Statistical Analysis of Bitcoin and Stochastic Modelling using Black-Scholes

Anthony Hills
Department of Physics, University of Surrey
E-mail Address: ah00446@surrey.ac.uk, URN: 6317482

Abstract - Renowned for its high volatility, the recent price of Bitcoin has attracted investors willing to take high risks to make profits. In this regard, this report investigates how the price of Bitcoin (BTC) has performed compared to the British Pound (GBP) over the past 6 years by analysing its annual drift and volatility. The average drift and volatility for bitcoin for the past year was calculated as 0.00753 and 0.847 respectively. Using these results, the 3 and 6 month at the money call option price for BTC-GBP were calculated as £669.41 ± 8.51 and £1646.95 ± 20.95 respectively. In the final section of this report, a stochastic share price model for a share of known drift, volatility and initial share price was simulated using the Black-Scholes model. A statistical analysis for this share is performed and potential hedging strategies to maximize capital discussed.

1. Introduction

Bitcoin, the first digital form of cryptocurrency was created in 2009 and offers a peer-to-peer electronic cash system utilizing the blockchain [1]. Created by the anonymous author under the pseudonym of Satoshi Nakamoto, it allows for online payments to be made without the need of a third party financial institution.

Ever since people have been trading Bitcoin, it has been renowned for its high volatility and potential for profit. Trading from £790 at the start of 2017 to its peak at £14,700 only 12 months later [2], the potential for making profits from trading Bitcoin has attracted investors willing to take on its high risk.

In this regard, this section aims to perform a statistical analysis of the price action of the price of Bitcoin compared to the British Pound.

2. Distribution of Daily Returns for BTC-GBP

The raw data used to perform this statistical analysis was the historical share price for the price of Bitcoin with respect to the British Pound, over a 6 year period from between 30th April 2012 to 30th April 2018 [2].

Using the adjusted daily close prices from this source, the daily returns, R, were calculated as a function of today's price S_i with respect to yesterday's price S_{i-1} as follows:

$$R = ln(\frac{Si}{S_{i-1}})$$

Equation 1. Daily return as a function of price

Having calculated the daily returns, a histogram over a 6 year period, from 30th April 2012 to 30th April 2018, was plotted in figure 1.

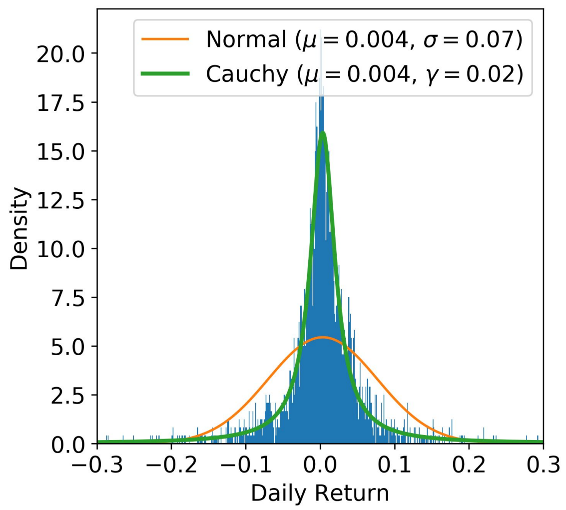

Figure 1. Histogram for the daily returns of BTC-GBP for the past 6 years. Probability distribution functions of the normal distribution and Cauchy distribution are overlaid, with their

relevant parameters highlighted in the top legend.

As seen in figure 1, the price of BTC-GBP does not closely follow a normal distribution - but rather fits a Cauchy distribution fairly accurately. The mean of this histogram of daily returns was 0.004 and the standard deviation was 0.07. By creating a normal quantile plot, the extent to which the daily returns follow a normal distribution can be observed in figure 2 below:

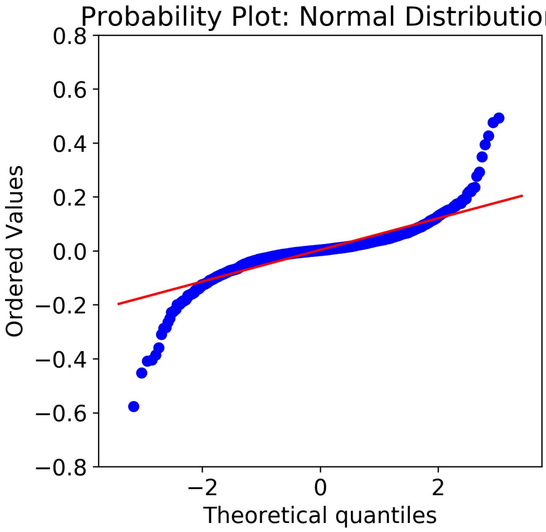

Figure 2. Normal quantile plot for BTC-GBP daily returns. Extreme values above 0.8 and below -0.8 have been removed to improve the clarity of the plot.

For a normal distribution, one would expect a straight line that intersects the origin. Although this behaviour can be observed in the central region of the plot, the tails are more curved showing higher kurtosis, which was calculated as 111. From figure 2 it can also be observed that events have a positive skew with more extreme positive results than negative - with a skew of 3.87.

The plots presented in figure 1 and figure 2 were calculated using daily intervals in the price. Repeating this analysis but using 5 day and 10 day intervals, we observe that the normal probability distribution of the price of Bitcoin changes as follows in figure 3 below:

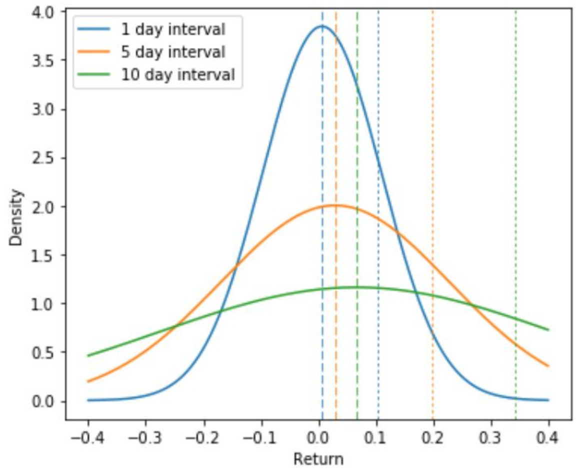

Figure 3. Change in the probability density function of Bitcoin (assuming a normal distribution) for 1 day, 5 day, and 10 day intervals. The mean of the distributions can be observed as the dashed lines, and the standard deviations as the dotted lines.

It can be observed in figure 3 that for larger time intervals, the mean expected return increases in value, along with its standard deviation. This suggests that holding Bitcoin for longer periods before making trades on average will yield higher returns - but coming at a higher risk due to its higher standard deviation.

3. Drift and Volatility of Bitcoin

The annual drift and volatility are calculated as follows:

$$\mu = \frac{\bar{U} + S^2/2}{\triangle t}$$

Equation 2. Drift

$$\sigma = \frac{S}{\sqrt{\triangle t}}$$

Equation 3. Volatility

Where S is the standard deviation and $\bar{U}$ is the mean of equation 4 below:

$$U = \ln \frac{S_i}{S_{i-1}}$$

Equation 4. S_i is the current price, and S_{i-1} is yesterday's price

Using a 60 day rolling averages, the drift and volatility for the price of Bitcoin over the last year was determined as follows using both the mean as an average, and a more robust robust estimate for the average using the median to find the drift, and the median absolute deviation (MAD) to find the volatility:

$$MAD = 1.4826\text{Median}\left(|x_i - \text{Median}(x)|\right)$$

Equation 5. Median absolute deviation

Plotting these results over a course of the previous year, the following figures were obtained:

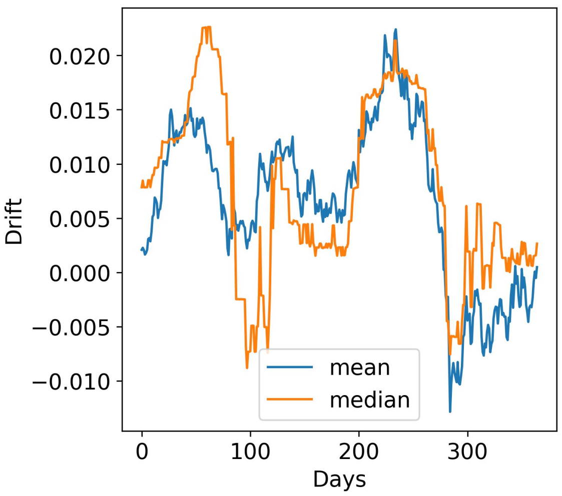

Figure 4. Drift of BTC-GBP from between 30/04/2017 to 30/04/2018 using a 60 day rolling average.

