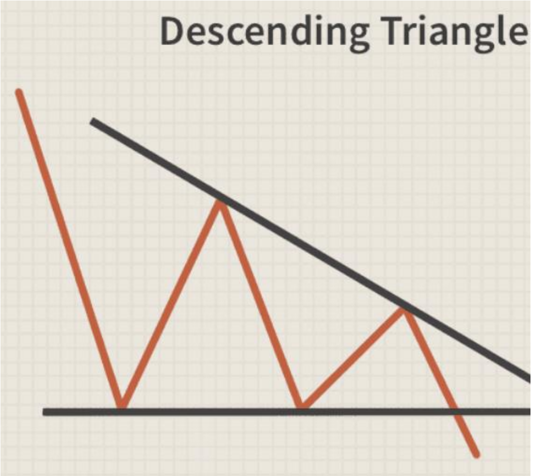
where K is an arbitrary constant to be defined using a specific stock and y_0 is defined as the initial value in a given period of the stock's history. (prescribed conditions)

2. Descending Triangle

Features: This pattern must include 2 or more relatively equal stock lows along with 2 or more decreasing stock highs. After exhibiting this pattern, stocks typically trend downwards with a slope near negative two.

Model:

$$\begin{array}{ll}
y_{n+1} = y_n - 4K & 1 \leq n \leq 10 \\
y_{n+1} = y_n - 4K & 10 < n \leq 20 \\
y_{n+1} = y_n + 3K & 20 < n \leq 30 \\
y_{n+1} = y_n + 3K & 30 < n \leq 40 \\
y_{n+1} = y_n - 3K & 40 < n \leq 50 \\
y_{n+1} = y_n - 3K & 50 < n \leq 60 \\
y_{n+1} = y_n + 2K & 60 < n \leq 70 \\
y_{n+1} = y_n + 2K & 70 < n \leq 80 \\
y_{n+1} = y_n - 2K & 80 < n \leq 90 \\
y_{n+1} = y_n - 2K & 90 < n \leq 100
\end{array}$$

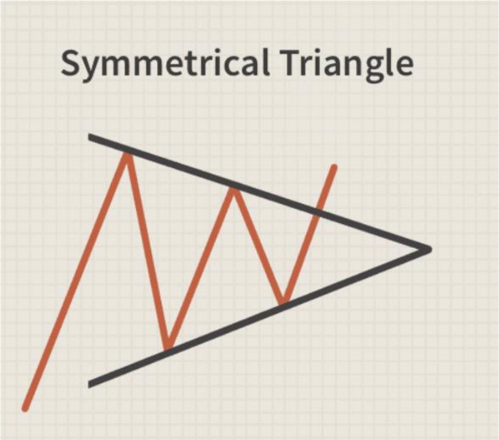


3. Symmetrical Triangle:

Features: This pattern must include 2 or more increasing stock lows alternating with 2 or more decreasing stock highs. Upon exhibiting this behavior, stocks typically increase with a slope near one.

Model:

$$\begin{array}{ll}
y_{n+1} = y_n + 4K & 1 \leq n \leq 10 \\
y_{n+1} = y_n + 4K & 10 < n \leq 20 \\
y_{n+1} = y_n - 3K & 20 < n \leq 30 \\
y_{n+1} = y_n - 3K & 30 < n \leq 40 \\
y_{n+1} = y_n + 2K & 40 < n \leq 50 \\
y_{n+1} = y_n + 2K & 50 < n \leq 60 \\
y_{n+1} = y_n - K & 60 < n \leq 70 \\
y_{n+1} = y_n - K & 70 < n \leq 80 \\
y_{n+1} = y_n + K & 80 < n \leq 90 \\
y_{n+1} = y_n + K & 90 < n \leq 100
\end{array}$$

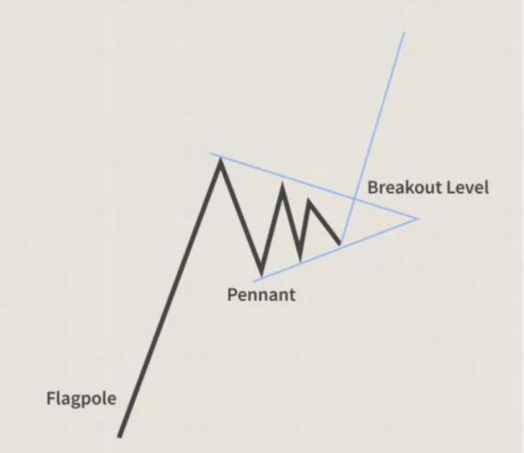


4. Pennant

Features: A pennant pattern is characterized as when a stock exbibits a strong upward trend, but then slows down and starts to form a pennant shape like the symmetric triangle pattern. After displaying this behavior, a stock will typically break out with a slope of 4 (adjusted depending on K).

Model:

