Active trading strategy based on price & volume data

Aarmir Murad - B00924776, Amy Truong - B00917827, Dhruvit Raval - B00902291,
Pal Shah - B00942071, Samarth Jariwala - B00899380
Project Group 5 – CSCI 6409 - Process of Data Science
Instructor: Prof. Evangelos Milios
Fall 2022

I. ABSTRACT

The popularity of cryptocurrency trading has increased significantly in recent years and bitcoin (BTC) is one of the most traded and popular cryptocurrencies in the market. Machine learning can be a powerful tool for improving bitcoin trading decisions. In this report, we aim to use both supervised and unsupervised machine learning algorithms, including K-means Clustering and Logistic Regression to predict trading actions in the next hour that tries to give better profits for investors than using a simple buy and hold strategy. To create these models, we will use historical hourly bitcoin trading data and follow a process that includes exploratory data analysis, data preprocessing, model training, parameter tuning, model evaluation, and performance comparison. Finally, we will provide an overall analysis of our models and discuss their advantages and disadvantages.

II. INTRODUCTION

A. Cryptocurrency and bitcoin

Nowadays, there are many different types of assets that people can choose to invest in, and **cryptocurrency** (crypto) might be the most exciting one. Cryptocurrency is a virtual currency that can be used to make transactions anonymously and securely over the internet [1]. In recent years, investing in cryptocurrency has been growing in popularity. Investors trade cryptocurrencies mostly on cryptocurrency exchanges, which are online platforms that allow users to buy and sell cryptocurrencies using various fiat currencies or other cryptocurrencies. Some examples of popular exchanges include Coinbase, Binance, and Kraken. According to an Ipsos survey, 35% of global internet users say they are likely to invest in crypto as a short-term investment, and 36% are likely to invest in this asset as a long-term investment [2].

Among all cryptocurrencies, **bitcoin (BTC)** is the oldest and most well-known. It was created in 2009 and is based on a decentralized ledger technology called the blockchain. It is considered the first successful implementation of a decentralized digital currency, and it has inspired the creation of numerous other cryptocurrencies. Since its creation, bitcoin has become the most widely used and valuable cryptocurrency, with a market capitalization that exceeds the value of all other cryptocurrencies combined. It has also attracted significant attention and adoption, and is now accepted as a legitimate form of payment by merchants and individuals around the world.

BTC-USD is an abbreviation that refers to the price of Bitcoin in US dollars. It is a common way to express the value of Bitcoin and is used by many cryptocurrency exchanges, brokers, and other platforms to quote the price of Bitcoin. For example, if the price of BTC-USD is $20,000, it means that one Bitcoin is worth 20,000 US dollars. Traders and investors often use BTC-USD as a benchmark for the value of Bitcoin, and may use it to make buying and selling decisions based on the current market price. It is also used to compare the performance of Bitcoin to other assets, such as stocks, commodities, or other cryptocurrencies.

B. The challenges of bitcoin trading

Trading bitcoin carries inherent risks due to its highly volatile nature and complexity in understanding the technology and market. The value of bitcoin is determined by supply and demand on exchanges and influenced by many other external factors such as market conditions and global economic events. These factors make bitcoin prices fluctuate significantly over short periods of time, which can make it difficult to predict its direction and make informed trading decisions. Also, Bitcoin and other cryptocurrencies can be complex and challenging to understand, especially for those new to trading. Trading bitcoin requires a certain level of knowledge and understanding of the underlying technology, as well as the market dynamics. This can make it difficult for traders to make informed decisions and may increase the risk of mistakes or losses.

In fact, many people investing in bitcoin, or cryptocurrency in general, do not have sufficient knowledge of investment and crypto. According to NerdWallet's survey conducted in May 2022, 68% of Americans claim they do not fully understand the crypto concept and how it works, and 37% of crypto owners admit the same [3]. If an investor is not a digital native, the idea

of tokens and how they fluctuate may not come naturally to them. It is risky trying to invest in something you do not completely understand, especially in an asset that is incredibly volatile as crypto. A crypto's price can soar to unimaginably high, but it can also drop to terrible lows just as quickly. Thus, feelings will lead to the easiest mistake of buying high and selling low. Some investors' investment decisions are solely based on word-of-mouth trading advice from various social media platforms such as Twitter, Youtube, etc. [4], or information generated from improper simple analysis, for example, prediction based only on some previous days' closing prices. These inappropriate or incomplete information sources often lead to fear of missing out which presses them to make bad impulse trading decisions. The lack of understanding of how a bitcoin price fluctuates, blind beliefs in financial advice from improper sources & improper analysis, and limited capabilities in predicting future prices will result in bad investment decisions, and cause investors to endure major losses when trading BTC-USD and crypto in general.

C. What is machine learning and how it can help

Machine learning is a field of artificial intelligence that involves the use of algorithms and statistical models to enable computers to learn and make predictions or decisions without being explicitly programmed. Machine learning algorithms can analyze large amounts of data and identify patterns and relationships that might not be immediately apparent to humans.

In the context of bitcoin trading, machine learning can be a powerful tool for improving bitcoin trading decisions by helping traders better understand market conditions and identify profitable opportunities. Machine learning algorithms can be trained on historical data to identify patterns and trends in the price of bitcoin. This can help investors make informed decisions about when to buy and sell based on the predicted direction of the price. Machine learning can also be used to automate trading strategies. For example, a trader could use machine learning to design a system that buys bitcoin when the price falls below a certain threshold and sells when it rises above another threshold. This helps traders take advantage of market fluctuations without having to constantly monitor the market. Machine learning algorithms can also be used to analyze market conditions such as volume, liquidity, and order flow, allowing investors to understand the state of the market and make better-informed decisions about when to enter or exit trades.

