UNIVERSITY PARTNER
UNIVERSITY OF WOLVERHAMPTON
HERALD COLLEGE
KATHMANDU

6CS030 - Big Data

Portfolio

Big Data Analytics for Mitigating the Adverse Economic Impact of Nepal Stock Market Crash

Student Name	- Sudip Neupane
University ID	- 2050436
Section	- L6CG9
Module leader	- Mr. Jnaneshwar Bohara
Lecturer	- Mr. Chiranjivi Khanal
Tutor	- Mr. Yamu Poudel
Submitted Date	- 2022-05-02

Abstract

The stock market generates a large amount of data that contains some hidden information that can help in making better decisions. Different analyses on the basic hypothesis have been performed in order to provide appropriate results and make successful data-driven decisions. After the covid pandemic there has been a massive increment in traders. But the market is facing down trends which directly effects country GDP. As per Napal stock exchange limited, Nepal stock market fall down to 23 percentage in just 3 months. The NEPSE index is based on the total market capitalization of the transactions of firms. Selecting patterns and trends to compare with the news and political situation aids in the development of a relationship with the changes. Stop-loss orders are recommended by technical analysis to safeguard investors from a huge loss in the future. Dataset has been fetched from Kaggle store. The data was in multiple files which required some data cleaning. Multiple methods of visualization, analysis has been done to extract information. Two different models were used in evaluation the performance of dataset. The work done can help new traders to get in market with a surface knowledge. Although the market is not that easy to predict by a number in a day, but we can predict inside a certain boundary.

Table of Contents

Table of Figures

1. Background of the study

1.1 Generic Information

The share market is a vital institutional mechanism that directs investment to where it is most required in the economy. The stock market's operations of buying and selling securities are critical to the effective allocation of capital within economies. If the financial transmission mechanism, such as share market, is inefficient, the cash to actual investment would be hampered, and activity levels will fall beyond potential. Banks and the stock market compete to shift savings into investment. Stock markets, on the other hand, outperform banks when it comes to allocation efficiency. The importance of data analysis in market analysis cannot be overstated. It gives a solid foundation for important judgments. We cannot truly predict the future price but can have insight from past data to see verities of patterns to outstand the possibilities of future (Mainali, 2011). According to the board, due to the expansion of the securities market and the reforms implemented by the board, the securities have over four million traders in the primary market, which is about 14% of the total population. Small investors should have easy access to finance so they can participate in the stock market (Singh, 2021).

1.2 Problem Statement

Nepal Stock Exchange (NEPSE) is the only stock exchange in Nepal. It has been facing bearish trends. NEPSE index was once 3198.60 but as of today (2022 April 20) it is 2,320.52 (Nepal Stock Exchange Ltd, 2022). Stock prices are falling, consumers and companies have less wealth and optimism, leading to less spending and lower GDP (Gross domestic product). Following the covid epidemic, there has been a tremendous increase in dealers. However, the market is experiencing downward tendencies, which has a direct impact on the country's production. The Nepal stock market has dropped by 23 percent in last three months.

1.3 Aim/objective of the work

NEPSE index depends on total market capitalization of companies' transactions. The mythical change in price is by the news. Picking that patterns and trends to compare with the news and political condition helps in creating relation with the changes. The higher the relation, higher is its impact. The pattern in different sub-indexes will give a relation between sub-indexes. Technical analysis favors the use of stop losses because stock prices are affected by changing investor sentiment based on news and events. This will save investors from huge losses in future. Technical analysis is only useful for companies with high demand and high trading volume..

1.4 Contributions of the work connected with Methodology

Nepal stock market has lots of up and downs through the past years. So, I picked that problem for gaining some insights to solve it. Dataset was too large, I divided it in multiple sectors by myself. There were some initial hypotheses that I was trying to meet. Multiple sources of research and past work were viewed to know where to start. All the data were preprocessed, and the analysis was done. The visual representation can help in technical analysis of the company and sector. Heatmap correlation helps in finding the relation of a attribute with all other attribute. Then the model was trained with two supervised learning algorithms. The outcome of both model was better with an accuracy of 99 percentage. I recap my findings and evaluate the findings relating to my hypothesis, aims and objectives.

1.5 Organization of the Report

Cover Page:

Templet provided by University which includes title of project

Abstract:

In this part, you will get a quick glimpse at the complete case study.

Background of the study:

It is a whole summary of choosing the dataset as a problem domain, goals that might help in solving that problem domain, objectives of work and Methodology-related work contributions.

Related Work:

It discusses various works done in past related to market analysis. It is a literature review and a summary of different work in the same field.

Methodology:

In this part, the methodology for a project is briefly addressed, using the usage of a Block Diagram/Phases to illustrate the method.

Result and Discussion:

Different approaches of reading, cleaning, analyzing, visualizing and building model for the data is discussed in result and discussion section. The findings of work done and analyzing the findings of work done are also included here.

Conclusion:

It recaps overall work carried out by comparing the different approaches to complete intended goals and objectives. The overall outcomes of work and report are described here.

Reference:

It has all source of information used to complete the research.

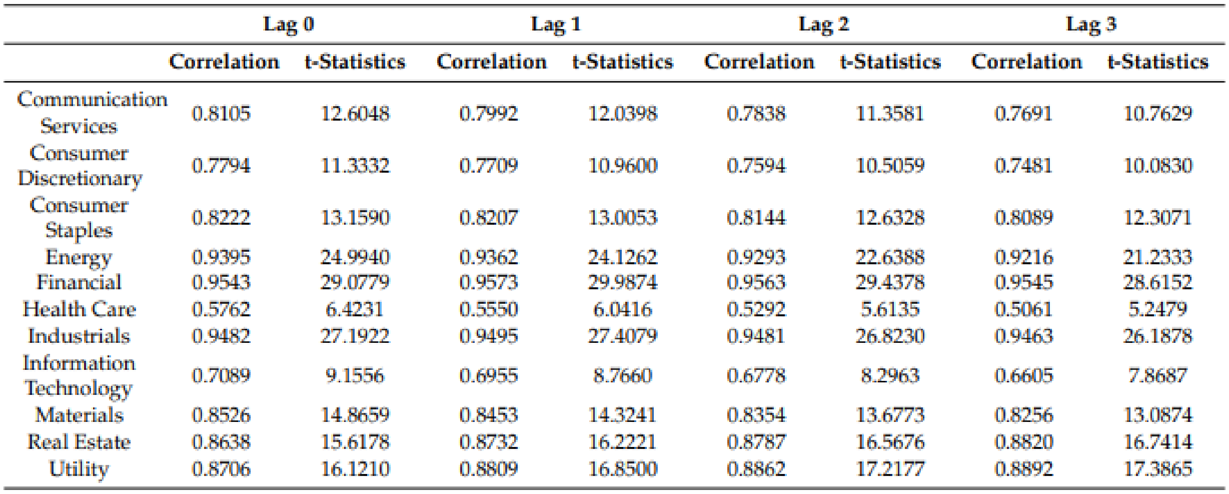
2. Related Work:

During the coursework development and analysis, several literatures and related works were researched. From them, a couple of relevant work are discussed in sections below.

2.1 Using Big Data to Investigate the Initial Impact of COVID-19 Sentiment on the US Stock Market

Using big data, this study examines the initial impact of the COVID-19 mood on US stock markets. This survey uses Daily News Sentiment Index (DNSI) and Google Trends data for coronavirus-related search queries to reveal COVID-19 sentiment and selected US stock market index from January 21, 2020, to May 20, 2020. Many sentiment analysis studies use Twitter data to predict stock market movements, but very little DNSI or Google Trends data. Furthermore, by building a time series regression model with additional industry returns as dependent variables, the purpose of this study is to investigate how changes in DNSI predict the outcome of the US market.

The Fama-French three-factor model is used to calculate higher stock market returns. The results of this study provide a comprehensive overview of the key impacts of COVID-19 on the US stock market, as well as strategic investment planning based on time lag perspectives, by showing correlation level adjustments through time lag changes. Part of the visualization performed in this research using the sample data set is shown in the following figures. The dataset shows test results for the Daily Emotions News Index. And the visualization is related to the change in the level of relationship between the 11 segment indicators and the DNSI with the difference in time delay. (Lee, 2020)



	Lag 0		Lag 1		Lag 2		Lag 3	
	Correlation	t-Statistics	Correlation	t-Statistics	Correlation	t-Statistics	Correlation	t-Statistics
Communication Services	0.8105	12.6048	0.7992	12.0398	0.7838	11.3581	0.7691	10.7629
Consumer Discretionary	0.7794	11.3332	0.7709	10.9600	0.7594	10.5059	0.7481	10.0830
Consumer Staples	0.8222	13.1590	0.8207	13.0053	0.8144	12.6328	0.8089	12.3071
Energy	0.9395	24.9940	0.9362	24.1262	0.9293	22.6388	0.9216	21.2333
Financial	0.9543	29.0779	0.9573	29.9874	0.9563	29.4378	0.9545	28.6152
Health Care	0.5762	6.4231	0.5550	6.0416	0.5292	5.6135	0.5061	5.2479
Industrials	0.9482	27.1922	0.9495	27.4079	0.9481	26.8230	0.9463	26.1878
Information Technology	0.7089	9.1556	0.6955	8.7660	0.6778	8.2963	0.6605	7.8687
Materials	0.8526	14.8659	0.8453	14.3241	0.8354	13.6773	0.8256	13.0874
Real Estate	0.8638	15.6178	0.8732	16.2221	0.8787	16.5676	0.8820	16.7414
Utility	0.8706	16.1210	0.8809	16.8500	0.8862	17.2177	0.8892	17.3865

Figure 1 Test Results for Daily News Sentiment Index

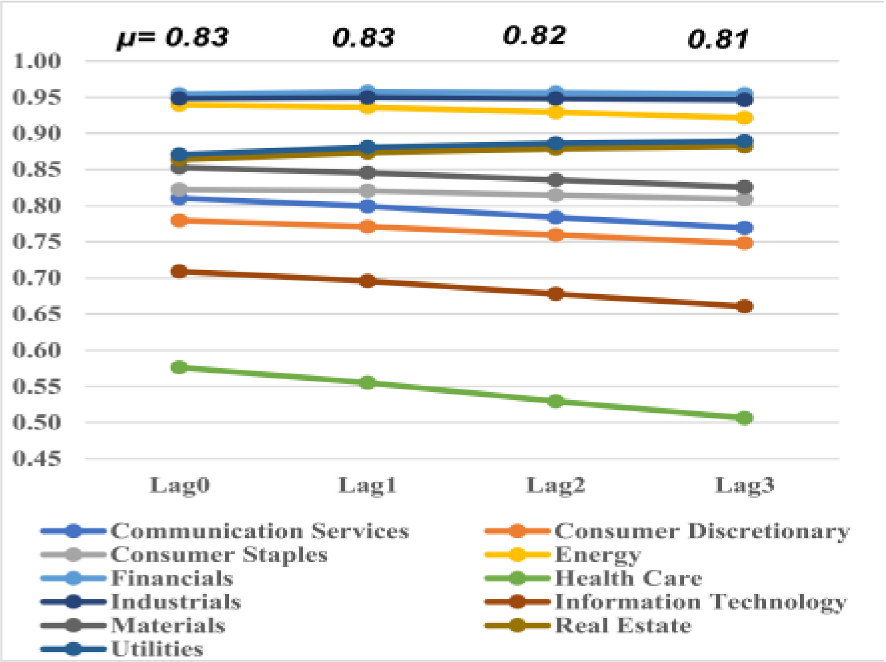


Figure 2 visualization is about the variation in the relationship level between 11 sector indices and DNSI by time lag difference

2.2 Stock Market Prediction Using Machine Learning:

LSDM - Lagrangian stochastic dispersion model

Machine learning is the latest trend in stock market forecasting technologies, with forecasters training their previous values based on the values of existing stock market indices. Machine learning uses a variety of models to make accurate predictions. Stock price forecasting research using regression and LSDM based machine learning (long-term memory). Using machine learning techniques, this study attempted to anticipate a company’s stock values with enhanced reliability and accuracy. The key role of researchers is to implement an innovative LSTM model as a technique for estimating stock prices.