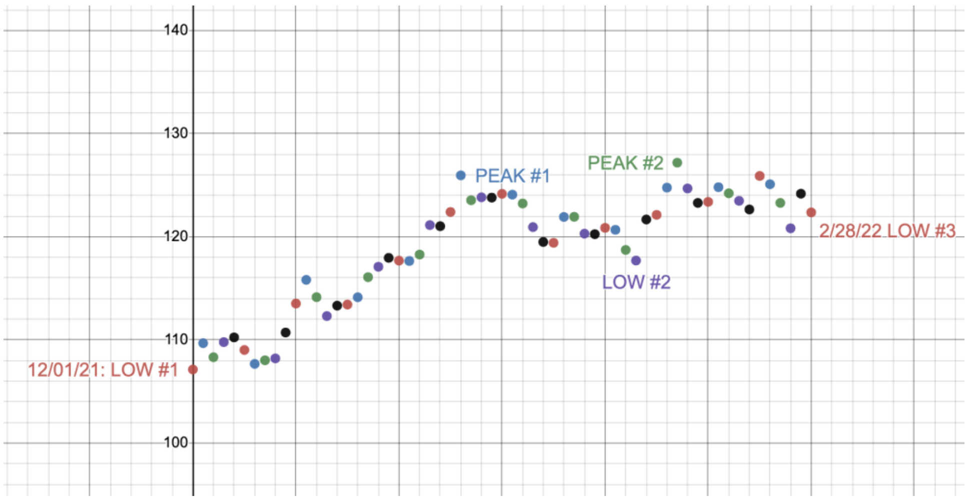
$$
\begin{array}{ll}
y_{n+1} = y_n + 4K & 1 \leq n \leq 10 \\
y_{n+1} = y_n + 4K & 10 < n \leq 20 \\
y_{n+1} = y_n + 4K & 20 < n \leq 30 \\
y_{n+1} = y_n - 2K & 30 < n \leq 40 \\
y_{n+1} = y_n - 2K & 40 < n \leq 50 \\
y_{n+1} = y_n + K & 50 < n \leq 60 \\
y_{n+1} = y_n + K & 60 < n \leq 70 \\
y_{n+1} = y_n - K & 70 < n \leq 80 \\
y_{n+1} = y_n + 4K & 80 < n \leq 90 \\
y_{n+1} = y_n + 4K & 90 < n \leq 100
\end{array}
$$



Pattern Detection

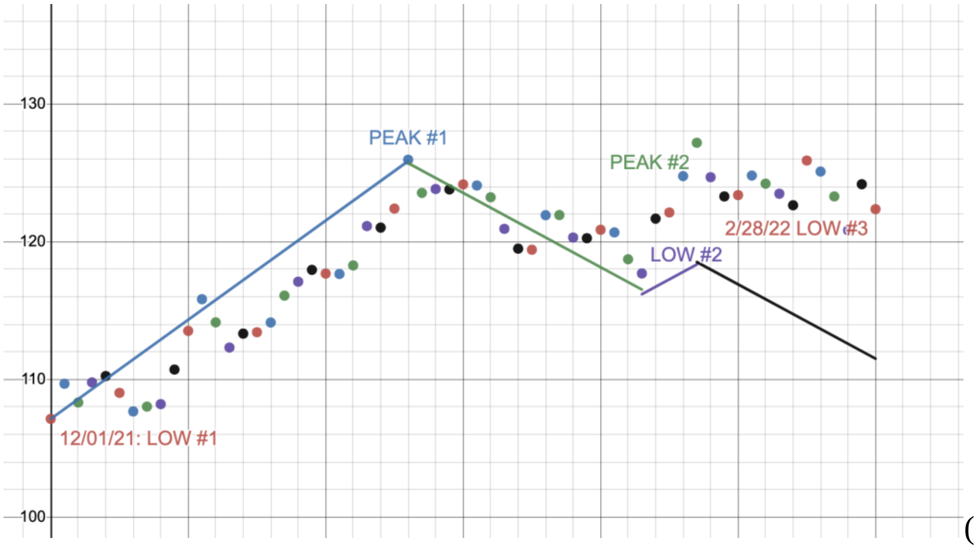
To detect the patterns described and modeled above, one must use data analysis to determine if the features of the stock data correlate with the described pattern. Once a pattern has been detected, the multiplier coefficient can be determined by optimizing the coefficient so that the derivative of the pattern deviates from the actual derivative of the stock graph as little as possible.

As a clear example of pattern detection, one can examine the stock for the Allstate Insurance Company from December 2021 to May 2022. As plotted to the right, the stock experiences a low, a high, a less dramatic low, another similar high, and then an even less dramatic low. This pattern is consistent with the ascending triangle pattern described in the previous section.

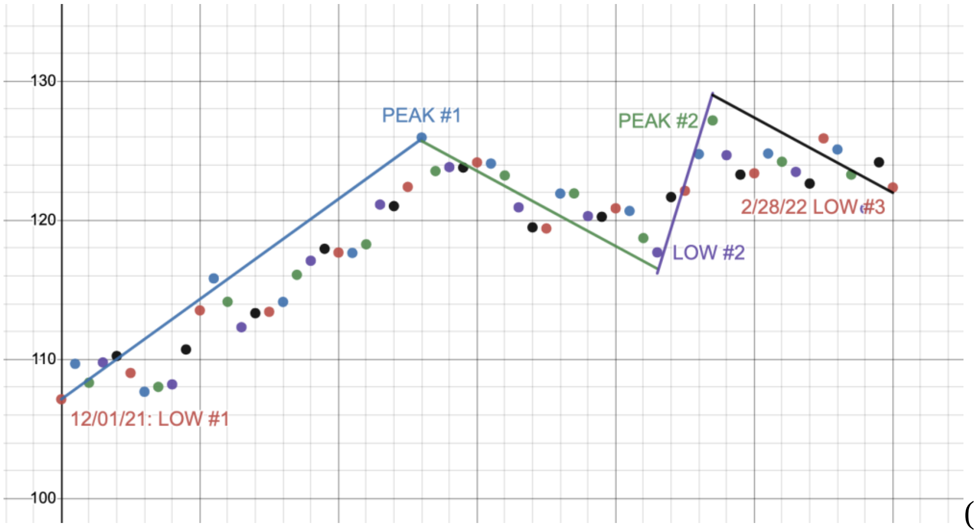


Using a process of trial and error, the coefficient K can be appropriately picked to best resemble the actual behavior expressed by the stock. The model to the right represents the baseline stock pattern that will later be adjusted depending on the stock’s perceived volatility and the results of the NLP model.

By defining K as 0.18 and adjusting the domain of the difference equation model, the difference equation model begins to loosely resemble the stock itself (1). A quick adjustment to the slope of the third segment by a factor of 6 causes the pattern to line up almost identically (2).


