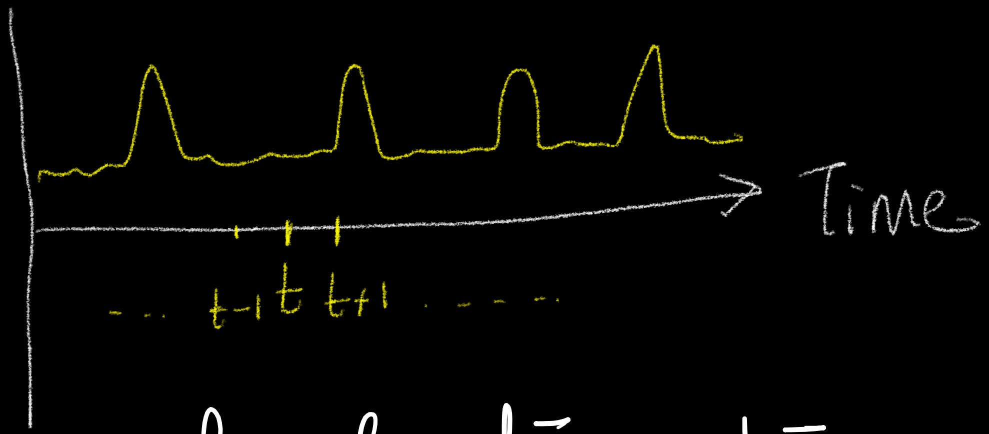
Time Series Analysis and Forecasting

Agenda:

1. Basic Terminology
2. AR, MA, Difference, ARIMA
3. Topics already covered
   - → Fourier Series
   - → Forecasting as Regression
   - → Deep Learning.

Terminology:

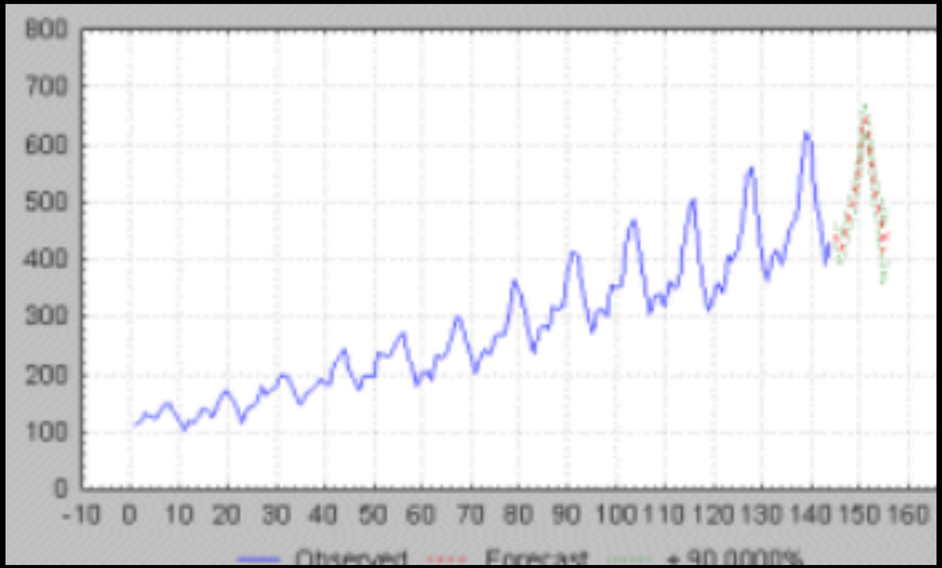
Time-series:



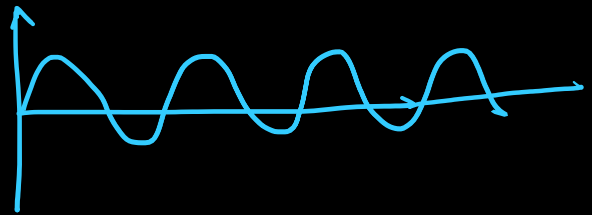
heart-rate, stock-market,

items sold per hour,

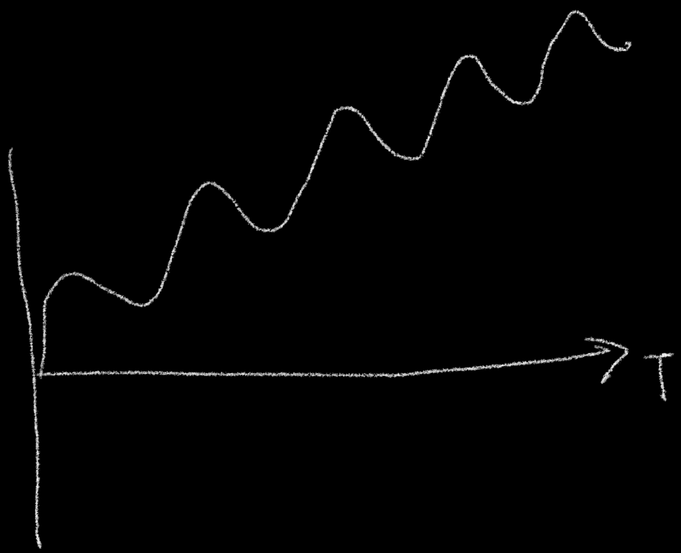
cabs required per 10 min,



Stationary time-series

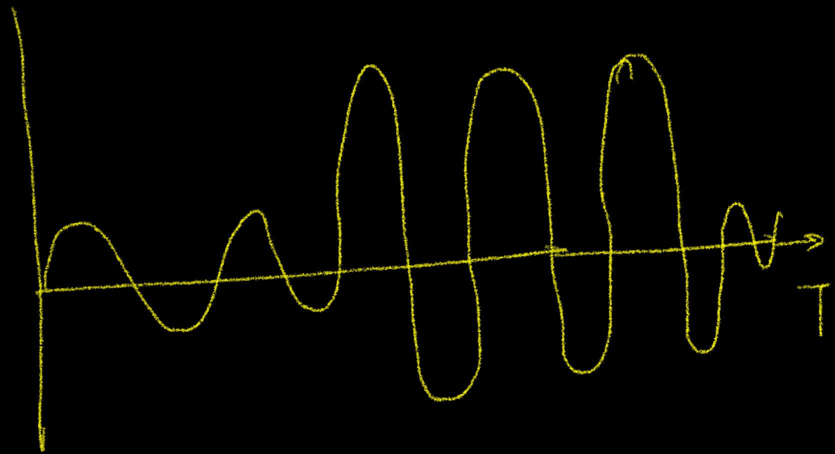

time

1. mean constant over time

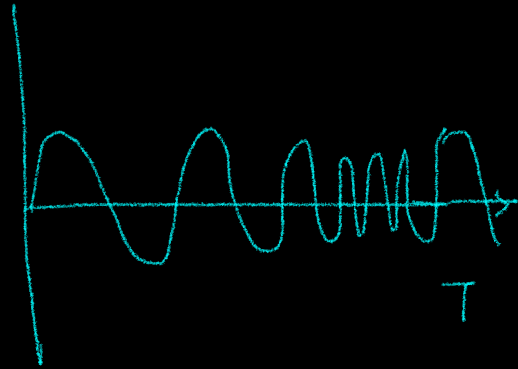


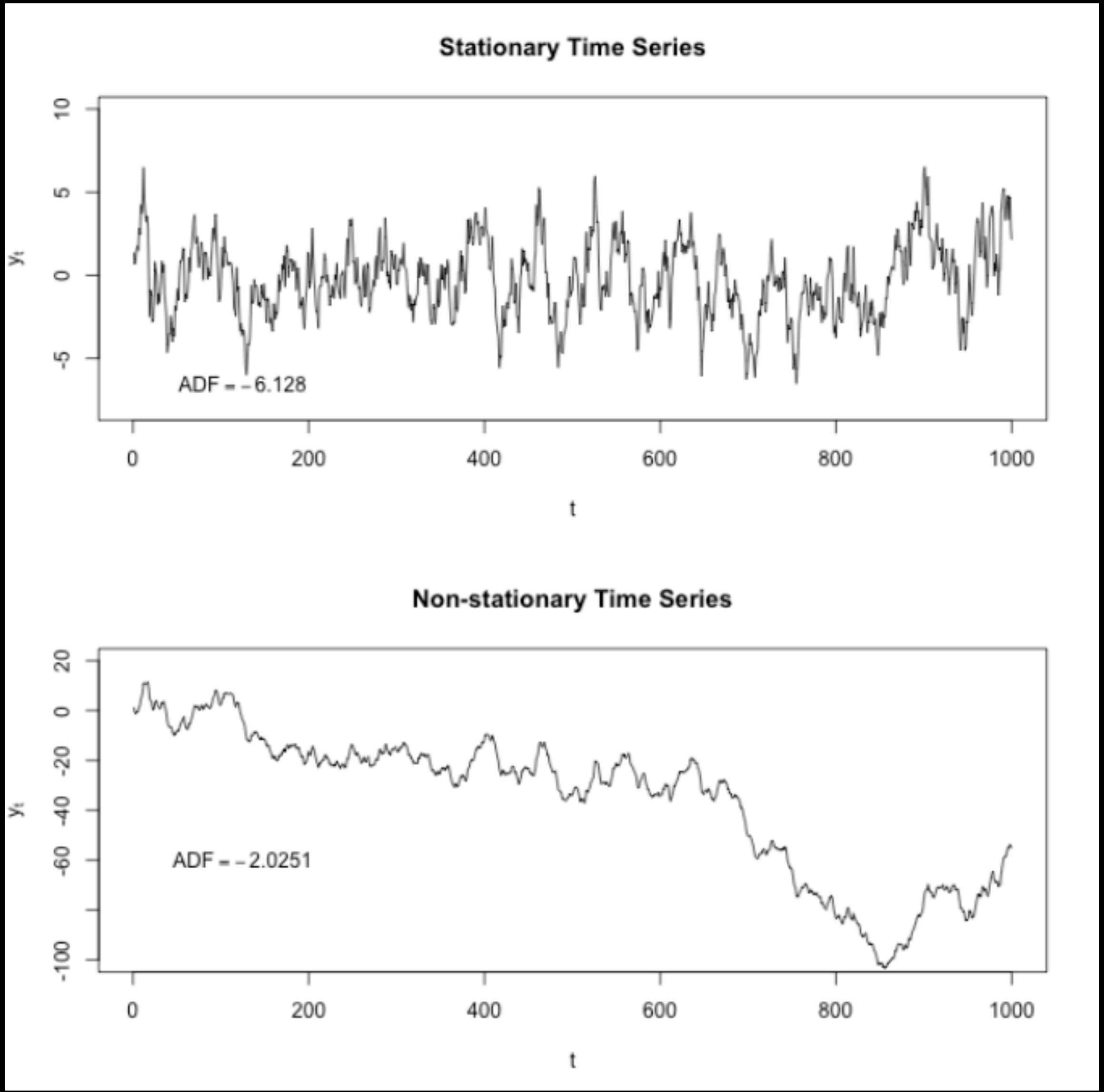
Trending

2. Variance constant



3. co-variance is only a function of gap.





wikipedia

Augmented Dickey-Fuller test (ADF)

(or)

Simply-plot

Notation:

$y_1, y_2, \ldots\ldots y_{t-1}, y_t, y_{t+1}, \ldots\ldots$ ← observations

$\ldots\ldots \hat{y}_{t-1}, \hat{y}_t, \hat{y}_{t+1}, \ldots\ldots$ ← Predictions

$\ldots\ldots, \epsilon_{t-1}, \epsilon_t, \epsilon_{t+1}, \ldots\ldots$ ← errors

$\epsilon_t = y_t - \hat{y}_t$

μ ← mean-value of y_i's.

Auto Regressive (AR) model:

AR(p):

$$\underbrace{y_t}_{} = \underbrace{\mu}_{\text{constant}} + \underbrace{\alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \dots + (\alpha_p y_{t-p})}_{\alpha = [\alpha_1\ \alpha_2 \dots \alpha_p] \text{ is the parameter}} + \underbrace{\epsilon_t}_{\text{error}}$$

* AR(p) is like a linear regression model on previous 'p' values in the series.

Moving Average Model: MA(q)

$$y_t = \underbrace{\mu}_{c} + \underbrace{\epsilon_t}_{\text{error}} + \underbrace{\theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q}}$$

→ $\epsilon_i = y_i - \hat{y}_i$

→ $\theta = [\theta_1\ \theta_2 \cdots \theta_q]$ are params

→ Linear regression of previous q-errors

Differencing:

→ one of the strategies to make a time series stationary

like Δ or d/dx

$$y_t' = y_t - y_{t-1}$$ (1st order)

$$y_t'' = y_t' - y_{t-1}'$$ (2nd order)

$$= y_t - 2y_{t-1} + y_{t-2}$$

APPLIED COURSE

$$y_t = y'_t + y_{t-1}$$

(Re-construction)

$$y_t = y''_t + 2y_{t-1} - y_{t-2}$$

d: # times we have to difference

→ try 1, 2, 3, . . . to find the best value

Log-Transform (to make a series stationary)

$$y_t = \log(y_t)$$

ARIMA(p, q, d)

→ I: Integrated [opposite of differencing]

ARMA(p, q):

