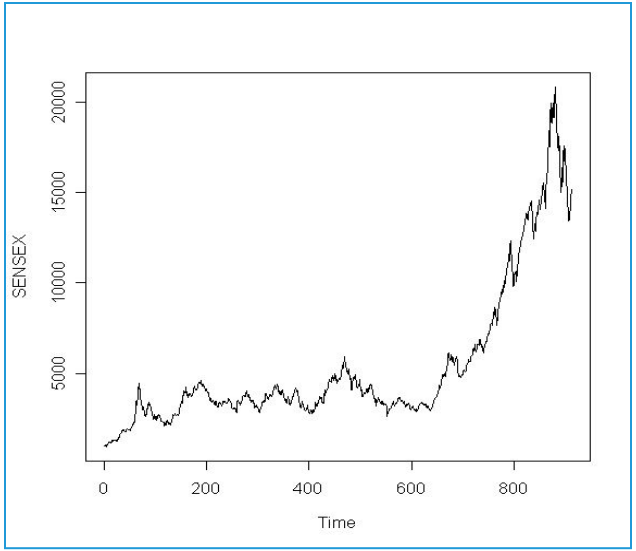
Introduction to

Time Series Analysis

Contents

What is Time Series ?

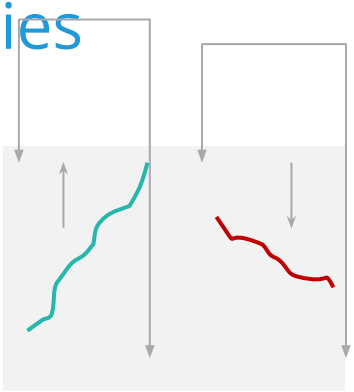
Time Series is a sequence of values observed over time

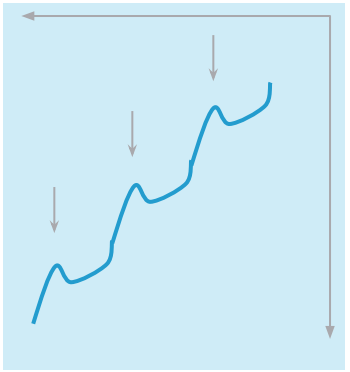


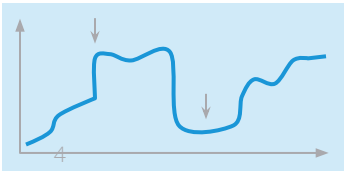
Types of Time Scale	
Discrete	Value changes after jumping from one time period to other. Example: Dow Jones Index -End of Day Values
Continuous	Value changes within an infinitely short amount of time. Example. Temperature, Dow Jones Index Tracked Real Time