130
120
110
100
PEAK #1
PEAK #2
LOW #2
2/28/22 LOW #3
12/01/21: LOW #1

(1)


130
120
110
100
PEAK #1
PEAK #2
LOW #2
2/28/22 LOW #3
12/01/21: LOW #1

(2)

The revised difference equation model is presented below. As shown, the model predicts the linear growth of the stock by a rate of 36 cents per day.

$$\begin{aligned}
y_{n+1} &= y_n + 4(0.18) & 1 \leq n \leq 13 \\
y_{n+1} &= y_n + 4(0.18) & 13 < n \leq 26 \\
y_{n+1} &= y_n - 3(0.18) & 26 < n \leq 34 \\
y_{n+1} &= y_n - 3(0.18) & 34 < n \leq 43 \\
y_{n+1} &= y_n + 3(0.18)(6) & 43 < n \leq 45 \\
y_{n+1} &= y_n + 3(0.18)(6) & 47 < n \leq 49 \\
y_{n+1} &= y_n - 2(0.18) & 49 < n \leq 55 \\
y_{n+1} &= y_n - 2(0.18) & 55 < n \leq 60 \\
y_{n+1} &= y_n + 2(0.18) & 60 < n \leq 70 \\
y_{n+1} &= y_n + 2(0.18) & 70 < n \leq 80 \\
y_0 &= 107.13
\end{aligned}$$

Stock Volatility Index

Traditional stock volatility models rely on a procedure involving the standard deviation of stock prices. However, simplifying volatility down to standard deviation does not account for the way/rate in which the rate of change of the stock is being manipulated by market fluctuations. Simply put, under a standard deviation model, volatility could be high even if a stock follows a smooth linear pattern in which the slope rarely changes.

To account for the changes in the slope of the stock that occur throughout a stock’s lifetime, a differential equation relying on the second order derivative of a stock can be utilized. One should note that the process defined below is primarily theoretical as the first and second derivatives oftentimes are of opposite signs, causing them to cancel out and give a false volatility score. The equation is arbitrarily defined below.

$$20(d^2y/dx^2) + dy/dx = v(x)$$
$$volatility = (5/y_0)|(\int_a^b v(x)\ dx)|/(b-a)$$

Before utilizing this arbitrary equation, one must assess the line of best fit for the original stock graph. In the case of the Allstate Insurance Corporation stock, the values in the scatterplot can be placed into an online algorithm[1] to find a cubic function that best resembles the actual stock graph. The function is shown below:

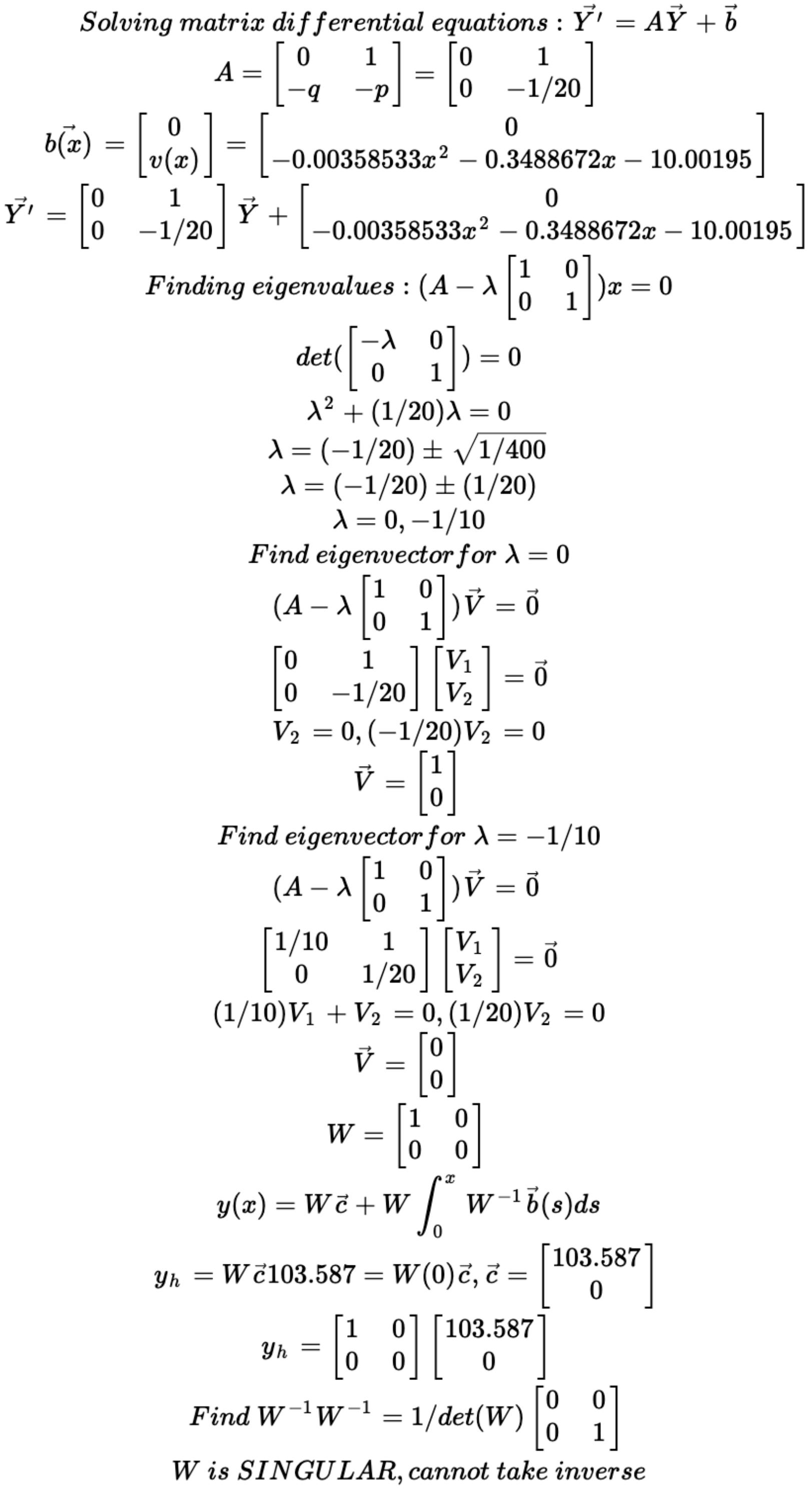
$$y(x) = -0.00119511x^3 - 0.102727x^2 + 3.10605x + 103.587$$

