# Level Measurement defect classification using ML

## Features for the dataset :

1.water_level → directly measured by the tank sensor in the model.

2.level_rate → can be computed using a derivative block on water_level.

3.valve_position → available as the actuator output (pump control voltage or valve opening).

4.pump_status → you can create a binary signal based on whether the pump voltage is above a threshold.

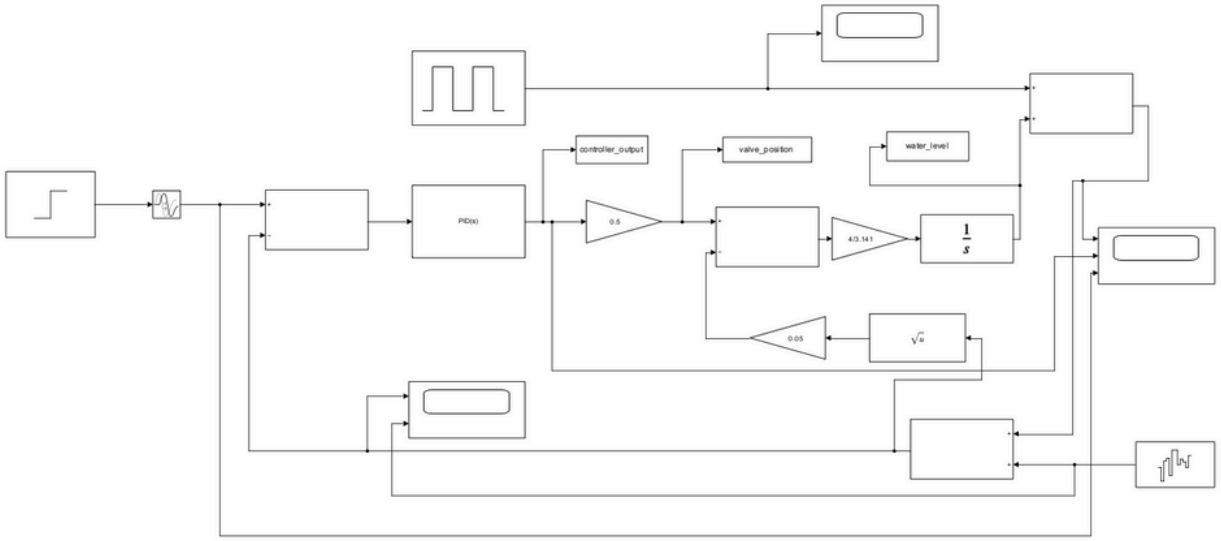
5.controller_output → PID controller output is available.

6.setpoint → the reference water level input to the system.

7.error_signal → can be computed as setpoint - water_level.

## Target Values :

| Output Label | Description |
|---|---|
| 0 | Normal operation |
| 1 | Leakage |
| 2 | Overflow |
| 3 | Clogged valve |
| 4 | Pump failure |
| 5 | Sensor fault |
| 6 | Controller fault |




Normal model (water_level_control_original.slx)

