TEMILOLUWA SAMUEL ADEBOWALE
stemitom@gmail.com

1. Essay on why the bitcoin stock-to-flow model is bad

STOCK TO FLOW

The stock to flow model is a way to measure the abundance or scarcity of an asset. The stock to flow ratio is the current supply(stock) of an asset by the new supply(flow) of that asset.

The stock to flow model tends to be popular for some assets such as gold, silver, bitcoin, and some other cryptocurrencies. Taking gold for an example, the amount(stock) of gold ever mined is around 190,000 tons and the current amount mined every year is about 2,000 tons. This means that the stock to flow ratio of gold is about 95. This high stock to flow ratio is what makes gold very valuable.

According to the stock to flow model, an asset has more value when the stock to flow ratio is high, that is when the stock is very large compared to the new supply. So the scarcity of a resource leads to an increase in the worth of that resource.

STOCK TO FLOW AND BITCOIN

As resources such as gold and silver are limited, the stock to flow model applies to bitcoin as the amount of newly produced bitcoin is extremely small compared to the existing stock.

Bitcoin halving is a system of dividing the wide variety of generated rewards consistent with a block, which caps the total supply of Bitcoin and ensures that it does not exceed 21 million coins.

Lead by a twitter user with the name “PlanB”, according to the stock to flow model and based on its increasing scarcity, bitcoin will be fifty-five thousand dollars ($55,000) by 2021

WHY STOCK TO FLOW MODEL IS BAD

The stock to flow model for example cannot be used to explain the prices of other cryptocurrencies. This makes it so that the value of a cryptocurrency is based entirely on its demand. A cryptocurrency or coin could have a high stock to flow ratio but if there isn't enough demand, it still would not be valuable. The issue with other coins can be solved with decentralization.

Analyzing the fundamental basis of the model, it becomes clear that the model is based on the rather strong postulation that the rate of new supply of a monetary good (e.g. gold and silver) directly dictates the USD market capitalization. There is no proven research or evidence to support this theory, other than the singular data points selected to chart gold and silver market capitalization against bitcoin’s trajectory.

The stock to flow model does not take into consideration the deceleration factor. The model assumes that there will be a constant exponential growth. This is not achievable as there will be a time where the market will reach saturation and exponential growth would not be possible. The growth will slow down as bitcoin value gets higher at some point.

A nonstop rise does not happen with any asset. It has never happened before in history. At some point in time, growth will be slow for a long while.

2. Black Scholes Call Price Calculation

Black Scholes Call Price

$C = SN(d_1) - BN(d_2)$

Where

$B = Xe^{-rT}$

$d_1 = \frac{\log_e(S/B) + \frac{1}{2}\sigma\sqrt{T}}{\sigma\sqrt{T}}$

$d_2 = \frac{\log_e(S/B) - \frac{1}{2}\sigma\sqrt{T}}{\sigma\sqrt{T}}$

N = Normal distibution.

Variables

$S = 40$

$T = 4/12 = 1/3$

$r = 0.03$

$\sigma = 0.4$

$X = 45$

$B = Xe^{-rT}$

$B = 45 \times e^{-0.03(1/3)}$

$B = 45 \times e^{-0.01}$

$B = 44.5522$

$d_1 = \frac{\log_e(S/B) + \frac{1}{2}\sigma\sqrt{T}}{\sigma\sqrt{T}}$

$d_1 = \frac{\log_e(40/44.5522) + \frac{0.4 \times \sqrt{1/3}}{2}}{0.4 \times \sqrt{1/3}}$

$d_1 = -0.4667 + 0.1155$

$d_1 = -0.3512$

$d_2 = \frac{\log_e(S/B) - \frac{1}{2}\sigma\sqrt{T}}{\sigma\sqrt{T}}$

$d_2 = -0.4667 - 0.1155$

$d_2 = -0.5822$

$N(d_1) = N(-0.3512) = 0.363$

$N(d_2) = N(-0.5822) = 0.280$

$C = 40(0.363) - 44.5522(0.280)$

$C = 2.045384$

Black Scholes Call Price, $C = 2.045384$

3. Why is it a bad idea to use a recursion method to find fibonacci of a number?

Using recursion to find the fibonacci sequence of a number causes multiple redundant calls(overlapping) by recomputing the same values over and over again.

fib(10) = fib(9) + fib(8)
fib(9) = fib(8) + fib(7)
fib(8) = fib(7) + fib(6)

Using this method, fib(5) will make 15 calls, fib(10) will make 177 calls and
fibonacci(20) will make 20,000 calls.

This can be overcome by using Memoization or by using an iterative method to calculate the fibonacci of a number.

4. Code that takes in a proth number and uses proth theorem to determine if number is prime

```
def is_prime(num):
    if num == 2: return True
    if num < 2 or num % 2 == 0: return False

    sqr = int(num**0.5) + 1

    for div in range(3, sqr, 2):
        if num % div == 0:
            return False
    return True

def isPowerOfTwo(num):
    return (num and (num & (num - 1)) == False)

def proth_number(num):
    num -= 1
    k = 1

    while (k < (num//k)):
        if (num % k == 0):

            if(isPowerOfTwo(num//k)):
                return True
        k = k + 2
    return False

def check_proth_prime(num):
    if proth_number(num):
        if is_prime(num):
            return True
        return False
    return False

print(check_proth_prime(13))
```

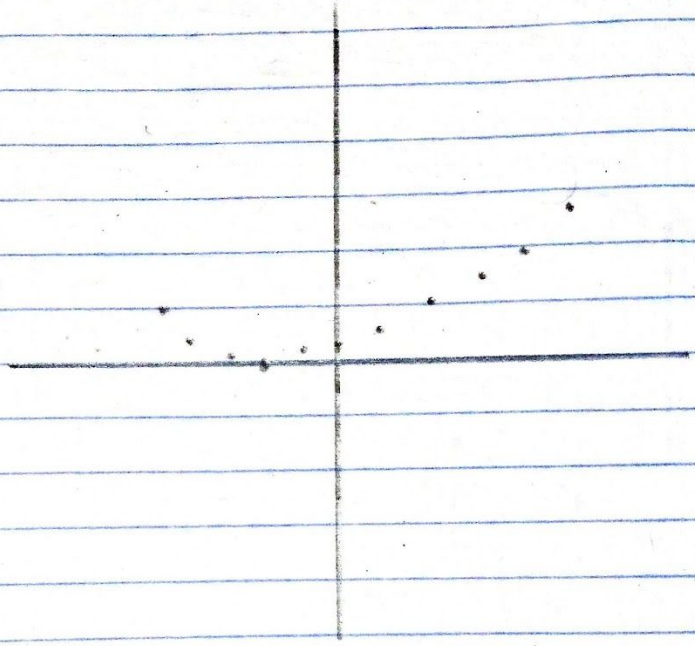
5. Minium value of a function

$y = \sqrt{(x+6)^2 + 25} + \sqrt{(x-6)^2 + 121}$

Using the graph method

Taking values from -5 to 5

x	y
-5	20.66
-4	20.25
-3	20.04
-2	20.01
-1	20.11
0	20.34
1	20.69
2	21.14
3	21.70
4	22.36
5	23.13



The minimum value of y is when x is -2

y_{min} 2 20

Minimum value of y is 20