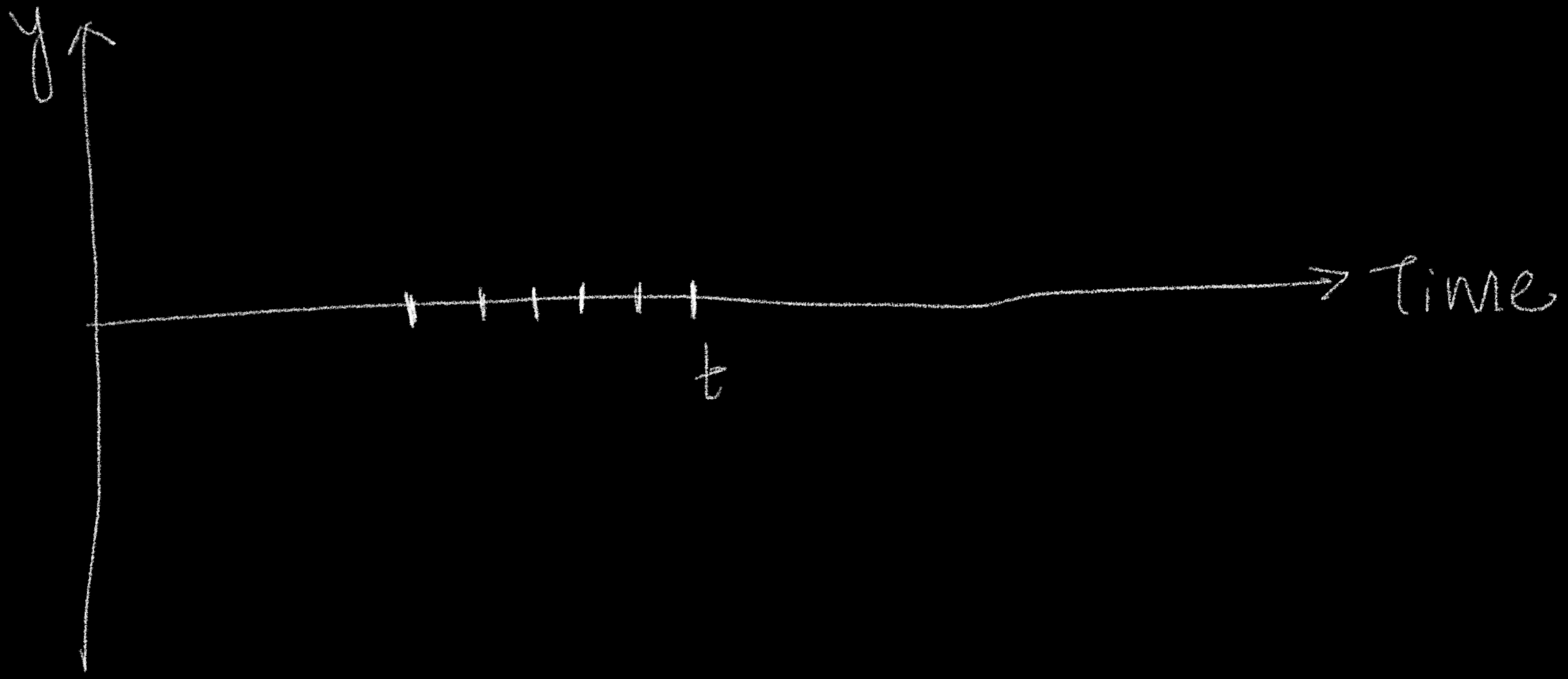
$$y_t = \mu + (\alpha_1 y_{t-1} + \cdots + \alpha_p y_{t-p}) + (\theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots \theta_q \epsilon_{t-q}) + \epsilon_t$$

→ p, q & d are hyper-params

→ Linear regression model on previous 'p' values & 'q' errors post differencing.

→ a.k.a Box-Jenkins model (1976)

Auto Correlation & ACF:



ACF
PACF
→ https://machinelearningmastery.com/gentle-introduction-autocorrelation-partial-autocorrelation/

ACF/PACF can help us guess p in ARIMA (p,q,d)

| | |
t-x t

x is a good guess for p.

[p,q,d : hyper-param tuning]

https://machinelearningmastery.com/arima-for-time-series-forecasting-with-python/

→ ARIMA in Python

→ Analysis of errors.

* ARIMA is a simple linear model.

Topics already covered

1. Fourier transform: (Periodicity)

https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3103/fourier-decomposition/5/module-5-feature-engineering-productionization-and-deployment-of-ml-models

https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3206/data-preparation-time-series-and-fourier-transforms/7/module-6-machine-learning-real-world-case-studies

② Forecasting as Regression:

features: $y_{t-1}, y_{t-2}, y_{t-3}, \ldots, y_{t-p}$

$\epsilon_{t-1}, \epsilon_{t-2}, \epsilon_{t-3}, \ldots, \epsilon_{t-p}$

DOW, TOD, isHoliday, ...

(*) Using differencing /Transformation + Regression is a good strategy

(4) ARIMA is a special case.

Taxi-Demand Prediction Case-Study:

https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3189/businessreal-world-problem-overview/7/module-6-machine-learning-real-world-case-studies

③ State of the art: LSTMs; GRUs,

Attention-models

Transformers

Start:

https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/3430/why-rnns/8/module-8-neural-networks-computer-vision-and-deep-learning