## Faults:

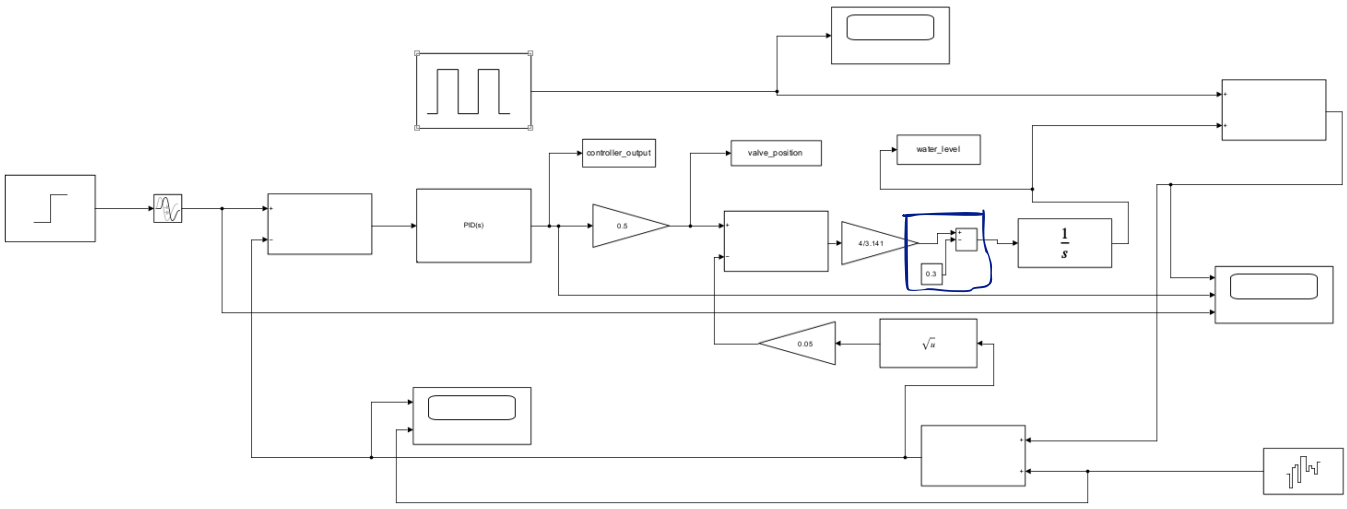
### 1 Leakage

Water leaks OUT of the tank continuously

Water level drops faster than normal

Pump works harder (controller output increases)

Valve opens more to compensate



**2 Overflow**

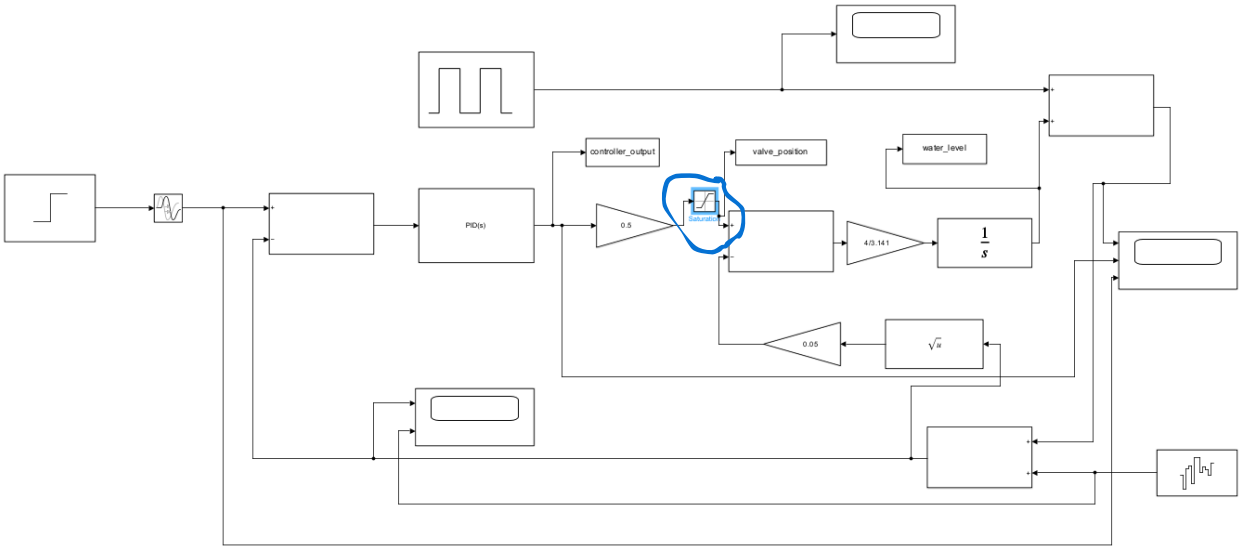
Water level rises above normal setpoint

Too much inflow OR restricted outflow

Controller tries to close valve

Upper limit: 0.4 (valve can only open 40%)

