# Indian Equity Short Term Trading Model

**Author Name(s) and Affiliation(s)**

|  |  |
| --- | --- |
| **Name** | **Email** |
| Aditya Prasann | aditya.prasann@gmail.com |
| Raghvendra Goyal | raghvendragoyal@gmail.com |
| Nilbabul Patel | nilbabul.patel@gmail.com |

**Master of Science in Financial Engineering**

Submission: August 2020

Supervisor: Greg Ciresi

# *Abstract*

The **Indian Equity Market** is a mature market with most of regulation and practices in place. The leading exchange in India is NSE and NIFTY50. A selection of NIFTY50 stocks has been taken as our universe to ensure sufficient liquidity in order to build a practical short-term trading model. The aim of the paper is to investigate different aspects of building **short time trading models** like **data collection**, **framework for back testing**, **strategy selection** and **capital allocation**. The final goal is to present a **trading portfolio** which will be combination of 5 to 10 independent and low correlated strategies with high sharpe ration and least drawdown.

**Background and Motivation:**

Prediction of future stock prices is a widely intriguing subject in many fields like computer science, trading, finance, statistics to name a fews. Chief motivator for which is to predict the direction of future prices such that profit can be made from purchase and sale of equities. Traders typically use fundamental and technical analysis to analyse equities to arrive at trading Strategies. Fundamental analysis is the traditional approach involves a study of company fundamentals such as revenues and expenses, market position, annual growth rates, and so on.

Technical analysis, is data based on the study of historical price fluctuations. Practitioners of technical analysis study price charts for price patterns and use price data in different calculations to forecast future price movements. There is a correlation between price and company which should be selected and data helps in determine time to enter and exit the market.

For finance and computer science, most traditional models of stock price prediction use statistical models and neural network models derived from price data. Trending strategy in computer science seems to be using evolutionary algorithms or machine learning neural networks.

**Impetus and Scope:**

The attempt is to present a **trading portfolio** which will be combination of bunch of independent and uncorrelated strategies with high sharpe ration and least drawdown. This model could be a decision support tool to a trader or can be extended for robot trader with an interface to stock exchanges. The first step is to procure historical data, filter it, have a chronology on top of it, adjustment to events like corporate actions and resample it. Next is to have a robust backtesting strategy having rich multi stock and multi processing capability. We have after considering several open source packages arrived at back trader.

**Methodology:**

In order to achieve the stated objective of practical track of building short term trading models following methodology has been followed -

Data collection: The first step to most financial analysis is to have data to analyse. We came up with a process to build time/volume/tick/dollar bars from raw data. A comparison was done on the data parameters to come up with best suited data format for short term trading models.

Logic and Strategy: We identified short term trends and patterns through combination of technical indicators and candlestick patterns and test it on training data. The next step was to build a back testing framework. It involved feeding the data collected as the input to arrive at the trade sheet as a result.

Validation of testing strategies and statistics: We checked the equity curve, drawdowns, sharpe ratio, SQN and other performance parameters if a strategy is worth trading. A comparison of different input was done by running multiple simulation.Regime modelling to distinguish mean reverting and trending market, which will help in allocating capital to different strategies. Portfolio of strategies will be tested on out of sample data to check robustness of trading model.

**Data Preparation:**

We have procured data from vendor for NSE stock. NSE is the leading stock exchange operating in India. We have in scope 49 stocks and data scope is from 1st August 2017 to 30th April 2020. Although our focus in this document is on stocks listed on the National Stock Exchange, the developed model can just as easily be used for any other stock exchange where a sufficient amount of daily historical prices are available.

1. The data of NSE stock for the period 1st August 2017 to 30th April 2020 was procured from data vendor.
2. Data filtering: Each stock has futures available for current month, next month and next to next to next month, for example on a particular day of July there will be three futures contracts available for trading as follows:
3. July future expiring on last Thursday of July,
4. August future expiring on last Thursday of August,
5. September future expiring on last Thursday of September,

The data provided was filtered to include only current month expiry (next 2 month contracts were removed), so that, a continuous data series can be formed.

