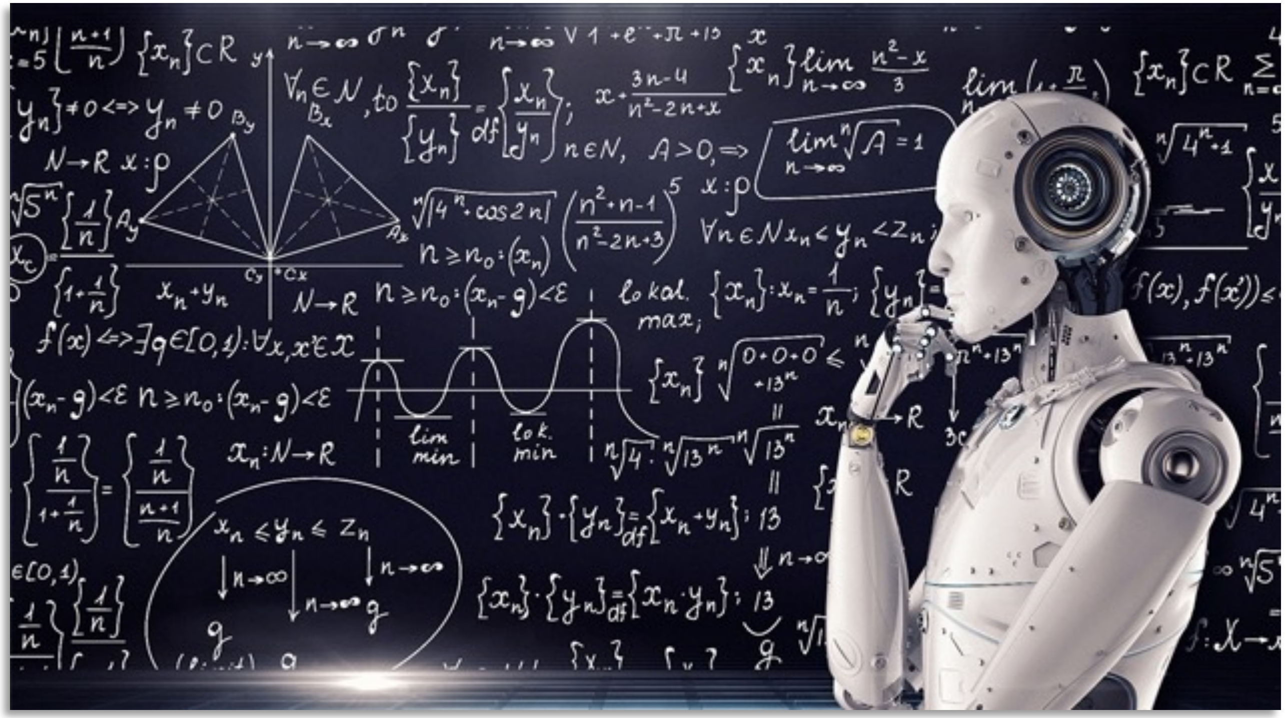
UMBC

INTRODUCTION TO MACHINE LEARNING
DATA 602 Lecture 3



Dr. Tony Diana
tonydian@umbc.edu

What are Time Series?

What is a Time Series?

- *An ordered sequence of values of a variable at equally spaced time intervals* (Engineering Statistics Handbook)

Why are Time Series Used for?

- Obtain an understanding of the **underlying forces** and **structure** that produced the observed data, and
- **Fit a model** and proceed to **forecasting, monitoring**, or even feedback and feedforward **control**

What are the Components of Time Series?

- **Seasonal variations** that repeat over a specific period such as a day, week, month, season, etc.
- **Trend variations** that move up or down in a reasonably *predictable* pattern
- **Cyclical variations** that correspond with business or economic 'boom-bust' cycles or follow their own peculiar cycles, and
- **Random variations** that do not fall under any of the above three classifications

Two Classes of Time Series Analysis

- **Frequency-Domain**: spectral and wavelet analysis
- **Time-Domain**: Autocorrelation and cross-correlation

- In this lecture, we are assuming univariate series in the form y_1, y_2, … , y_t … where every observation y_t depends on time
- A process that is determined by law is **deterministic**. Otherwise, it is **stochastic**
- The temporal mean of the process is μ_t = $E[y(t)]$
- The temporal variance is σt_2 = $E[(y(t) - E[y(t)])^2]$
- The autocovariance function is defined as $cy(t_{1,}\ t_2)$ = $E[(y(t_1) - E[y(t_1)]]*[(y(t_2) - E[y(t_2)]]$

