

# Jaringan dan Infrastruktur Sistem Cerdas (JSC)

## Pertemuan 6 : Dasar Pengolahan Data

Alauddin Maulana Hirzan

Fakultas Teknologi Informasi dan Komunikasi  
Universitas Semarang

# Outline

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

- 1 Dasar Pengolahan Data
- 2 Moving Average
- 3 Low-Pass Filter
- 4 Mikrokontroler ESP8266 dan Jaringan WiFi
- 5 Lastly

# Data Sensor

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

Data yang dihasilkan oleh sensor umumnya memiliki tiga karakteristik utama, yaitu noise, fluktuasi, dan delay. *Noise* merupakan gangguan acak yang menyebabkan nilai sensor tidak stabil meskipun kondisi sebenarnya tetap. Fluktuasi adalah perubahan nilai yang naik-turun secara cepat dalam waktu singkat, sedangkan delay adalah keterlambatan respon sensor terhadap perubahan kondisi lingkungan.

Contoh: sensor suhu membaca 30°C, lalu tiba-tiba 32°C, kemudian 29°C dalam beberapa detik.

# Data Timeseries

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

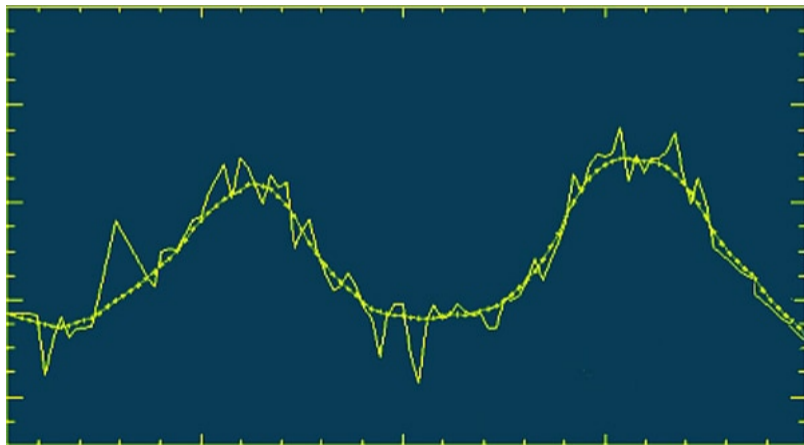
Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

Data sensor biasanya berbentuk time series, yaitu data yang dikumpulkan secara berurutan berdasarkan waktu. Setiap data memiliki timestamp tertentu, dan jarak antar pengambilan data disebut sampling rate. Semakin tinggi sampling rate, semakin detail data yang diperoleh.

Contoh: sensor suhu mencatat data setiap 1 detik atau setiap 1 menit.

# Contoh Noise



Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

# Permasalahan Data Sensor

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

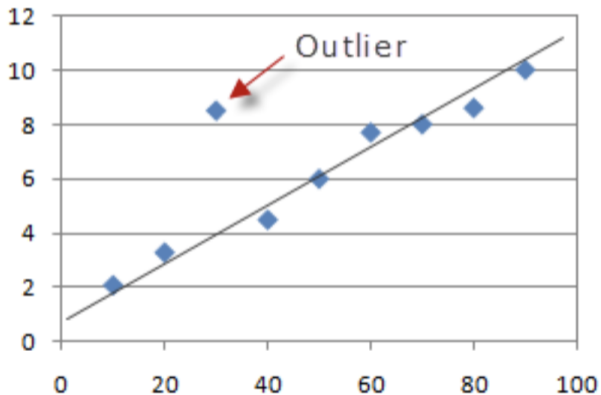
Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

Data mentah dari sensor sering kali mengandung masalah seperti outlier, noise tinggi, dan ketidakstabilan. *Outlier* adalah nilai yang menyimpang jauh dari pola normal, misalnya pembacaan tiba-tiba sangat tinggi atau rendah. *Noise* tinggi menyebabkan data sulit dianalisis, sementara ketidakstabilan membuat sistem sulit mengambil keputusan yang akurat.

Contoh: sensor pH biasanya membaca 6–7, tetapi tiba-tiba muncul nilai 12.

# Contoh Outlier



# Filtering

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

*Filtering* bertujuan untuk memperbaiki kualitas data sensor dengan mengurangi noise dan menstabilkan sinyal. Dengan proses ini, data menjadi lebih halus (smooth) dan representatif terhadap kondisi sebenarnya, sehingga dapat digunakan untuk analisis atau pengambilan keputusan.