D. Objectives of the report

The purpose of the report is to build a reliable predictive model based on both supervised and unsupervised machine learning techniques and historical bitcoin trading information to improve traders' profits over a period of time. To be specific, our proposed models are expected to predict trading decisions (to buy, hold, or sell) that a bitcoin investor should take every hour, to receive higher profits than simply using the buy and hold strategy.

The buy and hold strategy, also known as "HODL", is a long-term investment approach that involves buying a cryptocurrency, such as Bitcoin, and holding onto it for an extended period of time, rather than actively trading it, until it appreciates in value, at which point it can be sold for a profit. This strategy is based on the belief that bitcoin price has the potential to increase in the long term and helps investors to not worry about any short-term price fluctuations. However, one disadvantage of the buy and hold strategy is the opportunity cost of not being able to take advantage of short-term price movements. If an investor holds onto their Bitcoin for a long period of time, they may miss out on potential profits from short-term price swings. Besides, there is always the risk that the value of Bitcoin could decrease over time, leading to a loss for the investor.

Our machine learning models to predict whether to buy, hold, or sell Bitcoin on an hourly basis can potentially offer several benefits for investors, as opposed to using a buy and hold strategy. One benefit is the potential for increased flexibility and the ability to gain profits from short-term price changes. The buy and hold strategy involves holding onto an asset for an extended period of time, regardless of short-term price fluctuations. By contrast, using our machine learning models to make hourly predictions allows investors to potentially adjust their positions more frequently, taking into account short-term price movements. This can potentially allow investors to take advantage of short-term opportunities and potentially increase profits. Another benefit is the potential for increased accuracy in predicting price movements. This allows investors to make more informed decisions about when to buy, hold, or sell Bitcoin, rather than relying on gut feelings or guesses. However, it is important to note that there is always the possibility of making incorrect predictions; thus, our goal for the algorithm is not to make correct and profitable trading decisions every hour but to deliver higher returns over time.

III. LITERATURE REVIEW

A. Related works

There has been a significant amount of work done on developing machine learning algorithms to improve trading decisions for bitcoin and cryptocurrency investors, and most of the literature's approach involves providing bitcoin price prediction, in short-term and long-term, as an informed reference for investors to make better decisions in trading. Previous research on predicting cryptocurrency prices can be broadly classified into three main models:

statistical methods, machine learning, and ensemble learning. These models use various types of data: price data, economic data, and sentiment indicators calculated from natural language processing of text data from social media. Other types of data that have been used in some studies include blockchain hash rate, number of online nodes, active addresses, and Google trends, as well as other financial indexes.

Early research on predicting the price of Bitcoin mainly used statistical methods. P. Katsiampa et al. (2017) used price data and certain types of GARCH models to calculate daily closing prices and found that the AR-GARCH model was the best [5]. S. Roy et al. (2018) used price data and performed ARIMA, autoregressive (AR), and moving average (MA) models on a time series dataset and found that the ARIMA model had an accuracy rate of 90.31% [6]. In a more recent study, R. K. Jana et al. (2021) proposed a regression framework based on differential evolution was proposed to predict Bitcoin prices, using a decomposition of the original sequence into linear and nonlinear components and fitting polynomial regression with interaction (PRI) and support vector regression (SVR) on both components to obtain projections [7]. Jong-Min Kim et al. (2022) proposed using linear and nonlinear error correction models to predict Bitcoin log returns and compared them with a neural network, ARIMA, and other methods. The results showed that the error correction model had the best performance in all evaluation indexes, with a mean absolute error as low as 1.84, while the other comparison models had mean absolute errors above 3.2 [8].

In recent years, advances in machine learning have led to the development of more accurate methods for predicting cryptocurrency prices. Jang et al. (2017) have used price data, blockchain data, economic indices, and currency exchange rates in Bayesian neural network models to predict the price of Bitcoin [9]. Li et al. (2020) used price data, technical indicators, and complex neural networks like CNN-LSTM to predict cryptocurrency prices [10], while Jay et al. (2020) used stochastic neural networks that incorporated layer-wise randomness to simulate market fluctuations and used market transaction data, blockchain data, and Twitter and Google Trends data [11]. In a study published in 2021, Loginova and colleagues used linguistic analysis to examine text data from Reddit, CryptoCompare, and Bitcointalk to develop a method for predicting the direction of bitcoin prices. They combined the sentiment analysis model JST with TS-LDA, and the resulting model had an accuracy of 57%. This was an improvement over the same model without JST and TS-LDA [12]. In a separate study published in 2022, Parekh et al. proposed hybrid frameworks that take into account the interdependence of cryptocurrencies and market sentiment. They validated their model using Bitcoin Cash data from March 2021 to April 2021, and the model had an MSE value as low as 0.0011 [13].

Ensemble learning is a technique that has been used to improve the accuracy and reliability of predictions in the context of bitcoin price forecasting. For example, Ibrahim (2017) applied ensemble learning to price and sentiment data in order to construct an XGBoost-Composite model to predict bitcoin prices [14]. Livieris et al. (2020) compared different ensemble models using price data, including averaging, bagging, and stacking, and found that stacking had the best performance [15]. Shin et al. (2020) used price data and combined long short-term memory (LSTM) models trained for different periods of time, such as days, hours, and minutes, in order to create an integrated model that outperformed individual models [16].

B. Our report's contribution to the literature

Previous research has demonstrated that machine learning algorithms can be useful in forecasting the price of bitcoin. However, there is a lack of literature on how to use this information to inform specific trading decisions, such as whether to buy, hold, or sell bitcoin at a particular time. This is an important gap in the field because simply knowing the future price of bitcoin does not necessarily provide enough information to make informed trading decisions. For example, if a trader correctly predicts that the price of bitcoin will drop in the coming month, it may not be clear whether they should sell all of their bitcoin immediately or hold on to it in case the price recovers in the next three months. Without further guidance on how to apply this knowledge in a practical way, it can be difficult for investors to maximize their returns. In order to address this gap, our report aims to provide machine learning algorithms that can help investors make specific trading decisions in order to improve their profits.

