



<https://creativecommons.org/licenses/by/4.0/>

# SOFTWARE DE PREDICCIÓN DEL NIVEL DE MONÓXIDO DE CARBONO (CO) DEL AIRE EN LA UNIVERSIDAD LIBRE SEDE BOSQUE

## *Software for predicting the level of carbon monoxide (CO) in the air at the Libre University Campus Bosque*

JAIME ALFONSO LEMUS CASAS<sup>1</sup> Y RICARDO ANDRES SANTA QUINTERO<sup>2</sup>

Recibido:25 de noviembre de 2022. Aceptado:16 de enero de 2023

DOI: <http://dx.doi.org/10.21017/rimci.2023.v10.n19.a128>

### RESUMEN

Este artículo presenta el panorama completo del proyecto que permite medir la calidad del aire por medio de estaciones de medición que tiene dispuestas el gobierno local y también por medio de dispositivos CANSAT que son picos satélites de construcción propia que por su bajo costo pueden colocarse directamente en los lugares donde se quiere medir el impacto de la contaminación. Estos dos mecanismos de obtención de datos medioambientales brindan una gran cantidad de datos que deben ser almacenados, procesados y validados antes de ser utilizados por los algoritmos que generan las predicciones de niveles de monóxido de carbono. Se describen las más fuertes problemáticas de los altos niveles de monóxido de carbono en los organismos vivos. La presente aplicación brinda no solo las mediciones, también las predicciones para cada área monitoreada en los alrededores de la Universidad Libre sede El Bosque.

**Palabras clave:** Predicción; Aire; Monóxido de Carbono; Cansat; Big Data.

### ABSTRACT

This article presents the complete panorama of the project that allows to measure the air quality by means of measurement stations that the local government has arranged and by means of CANSAT devices that are self-constructed picosatellite that, due to their low cost, can be placed directly in the places where you want to measure the impact of pollution. These two mechanisms for obtaining environmental data provide a large amount of data that must be stored, processed, and validated before being used by the algorithms that generate predictions of carbon monoxide levels. The strongest problems of elevated levels of carbon monoxide in living organisms are described and it provides not only the measurements, but also the predictions for each monitored area in the surroundings of the Universidad Libre, El Bosque headquarters.

**Keywords:** Prediction; Air; Carbon Monoxide; Cansat; Big Data.

## I. INTRODUCCIÓN

LA CONTAMINACIÓN del aire es una de las problemáticas más considerables a las cuales el mundo hace frente hoy en día, se está viviendo una época de consumismo y sobrepoblación, la

cual obliga a la explotación de los recursos naturales de forma acelerada y sin control en algunas ocasiones.

En este contexto la ONU en 2015, anunció una agenda global la que denominó 17 objetivos para

1 Estudiante de Ingeniería de sistemas, Universidad Libre. ORCID: <https://orcid.org/0000-0001-9280-6834> Correo electrónico: [jaima-lemusc@unilibre.edu.co](mailto:jaima-lemusc@unilibre.edu.co)

2 Magister en tecnologías digitales aplicadas a la educación. Especialista en Gerencia de proyectos de Sistemas. Ingeniero de sistemas. Docente de ingeniería de sistemas, Universidad Libre. ORCID: <https://orcid.org/0000-0002-8399-2425> Correo electrónico: [ricardo.santaq@unilibre.edu.co](mailto:ricardo.santaq@unilibre.edu.co)

transformar al mundo más conocidos como objetivos de desarrollo sostenible en estas se destacan el objetivo 3.8 y 11.6. En estos se habla sobre la reducción de muertes a causa de productos químicos peligrosos y contaminación del aire y la reducción del impacto ambiental en las ciudades, enfatizando en la calidad del aire[1].

Así mismo Giddens y Hutton, elaboraron un libro sobre la vida en el capitalismo global, en dicho libro se intenta concientizar por medio de una escena ocurrida en 1992 donde ciudadanos y gobiernos se reunieron para discutir temas ambientales en la ciudad de Río-Brasil, mejor distinguido como la cumbre de la tierra, aunque se hablaron de temas muy puntuales, los autores señalan el término: "Producción de ignorancia", ya que al comenzar un nuevo milenio la falta de control de organismos internacionales sobre la explotación de primer mundo hacia las zonas y comunidades más vulnerables de tercer mundo contribuyen a que 2/3 de estas poblaciones termina empobreciéndose a falta de los recursos naturales, y se plantea la pregunta sobre: ¿hasta qué punto una industria o país se puede colocar sobre los recursos naturales los cuales muchos de ellos son manejados por conocimientos heredados de generación en generación?[2].