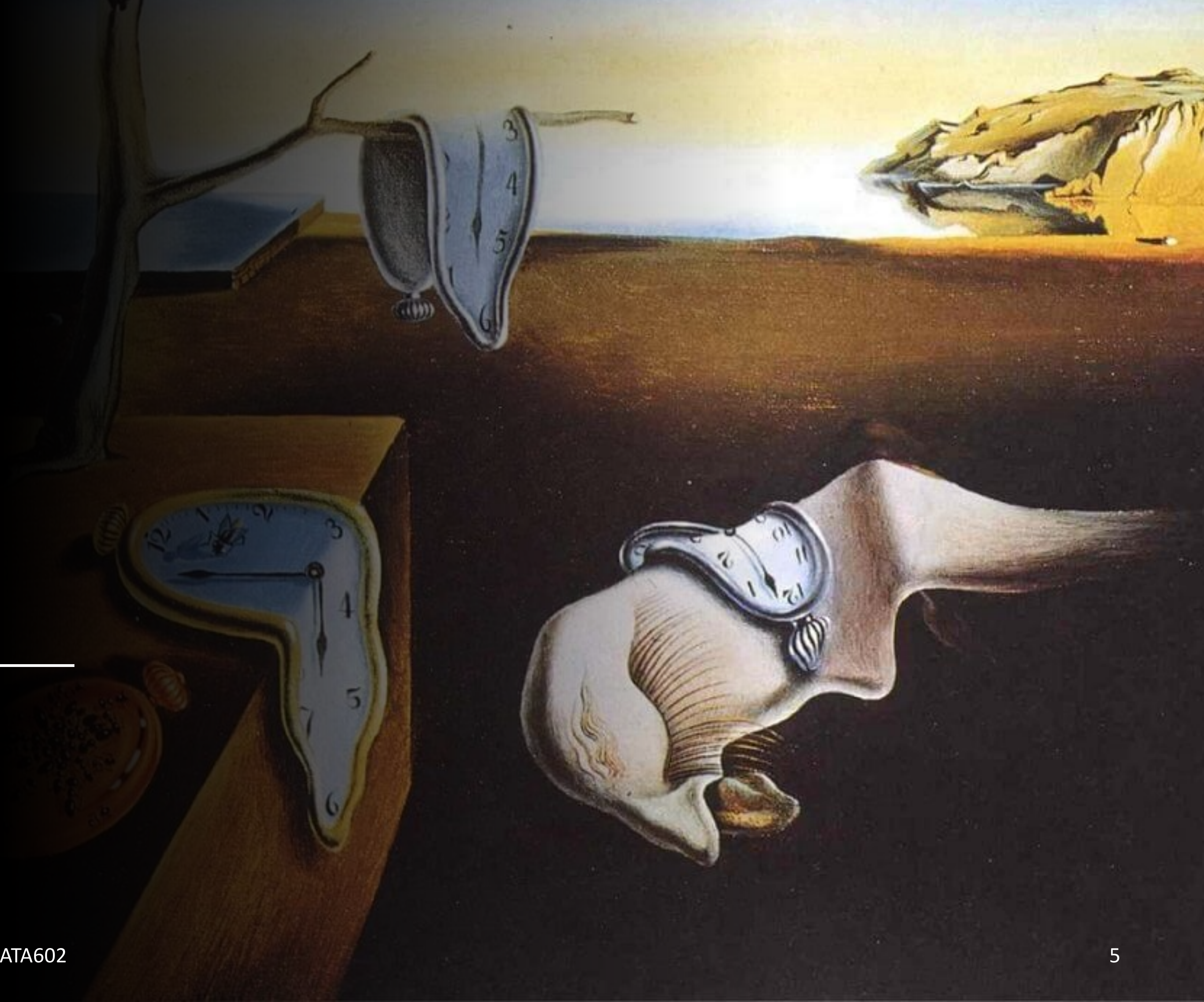
Types of Data

- **Time Series**
  - A set of observations on the values that a variable takes at different times
- **Cross-Sectional Data:**
  - Data of one or more variables, collected at the same point in time
- **Pooled Data:**
  - A combination of time series data and cross-sectional data

Types of Analysis

- **Parametric**: assumption of an underlying process
- **Non-Parametric**: no assumption of a specific structure

