Time Series Data Analysis and Forecasting

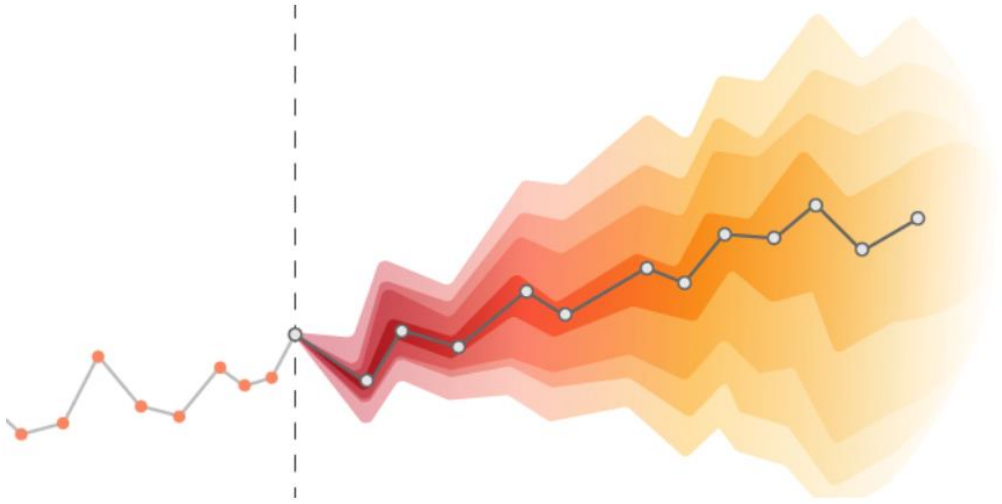
MSDS

Time Series forecasting Applications

Module 8

Topic Covered

- Time-series analysis and forecasting in Business domain
- Financial time series forecasting
- Financial applications
- Emerging trends
- Machine learning for TS forecasting
- Challenges in TS forecasting



Time-series analysis and forecasting in Business

- Time series analysis is mainly used to explain, describe, and predict changes via the time of chosen variables.
- Many companies use time series forecasting, and analysis to develop business strategies.
- These techniques help businesses in measuring, tracking, defining, and predicting business trends.

Time-series analysis and forecasting in Business domain

- Online users forecast
- Traffic forecast
- Sales Prediction
- Customer satisfaction forecast
- User spending habits forecast

Online User Forecasts

- A Time Series model predicts ~600,000 people to log in online in next few hours.
- Online sports streaming platform already knows there would be a lot of people online due to a big event happening then.
- But now it can better plan for *how many* additional servers and infrastructure are needed for the online platform, based on *how many* online users are predicted.
- Also, those servers are only used for that particular time of day, switching them off for rest of the day to save money.
- Another Time Series model predicts a significant increase in online users from last year and even more so the year before then.
- The company decides it has reached a point of continuing significant growth, and now is the right time to invest in better infrastructure for the year ahead and coming years ahead.

Traffic forecast

- A sensor device records the **number of vehicles** that cross an intersection every 20 minutes.
- Using these counts of vehicles taken every 20 minutes, a Time Series model predicts that in the next 20 minutes traffic at the intersection is likely to spike by a *huge* amount.
- Now your trip planning app decides to re-route you to avoid this congested, problematic intersection, distributing the traffic load more evenly across roads.

Sales Prediction

- In the bustling world of retail, accurate sales predictions can mean the difference between success and stagnation.
- Imagine having power to anticipate **customer demand**, **optimize inventory**, and **plan promotions** precisely.
- main challenge faced by any retail store is predicting in advance the sales and inventory required at each store to avoid over-stocking and under-stocking.
- This helps business to provide the best customer experience and avoid getting into losses, thus ensuring the store is sustainable for operation.