1. The data was adjusted for corporate actions.
2. Then each stock data is resampled to 15 min, to form 15 min OHLC candles.

**Moving Average**

Moving average is a simple, technical analysis tool. Moving average statistically is a calculation used to analyse data points by creating a series of averages of different subsets of the full data set. The impacts of random, short-term fluctuations on the price of a stock over a specified time-frame are mitigated by calculating the moving average. Moving averages are usually calculated to identify the trend direction of a stock or to determine its support and resistance levels. It is a trend-following—or lagging—indicator because it is based on past prices. The longer the time period for the moving average, the greater the lag. A 100-day moving average will have a much greater degree of lag than a 10-day moving average because it contains prices for the past 100 days. Moving averages are a totally customisable indicator, which means that a trader can freely choose whatever time frame they want when calculating an average. The shorter the time span used to create the average, the more sensitive it will be to price changes. The longer the time span, the less sensitive the average will be.

**Types of Moving Averages**

**Simple Moving Average**

The simplest form of a moving average, known as a simple moving average (SMA), is calculated by taking the arithmetic mean of a given set of values. In other words, a set of numbers–or prices in the case of financial instruments–are added together and then divided by the number of prices in the set. The formula for calculating the simple moving average of a security is as follows:

​

﻿

**Exponential Moving Average (EMA)**

The exponential moving average is a type of moving average that gives more weight to recent prices in an attempt to make it more responsive to new information. To calculate an EMA, you must first compute the simple moving average (SMA) over a particular time period. We must calculate the multiplier next for weighting the EMA (referred to as the "smoothing factor"), which typically follows the formula: [2 ÷ (selected time period + 1)]. So, for a 20-day moving average, the multiplier would be [2/(20+1)]= 0.0952. Then you use the smoothing factor combined with the previous EMA to arrive at the current value. The EMA thus gives a higher weighting to recent prices, while the SMA assigns equal weighting to all values.

**Strategy**

We call a plan that includes trading signals for prompting trades, a rule for deciding how much of the portfolio to risk on any particular strategy, and a complete exit strategy for any trade an overall trading strategy. Our concern now is to design and evaluate trading strategies.

We suppose that the amount of money in the portfolio involved in any particular trade is a fixed proportion; 10% seems like a good number. We will also say that for any trade, if losses exceed 20% of the value of the trade, we will exit the position. Now we need a means for deciding when to enter position and when to exit for a profit.

We plan to use a moving average crossover strategy. We will use two moving averages, one we consider “fast”, and the other “slow”.

The strategy is:

* Trade the asset when the fast moving average crosses over the slow moving average.
* Exit the trade when the fast moving average crosses over the slow moving average again.
* A long trade will be prompted when the fast moving average crosses from below to above the slow moving average, and the trade will be exited when the fast moving average crosses below the slow moving average later.
* A short trade will be prompted when the fast moving average crosses below the slow moving average, and the trade will be exited when the fast moving average later crosses above the slow moving average.

**Event Based back testing framework:**

Backtesting is arguably the most critical part of the Systematic Trading Strategy (STS) production process, sitting between strategy development and deployment (live trading). If a strategy is flawed, rigorous backtesting will hopefully expose this, preventing a loss-making strategy from being deployed. Backtesting uses historic data to quantify STS performance.Simulated/live trading deploys a tested STS in real time: signalling trades, generating orders, routing orders to brokers, then maintaining positions as orders are executed.

Event driven trading systems are widely used in algorithmic trading. They can simulate historical data or to mimic live trading environment. It allows back-testing of strategies in a manner that is extremely similar to live execution. It can run for a specific period after an initial setup and executes it's trading logic when specific event occurs. These events can be driven by live data feed or historical data. It can connect back-testing to historical market data and simulated broker whereas live trading strategy to real-time market data feed and broker. There are many open source event driven trading systems are available like ZipLine, PyAlgoTrade, BackTrader. After exploring we have chosen back-trader as our preferred choice for backtesting. Back-trader is functionally very rich, can trade multi-stocks and has inbuilt multiprocessing.

