SCALA PROJECT

Hedge Fund Application:
Real Time Risk Analysis

Team 2:
Shreya Nair
Amit Pingale
Mayank Gangrade

Goal

Building reactive application for portfolio management and risk analysis

Leveraging:

- Kafka + Spark streaming
- Spark Analysis engine
- MongoDB for maintaining historic records + batch processing

Why Real-time Big Data Pipeline Important

It is estimated that by 2020 approximately 1.7 megabytes of data will be created every second. This results in an increasing demand for real-time and streaming data analysis. For historical data analysis descriptive, prescriptive, and predictive analysis techniques are used. On the other hand, for real-time data analysis, streaming data analysis is the choice. The main benefit of real-time analysis is one can analyze and visualize the report on a real-time basis.

Technology and Tools:

1. Alpha Vantage API for Real Time Data
2. Kafka for capturing Real Time Data
3. Spark Streaming For Consuming Data
4. Spark + Scala for performing Analysis on the data
5. Spark Machine Learning Library
6. MongoDb*
7. Tableau
8. Jupyter Notebook - Scala kernel

* Dumping historic and predicted data on NoSQL database like MongoDB

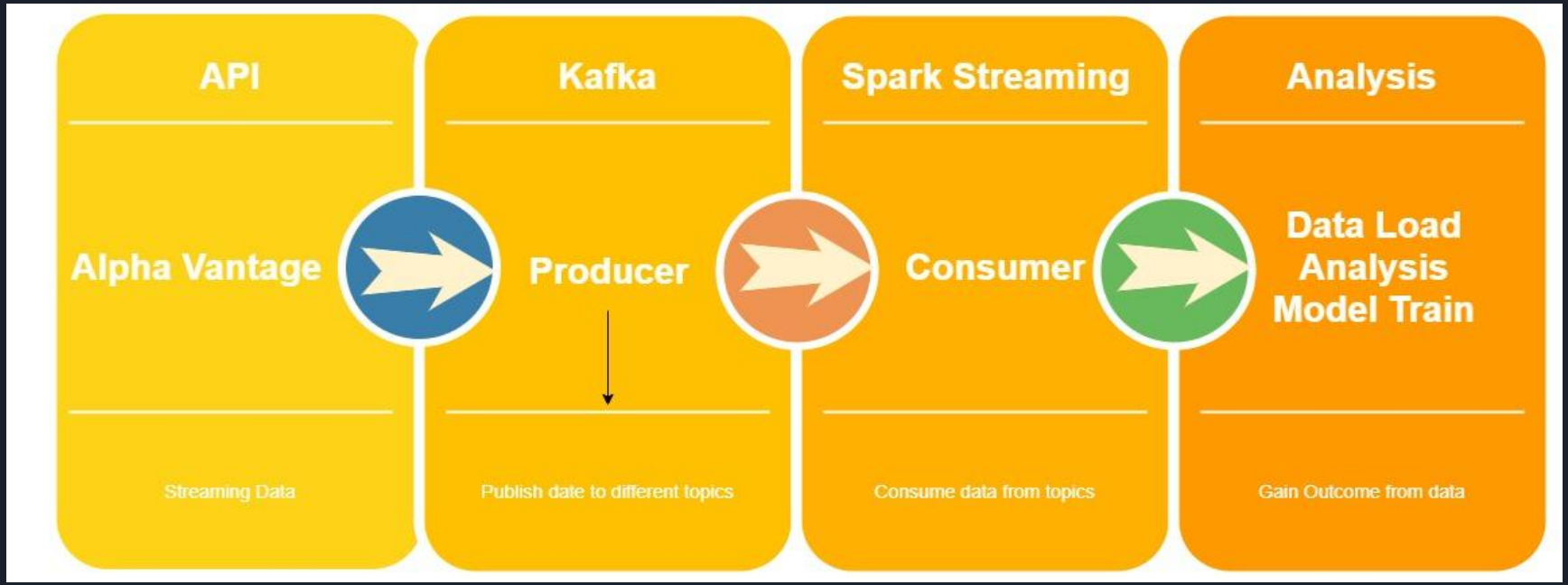
Analysis on Data

1. Linear Regression
2. Decision Tree Regression
3. Random forest Regression
4. Gradient Boosting Regression

Data features: Stocks

1. Open
2. High
3. Close
4. Adj Open
5. Volume
6. Low

High Level Architecture



Acceptance criteria

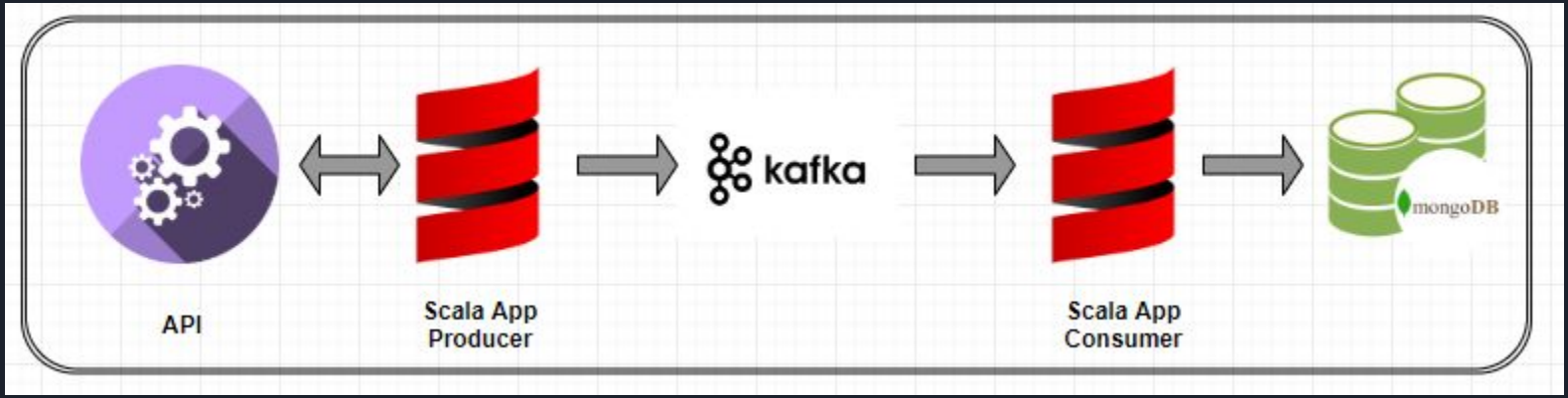
- Fetching stock/option data every 5 mins and Publishing the same on Kafka Topics
- Consume the data from Kafka as soon as it arrived and store the same in to Database
- Developing ensemble of 4 machine learning models
- Selecting best model depending upon RMSE value (RMSE < 0.7)
- Update data in real time interactive dashboards

Project subdivision:

- Data Engineering
- Machine Learning
- Data Analytics

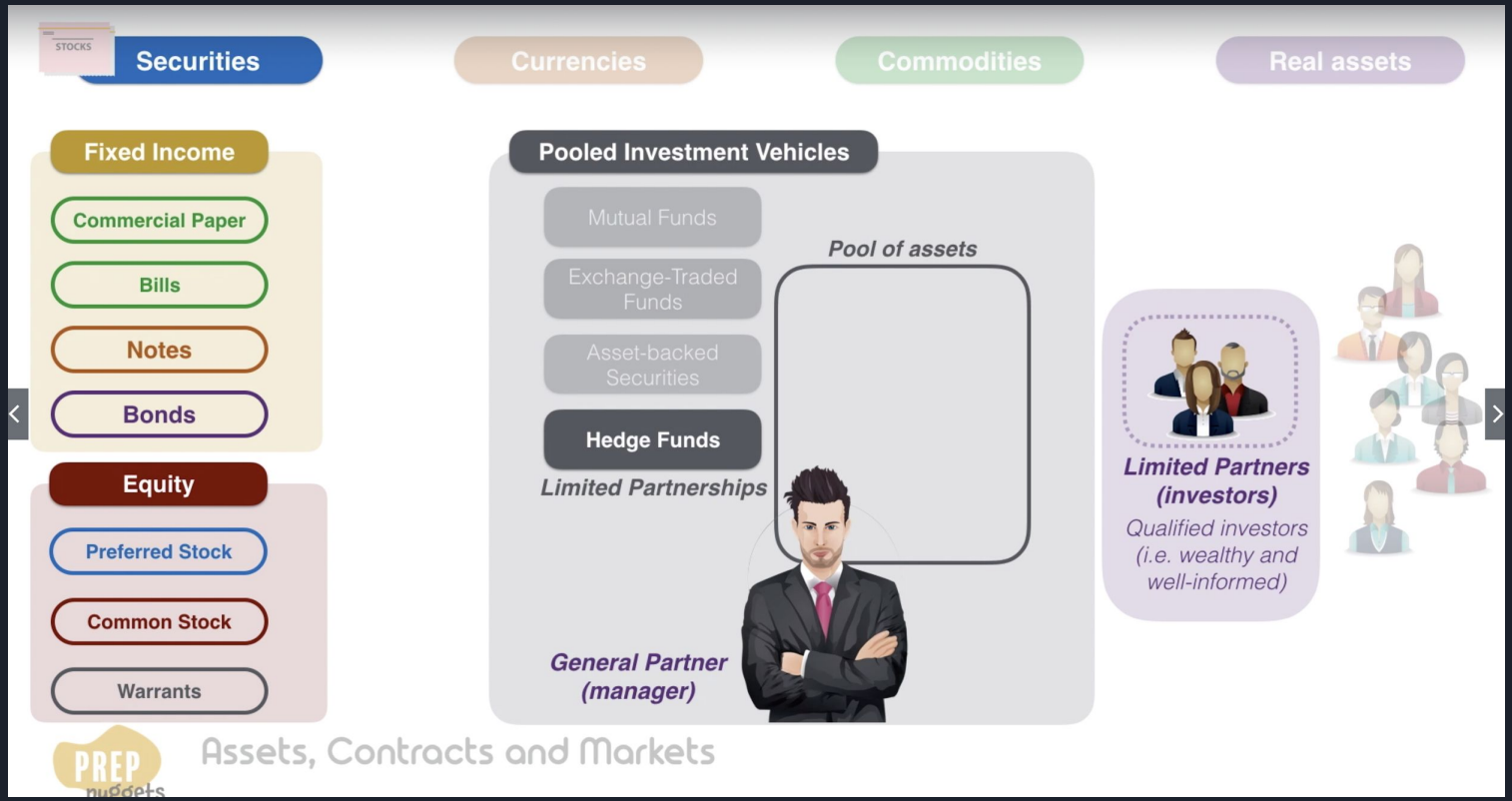
Data Engineering

Data Engineering:



Machine Learning

What is Hedge Fund?

