May 1, 2025

C964: Computer Science Capstone

Task 2 parts A, B, C and D

Part A: Letter of Transmittal

May 1st, 2025

Senior Leadership

ES Financial Services

Dear Senior Leadership,

I am writing with the intent of proposing a new machine-learning solution that would be of benefit to your company. The main premise is that the machine learning program will be able to aid in forecasting the movements of companies in the S&P 500. With the continual improvement in technology, it is crucial to develop systems that would benefit when it comes to gaining an advantage in the stock market. Continual reliance on older methods, such as trend analysis and financial modeling, could lead to missing out on pivotal market movements.

To provide ease of use, I am proposing a web-based application with a simple, intuitive UI design that utilizes a supervised machine learning algorithm to predict the closing price of any company in the S&P 500. Using the new web application, its users can select the ticker symbol and the number of days of historical data used to make the prediction. After the stock ticker and the number of days are selected and the prediction is made, three graphs will be displayed alongside the prediction. These graphs will aid in continuing the analysis and help visualize how the prediction was made. The data required to power this new machine-learning algorithm can be found on the Kaggle website, making building the training set a simple endeavor. The organizational benefits of this new web application include increased confidence in decision-making and faster predictions. With increased confidence in decision-making, the stakeholder impact will be an increased revenue stream, and missed opportunities from prior systems will be lower.

The development timeframe will be about a month, with phases dedicated to data preparation, development, and testing. For the development of this project, I propose utilizing the CRISP-DM framework to plan the project due to its suitability when it comes to machine learning algorithm development. The total cost for this project will be $299,330 and is comprised of hardware, software, labor, and environment costs. This project has no ethical concerns as the sourced data has a public domain license, and all information used and derived cannot be connected to any individuals. All aspects that involve this project are fields that I have relevant expertise in, such as CRSIP-DM and supervised machine learning algorithms that will aid in speeding up the development of this project. In addition, the entire program will be built using Python and the Streamlit framework, both of which I have prior experience with.

Sincerely,

****

********, Developer

Part B: Project Proposal Plan

Project Summary

It is crucial for financial companies to use all their means to have an advantage when trading in the stock market. A quick and efficient program that can predict prices can be the difference between making or losing money. For ES Financial Services, ensuring they can make fast and accurate predictions is of the utmost importance. This is why I am proposing a web application that utilizes a supervised machine learning algorithm such as linear regression to make predictions. After all the data preparation and development, the planned deliverables include a web application and a user guide with instructions on how to get the application up and running. The web application will aid ES Financial Services by having an intuitive UI that can make predictions fast and offers the ability to select the number of days of financial data to use in making the predictions.

Data Summary

The original data source that contains the historical data of the S&P 500 from 2010 to 2024 comes from Kaggle, a website that hosts datasets created by people. Once the data is loaded into Python using Pandas, it is processed by removing all rows that contain missing data and ensuring that the data is organized by date. The data collected from the Kaggle dataset meets the project's needs because it includes 14 years of accurate financial data to train the model. There are no legal concerns about using this data because the dataset on Kaggle uses the CC0: Public Domain license. This allows for the commercial usage of the dataset with no repercussions to worry about.

Implementation

The industry-standard methodology for this project will be CRISP-DM as it is a respected framework used for developing machine learning algorithms. CRISP-DM is split into six phases: business understanding, data understanding, data preparation, modeling, evaluation, and deployment. The implementation plan for this project using CRISP-DM will begin with a business understanding that the issue to be identified is predicting S&P 500 company closing stock prices. The data understanding and preparation phase is where the data is loaded and cleaned of missing data, after which the graphs used to accompany the web application are created. The linear regression algorithm is trained with split data in the modeling phase. In the evaluation phase, the R^2 score and the squared mean error metrics are used to validate the predictions. The UI is pushed to the web using Streamlit share in the deployment phase.

