Define problem statement

Identify Data Sources

Clean Data

Discuss results and analysis

Implement ML algorithm(s)

Identify one or more ML algorithms

Feature selection

Conclusion

Data Cleaning steps:

S&P data:

* Take S&P 500 Historical data\_Start till 2018\_csv from datahub.io \cite{website12}

This gives S&P500 data till 2018. Has 1768 rows

Attributes: Data, SP500, CPI, Long Interest Rate

* Take S&P 500 Historical Data.xls file from Investing.com \cite{website11}

Sort the file by date from old to new. Took 59 records from this data set

Take S&P500 data from 5/1/2018 till 2023 from this file

Attributes: Price

* Random sampling to check the consistency of between data sets. The Price in the First data set lies within the High and Low prices in the second data set for each month. Hence good to use

GDP data:

* Take GDP data set from Fed \cite{website13}. This gives monthly GDP from 1992

Attribute: Nominal GDP

* Change all date formats on all Data sets to be consistent MM/DD/YYYY
* Take Pre 1992 GDP data from the website \cite{website14}

Change the Date format on this data set to match the date format on the Consolidated Data set. YYYY-MM-DD is changed to MM/DD/YYYY

Do Vlook up on date from the Consolidated Data set and obtain the GDP numbers from this data set

* Random sampling to check the consistency of between data sets. The Price in the First data set is pretty close to that from the second data set for each month. Hence good to use

Fed rate:

* Take the Fedfunds rate data from \cite{website06}

Do Vlook up on date from the Consolidated Data set and obtain the Fed rate numbers from this data set

Inflation data:

* Use the Historical Inflation data obtained from BLS
* This data needs significant transformation

Unemployment data:

* Use the Historical Inflation data obtained from BLS
* This data needs significant transformation

MACD:

* Plug in the Data and SP500 values from the Consolidated Data set into the MACD template
* Do Vlook up on date from the Consolidated Data set and obtain the MACD numbers from this data set

RSI:

* Plug in the Data and SP500 values from the Consolidated Data set into the RSI template
* Do Vlook up on date from the Consolidated Data set and obtain the RSI numbers from this data set

=IF(A2="Jan",1,IF(A2="Feb",2,IF(A2="Mar",3,IF(A2="Apr",4,IF(A2="May"=5,IF(A2="Jun",6,IF(A2="Jul",7,IF(A2="Aug",8,IF(A2="Sep",9,IF(A2="Oct",10,IF(A2="Nov",11,IF(A2="Dec",12,""))))))))))))

**Data Cleaning:**

Missing Data: Deleted rows from 1871 to 1957 because no data was available for GDP, Fed rate, Inflation and Unemployment. Not even enough data to fill in missing values. Left with 783 rows

Fed funds rate for 3/1/23 was missing. However, Fed announced a rate hike of 0.25% on 3/22/23. So added .25 to the previous value from 2/1/23

Inflation rate for 3/1/23 was missing. Inflation has remained between 5.5-6% for the past couple of months. Hence, an average of the past four month inflation rates would be a close approximation for the Inflation rate for March 2023.

Unemployment data hasn’t been published rate for 3/1/23.

Unemployment rate has remained between 3.5 – 3.7% for the past couple of months. Hence, an average of the past four month inflation rates would be a close approximation for the Inflation rate for March 2023.

GDP numbers for the initial years are available for each quarter.

Month 1 GDP = GDP1

Month 2 GDP = Not available

Month 3 GDP = Not avai

Month 4 GDP = GDP4

Month 2 GDP calculated as GDP1 + (GDP4-GDP1)/3

Month 3 GDP calculated as GDP1 + 2\*(GDP4-GDP1)/3

This gives a good approximation of the missing GDP values

Transformation: Date formats for each data set had to be transformed

Merge/Consolidate similar columns:

* Multiple columns with SP500 data is consolidated into one column
* Multiple columns of GDP data is consolidated into one column
* Multiple columns of Fed funds data is consolidated into one column

Merge the 12 CSV files into one consolidated CSV

1827 records

Standardize data in each data set. Sort records by date and format date in mm/dd/yyyy format

12 CSV files

Cleaned Data set with 783 rows

Fill in missing values in GDP column using formula

Remove rows from 1/1/1871 to 1/1/1958

**Exploratory Data Analysis:**

**Part 1**

1. What is exploratory data analysis? Why is it essential in a data science or data analytics project?

Exploratory Data Analysis (EDA) is data exploration technique to understand various aspects of data. IT is a process of investigating dataset to uncover patterns, anomalies etc and understand the dataset better. The relationship between the data variables is understood through EDA. Insights are gathered about the data during this phase before moving on to more complex processes such as using algorithms/ predictive modelling etc in the Data analysis/ Data science life cycle. This is a method used to analyze and summarize data sets