Both strategies showed an increase in prediction accuracy, which leads to better results, making the LSTM model more efficient. The results are encouraging and support the theory that machine learning techniques can be used to assess stock market movements. Additionally, you can explore other machine learning models to see what percentages of accuracy they produce. Sentiment analysis using machine learning to see how news affects company stock values is an interesting topic. It can also be predicted using other models based on deep learning. The two figures supporting this result show a chart using the inverse between price and date and the results of the LSTM-based model are shown. (Agarwal, Parmar, & Saxena, 2018)

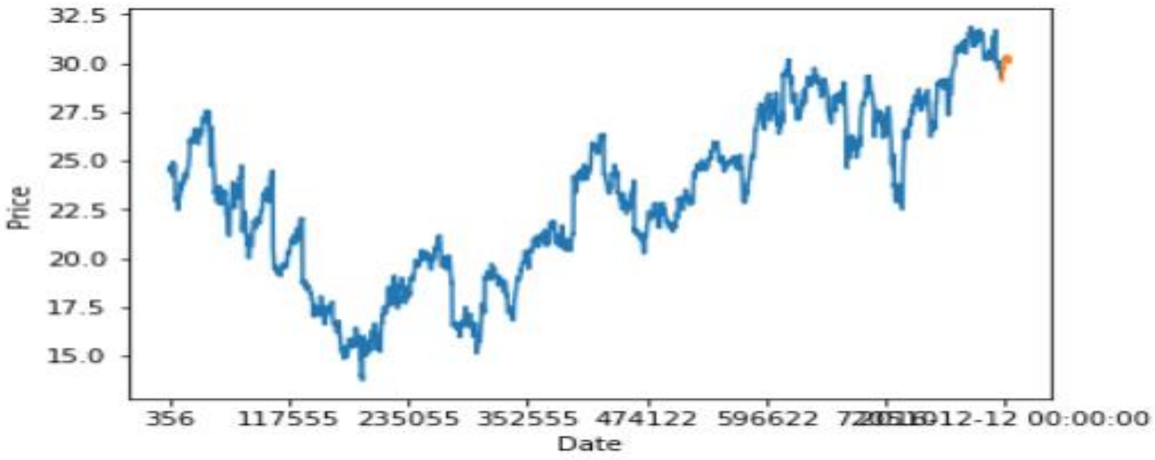


Figure 3 Plot between Date and Price using Regression

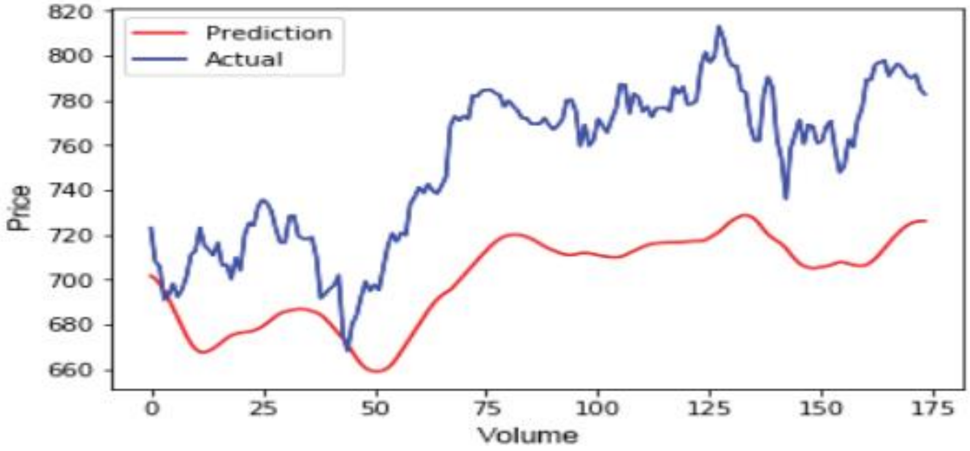


Figure 4: Plot between predicted and actual trend of LSTM

2.3 Stock Market Analysis using Supervised Machine Learning

To help make this unpredictably volatile business model more predictable, the paper recommends using a machine learning model to forecast future stock values for exchange using open-source libraries and pre-existing algorithms. We'll see if this straightforward approach produces satisfying outcomes. The outcome is fully reliant on numbers, and it is based on a number of assumptions that may or may not hold true in practice, such as the forecast time. WIKI retrieved the dataset for GOOGL. The beginning price, the highest price of the day, the lowest price, the closure price, and the volume are all included in the dataset (total transactions).

We employ "Adjusted Open, Adjusted High, Adjusted Low, Adjusted Close, and Adjusted Volume" to extract the information that will help us anticipate the result better. We will use close price attribute as our target. Adjusted close is a crucial source of data since it determines the market's opening price and volume expectation for the next day. High low is derived by

$$HLpCT = \frac{Adj.High - Adj.Low}{Adj.Close} \times 100$$

$$PCTchange = \frac{Adj.Close - Adj.Open}{Adj.Open} \times 100$$

Adjusted volume is a critical selection element since volume traded has the most direct influence on future stock prices of any feature. As a result, we'll apply it to our situation.

The simplest classification is linear regression, which is defined by the Sklearn library in the Scikit-learn package. Linear regression is a widely used tool for data analysis and forecasting. It uses only key attributes to predict relationships between variables based on their dependence on other variables. If you look at the classification attribute, it only remembers the label. In this example a few days ago, he recalls the combination of qualities and the position associated with them. This will find out which model the functions use to create your own label. Here's how it works using guided machine learning. (Pahwa & Agarwal, 2019)

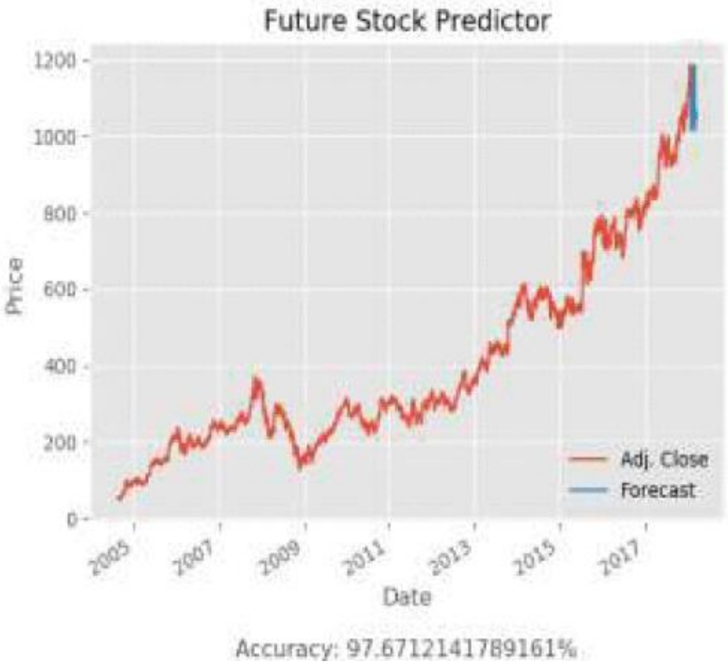


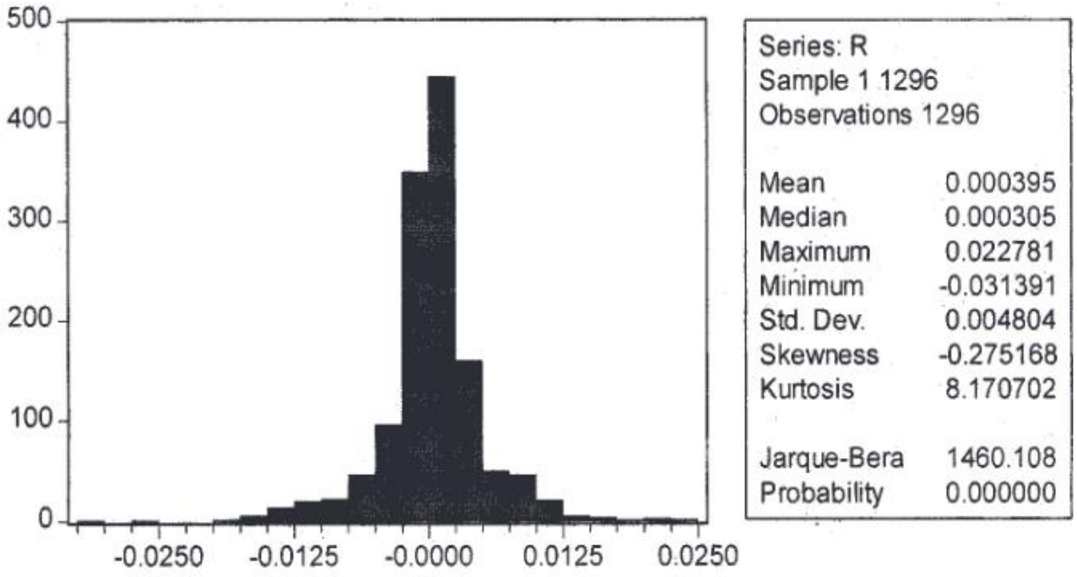
Figure 5 stock price of GOOGL from 2005 to 2018. Red line represents given data and blue represents the predicted value

2.4 Volatility Analysis of Nepalese Stock Market

GARCH - Generalized Autoregressive Conditional Heteroskedasticity

With the identification of time-varying variance, volatility clustering, and asymmetric response of volatility to price fluctuations, financial economists have focused on analysis and forecasting capital economic uncertainty. Given the expected expansion of the Nepalese stock market and the growing interest of investors in investing in the Nepalese stock market, it is critical to comprehend the stock market's volatility trend. The volatility of the Nepalese stock market is modeled in this research utilizing a daily return series of 1297 data from July 2003 to February 2009, as well as several classes of estimators and volatility models.