Timeline

Milestone or deliverable	**Project Dependencies**	**Resources**	**Start and End Date**	**Duration**
Determining Project Scope	NA	Stakeholders	5/1 – 5/3	3 Days

Data Collection and Processing	Project Scope	Python, Pandas, Kaggle	5/4 – 5/5	1 Day
Graph Creation	Data Collection Data feature Creation	matplotlib, seaborn	5/5 – 5/6	1 Day
Data feature creation	Data Collection	Pandas, Python	5/6 - 5/6	1 Hour
Test Data split creation	Data feature creation	Pandas, Python	5/6 – 5/7	1 Day
Model training	Data feature creation Data Collection Project Scope	scikit-learn, Linear Regression	5/7 – 5/9	3 Days
Model Validation	Model Training	scikit-learn, Metrics	5/10 – 5/11	2 Days
Application Interface Development	Graph Creation Project Scope	Streamlit	5/12 – 5/17	5 Days
User Guide and Documentation	Application Interface Development	Word	5/17 – 5/23	7 Days
Deployment	All Prior milestones	Github, Streamlit Share	5/23 – 5/24	1 Day

The timeline of this project is about 3 weeks and contains about 10 deliverables/milestones. The first step is the creation of the project scope, which will determine the goals of the entire project. Afterwards, the data collection, graph creation, and data feature creation can be worked on in tandem but must be done sequentially as they depend on one another. After these steps are accomplished, the creation of the test data split can commence, and preparations for training the linear regression model can begin. Once the model is trained, it must then be validated to ensure it properly runs and makes predictions that are accurate according to scikit-learn metrics such as R^2 and Mean Square Error. Once the model training has been finalized, the application user interface can be developed using the Streamlit framework. User interactive features that allow the user to customize their prediction, as well as the graphs, would then be displayed on the web page. A user guide that provides two different ways to access the app will be created to aid user adoption. The finalized prediction model and user interface are to be then deployed through the Streamlit share platform.

Evaluation Plan

While the data is being prepared, a validation method used to verify data integrity is removing rows containing missing data. During the feature value creation, the values are to be inspected, and their accuracy with the data is verified. During model training, a test split is to be created using data that is sorted by date to ensure that data is trained in chronological data. For post-development validation, the validation method to be used is the R^2 score and the mean squared error. The R^2 score will measure how well the data being used to making the prediction fits. The mean squared error result will measure the difference between the model prediction and the actual data when using the training split.

Resources and Costs

Hardware Cost

Item	Description	Cost
Developer Desktop		$2000
Developer Laptop		$1500
Monitor		$200
Peripherals	Mouse, keyboard, etc.	$120

Software Cost

PyCharm Professional	Development Environment	$249 per year

Python	Programming Language	Free
Streamlit	Python framwork	Free

Labor Costs

Machine Learning Engineer	Model Development	$120,000 per year
Project Manager	Project Manager	$80,000 per year
QA Engineer	Testing and Validation	$95,000 per year

Environment Costs

AWS	Hosting	200 per year
Custom Domain name		10 per year
Maintenance		300 per year

Total Cost = $299,330

Part C: Application

Submitted Files

- stock_price_model.py
- sp500_stocks.csv
- requirements.txt
- Website Link: https://stockpricemodelpy.streamlit.app/

Part D: Post-implementation Report

Solution Summary

ES Financial Services was searching to utilize the latest technology to aid in predicting the closing price of companies in the S&P 500. The solution to this was to use the historical data of the S&P 500 to feed a supervised machine learning algorithm and use linear regression to predict the movements of companies in the S&P 500. The application that was built allows the user to select the ticker from all companies in the S&P 500 and make a prediction on whether a stock should be bought or shorted.

Data Summary

The historical data of the S&P 500 is collected from a Kaggle dataset. For the design phase important data columns from the dataset such as close, high, low, open, and volume were selected to be used as well as derived data such as change and average close. During the development phase the data was then put into a Pandas data frame and the data was cleaned. For example, all rows that had missing values were excluded from the training set. The data was then filtered based on the ticker symbol to be selected, and a linear regression prediction was made. For the maintenance phase, the machine learning algorithm's visuals can be used to monitor the algorithm's performance.

Machine Learning

The web application allows the user to select a ticker symbol of a company from the S&P 500 and how many days of historical market data they wish to use to predict what the company share value will close on. The machine learning algorithm uses historical market data metrics such as close, high, low, open, and volume to train the supervised learning algorithm to make a prediction. The libraries used include Pandas and Scikit-learn. Pandas were used to read from the CSV file and put the data into a data frame. The Scikit-learn library was used to access the linear regression model and make predictions.