IV. METHODOLOGY

A. Overview of the Proposed Methodology

The methodology outlined in the report can be seen in Figure 1, which presents an overview of the steps taken to solve the problem using machine learning. These steps include (1) the Business Understanding phase, in which we specify the problem to be solved by machine learning algorithms, propose possible machine learning models for supervised and unsupervised learning, and choose meaningful evaluation metrics to evaluate our solutions; (2) Data Collection phase with the dataset and source; (3) Data Exploration phase, in which we visualize the features, prepare data quality report and data quality plan; (4) the Formation of our Baseline Model which is the buy and hold strategy; (5) Data preparation phase, in which we retain necessary features and make derived features for our supervised and unsupervised models; (6) Build the K-

means model (unsupervised) and Logistic Regression model (supervised); (7) Experiments phase, in which we evaluate and compare our three models based on previously chosen metrics; and (8) Conclusions.

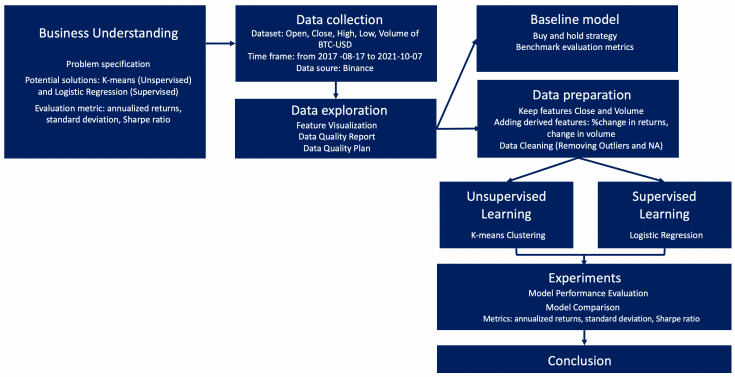

Figure 1. Methodology Overview

B. Business Understanding

We have defined our problem statement and objectives in the Introduction section of this report. Our approach to solving the problem is to develop a K-means Clustering algorithm for the unsupervised approach and Logistic Regression for the supervised approach, calculate the performance of these models in terms of generating returns, and compare them with our benchmark buy and hold strategy. For this baseline strategy, we assume the investor invested 1 USD in bitcoin on 2017-08-17 and is still holding it until 2021-10-07. The models will be developed using the hourly closing price and trading volume of BTC-USD. The goal of our proposed models is to bring higher returns for the investor when compared to using the buy and hold strategy. Thus, in our report, we will present a baseline model, based on buy and hold strategy, and two proposed algorithms, including K-means Clustering and Logistic Regression. In order to assess the performance of our models, we will use the following relevant evaluation metrics:

- **Annualized return** is a measure of the average return on an investment over a given period of time, expressed as a percentage. This metric allow us to compare the returns of different strategies based on our proposed model on a consistent basis, taking into account the length of time that the investment was held.
- **Annualized standard deviation** is a measure of the volatility or risk of an investment over a given time period, expressed as a percentage and annualized to account for the time value of money. In the context of Bitcoin trading, annualized standard deviation allows us to compare the volatility of different investments or trading strategies. The higher the annualized standard deviation of an investment, the more volatile or risky it is considered to be. Conversely, a lower annualized standard deviation indicates that an investment is less volatile and may be less risky.
- **Sharpe ratio** is a measure of the risk-adjusted return of an investment. In our report, the Sharpe ratio can be used to assess the performance of a Bitcoin investment relative to the level of risk involved. The higher the Sharpe ratio of an investment, the better the risk-adjusted return is considered to be. A Sharpe ratio of 1.0 or higher is generally considered to be good, while a ratio below 1.0 may indicate that the level of risk is not justified by the level of return.

The computation for these evaluation metrics will be explained in detail in the 'Experiments' section.

We will also plot a graph to visualize the performance of our three models to make it easier for comparison.

C. Data Collection and Data Exploration

Our dataset is downloaded from Binance, a digital currency exchange platform that allows users to buy, sell, and trade a variety of cryptocurrencies. The timeframe of our dataset consists of hourly historical BTC-USD prices from 2017-08-17 to 2021-10-07. In the dataset, there are five features: open, close, high, low, and volume.

For the purpose of our report, we will only use close price and trading volume from the downloaded dataset. We use the real-time values as our index to get a better understanding of the dataset and it also ensures that any model, we implement isn't affected by those time stamps. Then we add a derived feature, which is 'returns', the percentage change in closing prices every hour. Our new dataset has 36,168 rows and 3 columns as follows:

Date	Close	Volume	returns
2017-08-17 04:00:00	4308.83	47.181009	NaN
2017-08-17 05:00:00	4315.32	23.234916	0.001505
2017-08-17 06:00:00	4324.35	7.229691	0.002090
2017-08-17 07:00:00	4349.99	4.443249	0.005912
2017-08-17 08:00:00	4360.69	0.972807	0.002457
...	...	...	...
2021-10-07 05:00:00	54735.76	2251.122020	-0.006146
2021-10-07 06:00:00	54534.16	1783.004260	-0.003690
2021-10-07 07:00:00	54755.92	4163.431360	0.004058
2021-10-07 08:00:00	54538.30	2049.382180	-0.003982
2021-10-07 09:00:00	53995.50	2739.153610	-0.010002

Figure 2. Dataset with features: close, volume, and returns

In terms of data exploration, we visualize the three features used to build our models, which are close price, trading volume, and returns as seen in Figure 3, Figure 4, and Figure 5. These figures demonstrate there is a vigorous change in the volume of bitcoin over the observed period. Furthermore, the maximum trading volume of bitcoin reached nearly 46,000 in 2020, which is the highest, and the minimum volume was in 2017, which is around 100 bitcoins.