Lower limit:0




**3 CLOGGED VALVE FAULT**

Valve stuck partially closed

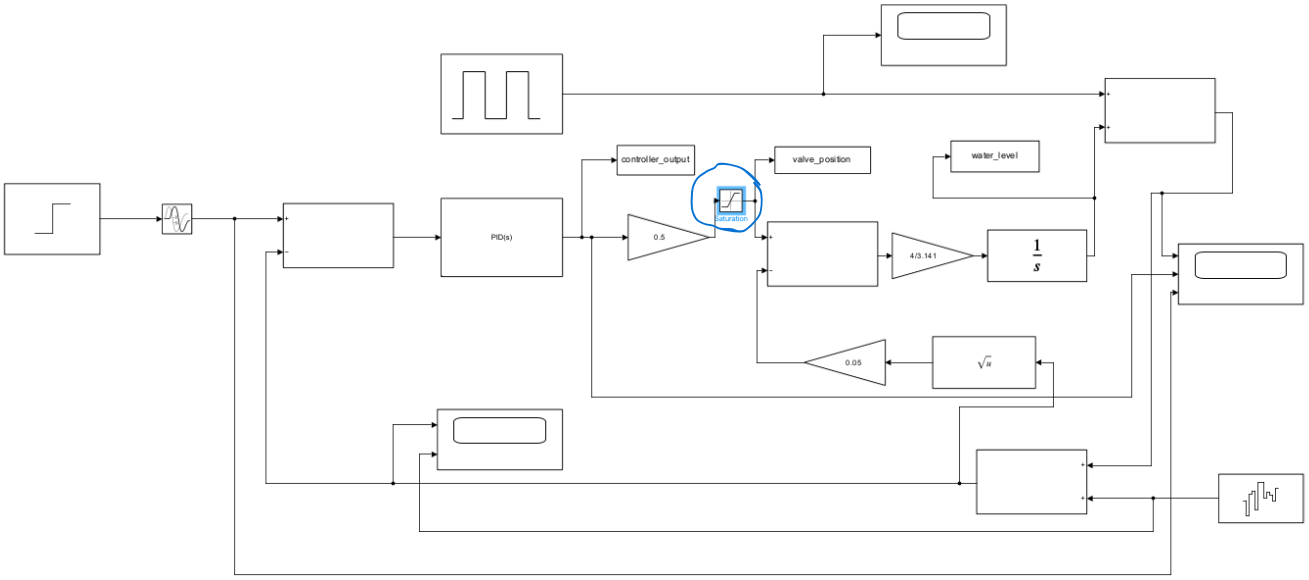
Water cannot flow out properly

Water level rises

Controller output tries to open valve more (but fails)

Upper limit: 0.25 (valve stuck at 25% max opening)

Lower limit: 0




## 4 PUMP FAILURE FAULT

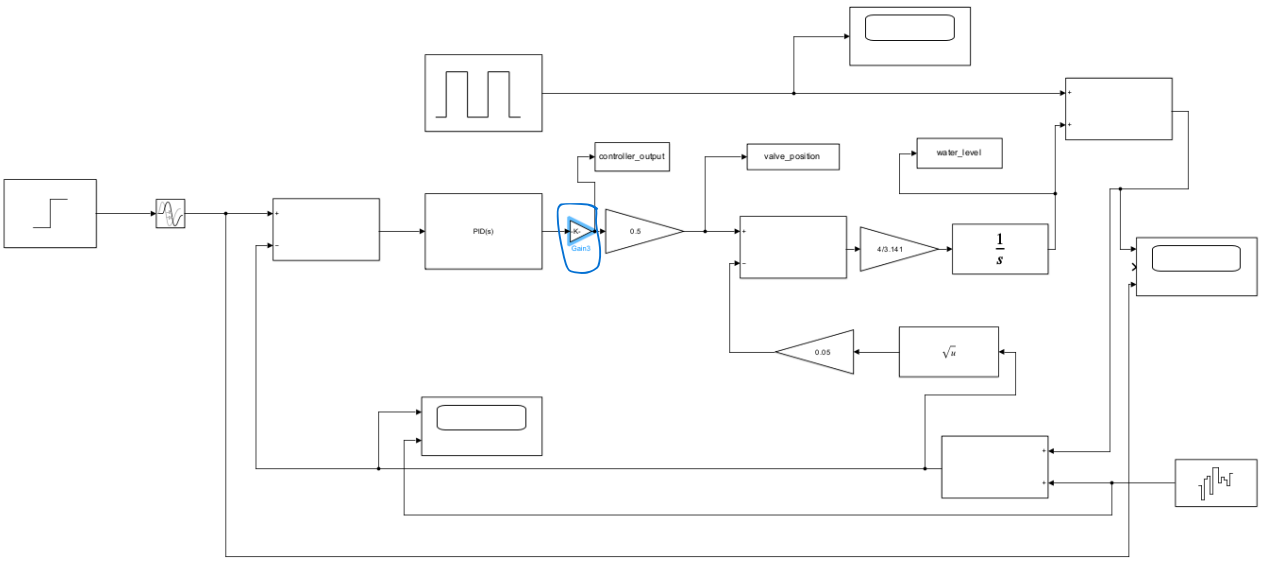
Pump not working at full capacity

Less water pumped into tank

Water level drops

Controller output maxes out trying to compensate

Gain value: 0.25 (pump only 25% effective)