EDA helps in identifying faulty points in Data even after the Data cleaning phase. Data can be further cleaned during this process with a true understanding of the relationship between variables. Which gives a true understanding of data

1. Understand the data (variables, number of columns) etc
2. Clean the data -> Already done in the previous phase
3. Analyze relationship between variables
4. What are the various exploratory data analysis techniques? What specific techniques work for your project?
   1. Clustering and dimension reduction techniques
   2. Univariate visualization of each field in the raw dataset
   3. Bivariate visualizations and summary statistics to assess the relation between the dependent and independent variables
   4. Multivariate visualizations to understand interactions between different fields in the data
   5. K-means clustering: Data points are assigned into K groups. The data points closest to a particular centroid will be clustered under the same category. This method is used in market segmentation, pattern recognition, image compression
   6. Predictive models, such as linear regression and statistics

Techniques mentioned in #2, #3, #4 seem to be applicable for my project

1. Explain the details of the techniques and the results of your exploratory analysis. (You may include a diagram to explain your exploratory analysis phase.)
   1. Imported the necessary libraries (numpy, pandas, matplotlib, seaborn) and read the data set into a data frame
   2. Displayed the head (first 5 rows)
   3. Used df.shape to display the total rows and columns of the data frame (783 rows and 8 columns)
   4. Displayed the data types of the columns using df.info()
   5. Converted the Date format from object to date using

df["Date"] = pd.to\_datetime(df["Date"])

Now all the columns are in float format

* 1. Described the data (Counts, Median, Max, Min etc) using df.describe()

Table

Description automatically generated

* 1. Checked for Null values using df.isnull().sum()

No data is null. So concluded that the data is clean

Text

Description automatically generated

* 1. Checked for Unique values using df.nunique()

Most rows are unique

Text, letter

Description automatically generated

* 1. **Univariate visualizations:** 
     1. Generated Histograms for each variable to understand distribution for data
     2. Generated Box plots for each variable
  2. **Bivariate visualizations:** Generated Scatter plots for each independent variable on X axis and the dependent variable (SP500) on the Y axis
  3. **Multivariate Visualizations**:
     1. Relationship analysis: Did correlation analysis and generated Heat maps

corelation = df.corr()

sns.heatmap(corelation, xticklabels=corelation.columns, yticklabels=corelation.columns, annot=True)

Chart

Description automatically generated

This shows that that there is not much correlation between the independent variables. So all the columns can be used and no need to delete any

* + 1. Generated Pair plots

sns.pairplot(df)

Diagram, engineering drawing

Description automatically generated

1. What insights are you learning from this particular phase?

I began getting a vision of the process/path towards the end goal. The data which was all numbers during the earlier stages is beginning to make sense. I got a picture of the boundaries of each variable, what values for each value seem normal (more frequently occurring), what values are extreme on both sides etc. I got a good understanding of the relationship between the variables. Some relationships are intuitive but some relationships are not, like the relationship between GDP and Inflation, Inflation and unemployment, Feds rate and unemployment etc. Overall, the exploratory data analysis phase gave me an idea of how to proceed next

**Part 2**

1. Clickable link to your GitHub repo: <https://github.com/pamidisushma02/Sushma_SP500Prediction_Capstone_Project>
2. Clickable link to your Overleaf report: <https://www.overleaf.com/read/jpdsbxmjtyvq>

**Predictive Modelling:**

RNN, LSTM

<https://www.youtube.com/watch?v=lncoLfue_Y4>

Linear Regression

Logistics Regression

Decision Tree

Random Forest

Xgboost

Stochastic Gradient Boosting

KNN

SVM

Naïve Bayes

k-means clustering

Deep Neural Networks

**Part 1**

1. What mechanism or a pipeline is are you using to accomplish a predictive application? You may use a diagram to explain your mechanism. Cite your source or self as author.

Here is a general high-level diagram of a machine learning pipeline that could be used for building predictive applications:

Import selected Models

Select a model or multiple models

Split data into train and test data

Cleaned Data set from Data Curation step

Train/ Fit model to training data

Summarize & compare results/predictions for all models with actual data

Evaluate model(s) by capturing metrics (MSE, MAE, R Square)

Predict Target variable using test data for the models

This pipeline typically consists of the following steps:

1. Split the data: Use the data set obtained from the Data curation step and divide the data into training and testing sets. The training set is used to build the model, while the testing set is used to evaluate the model's performance.
2. Select a model: Choose a model that is appropriate. I will be using multiple models as I was not able to arrive at one particular model from the Data exploration phase.
3. Train the model: Fit the model to the training data using the chosen models
4. Evaluate the model: Evaluate the model's performance on the testing set. This may involve using metrics such as accuracy, precision, recall, and F1 score.
5. Predict Target variable: Once we have a final model that performs well on the testing data, we will use it to make predictions on testing data. However, I plan to predict using all the chosen models
6. Summarize results and compare models: Summarize & compare results/predictions for all models with actual data
7. What are machine learning algorithms used to analyze your data?