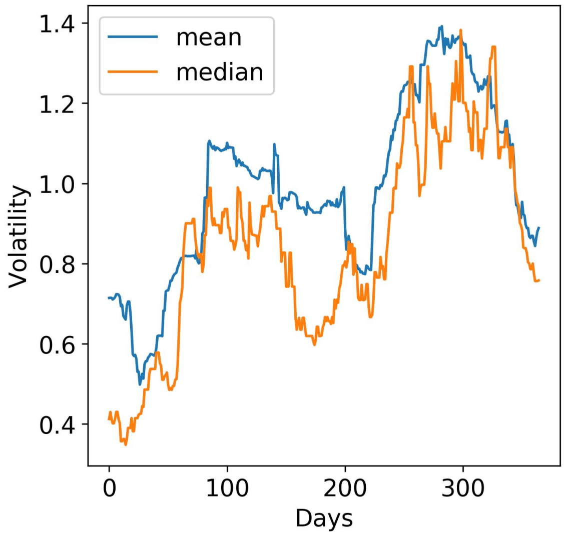

Figure 5. Volatility of BTC-GBP from between 30/04/2017 to 30/04/2018 using a 60 day rolling average.

Taking the average of the median results presented in figures 4 and figure 5, the annual drift was found to be 0.00753 and volatility 0.847.

4. Pricing a Call Option for Bitcoin

We can calculate the value of an at-the-money call option using the results found for the drift and volatility in figures 4 and 5 [3]. Assuming no transaction fees, no dividends, and using LIBOR as the risk free borrowing rate [4], the following equations were used to price the call options for Bitcoin:

$$Z_{\pm} = \frac{(r \pm \frac{\sigma^2}{2})t}{\sigma\sqrt{t}}$$

Equation 6.

$$C = S_0[Norm.dist(Z_+) - e^{-rt}Norm.dist(Z_-)]$$

Equation 7. At-the-money call option

To find the uncertainty in these call options equation 8 was used, where V is the Greek Vega:

$$dC = V d\sigma$$

***Equation 8* .** Uncertainty in pricing the call options

Hence, the 1 month and 6 month at-the-money call options for BTC were calculated as £669.41 ± 8.51 and £1646.95 ± 20.95 respectively.

5. Hedging Strategies for a Hypothetical Stock

In this section we consider a hypothetical situation in which a hedge fund has purchased data showing a particular stock will rise above the risk free rate. Using the Black-Scholes model, the fund assumes the share will evolve according to equation 9 below:

$$\frac{dS}{S} = \mu dt + \sigma dW$$

***Equation 9* .** Black-Scholes model

Where dS is the change in the price in a time interval dt, μ the growth rate of the price, and σ the volatility. The growth rate and volatility are assumed as constant.

Assuming the hedge fund starts with no assets at time t = 0, the following strategy is proposed: Sell a bond at t = 0 and buy one share at the initial price of $S_0 = 100$. When the share price reaches a threshold price, B, described in equation 10 below:

$$B = S_0(1+k)e^{rt}$$

Equation 10. Threshold sell price

The time at which this threshold price is reached will be referred to as t*. When the share value reaches this threshold price, the share is sold and a risk free bond is purchased. This proposed strategy helps reduce the risk of the hedge fund, as it allows for their investments to be more likely sold when the share price is high.

Using equation 10 above, the hedge funds portfolio's discounted value to today is hence:

$$v(t) = \begin{cases} S(t)e^{-rt} - S_0 & t \leq t^* \\ S_0 k & t > t^* \end{cases}$$

Equation 11. Portfolio's discounted value for any given time based on the proposed hedging strategy.

By simulating this strategy through a computer program, its long term profitability and risk was analysed. In this program the risk free rate, r, was taken as r = 0.04. The value of k was taken as 0.2, the annual drift as 0.16, and the annual volatility taken as 0.2.