## 5 SENSOR FAULT

Level sensor gives noisy/incorrect readings

Controller receives wrong information

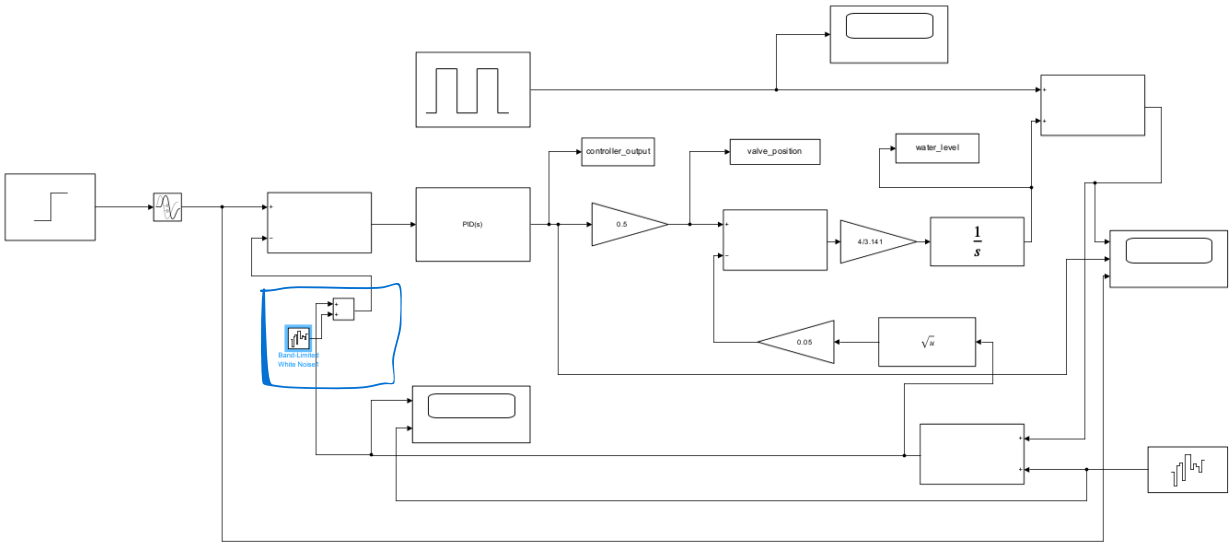
Erratic control actions

Actual level may be OK, but measured level is noisy

Noise power: 1.5 (high noise)[Band limited white noise]

