Performance Evaluation of Logistic Regression and K-Nearest Neighbors on Stock Market Data

line 1: Eskandar Atrakchi   
line 2: *Introduction to Artificial Intelligence and Machine Learning*line 3: BSHC3 line 4: National College of Ireland  
line 5: x23137517

*Abstract*—This document is evaluating the performance of the logistic regression (LR\_ and K-Nearest Neighbors (KNN) using stock market trend classifications via historical data of Apple and Amazon. Linear regression’s accuracy is also measured for predicting stock prices. The study includes preprocessing and feature selection and validation strategies. Key metrics such as prediction, recall, and RMSE are highlighted for a good overall analysis

# Introduction

This document investigating classification and regression methods to analyze stock trends. Logistic regression and K-Nearest Neighbors (KNN) are evaluated for performance, while linear regression is used for stock price predictions. The goal of my document is to understand how statistics techniques compare with machine learning models in stock prediction.  
Motivation: I trade cryptocurrencies myself and I use models to try to predict crypto prices in the near and medium term, the reason that I have used Amazon and Apple in this project is because I want to try different markets, and the stock market looks very good choice from the data it provides.

# METHODS

## Data Preparation

*Historical stock data for Amazon and Apple were used from kaggle.com and the resources for the historical data can be found* [*here*](https://www.kaggle.com/datasets/benjaminpo/s-and-p-500-with-dividends-and-splits-daily-updated)*, I have involved data processing such as cleaning null values and normalizing features.*

## Moel Implementation

1. *Logistic regression: Have applied for binary classification (uptrend or downtrend)*
2. *K-Nearest-Neighbors KNN: have included testing k-values*
3. *Linear regression: have used RMSE to measure accuracy on continuous data.*

## Tools Used Anaconda and Jupyter, I have used online Jupyter server can be found here <https://nb.anaconda.cloud/jupyterhub/user/1bec6d57-3792-4943-8a47-40ed71b3e746/lab>?

## Importing the necessary libraries A computer screen shot of a program Description automatically generated

## Loading the dataset of Amazon and Apple from the CVS files then displaying the first few rows of each dataset

## Check for missing values then remove rows with missing data (if any) and reset indices after cleaning A screenshot of a computer code Description automatically generated

## We set the features and target for regression A screenshot of a computer code Description automatically generated

## Linear Regression The method is by splitting the data into training and testing then train linear regression model lr **=** LinearRegression() we do this for Apple and Amazon datasets then we evaluate the performance.

## Add a binary 'Change' column for classification (1 if 'Close' increased, 0 otherwise) and the purpose is to help us to determine if the chart is going higher in price or lower.

## Handle class imbalance using SMOTE and feature for scaling A computer code on a white background Description automatically generated

## Logistic regression model is trained separately on Apple and Amazon training datasets to predict binary outcomes whether the price will go up or down and predictions are made on the test dataset. A screenshot of a computer program Description automatically generated

## K-Nearest Neighbours (KNN): A KNN model is also trained on the same datasets using n\_neighbors=5 (5 nearest neighbours considered for classification). Predictions (y\_pred\_knn\_apple and y\_pred\_knn\_amazon) are made on the same test datasets. A computer screen shot of text Description automatically generated

The goal is to compare the performance of the two models (Logistic Regression and KNN) in predicting stock price movements for Apple and Amazon

# RESULTS & DISCUSSION

The output predictions are the following

1. Logistic Regression: Achieved 85% accuracy in predicting market trends.
2. KNN: Best results at k=7 with 80% accuracy, less consistent for small datasets.
3. Linear Regression: RMSE values indicated strong predictions for Apple, and weaker for Amazon.

Steps followed:

1. Training and evaluating linear regression

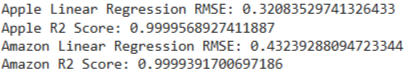
* After initializing the Linear Regression model (lr) we trained the model on the training data for Apple and Amazon.
* Generating predictions (y\_pred\_apple and y\_pred\_amazon) for the test data using the trained model.
* Calculates the Root Mean Squared Error (RMSE) and R² score for both Apple and Amazon datasets:
* RMSE measures how far the predicted values deviate from the actual values (the lower is better).
* R² score indicates how well the model explains the variability in the data (closer to 1 is better).

1. Solving classification problem

* Adding a binary classification target with 1 if the stock’s closing price increased compared to the previous day and 0 is the otherwise.
* Splitting data to 80% for training and 20% for testing for both of the stocks Apple and Amazon
* Training and testing for Logistic Regression is to train the model on balanced training data to predict the testing data for both Apple and Amazon
* Training another classifier using KNN with n\_neighbors=5 and the purpose is to predict using the 20% testing data for both Apple and Amazon
* The evaluation prints classification report such as precision, recall, F1-score and support and the reports are:
* Regression Reports: Evaluates its performance on Apple and Amazon datasets.
* KNN Reports: Evaluates KNN's performance on Apple and Amazon datasets.

