**Determining features predictive of short-term stock market opening price**

Team 5: Gary Takahashi, Akram Mahmoud and Daniel Ackermann

Candidate paper for the degree of M.S. Applied Artificial Intelligence

AAI-500-01-SU23 – Probability and Stats for AI

Shiley-Marcos School of Engineering, University of San Diego

Dr. Ebrahim Tarshizi

June 26, 2023

**Determining features predictive of short-term stock market opening price**

The concept of market momentum posits a relationship between the current price of a stock to its value in the recent past, and is useful as a measure of overall market sentiment. As such, we hypothesized that time-sequenced analysis of a stock might predict short-term trends in stock price. One time-sequence analytic method is the moving average determination. We analyzed the moving averages of 30 stocks in the Dow Jones industrial average, and selected the parameters (features) of weekly opening price, weekly highs, weekly lows, and weekly closing prices, comparing the moving averages trained over the 4 week period, to the actual opening price of the following week. Ordinary least squares regression was used to determine the relative estimated effect of each moving average on the opening price of the following week. Our analysis revealed significant multicollinearity, suggesting that these variables were not independent. While each moving average taken collectively tracked closely around the opening stock price of the following week, none of these features were significant predictors of the actual opening price. There was a trend to a positive correlation between weekly high price and the next opening price, as well as a negative correlation between weekly low price and the next opening price, but neither of these achieved statistical significance with α < 0.05. The moving averages of opening and closing prices from the week before showed an even weaker correlation with the opening price the following week. Some of the factors behind these observations are discussed.

**Introduction**

Forecasting the direction of stock market prices has been a perennial challenge for mathematicians, data scientists and many others. Many still look to technical analysis to guide investment decisions. As such, we sought to explore one of the many promising data analytic techniques to gain insight into identifying market trends and the features that are most influential in creating a predictive model. We understand that there are many variables that play a pivotal role in determining daily market prices for equities, such as political events, global economic decisions, natural disasters, but also human factors, such as data from the Bureau of Labor Statistics and the Bureau of Economic Analysis, and the Federal Reserve. Other factors, not intrinsic to the company itself, can play a moderating role in the value of the market performance of a company, such as its ability to compete effectively against competitors in its sector, balancing the needs of growth and value.

As such, we felt that a company's market price is a very reasonable target parameter (ŷ) upon which to develop a model to make short-term predictions on market performance. We understand that making long-term predictions of stock prices (i.e., in terms of months) is as imprecise as making long-term weather predictions. However, by restricting the window of prediction to a week, it might be possible to determine, with reasonable accuracy, what features may predict short-term market price change.

**Data Cleaning / Preparation**

A dataset of the Dow30 Industrials Index (Brown, 2014), obtained from the University of California, Irvine, contains the stock market prices from the following companies: AA, AXP, BA, BAC, CAT, CSCO, CVX, DD, DIS, GE, HD, HPQ, IBM, INTC, JNJ, JPM, KO, KRFT, MCD, MMM, MRK, MSFT, PFE, PG, T, TRV, UTX, VZ, WMT and XOM (See Appendix 1 for stock names). The data were restricted to the period 1/14/2011 to 6/24/2011, and 24 weekly datapoints for each company were present. For each company, the following numbers were provided: opening price (Monday morning each week), weekly high price, weekly low price, weekly closing price (end of day Friday), volume of trades, % price change in the week, % volume over last week, previous week volume, next week's opening price (Monday morning), next week's closing price (end of day Friday), % change next week's price, days to next dividend, % return at next dividend.

To perform this analysis, we selected nine of the fifteen features as candidate features for our analysis.  These parameters are:

1. yearly quarter (i.e., 1Q, 2Q, etc.)
2. opening price at beginning of the week (Monday, opening bell)
3. highest price during the week
4. lowest price during the week
5. closing price at the end of the week (Friday, closing bell)
6. volume of shares traded during the week
7. previous week's volume
8. percent change in volume compared with last week
9. percent change in price throughout the week.

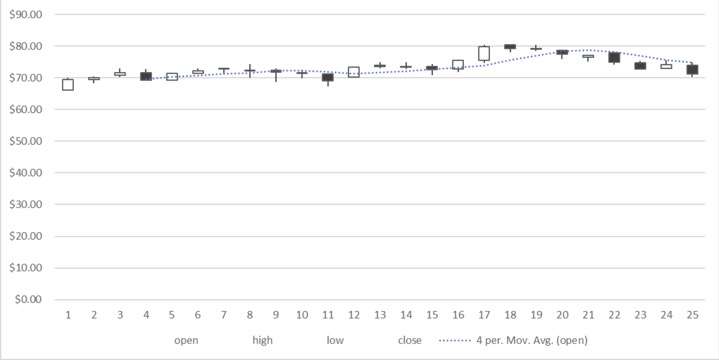
These features were used to predict one of the following values determine for the following week:

1. Next week's opening price
2. Next week's closing price
3. Percent chance in next week's price compared with that of the previous week.
4. Percent return in the next dividend (percentage return)

The dataset was pre-processed by removing rows with missing values. Without exception, these were the rows at the beginning of each company's collection, where much of the data on the previous week's parameters were missing. We did not find any NaN datapoints in our dataset. Since each company's values ranged within 10% of the mean price, we felt that there was no reason to normalize the data point to be centered with a μ of 0, as this would not impact the accuracy of the moving average calculations. As such, the effort made to preprocess the data was minimal.