After the data was extracted from the CSV file, it was cleaned and processed to be used to train the model. This included removing missing data rows and making sure the data was sorted by date. After the important data to be used to train the model is selected, new data derived from existing data is created, such as change and average close which would improve the model prediction accuracy. Linear regression is then used to train the model and make the necessary prediction.

Given the nature of the data being historical data and the goal to predict based on that data, a supervised learning algorithm was selected. The linear regression algorithm was chosen because it is well suited for the pattern recognition necessary to make a prediction. The metrics used to train the model were selected because they are the common indicators used to determine stock's desirability. New metrics were created, such as change and average close, to reveal trends further and aid the algorithm in making a prediction.

Validation

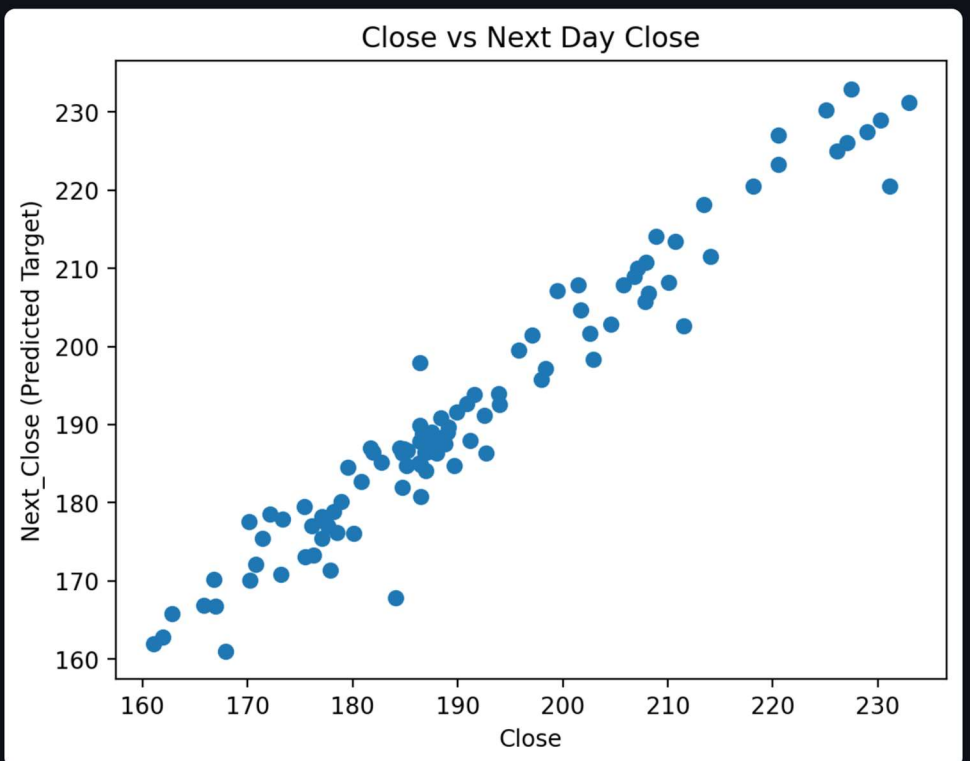
The results of the linear regression algorithm's prediction were validated using the Scikit-learn library's R^2 Score and mean square error. The R^2 score will measure how well the data is used to make the