Customer satisfaction forecast

- **Customer reviews** are web scraped and analyzed every day, with an overall score on their sentiment that shows whether they are happy with the company or upset. A number anywhere from -1 (most upset) to +1 (most happy) is recorded each day.
- company has seen that score has dropped gradually over time.
- company is thinking about whether it should hold off on acting on this right away and save time and resources, as it might turn around to be positive again.
- A Time Series Forecast says it is not likely to get any better, and in next few days it will continue to *drop* to a score that is unacceptable for the company.
- company has decided, based on this model, to pull in extra resources to assist customer service team so that they can pay extra special attention to customers and hopefully turn that trend in the right direction.

User spending habits forecast

- Online retail site usually sees periods of **peaks and falls** in its sales.
- However, ordering stock during peak times is far more expensive than during off-peak times.
- Retailer is using forecasting models **to predict or get a closer estimation of *how much* stock is needed for the next peak season ahead**.
- Time series model can estimate **how many customer purchases** are to come during peak season.
- Retailer decides to order extra stock than expected during the off-season before it becomes exuberantly expensive to order extra stock during on season.

Financial Time Series Forecasting

- **Financial forecasting** is a financial endeavor to forecast and estimate future **financial outcomes** using historical data, current trends, and relevant information.
- It uses quantitative and qualitative approaches to forecast future financial performance, including **revenue**, **costs**, **profits**, **cash flows**, and various financial parameters.
- Companies can allocate resources aptly, establish financial objectives, evaluate prospects for investment, scrutinize the financial viability of ventures, gauge future financial exigencies, and craft strategies that foster financial performance optimization by engaging in financial forecasting.

Financial Time Series Forecasting Application

- Risk management
- Identifying Financial Frauds/Anomaly Detection
- Stock Price forecasting
- Portfolio management/optimization

Time Series forecasting in Risk management

- Risk management involves identifying, analyzing, and accepting or mitigating uncertainty in investment decisions.
- it is process of monitoring and dealing with the financial risks associated with investing.
- Risk management occurs when an investor or fund manager analyzes and attempts to quantify potential for losses in an investment,
- Analyzes **moral hazard** and then takes the appropriate action (or inaction) to meet their objectives and **risk tolerance**

Risk Management Keypoints

- Risk management is the **process of identification, analysis, and acceptance or mitigation of uncertainty** in investment decisions.
- Risk is inseparable from return in the investment world.
- Risk management strategies include avoidance, retention, sharing, transferring, and loss prevention and reduction.
- method to ascertain risk is standard deviation, which is a statistical measure of dispersion around a central tendency.

Time Series forecasting in Risk management

- A multinational corporation planning to expand into a new market uses economic forecasts to assess **growth potential**, ***currency fluctuations***, and **regulatory changes**. These insights guide *their strategic decisions*.
- A bank assesses **credit risk** by forecasting borrowers' repayment capabilities. Timely identification of potential defaults enables ***risk mitigation strategies***.