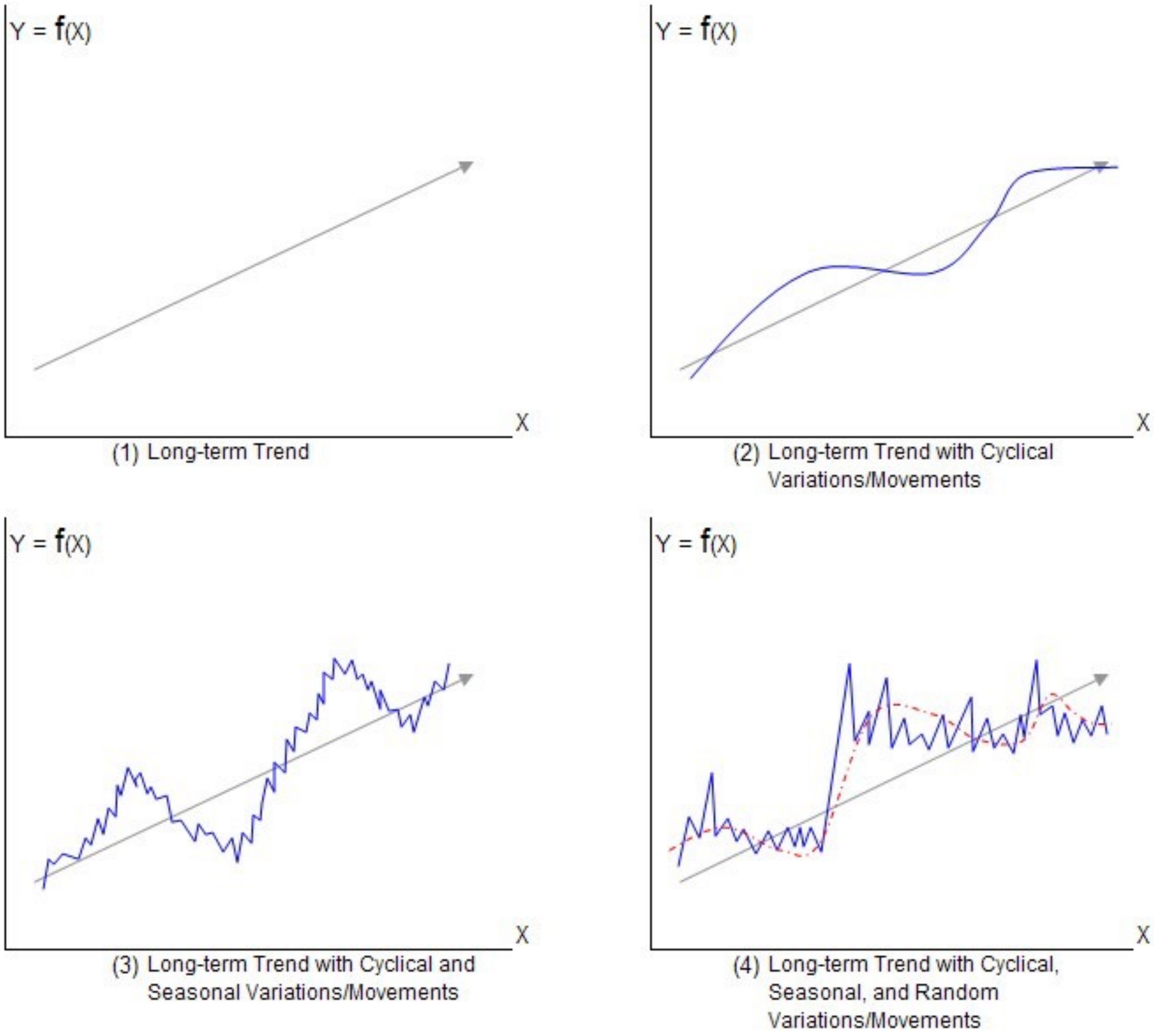
Time Series Components



The Components of a Time Series

- The process is strongly **stationary** if the full joint probability distribution is invariant to time shifts
- A **weakly stationary processes** is more realistic and characterized by constant temporal mean and variance and $c_y(t_1, t_2) = c_y(t_2 - t_1) = c_y(\tau)$
- If the vertical mean (the mean obtained after having fixed a time instant) equals the temporal mean, the process is said to be ergodic.
  - A stochastic process is said to be ergodic **if its statistical properties can be deduced from a single, sufficiently long, random sample of the process**
- A **white noise** process is a Gaussian process (even if this is not a fundamental requirement), with null mean, fixed variance, and uncorrelated realizations (that is, cov$[y_t, yq] = 0 \ \forall \ t, q$)
- A non-stationary process can be decomposed as $y_t = t_t + s_t + e_t$ where t refers to a **trend**, s to a **seasonal component**, and e to a **random component**
- **Smoothing** is a method designed to make a time series **stationary** based on the assumption that y_t depends on previous values such as y_{t-1}, y_{t-2} …
- **Exponential smoothing** is characterized as $y_t = s_t - \lambda s_{t\text{-}1}$ / 1 - λ where each term y_t is the **weighted difference** of two consecutive terms s_t

Components of Time Series Data | Types of Variation



What are Seasonal Variations?

- **Seasonal variation** is a component of a **time series** which is defined as the **repetitive** and **predictable** movement around the trend line in one year or less
- It is detected by measuring the quantity of interest for small **time** intervals, such as days, weeks, months, or quarters
- These are **short-term movements** that occur in data due to seasonal factors
- Short term is generally considered as a period in which changes happen in a time series with variations in weather or festivities, for instance
- **Seasonality** refers to **predictable** changes that occur over a one-year period in a business or economy based on the **seasons** including **calendar** or **commercial seasons**
    - One **example** of a **seasonal** measure is retail sales, which typically sees higher spending during the fourth quarter of the calendar year

What are Trend Variations?

- The **trend** is the component of a time series that represents **variations** of low frequency in a time series as the high and medium frequency fluctuations having been filtered out
- The **secular trend** is the main component of a time series, which results from long-term effects of socio-economic and political factors
- A trend may show **growth** or **decline** in a time series over a **long** period
- Trend variations imply a tendency which continues to persist for a very long period

What are Cyclical Variations?

- These are long-term fluctuations occurring in a time series
- **Cyclical fluctuations** are alternating periods of contraction and expansion than can last 18 months or longer from the peak to the trough of the cycle
  - Consumer and business demand falls during contraction and rises during expansion
- These fluctuations are mostly observed in economics data and the periods of such oscillations are generally extended from five to twelve years or more
- Cyclical variations are associated with the well-known business cycles
- Cyclic movements can be studied provided a long series of measurements, free from irregular fluctuations, is available
- **Cyclical** is recurring at **regular intervals**, while **seasonal** is related to or reliant on a **season** or period of the year
- A **cyclical process** is one in which a series of events happens again and again in the same order

Stationarity

What are Random Variations?

- These are **sudden changes** occurring in a time series which are **unlikely to be repeated**
- They are components of a time series which **cannot be explained** by trends, seasonal or cyclic movements
- These variations are sometimes called **residual or random components**
- These variations, though accidental in nature, can cause a continual change in the trends, seasonal and cyclical oscillations during the forthcoming period