Por lo cual se plantea un sistema de predicción de monóxido de carbono, utilizando el lenguaje de programación Python, este lenguaje es comúnmente utilizado para Big Data, machine learning, analítica de datos entre otros[3]. Se emplea especialmente por ser un lenguaje sencillo, versátil y flexible, se puede usar en la mayoría de los sistemas operativos, además, se pueden emplear soluciones rápidamente y con menos líneas de código con respecto a otros lenguajes de programación[4].

## II. PLANTEAMIENTO DEL PROBLEMA

En este contexto Ferran Ballester, afirma que en ese momento la ONU atribuye el 1.4% de fallecimientos en el mundo son debido a la contaminación del aire, especialmente de material particulado. Mayormente por el desarrollo industrial del momento, sin embargo, grandes comunidades estaban expuestas a este tipo de contaminación al cual no estaban adecuados, estos eran más propensos a tener afecciones negativas en su salud. Se habla de

fuentes de emisión primarias y secundarias. El primer grupo corresponde en su mayoría a los que provienen directamente de la fuente de emisión, como el  $\text{SO}_2$  o dióxido de azufre, el cual procede de procesos industriales y vehículos. La segunda fuente de la cual habla el texto es de la cual deriva de procesos químicos y físicos como el gas, aunque también puede provenir de vehículos, también procede del humo del tabaco. Como conclusión se plantean algunos retos como establecer monitoreo de calidad del aire (el cual ya está implementado) y desarrollar modelos de predicción sobre efectos en la salud (se han implementado sistemas de predicción de calidad del aire, pero no hacia la salud)[5].

En comparación el párrafo anterior, la ONU en 2021, por medio del secretario General de la organización, en el primer día internacional del aire limpio, se conoció la cifra de muertes por contaminación atmosférica; las cuales son aproximadamente 7 millones de personas fallecidas antes de la edad promedio de la población en la que habita, se le conoce mejor como muerte prematura, sumado a los bajos ingresos de las personas en ciertas poblaciones lo cual contribuye a la pobreza mundial. De la misma forma, el secretario de la ONU afirma que, 9 de 10 personas o aproximadamente el 90% del mundo respira aire lleno de contaminación o polución, estas tienen un efecto directamente proporcional a la salud de la persona, lo cual resulta en enfermedades cardíacas, accidentes cerebrovasculares, cáncer de pulmón entre otras. También afirmó que, es necesario reducir el nivel de calentamiento global por lo menos a 1,5 grados, esto permitirá bajar drásticamente los niveles de polución en el aire[6].

Sin embargo, hay quienes creen que el desperfecto de la calidad del aire junto con la capa de ozono comienza desde la primera revolución industrial. Por otro lado, Elena Boldo, dice que la atmósfera es dinámica y ha tenido varios cambios en el tiempo, afirmando que desde el comienzo de las civilizaciones se ha comenzado este deterioro, en los últimos dos siglos ha cambiado de forma rápida y éste altera gran parte ciclos naturales que se habían conocido gracias a civilizaciones pasadas. Desde 1661 John Evelyn escribió al Rey de Inglaterra: Carlos II, preocupado por el cambio climático y las afecciones en la salud que el pueblo recibía. Se puede afirmar que,

el problema de la contaminación es muy antiguo, pero gracias a los pocos controles que existían sobre la calidad del aire en la primera revolución industrial se logró un avance tecnológico importante, pero un retroceso en la salud y calidad de vida de las personas; ejemplo de esto, en 1952 una niebla intensa causada por la contaminación, reposo en Londres durante 4 días, a este suceso se le atribuyen de 4000 a 12000 muertes a ese incidente solo ese año[7].

Visto que la contaminación es un problema mundial, según Alejandrina, G. P., Anselmo, G. F. I., Benjamín, S. G., Pilar, F. H., María, G. M. R., Carlos, B. Y. J., Ángel, P. P. J., Reyes, N. P., & Senén, D. A. J, afirman que, el aire es un derecho fundamental para cada ser humano, por lo menos necesitamos entre 10 y 20 metros cúbicos ( $m^3$ ) pero gracias a la expansión demográfica y desarrollo industrial crecientes alrededor del mundo han limitado en gran medida este derecho, especialmente en las ciudades, según un estudio citado de la organización mundial de la salud (OMS), entre las fechas comprendidas de 1997-2006 se registró un aumento del 32% del material particulado mayor a 10 micrómetros (PM10), el dióxido de nitrógeno ( $NO_2$ ) se elevó un 42% y el oxígeno apenas aumentó un 18%. Así mismo, afirma que la contaminación se acentúa principalmente en zonas urbanas y suburbanas. Como conclusión, la sociedad se ve afectada por esta problemática, pero los niños, ancianos, personas con enfermedades en los pulmones o corazón y mujeres embarazadas, se ven más vulnerables ante esta[8].