Time series models used in risk management

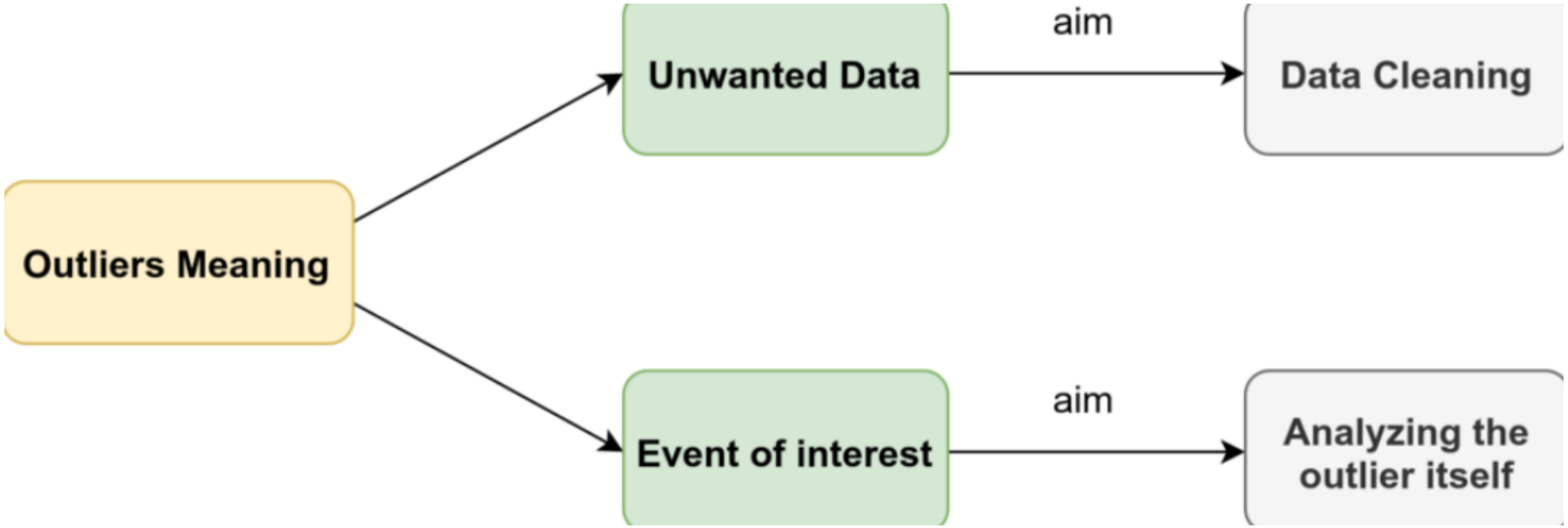
- ARIMA
- MA
- Exponential smoothing model
- Neural network, machine learning model
- Support Vector machine

Identifying Financial Frauds/Anomaly Detection using time series modeling

- financial fraud, which includes credit card fraud, insurance fraud, money laundering, healthcare fraud and securities and commodities fraud, has garnered a great deal of unwanted attention from efforts and interests seeking to prevent them.
- serious concerns are posed by alarmingly increasing rates of financial fraud.
- Total global losses annually due to financial fraud have been shown to be in the range of billions of dollars, with some figures suggesting the yearly cost to the US being in excess of $400 billion
- Anomaly detection techniques have been intensively studied for this purpose

Identifying Financial Frauds/Anomaly Detection

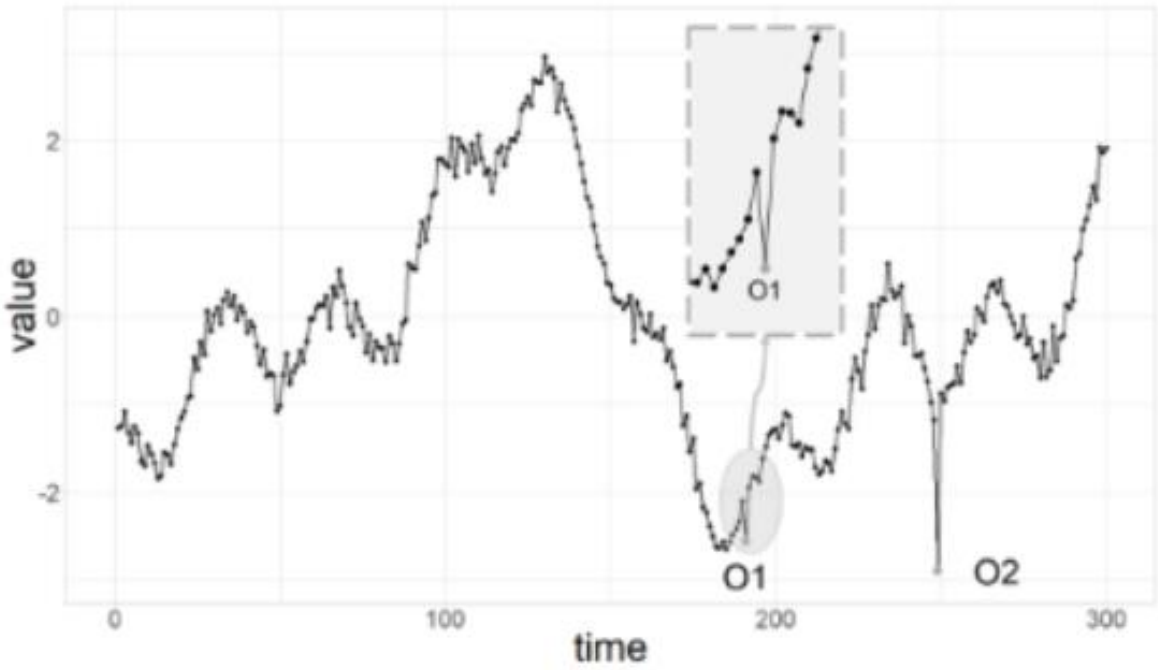
- Anomaly detection has a significant role in financial fraud detection and is used to identify and extract information from vast data quantities.
- There have been significant amounts of literature applying statistical methods, as well as AI and machine learning techniques to approach credit card and insurance fraud detection.