**Exploratory Data Analysis**

In visualizing the dataset, it is clear that each stock's price graph over a month time period is irregular, and therefore, use of simple linear regression would not be appropriate. The distribution of individual stock prices around a weekly mean is also not expected to have an underlying normal distribution, as the *x*-axis establishes a floor for stock value, while there is no intrinsic upper limit. Furthermore ask/bid distribution is not likely to be uniform for each stock and may even depend on which exchange the stock is listed with. Daily high-low prices vary widely, and a glance at a candlestick chart makes it evident that there is also no expectation of a normal distribution around opening, closing, high or low prices.

Fig 1. Example 4-week moving average line plotting on a candlestick chart for the company BA.

**Model Selection**

As such, many machine learning approaches to training on stock market data use time-sequence analysis algorithms since the underlying dataset is time-sequenced. These techniques include time-sequenced regression and ARIMA models. Some of these models assume linear relationships with other parameters or involve complex mathematics, and the advantage of the moving average is that the underlying concepts are straightforward and do not require heavy computational demands. Thus, using moving average data to predict forward datapoints has been a favored technique (Biswal, 2023; Sharma, 2023; Vijh, 2019, Chandola, 2022). The moving average combines the latest price point with prior data points, and determines a non-weighted (simple) or weighted (exponential) average to predict the next data point (Hyndman, 2018). For our analysis, we selected the simple moving average model, which did not confer additional weight to more current datapoints.

Our strategy was to plot a moving average of various parameters of the data and determine which of these are most closely associated with next week's opening price. Machine-learning algorithms are used to evaluate moving averages often using LSTM models, which were developed for analysis in natural language processing. As such, these models lend themselves well to the time-sequenced nature of stock market prices (Li, 2023). We had explored using Keras/Tensorflow for this analysis, but the datasets are not large enough to allow partitioning of the limited dataset into training and test sets for meaningful data analysis.

For our project, we therefore decided to use the moving average functionality in Microsoft Excel, a spreadsheet software application produced by Microsoft Corp, Redmond, WA (Microsoft Excel for Microsoft 365 MSO (Version 2305 Build 16.0.16501.20074, 64-bit, running on Windows 11 Home Version 22H2)). It offers a comprehensive set of statistical tools which can be applied to data entered into the spreadsheet. Later versions have allowed the addition of moving averages charting as part of the options in formatting trendlines. Forecasting can be selected as well, and may be useful to suggest near-term predictions that we can select as ŷ to compare with the actual data we have available in the dataset (ȳ).

Excel allows for averaging the last *n* datapoints in calculating the moving average, and in our project, we decided to use 4 periods in the calculation, which we felt was the best balance between under- and over-smoothing.

Microsoft Excel does not offer ordinary least squares (OLS) analysis without the addition of statistics add-on packages, and therefore we used the statsmodels library (version 0.14.0) in a Jupyter Lab (version 4.0.0) environment, using Python 3.11.4, Numpy 1.25, Matplotlib 3.7, and Pandas 2.0.2.

Each company was evaluated separately, and a 4-week moving average was calculated for each weekly time-point. Once the analysis was performed, a prediction was made from this analysis and this figure was compared with the feature set of the final week, using multilinear regression. The goal was to determine which feature was most closely associated with the prices of the following week (and which were least correlated).

**Model Analysis**

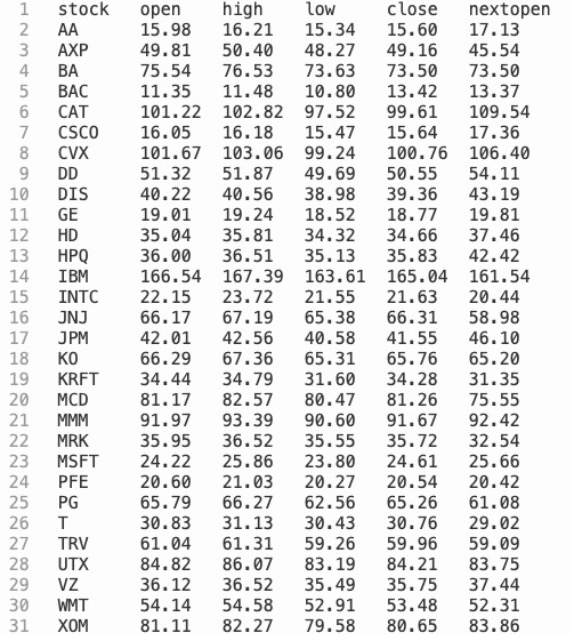
Using Microsoft Excel, moving average values were calculated for each stock. For each stock, the 4-week moving average data from the penultimate week were calculated (ŷ) and compared with the actual opening price of the ultimate week given in the dataset (ȳ), to see how closely the figures correlated with the actual figure.

