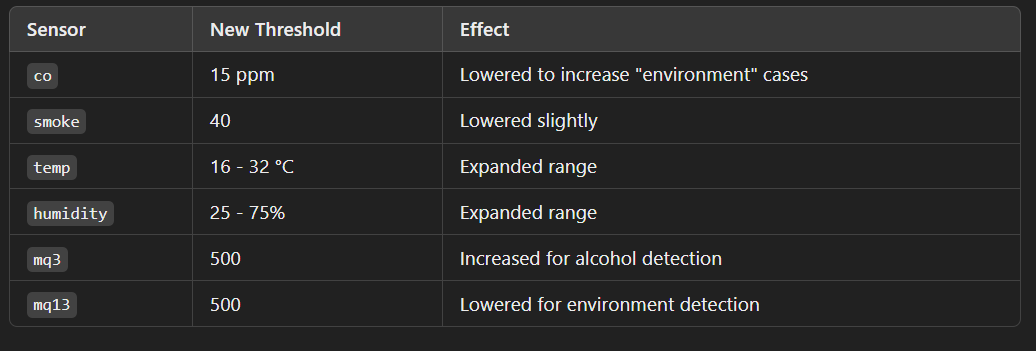
[https://www.kaggle.com/datasets/garystafford/environmental-sensor-data-132k](https://www.kaggle.com/datasets/garystafford/environmental-sensor-data-132k" \t "_blank)

<https://github.com/TakMashhido/Gas-Sensors-Measurements-Dataset/blob/main/Gas_Sensors_Measurements.csv>



**How the Dataset Was Created**

**Step 1: Extract Relevant Data from CSVs**

* The two datasets were **merged** by aligning rows from:
  + **Gas\_Sensors\_Measurements.csv** → (mq3, mq13)
  + **iot\_telemetry\_data.csv** → (co, smoke, temp, humidity)
* Since mq3 values started from **400**, the threshold for alcohol detection was **adjusted** accordingly.

**Step 2: Define Drowsiness Thresholds**

**Thresholds were adjusted** to ensure a **balanced distribution** between alcohol and environment-related drowsiness.

| **Sensor** | **New Threshold** | **Effect** |
| --- | --- | --- |
| co | 15 ppm | Lowered to increase "environment" cases |
| smoke | 40 | Lowered slightly |
| temp | 16 - 32 °C | Expanded range |
| humidity | 25 - 75% | Expanded range |
| mq3 | 500 | Increased for alcohol detection |
| mq13 | 500 | Lowered for environment detection |

**Step 3: Assign Drowsiness Labels**

A function **classified drowsiness causes** based on thresholds:

* If **both alcohol and environment** exceeded thresholds → "both"
* If **only alcohol** exceeded thresholds → "alcohol"
* If **only environmental factors** exceeded thresholds → "environment"
* If **none of the thresholds** were exceeded → "none"

**Step 4: Add Noise for Realism**

To **simulate sensor fluctuations**, Gaussian noise was added to all numerical values:

📌 **Formula Used for Noise:**

noise=N(0,0.05×σ)\text{noise} = \mathcal{N}(0, 0.05 \times \sigma)noise=N(0,0.05×σ)

* **5% of standard deviation** was used as noise variation.
* This makes the dataset more **realistic and generalizable**.

**Step 5: Balance the Dataset**

To avoid **bias towards any class**, we **sampled equal numbers** from each class (alcohol, environment, both, none).

* Ensured equal distribution across categories.
* Used **oversampling (replace=True)** to reach **5000 rows**.
  + **Alcohol-induced**: 1202 samples
  + **Environment-induced**: 1222 samples
  + **Both**: 1312 samples
  + **None**: 1264 samples

import pandas as pd

import numpy as np

thresholds = {

    "co(mq7)": 400, "smoke(mq135)": 600, "temp\_low": 10, "temp\_high": 40,

    "humidity\_low": 10, "humidity\_high": 85, "mq3(alcohol)": 450

}

def classify\_drowsiness\_cause(row):

    alcohol\_drowsy = row['mq3'] > thresholds['mq3']

    environment\_drowsy = (

        row['co'] > thresholds['co'] or row['smoke'] > thresholds['smoke'] or

        row['temp'] < thresholds['temp\_low'] or row['temp'] > thresholds['temp\_high'] or

        row['humidity'] < thresholds['humidity\_low'] or row['humidity'] > thresholds['humidity\_high'] or

        row['mq13'] > thresholds['mq13']

    )

    if alcohol\_drowsy and environment\_drowsy:

        return "both"

    elif alcohol\_drowsy:

        return "alcohol"

    elif environment\_drowsy:

        return "environment"

    else:

        return "none"

num\_samples = 1\_000\_000

data = {

    'co': np.random.randint(0, 50, num\_samples),  # CO levels

    'smoke': np.random.randint(0, 100, num\_samples),  # Smoke levels

    'temp': np.random.uniform(10, 40, num\_samples),  # Temperature in degrees Celsius

    'humidity': np.random.uniform(10, 90, num\_samples),  # Humidity percentage

    'mq3': np.random.randint(300, 700, num\_samples),  # MQ3 sensor values

    'mq13': np.random.randint(300, 700, num\_samples)  # MQ13 sensor values

}

df = pd.DataFrame(data)

df = df.drop\_duplicates().reset\_index(drop=True)

df['drowsiness\_cause'] = df.apply(classify\_drowsiness\_cause, axis=1)

df.to\_csv("drowsiness\_1M.csv", index=False)

print("Dataset created successfully with", len(df), "unique rows.")

car number, date timestamp of drowsiness, cause of drowsy,accident occurance (yes/no) , insurance claim(yes/no)

thresholds = { "co(mq7)": 400, "smoke(mq135)": 600, "temp\_low": 10, "temp\_high": 40, "humidity\_low": 10, "humidity\_high": 85, "mq3(alcohol)": 450 }