The outcomes show that GARCH is the best model for demonstrating unpredictability in the Nepalese market, as there is no significant imbalance in the contingent instability of profits. The review figured out significant proof of opportunity fluctuating unpredictability in the Nepalese financial exchange, as well as an inclination for high and low instability periods to group, as well as high instability industriousness and consistency. (G.C., 2008)



Series: R
Sample 1 1296
Observations 1296

Mean	0.000395
Median	0.000305
Maximum	0.022781
Minimum	-0.031391
Std. Dev.	0.004804
Skewness	-0.275168
Kurtosis	8.170702
Jarque-Bera	1460.108
Probability	0.000000

Figure 6 Descriptive Statistics of NEPSE Index Return Series

3. Methodology

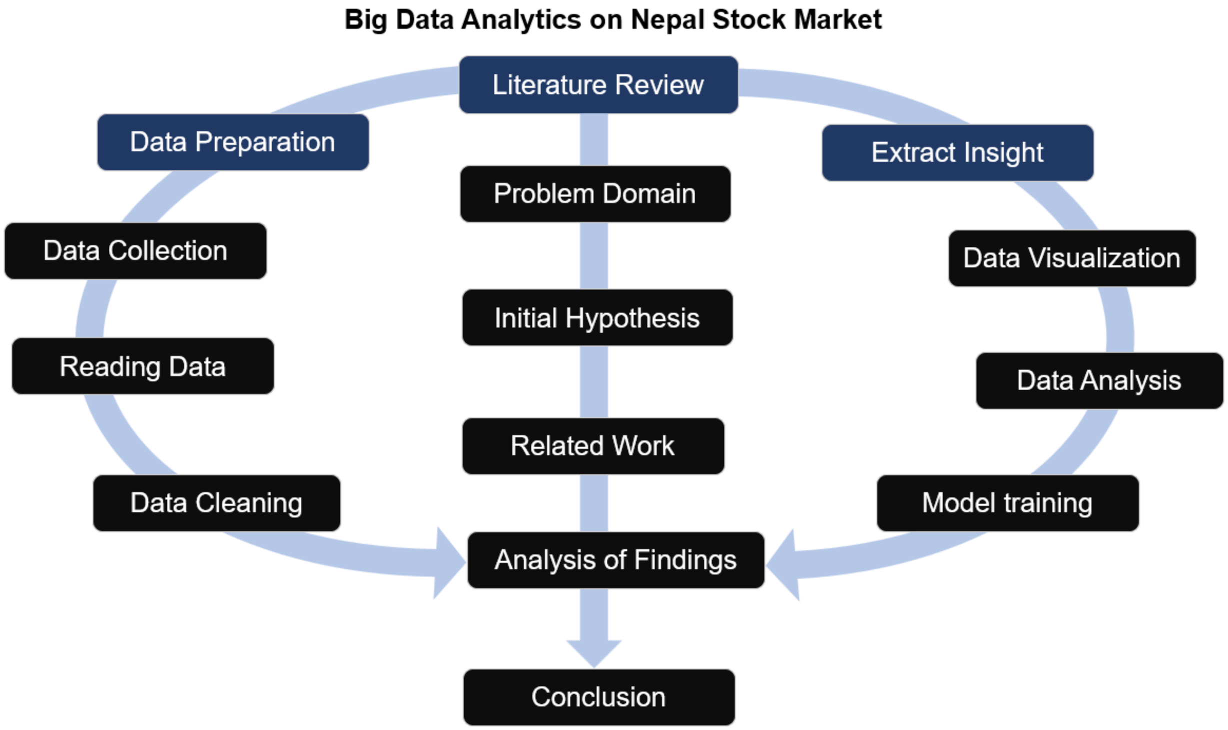


Figure 7 Entire work process on a Block Diagram using design science research methodology

The below table describes the overall methodology used for research and findings to accomplish the task. Having a methodology aid in determining the best ways to plan, execute, control, and deploy a project throughout the continuous implementation phase until it is completed and terminated successfully.

DSRM	Major Activities	Activity Description
Literature Review	Problem Domain	Recognize the problem's importance, as well as its current solutions and their flaws.
	Initial Hypothesis	Making some assumption can help in early research
	Related Work	Previous work has given me an understanding of what is conceivable and achievable. Knowledge of methodologies, technologies, and ideas that can aid in the definition of goals by understanding relevant previous work.
	Data Collection	Collecting data from relevant source. NEPSE data can be found on Kaggle.

Data Preparation	Reading Data	Loading multiple files into a single workflow, creating it understandable and extractable easily.
	Data Cleaning	Correcting or replacing data in a dataset that is inaccurate, corrupted, poorly structured, duplicate, or missing.
Extract Insight	Data Visualization	Data visualization tools make it easy to observe and identify trends, outliers, and patterns in the data by employing visual components like charts, graphs, and images.
	Data Analysis	Using statistical and/or analytical approaches to analyze and demonstrate, simplify and summaries, and evaluate information in a logical way.
	Model Training	Multiple approach of training, validating, and evaluating the different models for better performance.
Analysis of Findings		Using statistical and/or analytical approaches to analyze and demonstrate, simplify and summaries, and evaluate information in a logical way.
Conclusion		An opinion or decision that has been produced after the completion of study and investigation.

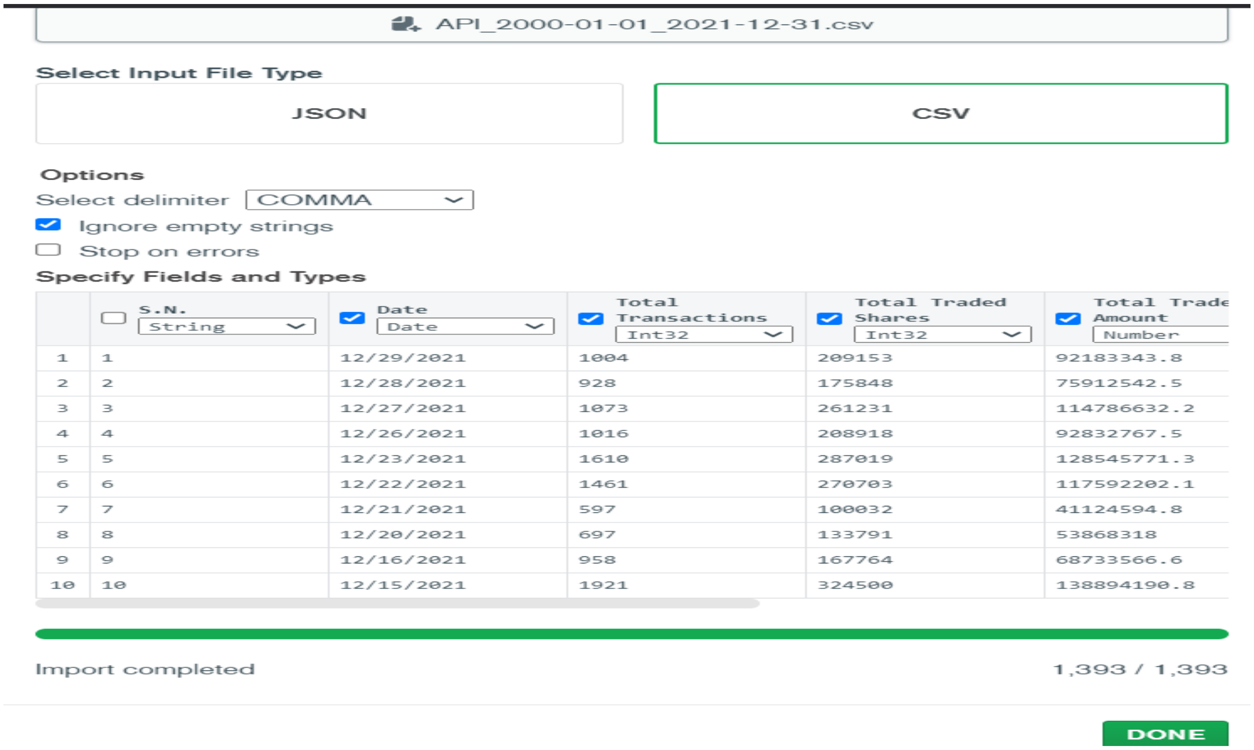
4. Result and Discussion

4.1 Experimental Setup

4.1.1 Read in and Explore Data

Reading data in mongoDB

Mongodb provides both cli and gui for importing data. We can import both csv and json file. While importing data, I remove the serial number column and change the datatypes of column as per required.



API_2000-01-01_2021-12-31.csv

Select Input File Type

JSON | CSV

Options

Select delimiter COMMA

☑ Ignore empty strings

☐ Stop on errors

Specify Fields and Types

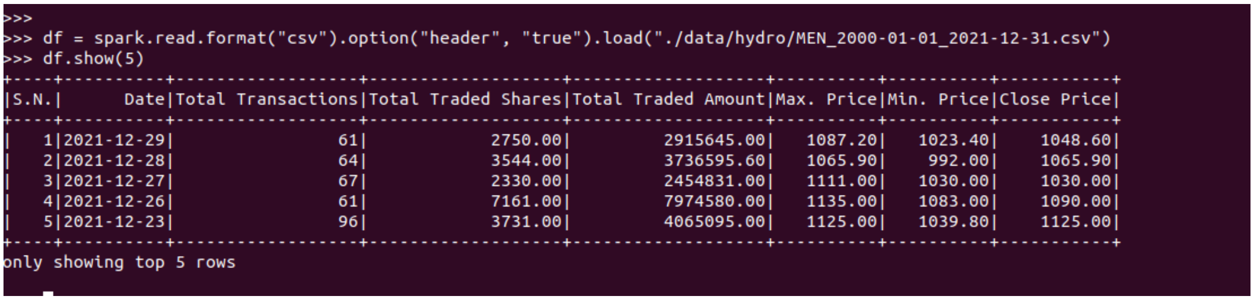
	☐ S.N. String	☑ Date Date	☑ Total Transactions Int32	☑ Total Traded Shares Int32	☑ Total Trade Amount Number
1	1	12/29/2021	1004	209153	92183343.8
2	2	12/28/2021	928	175848	75912542.5
3	3	12/27/2021	1073	261231	114786632.2
4	4	12/26/2021	1016	208918	92832767.5
5	5	12/23/2021	1610	287019	128545771.3
6	6	12/22/2021	1461	270703	117592202.1
7	7	12/21/2021	597	100032	41124594.8
8	8	12/20/2021	697	133791	53868318
9	9	12/16/2021	958	167764	68733566.6
10	10	12/15/2021	1921	324500	138894190.8