Components of Time Series

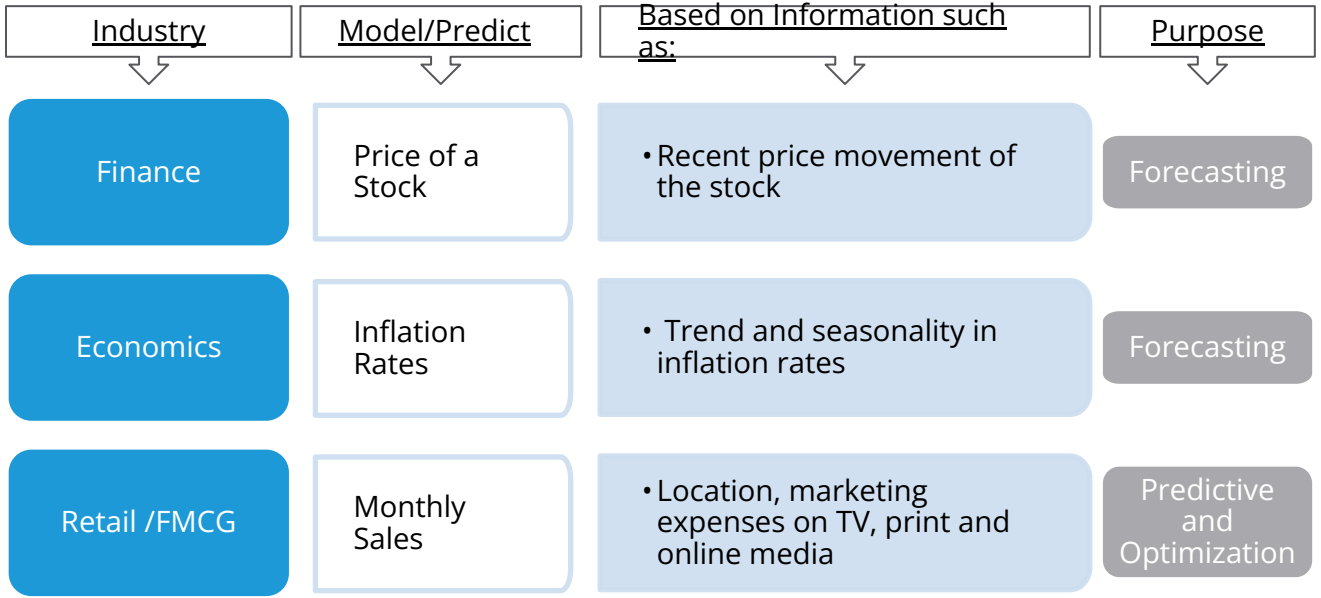
Trend	Long-term increase or decrease in the time series. There may be increase/decrease in short term but overall trend in the long term can be increasing or decreasing.
Seasonality	**Predictable and recurring trends and patterns** over a period of time, normally a year. An example of a seasonal time series is retail data, which sees spikes in sales during holiday seasons like Christmas. Seasonality is reflected only when data is available for more than one year
Cyclic Pattern	Exists when data exhibit rises and falls that are not of fixed period. The duration of these fluctuations is usually of at least 2 years







Application Areas



Case Study

Background

- Annual Sales for a specific company from year 1961 to 2017

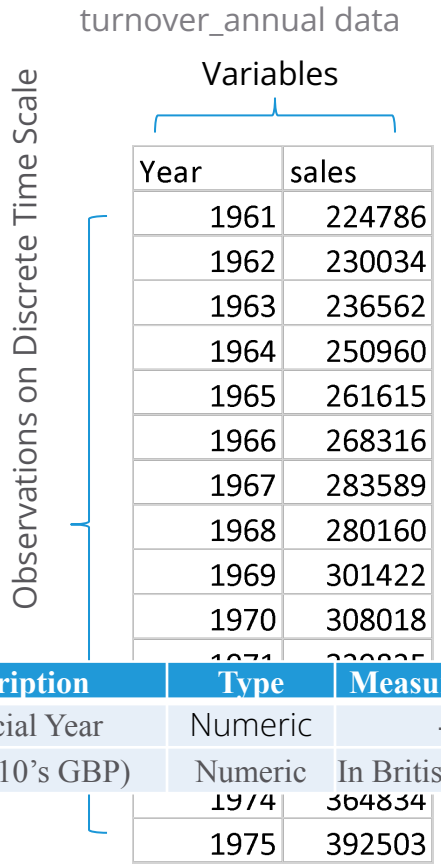
Objective

- To plot a time series object

Available Information

- Number of cases: 57
- Variables: Year, sales(in 10's GBP)

Data Snapshot



Columns	Description	Type	Measurement	Possible values
Year	Financial Year	Numeric	-	-
sales	sales(in 10's GBP)	Numeric	In British Pound	Positive values