Most frameworks go beyond backtesting to include some live trading capabilities.

**The Components of a Backtesting Framework**

Data and STS acquisition: The acquisition components consume the STS script/definition file and provide the requisite data for testing. If the framework requires any STS to be recoded before backtesting, then the framework should support canned functions for the most popular technical indicators to speed STS testing. Users determine how long of a historical period to backtest based on what the framework provides, or what they are capable of importing.

Performance testing applies the STS logic to the requested historic data window and calculates a broad range of risk & performance metrics, including max drawdown, Sharpe & Sort ratios. Most all of the frameworks support a decent number of visualisation capabilities, including equity curves and deciles-statistics.

Optimisation tends to require the lion’s share of computing resources in the STS process. If your STS require optimisation, then focus on a framework that supports scalable distributed/parallel processing.

In the context of strategies developed using technical indicators, system developers attempt to find an optimal set off parameters for each indicator. Most simply, optimisation might find that a 6 and 10 day moving average crossover STS accumulated more profit over the historic test data than any other combination of time periods between 1 and 20. Already with this trivial example, 20 \* 20 = 400 parameter combinations must be calculated & ranked.

In a portfolio context, optimisation seeks to find the optimal weighting off every asset in the portfolio, including shorted and leveraged instruments. On a periodic basis, the portfolio is rebalanced, resulting in the purchase and sale of portfolio holdings as required to align with the optimised weights.

Position sizing is an additional use of optimisation, helping system developers simulate and analyse the impact of leverage and dynamic position sizing on STS and portfolio performance.

Standard capabilities of open source Python backtesting platforms seem to include:

* Event driven
* Very flexible, unrestrictive licensing
* Decent collection of pre-defined technical indicators
* Standard performance metric calculation/visualisation/reporting capabilities

### **Back trader**

This platform is exceptionally well documented, with an accompanying blog and an active on-line community for posting questions and feature requests. Back trader supports a number of data formats, including CSV files, Pandas DataFrames, blaze iterators and real time data feeds from three brokers. These data feeds can be accessed simultaneously, and can even represent different timeframes. Supported brokers include Oanda for FX trading and multi-asset class trading via Interactive Brokers and Visual Chart.

* Project Page: [www.backtrader.com](https://www.backtrader.com/)
* Github: [github.com/mementum/backtrader](https://github.com/mementum/backtrader)
* License: [GPL v3.0](https://github.com/mementum/backtrader/blob/master/LICENSE)

We have scripted a python code which will be utilised in testing trading idea (code attached in a separate file and in appendix) throughout the project.

Few notes for this code:

1. Initial capital value is set at 2.5 million for each strategy.
2. The logic is tested on all 49 stocks simultaneously; each trade will be taken on 2% of capital.

**Performance measurement:**

**Sharpe Ratio:**

Sharpe Ratio is a widely used risk-adjusted return measure and it takes risk-free return as its benchmark Chan[2]. Since, there is no risk on a risk-free asset, standard deviation of risk free return is zero.

Chan argues that Sharpe ratio facilitates performance comparison across different strategies. Sharpe ratio is usually low if a strategy trades only few times a year and has a deep or lengthy drawdown. Sharpe ratio has been considered as performance measure for this paper.

**Maximum Drawdown:**

Maximum drawdown focuses on capital preservation. It approximates worst-case scenario for a trading strategy for a period. It is the maximum loss observed from peak to trough, for a portfolio, before a new peak is attained. It helps to measure the risk incurred by trading strategy and helps to determine if its practical to take on such risk. Miner[3] argues capital preservation is key to long-term successful trading and an important criterion for most investors. Chan[2] states that it is a better measure of tail risks, as it does not rely on measuring just the second moment of returns distribution. It is also asymmetric i.e. not concerned about periods with extremely positive returns.