Using the Black-Scholes model, figure 6 below shows an estimate for how the share price might evolve over the course of a year using 250 trading days.

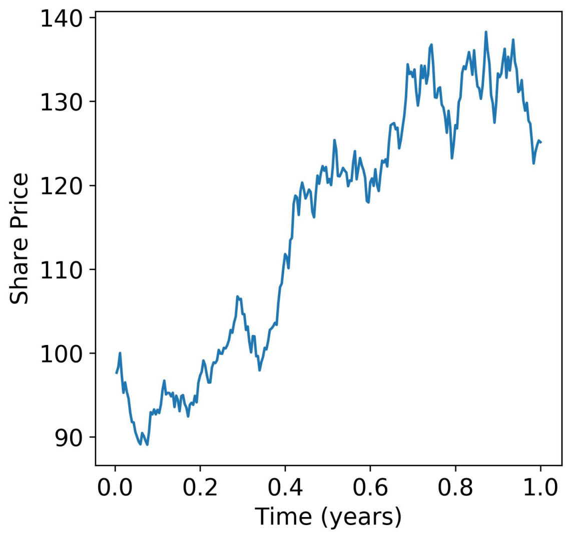

Figure 6. Stochastic share price simulation for one year with a volatility of 0.2 and drift of 0.16.

Using the strategy described in equation 11, figure 7, 8, and 9 below illustrate the final portfolio value after 1, 2, and 5 years respectively for 1000 realisations of the stock price.

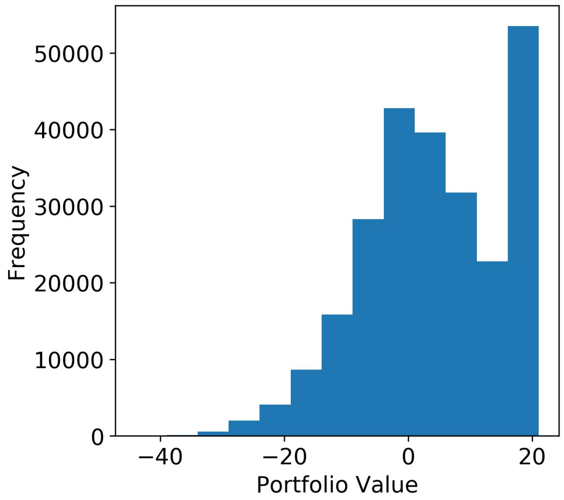

Figure 6. Portfolio value after 1 year of stochastic simulation for 1000 realisations, with a bin width of 5.

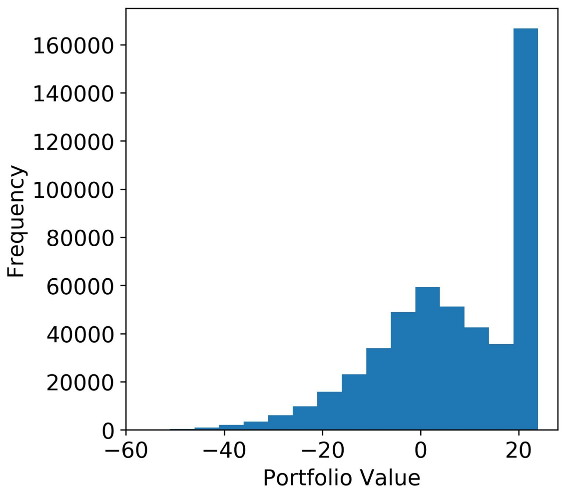

Figure 7. Portfolio value after 5 years of stochastic simulation for 1000 realisations, with a bin width of 5.

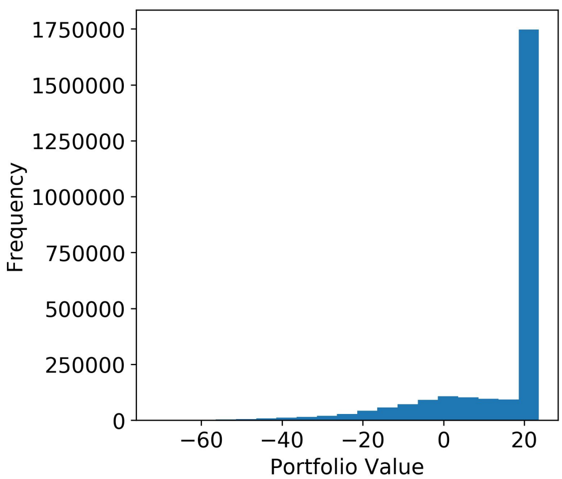

Figure 8. Portfolio value after 10 years of stochastic simulation for 1000 realisations, , with a bin width of 5.