Time Series in Python

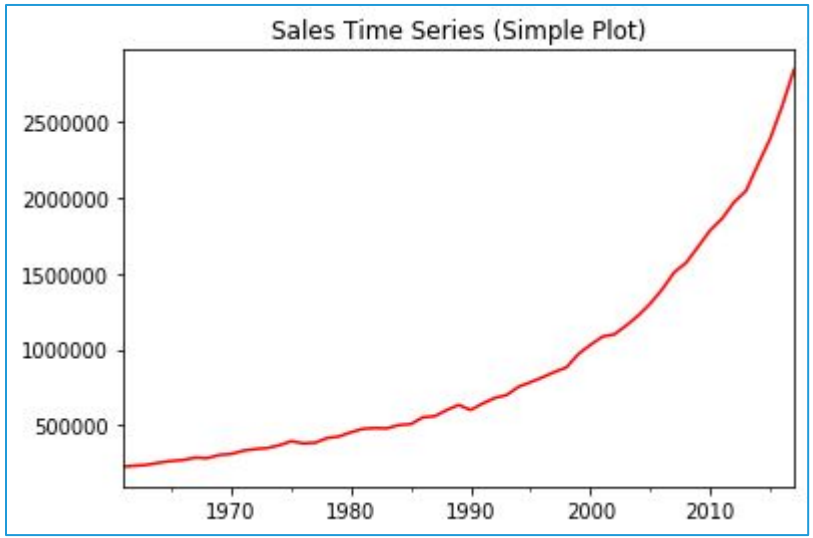
```
# Import turnover_annual Data
import pandas as pd
salesdata = pd.read_csv('turnover_annual.csv')

# Creating a Time Series Object

rng = pd.date_range('01-01-1961','31-12-2017',freq='Y')
s = salesdata.sales.values
salesseries = pd.Series(s, rng)
```

- **date_range()** creates pandas date object.
- When the time series has seasonal components, argument **freq =** can be included. It denotes number of observations per unit of time. Eg. If data is quarterly: **freq = 'Q'**, if data is monthly: **freq = 'M'**.
- **pd.Series()** combines time series variable object "s" and date object "rng".
  The new object salesseries will be used for further analysis.

Plotting Time Series in Python

```
# Plotting a Time Series Object
salesseries.plot(color='red', title ="Sales Time Series (Simple Plot)")
# Output
```

plot() generates a simple line chart.



Interpretation :

- The time-series clearly shows upward trend.

Subsetting Time Series in Python

- Large volumes of data are required for most real world analytics, time series is no exception.
- Subsetting is an important tool as it facilitates partitioning the data within Python for micro-level specific analysis.

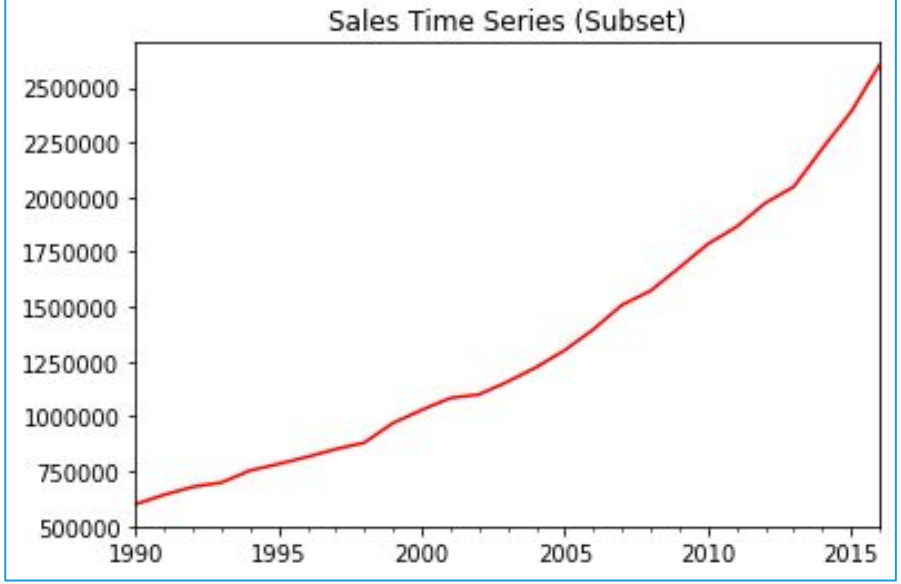
```
# Subsetting a Time Series Object
salesseries2 = salesseries.loc['1990-12-31':'2016-12-31']
```

loc[ ] is a generic function which extracts the subset of the object x observed between the times **specified within the range.**

In Pandas we can directly subset the time series object using `loc[]` function

Subsetting Time Series in Python

```
salesseries2.plot(color='red', title ="Sales Time Series (Subset)")

# Output
```



Plot from 1990 to 2016 shows increasing trend

Quick Recap

Time Series	• Sequence of values measured over time • Time scale can be discrete or continuous
Time Series in Analysis	• Analyze and forecast time series values
Time Series in Python	• pd,date_range created date object. • pd.Series creates time series object by combining date object and time series variable