Fig 2. Table of stock prices, moving average prices of open, high, low, close data, compared with next opening price.

Scatterplots were created to visualize the data distribution to look for irregularities, outliers and to set expectations.

A picture containing text, line, plot, diagram

Description automatically generated

Fig 3. Opening price at the end of the dataset, as correlated with the penultimate moving average opening price for each stock.

A picture containing text, line, plot, diagram

Description automatically generated

Fig 4. Opening price at the end of the dataset, as correlated with the penultimate moving average weekly low price for each stock.

A picture containing text, line, plot, diagram

Description automatically generated

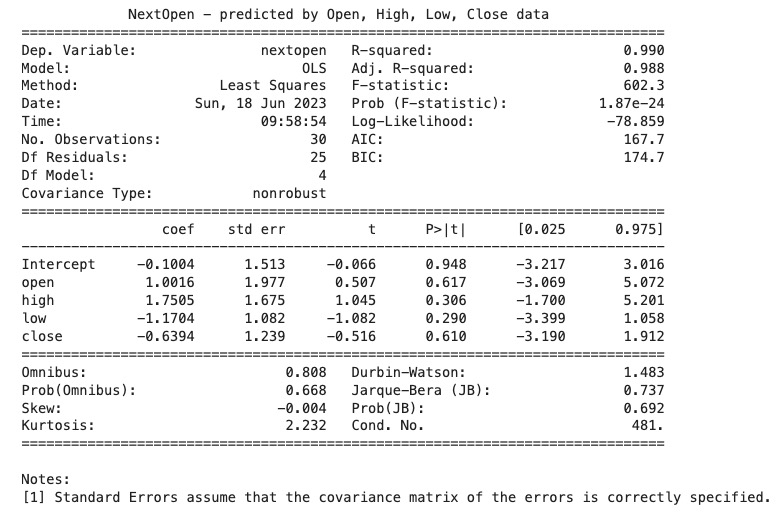
Fig 5. Opening price at the end of the dataset, as correlated with the penultimate moving average weekly high price for each stock.

A picture containing text, line, diagram, plot

Description automatically generated

Fig 6. Opening price at the end of the dataset, as correlated with the penultimate moving average closing price for each stock.

Using the Python statistical package, statsmodels, an ordinary least squares multivariable analysis was conducted to determine the relationship between the four variables selected, and the opening price for the last week in our dataset for which we could validate our predictions.

The R-squared value was 0.990 which was essentially identical to the adjusted R-squared value of 0.988, suggesting that there was negligible external variance factor to consider. The F-statistic was very high, with an F-statistic probability of 1.87e-24, indicating a good fit to the data. However, the condition number was also high at 481, suggesting a high degree of multicollinearity, and that our variables share some dependency. Of the individual features, none achieved statistical significance as being correlated to the next opening price, and the features with the best association were the weekly high price (a positive correlation) and the low weekly price (a negative correlation). For the opening and closing prices of the week, the P-valued were 0.6, suggesting that the probability of denying H0 was not far from that of a coin flip.

**Conclusion and Recommendations**

Based on our analysis, we were not able to identify an association between a stock's open, close, weekly high or low moving average over the prior four weeks, as a dependable predictor of the opening price going forward. This may be due to factors present during the weekend that affect the price of the stock on the next opening day, as can happen with major economic or global events. It may be that a more accurate predictor of near-term price targets would derive from incorporating longer periods into the moving average calculation. Decreasing the number of periods in this calculation would lessen its utility however, and would essentially render it indistinguishable from just looking at Friday's closing price, or the unadjusted weekly high/low data. We only examined the stocks that are part of the Dow Jones stocks, and an analysis of a larger number of corporate data might yield factors better suited as short-term predictors of stock price.

**References**

Brown, M. (2014). Dow Jones Index. UCI Machine Learning Repository. https://doi.org/10.24432/C5788V.