Especialmente en niños la contaminación atmosférica genera algunos problemas de salud graves. En ese contexto Ubila y Yohannessen, explicaron las secuelas de la contaminación en el ambiente para este colectivo. Gracias a los agentes tóxicos presentes en el aire, producto de la contaminación, los pulmones tanto de niños como de los fetos se ven gravemente afectados, a tal punto de generar en muchas ocasiones lesiones pulmonares que son irreparables si no se detectan dentro de cierto tiempo, estas lesiones se pueden dividir en 5 grupos, el primero es la interferencia de actividades normales del niño, el segundo son enfermedades respiratorias regulares, enfermedades discapacitantes, daño respiratorio permanente y alteración respiratoria progresiva. De la misma forma afecta el desarrollo del metabolismo para todas las edades.

Por último, se afirma que hay suficiente evidencia para relacionar la mala calidad del aire con la afectación de la salud respiratoria[9].

En Bogotá Colombia Gaitán, Cancino y Behrentz, analizaron, los diferentes conjuntos de calidad del aire recolectado de Red de Monitoreo de Calidad del Aire de Bogotá (RMCAB) entre 1998 a 2005, en el 2001 el PM10 aumentó un 40% en los primeros días del año. Lo que este estudio revela es que los Bogotanos en esos meses específicos estaban expuestos a niveles no recomendados por la Organización mundial de la salud (OMS) y la Agencia de Protección Ambiental de los Estados Unidos. La concentración de material particulado se encuentra mayormente en la zona industrial, donde persiste alrededor de un 90% de los días del año. De la misma forma se denota el deterioro de la calidad del aire en Bogotá cerca de un 45%, esto generado mayormente por el crecimiento económico, asentamiento de grandes industrias en la ciudad y poco manejo de políticas ambientales para ese momento por parte del gobierno Nacional[10].

### III. OBJETIVOS

#### A. Objetivo General

Diseñar un software de predicción del nivel de monóxido de carbono (CO) del aire en la universidad Libre.

#### B. Objetivos Específicos

- Definir los requerimientos funcionales y no funcionales para el diseño del software y hardware de predicción de monóxido de carbono.
- Diseñar el software para la predicción del nivel de monóxido de carbono.
- Construir el software para la predicción del nivel de monóxido de carbono.
- Implementar el software de predicción del nivel de monóxido de carbono.
- Probar el software de predicción del nivel de monóxido de carbono.

## IV. JUSTIFICACIÓN

El monóxido de carbono, como se ha venido hablando, ocupa un lugar en la sociedad gracias a diferentes factores como procesos industriales, vehículos, mal tratamiento de los desechos etc. A la exposición a monóxido de carbono provoca primero, síntomas como dolor de cabeza, falta de fuerza física, falta de aire, vómito, náuseas, desmayos y mareos. Estos síntomas son causados por que el monóxido entra directamente por vías respiratorias hasta llegar a los pulmones; allí pasan al torrente sanguíneo, esto ocasiona que las células no puedan utilizar el oxígeno que les llega, el monóxido de carbono afecta principalmente al cerebro y al corazón. Una de las preocupaciones más comunes sobre este es que es un asesino invisible, se le llama de esta forma porque en sus características no tiene: color - sabor - olor - no irrita los ojos, por lo cual es necesario detectarla antes de que tenga un impacto mayor sobre la salud de la población en general[11].

Para solucionar este problema se ha planteado utilizar dos metodologías: La primera sería utilizar un pico satélite más conocido como CanSat, la cual es una iniciativa comenzada por algunas agencias aeroespaciales de Estados Unidos y Europa para estudiantes sobre la construcción de un aparato del tamaño de una lata de refresco o gaseosa con el fin de recopilar datos sobre la información que sus sensores capturen[12]. La segunda metodología será por medio del uso del lenguaje de programación Python, este lenguaje es comúnmente utilizado para Big Data, machine learning, analítica de datos entre otros[13]. Se emplea especialmente por ser un lenguaje sencillo, versátil y flexible, se puede usar en la mayoría de los sistemas operativos, además, se pueden emplear soluciones rápidamente y con menos líneas de código con respecto a otros lenguajes de programación[14].