Filtering sangat berguna untuk membuat Data Training model cerdas, khususnya prediksi regresi

Contoh: data suhu yang awalnya naik-turun drastis menjadi lebih konsisten setelah difilter.

# Contoh Filtering

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly



Complete Data



Code which filters the  
data on certain condition



Required Subset of Data

# Jenis Filtering

Beberapa metode filtering yang umum digunakan:

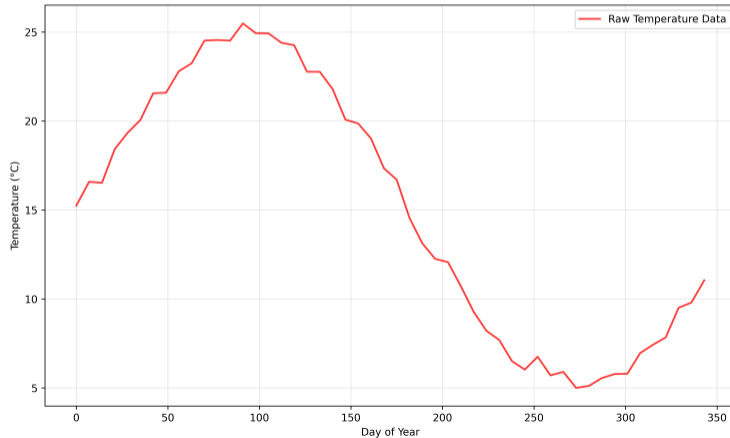
- 1 low-pass filter,
- 2 moving average, dan
- 3 median filter.

Low-pass filter hanya melewatkan perubahan lambat dan menghilangkan perubahan cepat (noise). Moving average menghitung rata-rata dari beberapa data terakhir untuk menghasilkan nilai yang lebih stabil. Median filter mengambil nilai tengah untuk menghilangkan outlier.

Contoh: data [30, 31, 100, 32, 30] akan menjadi lebih stabil dengan median filter (nilai 31 atau 32).

# Ilustrasi Filtering

## Data Mentah



Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

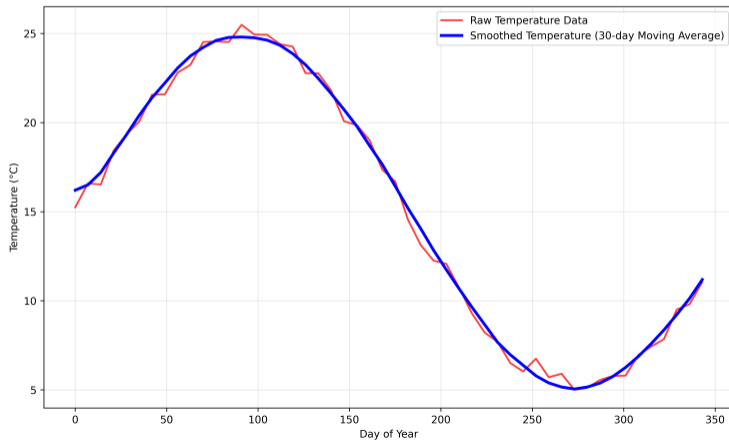
Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

# Ilustrasi Filtering

## Moving Average



Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

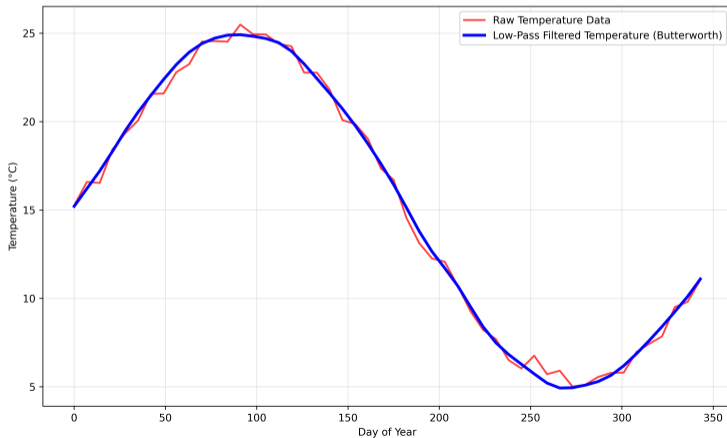
Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