Finding Volatility Using the Line of Best Fit

One can evaluate v(x) for this example using the procedure defined below:

$$dy/dx = -0.00358533x^2 - 0.205454x + 3.10605$$
$$d^2y/dx^2 = -0.00717066x - 0.2054$$
$$v(x) = -0.00358533x^2 - 0.3488672x - 1.00195$$

To verify the function v(x) and demonstrate the procedure to solve matrix differential equations, v(x) can be plugged into the original formula and then y(x) can be evaluated to ensure that it is equivalent to the line of best fit. However, since the x coefficient in the differential equation formula is 0, the Wronskian matrix will be singular and therefore the differential equation is not solvable. This demonstrates a major flaw in the matrix procedure to solve higher order differential equations. Higher order differential equations are only solvable on a case-by-case basis, and this is a prime example of that.



$$Solving\ matrix\ differential\ equations: \vec{Y'} = A\vec{Y} + \vec{b}$$

$$A = \begin{bmatrix} 0 & 1 \\ -q & -p \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -1/20 \end{bmatrix}$$

$$b(\vec{x}) = \begin{bmatrix} 0 \\ v(x) \end{bmatrix} = \begin{bmatrix} 0 \\ -0.00358533x^2 - 0.3488672x - 10.00195 \end{bmatrix}$$

$$\vec{Y'} = \begin{bmatrix} 0 & 1 \\ 0 & -1/20 \end{bmatrix} \vec{Y} + \begin{bmatrix} 0 \\ -0.00358533x^2 - 0.3488672x - 10.00195 \end{bmatrix}$$

$$Finding\ eigenvalues: (A - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix})x = 0$$

$$det(\begin{bmatrix} -\lambda & 0 \\ 0 & 1 \end{bmatrix}) = 0$$

$$\lambda^2 + (1/20)\lambda = 0$$

$$\lambda = (-1/20) \pm \sqrt{1/400}$$

$$\lambda = (-1/20) \pm (1/20)$$

$$\lambda = 0, -1/10$$

$$Find\ eigenvector for\ \lambda = 0$$

$$(A - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix})\vec{V} = \vec{0}$$

$$\begin{bmatrix} 0 & 1 \\ 0 & -1/20 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \vec{0}$$

$$V_2 = 0, (-1/20)V_2 = 0$$

$$\vec{V} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$Find\ eigenvector for\ \lambda = -1/10$$

$$(A - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix})\vec{V} = \vec{0}$$

$$\begin{bmatrix} 1/10 & 1 \\ 0 & 1/20 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \vec{0}$$

$$(1/10)V_1 + V_2 = 0, (1/20)V_2 = 0$$

$$\vec{V} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$W = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

$$y(x) = W\vec{c} + W\int_0^x W^{-1}\vec{b}(s)ds$$

$$y_h = W\vec{c}103.587 = W(0)\vec{c}, \vec{c} = \begin{bmatrix} 103.587 \\ 0 \end{bmatrix}$$

$$y_h = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 103.587 \\ 0 \end{bmatrix}$$

$$Find\ W^{-1}W^{-1} = 1/det(W) \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$W\ is\ SINGULAR, cannot\ take\ inverse$$

Despite not being able to validate v(x) with the procedure to solve matrix differential equations, one can still assess volatility using the original formula described above.

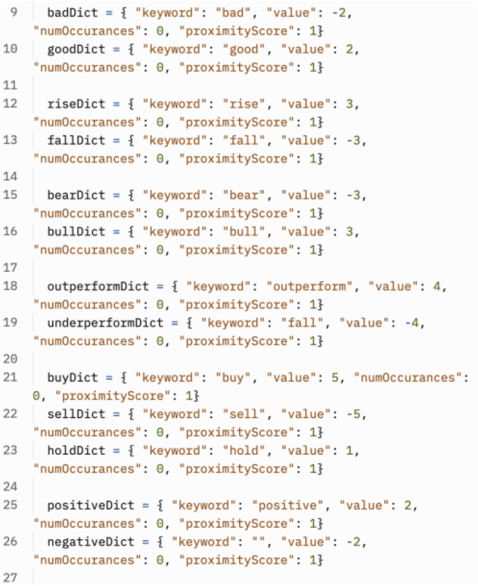
$$volatility = (5/y_0)|(\int_a^b v(x)\,dx)|/(b-a)$$
$$= (5/103.587)|(\int_0^{60} -0.00358533x^2 - 0.34488672x - 1.00195\,dx)|/(60)$$
$$= (5/103.587)(939.05)/(60)$$
$$= 0.75544389418$$

As one can observe, the volatility index for Allstate stock is 0.755.

Natural Language Processing Model

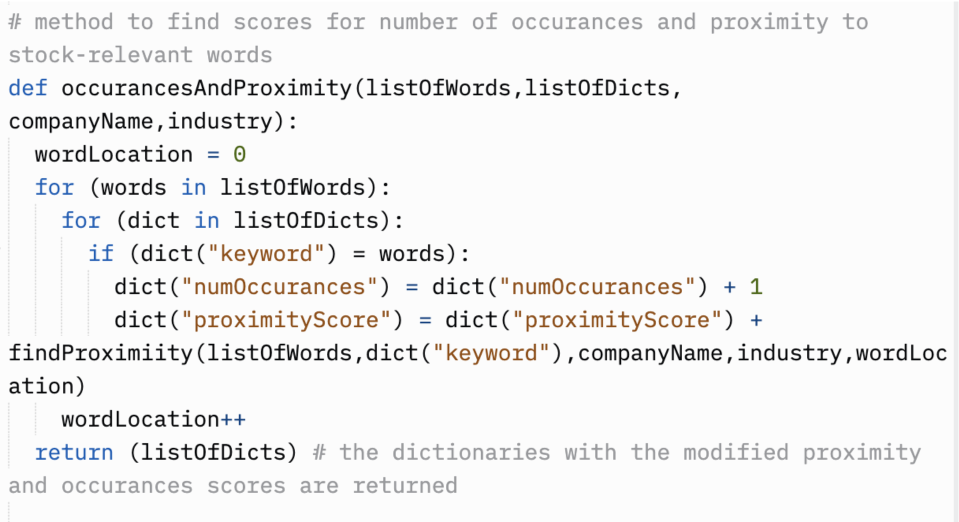
When analyzing an article or social media manuscript to find its impact on a stock, the two most principal factors are the positivity of the text towards the stock and how relevant the text is to the stock. The NLP model created for this analysis tool involves two main methods (one for the positivity of the text and another for the relevancy) that can be multiplied together to determine the overall impact on the stock.