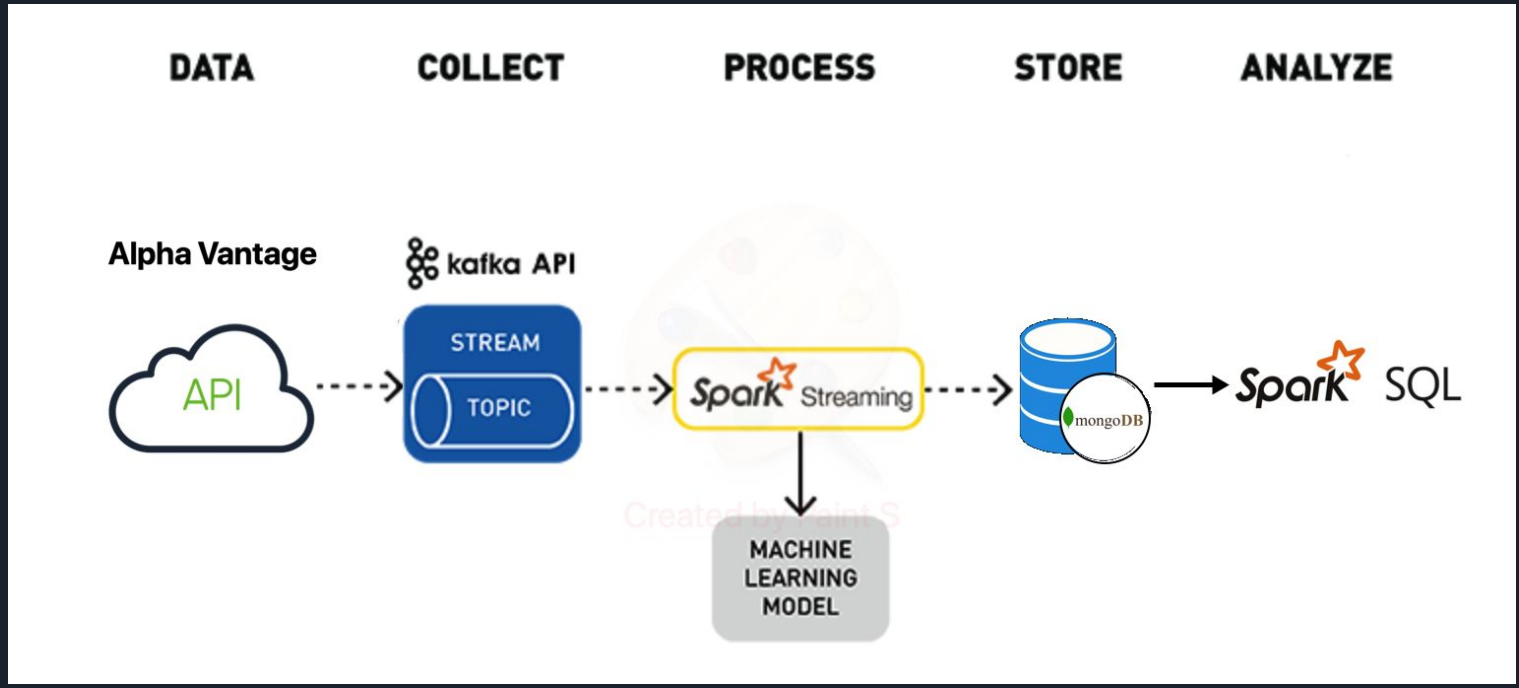


Risk Management in Portfolio Construction

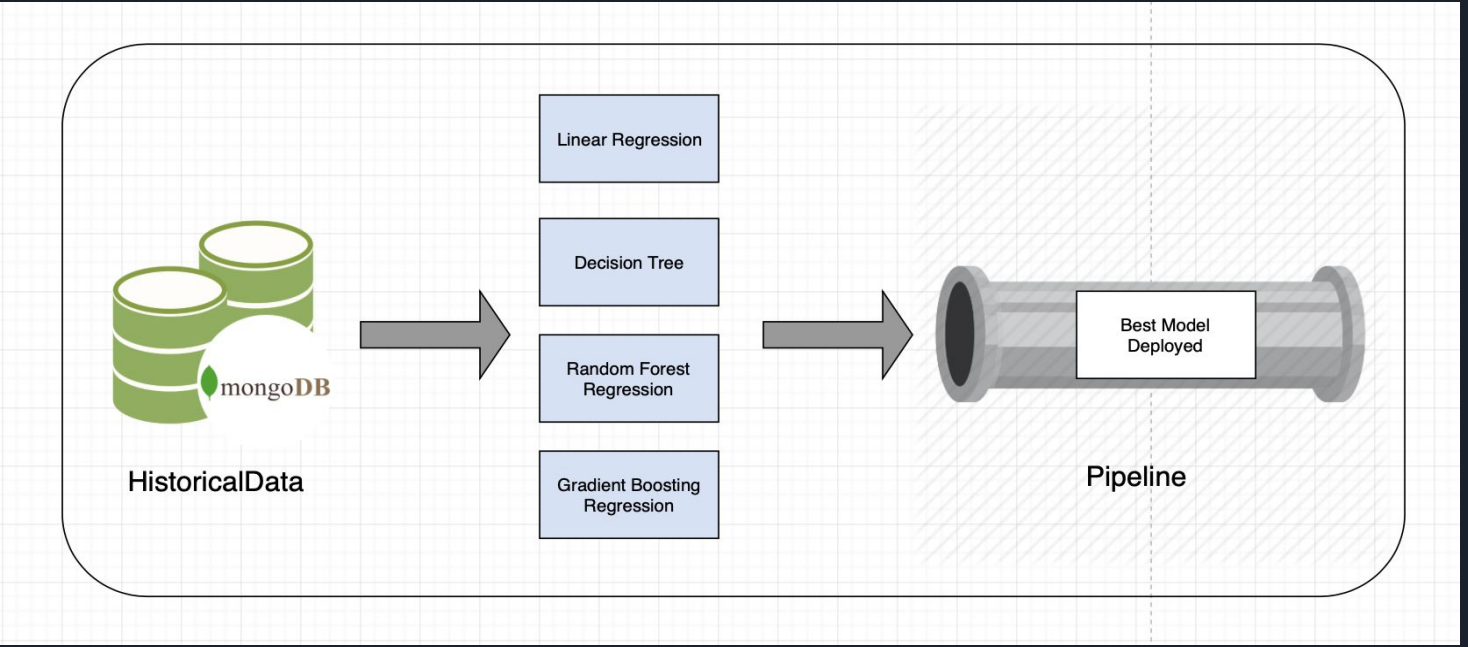
Methods Used in Hedge Fund Company:

- Inverse correlation of the securities
- Protected Put (Long Position on a Asset + Long Put Position on the same Option Market Asset)
- Fiduciary Call (Long Call on the Options Market + Risk Free Bond)

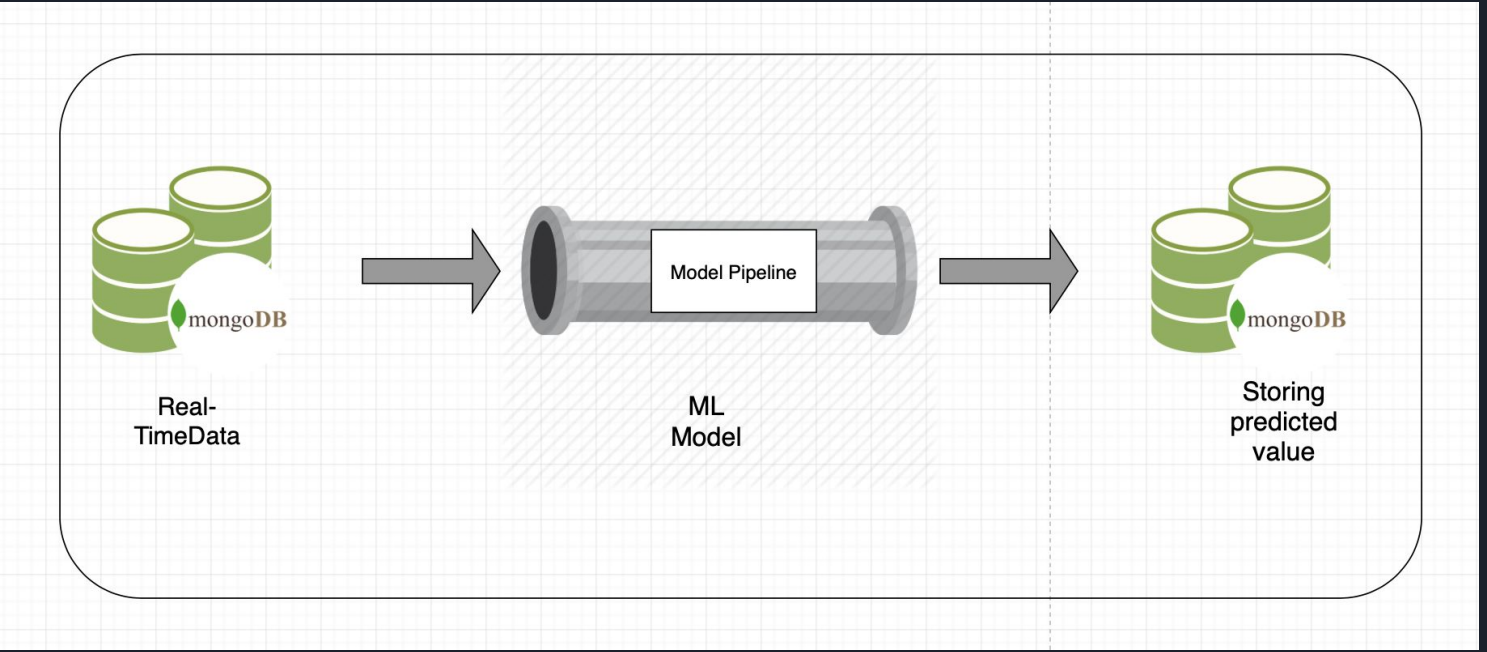
Architecture



ML Pipeline Construction



Real-Time Prediction

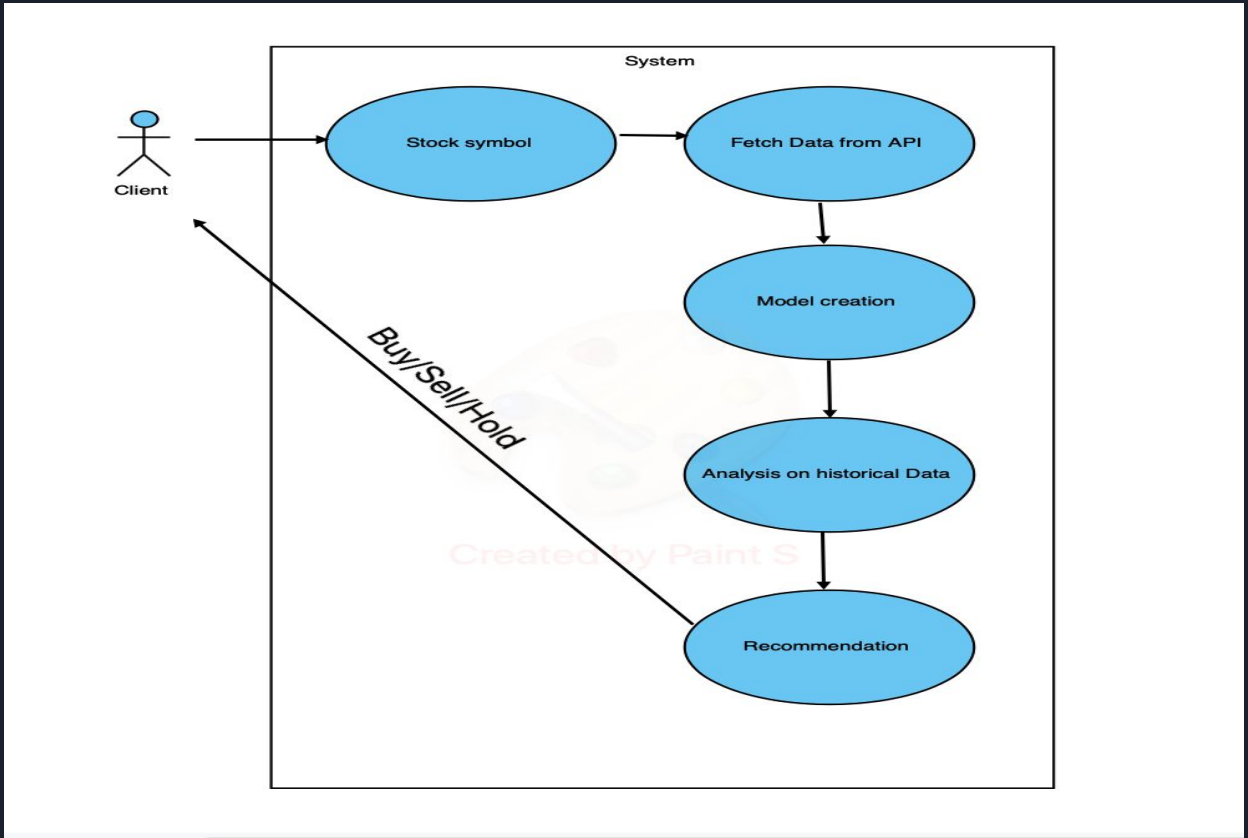


Trading Strategy

Assumption:

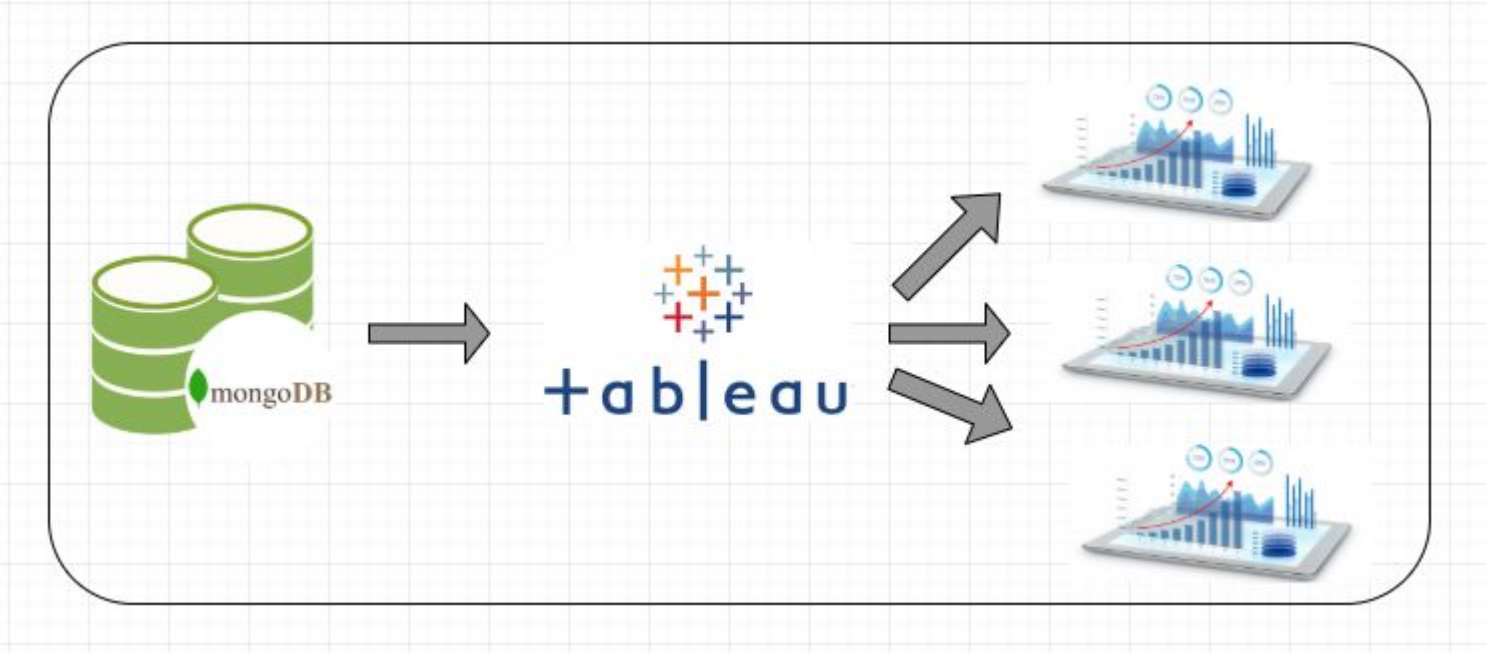
- The market value of a security is corrupted by noise
  - High Frequency Trading
  - Market Sentiment
  - False Information
- The model constructed is immune to these noises
- Conclusion:
  - Real Value > Predicted Value ---> Security is overvalued ---> Sell/Short Position
  - Real Value < Predicted Value ---> Security is undervalued ---> Buy/Long Position
  - Real Value == Predicted Value ---> Security is properly valued ---> Hold Position