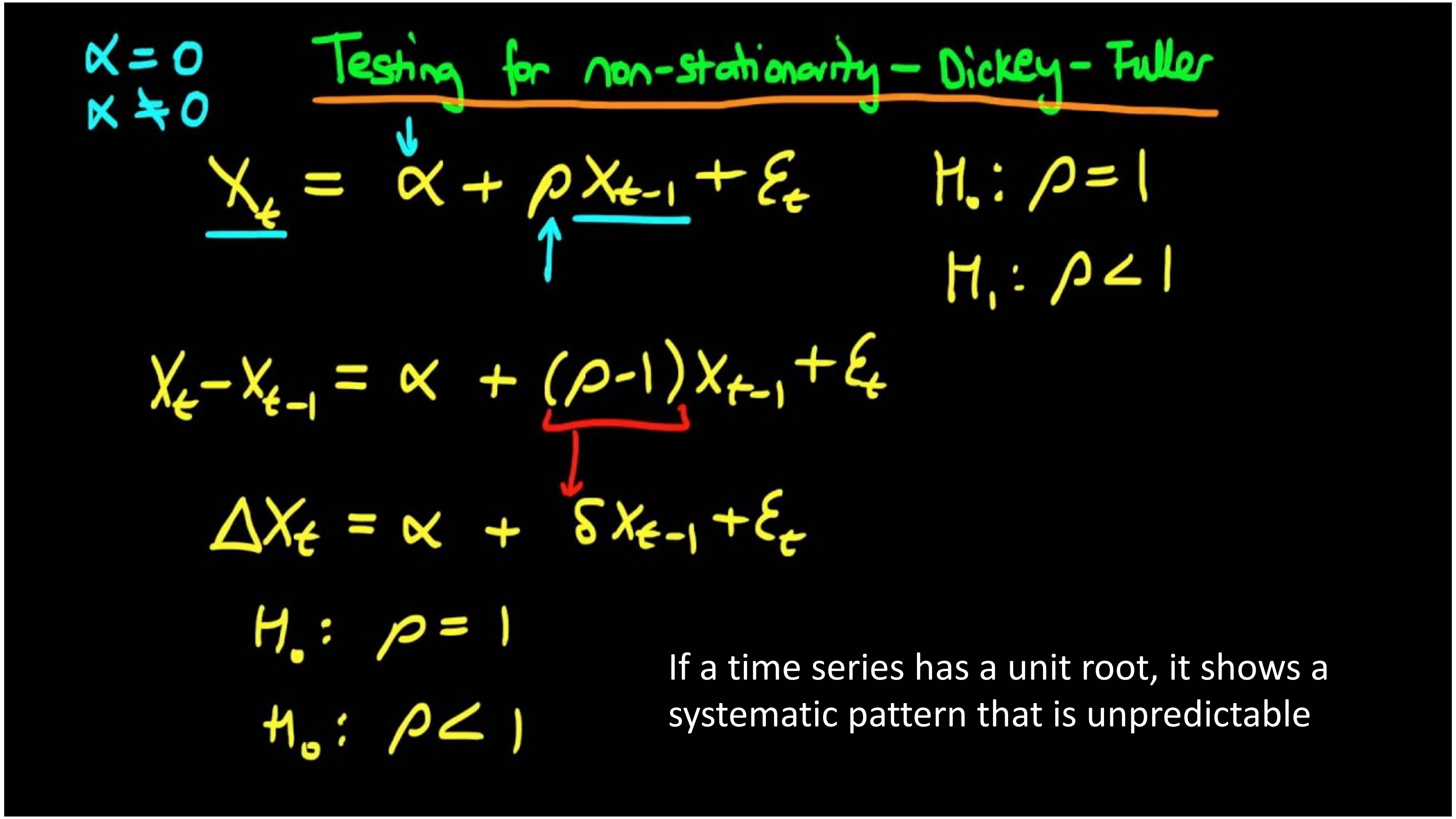
What is Stationarity?

- A stationary process has the property **that *the mean, variance, and autocorrelation structure do not change over time***
- Caution: Stationarity does not mean that the series does not change over time
  - Stationarity shows the **mean value of the series that remains constant over a time period**
  - *If past effects accumulate and the values increase toward infinity, then stationarity is not met*
- Stationarity has become a common assumption for many practices and tools in time series analysis
- These include trend estimation, forecasting, and causal inference, among others

- Stationarity is a property of a **stochastic process**
  - A **stochastic process** is any process describing the evolution in time of a random phenomenon
- There are **strong** and **weak** stationary processes
- A **white noise** process is an example of a weak stationary process:
  - It is a serially uncorrelated stochastic process with ***a mean of zero and a constant and finite variance***

- There are several **orders** of stationarity:
  - The **first order stationarity** describes a series that has a mean that never changes with time, but for which any other moment (like variance) can change
  - The ***n*-th order** stationarity requires a shift-invariance (in time) of the distribution of any *n*-samples of the stochastic process, for all *n* up to order *N*

Test of Stationarity: Dickey Fuller Test



Source: Ben Lambert, youtube

- α is the stochastic intercept, X_t is the variable of interest, t is the time index, ρ is a coefficient, and ϵ_t the error term
- A **unit root** (also called a **unit root process** or a difference stationary process) is a stochastic trend in a time series, sometimes called a "random walk with drift". If a time series has a unit root, it shows a **systematic pattern** that is **unpredictable**
- A unit root is present if $\rho = 1$. The model would be non-stationary in this case
- Details on Unit Root and Test at https://faculty.chicagobooth.edu/ruey.tsay/teaching/uts/lec11-08.pdf

Other Key Concepts in Time Series Analysis

Dependence

- Dependence refers to the **association of two observations with the same variable**, at prior time points

Differencing

- Differencing is used to make the series stationary, to de-trend, and to control the autocorrelations
- Differencing of a time series in discrete time is the **transformation** of the series to a new time series where the values are the **differences** between **consecutive values** of the time series
- Differencing can help stabilize the mean of a time series by removing changes in the level of a time series, and, therefore, by eliminating (or reducing) trend and seasonality
- However, some time series analyses do not require differencing and over-differenced series can produce inaccurate estimates

Specification

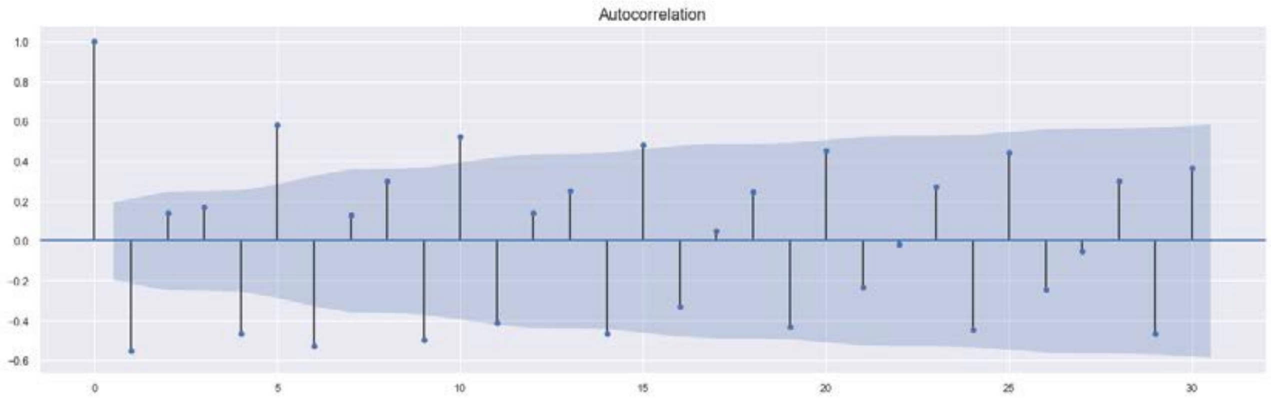
- It refers to the testing of linear or non-linear relationships of dependent variables by using models such as ARIMA, ARCH, GARCH, VAR, co-integration, etc.

