# ISA 444: Business Forecasting

## 04: Plotting a Single Time Series in Python (Cont.)

Fadel M. Megahed, PhD

Professor
Farmer School of Business
Miami University

@FadelMegahed
fmegahed
fmegahed@miamioh.edu
Automated Scheduler for Office Hours

Spring 2025

# Quick Refresher of Last Class

- [x] Generate and interpret simple line charts
- [x] Create seasonal plots and subplots

# Learning Objectives for Today's Class

- Describe a lag and create lag scatter plots.
- Plot and interpret the autocorrelation function (ACF) for a time-series.

# Describe a lag and create lag scatter plots

# What is a Lag?

A **lag** refers to a time delay between an event and its observable effect or measurement.

In time series data, a **lag** is simply a previous observation in the series. For example:

- **Lag 1:** Refers to the value immediately preceding the current value.
- **Lag 2:** Refers to the value two time periods before the current value, and so on.

# What is a Lag (Cont.)?

Imagine a time series of daily sales:

- Day 1: $100
- Day 2: $150
- Day 3: $120

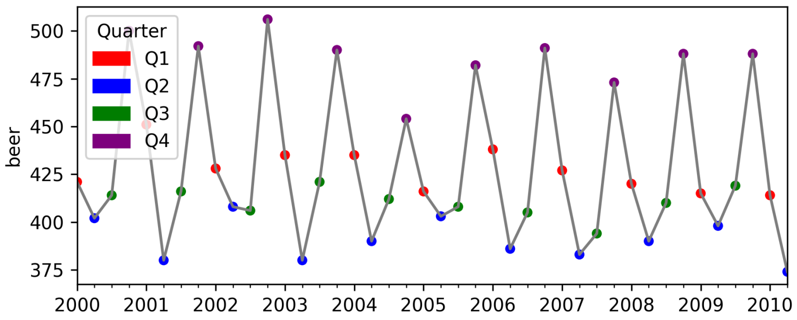
For Day 3, the lag 1 value is the sales from Day 2 ($150). Similarly, the lag 2 value is from Day 1 ($100).

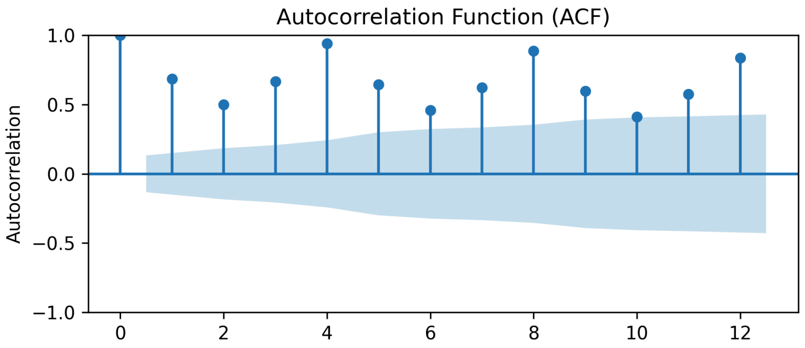
In mathematical notation, $y_t$ represents the value at time $t$, then:

- $y_{t-1}$ is the lag 1 value of $y_t$
- $y_{t-2}$ is the lag 2 value of $y_t$

# Importance of Lags

- **Lags** are important in time series analysis because they can help us understand the relationship between past and present values. They are used in **Forecasting**:
  - **Autoregressive models** Models such as AR (AutoRegressive) and ARIMA models use past values (lags) of the series to predict future values. For example, an AR(1) model uses predicts $y_t$ solely based on $y_{t-1}$.
  - **Seasonality:** Lags help capture seasonal patterns in data.



# Importance of Lags

- **Lags** are important in time series analysis because they can help us understand the relationship between past and present values. Hence, they are used in:
  - **Causal Analysis:**
    - **Cause and Effect:** In many business scenarios, past events (lags) can be predictors for current outcomes; e.g., past advertising spend might affect current sales.
    - **Policy Evaluation:** Understanding the delayed impact of policies or changes (e.g., price changes, marketing campaigns) is essential.