Para que el CanSat logre predecir el monóxido de carbono, se necesitará comprar y programar diferentes sensores para la medición de calidad del aire como el MQ135, DHT22 entre otros. Al mismo tiempo se tendrá que programar dentro del ambiente de programación Arduino. Adicionalmente se implementará también la tarjeta Wemos Lolin 32, la cual cuenta con un servidor web. Allí se publicarán los resultados de los datos tomados. Pero la tarjeta tiene un corto almacenamiento. Para

garantizar que los datos se vean de forma correcta, se utilizará la base de datos FireBase de Google, la cual es una base de datos NoSql, donde los datos serán enviados y allí se podrá consultar, modificar y descargar los diferentes datos recolectados.

La diferencia con respecto a otros proyectos es que los datos no se publican directamente en la página, se usará el lenguaje de programación Python para predecir los posibles valores futuros del monóxido de carbono. Los sensores del CanSat recolectan información de la nube, esta información es recogida y procesada dentro del programa que se creará en Python. Allí se le podrá aplicar un modelo de predicción de datos, este arroja un valor aproximado sobre los datos de la predicción del monóxido de carbono, al terminar, la información se organizará y se podrán enviar a la página web creada en la tarjeta, garantizando así la integridad de los datos.

## V. MARCO TEÓRICO

### A. Analítica de datos y sus técnicas

Según Moncayo y SAS Industry Solutions, la analítica de datos ha tomado un lugar muy importante en el desarrollo de las empresas a nivel mundial, para el 2022 solo en Colombia se han empleado más de 40 mil millones de pesos para la implementación de Big Data y tecnologías la cual faciliten el análisis de esta; aproximadamente más del 40% de las empresas han comenzado a implementar la analítica en el camino de la toma de decisiones. Dentro de las industrias que más destacan por la analítica de datos en Colombia según el SAS, son entidades bancarias, entre esas Bancolombia y BBVA. Estas entidades se han dado cuenta que, gracias al manejo de los datos, pueden identificar patrones de comportamiento similares entre distintos usuarios, y entre más datos recopilados, mayor será la detección de estos patrones[15][16].

Así mismo, gracias a esos patrones de comportamiento, se pueden diseñar planes diferentes para la consolidación del cliente y nuevas ocasiones de negocio sobre la entidad, así mismo, análisis temprano de diferentes riesgos para contrarrestar el fraude, etc. De la misma forma, esta se usa mucho en las empresas para asistir en la toma de decisiones importantes con modelos probabilísticos[16].

Además, no solo se estudiarán los datos y comportamientos de los clientes, algunas empresas analizan también el comportamiento interno de la empresa[17]. En áreas como sanidad, educación, marketing, producción, logística, se toman los datos y por medio de la implementación de modelos analíticos, se logra la optimización de los procesos en esas áreas, si se logra hacer un ajuste, este aumentara su productividad y de forma consecuente genera competencia con otros negocios del mismo tipo[18].

Existen diferentes técnicas las cuales usan las empresas para tomar decisiones importantes como, ¿Dónde colocar una tienda?, ¿Cuál es el camino más corto para ahorrar en gasolina?, ¿En qué bolsa invertir? y ¿cuánto invertir? etc. Ahí es donde entra el aprendizaje automático o Inteligencia Artificial o machine learning, este se ocupa del desarrollo de algoritmos para el análisis de comportamientos a los cuales esté dirigido, usa por lo general datos empíricos para sus predicciones y se divide en dos grupos: supervisados y no supervisados. Aunque ambos son diferentes, en ambos casos el objetivo del algoritmo es ordenar los datos y/o describir diferentes estructuras encontradas en la data[19].

Un ejemplo de estructuras es el análisis espacial. Este tipo de análisis existe en las empresas para analizar una data sobre la ubicación geográfica con respecto al comportamiento de los consumidores. en esa área específica Esto con el fin de detectar y cuantificar estos patrones, este aportará nueva información para tomar decisiones en las empresas[20]. Un valor agregado es que esta técnica puede analizar varias capas, como la trascendencia de un lugar y la marca sobre el tiempo, para dar información visualmente más sencilla de entender. Un ejemplo de este análisis espacial es el consumo de un producto, junto con la distancia a los centros comerciales que el usuario tendría que recorrer. Como se observa en la Fig. 1, los puntos en donde más se concentra el color, son donde sería mejor poner un lugar para que la mayoría de los usuarios hagan un recorrido más corto.