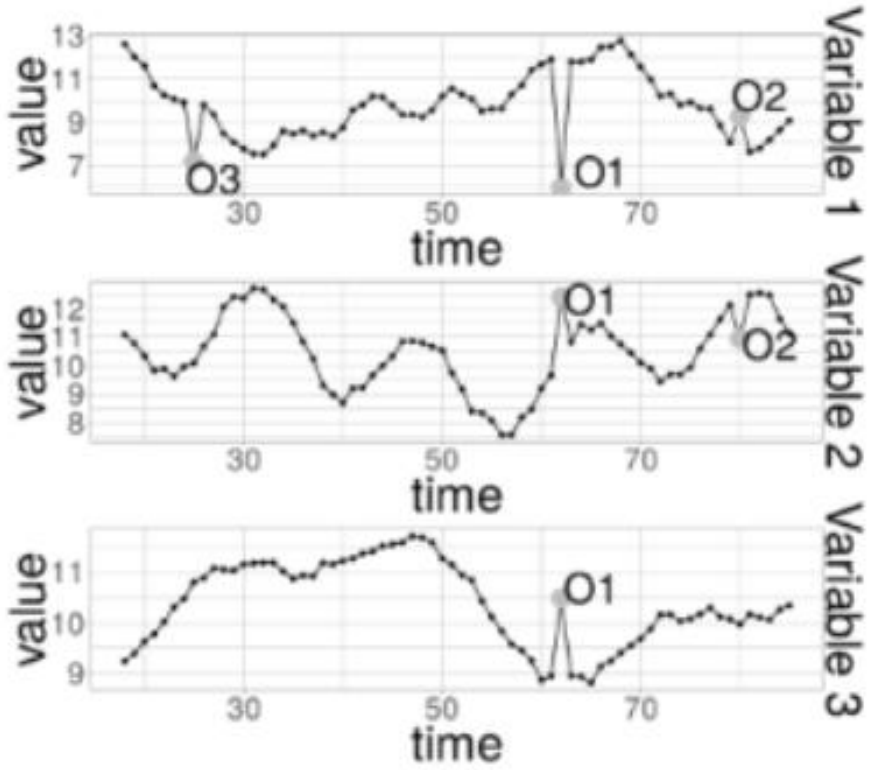
Types of Anomalies

- **Point outlier-**A point outlier is a datum that behaves unusually in a specific time instance when compared either to the other values in the time series (global outlier), or to its neighboring points (local outlier).
- **Subsequence outlier-**means consecutive points in time whose joint behavior is unusual, although each observation individually is not necessarily a point outlier. Subsequence outliers can also be global or local, and can affect one (univariate subsequence outlier) or more (multivariate subsequence outlier) time-dependent variables.