The first step in creating the positivity method (NLPPositivity) was to define a list of keywords that are commonly associated with stock performance. These keywords are then turned into dictionaries so that they can be directly associated with a multiplier value (how big of an impact an occurrence of the word has on the stock), the number of occurrences, and the proximity of these occurrences to the name of the company the stock represents and the industry in which the stock belongs.

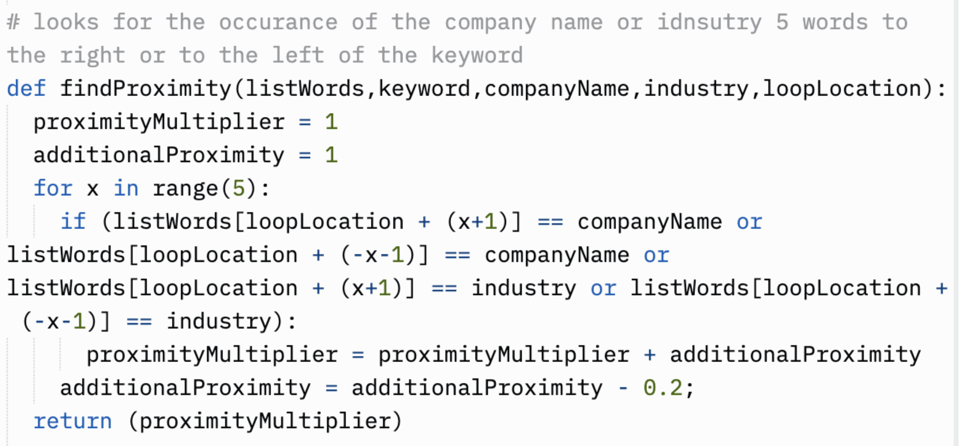


```
  badDict = { "keyword": "bad", "value": -2, "numOccurances": 0, "proximityScore": 1}
  goodDict = { "keyword": "good", "value": 2, "numOccurances": 0, "proximityScore": 1}

  riseDict = { "keyword": "rise", "value": 3, "numOccurances": 0, "proximityScore": 1}
  fallDict = { "keyword": "fall", "value": -3, "numOccurances": 0, "proximityScore": 1}

  bearDict = { "keyword": "bear", "value": -3, "numOccurances": 0, "proximityScore": 1}
  bullDict = { "keyword": "bull", "value": 3, "numOccurances": 0, "proximityScore": 1}

  outperformDict = { "keyword": "outperform", "value": 4, "numOccurances": 0, "proximityScore": 1}
  underperformDict = { "keyword": "fall", "value": -4, "numOccurances": 0, "proximityScore": 1}

  buyDict = { "keyword": "buy", "value": 5, "numOccurances": 0, "proximityScore": 1}
  sellDict = { "keyword": "sell", "value": -5, "numOccurances": 0, "proximityScore": 1}
  holdDict = { "keyword": "hold", "value": 1, "numOccurances": 0, "proximityScore": 1}

  positiveDict = { "keyword": "positive", "value": 2, "numOccurances": 0, "proximityScore": 1}
  negativeDict = { "keyword": "", "value": -2, "numOccurances": 0, "proximityScore": 1}

```

Through a nested iterative process, the number of occurrences of each keyword can be found. Additionally, the program has the functionality to detect whether the name of the company or its industry is within a five-word radius of the keyword and create a “proximity score” to accordingly adjust the impact of the keyword as proximity indicates a close relation between the keyword and the stock itself.

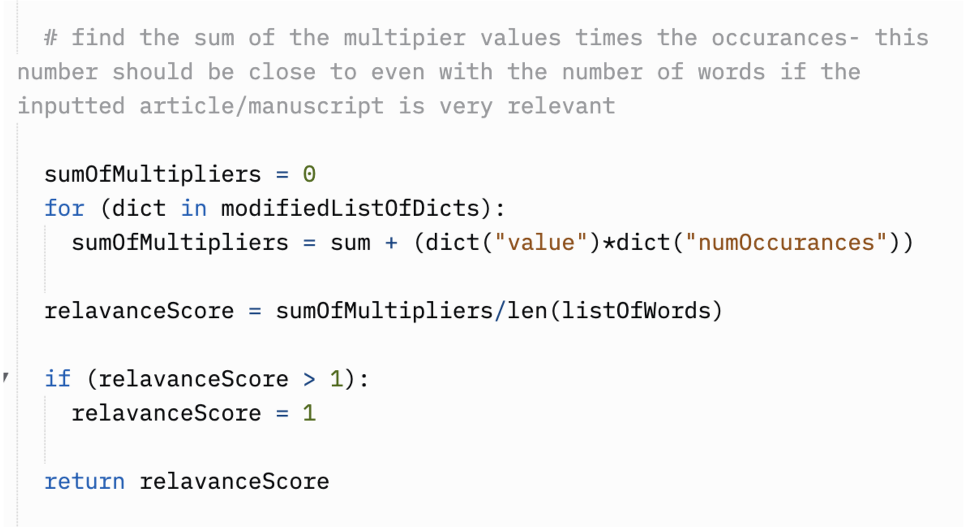


```
# method to find scores for number of occurances and proximity to
stock-relevant words
def occurancesAndProximity(listOfWords,listOfDicts,
companyName,industry):
  wordLocation = 0
  for (words in listOfWords):
    for (dict in listOfDicts):
      if (dict("keyword") = words):
        dict("numOccurances") = dict("numOccurances") + 1
        dict("proximityScore") = dict("proximityScore") +
findProximiity(listOfWords,dict("keyword"),companyName,industry,wordLoc
ation)
    wordLocation++
  return (listOfDicts) # the dictionaries with the modified proximity
and occurances scores are returned
```



```
# looks for the occurance of the company name or idnsutry 5 words to
the right or to the left of the keyword
def findProximity(listWords,keyword,companyName,industry,loopLocation):
  proximityMultiplier = 1
  additionalProximity = 1
  for x in range(5):
    if (listWords[loopLocation + (x+1)] == companyName or
listWords[loopLocation + (-x-1)] == companyName or
listWords[loopLocation + (x+1)] == industry or listWords[loopLocation +
 (-x-1)] == industry):
      proximityMultiplier = proximityMultiplier + additionalProximity
    additionalProximity = additionalProximity - 0.2;
  return (proximityMultiplier)
```