Import completed 1,393 / 1,393

DONE

Figure 8 Reading API csv data in mongoDB

Reading in pyspark

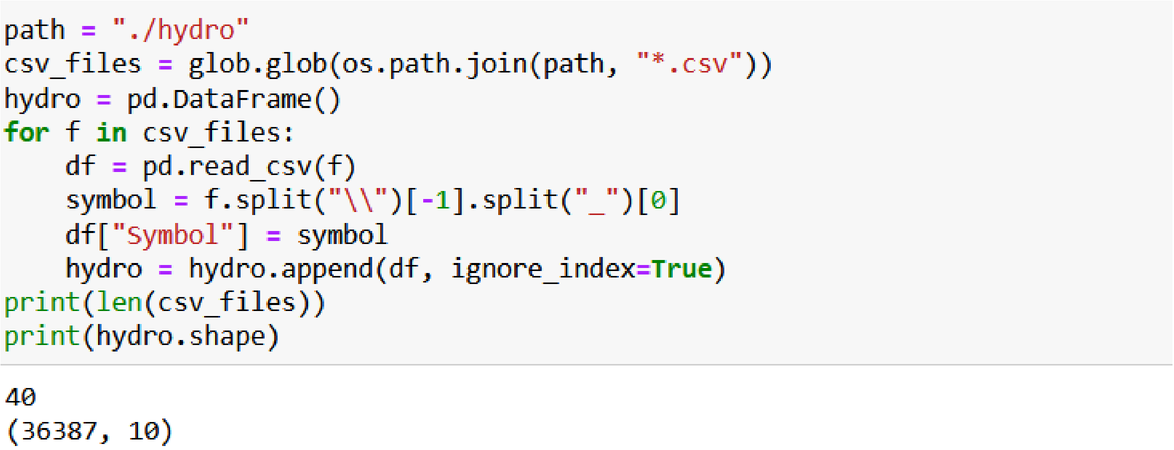


```
>>>
>>> df = spark.read.format("csv").option("header", "true").load("./data/hydro/MEN_2000-01-01_2021-12-31.csv")
>>> df.show(5)
+----+----------+------------------+-------------------+-------------------+----------+----------+-----------+
|S.N.|      Date|Total Transactions|Total Traded Shares|Total Traded Amount|Max. Price|Min. Price|Close Price|
+----+----------+------------------+-------------------+-------------------+----------+----------+-----------+
|   1|2021-12-29|                61|            2750.00|         2915645.00|   1087.20|   1023.40|    1048.60|
|   2|2021-12-28|                64|            3544.00|         3736595.60|   1065.90|    992.00|    1065.90|
|   3|2021-12-27|                67|            2330.00|         2454831.00|   1111.00|   1030.00|    1030.00|
|   4|2021-12-26|                61|            7161.00|         7974580.00|   1135.00|   1083.00|    1090.00|
|   5|2021-12-23|                96|            3731.00|         4065095.00|   1125.00|   1039.80|    1125.00|
+----+----------+------------------+-------------------+-------------------+----------+----------+-----------+
only showing top 5 rows
```

Figure 9 Reading hydro company MEN data in pyspark

Reading data in jupyter notebook

I used three sub-indices for this work: Hydropower, Commercial Banks, and Microfinance. In each directory it has its related listed company's data from 1st January 2000 to 31st December 2021. Each data frame holds its belonging data by adding a new symbol column to identify it properly.



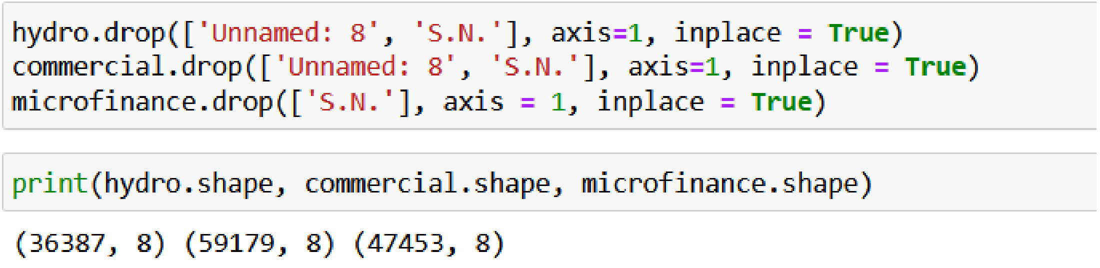
```
path = "./hydro"
csv_files = glob.glob(os.path.join(path, "*.csv"))
hydro = pd.DataFrame()
for f in csv_files:
    df = pd.read_csv(f)
    symbol = f.split("\\")[-1].split("_")[0]
    df["Symbol"] = symbol
    hydro = hydro.append(df, ignore_index=True)
print(len(csv_files))
print(hydro.shape)
```

```
40
(36387, 10)
```

Figure 10 Reading all hydro company data in a single data frame in jupyter notebook

4.1.2 Data Cleaning

Some file was having an extra raw column which has only a value or a character. It was not needed to have that column along with serial number column.



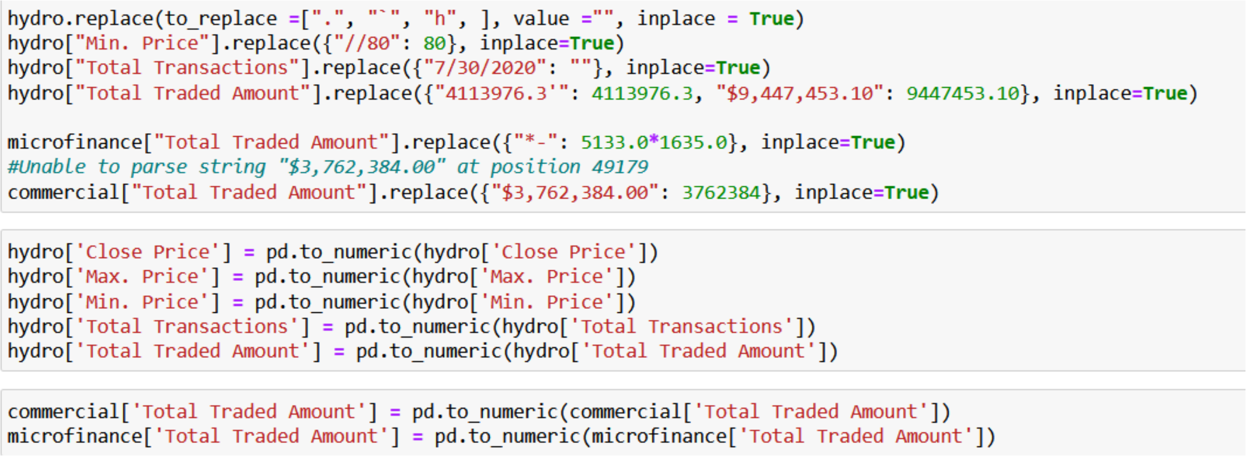
```
hydro.drop(['Unnamed: 8', 'S.N.'], axis=1, inplace = True)
commercial.drop(['Unnamed: 8', 'S.N.'], axis=1, inplace = True)
microfinance.drop(['S.N.'], axis = 1, inplace = True)
```

```
print(hydro.shape, commercial.shape, microfinance.shape)
```

```
(36387, 8) (59179, 8) (47453, 8)
```

Figure 11 Dropping unrequired columns from all sectors data

We should convert our column type to numeric before processing but not to integer when we have column like price or number where decimal value matters. While converting it to number, there was error because our column holds some characters which con not be converted into number. We manually replaced it with null value, but if it has some value within, we replaced it with that value.



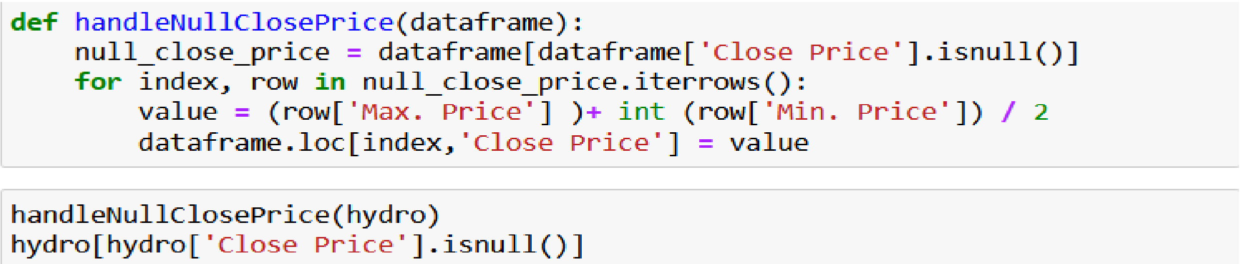
```
hydro.replace(to_replace =[".", "`", "h", ], value ="", inplace = True)
hydro["Min. Price"].replace({"//80": 80}, inplace=True)
hydro["Total Transactions"].replace({"7/30/2020": ""}, inplace=True)
hydro["Total Traded Amount"].replace({"4113976.3'": 4113976.3, "$9,447,453.10": 9447453.10}, inplace=True)

microfinance["Total Traded Amount"].replace({"*-": 5133.0*1635.0}, inplace=True)
#Unable to parse string "$3,762,384.00" at position 49179
commercial["Total Traded Amount"].replace({"$3,762,384.00": 3762384}, inplace=True)
```

```
hydro['Close Price'] = pd.to_numeric(hydro['Close Price'])
hydro['Max. Price'] = pd.to_numeric(hydro['Max. Price'])
hydro['Min. Price'] = pd.to_numeric(hydro['Min. Price'])
hydro['Total Transactions'] = pd.to_numeric(hydro['Total Transactions'])
hydro['Total Traded Amount'] = pd.to_numeric(hydro['Total Traded Amount'])
```

```
commercial['Total Traded Amount'] = pd.to_numeric(commercial['Total Traded Amount'])
microfinance['Total Traded Amount'] = pd.to_numeric(microfinance['Total Traded Amount'])
```