Point outlier

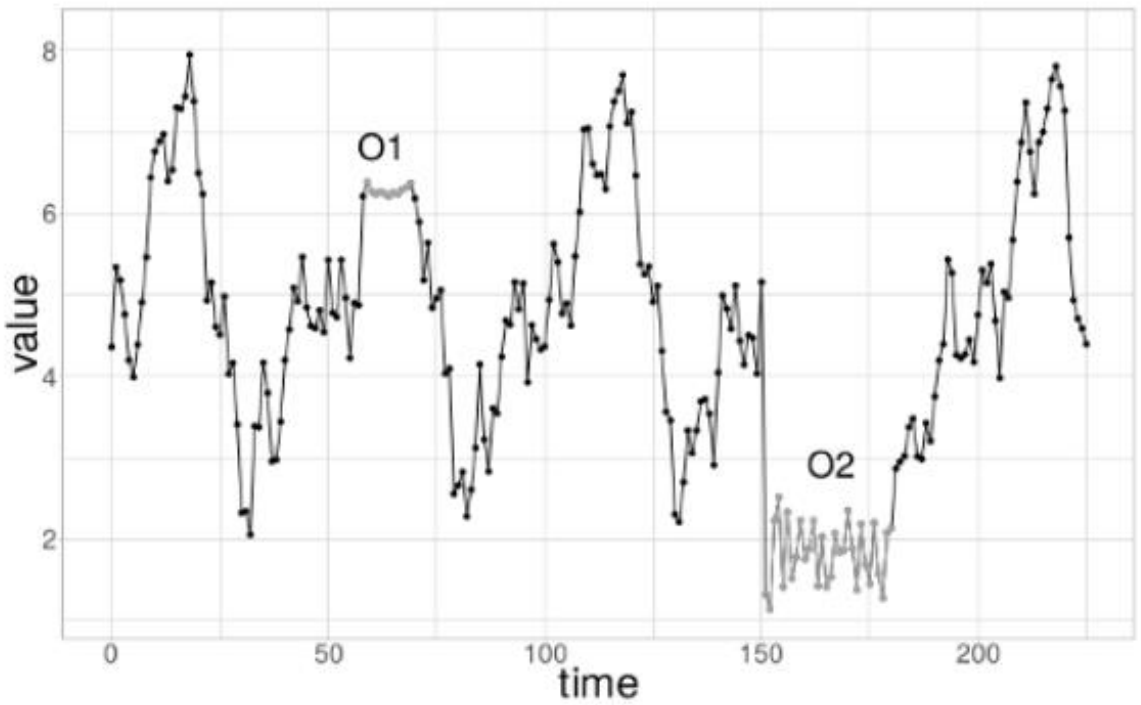


(a) Univariate time series.

