National University of Computers and Emerging Sciences -FAST

Artificial Intelligence

Project Report

Group Members:

Muhammad Ali 20k-1888

Huzaifa Asad 21K-4838

Class: BCS-6E

Instructor: Muhammad Osaid

Submitted on 16th May, 2024

Stock Prediction Using KNN, Naive Bayes, and Logistic Regression

Abstract

This project aims to predict stock prices using three different machine learning models: K-Nearest Neighbors (KNN), Naive Bayes, and Logistic Regression. The goal is to compare the effectiveness of these models in the context of stock market prediction and determine which model provides the most accurate forecasts.

Introduction

The stock market is inherently volatile and unpredictable. Investors and financial analysts have long sought tools that can provide an edge in predicting stock movements. Machine learning offers powerful techniques to model and predict stock prices based on historical data. This report focuses on three algorithms: KNN, Naive Bayes, and Logistic Regression, each offering different approaches to the prediction task.

Objectives

•To implement and train KNN, Naive Bayes, and Logistic Regression models on historical stock data.

•To evaluate and compare the performance of each model in terms of accuracy and efficiency.

•To determine which model is most suitable for stock price prediction.

Methodology

Data Collection

Data is collected from Kaggle

Feature Selection

•Price Change

•Typical Price

•Mean Deviation

•Avg Gain and Avg Loss

•RSI (Relative Strength Index)

•CCI (Commodity Channel Index)

• ROC (Rate of Change)

•EMA12 and EMA26 (Exponential Moving Averages)

•MACD(Moving Average Convergence Divergence)

•Bollinger Bands (BB_Middle, BB_Upper, BB_Lower)

•Momentum

•Momentum

•Label

Model Implementation

K-Nearest Neighbors (KNN)

•In the KNN code, features are first scaled using StandardScaler to ensure uniformity in distance calculations. The KNN classifier is then trained with a specified number of neighbors (k=5) to predict stock prices based on the similarity of feature vectors (like OPEN, CLOSE prices) to the k-nearest labeled data points.

•**Usage**: KNN is used to classify stock price movements based on historical price data, where the prediction is influenced by the most similar historical instances.

Naive Bayes

• Gaussian Naive-Bayes is utilized where the features (like OPEN, CLOSE prices) are assumed to follow a Gaussian (normal) distribution. The algorithm calculates the probabilities of the outcomes (stock price movements) based on the Bayes theorem, treating each feature as independent.

•**Usage**: It's employed for quick and straightforward probabilistic predictions in stock movements, making it effective in scenarios with assumption of normal distribution in feature sets and independence among them.

Logistic Regression

• Logistic regression is implemented by first selecting relevant financial indicators as features, followed by feature scaling for normalization. The model then uses these features to predict the probability of a stock's price increasing, applying a sigmoid function to output probabilities.

•**Usage**: This model is particularly apt for binary outcomes (e.g., price up or down), using weighted sums of features passed through a logistic function to estimate the likelihood of stock price increments.

Model Training

the models—K-Nearest Neighbors (KNN), Logistic Regression, and Gaussian Naive-Bayes—are trained using a typical workflow that includes loading data, preprocessing, feature scaling, training, and evaluation. Here’s how each model is trained:

K-Nearest Neighbors (KNN):-

the dataset is split into training and test subsets using train test split.

A KNeighbors Classifier is instantiated with a predefined number of neighbors (k=5).

The model is then trained (fit) on the training data, which consists of feature vectors and the corresponding labels. In this context, the labels could be whether the stock price increased or decreased, which is derived from the stock data.

Logistic Regression:-

The data is split into training and testing sets.

A Logistic Regression model is initialized, which internally uses a sigmoid function to estimate probabilities that the dependent variable belongs to a particular class.

The model is trained on the training dataset, learning to associate the input features with the probability of the target class (e.g., price increase vs. decrease).

Gaussian Naive-Bayes:-

The Gaussian Naive-Bayes model is particularly fast for training and predictions, useful for making quick inferences about stock price movements based on the calculated probabilities from the training data.

Challenges and Learning Points:

Handling Data Imbalance: We encountered challenges related to imbalanced data, which were mitigated by employing appropriate preprocessing techniques and model evaluations to ensure robust performance.

Model Selection and Optimization: Selecting the right model parameters and understanding the trade-offs between model complexity and performance were pivotal learning aspects of this project.

Future Directions:

Deep Learning: Exploring deep learning techniques such as Recurrent Neural Networks (RNNs) or Long Short-Term Memory networks (LSTMs) could potentially improve predictions by capturing sequential patterns in stock price movements over time.

Real-Time Data Application: Implementing the models in a real-time prediction system to provide dynamic stock trading insights.

Feature Engineering: Further exploration into feature engineering, possibly incorporating external data sources such as economic indicators or news sentiment analysis to enhance prediction accuracy.

Conclusion:

This project not only reinforced the applicability of machine learning in financial analytics but also highlighted the importance of methodical data analysis, thoughtful model choice, and the continuous need for model assessment and improvement. Through the judicious application of KNN, logistic

regression, and Gaussian Naive Bayes, we demonstrated that machine learning could provide substantial insights and predictive power in the complex domain of stock market trading. As we look forward, the integration of more sophisticated models and diverse data sources stands as the next frontier in enhancing the predictive accuracy and applicability of our financial models.

Inspiration:-

https://www.tradingview.com/

Binance.com