Algorithm is a step by step process/instructions to solve a program using a Machine/Computer. Machine learning algorithms are mathematical models that learn from data and make predictions or decisions based on that learning. In other words, machine learning algorithms are the core components of machine learning systems, which enable computers to learn and improve from experience without being explicitly programmed.

The Target data for my project is a continuous variable and hence the below Algorithms will be used to analyze the data

* 1. Linear regression
  2. Decision trees
  3. Random forests
  4. k-nearest neighbors
  5. Support vector machines
  6. Deep neural networks
  7. XGBoost
  8. Stochastic Gradient Boosting (SGB)

1. Explain the training and testing process related to your data?

The training and testing process is a crucial part of the machine learning workflow, which involves building a model that can learn from data and make accurate predictions on new data. Here's how the training and testing process works in machine learning:

1. Data Preparation: The first step is to collect and preprocess the data to be used for training and testing the model. This includes cleaning the data, removing any missing values etc. This step was already performed during Data curation phase
2. Data Splitting: The next step is to split the data into two parts: a training set and a testing set. The training set is used to train the model, while the testing set is used to evaluate the performance of the model.

Graphical user interface, text, application, email

Description automatically generated

1. Model Training: The model is trained on the training set using a selected algorithm or technique. During training, the model learns from the input data and updates its parameters to improve its predictions.
2. Model Testing: After the model is trained, it is evaluated on the testing set to measure its performance. This involves making predictions on the test set and comparing them with the actual values to calculate the model's accuracy.
3. Model Evaluation: The performance of the model is evaluated based on various metrics such as Mean Squared Error, Mean Absolute Error, R Square

In summary, the training and testing process involves preparing the data, splitting it into training and testing sets, training the model on the training set and evaluating the model's performance on the testing set. This process is critical to building accurate and reliable machine learning models that can make accurate predictions on new data.

1. Explain the implementation and evaluation process of your analysis. Ensure you include GitHub repository URLs of your source code in the references.

Implementation starts with splitting the Data set into training (20%) and testing (80%) data

1. What are the results of your analysis, and display them in an appropriate format?

Below are the results of my analysis:

Of all the Models tested, Stochastic Gradient Boosting (SGB) and K-nearest neighbors seem to be performing considerably well. The MSE and MAE for these models is low compared to others. And the R-Square value is very high > 99.8% suggesting these models fit the data well

Table

Description automatically generated

**Part 2**

1. Clickable link to your GitHub repo: <https://github.com/pamidisushma02/Sushma_SP500Prediction_Capstone_Project>
2. Clickable link to your Overleaf report: <https://www.overleaf.com/read/jpdsbxmjtyvq>

# Support Vector Machines

from sklearn.svm import SVR

from sklearn.metrics import mean\_squared\_error, r2\_score, mean\_absolute\_error

from sklearn.model\_selection import train\_test\_split

# Splitting the data into training and testing sets was already done

# create SVM regressor object with a linear kernel

svm = SVR(kernel='linear')

# fit the regressor with the training data

svm.fit(X\_train, y\_train)

# predict the target values for the test data

y\_pred = svm.predict(X\_test)

# calculate mean squared error

mse = mean\_squared\_error(y\_test, y\_pred)

print("Mean Squared Error:", mse)

# calculate R squared score

r2 = r2\_score(y\_test, y\_pred)

print("R Squared Score:", r2)

# calculate mean absolute error

mae = mean\_absolute\_error(y\_test, y\_pred)

print("Mean Absolute Error:", mae)

print()

# Print all values in X\_test, y\_pred, and y\_test

for i in range(len(X\_test)):

print("X\_test:", X\_test.iloc[i])

print("y\_pred:", y\_pred[i])

print("y\_actual:", y\_test.iloc[i])

print("Diff between Actual and predicted value", y\_test.iloc[i] - y\_pred[i])

print()

# Gaussian Naive Bayes

from sklearn.naive\_bayes import GaussianNB

from sklearn.metrics import mean\_absolute\_error, mean\_squared\_error, r2\_score

# Splitting the data into training and testing sets was already done

# fit the Gaussian Naive Bayes model to the training data

gnb = GaussianNB()

gnb.fit(X\_train, y\_train)

# make predictions on the testing data

y\_pred = gnb.predict(X\_test)

# print evaluation metrics

print('Mean Absolute Error:', mean\_absolute\_error(y\_test, y\_pred))

print('Mean Squared Error:', mean\_squared\_error(y\_test, y\_pred))

print('R-squared:', r2\_score(y\_test, y\_pred))