Sound sensor variability test

2025-04-03

* Two identical sound sensors were used for the test.
* Both sensors were connected to an ADS1115 ADC module.
* A constant sound was played near the sensors for 2 minutes.
* Sensor readings were recorded continuously during this period.
* Each reading was timestamped using the Arduino millis() function.
* Data was saved to an SD card in the format: timestamp, sensor1, sensor2.
* The recorded ADC values were later converted to voltage using the formula:  
  (ADC reading / 32768) \* 6.144.
* Readings were filtered to include only data up to 147019 milliseconds.
* Mean and standard deviation were calculated for both sensors.
* The goal was to assess variability and consistency between the two sensors.

knitr::opts\_chunk$set(echo = TRUE)  
library(readr)  
# Read the log file containing time-series sound sensor data.  
# Assign column names: timestamp (in milliseconds), sensor1, and sensor2  
data <- read\_csv("LOG.TXT", col\_names = c("timestamp", "sensor1", "sensor2"))

## Rows: 7420 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## dbl (3): timestamp, sensor1, sensor2  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

# Filter data to include only entries up to 147019 milliseconds  
filtered\_data <- data[data$timestamp <= 147019, ]  
  
# Define voltage reference based on ADC gain setting (GAIN\_TWOTHIRDS = ±6.144V)  
VREF <- 6.144  
  
# Convert raw ADC readings to voltages using:  
# Voltage = (raw reading / 32768.0) \* VREF  
filtered\_data$Sensor1\_Voltage <- (filtered\_data$sensor1 / 32768.0) \* VREF  
filtered\_data$Sensor2\_Voltage <- (filtered\_data$sensor2 / 32768.0) \* VREF  
  
# Calculate mean and standard deviation in volts for Sensor 1  
sensor1\_mean <- mean(filtered\_data$Sensor1\_Voltage, na.rm = TRUE)  
sensor1\_sd <- sd(filtered\_data$Sensor1\_Voltage, na.rm = TRUE)  
  
# Calculate mean and standard deviation in volts for Sensor 2  
sensor2\_mean <- mean(filtered\_data$Sensor2\_Voltage, na.rm = TRUE)  
sensor2\_sd <- sd(filtered\_data$Sensor2\_Voltage, na.rm = TRUE)  
  
# Display summary statistics in volts  
cat("Sensor 1 (Volts):\n")

## Sensor 1 (Volts):

cat(" Mean =", round(sensor1\_mean, 4), "V\n")

## Mean = 1.7448 V

cat(" Standard Deviation =", round(sensor1\_sd, 4), "V\n\n")

## Standard Deviation = 0.0984 V

cat("Sensor 2 (Volts):\n")

## Sensor 2 (Volts):

cat(" Mean =", round(sensor2\_mean, 4), "V\n")

## Mean = 1.7491 V

cat(" Standard Deviation =", round(sensor2\_sd, 4), "V\n")

## Standard Deviation = 0.1121 V