Order

- The **order** of an autoregression is the number of immediately preceding values in the series that are used to predict the value at the present time

Autocorrelation

- It is the correlation between two values of the same variable at times X_i and X_{i+k}
- It allows to detect **non-randomness** in data and identify an appropriate model
- The autocorrelation function measures the correlation that the process has with itself when it is sampled at time t and $t + \tau$
- The larger the randomness, the smaller the autocorrelation. Moreover, $ry(\tau)$ very often naturally decays with τ because we assume a stronger correlation in a short time range and a much weaker one when $\tau \rightarrow \infty$



- The first evaluation (τ = 0) is always equal to 1 because any y_i is fully correlated with itself
- The remaining indicators are placed at fixed-distance lags

Time Series Models

Autoregressive Models

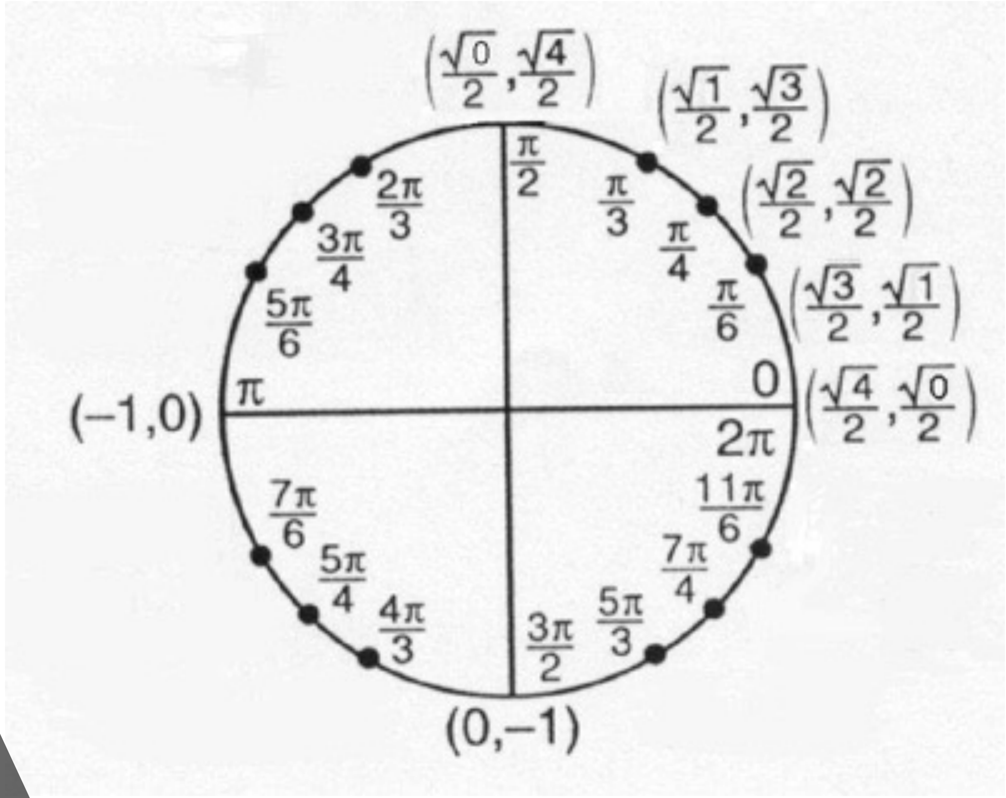
AR(1): First-Order Autoregression

- $y_t = \beta_0 + \beta_1 y_{t-1} + \epsilon_t$

- This process is called AR(p) or autoregressive of order p because y_t depends on past values of p (through a regression on itself)
- An **autoregressive** (AR) **model** predicts future behavior based on past behavior
- It is used for forecasting when
  - There is some correlation between (1) values in a time series and (2) the values that precede and succeed them
- You only use past data to model the behavior, hence the name autoregressive
- The process is basically a linear regression of the data in the current series against one or more past values in the same series
- The AR process is an example of a stochastic process, which has degrees of uncertainty or randomness built in
- The randomness means that you might be able to predict future trends well with past data, but you are never going to get 100 percent accuracy. Usually, the process gets “close enough” for it to be useful in most scenarios
- The term ϵ_t is white noise, therefore Cov[$\epsilon_t \epsilon_q$] = 0 $\forall\ t, q$
- The stationarity of this process depends on the roots of the z-transform of the model, which is equivalent to computing the roots of the complex polynomial:

$$\alpha_p{}^{z-p} + \alpha_{p-1} z^{-p+1} + \cdots + 1 = 0$$

- If the roots lie within the **unit circle**, the process is stationary
- AR models are sometimes called conditional models, Markov models, or transition models