Use Case Diagram



Data Analysis

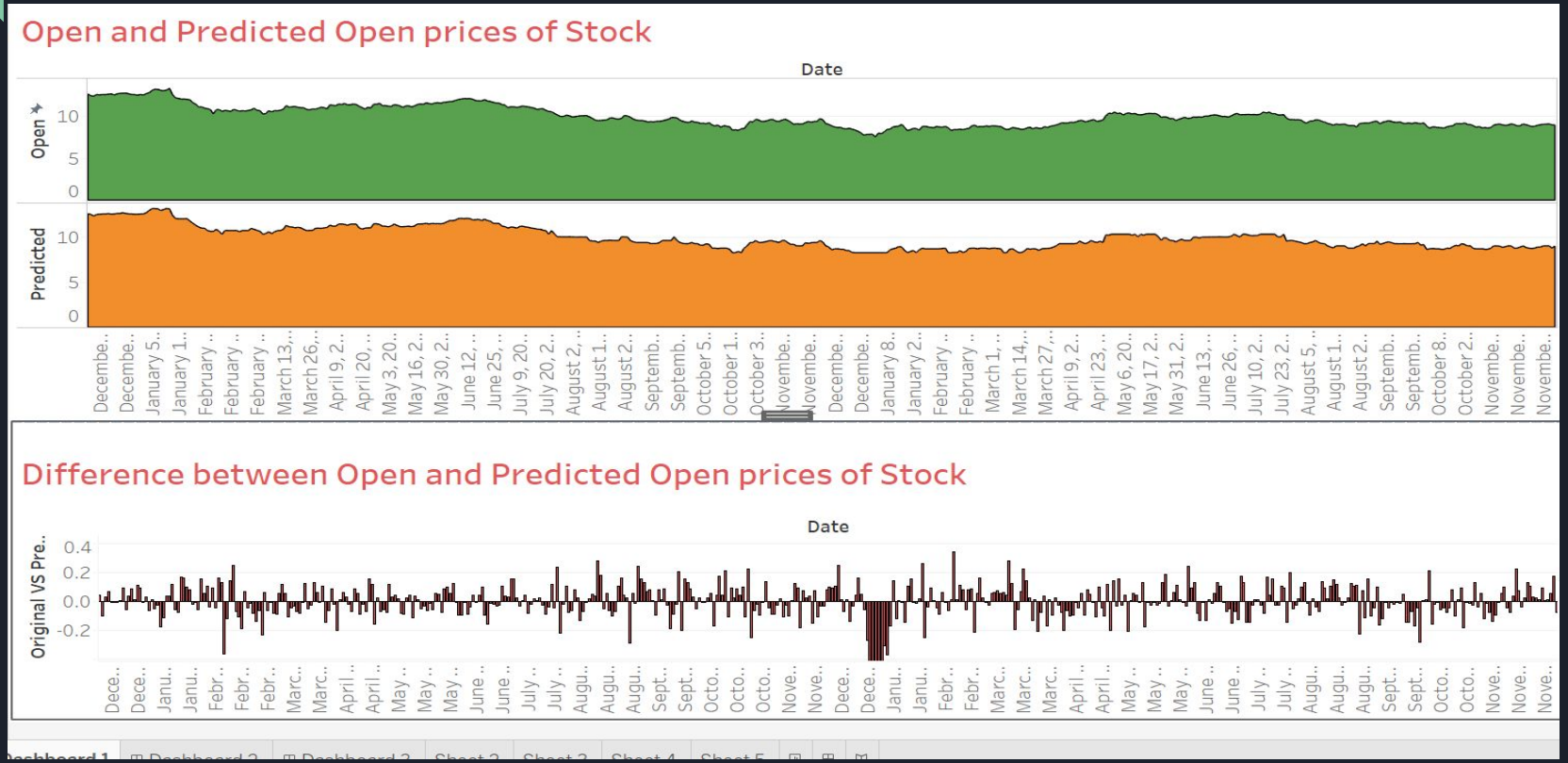
Data Analytics:



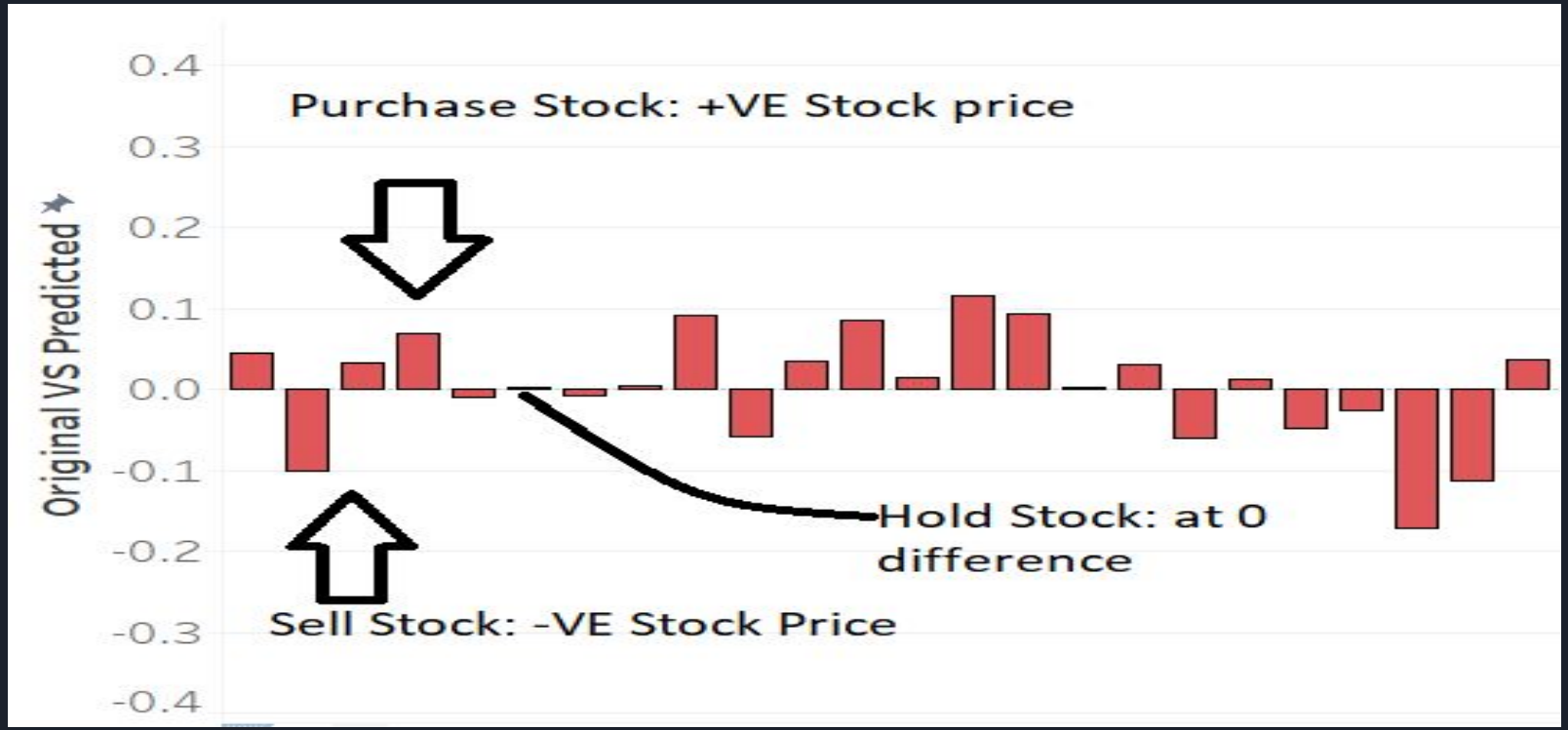
Data Analytics

1. Live data Streaming: Fetching Open stock price and Predicted stock price getting fed in MongoDB on real time basis
2. Connecting MongoDB with Tableau using BI Connector - Simba for Real Time data visualisation
3. In Tableau, presenting dynamic charts (changing with respect to Date-time) as per the data input
4. Showing Error (Original - Predicated stock price) , to guide user, if to sell or purchase the stock
5. Showing how change in Open and Predicted Open stock affects the Volume of the stock

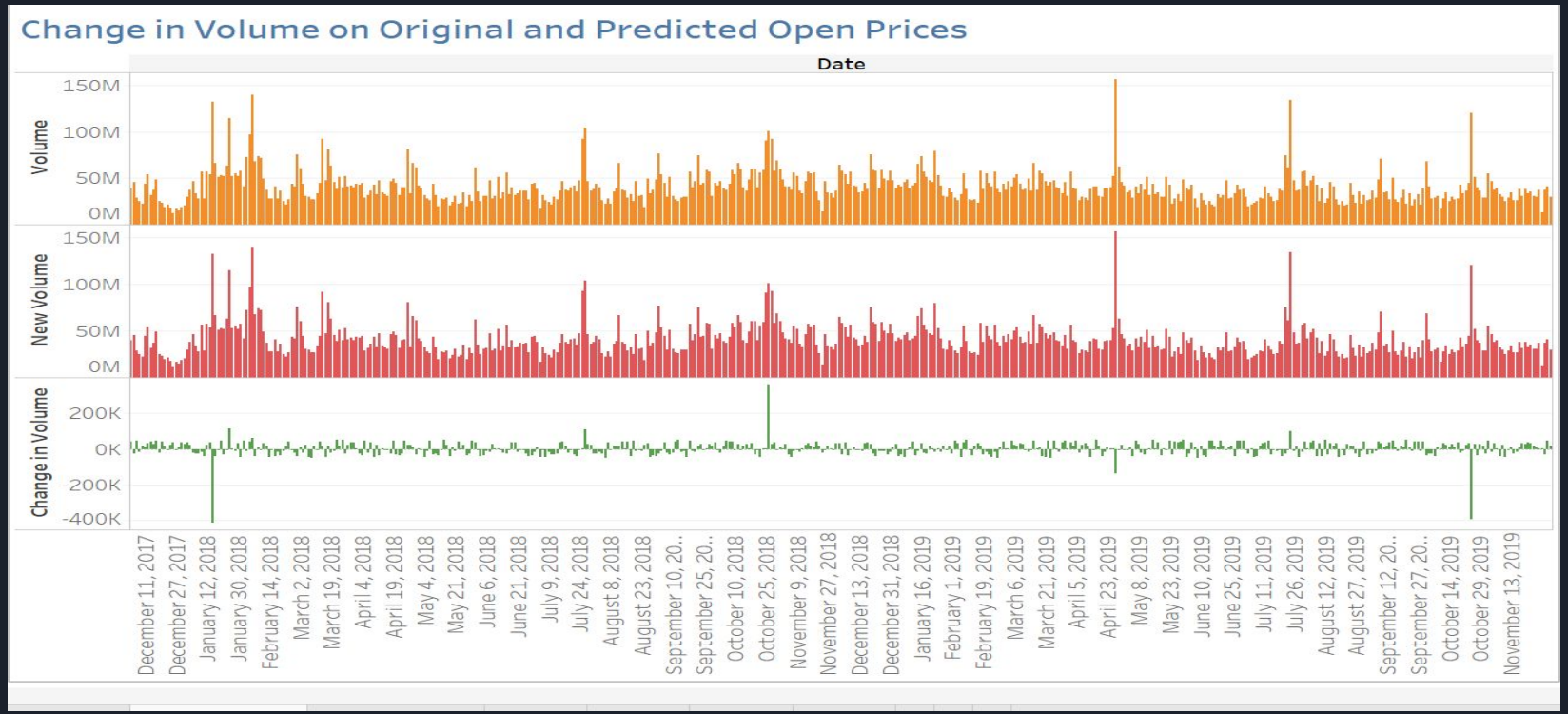
Data Analytics (a)