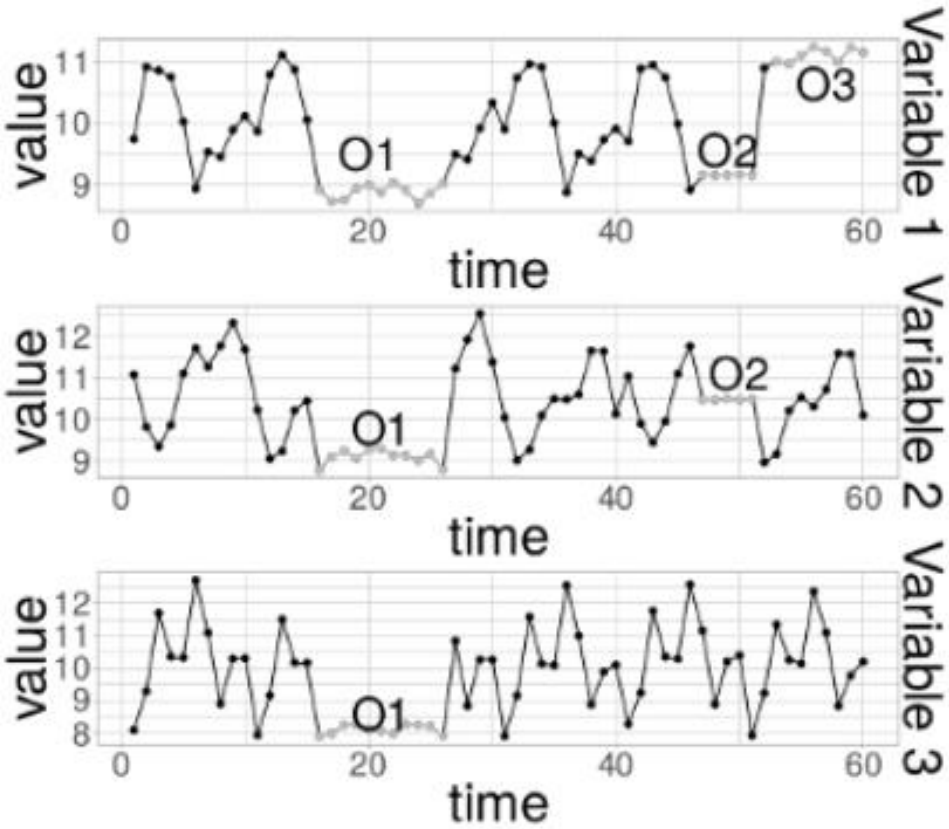


(b) Multivariate time series.

Subsequence outlier



(a) Univariate time series.



(b) Multivariate time series.

Feature Engineering for Anomaly Detection

1. **Rolling Statistics**: Calculate rolling mean, median, or standard deviation over a moving window to capture trends and variations.
2. **Lagged Variables**: Create lagged versions of your time series to capture historical patterns and correlations.
3. **Fourier Transforms**: Convert time-domain data into frequency-domain data using Fourier transforms to identify periodic patterns.

Time series models for anomaly detection

- ARIMA
- MA
- Exponential smoothing
- RNN/LSTM
- DBSCAN clustering

Stock Price forecasting Using Machine Learning and Deep learning models

- Using ML and DL following questions can be answered:-

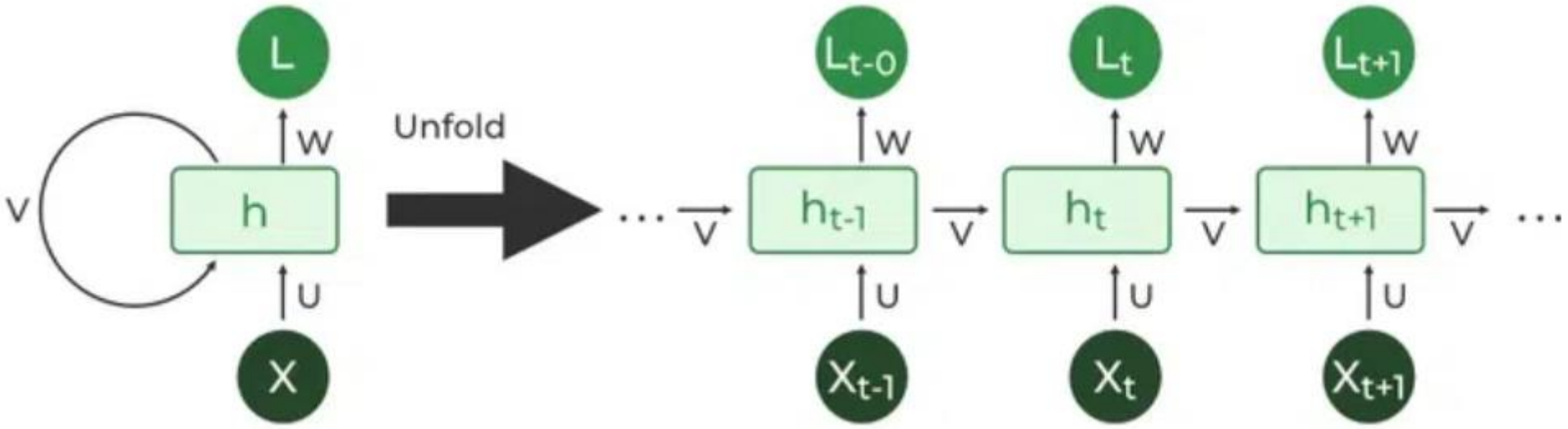
1. How has the stock price evolved over time?
2. What is the average **daily return** of the stock?
3. How does the **moving average** of the stocks vary?
4. What is the **correlation** between different stocks?
5. How can we forecast future stock behavior, exemplified by predicting the closing price of Apple Inc. using LSTM or GRU?"