Figure 12 Replacing characters and converting every column to numeric

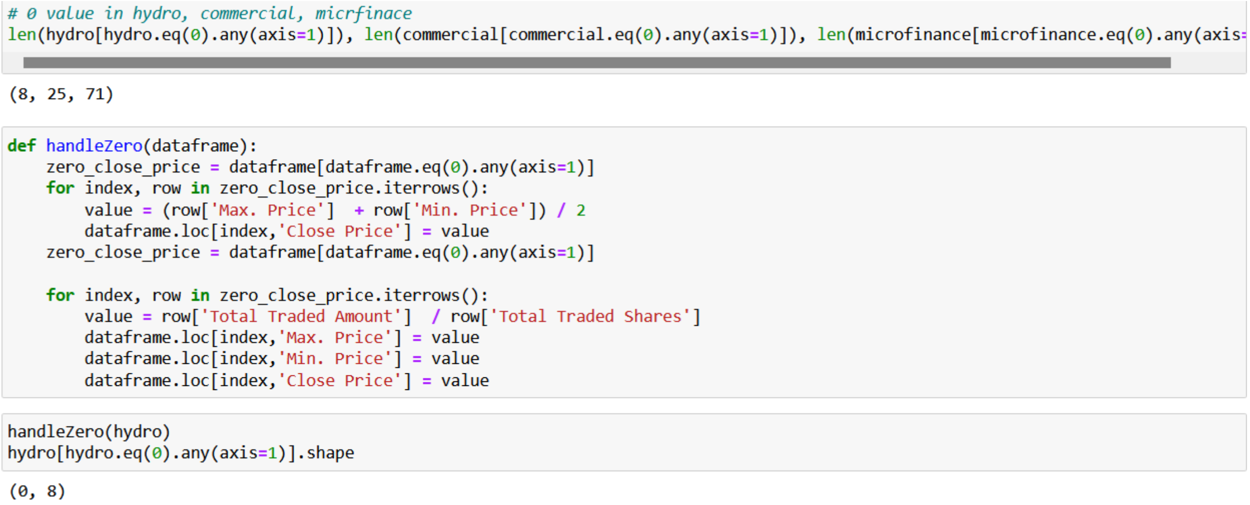
I handle null value and zero value not by any filling methods or by filling any statics values. But a dynamic approach like, if a row has null or zero value in close price column, then it must be between maximum and minimum value of that day. If we try to fit some mean, median or quartile value, it might go beyond max and min price of that day and I think such things are not realistic to happen in market. Additionally, maximum value should be more or equal than close value, totally trading amount should be multiple of total traded shares and average of close, max and min price.



```
def handleNullClosePrice(dataframe):
    null_close_price = dataframe[dataframe['Close Price'].isnull()]
    for index, row in null_close_price.iterrows():
        value = (row['Max. Price'] )+ int (row['Min. Price']) / 2
        dataframe.loc[index,'Close Price'] = value
```

```
handleNullClosePrice(hydro)
hydro[hydro['Close Price'].isnull()]
```

Figure 13 Replacing null values of close price column by average of Max and Min price of that day



```
# 0 value in hydro, commercial, micrfinace
len(hydro[hydro.eq(0).any(axis=1)]), len(commercial[commercial.eq(0).any(axis=1)]), len(microfinance[microfinance.eq(0).any(axis=
```

```
(8, 25, 71)
```

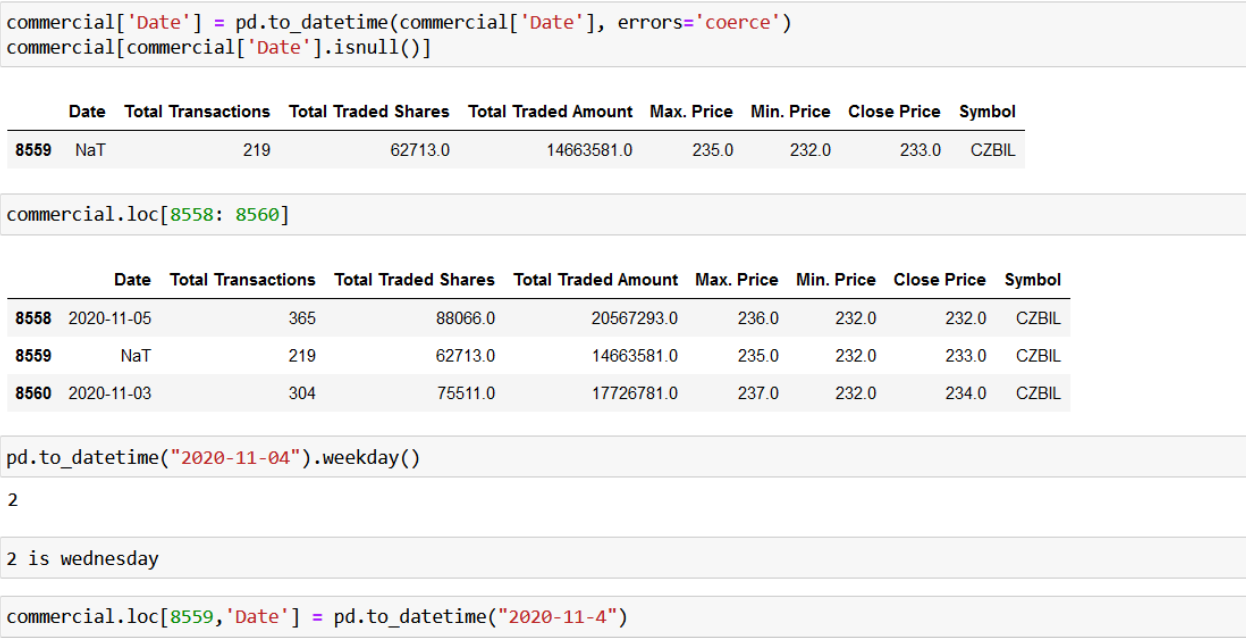
```
def handleZero(dataframe):
    zero_close_price = dataframe[dataframe.eq(0).any(axis=1)]
    for index, row in zero_close_price.iterrows():
        value = (row['Max. Price']  + row['Min. Price']) / 2
        dataframe.loc[index,'Close Price'] = value
    zero_close_price = dataframe[dataframe.eq(0).any(axis=1)]

    for index, row in zero_close_price.iterrows():
        value = row['Total Traded Amount']  / row['Total Traded Shares']
        dataframe.loc[index,'Max. Price'] = value
        dataframe.loc[index,'Min. Price'] = value
        dataframe.loc[index,'Close Price'] = value
```

```
handleZero(hydro)
hydro[hydro.eq(0).any(axis=1)].shape
```

```
(0, 8)
```

Figure 14 Handling zero values of every column

For handling missing date, I sort date column in an order. Printing the before and after two values by excessing the index of sorted data frame. If you short a data and you found something like 2,3,x,5,6 then you can easily say the missing value is 4. Every data is imported in market analysis because a single moment can have a lot of information about the trend.



```
commercial['Date'] = pd.to_datetime(commercial['Date'], errors='coerce')
commercial[commercial['Date'].isnull()]
```

	Date	Total Transactions	Total Traded Shares	Total Traded Amount	Max. Price	Min. Price	Close Price	Symbol
8559	NaT	219	62713.0	14663581.0	235.0	232.0	233.0	CZBIL

```
commercial.loc[8558: 8560]
```

	Date	Total Transactions	Total Traded Shares	Total Traded Amount	Max. Price	Min. Price	Close Price	Symbol
8558	2020-11-05	365	88066.0	20567293.0	236.0	232.0	232.0	CZBIL
8559	NaT	219	62713.0	14663581.0	235.0	232.0	233.0	CZBIL
8560	2020-11-03	304	75511.0	17726781.0	237.0	232.0	234.0	CZBIL

```
pd.to_datetime("2020-11-04").weekday()
```

```
2
```

```
2 is wednesday
```

```
commercial.loc[8559,'Date'] = pd.to_datetime("2020-11-4")
```

Figure 15 Handling missing date from data column

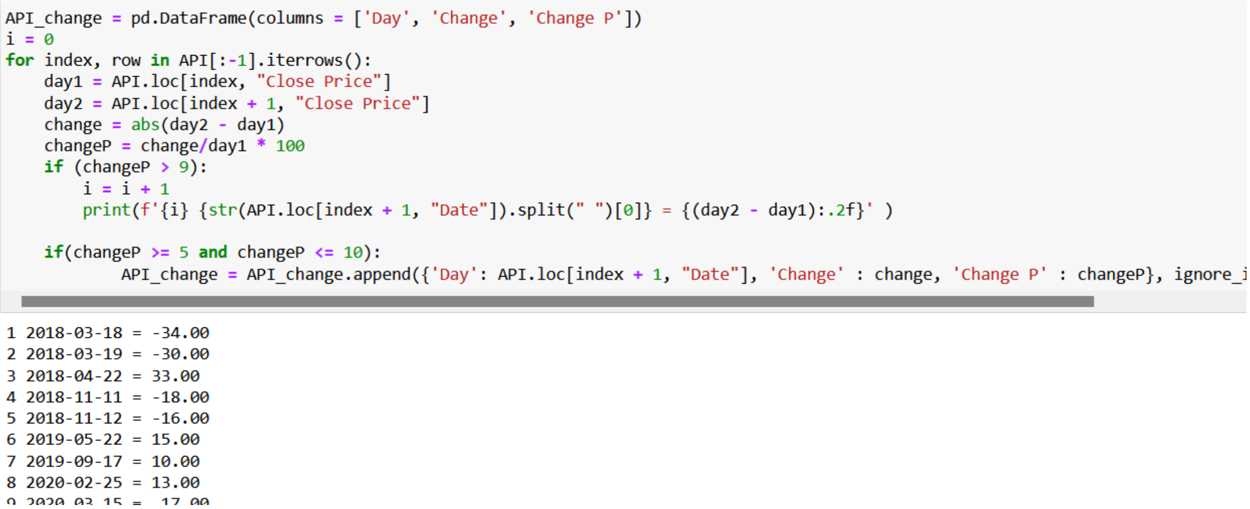
4.1.3. Data Analysis

Data analysis in jupyter notebook

A built-in python function describe can give a basic information about the distribution of our data. Info function prints column name, count values, and column data type with data frame shape. But it does not give complete information, we need to extract.

Let's extract some major changes happened on a hydropower company API.