# Ilustrasi Filtering

## Low-Pass Filter



Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

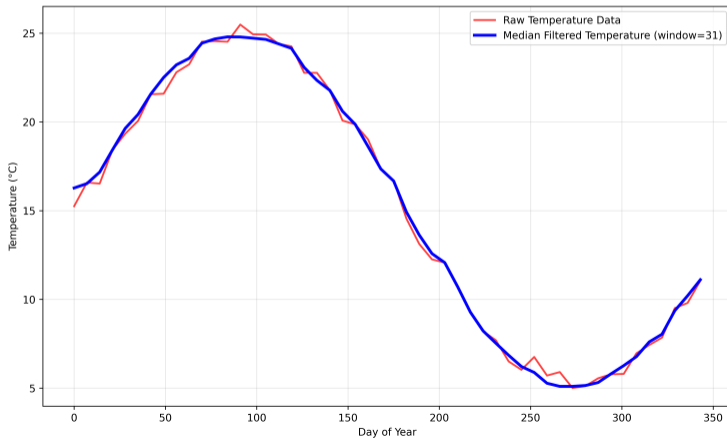
Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

# Ilustrasi Filtering

## Median Filter



Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

# Resiko Filtering

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

Dalam proses filtering terdapat trade-off antara kecepatan respons (responsiveness) dan kehalusan data (smoothing). Filter yang terlalu halus akan mengurangi noise tetapi menyebabkan respon lambat terhadap perubahan nyata. Sebaliknya, filter yang responsif cepat menangkap perubahan tetapi masih mengandung noise.

- Data Mentah -> Filtering -> Noise Hilang -> Kualitas Data Bagus
- Data Mentah -> Tanpa Filtering -> Noise Ada -> Kualitas Kurang

# Definisi

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

Moving Average adalah metode filtering yang menghitung rata-rata dari sejumlah data terakhir sebanyak  $N$  untuk menghasilkan nilai yang lebih halus. Setiap kali data baru masuk, data lama akan “digeser keluar” dari perhitungan.

Contoh: jika  $N = 3$ , maka setiap output dihitung dari 3 data terakhir.

# Rumus Moving Average

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

## Perhitungan Manual

$$y[n] = \frac{1}{N} \sum_{i=0}^{N-1} x[n - i]$$

### Contoh

Jika

data = [30, 32, 28]

maka  $N = 3$  data adalah

$(30+32+28)/3 = 30$

# Contoh Perhitungan

Diketahui:

$$X = [10, 20, 15, 12, 15, 17, 14, 12]$$

Hitung *moving average* dengan N adalah 2 adalah

- 1 Langkah 1 :  $(10 + 20) / 2 = 15$
- 2 Langkah 2 :  $(20 + 15) / 2 = 17.5$
- 3 Langkah 3 :  $(15 + 12) / 2 = 18.5$
- 4 dst

Hasil akhir:  $X_{ma} [15, 17.5, 18.5, \dots]$

# Kelebihan dan Kekurangan Moving Average

## Kelebihan

Metode ini memiliki kelebihan utama yaitu sederhana dan menghasilkan sinyal yang stabil. Perhitungannya ringan sehingga sangat cocok untuk mikrokontroler seperti ESP8266 atau ESP32.

## Kekurangan

Kekurangan utama adalah adanya delay yang cukup tinggi, terutama jika nilai  $N$  besar. Hal ini menyebabkan sistem menjadi kurang responsif terhadap perubahan mendadak.

# Definisi

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

Low-pass filter adalah metode filtering yang hanya melewatkan perubahan sinyal yang lambat (low frequency) dan meredam perubahan cepat yang biasanya merupakan noise. Filter ini sangat umum digunakan pada sistem embedded karena ringan dan tidak membutuhkan banyak memori.

Contoh: sensor suhu yang berubah perlahan akan dipertahankan, sedangkan lonjakan tiba-tiba akan diredam.

# Rumus Low Pass Filter

$$y[n] = \alpha x[n] + (1 - \alpha)y[n - 1]$$

## Penjelasan

Nilai output saat ini merupakan kombinasi antara data terbaru (input) dan hasil sebelumnya (state). Parameter  $\alpha$  (alpha) menentukan seberapa besar pengaruh data baru terhadap hasil akhir.

Contoh: jika  $\alpha = 0.3$ , maka 30% berasal dari data baru dan 70% dari nilai sebelumnya.

# Parameter Alpha

Parameter  $\alpha$  memiliki rentang antara 0 hingga 1 dan sangat menentukan karakteristik filter.