# Importance of Lags (Cont.)

- **Lags** are important in time series analysis because they can help us understand the relationship between past and present values. Hence, they are used in:
  - **Data Preparation and Feature Engineering:**
    - **Feature Creation:** In machine learning models, creating lag features from time series data can help the model understand historical patterns and trends.
    - **Smoothing and Differencing:** Lags are also used in techniques like differencing to stabilize the mean of a time series, which is a common preprocessing step.

# Implementing Lags in Python

In Python, we can create lags using the `shift()` method in pandas.

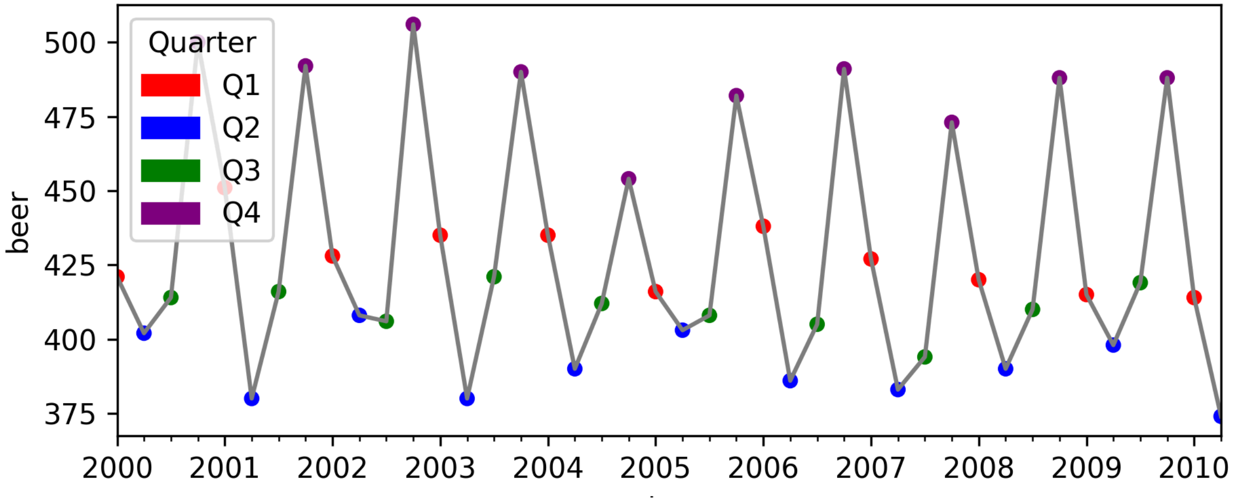
```python
import pandas as pd

df = (
  pd.DataFrame(
    {'date': pd.date_range(start='2025-01-01', periods=5, freq='D'),
    'sales': [100, 150, 120, 200, 180] }
    )
  .assign(
      lag1 = lambda x: x['sales'].shift(1),
      lag2 = lambda x: x['sales'].shift(2)
    )
)

df.head(3)
```

```
##          date  sales   lag1   lag2
## 0 2025-01-01    100    NaN    NaN
## 1 2025-01-02    150  100.0    NaN
## 2 2025-01-03    120  150.0  100.0
```