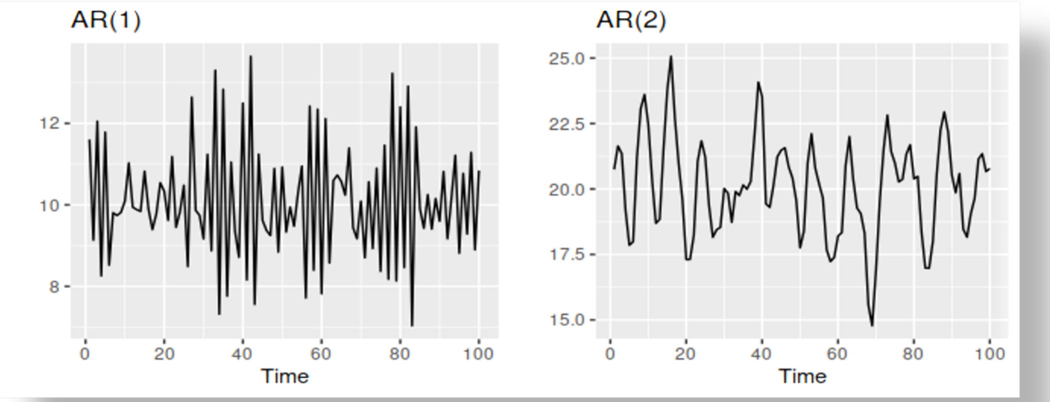
- A unit circle is a circle with a radius of one
- See https://science.howstuffworks.com/math-concepts/unit-circle.htm

- An **autoregressive model** of order p can be written as
  - $y_t = c + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \cdots + \varphi_p y_{t-p} + \varepsilon_t$, where ε_t is white noise
- This is like a multiple regression but with *lagged values* of y_t as predictors. We refer to this as an **AR(p) model**, an autoregressive model of order p
- Autoregressive models are remarkably flexible at handling a wide range of different time series patterns
- Changing the parameters $\varphi_1,\ldots,\varphi_p$ results in different time series patterns
- The variance of the error term ε_t will only change the scale of the series, not the patterns
- For an AR(1) model:
  - when $\varphi_1 = 0$, y_t is equivalent to **white noise**
  - when $\varphi_1 = 1$ and $c = 0$, y_t is equivalent to a **random walk**
  - when $\varphi_1 = 1$ and $c \neq 0$, y_t is equivalent to a **random walk with drift**
  - when $\varphi_1 < 0$, y_t tends to oscillate around the mean

We normally restrict autoregressive models to stationary data, in which case some constraints on the values of the parameters are required:

- For an AR(1) model: $-1 < \varphi_1 < 1$
- For an AR(2) model: $-1 < \varphi_2 < 1$, $\varphi_1 + \varphi_2 < 1$, $\varphi_2 - \varphi_1 < 1$