Sample time**:** 0.1

Click OK




**6 CONTROLLER FAULT**

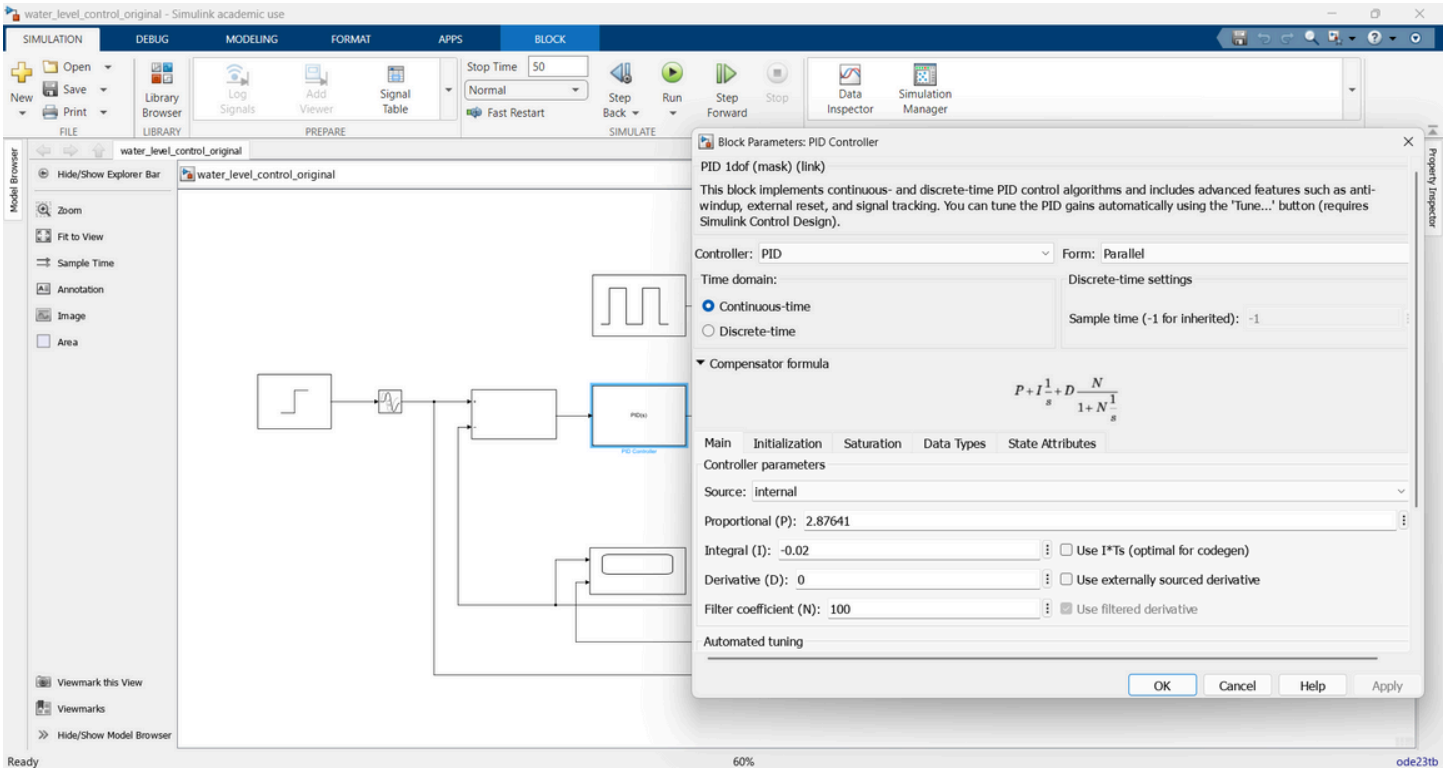
PID controller poorly tuned

Wrong gains cause poor control

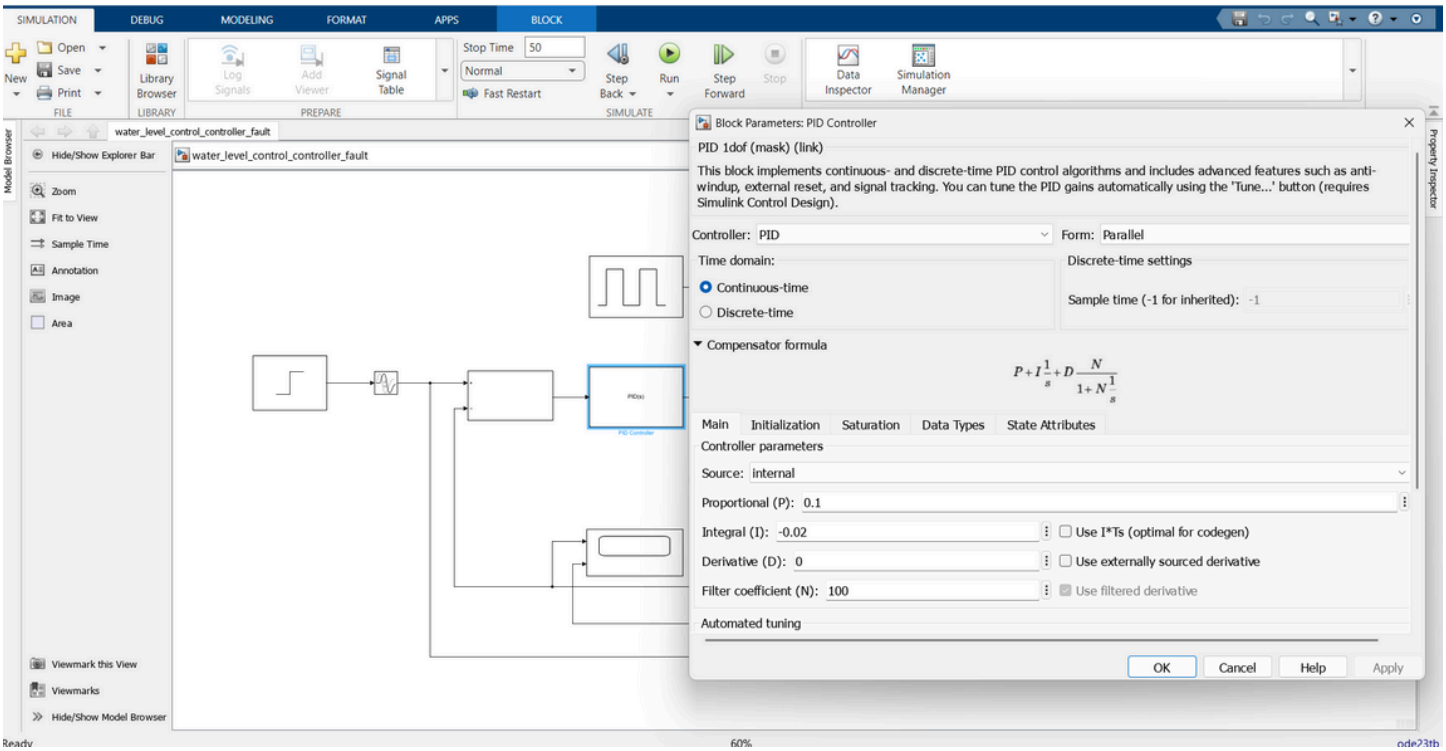
Oscillations, slow response, or instability