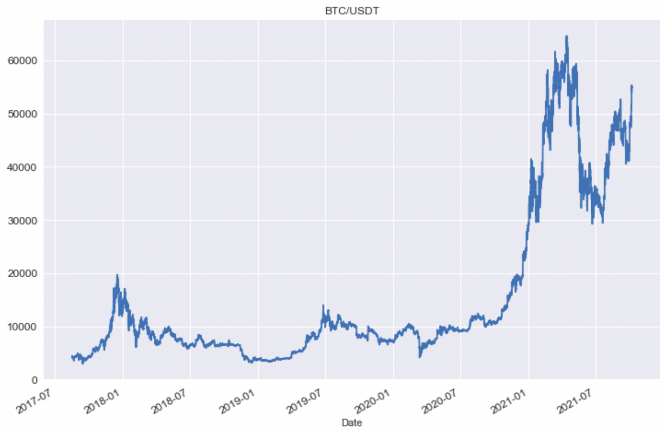

Figure 3. Graph of closing price over the whole timeframe

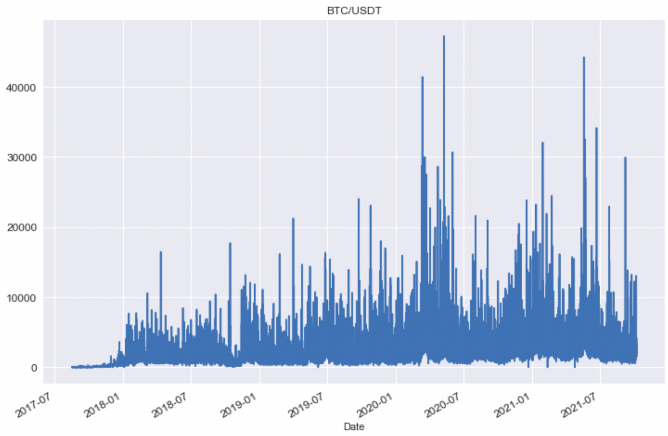

Figure 4. Graph of volume over the whole timeframe

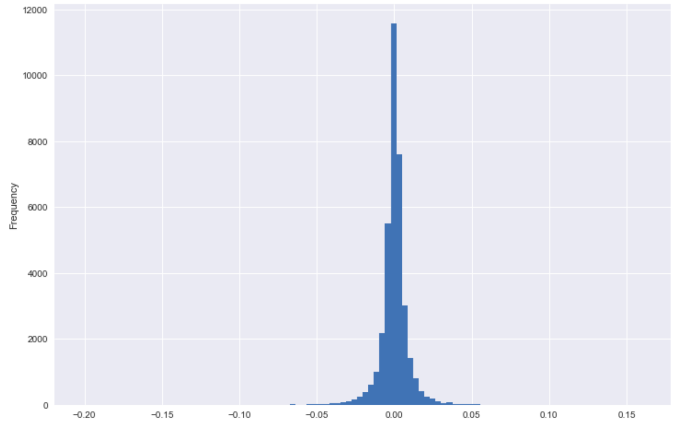

Figure 5. The distribution of the 'returns' feature

We also get the information on maximum positive returns and maximum negative returns, as in Figure 6 and Figure 7 respectively. Both these figures have shown the high volatility of bitcoin prices.

```
Date
2020-03-13 02:00:00    0.160280
2017-09-15 12:00:00    0.131731
2020-03-15 21:00:00    0.129546
2017-09-15 14:00:00    0.117777
2021-01-29 08:00:00    0.116145
2017-09-05 02:00:00    0.113257
2018-01-17 16:00:00    0.108790
2018-04-12 11:00:00    0.103325
2018-10-15 06:00:00    0.100727
2019-07-18 14:00:00    0.089576
Name: returns, dtype: float64
```

Figure 6. Maximum positive returns

Figure 6 shows that the highest positive return in an hour is 16%, demonstrating the high reward.

```
Date
2020-03-12 10:00:00   -0.201033
2020-03-12 23:00:00   -0.189707
2020-03-13 01:00:00   -0.119449
2017-12-28 02:00:00   -0.108097
2017-12-22 13:00:00   -0.107858
2017-09-05 01:00:00   -0.099818
2017-08-22 04:00:00   -0.098295
2020-03-15 22:00:00   -0.095180
2021-05-19 12:00:00   -0.093810
2019-09-24 18:00:00   -0.093730
Name: returns, dtype: float64
```

Figure 7. Maximum negative returns

Figure 7 shows that the highest loss is 20% in an hour, demonstrating high risk.

All of our features are continuous, and their characteristics are reflected in the below feature report:

	Close	Volume	returns
count	36168.000000	36168.000000	36167.000000
mean	15211.287479	2121.344201	0.000070
std	14918.059912	2211.660869	0.009669
min	2919.000000	0.000000	-0.201033
25%	6619.987500	910.157520	-0.002955
50%	9110.620000	1551.676864	0.000139
75%	13411.242500	2603.584828	0.003258
max	64577.260000	47255.762685	0.160280

Figure 8. Continuous Feature Report

When examining the dataset downloaded from Binance, we add the following notes in our Data Quality Report and our Data Preprocessing Plan:

- Our strategy is based on price and volume change. Hence, we only need those 2 columns for our machine learning model. Therefore, the first step would be to remove all the other unnecessary columns.
- After performing the first step we would be left with just 2 independent features. We cannot directly use those two columns because we want our models to work on changes in those values. Hence, for derived features, we will be getting two new columns which would have instances having values calculated based on the changes in the independent features every hour.
- More insight: The absolute price and volume changes are not useful in solving our problem. Thus, we will add new derived features, which are the percentage changes in those two independent features.
- There will be an issue of outliers as some values in the derived features would take values either equal to -infinity or +infinity. We need to tackle those outliers and we will replace those values with missing values and going ahead further would drop the rows containing those missing values. This is because the k-means clustering which we will be using in this algorithm doesn't allow missing values.