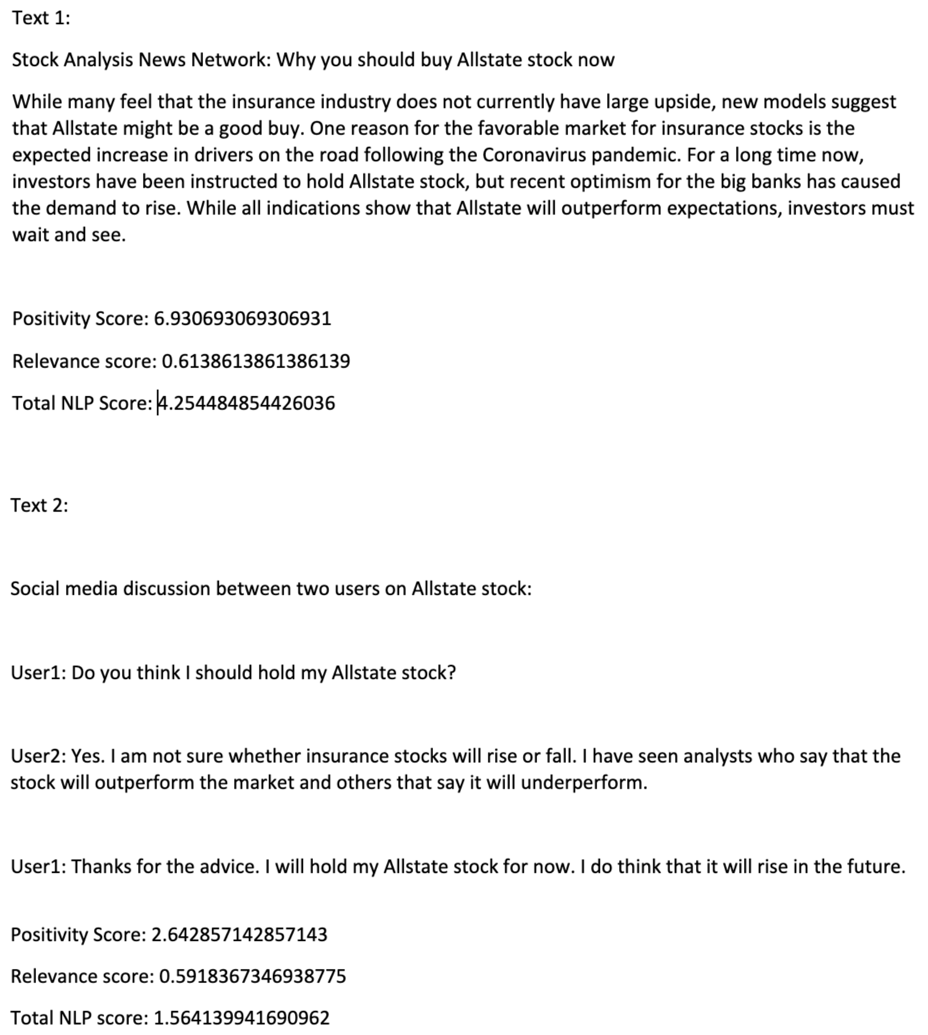
After each keyword has been assigned a multiplier value, number of occurrences, and a proximity score, these numbers can be multiplied and then summed together. Furthermore, this sum is divided by the number of words in the text to account for the differing lengths of text. The value that is produced from this operation becomes the “positivity score” of the given text. The score is meant to resemble the percent of the previous stock price that will be added onto the price in the next iteration of the discrete difference equation model. The positivity score is continued in a range of –10 to 10 as published texts are assumed to not have the ability to impact stocks at a level higher than 10 percent.



```
  # find the sum of the multipier values times the occurances- this
number should be close to even with the number of words if the
inputted article/manuscript is very relevant

  sumOfMultipliers = 0
  for (dict in modifiedListOfDicts):
    sumOfMultipliers = sum + (dict("value")*dict("numOccurances"))

  relavanceScore = sumOfMultipliers/len(listOfWords)

  if (relavanceScore > 1):
    relavanceScore = 1

  return relavanceScore
```

In addition to the positivity score, a “relevancy score” is also produced by the NLP model by quantifying the appearance of words associated with the specific company and the stock market in general. The relevancy score is on a scale of 0 to 1 and is multiplied by the positivity score to produce the result of the NLP model.