From figures 6-8, it can be observed that the average portfolio value increases with the amount of time held. This is confirmed in figures 9 and 10. The large peaks in frequency at a portfolio value of 20 for figures 6-8 are due to the threshold condition discussed in equation 10.

Figures 9 and 10 below show how the mean and variance of the portfolio value for 20 years varies with each trading day.

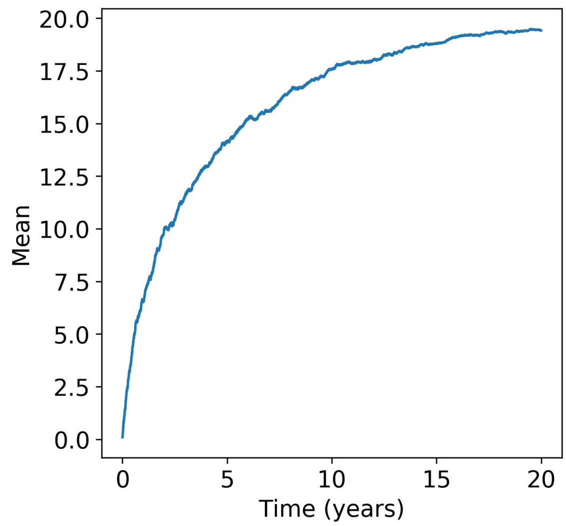

Figure 9. Mean of the portfolio value for every trading day for 20 years, using 1000 realisations of the stock price.

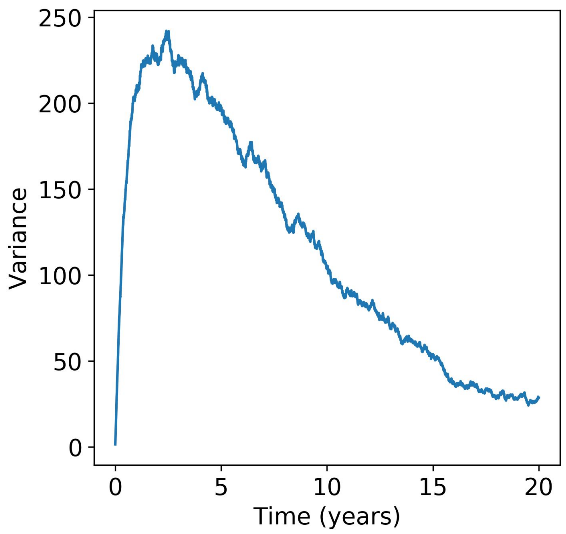

Figure 10. Variance of the portfolio value for every trading day for 20 years, using 1000 realisations of the stock price.

The mean and variance from figures 9 and 10 are both useful tools for a potential investor to be mindful of, as investors should look to maximise their profit (i.e. the mean), while minimizing their risk (variance).

By observing the results in figure 9, it would be ideal to hold this share for approximately 10 years, as it is near this point where the mean reaches its peak value, and the variance has been reduced significantly. Selling after 10 years would help reach maximum profit while having reduced risk, but at an increased opportunity cost.

The probability of a loss as a function of time for 20 years was analysed and presented below in figure 11:

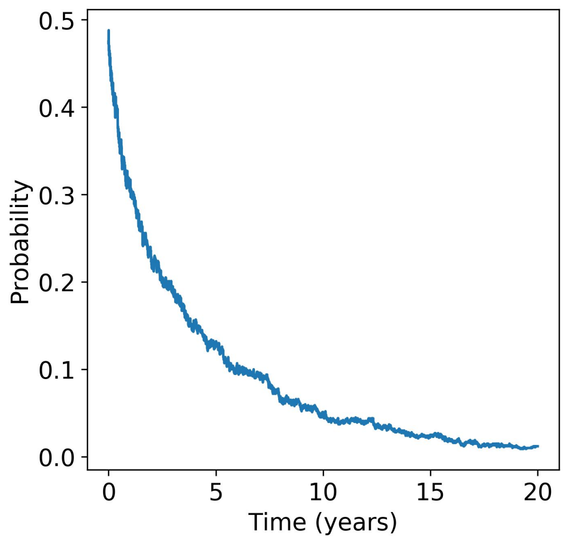

Figure 11. Probability of a loss for 20 years, using 1000 time steps

From figure 11 the probability of a loss can be seen to decrease exponentially with time. The probability of a loss over a short initial time frame is observed to be relatively high - suggesting a risky short term investment. However, the probability of a loss after 5 years can be seen to fall by a factor of 10 to only 5%. This finding suggests that this share is a relatively risk-free long term portfolio option due largely because of the boundary condition described in equation 10. However, the major drawback to this strategy is that it limits the mean return to 20 as observed in figure 9.

Having analysed the share for a volatility of 0.2, we compare this strategy for a the case when the volatility is now 0.6. Results for the mean, variance, and probability of a loss can be observed in figures 12, 13, and 14 respectively:

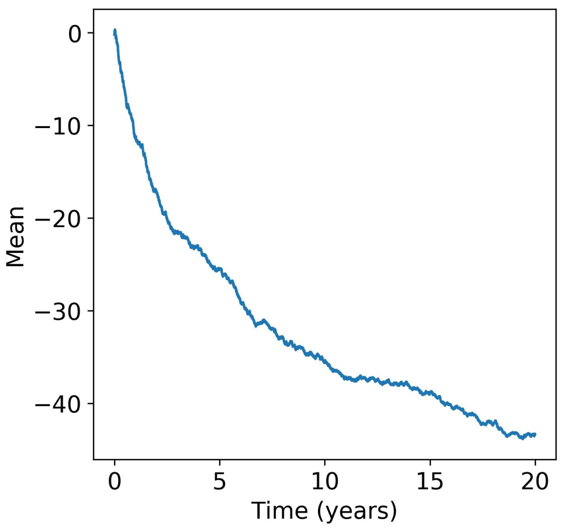

Figure 12. Mean portfolio value for a volatility of 0.6 using 1000 iterations

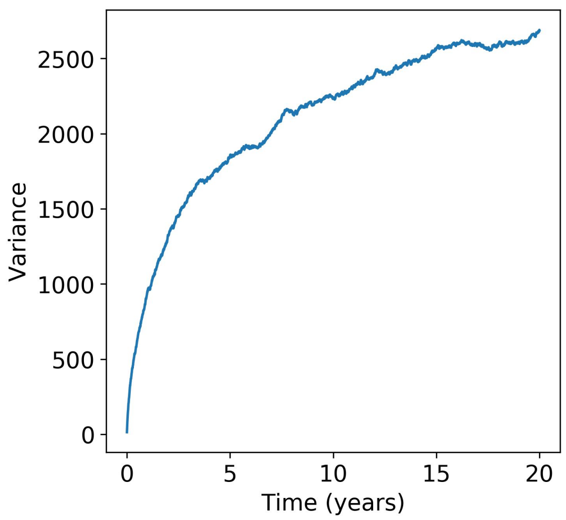

Figure 13. Variance of portfolio value for a volatility of 0.6 using 1000 iterations

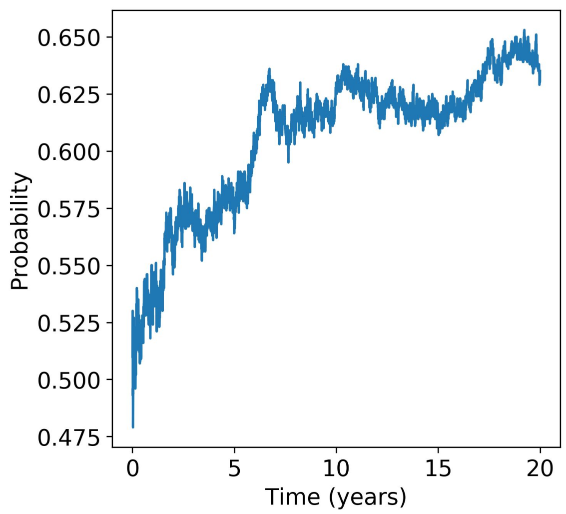

Figure 14. Probability of a loss using 1000 iterations

From these new figures it can be observed that the portfolio performs worse in essentially every sense with the current hedging strategy: with a lower mean, higher variance and larger probability of a loss. Not only is the risk significantly higher with a probability of a loss converging greater than 60% after 20 years, but the mean portfolio value is almost always negative.