C. Baseline Model

Our baseline model is based on the buy and hold strategy. First, we calculate the investment multiple for every hour and normalized prices with base value 1, which is reflected in another derived feature 'creturns' as follows:

Date	Close	Volume	returns	creturns
2017-08-17 04:00:00	4308.83	47.181009	NaN	NaN
2017-08-17 05:00:00	4315.32	23.234916	0.001505	1.001506
2017-08-17 06:00:00	4324.35	7.229691	0.002090	1.003602
2017-08-17 07:00:00	4349.99	4.443249	0.005912	1.009552
2017-08-17 08:00:00	4360.69	0.972807	0.002457	1.012036
...	...	...	...	...
2021-10-07 05:00:00	54735.76	2251.122020	-0.006146	12.703161
2021-10-07 06:00:00	54534.16	1783.004260	-0.003690	12.656373
2021-10-07 07:00:00	54755.92	4163.431360	0.004058	12.707839
2021-10-07 08:00:00	54538.30	2049.382180	-0.003982	12.657334
2021-10-07 09:00:00	53995.50	2739.153610	-0.010002	12.531360

Figure 9. Our dataset after adding column 'creturns'

The visualization of 'creturns' is demonstrated in the below figure:

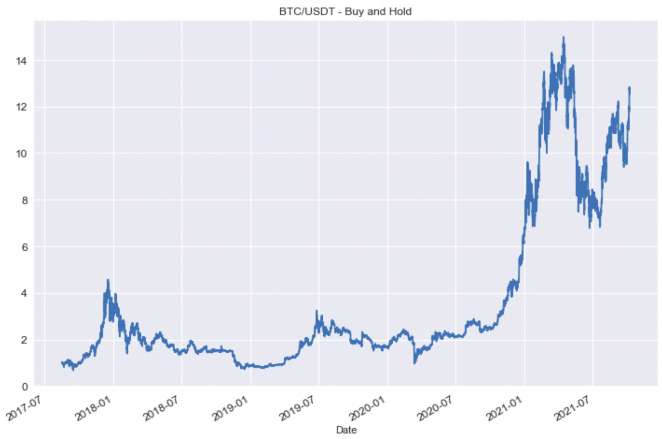

Figure 10. Visualized graph of 'creturns' feature

Then we compute the evaluation metrics of this strategy, which will be used as benchmark metrics for our later proposed models, and receive the following results:

- Benchmark annualized returns: 0.845
- Benchmark standard deviation: 0.905
- Benchmark Sharpe ratio: 0.934

D. Data preprocessing for Building the Trading Strategy

Now, we will add one more derived feature 'vol_ch' to reflect the percentage change of volume hourly. Our dataset is as follows:

Date	Close	Volume	returns	creturns	vol_ch
2017-08-17 04:00:00	4308.83	47.181009	NaN	NaN	NaN
2017-08-17 05:00:00	4315.32	23.234916	0.001505	1.001506	-0.708335
2017-08-17 06:00:00	4324.35	7.229691	0.002090	1.003602	-1.167460
2017-08-17 07:00:00	4349.99	4.443249	0.005912	1.009552	-0.486810
2017-08-17 08:00:00	4360.69	0.972807	0.002457	1.012036	-1.518955
...	...	...	...	...	...
2021-10-07 05:00:00	54735.76	2251.122020	-0.006146	12.703161	0.439863
2021-10-07 06:00:00	54534.16	1783.004260	-0.003690	12.656373	-0.233129
2021-10-07 07:00:00	54755.92	4163.431360	0.004058	12.707839	0.848040
2021-10-07 08:00:00	54538.30	2049.382180	-0.003982	12.657334	-0.708801
2021-10-07 09:00:00	53995.50	2739.153610	-0.010002	12.531360	0.290111

Figure 11. Our dataset after adding the volume change ('vol_ch') feature

When examining our volume change feature, we found that some instances had a value equal to $\pm infinity$, because there is a chance that there was absolutely no volume change in a given hour. Those are considered to be outliers and should be removed.

E. K-means Clustering

Our report applies k-means clustering to develop a trading strategy (Unsupervised Learning). There are two questions that we attempt to find the answer:

Question 1: Is there a relationship between price changes and volume changes?

(e.g. rapid Increase in Trading Volume triggers extreme Price changes)

Question 2: Can we use return/vol_ch clusters to (partly) forecast future returns? Can our machine learning model pick that relation and use it?

We plot the two-dimensional graph showing the percentage price and volume changes values below:

Figure 12. A graph to visualize the percentage price and volume changes values

To perform clustering, we will transform our dataset to remove the time index, and only use volume change feature and returns feature for our clustering. The parameters for the clustering include: n_clusters = 10, init = 'kmeans++'. Our clustering can be visualized as below:

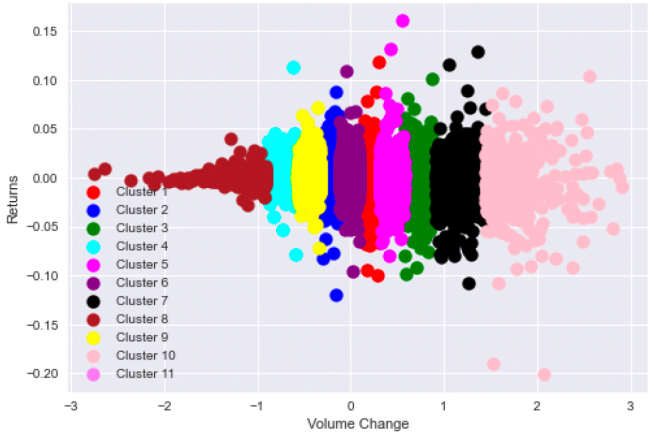

Figure 13. Visualization of the clustering

Figure 13 shows that the K-means Clustering tries to cluster the points in such a way that it can establish a relationship between price changes and volume changes, and hence, clusters those points to depict the same. Then we use the formed clusters to predict a decision for the trading strategy for the next hour.