- 1**  $\alpha$  kecil (misalnya 0.1) -> hasil sangat halus, tetapi lambat merespon perubahan
- 2**  $\alpha$  besar (misalnya 0.8) -> respons cepat, tetapi masih mengandung noise

Contoh:

- 1**  $\alpha = 0.1$  cocok untuk monitoring stabil (misalnya suhu ruangan)
- 2**  $\alpha = 0.7$  cocok untuk sistem real-time (misalnya kontrol motor)

# Contoh Perhitungan

Misalkan data sensor:

$$X = [30, 40, 35]$$

dengan  $\alpha = 0.5$  dan nilai awal  $y[0] = 30$

- Langkah 1 :  $y[1] = 0.5(40) + 0.5(30) = 35$
- Langkah 2 :  $y[2] = 0.5(35) + 0.5(35) = 35$

Terlihat bahwa nilai ekstrem (40) tidak langsung mendominasi hasil, tetapi “diredam” secara bertahap.

# Kelebihan dan Kekurangan

## Kelebihan

Low-pass filter memiliki kelebihan utama yaitu respons yang relatif cepat dan hasil yang cukup halus (smooth). Selain itu, metode ini sangat ringan sehingga cocok untuk mikrokontroler dengan resource terbatas seperti ESP8266 atau ESP32.

## Kekurangan

Kelemahan utama dari low-pass filter adalah tidak mampu menghilangkan outlier ekstrem secara efektif. Jika terdapat nilai yang sangat menyimpang, hasil filter tetap akan terpengaruh karena metode ini menggunakan pembobotan, bukan seleksi nilai.

# Definisi

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

## ESP8266

ESP8266 adalah mikrokontroler dengan modul Wi-Fi terintegrasi yang dirancang untuk aplikasi Internet of Things (IoT). Perangkat ini memungkinkan mikrokontroler terhubung langsung ke jaringan internet tanpa modul tambahan.

## ESP32

ESP32 adalah pengembangan lanjutan dari ESP8266 dengan performa lebih tinggi serta fitur tambahan seperti Bluetooth. ESP32 banyak digunakan untuk aplikasi IoT yang lebih kompleks dan real-time.

# Ilustrasi

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly



# Komparasi Spesifikasi

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

Komponen	ESP8266	ESP32
CPU	32-bit Tensilica L106	Dual-core Xtensa LX6
Clock Speed	80–160 MHz	240 MHz
RAM	~50 KB (usable)	520 KB SRAM
Wi-Fi	802.11 b/g/n	802.11 b/g/n
Use Case	IoT sederhana	IoT kompleks, AIoT

# Mode Operasi

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

## Jaringan

- 1 Station Mode (STA)
- 2 Access Point Mode (AP)
- 3 STA + AP Mode

## Daya

- 1 Active mode
- 2 Light sleep
- 3 Deep sleep
- 4 Modem sleep (ESP32)
- 5 Hibernation (ESP32)

# Proses Koneksi Wi-Fi

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

Proses koneksi perangkat ke Wi-Fi terdiri dari tiga tahap utama:

- 1 *Scan* → mencari jaringan yang tersedia
- 2 *Authenticate* → verifikasi password
- 3 *Connect* → bergabung ke jaringan

Setelah berhasil terhubung ke jaringan, perangkat akan meminta IP Address dari DHCP server (biasanya router). IP ini digunakan untuk komunikasi lebih lanjut seperti pengiriman data ke server.

# Kode Koneksi Internet

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

## MicroPython

```
ssid = "Nama_WiFi"  
password = "Password_WiFi"  
  
wlan = network.WLAN(network.STA_IF)  
wlan.active(True)  
wlan.connect(ssid, password)  
  
while not wlan.isconnected():  
    time.sleep(1)  
print("Connected!")
```

# Kode Koneksi Internet

Jaringan dan  
Infrastruktur  
Sistem  
Cerdas  
(JSC)

Alauddin  
Maulana  
Hirzan

Dasar  
Pengolahan  
Data

Moving  
Average

Low-Pass  
Filter

Mikrokontroler  
ESP8266  
dan Jaringan  
WiFi

Lastly

## Arduino IDE

```
const char* ssid = "Nama_WiFi";  
const char* password = "Password_WiFi";  
  
WiFi.begin(ssid, password);  
  
while (WiFi.status() != WL_CONNECTED) {  
    delay(500);  
}
```

# Terima Kasih