Based on these results, and since this hedge fund is mainly interested in the long term profitability of this strategy - if the volatility were to change as described, then this hedging strategy would no longer be recommended - as they would on average lose money in nearly all time frames, with even greater probability of losses in the long term.

<u>References</u>

[1] - Nakamoto, Satoshi. (2009). Bitcoin: A peer-to-peer electronic cash system. [Online]. Available: https://bitcoin.org/bitcoin.pdf. [Accessed: 01/05/2018]

[2] - Historical price of Bitcoin GBP. [Online]. Available: https://uk.finance.yahoo.com/quote/BTC-GBP?ql=1&p=BTC-GBP

[3] - J. Adams, "Option pricing: Binomial Trees", *surreylearn.surrey.ac.uk*, 2018. [Online]. Available: https://surreylearn.surrey.ac.uk/d2l/le/content/156852/viewContent/1298012/View. [Accessed: 01/05/2018]

[4] - "Bank of England Statistical Interactive Database | Main Page", *bankofengland.co.uk*, 2018. [Online]. Available: http://www.bankofengland.co.uk/boeapps/iadb/

NewInterMed.asp?Travel=NIxAZxI1xSCxSUx
. [Accessed: 01/05/2018].

Appendix A: Fortran code used to analyse the hedging strategy

```
!==================================
==================================
===========
! Name: Anthony Hills
! URN: 6317482
! PHY3048
!
! This program analyses the long term
profitability and risk of the strategy
! of a hedge fund.
!
! The Black-Scholes model is used to simulate
the share price over time, and
! the portfolio's performance is evaluated by
measuring its mean, variance,
! and probability of a loss as a function of time.
!==================================
==================================
==========
PROGRAM statistical_arbitrage
  IMPLICIT NONE

  ! Parameters
  INTEGER, PARAMETER :: realisations =
1000  ! number of simulations
  REAL, PARAMETER :: mu = 0.16  ! drift
  REAL, PARAMETER :: sigma = 0.6  !
volatility
  REAL, PARAMETER :: S0 = 100.0  ! share
price at t=0
  REAL, PARAMETER :: dt = (1.0/250.)  !
time step
  REAL, PARAMETER :: r = 0.04  ! risk-free
rate
  REAL, PARAMETER :: k = 0.2
  REAL, PARAMETER :: pi = 4.*ATAN(1.)

  ! Helpers
  REAL :: x1, x2, z1, z2, dW, dS, S, t, mean,
variance, prob_loss
  INTEGER :: realisation, trading_days, i_seed,
years, day, losses, i
  REAL, DIMENSION(:,:), ALLOCATABLE
:: portfolio

!
-------------------------------------------------------
-------------------
  ! Allocate the portfolio array using the user
input number of trading days
  WRITE(6,*) 'How many years would you like
to simulate for?'
  READ(5,*) years
  trading_days = years / dt
  ALLOCATE(portfolio(trading_days,
realisations))

  ! Open files to be written to
  OPEN(10,file='share_price_simulation.dat')

OPEN(20,file='portfolio_value_histogram.dat')
  OPEN(30,file='probability_of_loss.dat')
  OPEN(40,file='portfolio_mean_variance.dat')
  OPEN(50,file='box_muller.dat')

!
-------------------------------------------------------
-------------------
  ! Simulate value of portfolio for chosen
number of realisations
  portfolio(:,:) = 0.;  variance = 0.  ! Initialize
variables
  DO realisation=1, realisations
     S = 100.; t = 0.0  ! Reinitialize initial share
price at t=0
     DO day=0, trading_days  ! Evaluate value
for each day
        t = t + dt
        dW = SQRT(dt) * rand_box_muller()  !
Wiener process
        dS = S * (mu * dt + sigma * dW)
        S = S + dS  ! Update current share price
        IF (realisation == 1) THEN