There are 47 major changes(> 9%) from 2015 November 22 to December 2021 and as per NEPSE trading rules, A company can only have change percentage below 10 (भट्टराई, 2018). Some of the change has occurred in continuous date that means the news, actions, incidents before that have a great impact on price change.



```
API_change = pd.DataFrame(columns = ['Day', 'Change', 'Change P'])
i = 0
for index, row in API[:-1].iterrows():
    day1 = API.loc[index, "Close Price"]
    day2 = API.loc[index + 1, "Close Price"]
    change = abs(day2 - day1)
    changeP = change/day1 * 100
    if (changeP > 9):
        i = i + 1
        print(f'{i} {str(API.loc[index + 1, "Date"]).split(" ")[0]} = {(day2 - day1):.2f}' )

    if(changeP >= 5 and changeP <= 10):
            API_change = API_change.append({'Day': API.loc[index + 1, "Date"], 'Change' : change, 'Change P' : changeP}, ignore_i
```

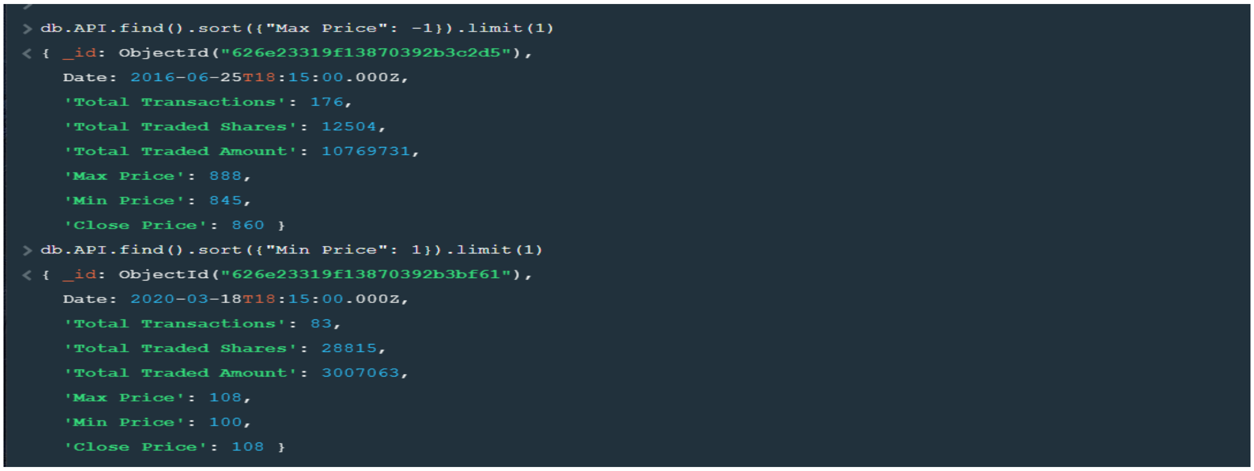
```
1 2018-03-18 = -34.00
2 2018-03-19 = -30.00
3 2018-04-22 = 33.00
4 2018-11-11 = -18.00
5 2018-11-12 = -16.00
6 2019-05-22 = 15.00
7 2019-09-17 = 10.00
8 2020-02-25 = 13.00
9 2020-03-15 = -17.00
```

Figure 16 Major changes above 9 percentage in API hydro company

Data analysis in mongoDB

I write some queries in mongodb cli for extracting information from data.

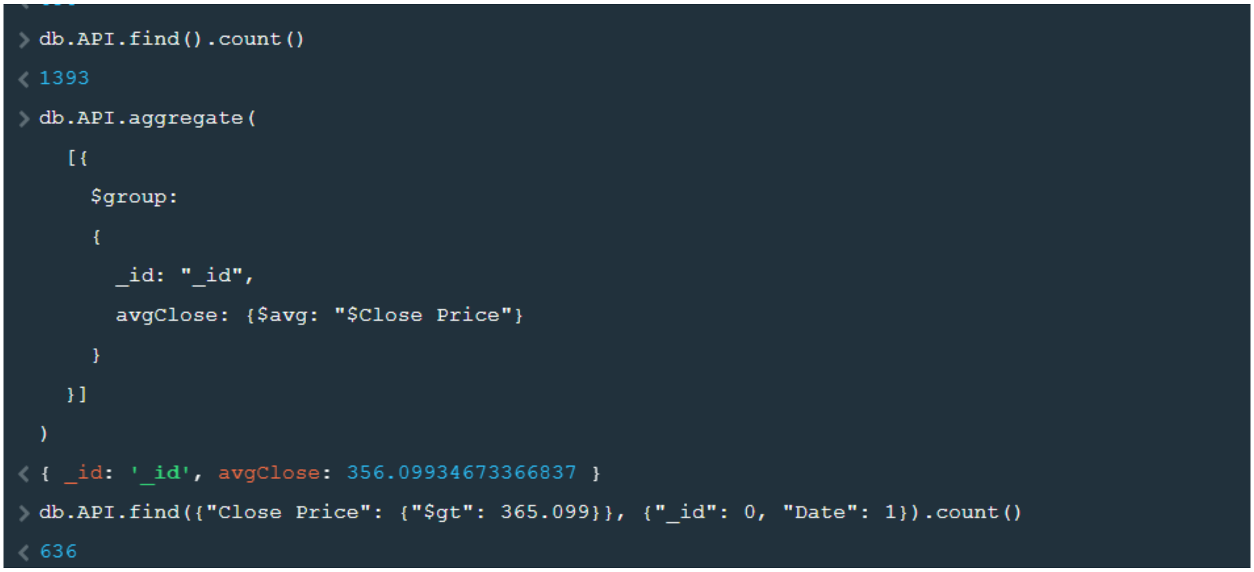
Finding a collection with maximum price from max price field and a collection with minimum price from min price field from API.



```
> db.API.find().sort({"Max Price": -1}).limit(1)
< { _id: ObjectId("626e23319f13870392b3c2d5"),
    Date: 2016-06-25T18:15:00.000Z,
    'Total Transactions': 176,
    'Total Traded Shares': 12504,
    'Total Traded Amount': 10769731,
    'Max Price': 888,
    'Min Price': 845,
    'Close Price': 860 }
> db.API.find().sort({"Min Price": 1}).limit(1)
< { _id: ObjectId("626e23319f13870392b3bf61"),
    Date: 2020-03-18T18:15:00.000Z,
    'Total Transactions': 83,
    'Total Traded Shares': 28815,
    'Total Traded Amount': 3007063,
    'Max Price': 108,
    'Min Price': 100,
    'Close Price': 108 }
```

Figure 17 All-time maximum and minimum price of API hydro company

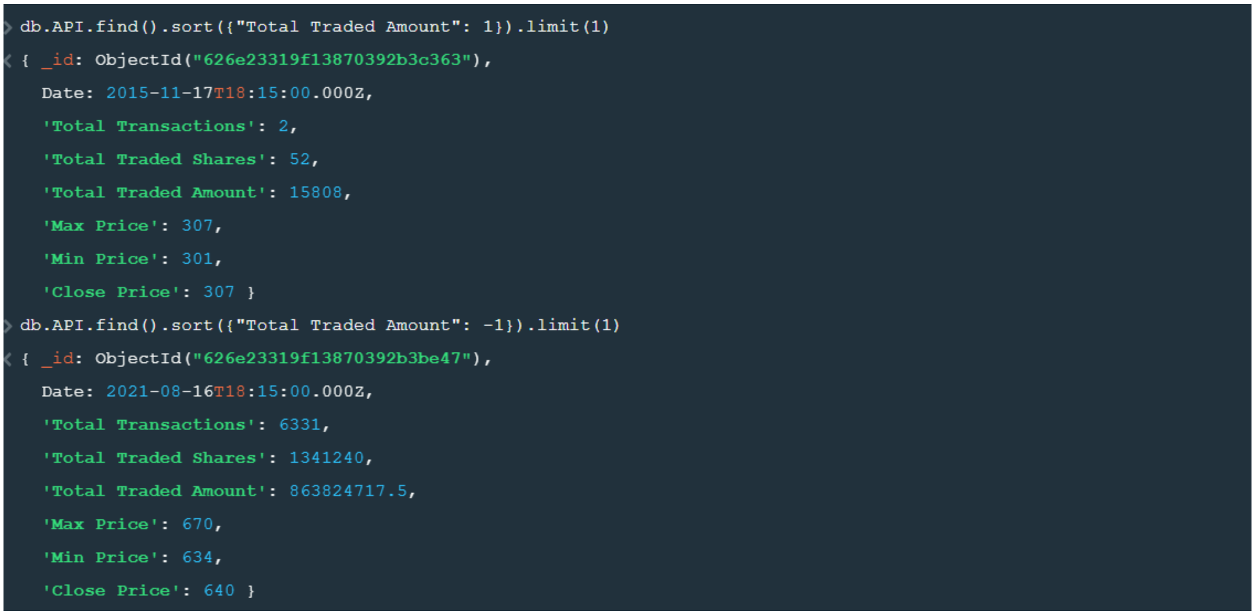
Finding average close price of all time and counting the traded day which have close price more than average close price.



```
> db.API.find().count()
< 1393
> db.API.aggregate(
    [{
      $group:
      {
        _id: "_id",
        avgClose: {$avg: "$Close Price"}
      }
    }]
  )
< { _id: '_id', avgClose: 356.09934673366837 }
> db.API.find({"Close Price": {"$gt": 365.099}}, {"_id": 0, "Date": 1}).count()
< 636
```

Figure 18 Days which have close price greater than average close price.

Finding a collection with maximum traded amount and minimum traded amount from total traded amount field.

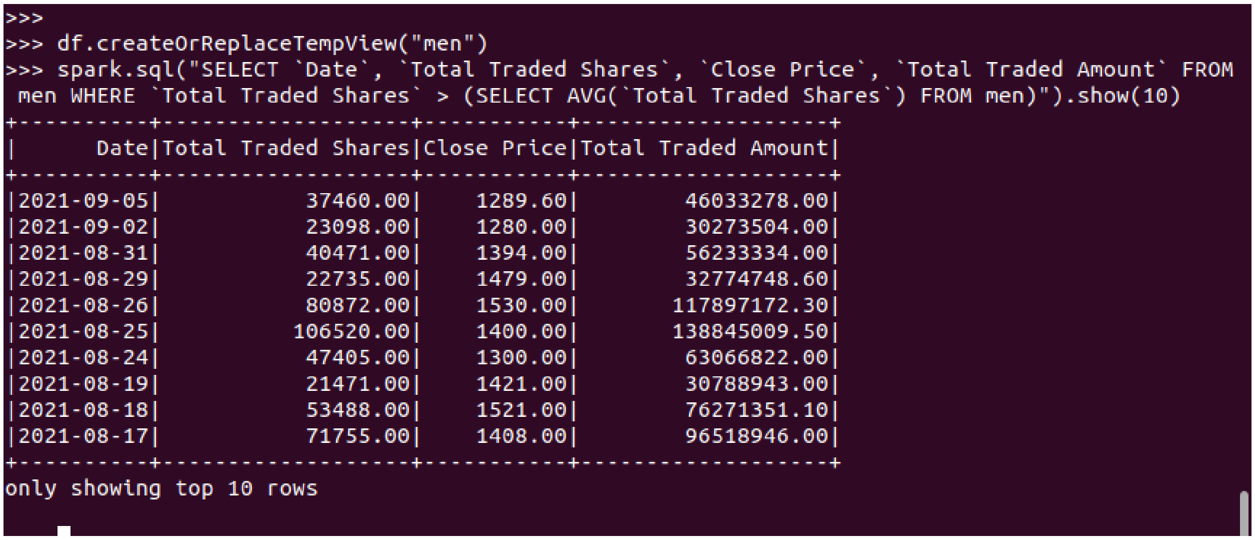


```
> db.API.find().sort({"Total Traded Amount": 1}).limit(1)
< { _id: ObjectId("626e23319f13870392b3c363"),
    Date: 2015-11-17T18:15:00.000Z,
    'Total Transactions': 2,
    'Total Traded Shares': 52,
    'Total Traded Amount': 15808,
    'Max Price': 307,
    'Min Price': 301,
    'Close Price': 307 }
> db.API.find().sort({"Total Traded Amount": -1}).limit(1)
< { _id: ObjectId("626e23319f13870392b3be47"),
    Date: 2021-08-16T18:15:00.000Z,
    'Total Transactions': 6331,
    'Total Traded Shares': 1341240,
    'Total Traded Amount': 863824717.5,
    'Max Price': 670,
    'Min Price': 634,
    'Close Price': 640 }
```