𝑀𝑎𝑥𝑖𝑚𝑢𝑚 𝐷𝑟𝑎𝑤𝑑𝑜𝑤𝑛 = 𝑇𝑟𝑜𝑢𝑔ℎ 𝑉𝑎𝑙𝑢𝑒 − 𝑃𝑒𝑎𝑘 𝑉𝑎𝑙𝑢𝑒 𝑃𝑒𝑎𝑘 𝑉𝑎𝑙𝑢𝑒

From trading strategy development point of view, it can be visualised during back-testing phase. As this research involves back-testing, Maximum Drawdown has been considered as one of the measures of performance for this research.

**System Quality Number(SQN):**

Dr. Van Tharp developed this Formula for Trading Systems. Later he found out that the SQN Indicator is a very powerful to measure the trendiness.

He applied the SQN formula to the daily percent price change of a stock or an index, it proved to be an excellent measure of the quality of a trend.

The large the N, the more trading opportunities you have.

The large the average P&L, the better you are obviously.

The smaller the Std dev (P&L), the more regular are your results and the smaller are the drawdowns.

Van Tharp describes the Quality of Stocks, Indices, Currencies and Commodities with this SQN Scores:

Score: 1.6 - 1.9 Below average, but trade-able

Score: 2.0 - 2.4 Average

Score: 2.5 - 2.9 Good

Score: 3.0 - 5.0 Excellent

Score: 5.1 - 6.9 Superb

Score: 7.0 - Keep this up, and you may have the Holy Grail.

What says the SQN Score: For Example a stock have an SQN Value of 2.8 that means the Performance over the last 100 Periods was 2.8 bigger than the Standard Deviation of the Stock.

**Trade Analyser:**

This is a custom analyser for backorder tool. From the documentation[5] following about trade analyser -

Provides statistics on closed trades (keeps also the count of open ones)

* Total Open/Closed Trades
* Streak Won/Lost Current/Longest
* ProfitAndLoss Total/Average
* Won/Lost Count/ Total PNL/ Average PNL / Max PNL
* Long/Short Count/ Total PNL / Average PNL / Max PNL
* Won/Lost Count/ Total PNL/ Average PNL / Max PNL
* Length (bars in the market)
* Total/Average/Max/Min
* Won/Lost Total/Average/Max/Min
* Long/Short Total/Average/Max/Min
* Won/Lost Total/Average/Max/Min

**Transactions:**

This is a custom analyser for backorder tool. From the documentation[6] following about trade analyser - This analyser reports the transactions occurred with each an every data in the system. It looks at the order execution bits to create a Position starting from 0 during each next cycle. The result is used during next to record the transactions.

**Simulation and Run:**

Assumptions / standard parameters for trading portfolio

We have taken 0.0001 as commission which roughly translates to 1000 per crore of transaction or 1000 per 10 million of transaction

Total no of stocks 49

We have set initial portfolio value of 2.5 million for each strategy

Each trade will be taken on 2% of portfolio value, if there is trade on

each and every stock at one point of time then it should suffice.

After several trial and error strategies we finalised the following three for this report. The advantage was have high PnL and most profitable.

**Strategy 1 - Stochastic reversal**

The stochastic oscillator is an indicator that helps determine when the price of an asset is about to change direction. It does this by giving signals on whether an asset is overbought or oversold. If the asset is overbought, it could be due for a reversal to the downside and if oversold, it could be due for a reversal to the upside. Traders use the stochastic oscillator to help them exit existing trades before a trend changes. They also use it to enter trades just as a new trend is beginning. The stochastic oscillator comprises two moving average lines. It measures an asset's recent closing price relative to its high-low range over a period of time. It is made up of two moving average lines that travel in between three zones on a chart: overbought (80-100), neutral (20-80) and oversold (0-20).