# Class Activity: Plot the `aus_beer` Time Series

05:00

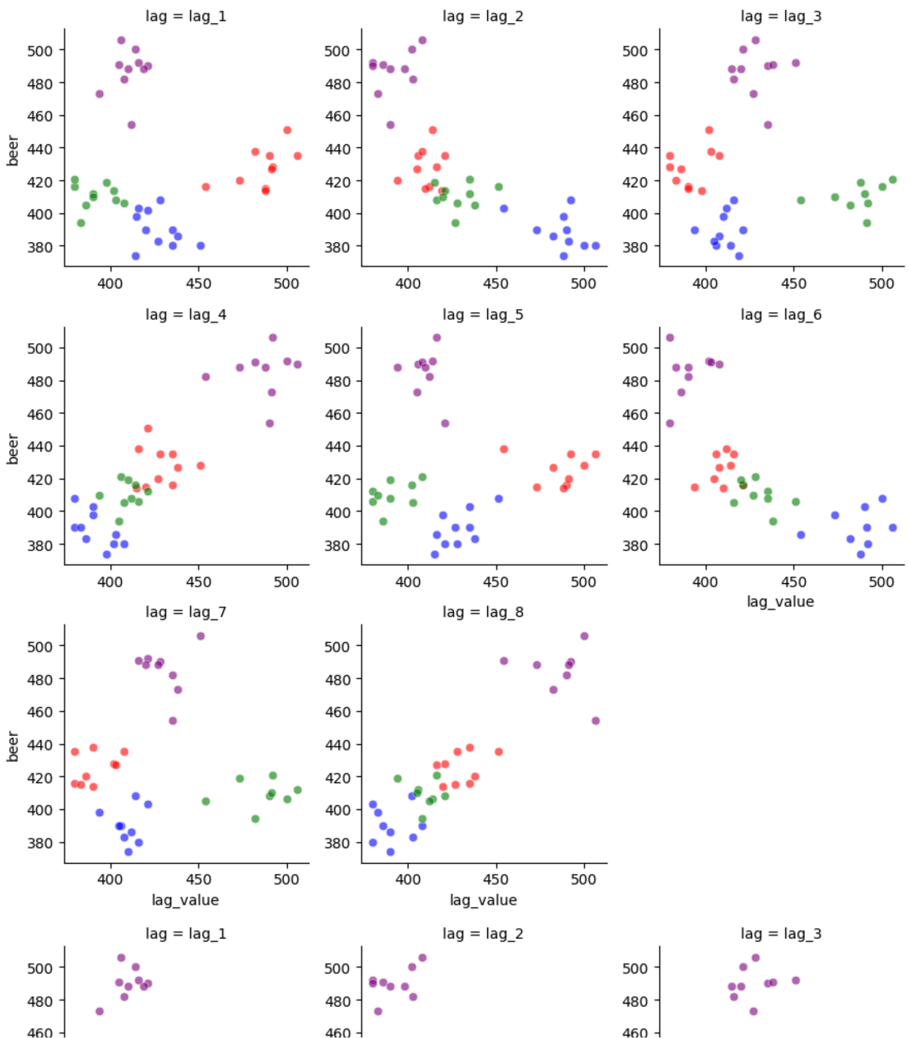
- Load the `aus_beer.csv` dataset from Canvas.
- Create a line plot of the `beer` column.
- Color the plot by quarter.
- Add a legend to the plot.



# Demo: Creating and Interpreting Lag Scatter Plots

In this demo, we will:

- Create lag scatter plots for a time series.
- Interpret the relationship between the current value and its lagged values.

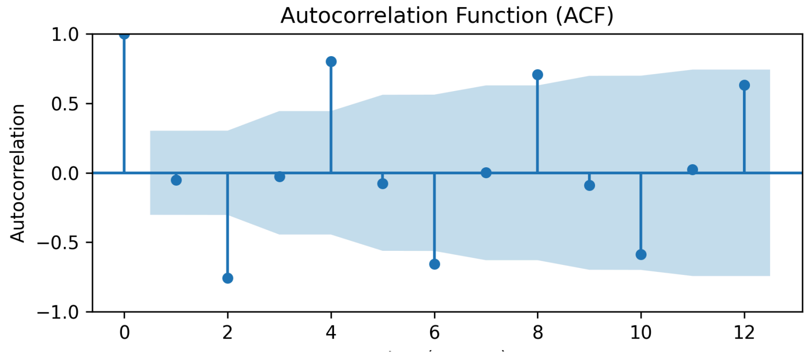


# Plot and interpret the ACF for a Time Series

# What is the Autocorrelation Function (ACF)?

The **Autocorrelation Function (ACF)** is a measure of the correlation between a time series and a lagged version of itself.

```
from statsmodels.graphics.tsaplots import
import matplotlib.pyplot as plt

df = pd.read_csv('../../data/aus_beer.csv

plot_acf(df['beer'], lags=12)
plt.title('Autocorrelation Function (ACF)
plt.xlabel('Lag (quarter)')
plt.ylabel('Autocorrelation')
plt.show()
```



# Recap

# Summary of Main Points

By now, you should be able to do the following:

- Describe a lag and create lag scatter plots.
- Plot and interpret the ACF of a time-series.

# 📝 Review and Clarification 📝

- **Class Notes**: Take some time to revisit your class notes for key insights and concepts.
- **Zoom Recording**: The recording of today's class will be made available on Canvas approximately 3-4 hours after the session ends.
- **Questions**: Please don't hesitate to ask for clarification on any topics discussed in class. It's crucial not to let questions accumulate.