System never settles properly

proportional gain is decreased(kp).




original gain values




fault controller gain values

**##Code for generating values from the simulink model**

```m
function save_fault_data(fault_name, fault_label, setpoint_value, ...
                         water_level, controller_output, valve_position, tout)
    %% Save fault data from simulation signals
    % fault_name      : string (e.g., 'Normal', 'Leak', etc.)
    % fault_label     : integer label (e.g., 0 for normal, 1 for fault)
    % setpoint_value  : numeric setpoint
    % water_level     : vector of water level values
    % controller_output : vector of controller outputs
    % valve_position  : vector of valve positions
    % tout            : vector of time values

    fprintf('Processing %s data (Label: %d)...\n', fault_name, fault_label);

    % Ensure column vectors
    wl = water_level(:);
    co = controller_output(:);
    vp = valve_position(:);
    time = tout(:);

    % Calculate sampling time
    dt = mean(diff(time));

    % Calculate derived features
    level_rate = [0; diff(wl)] / dt;           % derivative of water level
    pump_status = double(co > 0.1);            % binary pump status

    % Create setpoint vector
    setpoint = setpoint_value * ones(size(wl));

    % Calculate error signal
    error_signal = setpoint - wl;

    % Combine all 7 features into matrix
    features = [wl, level_rate, vp, pump_status, co, setpoint, error_signal];
```

```m
    % Create labels (all same for this fault type)
    labels = fault_label * ones(length(wl), 1);

    % Save to .mat file
    filename = sprintf('fault_data_%s_label%d.mat', fault_name, fault_label);
    save(filename, 'features', 'labels', 'fault_name', 'fault_label', 'time');

        % Save to Excel file also
    excel_filename = sprintf('fault_data_%s_label%d.xlsx', fault_name, fault_label);

    % Convert to table with headers for readability
    T = array2table(features, 'VariableNames', ...
        {'WaterLevel', 'LevelRate', 'ValvePosition', 'PumpStatus', ...
         'ControllerOutput', 'Setpoint', 'ErrorSignal'});

    % Add time and labels as extra columns
    T.Time   = time;
    T.Labels = labels;

    % Write to Excel
    writetable(T, excel_filename);

    fprintf('Excel file saved: %s\n', excel_filename);


    % Display summary
    fprintf('Saved: %s\n', filename);
    fprintf('   Samples: %d\n', length(labels));
    fprintf('   Duration: %.1f seconds\n', max(time));
    fprintf('   Water level range: [%.2f, %.2f] m\n', min(wl), max(wl));
    fprintf('   Controller output range: [%.2f, %.2f]\n', min(co), max(co));
    fprintf('   Valve position range: [%.2f, %.2f]\n', min(vp), max(vp));
    fprintf('\n');
end
```