When $p \geq 3$, the restrictions are much more complicated



Example of Time Series Analysis with Python Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.seasonal import seasonal_decompose
# Read the AirPassengers dataset
airline = pd.read_csv('/content/AirPassengers.csv’, index_col ='Month’, parse_dates = True)
# Print the first five rows of the dataset
airline.head()
# Error Trend Seasonal (ETS) Decomposition (i.e. Exponential Smoothing Algorithm)
result = seasonal_decompose(airline['#Passengers’], model ='multiplicative')
# ETS plot
result.plot()
# Autocorrelation Plot
from pandas.plotting import autocorrelation_plot
autocorrelation_plot(airline)
plt.show()
```

Example of Time Series Analysis with Python Libraries (Cont.)

 # Use an ARIMA(2,1,2) model

from statsmodels.tsa.arima_model import ARIMA

 # Fit model

model = ARIMA(airline, order=(2,1,2))

model_fit = model.fit(disp=0)

print(model_fit.summary())

from pandas import DataFrame

 # Plot residual errors

residuals = DataFrame(model_fit.resid)

residuals.plot()

plt.show()

residuals.plot(kind='kde')

plt.show()

print(residuals.describe())

Autoregressive Models (Cont.)

ARIMA(p,d,q) | Box-Jenkins Models:

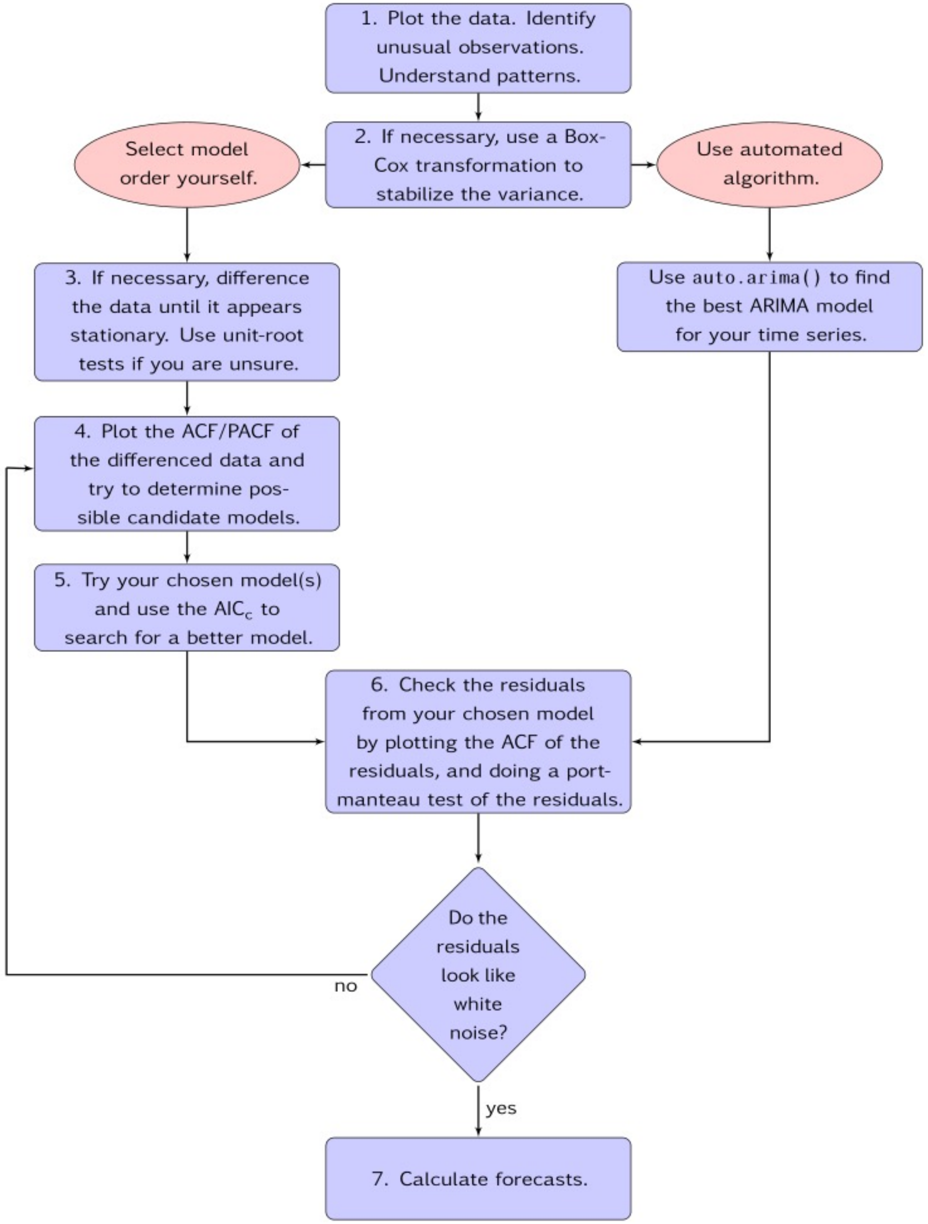
- In ARIMA, p is **autoregressive** order (AR), d is the order of **differencing** (I), and q is the **moving average** order (MA)
- To construct the ARIMA model:
  - **Stationarize the series** by differencing, logging, deflating, among several strategies
  - Study the **pattern of autocorrelation** and **partial autocorrelations** to determine the lags of the stationarized series and/or lag of the forecast errors
  - **Fit the model** and check the **residual diagnostics**, especially the **residual autocorrelation function** (ACF) and **partial autocorrelation function** (PACF)

Forecasting equation for *y*

$$\hat{y}_t = \overbrace{\mu}^{\text{constant}} + \overbrace{\phi_1 y_{t-1} + \ldots + \phi_p y_{t-p}}^{\text{AR terms (lagged values of } y\text{)}}$$

$$\underbrace{-\theta_1 e_{t-1} \ldots - \theta_q e_{t-q}}_{\text{MA terms (lagged errors)}}$$

By convention, the AR terms are + and the MA terms are –



Modeling Procedure and Box Cox Transformation

- A **Box Cox transformation** is a way to transform non-normal dependent variables into a normal shape
- **Normality** is an important assumption for many statistical techniques
- If the data are not normal, apply a **Box-Cox transformation**
- At the core of the **Box Cox transformation** is an exponent, lambda (λ), which varies from -5 to 5
- All values of λ are considered and the optimal value for your data is selected
- The "optimal value" is the one which results in the best approximation of a normal distribution curve The transformation of Y has the form:

$$y(\lambda) = \begin{cases} \frac{y^\lambda - 1}{\lambda}, & \text{if } \lambda \neq 0; \\ \log y, & \text{if } \lambda = 0. \end{cases}$$

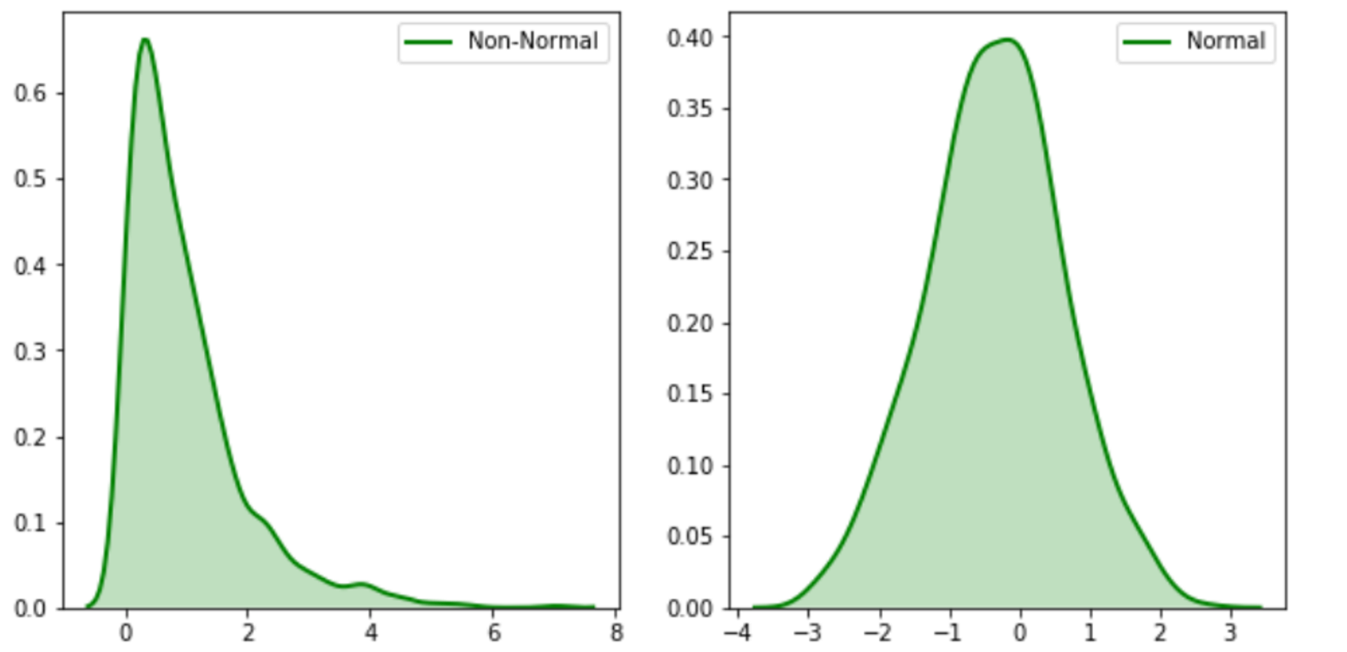
- This test only works for positive data. However, Box and Cox did propose a second formula that can be used for negative y-values:

$$y(\boldsymbol{\lambda}) = \begin{cases} \frac{(y+\lambda_2)^{\lambda_1} - 1}{\lambda_1}, & \text{if } \lambda_1 \neq 0; \\ \log(y+\lambda_2), & \text{if } \lambda_1 = 0. \end{cases}$$

Modeling Procedure and Box Cox Transformation (Cont.)