Then we add the cluster to our dataset and calculate mean returns for the next hour and plot it in a matrix for each cluster by calculating the average mean return for the next hour over the whole timeframe. A graph of the matrix for better visualization and understanding is as below:

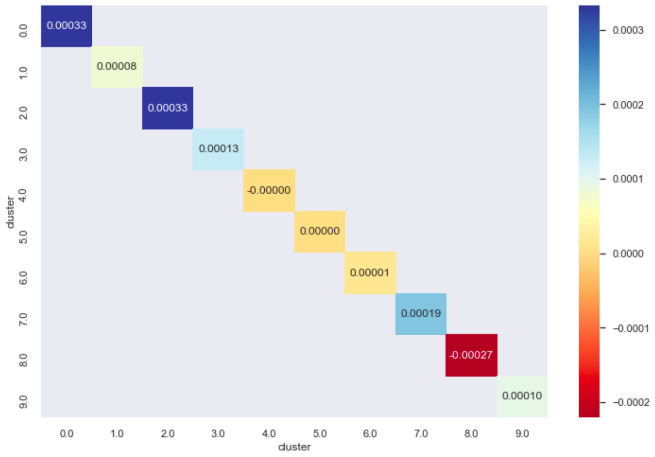

Figure 14. Matrix of mean returns for the next hour

Figure 14 shows two clusters having negative returns in the next hour: cluster 4 and cluster 8. We then add another column, 'trading position' to our dataset with either 1 or 0. We set value=1 (hold onto the asset) for every instance initially because we only want to sell our asset when we get a selling signal (value=0) from our model. We generate sell signals for the clusters having negative returns for the next hour with 2 conditions as follows:

condition1 = data_clustering.cluster == 4

condition2 = data_clustering.cluster == 8

As a result, we have 27,259 hold decisions and 8,891 sell decisions. A part of this result is shown as follows:

	vol_ch	returns	cluster	position
1	-0.708335	0.001505	3	1
2	-1.167460	0.002090	7	1
3	-0.486810	0.005912	8	0
4	-1.518955	0.002457	7	1
5	2.403742	0.018925	9	1
...	...	...	...	...
36163	0.439863	-0.006146	4	0
36164	-0.233129	-0.003690	1	1
36165	0.848040	0.004058	2	1
36166	-0.708801	-0.003982	3	1
36167	0.290111	-0.010002	0	1

Figure 15. Adding trading position (1 = hold and 2 = sell)

F. Logistic Regression

Our report applies Logistic Regression to develop a trading strategy (Supervised Learning). First, we generate hold and sell values based on the change in absolute values of volume, and populate our dataset with obtained values:

	vol_ch	returns	cluster	position	strategy_kmeans	cstrategy_kmeans	creturns	output
1	-0.708335	0.001505	3	1	NaN	NaN	1.001506	1
2	-1.167460	0.002090	7	1	0.002090	1.002093	1.003602	1
3	-0.486810	0.005912	8	0	0.005912	1.008034	1.009552	1
4	-1.518955	0.002457	7	1	0.000000	1.008034	1.012036	0
5	2.403742	0.018925	9	1	0.018925	1.027292	1.031370	0
...	...	...	...	...	...	...	...	...
36163	0.439863	-0.006146	4	0	-0.006146	47.646763	12.856169	1
36164	-0.233129	-0.003690	1	1	-0.000000	47.646763	12.808817	0
36165	0.848040	0.004058	2	1	0.004058	47.840516	12.860904	1
36166	-0.708801	-0.003982	3	1	-0.003982	47.650380	12.809790	0
36167	0.290111	-0.010002	0	1	-0.010002	47.176133	12.682299	0

Figure 16. Our dataset after adding hold and sell values

After dropping nan values, we will create variables for the Logistic Regression model and train the model on it. We import the Logistic Regression class from the sklearn.linear_model library for that. Then we create an instance of that class and fit it to our model for training.

V. EXPERIMENTS

To evaluate and compare the effectiveness of our models, we use three evaluation metrics: annualized returns comparison, annualized standard deviation comparison, and Sharp ratio, and a graph to visualize all models' performance.

- **Annualized return computation:** To calculate the annualized return on a bitcoin investment, we first need to determine the total return on your investment over the given time period. This is done by taking the difference between the final value of the investment and the initial value and dividing it by the initial value. To annualize this return, we then divide the total return by the number of years that the investment was held and multiply the result by time period/year. Our time period/year =24*365.25 (365 days/year and 24 hours/day). The mathematical formula is expressed as below, where a is value of different strategy in the given formula and n is the number of instances.

$$Annualized\ returns = \frac{\sum_{i=1}^{\Pi}(a)}{n} * \text{time period/year}$$

- **Annualized Standard Deviation computation:** It is calculated by taking the standard deviation of the bitcoin investment's returns over a certain number of periods and multiplying it by the square root of the number of the time period/year. The mathematical of standard deviation and annualized standard deviation are as follows:

$$sd(a) = \sqrt{var(a)} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(a_i - \bar{a})^2}$$

Annualized standard deviation =sd *squareroot of (time period/year)

- **Sharpe ratio computation:** To calculate the Sharpe ratio for a Bitcoin investment, we divide the annualized returns with by the annualized standard deviation. The mathematical formula of Sharpe ratio can be expressed as follows, in which 'arc' is the annuliazed return and 'asd' is the annual standard deviation.