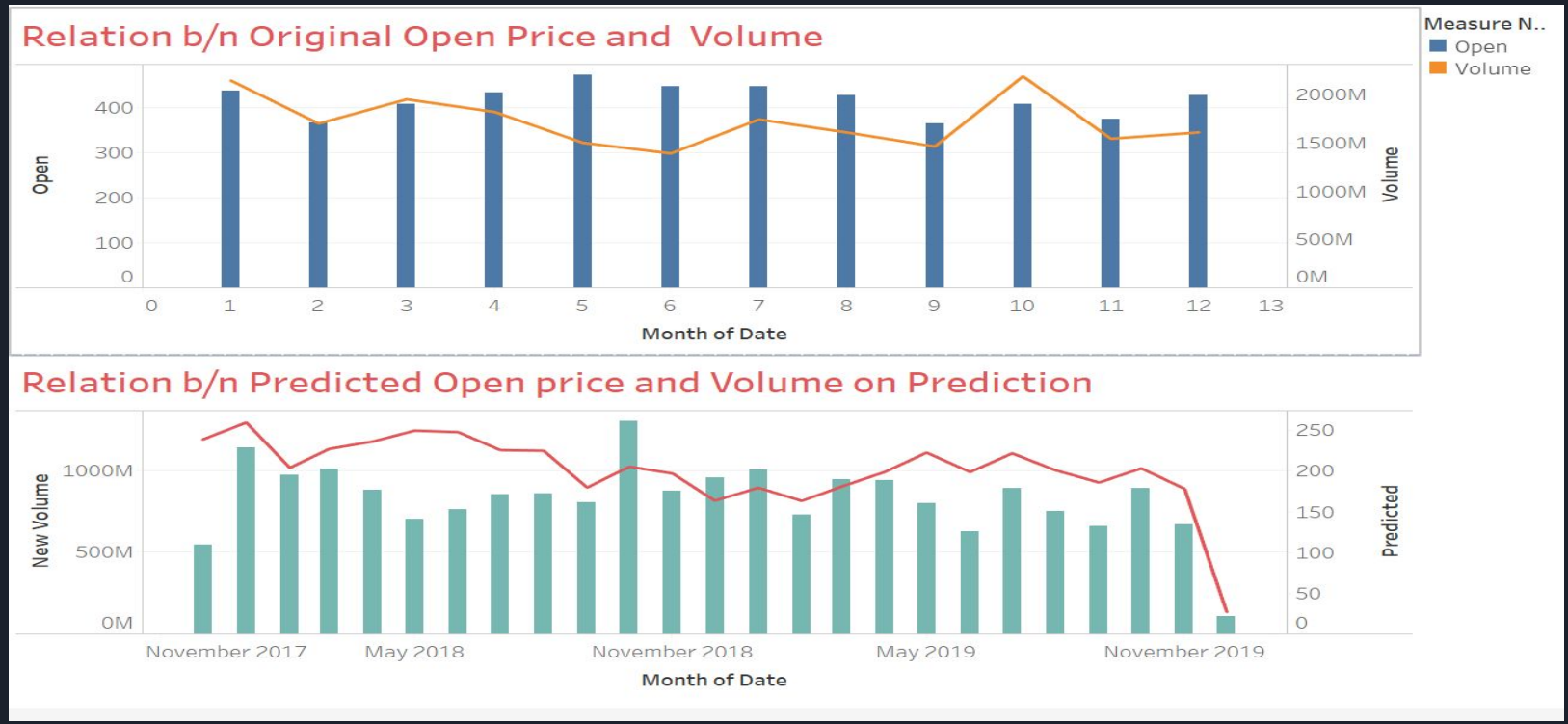
Data Analytics: When to sell, Purchase or Hold a stock



Data Analytics (b)



Data Analytics (c)



Thank You

Milestones

Nov 15 -> Project Proposal Acceptance

Nov 22 ->Infrastructure ready (API + Kafka + Spark Streaming + Spark Analytics Engine + MongoDb)

Nov 29 ->Build model + Tune Hyper-parameters

Dec 6 -> Test cases and final evaluation