           WRITE(10,*) t, S  ! Save simulated
share price using 1 realisation
        END IF

        ! Define hedging strategy
        portfolio(day,realisation) = S * EXP(-r * t)
- S0
```

```
      IF (S >= S0 * (1. + k) * EXP(r * t)) THEN
        portfolio(day,realisation) = S0 * k
      END IF
      WRITE(20,*) portfolio(day,realisation)  !
Save portfolio values to file
    END DO
  END DO

!
-------------------------------------------------------
------------------
  ! Evaluate portfolio's mean, variance and loss
probability with time
  t = 0.;
  DO day=0, trading_days
    losses = 0.; variance = 0.; mean = 0.  !
Initialize variables
    t = t + dt
    mean = SUM(portfolio(day,:)) / realisations
! Calculte mean
    DO realisation=1, realisations
       variance = variance +
(portfolio(day,realisation) - mean) ** 2
      IF (portfolio(day, realisation) < 0.) THEN
        losses = losses + 1
      END IF
      prob_loss = REAL(losses) / realisations  !
Probability of a loss
    END DO
    WRITE(30,*) prob_loss, t  ! Save
probability of a loss to file
    variance = variance / (realisations-1)
    WRITE(40,*) mean, variance, t  ! Save
portfolio statistics to file
  END DO

!
-------------------------------------------------------
------------------
  DO i=1, 10000
    WRITE(50,*) rand_box_muller()  ! Save
random numbers with mean 0 and var 1
  END DO

  CLOSE(10); CLOSE(20); CLOSE(30);
CLOSE(40); CLOSE(50)  ! Finish output

  ! Display information to user
  WRITE(6,*) 'A stochastic simulation for the
share price has been saved to:'
  WRITE(6,*) "
share_price_simulation.dat             "
  WRITE(6,*) "
  WRITE(6,*) "The portfolio value for 1000
realisations and for your chosen:"
  WRITE(6,*) "number of years has been saved
to:"
  WRITE(6,*) "
portfolio_value_histogram.dat "
  WRITE(6,*) "
  WRITE(6,*) "The probability of a loss as a
function of time was saved to:"
  WRITE(6,*) "
probability_of_loss.dat"
  WRITE(6,*) ""
  WRITE(6,*) "The mean and variance for the
value of the portfolio as a "
  WRITE(6,*) "function of time has been saved
to:"
  WRITE(6,*) "
portfolio_mean_variance.dat "
  WRITE(6,*) ""

!================================
Functions
===================================
==
CONTAINS
  ! Generates a normally distributed random
number with mean zero and variance
  ! one from the Box-Muller transform
  REAL FUNCTION rand_box_muller()
    REAL :: x1, x2, z1, z2
    REAL, PARAMETER :: pi = 4.*ATAN(1.)
    INTEGER :: i

    ! ----- variables for portable seed setting -----
    INTEGER :: i_seed
    INTEGER, DIMENSION(:),
ALLOCATABLE :: a_seed
    INTEGER, DIMENSION(1:8) :: dt_seed
    ! ----- end of variables for seed setting -----
    REAL :: r

    DO i=1, 2  ! Loop twice
      ! ----- Set up random seed portably -----
```

```
      CALL RANDOM_SEED(size=i_seed)
      ALLOCATE(a_seed(1:i_seed))
      CALL RANDOM_SEED(get=a_seed)
      CALL
DATE_AND_TIME(values=dt_seed)
      a_seed(i_seed)=dt_seed(8);
a_seed(1)=dt_seed(8)*dt_seed(7)*dt_seed(6)
      CALL RANDOM_SEED(put=a_seed)
      DEALLOCATE(a_seed)
      ! ----- Done setting up random seed -----

      IF (i == 1) CALL
RANDOM_NUMBER(x1);
      IF (i == 2) CALL
RANDOM_NUMBER(x2);
    END DO

    ! Generate normally distributed numbers,
according to Box-Muller transform
    z1 = sqrt(-2.0 * log(x1)) * cos(2.0 * pi * x2)
    z2 = sqrt(-2.0 * log(x1)) * sin (2.0 * pi * x2)

    rand_box_muller = z1
  END FUNCTION
END PROGRAM statistical_arbitrage
```