Sharp ratio=((e^arc)-1)/asd

- **Performance graph plotting:** to plot a graph to visualize the performance of our models, we normalize price with base = 1 for the proposed models and get the results in Figure 17.

	vol_ch	returns	cluster	position	strategy_kmeans	cstrategy_kmeans	creturns	output	y_pred	strategy_lr	cstrategy_lr
2	-1.167460	0.002090	7	1	0.002090	1.002093	1.003602	1	0	NaN	NaN
3	-0.486810	0.005912	8	0	0.005912	1.008034	1.009552	1	0	0.000000	1.000000
4	-1.518955	0.002457	7	1	0.000000	1.008034	1.012036	0	0	0.000000	1.000000
5	2.403742	0.018925	9	1	0.018925	1.027292	1.031370	0	1	0.000000	1.000000
6	0.837305	0.003594	2	1	0.003594	1.030991	1.035084	0	1	0.003594	1.003600
...	...	...	...	...	...	...	...	...	...	...	...
36163	0.439863	-0.006146	4	0	-0.006146	47.646763	12.856169	1	1	-0.006146	16.106992
36164	-0.233129	-0.003690	1	1	-0.000000	47.646763	12.808817	0	0	-0.003690	16.047668
36165	0.848040	0.004058	2	1	0.004058	47.840516	12.860904	1	1	0.000000	16.047668
36166	-0.708801	-0.003982	3	1	-0.003982	47.650380	12.809790	0	0	-0.003982	15.983888
36167	0.290111	-0.010002	0	1	-0.010002	47.176133	12.682299	0	1	-0.000000	15.983888

Figure 17. Results after normalizing price with base = 1

According to the above figures, the values that we will use to plot the performance of each model are from the 'creturns' column (for baseline model), 'cstrategy_kmeans' (for K-means model), and 'cstrategy_lr' column (for Logistic Regression model)

VI. RESULTS

The results of the evaluation metrics of our 3 models are summarized in the following table:

TABLE 1

Performance evaluation metrics of the models

Evaluation metrics	Baseline	K-means	Logistic Reg
Annualized returns	0.615	0.934	0.672
Annualized std-dev	0.905	0.788	0.70
Sharpe ratio	0.939	1.960	1.350

The above table shows that annualized return obtained by K-means clustering trading strategy is much higher than all other models. Regarding standard deviation, the risk associated with our asset has reduced the most in the Logistic Regression model. Standard deviation of 70% is still a high-risk investment but we already knew about that

from the domain knowledge that investing in cryptocurrencies is highly risky. The Sharpe ratio of K-means clustering model is more than double the Sharpe ratio of baseline model. It is also much higher than the logistic regression model. This indicates a much better performance from the unsupervised model which we implemented than both the other models.

The performance of our two proposed models and baseline model is shown in Figure 18, in which the green line represents the buy and hold strategy, the orange line represents the K-means strategy, and the green line shows the Logistic Regression strategy.

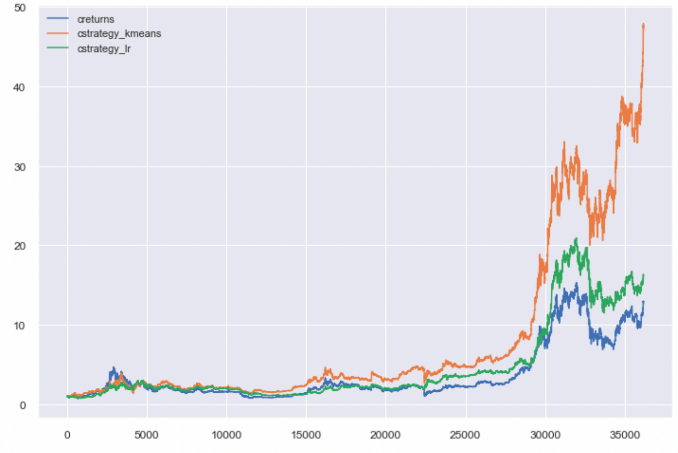

Figure 18. Visualization of our models' performance

According to the above graph, both our proposed models perform better than our benchmark model (buy and hold strategy). Our trading strategy based on K-means has the best performance result.

VII. CONCLUSION

After evaluating the performances of all three models we can conclude that the k-means clustering model outperforms the other two when it comes to annualized returns. This is because the clusters formed establish a strong relationship which is theoretically based on contrarian mean reverting signal. The investment multiple clearly shows how dominant this signal is and it helps the investor to nearly get 4 times the returns than the baseline model. The Logistic Regression model also performs better than the baseline model, but it is not a strong performer as the k-means model. This is because the number of features in this dataset is less so there was always a chance that the classification model under-fits. We would be better served to use such supervised models when we have more insights into the data so that it has more instances and features to train on.

The risk associated while trading any stock or cryptocurrency is an important parameter to consider. We have used standard deviation as a measure of that as it clearly shows the volatility in the given asset. The risk associated with the buy and hold strategy is the highest and it makes perfect sense because the investor is holding the asset no matter what all the time. This directly increases the risk as it makes the investor more susceptible to the high volatility periods where the risk associated with the asset is maximum. The values for standard deviation were lowest for the logistic regression model as it had more instances where the investor sells the asset which makes him less vulnerable to the volatile nature of the cryptocurrency.

To compare the performance using a technical indicator we calculated the Sharpe ratio for each strategy. The Sharpe ratio compares the return of an investment with its risk. The k-means clustering strategy has the highest Sharpe ratio among all the models indicating that it is technically superior to the other strategies as the value justifies the rewards to the risk ratio. However, it should be noted that the logistic regression model also has a good value for the Sharpe ratio. It is because the risk associated (the value of standard deviation) with that strategy was minimum and even though the rewards are not as significant as k-means, the risk factor makes up some ground for that and hence the model returns a good score of Sharpe ratio overall. As expected, the Sharpe ratio of the baseline model was found to be the least.

The final conclusion one can derive from this paper is that more emphasis should be placed on unsupervised learning models when it comes to trading cryptocurrencies or any other stock for that matter. The major research done in the field is based on supervised learning models and statistical models and even though many of them perform really well, it can be combined with unsupervised models to derive more insightful results and would benefit an investor in the longer run.