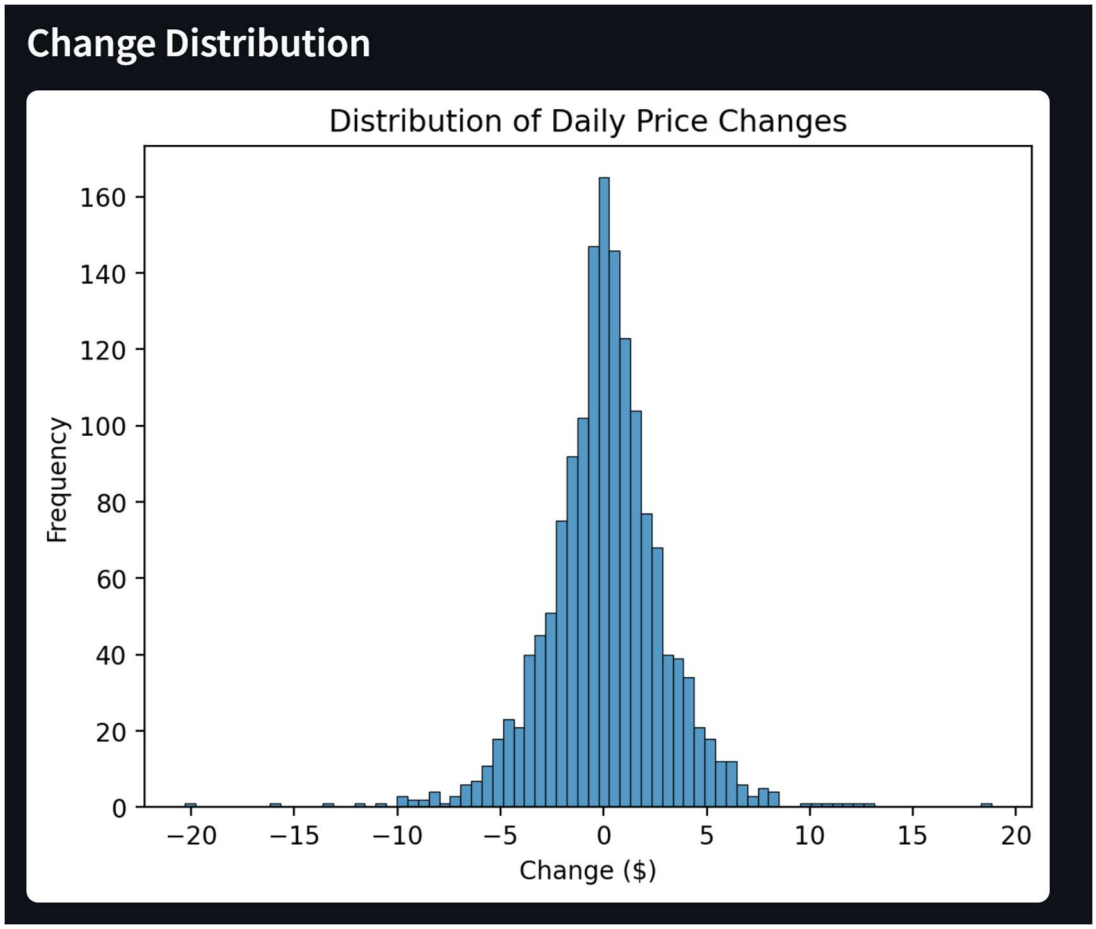
prediction fit. The mean square error measures how well the predictions match the actual values of the data. When using the ticker F, the R^2 score gave 92%, and the mean square error gave 0.08, signaling that the data used to make the predictions and the predictions themselves are accurate.

Visualizations

The three unique visualizations are located on the website after a ticker is selected alongside the number of days of historical data to be used.

Close vs Next Closing Price





User Guide

User Guide (Recommended)

1. Visit the website https://stockpricemodelpy.streamlit.app/
2. If the website has been inactive for too long simply select the option to wake up the app again and wait for it to initialize
3. Select a stock symbol from the drop-down menu
4. Use the slider to select the number of days to be used in the prediction
5. Scroll down to see the visualizations
6. Scroll down further to see the validation method values
7. Scroll all the way to the bottom to see the prediction value and the latest stock values.

User Guide (Local Setup)

1. Install Python
2. Install the necessary libraries streamlit, pandas, matplotlib, seaborn, scikit-learn
3. Open the terminal and clone the repository from GitHub

    a. git clone https://github.com/elijsanch/CapstoneProject.git
4. Navigate to the cloned folder
    a. cd CapstoneProject
5. Run the application
    a. streamlit run stock_price_model.py
6. Your browser should open automatically but if not then navigate to http://localhost:8501 on your browser.
7. Select a stock symbol from the drop-down menu
8. Use the slider to select the number of days to be used in the prediction
9. Scroll down to see the visualizations
10. Scroll down further to see the validation method values
11. Scroll all the way to the bottom to see the prediction value and the latest stock values.

An example of what the website should look like can be seen below.