Figure 19 All-time maximum and minimum traded amount

Data analysis in pyspark (SQL)

Finding trading days which have number of traded shares more than average traded shares of all time.



```
>>>
>>> df.createOrReplaceTempView("men")
>>> spark.sql("SELECT `Date`, `Total Traded Shares`, `Close Price`, `Total Traded Amount` FROM
 men WHERE `Total Traded Shares` > (SELECT AVG(`Total Traded Shares`) FROM men)").show(10)
+----------+-------------------+-----------+-------------------+
|      Date|Total Traded Shares|Close Price|Total Traded Amount|
+----------+-------------------+-----------+-------------------+
|2021-09-05|           37460.00|    1289.60|        46033278.00|
|2021-09-02|           23098.00|    1280.00|        30273504.00|
|2021-08-31|           40471.00|    1394.00|        56233334.00|
|2021-08-29|           22735.00|    1479.00|        32774748.60|
|2021-08-26|           80872.00|    1530.00|       117897172.30|
|2021-08-25|          106520.00|    1400.00|       138845009.50|
|2021-08-24|           47405.00|    1300.00|        63066822.00|
|2021-08-19|           21471.00|    1421.00|        30788943.00|
|2021-08-18|           53488.00|    1521.00|        76271351.10|
|2021-08-17|           71755.00|    1408.00|        96518946.00|
+----------+-------------------+-----------+-------------------+
only showing top 10 rows
```

Figure 20 Displaying recent 10 data which has traded shares greater than average traded shares

4.1.4 Data Visualization

I take 3 samples from each sub-indices to visualize the patterns inside the sector and outside the sector. Hydro companies and banking sectors follows some similarity among themselves. But in case of microfinance there might be some slight difference in patterns.

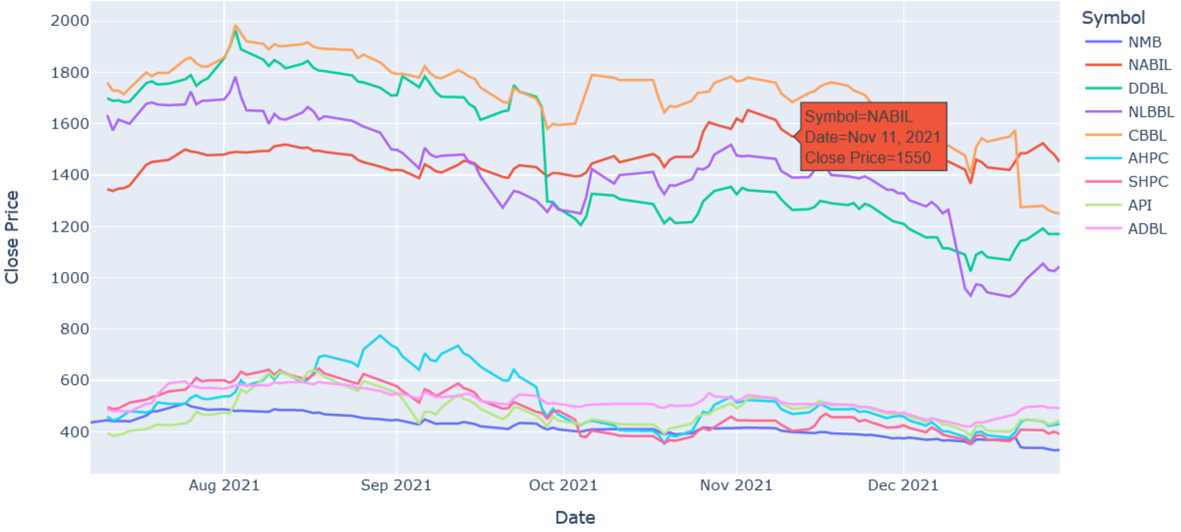


Figure 21 Visualization of 3-3 hydro, commercial and microfinance companies

Micro finances are the leading sectors in case of its price. In the last 5 months of transactions, microfinance is facing bearish trends facing down trends from an average of 1600 to 1300.

On an average closing price of each sector, microfinance is having more up and down trends. At the end of 2019 the average close price fall less than 1000 from more than 2000. Hydro and microfinance seem to have some common trends in the last few years. In case of banking sectors, there seem not more changes in from those years. But some companies has faced up and down across the line.

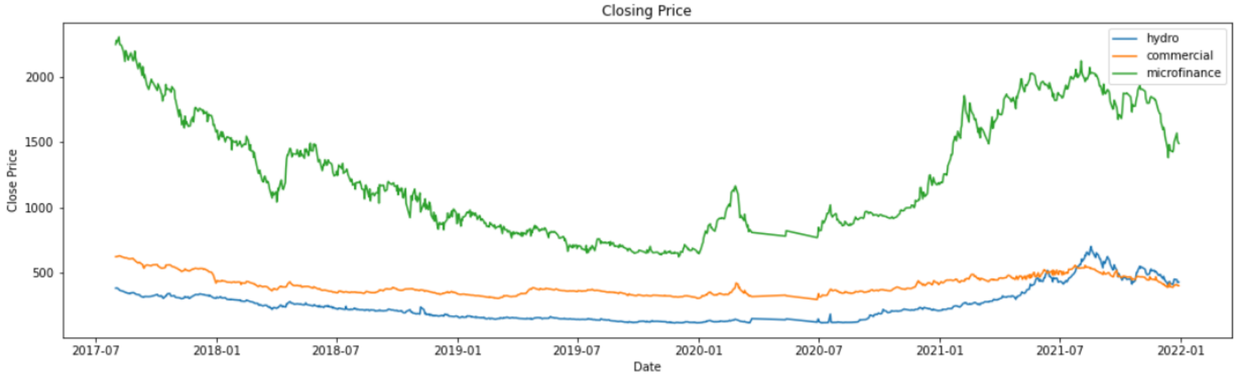


Figure 22 Multiple line graph of closing price of each sector in an average of overall companies from the sector

Even after having higher share values, microfinances are back in case of total traded amount. Commercial banks are more active and creates high flow of cash in market. In Year 2021, the transactions and flow of money reached more than 250 million just in a day. Turnover of commercial banks were 100 times higher in 2021 than early years.

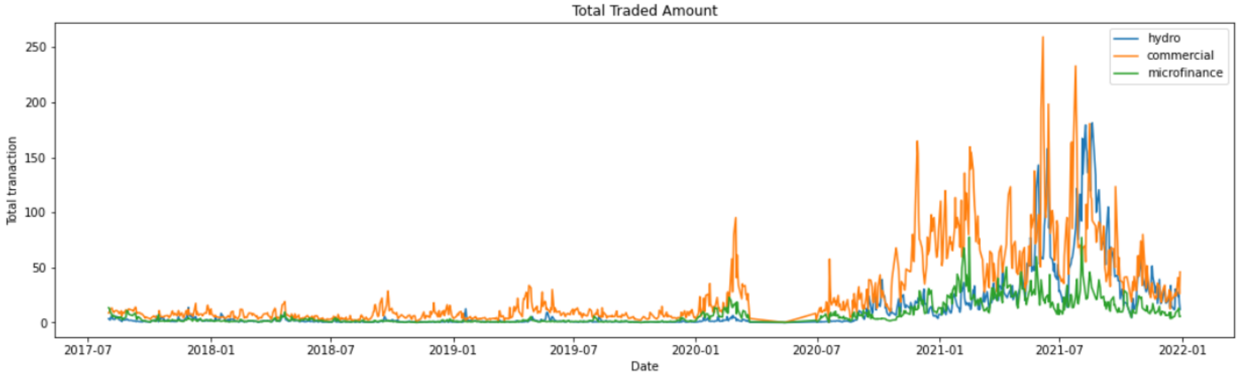


Figure 23 Multiple line graph of total traded amount of each sector in an average of overall companies from the sector

The correlation between max price, min price and close price is 1. It means they are fully dependent on other. If one increases than another will also increase. Total transactions and total traded shares have more relation with max price than min and close price i.e., traders want to buy or sell shares when the price is maximum.