Biswal, A. (4 April 2023). Stock Price Prediction Using Machine Learning: An Easy Guide! https://www.simplilearn.com/tutorials/machine-learning-tutorial/stock-price-prediction-using-machine-learning

Sharma, P. (26 April 2023). Machine Learning for Stock Market Prediction With Step-by-Step Implementation. <https://www.analyticsvidhya.com/blog/2021/10/machine-learning-for-stock-market-prediction-with-step-by-step-implementation/>

Singh, A. (2 May 2023). Stock Prices Prediction Using Machine Learning and Deep Learning. <https://www.analyticsvidhya.com/blog/2018/10/predicting-stock-price-machine-learningnd-deep-learning-techniques-python/>

Vijh, M, Chandola, D., Tikkiwal, V.A, Kumar, A. (2020). Stock closing price prediction using machine learning techniques. Int. Conf. Comp. Intellig. Data Sci. 167:599-606. <https://www.sciencedirect.com/science/article/pii/S1877050920307924>

Hyndman, R.J., & Athanasopoulos, G. (2018) *Forecasting: principles and practice*, 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2. Accessed on 17 Jun 2023. <https://otexts.com/fpp2/moving-averages.html>

Li, K (2023) Predicting stock prices using machine learning. Accessed on 17 Jun 2023. https://neptune.ai/blog/predicting-stock-prices-using-machine-learning

Cheusheva, S. (21 Mar 2023). Calculating moving average in Excel. Accessed on 17 Jun 2023. <https://www.ablebits.com/office-addins-blog/moving-average-excel/>

**Appendix 1 – Stock Names**

AA Alcoa

AXP American Express

BA Boeing

BAC Bank of America

CAT Caterpillar

CSCO Cisco Systems

CVX Chevron

DD DuPont de Nemours

DIS Disney Corp.

GE General Electric

HD The Home Depot

HPQ Hewlett-Packard

IBM International Business Machines

INTC Intel

JNJ Johnson and Johnson

JPM J. P. Morgan

KO Coca-Cola Company

KRFT Kraft

MCD McDonald's Corporation

MMM 3M Company

MRK Merck

MSFT Microsoft

PFE Pfizer

PG Proctor and Gamble

T AT & T

TRV The Travelers Companies

UTX United Technologies

VZ Verizon

WMT Walmart

XOM Exxon-Mobil

Appendix 2 – Python Code

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import scipy.stats as stats

import statistics

import statsmodels.api as sm

import statsmodels.formula.api as smf

dow = pd.read\_csv("Dow2.dat", sep="\s+") # Read the dataset into pandas

dow\_model = smf.ols(formula="nextopen ~ open + high + low + close", data=dow).fit() # run the OLS

print(dow\_model.summary(title="NextOpen - predicted by Open, High, Low, Close data")) # print summary

print(dow\_model.params)

plt.figure(figsize=(12,12), dpi=100) # set graph size

for i in range(len(dow)): # FOR loop to print scatterplot with annontations

plt.scatter(dow['open'][i],dow['nextopen'][i], marker='o', color='#0077dd', alpha=1.0, s=30)

plt.annotate(dow['stock'][i], (dow['open'][i],dow['nextopen'][i]), xytext=(6, -5), textcoords='offset points')

plt.xlabel("Weekly Opening Price Moving Average, as of previous week")

plt.ylabel("Opening price, end of dataset")

plt.title("Relationship of weekly opening price moving average to Opening Price at end of dataset")

plt.show()

plt.figure(figsize=(12,12), dpi=100)

for i in range(len(dow)):

plt.scatter(dow['high'][i],dow['nextopen'][i], marker='o', color='#0077dd', alpha=1.0, s=30)

plt.annotate(dow['stock'][i], (dow['high'][i],dow['nextopen'][i]), xytext=(6, -5), textcoords='offset points')

plt.xlabel("Weekly High Price Moving Average, as of previous week")

plt.ylabel("Opening price, end of dataset")

plt.title("Relationship of weekly high price moving average to Opening Price at end of dataset")

plt.show()

plt.figure(figsize=(12,12), dpi=100)

for i in range(len(dow)):

plt.scatter(dow['low'][i],dow['nextopen'][i], marker='o', color='#0077dd', alpha=1.0, s=30)

plt.annotate(dow['stock'][i], (dow['low'][i],dow['nextopen'][i]), xytext=(6, -5), textcoords='offset points')

plt.xlabel("Weekly Low Price Moving Average, as of previous week")

plt.ylabel("Opening price, end of dataset")

plt.title("Relationship of low price moving average to Opening Price at end of dataset")

plt.show()

plt.figure(figsize=(12,12), dpi=100)

for i in range(len(dow)):

plt.scatter(dow['close'][i],dow['nextopen'][i], marker='o', color='#0077dd', alpha=1.0, s=30)

plt.annotate(dow['stock'][i], (dow['close'][i],dow['nextopen'][i]), xytext=(6, -5), textcoords='offset points')

plt.xlabel("Weekly Closing Price Moving Average, as of previous week")

plt.ylabel("Opening price, end of dataset")

plt.title("Relationship of weekly closing price moving average to Opening Price at end of dataset")

plt.show()