To test the model and carry on the investigation into the stock of the Allstate Insurance Corporation, a total of 20 text sources were designed specifically for the model. Each text source is designed to resemble either a news publication or a social media forum. For simplicity’s sake, each source is assigned to a future day on the stock market and each source only impacts the day to which it is assigned. Examples of each type of source are shown below:



Text 1:

Stock Analysis News Network: Why you should buy Allstate stock now

While many feel that the insurance industry does not currently have large upside, new models suggest that Allstate might be a good buy. One reason for the favorable market for insurance stocks is the expected increase in drivers on the road following the Coronavirus pandemic. For a long time now, investors have been instructed to hold Allstate stock, but recent optimism for the big banks has caused the demand to rise. While all indications show that Allstate will outperform expectations, investors must wait and see.

Positivity Score: 6.930693069306931

Relevance score: 0.6138613861386139

Total NLP Score: 4.254484854426036

Text 2:

Social media discussion between two users on Allstate stock:

User1: Do you think I should hold my Allstate stock?

User2: Yes. I am not sure whether insurance stocks will rise or fall. I have seen analysts who say that the stock will outperform the market and others that say it will underperform.

User1: Thanks for the advice. I will hold my Allstate stock for now. I do think that it will rise in the future.

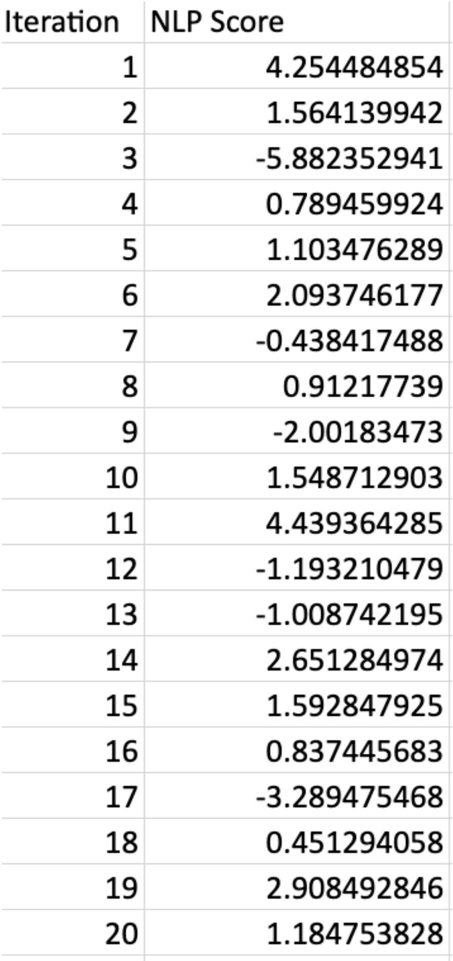
Positivity Score: 2.642857142857143

Relevance score: 0.5918367346938775

Total NLP score: 1.564139941690962

As one can determine from reading the two sources, the “News Article” was highly supportive of Allstate stock and was accordingly assigned a high NLP score of 4.25 by the model. The social media discussion (Text 2) was more neutral towards Allstate stock, gaining it an NLP score of 1.56.

Similar passages to the two above were inputted into the Python NLP model until a total of 20 NLP scores were generated. As a reminder, the result of the natural language processing model is supposed to be indicative of the percent of the previous iteration of the difference equation model of the stock graph that will be added to the next iteration. Like any preliminary model, coefficients are somewhat arbitrary, meaning that the results are not perfect indicators of the effect of the passage on the stock price. Other key factors to consider would be the reputation of the source of the text and the size of the audience. The 20 NLP scores are shown in the table to the right.



Iteration	NLP Score
1	4.254484854
2	1.564139942
3	-5.882352941
4	0.789459924
5	1.103476289
6	2.093746177
7	-0.438417488
8	0.91217739
9	-2.00183473
10	1.548712903
11	4.439364285
12	-1.193210479
13	-1.008742195
14	2.651284974
15	1.592847925
16	0.837445683
17	-3.289475468
18	0.451294058
19	2.908492846
20	1.184753828

The Predictive Algorithm

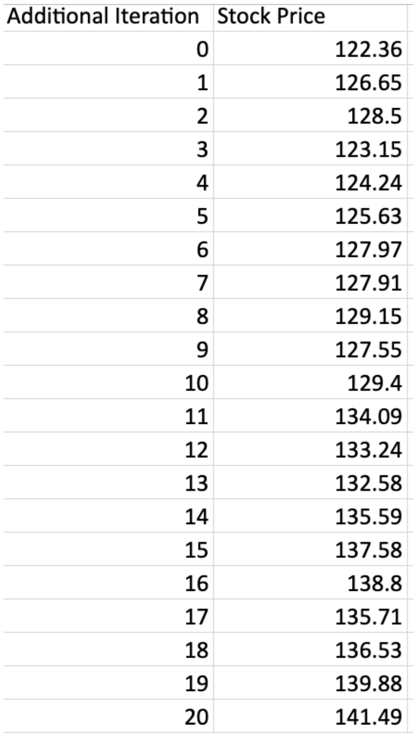
All the components discussed thus far can be combined to produce a difference equation model for 20 additional iterations of a given stock, with each iteration resembling a day of the stock being traded on the market. The predictive algorithm is defined as follows:

$$y_{n+1} = y_n + v(y_n r) + a$$
$$a = additive\ growth\ found\ through\ the\ pattern\ comparison$$
$$r = NLP\ model\ increase\ (NLP\ result/10)$$
$$v = volatility\ index$$