REFERENCES

[1] Binance Academy, "Cryptocurrency," Binance Academy. [Online]. Available: https://academy.binance.com/en/glossary/cryptocurrency. [Accessed: 18-Dec-2022].

[2] "Americans (24%) more likely than Canadians (17%) to invest in bitcoin ..." [Online]. Available: https://www.ipsos.com/en-ca/news-polls/americans-

more-likely-than-canadians-to-invest-in-bitcoin. [Accessed: 20-Dec-2022].

[3)] About the author: Elizabeth Renter's work as a senior writer and data analyst at NerdWallet has been cited by The New York Times, "Survey: Crypto is popular but plagued with misconceptions," NerdWallet. [Online]. Available: https://www.nerdwallet.com/article/investing/study-crypto-misconceptions. [Accessed: 20-Dec-2022].

[4] C. Badger, "Top 7 mistakes people make when investing in crypto," Medium, 13-Aug-2021. [Online]. Available: https://medium.com/coinmonks/top-7-mistakes-people-make-when-investing-in-crypto-24a4ade5056. [Accessed: 20-Dec-2022].

[5] Katsiampa, P. Volatility estimation for Bitcoin: A comparison of GARCH models. *Econ. Lett.* **2017**, *158*, 3–6. https://www.sciencedirect.com/science/article/pii/S0165176517302501?casa_token=_5jjYjEq9msAAAAA:3O57oh370Y7j51J6HB2NhXZ1CEppqfhttzOunFrm34J2jN6rWA5Rm0qXBZpzAcIA6gFX-rVPyAo

[6] Roy, S.; Nanjiba, S.; Chakrabarty, A. Bitcoin price forecasting using time series analysis. In Proceedings of the International Conference of Computer and Information Technology, Dhaka, Bangladesh, 21–23 December 2018; Volume 1, pp. 1–5.

https://scholar.google.com/scholar_lookup?title=Bitcoin+price+forecasting+using+time+series+analysis&conference=Proceedings+of+the+International+Conference+of+Computer+and+Information+Technology&author=Roy,+S.&author=Nanjiba,+S.&author=Chakrabarty,+A.&publication_year=2018&pages=1%E2%80%935&doi=10.1109/ICCITECHN.2018.8631923

[7] Jana, R.K.; Ghosh, I.; Das, D. A differential evolution-based regression framework for forecasting Bitcoin price. *Ann. Oper. Res.* **2021**, *306*, 295–320. https://link.springer.com/article/10.1007/s10479-021-04000-8

[8] Kim, J.M.; Cho, C.; Jun, C. Forecasting the Price of the Cryptocurrency Using Linear and Nonlinear Error Correction Model. *J. Risk Financ. Manag.* **2022**, *15*, 74. https://www.mdpi.com/1911-8074/15/2/74

[9] Jang, H.; Lee, J. An Empirical Study on Modeling and Prediction of Bitcoin Prices With Bayesian Neural Networks Based on Blockchain Information. *IEEE Access* **2018**, *6*, 5427–5437. https://ieeexplore.ieee.org/abstract/document/8125674

[10] Li, Y.; Dai, W. Bitcoin price forecasting method based on CNN-LSTM hybrid neural network model. *J. Eng.* **2020**, *2020*, 344–347. https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/joe.2019.1203

[11] Jay, P.; Kalariya, V.; Parmar, P.; Tanwar, S.; Kumar, N. Stochastic Neural Networks for Cryptocurrency Price Prediction. *IEEE Access* **2020**, *8*, 82804–82818. https://ieeexplore.ieee.org/abstract/document/9079491

[12] Loginova, E.; Tsang, W.K.; van Heijningen, G.; Kerkhove, L.; Benoit, D.F. Forecasting directional bitcoin price returns using aspect-based sentiment analysis on online text data. *Mach. Learn.* **2021**. https://link.springer.com/article/10.1007/s10994-021-06095-3

[13] Parekh, R.; Patel, N.P.; Thakkar, N.; Gupta, R.; Tanwar, S. DL-GuesS: Deep Learning and Sentiment Analysis-based Cryptocurrency Price Prediction. *IEEE Access* **2022**, *10*, 35398–35409. https://ieeexplore.ieee.org/abstract/document/9745117

[14] Ibrahim, A.; Kashef, R.; Li, M.; Valencia, E.; Huang, E. Bitcoin network mechanics: Forecasting the btc closing price using vector auto-regression models based on endogenous and exogenous feature variables. *J. Risk Financ. Manag.* **2020**, *13*, 189. https://www.mdpi.com/1911-8074/13/9/189

[15] Livieris, I.E.; Pintelas, E.; Stavroyiannis, S.; Pintelas, P. Ensemble deep learning models for forecasting cryptocurrency time-series. *Algorithms* **2020**, *13*, 121. https://scholar.google.com/scholar_lookup?title=Ensemble+deep+learning+models+for+forecasting+cryptocurrency+time-series&author=Livieris,+I.E.&author=Pintelas,+E.&author=Stavroyiannis,+S.&author=Pintelas,+P.&publication_year=2020&journal=Algorithms&volume=13&pages=121&doi=10.3390/a13050121

[16] Shin, M.; Mohaisen, D.; Kim, J. Bitcoin price forecasting via ensemble-based LSTM deep learning networks. In Proceedings of the 2021 International Conference on Information Networking (ICOIN), Jeju Island, Korea, 13–16 January 2021; Volume 1, pp. 603–608. https://ieeexplore.ieee.org/abstract/document/9333853?casa_token=4hvoSgb2ovYAAAAA:IGqplQtKK57bJgq9eskGYIE8g0R_EgJIWZ6W_IMpuVSRsTAdeifTx8mqXkoKckpYRnhWyPC2Eg