**Strategy 2 - Increasing Triple Exponential Average(TRIX)**

The Triple Exponential Average (TRIX) is a momentum indicator used by technical traders that shows the percentage change in a triple exponentially smoothed moving average. When it is applied to triple smoothing of moving averages, it is designed to filter out price movements that are considered insignificant or unimportant. As a powerful oscillator indicator, TRIX can be used to identify oversold and overbought markets, and it can also be used as a momentum indicator. When TRIX is used as a momentum indicator, a positive value suggests momentum is increasing while a negative value suggests momentum is decreasing. Many analysts believe that when the TRIX crosses above the zero line, it gives a buy signal, and when it closes below the zero line, it gives a sell signal. Also, any divergence between price and TRIX can indicate significant turning points in the market. Two main advantages of TRIX over other trend-following indicators are its excellent filtration of market noise and its tendency to be a leading than lagging indicator.

**Strategy 3 - Three Green Candles**

Three green soldiers is a bullish candlestick pattern that is used to predict the reversal of the current downtrend in a pricing chart. The pattern consists of three consecutive long-bodied candlesticks that open within the previous candle’s real body and a close that exceeds the previous candle's high. These candlesticks should not have very long shadows and ideally open within the real body of the preceding candle in the pattern. The three white soldiers candlestick pattern suggests a strong change in market sentiment in terms of the stock, commodity or pair making up the price action on the chart. When a candle is closing with small or no shadows, it suggests that the bulls have managed to keep the price at the top of the range for the session. Basically, the bulls take over the rally all session and close near the high of the day for three consecutive sessions. In addition, the pattern may be preceded by other candlestick patterns suggestive of a reversal.

**Simulation Results**

A csv file is attached to GitHub which has all the detailed results of all the run. The top run has following statistics.

Strategy 1 - Stochastic reversal

This follows stochastic reversal. The output is humber barely 0.13% positive.



Detailed Trade sheet and simulation run output can be found here -

<https://github.com/adityaprasann/WQUCapstoneRepository4Group44/tree/master/stochastic%20reversal>

Strategy 2 - Increasing Trix

This follows increasing trix, trending strategy. Thr top run gives a handsome 22% returns.



Detailed Trade sheet and simulation run output can be found here -

<https://github.com/adityaprasann/WQUCapstoneRepository4Group44/tree/master/increasing%20trix>

Strategy 3 - Three Green Candles

This follows trending strategy, three consecutive green candles, trix > 0 and solid body for buy. Top run has 12.5% return



Detailed Trade sheet and simulation run output can be found here -

<https://github.com/adityaprasann/WQUCapstoneRepository4Group44/tree/master/three%20green%20candles>

**Justification**

**Justification For Short term Trading Model and Indian Equities market**

The model presented in this report is based on Indian equity market. Indian equity markets are mature and have enough stocks and liquidity for such analysis to be done. However, it can be re purposed and re used in any Equity market which has enough trade data and liquidity.

**Justification for Moving Average**

Moving averages are pervasive in technical stock market analysis because they are able to smooth price data, form trend-lines, and create an easily interpreted visual aid. Developed as a statistical tool for use in conjunction with data sets that span a specific period of time, moving averages have proven to be well-suited for price charts and other indicators.

Simple moving averages (SMAs) are calculated by the sum of data points in a time interval divided by the number of time periods therein. For example, a standard 10-day moving average on a candle stick chart takes the value of each closing price, adds them together, then divides the resulting figure by 10. The length of the interval and data points chosen are left up to the individual trader, making moving averages highly pliable.

The exponential moving average (EMA) uses the same principles as the SMA, except it applies more weight to the most recent price bars. By emphasising recent action, EMAs reduce lag in the time data and avoid distortions from information that may no longer be relevant.

**Justification for Methodology**

We were able to execute and evolve different methodology. The ones which are presented in this report are -

* Stochastic reversal
* Three green candles
* Increasing Trix

However we tried other approaches and combination on the same data set over serval runs to arrive this. As highlighted in the result section the above approaches gave the most profitable deals.

**Justification for Back Testing Framework**

There are many open source event driven trading systems are available like ZipLine, PyAlgoTrade, BackTrader. After exploring we have chosen back-trader as our preferred choice for backtesting. Back-trader is functionally very rich, can trade multi-stocks and has inbuilt multiprocessing. Apart from these advantages, BackTrader is easy to install and use. This platform is exceptionally well documented, with an accompanying blog and an active on-line community for posting questions and feature requests. Back trader supports a number of data formats, including CSV files, Pandas DataFrames, blaze iterators and real time data feeds from three brokers. These data feeds can be accessed simultaneously, and can even represent different timeframes. Supported brokers include Oanda for FX trading and multi-asset class trading via Interactive Brokers and Visual Chart.