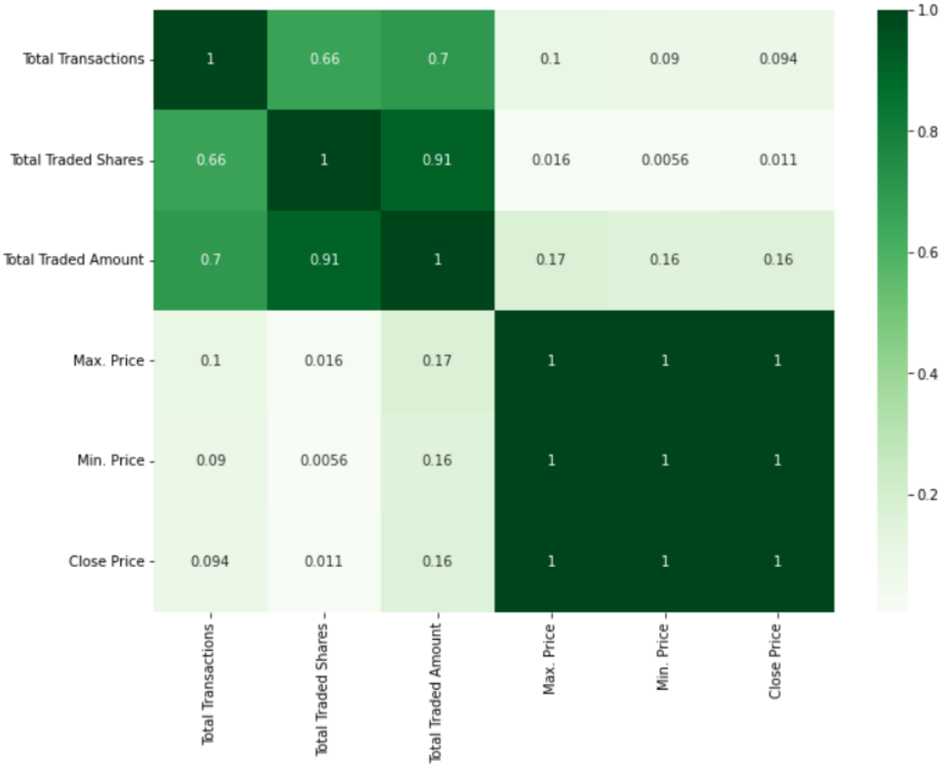
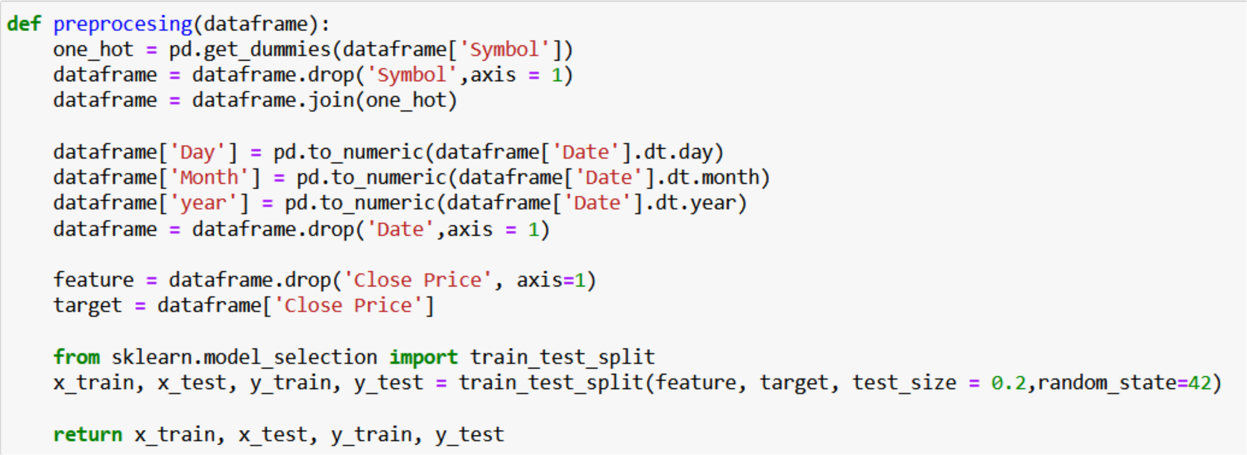


Figure 24 Heatmap plot of microfinance sector

4.1.5 Choosing the Best Model

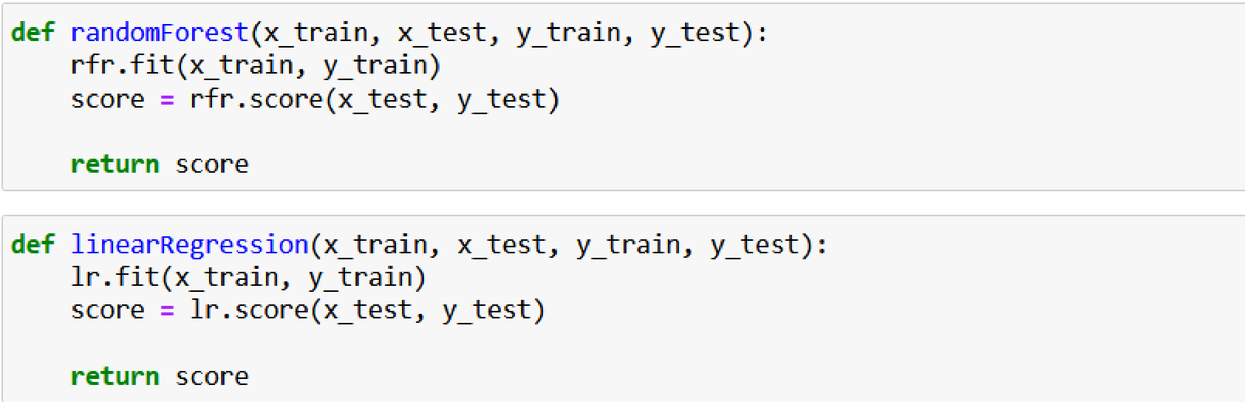
We will select close price as our target column. All other columns will be our feature column. But before we fit our data to model, we have to perform some preprocessing. Machine learning algorithms does not take string and objects. We will do grouping for textual data by using Boolean method to categories the records. Using dummies method create equal number of columns to unique value. A particular row can only have one string value. For that string column it will create value 1 and remaining will be 0. We can also use one hot encoder for converting string data into numerical form. We will add three more columns for handling date. Each column will have day, month and year.



```
def preprocesing(dataframe):
    one_hot = pd.get_dummies(dataframe['Symbol'])
    dataframe = dataframe.drop('Symbol',axis = 1)
    dataframe = dataframe.join(one_hot)

    dataframe['Day'] = pd.to_numeric(dataframe['Date'].dt.day)
    dataframe['Month'] = pd.to_numeric(dataframe['Date'].dt.month)
    dataframe['year'] = pd.to_numeric(dataframe['Date'].dt.year)
    dataframe = dataframe.drop('Date',axis = 1)

    feature = dataframe.drop('Close Price', axis=1)
    target = dataframe['Close Price']

    from sklearn.model_selection import train_test_split
    x_train, x_test, y_train, y_test = train_test_split(feature, target, test_size = 0.2,random_state=42)

    return x_train, x_test, y_train, y_test
```

Figure 25 Preprocessing before data is fit into model

I will use two different model: Random Forest Regressor and Linear Regression from sklearn library for comparing the performance of our dataset.



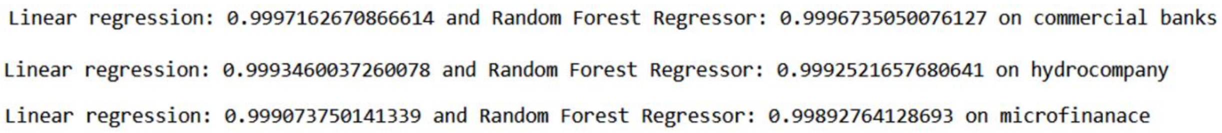
```
def randomForest(x_train, x_test, y_train, y_test):
    rfr.fit(x_train, y_train)
    score = rfr.score(x_test, y_test)

    return score
```

```
def linearRegression(x_train, x_test, y_train, y_test):
    lr.fit(x_train, y_train)
    score = lr.score(x_test, y_test)

    return score
```

Figure 26 Fitting data in two model and calculating score

4.2 Discussion of the finding

The study's main finding is that stock market growth has no positive influence on the country's economy. Small number of listed businesses, low market capitalization ratio, low turnover ratio, low value traded ratio, high volatility, high concentration and dangerous market characterize the Nepalese stock market.



```
Linear regression: 0.9997162670866614 and Random Forest Regressor: 0.9996735050076127 on commercial banks

Linear regression: 0.9993460037260078 and Random Forest Regressor: 0.9992521657680641 on hydrocompany

Linear regression: 0.999073750141339 and Random Forest Regressor: 0.99892764128693 on microfinanace
```

Figure 27 Accuracy of two model over all sectors

When we fit our past data in different model, we get 99 percentage of accuracy for every sector.

Note: What has been discovered from the data can be more found in 4.1.4 Data Visualization section with the visualization.

4.3 Analysis of the findings

The discovered models might be used by investors, financial experts, and regulators to forecast the daily NEPSE index and make changes in policy based on it. This report has given strong signal to investors and other stakeholders that the daily NEPSE index will be impacted by its prior values as well as unpredictable mistakes in the past. This indicates that both observable and random influences in the past will have an impact on the NEPSE index's potential price. Another consequence is that the effect of observable and random components is reflected every five days (since the Nepal stock market operates 5 days a week) and this cycle continues until the second week. The most important aspect of this study is the success of daily stock trading. Three components under NEPSE; Automatic result, moving average and seasonal with identified and related predictors; Should be considered. However, the price of an individual company may not be tracked properly. The process of following the NEPSE code.

The main takeaway from this research is that in order to remain effective in daily trading of companies listed on NEPSE, all three aspects must be considered: Autoregressive, Moving Average, and Seasonal Effect with the indicated predictions and corresponding sign. Individual business prices, on the other hand, may not follow the same procedure as the NEPSE index.

5. Conclusion

In conclusion, the stock exchange enhances economic growth through increased investment and productivity. The stock exchange must operate at its own pace and without external interference, and if it crashes, it will have a significant impact on many other areas of the economy, trapping the economy in a negative cycle. Investment in the stock exchange has grown as a result of the buildup of investable cash in banks and financial institutions, as well as reduced interest rates and decreased demand from the corporate sector. Nepal stock market is quite unpredictable because of its political instability. Besides that, we can do some technical analysis over the past data which helps when the situations get worst and better.

6. References

Agarwal, N., Parmar, I., & Saxena, S. (2018). Stock Market Prediction Using Machine Learning. *IEEE* (p. 9). Jalandhar, India: IEEE.

G.C., S. B. (2008). Volatility Analysis of Nepalese Stock Market. *The Journal of Nepalese Business Studies*, 76-84.

Lee, H. S. (2020). *Exploring the Initial Impact of COVID-19 Sentiment on US Stock Market Using Big Data.* Seoul, Korea: sustainability.

Mainali, P. K. (2011). Problems and prospects of stock market in Nepal. *SEBON Journal-V*, 35-38.

Nepal Stock Exchange Ltd. (2022, April 20). Retrieved from Nepal Stock Exchange Ltd: https://www.nepalstock.com.np/

Pahwa, K., & Agarwal, N. (2019). Stock Market Analysis using Supervised Machine Learning. *International Conference on Machine Learning, Big Data, Cloud and Parallel Computing*, 197-200. doi:10.1109/COMITCon.2019.8862225

Singh, S. M. (2021, December 23). *Nepal's stock market and economic growth: Politicians' sayings and impact* . Retrieved from The Himalayan Times: https://thehimalayantimes.com/opinion/nepals-stock-market-and-economic-growthpoliticians-sayings-and-impact

भट्टराई, र. (2018). *नेपालको सेयर बजार.* Kathmandu: Securities Research Center and Services Pvt. Ltd.