**Data Generated snap :**

| | WaterLevel | LevelRate | ValvePosition | PumpStatus | ControllerOutput | Setpoint | ErrorSignal | Time | Labels |
|---|---|---|---|---|---|---|---|---|---|
| 2 | 250 | 0 | -361.1726978 | 0 | -722.3453955 | 500 | -245 | 0 | 0 |
| 3 | 249.897732 | -11.29647644 | -361.0250589 | 0 | -722.0501178 | 500 | -244.897732 | 0.000221905 | 0 |
| 4 | 249.387021 | -56.41323904 | -360.2877681 | 0 | -720.5755362 | 500 | -244.387021 | 0.001331433 | 0 |
| 5 | 247.909664 | -163.1888296 | -358.1549775 | 0 | -716.3099549 | 500 | -242.909664 | 0.004553818 | 0 |
| 6 | 245.698415 | -244.2545531 | -354.962701 | 0 | -709.925402 | 500 | -240.698415 | 0.009412942 | 0 |
| 7 | 242.906302 | -308.4167966 | -350.9318578 | 0 | -701.8637157 | 500 | -237.906302 | 0.015611216 | 0 |
| 8 | 240.14586 | -304.9183736 | -346.9467368 | 0 | -693.8934735 | 500 | -235.14586 | 0.021809489 | 0 |
| 9 | 236.527826 | -399.6480721 | -341.7235481 | 0 | -683.4470963 | 500 | -231.527826 | 0.030041933 | 0 |
| 10 | 232.9642 | -393.6381741 | -336.5789048 | 0 | -673.1578097 | 500 | -227.9642 | 0.038274377 | 0 |
| 11 | 228.716164 | -469.2380213 | -330.4462103 | 0 | -660.8924206 | 500 | -223.716164 | 0.048253668 | 0 |
| 12 | 224.545443 | -460.6978332 | -324.4251303 | 0 | -648.8502606 | 500 | -219.545443 | 0.058232958 | 0 |
| 13 | 220.45063 | -452.3129881 | -318.5136346 | 0 | -637.0272691 | 500 | -215.45063 | 0.068212249 | 0 |
| 14 | 216.430346 | -444.0806618 | -312.7097298 | 0 | -625.4194597 | 500 | -211.430346 | 0.07819154 | 0 |
| 15 | 212.483233 | -435.9980806 | -307.011459 | 0 | -614.022918 | 500 | -207.483233 | 0.088170831 | 0 |
| 16 | 208.607961 | -428.0625216 | -301.4169006 | 0 | -602.8338011 | 500 | -203.607961 | 0.098150122 | 0 |
| 17 | 207.897393 | -78.48939231 | -309.6241798 | 0 | -619.2483595 | 500 | -202.897393 | 0.1 | 0 |
| 18 | 204.272612 | -400.3932327 | -304.3912432 | 0 | -608.7824863 | 500 | -199.272612 | 0.109249392 | 0 |
| 19 | 200.709022 | -393.6341836 | -299.2466428 | 0 | -598.4932856 | 500 | -195.709022 | 0.118498783 | 0 |
| 20 | 197.205589 | -386.989166 | -294.1888883 | 0 | -588.3777766 | 500 | -192.205589 | 0.127748175 | 0 |
| 21 | 193.7613 | -380.4562567 | -289.2165145 | 0 | -578.4330291 | 500 | -188.7613 | 0.136997567 | 0 |
| 22 | 190.375155 | -374.0335649 | -284.328081 | 0 | -568.6561621 | 500 | -185.375155 | 0.146246959 | 0 |
| 23 | 187.046174 | -367.7192316 | -279.5221716 | 0 | -559.0443432 | 500 | -182.046174 | 0.15549635 | 0 |
| 24 | 183.773392 | -361.5114293 | -274.7973939 | 0 | -549.5947879 | 500 | -178.773392 | 0.164745742 | 0 |
| 25 | 180.555862 | -355.4083611 | -270.1523793 | 0 | -540.3047585 | 500 | -175.555862 | 0.173995134 | 0 |
| 26 | 177.392652 | -349.4082606 | -265.5857818 | 0 | -531.1715637 | 500 | -172.392652 | 0.183244525 | 0 |
| 27 | 174.282844 | -343.5093912 | -261.0962786 | 0 | -522.1925573 | 500 | -169.282844 | 0.192493917 | 0 |