* Project Page: [www.backtrader.com](https://www.backtrader.com/)
* Github: [github.com/mementum/backtrader](https://github.com/mementum/backtrader)
* License: [GPL v3.0](https://github.com/mementum/backtrader/blob/master/LICENSE)
* Built-in support for several sources: *CSV*, Database-Sources, YahooFinance, Interactive Brokers, Oanda v1, ...
* Any number of simultaneous data feeds (memory constrained, obviously) can be run simultaneously
* Multiple timeframes can be mixed and run
* Integrated Resampling and Replaying
* The trading logic and the broker are always run on an event by event basis
* The calculation for indicators is vectorised if possible (source data can be preloaded)

**Limitations and Further Study**

Study is based on the assumption that new market information will generate a new trading signal resulting into trade execution. But in a live trading there may be many other unknown involved in this process. Transaction Costs, Slippage and Market Impact has been ignored in this study. The framework back trader  lacks straight forward way to do something like walk-forward analysis. It is not very intuitive to be able to tell a which dates we want to use for training, which we want to use for testing, and then run lots of these tests in batch. The paper requires further analysis study on the dangers of overfitting, transaction cost and transaction constrain.

Document concludes from number of experimental strategies involving moving average crossover, that non-linear time series study is a complex topic. It requires more study from many different angles and not just moving average price crossover of price time series. Price return and price volatility should be explored to make trading strategy more robust.

**Conclusion**

The goal of this paper was to develop, test and investigate short term trading model for liquid Indian equities and present a profitable portfolio. The outcome of this study exercise provides needed depth of knowledge of existing trading indicators. This paper has documented the working experience with trading model development, use of back-testing cum trading framework and trading model evaluation experience. As part of new trading model development exercise, many well-known existing trading strategies were studied and back-tested with historical data.

Multiple combination of trading strategy were tried during the paper. The entry signal, exit signal and trade size etc. has been specified clearly in the respective strategies. As described in this report the resulting final model has been back-tested with three years of historical daily price data. As stated in many academic literatures, three years’ worth of data for a wide variety of forty eight liquid equity indexes would constitute sufficient number of observations to determine its success. It was observed during the back testing that developing a trading strategy is essentially a task to determine whether prices for a certain time horizon will be mean-reverting or trending. The portfolio presented has the capability to act as assistance to a trader for deciding his short term trading strategy. It can be extended as an autonomous trading tool with an interface to a stock exchange. Paper concludes that more research would be require to make short term trading model more robust.

**References**

1. Jan Ivar Larsen, “Predicting Stock Prices Using Technical Analysis and Machine Learning” (2010).
2. Chan, E. 2017. Machine Trading: Deploying Computer Algorithms to Conquer the Markets. John Wiley & Sons.
3. Miner, C.R., 2008, High Probability Trading Strategies Entry Exit-Forex Futures Stock Markets, ISBN 978-0-470-18166-9, John Wiley & Sons Inc.
4. <https://www.quantshare.com/item-1541-system-quality-number-indicator>[Accessed July 2020]
5. <https://www.backtrader.com/docu/analyzers-reference/#tradeanalyzer> [Accessed July 2020]
6. <https://www.backtrader.com/docu/analyzers-reference/#transactions> [Accessed July 2020]
7. Choudhury, Manoj Kumar, and Prabina Rajib. 2017. Informational efficiency of national stock exchange (NSE), India: A comparison with seven selected markets. South Asian Journal of Management 24, (1) (Jan): 56-80, https://search.proquest.com/docview/1903429233[accessed July 2020].
8. <https://www.investopedia.com/terms/m/movingaverage.asp> [Accessed July 2020]
9. <https://ntguardian.wordpress.com/2017/06/12/getting-started-with-backtrader/> [Accessed July 2020]
10. <https://www.investopedia.com/ask/answers/122314/what-are-main-advantages-using-moving-averages-ma.asp> [Accessed July 2020]
11. <https://learn.tradimo.com/technical-analysis-how-to-work-with-indicators/stochastic-oscillator> [Accessed July 2020]
12. <https://www.investopedia.com/terms/t/trix.asp> Accessed July 2020]