```
# import modules
import numpy as np
from scipy import stats

# plotting modules
import seaborn as sns
import matplotlib.pyplot as plt

# generate non-normal data (exponential)
original_data = np.random.exponential(size = 1000)

# transform training data & save lambda value
fitted_data, fitted_lambda = stats.boxcox(original_data)

# creating axes to draw plots
fig, ax = plt.subplots(1, 2)

# plotting the original data(non-normal) and
# fitted data (normal)
sns.distplot(original_data, hist = False, kde = True,
        kde_kws = {'shade': True, 'linewidth': 2},
        label = "Non-Normal", color ="green", ax = ax[0])

sns.distplot(fitted_data, hist = False, kde = True,
        kde_kws = {'shade': True, 'linewidth': 2},
        label = "Normal", color ="green", ax = ax[1])

# adding legends to the subplots
plt.legend(loc = "upper right")

# rescaling the subplots
fig.set_figheight(5)
fig.set_figwidth(10)

print(f"Lambda value used for Transformation: {fitted_lambda}")
```

```
Lambda value used for Transformation: 0.30656155175590766
```



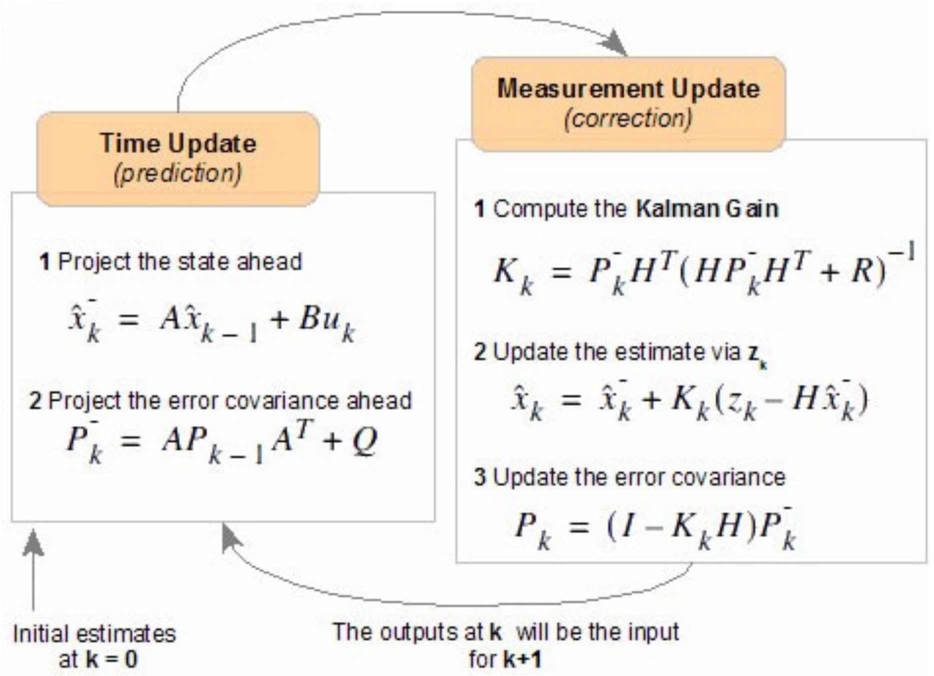
Box-Cox does not always guarantee normality because it never checks for normality. It only checks for the smallest standard deviation

UMBC

$$\overbrace{\hat{X}_k}^{\text{current estimation}} = \underbrace{K_k}_{\text{Kalman Gain}} . \overbrace{Z_k}^{\text{measured value}} + (1 - K_k) . \underbrace{\hat{X}_{k-1}}_{\text{previous estimation}}$$

- The k's on the subscript are states. We can treat them as discrete time intervals, such as k=1 means 1 ms, k=2 means 2 ms
- Our purpose is to find $\widehat{x_k}$, the estimate of the signal x
- K_k is the Kalman gain
- A Kalman filter finds the most optimum averaging factor for each consequent state
- It also remembers the past states

Kalman Filter (Cont.)



- The **Kalman filter** consists of two stages:
  - In the first stage, a **mathematical state model** is used to make a prediction about the system state
  - In the next stage, this **state prediction** is compared to **measured state values**. The difference between the predicted and measured state is smoothed based on **estimated noise** and **error** in the system and measurements, and a **state estimation** is output
  - The **output estimation** is then used in conjunction with the **mathematical state model** to predict the **future state** during the next time update, and the cycle begins again

Additional Resources

- https://towardsdatascience.com/time-series-forecasting-arima-models-7f221e9eee06
- Box, G. E. P., Jenkin, G. M., and Reinsel, G. C. (2008). Time Series Analysis: Forecasting and Control. Wiley
- Brillinger, D. R. (2001). Time Series: Data Analysis and Theory. SIAM
- Brockwell, P. J., and Davis, R. A. (2016). Introduction to Time Series and Forecasting. Springer
- Hamilton, J. D. (1994). Time Series Analysis. Princeton University Press
- Tanaka, K. (2017). Time Series Analysis. Wiley