Stock Price forecasting Using Machine Learning Model RNN

- model the temporal dependencies present in the data as it contains an implicit memory of previous inputs.
- time series data being sequential in nature is often used in RNN.
- For working with time series data in RNNs, TensorFlow provides a number of APIs and tools, like tf.keras.layers.RNN API, which allows to create of unique RNN cell classes and use them with data.
- Several RNN cell types are also supported by this API, including Basic

RNN

- Recurrent Neural Network(RNN) is a type of Neural Network, where the output from the previous step is fed as input to the current step.
- feature of RNN is its **Hidden state**, which remembers some information about a sequence. The state is also referred to as *Memory State* since it remembers the previous input to the network.



Steps to use RNN for Stock Price forecasting

- model the temporal dependencies present in the data as it contains an implicit memory of previous inputs.
- time series data being sequential in nature is often used in RNN.
- For working with time series data in RNNs, TensorFlow provides a number of APIs and tools, like tf.keras.layers.RNN API, which allows to create of unique RNN cell classes and use them with data.
- Several RNN cell types are also supported by this API, including Basic

Step 1: Import the required libraries.

- Numpy & Pandas – For data manipulation and analysis
- Matplotlib – For data visualization.
- Yahoo Finance – Provides financial data for analysis.
- Datetime – For working with dates and times.
- Math – Provides basic mathematical functions in

Step 2

- This code uses the yf.download() method of the yfinance library to download historical stock data for Google from Yahoo Finance.
- Using the dt.datetime() method of the datetime module
- the start and end dates of time period for which the data has been obtained are given.
- downloaded data is then shown using the print() function, where the Pandas DataFrame’s display options are configured using pd.set_option().

Step 3 Splitting data for training

- split the dataset into training and testing in the ratio 80:20.
- Only the first column of the data is chosen using iloc[:,:1] and the train_data contains the first *training_data_len* rows of the original data.
- The test_data, contains all of the remaining rows of the original data starting from *training_data_len* to the end

Step 4

- This creates a numpy array called *dataset_train* and populates it with the “Open” pricing values from the training data.
- 1-dimensional array is then transformed into a 2-dimensional array.
- The shape property, which returns the tuple (num_rows, num_columns) denoting the *dataset_train* array’s final shape.

Step 5

- Normalization is a crucial step in data preprocessing to enhance the effectiveness and interpretability of machine learning models.
- Hence MinMaxScaler from sklearn is imported to scale the dataset from 0 to 1.
- Using the sklearn fit transform() method, the training dataset is scaled.

Step 6

- The same data preprocessing is done for test data.

Step 7

- The time-series data must be divided into X_train and y_train from the training set and X_test and y_test from the testing set in this phase.
- It is done to turn time series data into a supervised learning problem that can be utilized to train the model.
- loop generates input/output sequences of length 50 while iterating through the time series data.
- forecast future values while taking into consideration the data's temporal dependence on prior observations.

Step 8

- data is converted into a format that is suitable for input to an RNN.
- ***np.reshape(X_train, (X_train.shape[0], X_train.shape[1], 1))*** transforms the X_train array, which was originally a 2-dimensional array of shape (samples, features), into a 3-dimensional array of shape (samples, time steps, features), where time steps denotes the number of time steps in the input sequence and features denotes the number of features in the input data.
- Size 1 is an additional dimension that serves as an indication that each time step only has a single feature.

Step 8

- The y_train array is transformed from a 1-dimensional array of shape (samples) into a 2-dimensional array of shape (samples, 1)
- ***np.reshape(y_train, (y_train.shape[0], 1))***, where each row represents the output value at a certain time

Step 9

Three RNN models are created in this step. The libraries needed for the model is imported.

Step 10

- The X_test data is then used to make predictions from all three models.