**Appendix I : Indian Equity Indexes**

Below table summaries the ticker code for NIFTY stocks which were used.

|  |  |
| --- | --- |
| Adani Ports and Special Economic Zone Ld | ADANIPORTS |
| Asian Paints Ltd | ASIANPAINT |
| Axis Bank Ltd | AXISBANK |
| Bajaj Auto Ltd | BAJAJ-AUTO |
| Bajaj Finserv Ltd | BAJAJFINSV |
| Bajaj Finance Ltd | BAJFINANCE |
| Bharti Airtel Limited | BHARTIARTL |
| Bharat Petroleum Corp Ltd | BPCL |
| Britannia Industries Ltd | BRITANNIA |
| Cipla Ltd | CIPLA |
| Coal India Ltd | COALINDIA |
| Either Motors Ltd | EICHERMOT |
| Gail (India) Ltd | GAIL |
| Grasim Industries Ltd | GRASIM |
| HCL Technologies Ltd | HCLTECH |
| HDFC Bank Limited | HDFCBANK |
| Housing Development Finance Corp Ltd | HDFC |
| Hero Motocorp Ltd | HEROMOTOCO |
| Hindalco Industries Ltd | HINDALCO |
| Hindustan Unilever Ltd | HINDUNILVR |
| ICICI Bank Ltd | ICICIBANK |
| Indusind Bank Ltd | INDUSINDBK |
| Bharti Infratel Ltd | INFRATEL |
| Infosys Ltd | INFY |
| Indian Oil Corporation Ltd | IOC |
| ITC Ltd | ITC |
| JSW Steel Limited Fully Paid Ord. Shrs | JSWSTEEL |
| Kotak Mahindra Bank Ltd Fully Paid Ord. Shrs | KOTAKBANK |
| Larsen & Toubro Limited | LT |
| Maruti Suzuki India Ltd | MARUTI |
| Mahindra and Mahindra Industries Ltd | MM |
| Nestle India Ltd | NESTLEIND |
| NTPC Limited | NTPC |
| ONGC Limited | ONGC |
| Power Grid Corporation of India Limited | POWERGRID |
| Reliance Industries Limited | RELIANCE |
| State Bank Of India | SBIN |
| Sun Pharmaceutical Industries Limited | SUNPHARMA |
| Tata Motors Limited Fully Paid Ord. Shrs | TATAMOTORS |
| Tata Steel Limited Fully Paid Ord. Shrs | TATASTEEL |
| Tata Consultancy Services Limited | TCS |
| Tech Mahindra Ltd | TECHM |
| Titan Company Ltd | TITAN |
| UltraTech Cement Ltd | ULTRACEMCO |
| UPL Ltd Fully Paid Ord. Shrs | UPL |
| Vedanta Limited | VEDL |
| Wipro Limited | WIPRO |
| Zee Entertainment Enterprises Limited | ZEEL |

**Appendix –II**

**Python code**

<https://github.com/adityaprasann/WQUCapstoneRepository4Group44>

Increasing Trix -

<https://github.com/adityaprasann/WQUCapstoneRepository4Group44/blob/master/increasing%20trix/increasing_trix.py>

Stochastic Reversal -

<https://github.com/adityaprasann/WQUCapstoneRepository4Group44/blob/master/stochastic%20reversal/stochastic_reversal.py>

Three Green Candles -

<https://github.com/adityaprasann/WQUCapstoneRepository4Group44/blob/master/three%20green%20candles/three_green_candles.py>

**Data files**

<https://github.com/adityaprasann/WQUCapstoneRepository4Group44/tree/master/data>