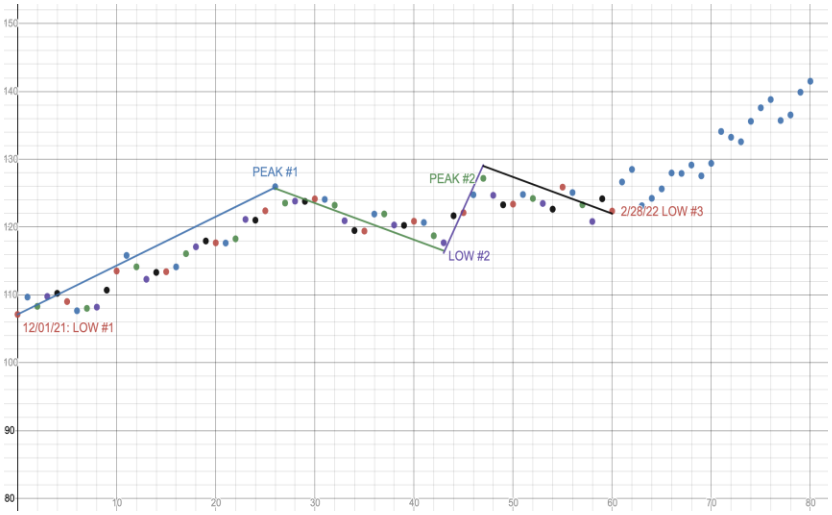
This algorithm can be applied to the example involving the Allstate Insurance Corp by utilizing the multipliers corresponding to the processing of the 20 text files by the NLP model. Since the NLP scores in the chart above are currently expressed as percentage increases, decimal points must first be moved two spaces to the left.

$$y_{n+1} = y_n + 0.755(y_n r) + 0.36$$

The algorithm can produce values for the next 20 workdays of Allstate's stock. These values are represented in both the chart and graph below.



Additional Iteration	Stock Price
0	122.36
1	126.65
2	128.5
3	123.15
4	124.24
5	125.63
6	127.97
7	127.91
8	129.15
9	127.55
10	129.4
11	134.09
12	133.24
13	132.58
14	135.59
15	137.58
16	138.8
17	135.71
18	136.53
19	139.88
20	141.49



The model produces a feasible prediction for Allstate stock as shown in the blue dots towards the end of the graph. However, since the text files inputted into the NLP model were written for the purpose of this report, the changes in Allstate stock shown in the graph above are partially random.

Discussion

The procedure outlined in this report gives insight into how one may go about analyzing articles and social media sources to weigh their impact on stock prices. To apply this procedure to real-world prediction of the stock market, a server would need to be created so that texts could be analyzed in real time as they are written. Furthermore, models would need adjustment so that certain coefficients are less arbitrary before they could be utilized for investment purposes. Despite these modifications for further utilization, the model as it stands demonstrates how difference and differential equations are essential to stock modeling and how one may go about taking advantage of the increased prominence of retail investing.

References

Fitzgerald, M. (2021, March 10). *Retail investors continue to jump into the stock market after GameStop Mania*. CNBC. Retrieved April 29, 2022, from https://www.cnbc.com/2021/03/10/retail-investor-ranks-in-the-stock-market-continue-to-surge.html

Unknown. (2022). *11 most essential stock chart patterns*. 11 Most Essential Stock Chart Patterns | CMC Markets. Retrieved April 29, 2022, from https://www.cmcmarkets.com/en/trading-guides/stock-chart-patterns

Unknown. (2021, June 16). *The rise of newly empowered individual investors*. The Wall Street Journal. Retrieved April 29, 2022, from https://deloitte.wsj.com/articles/the-rise-of-newly-empowered-individual-investors-01623870131

Hayes, A. (2022, February 8). *Volatility*. Investopedia. Retrieved April 29, 2022, from https://www.investopedia.com/terms/v/volatility.asp

Chen, J. (2022, April 7). *Pennant definition*. Investopedia. Retrieved April 29, 2022, from https://www.investopedia.com/terms/p/pennant.asp

MarketWatch, Inc. (2022, April 29). *Download all data: Allstate Corp.. Price Data*. MarketWatch. Retrieved April 29, 2022, from https://www.marketwatch.com/investing/stock/all/download-data?startDate=12%2F1%2F2021&endDate=2%2F28%2F2022

Tolberti. (n.d.). *IOST IOSTUSDT - Big Ascending Triangle on the weekly chart! for Binance:IOSTUSDT by Tolberti*. TradingView. Retrieved April 29, 2022, from https://www.tradingview.com/chart/IOSTUSDT/oPD6p5Cd-IOST-IOSTUSDT-Big-ascending-triangle-on-the-weekly-chart/

Sofien Kaabar, C. F. A. (2021, August 16). *Technical pattern recognition for trading in python.* Medium. Retrieved April 29, 2022, from https://towardsdatascience.com/technical-pattern-recognition-for-trading-in-python-63770aab422f

Schafroth, S. (n.d.). *Logistic Difference Equation*. BioSym. Retrieved April 29, 2022, from https://www.biosym.uzh.ch/modules/models/ETHZ/Logisticdifferenceequation/lde.xhtml

Bonhoeffer, S. (n.d.). *The logistic difference equation and the route to chaotic behavior*. ETH Zurich. Retrieved April 29, 2022, from https://ethz.ch/content/dam/ethz/special-interest/usys/ibz/theoreticalbiology/education/learningmaterials/701-1424-00L/lde.pdf

Mitchell, C. (2022, February 8). *Ascending triangle definition and tactics*. Investopedia. Retrieved April 29, 2022, from https://www.investopedia.com/terms/a/ascendingtriangle.asp#:~:text=An%20ascending%20triangle%20is%20generally,direction%20the%20price%20broke%20out.

Smith, A. (2021, April 13). *The reddit revolt: Gamestop and the impact of social media on institutional investors*. The TRADE. Retrieved April 29, 2022, from https://www.thetradenews.com/the-reddit-revolt-gamestop-and-the-impact-of-social-media-on-institutional-investors/

Boyce, W. E., DiPrima, R. C., & Meade, D. B. (2017). *Elementary Differential Equations and Boundary Value Problems* (11th ed.). Wiley.

Acknowledgements:

[1] Line of Best Fit found using the Wolfram Alpha Cubic Line of Best Fit Calculator

Graphs created using desmos.com

Equations created using TeX Equation Editor

Charts made on Microsoft Excel

Line of Best Fit found using the Wolfram Alpha Cubic Line of Best Fit Calculator

Code created in Python using the Replit compiler