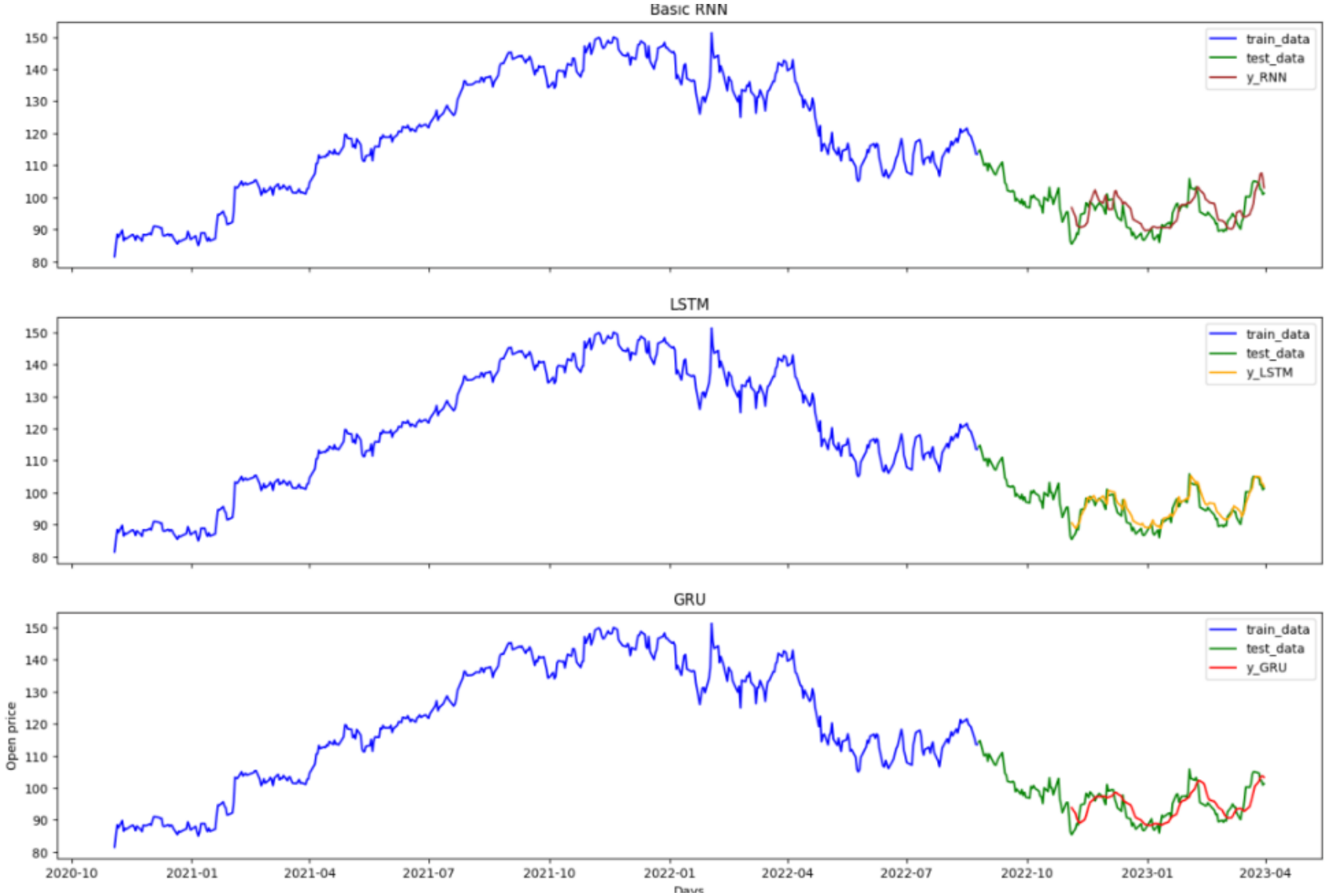
Step 11

- The predicted values are transformed back from the normalized state to their original scale using the inverse_transform() function.

Step 12

- Visualize the predicted prices using matplotlib.

Results


Basic RNN
train_data
test_data
y_RNN
LSTM
train_data
test_data
y_LSTM
GRU
train_data
test_data
y_GRU
Open price
150
140
130
120
110
100
90
80
2020-10
2021-01
2021-04
2021-07
2021-10
2022-01
2022-04
2022-07
2022-10
2023-01
2023-04
Days

Review Questions

- Discuss two application of Time series forecasting in Business sector.
- Discuss various application of Time series forecasting in Financial domain.