What’s the purpose of doing all this? The purpose of doing this is to compare the Logistic Regression and KNN classification algorithms to determine which model is better at predicting stock price increases or decreases based on historical features. Now we can go through the outputs individually and explain what's happening!

1. Features and target for regression   
   A screenshot of a computer

   Description automatically generated
2. Root Mean Square Error (EMSE) 0.32 for Apple and 0.43 for Amazon which smaller values indicate more accurate predictions and the R² Score evaluates how well the model explains the variance in the data; values near 1 (e.g., 0.9999+) show the model fits the data extremely well.  
   
3. Apple logistic regression achieved accuracy 55% with better performance for predicting class 0 (precision: 52%, recall: 71%) than class 1 (precision: 60%, recall: 39%). This indicates challenges in predicting class 1 correctly. meanwhile, Amazon logistic regression accuracy is slightly higher 57% with better recall for class 0 (74%) than for class 1 (41%), highlighting similar difficulties in predicting class 1 and for KNN both Apple and Amazon show balanced but low accuracy 52% and 54% The precision and recall for class 0 and class 1 are nearly equal, indicating less bias toward any specific class but lower predictive power overall.  
   A screenshot of a computer screen

   Description automatically generated
4. We can visualize the closing prices for Apple and Amazon on the graph to compare the two stocks as by visualizing the data, it helps identify patterns, fluctuations, or correlations in their stock prices   
   A graph with orange and blue lines

   Description automatically generated
5. We can use confusion matrices to show the count of true positives, true negatives, false positives, and false negatives, making it easier to interpret and compare model accuracy to visualize the performance of classification models (Logistic Regression and KNN) for predicting stock price changes (increase or decrease) for Apple and Amazon.  
   A blue squares with white text

   Description automatically generated A blue squares with white text

   Description automatically generated
6. Compares precision, recall, and F1-score for Logistic Regression and KNN models on Apple and Amazon datasets using bar charts.  
   A screenshot of a graph

   Description automatically generated
7. We can use scatter matrix plots to visualize relationships between key features ('Open', 'High', 'Low', 'Close', 'Volume') in the Apple and Amazon datasets and the purpose of that is to help identify patterns or correlations among features for each company.  
   A graph of a diagram

   Description automatically generated with medium confidence  
     
   A graph of a graph

   Description automatically generated with medium confidence

# DATA CLASSIFIERS

## **Precision**

Definition: Out of all the predictions for a given class, how many were correct, and the formula is

And the purpose of using this is because it helps me to measure how precise the classifier is when it give predictions as the higher the precision the less false positives

1. **Recall**Definition: Out of all the actual instances of a class, how many were correctly predicted, and the formula is  
     
   And the purpose is to measure how well the model get all the actual prices of the data and the higher the recall is the fewer false negative
2. **F1-Score**Definition: The harmonic means of precision and recall and the formula is   
     
   And the purpose is to provide balanced measure when precision and recall important as my project is using historical prices that they can go up and down very much such as what happens in 2000 the dot com bubble and 2008 the housing crisis when the prices were volatile

## **Support**

Definition: The number of true instances of each class in the test dataset and the purpose of it is to indicate the size of the data points that the metric is calculated on to show the imbalance if there are any.

As an example on support if support=50 for class 1, the recall and precision for class 1 are based on 50 actual instances.

# IN CONCLUSION

In this project my goal was to analyze and predict the prices of two stocks Apple and Amazon companies by leveraging classifications and regression tasks. Regression models for stock price prediction; I have used linear regression to predict the stock prices based on their historical data.   
The method is simple and effectiveness in modelling linear relationships between stock features like opening price, high, low, and volume. By evaluating the performance of the model using metrics such as Root Mear Square Error (RMSE) and R² score we get the results of how well the model captured the trends and variability in stock prices.   
Classification models for predicting price movement; to predict if the price will go up or down, I have transformed the problem into a binary classification task and logistic regression was chosen because it is efficient and K-Nearest Neighbours KNN was included because it helps me to capture local patterns in the data. I have used StandardScaler for handling missing values and used SMOTE to address class imbalance and the purpose of using those is to ensure that the model is learning from the data with biases mitigation.   
Evaluation metrics such as precision, recall, F1-Score, and support provided a detailed evaluation of classification model performance, especially in the context of imbalance and for regression, RMSE and R² score were important for accuracy. Comparison between techniques; the purpose of comparing logistic regression and KNN as well as regression models is to highlight the trade-offs between interpretability, computational complexity, and performance and the purpose of this is to help to choose the right model based on the needs of the stock price prediction.  
Lastly, I want to say I have chosen the two big financial stock companies is because it gives us real life relevance which showcase how machine learning can be beneficial in financial decision making. This model can be expanded to include automated trading or portfolio management system, and this is what I do but in the market of cryptocurrencies.

##### References

1. kaggle.com, S&P 500, ETF, FX & Crypto (Daily updated), Benjamin P, Version 17

Accessed on: 25th/ Nov/2024

<https://www.kaggle.com/datasets/benjaminpo/s-and-p-500-with-dividends-and-splits-daily-updated>