De la misma forma se usa el análisis de redes por medio de una data obtenida, el algoritmo se entrena para buscar conexiones o nodos de manera discreta, de esta forma se crea un grafo o el camino más corto también se le llama. Este algoritmo es usado especialmente en las empresas de

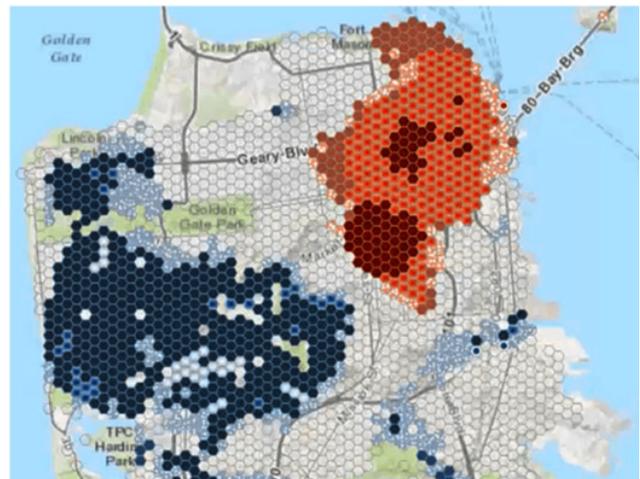


Fig. 1. Análisis espacial sobre consumo y ubicación. ArcGIS[20].

transporte para relacionar y predecir el tiempo de desplazamiento de un medio de transporte desde un punto inicio hasta un punto final[19]. Las empresas utilizan estos y más patrones para la toma de decisiones las cuales pueden subir y bajar. Pero se usan los datos para tener certeza de que hacia donde se está apuntando es el mejor camino para seguir y no generará considerables pérdidas de dinero a las empresas.

Según González y Alonso, las empresas especialmente de manufactura, pequeñas y medianas comienzan a implementar estos modelos de predicción sobre su sistema financiero, buscando evitar bancarrota dentro de organización, la cual no solo tiene repercusiones internas, también a nivel social por el desempleo el cual puede llegar a generar. Para eso se delimitó el proyecto en 5 áreas las cuales eran necesario predecir sobre el total de activos de la empresa, esas áreas son el: activo circulante y pasivo líquido, reservas, EBIT (Earnings Before Interest and Taxes) o patrimonio neto, capitalización de mercado y entradas de explotación. Usaron datos de 79 empresas las cuales publican sus datos en SABI (Sistema de Análisis de Balances Ibéricos) con 158 registros cada una para hacer el modelo predictivo. Donde plantearon las mismas áreas como  $X_n$  donde  $n$  es el número del área delimitado anteriormente. Como resultado, primero se hizo un análisis sobre la media y moda para ver cómo se comportan los datos, desde allí se logró plantear el modelo y predecir qué áreas podrían fallar más en un futuro, dando como resultado el patrimonio neto es más propenso a fallar junto con el activo circulante, al mismo tiempo, el modelo determina qué áreas que tienen poca relación[21].

## VI. MARCO METODOLÓGICO

### A. Tipo de investigación

El presente proyecto se enmarca dentro de una investigación aplicada donde se busca, la generación de nuevo conocimiento dentro del campo de calidad del aire, así mismo sobre la predicción de monóxido de carbono, ya que existen pocas investigaciones sobre este tema, teniendo en cuenta, la toma de datos dentro de la Universidad Libre sede el bosque, se llevará a cabo con la tecnología CanSat o pico satélite. Estos datos serán subidos a una base de datos NoSql, allí se implementará un algoritmo de predicción con el lenguaje de programación de Python, para predecir variables ambientales. Bajo esta metodología, el desarrollo del proyecto se estructurará bajo unas fases, las cuales se detallan a continuación:

**Fase 1.** Plantear un documento de software: la fase inicial de cualquier proyecto siempre será hasta donde quiere llegar el cliente y hasta donde se puede llegar a llegar con las herramientas que se tengan a la mano. Por lo cual se definirán los requerimientos, lenguajes de programación y planeación del proyecto, y se buscará tener un equilibrio entre alcance, hardware, software, calidad y presupuesto.

**Fase 2.** Construcción del software: una vez obtenidos los requerimientos, se comenzará a buscar un lenguaje de programación que satisfaga lo planteado en la fase anterior, para su posterior programación. Así mismo se construirá una base de datos NoSql dentro de FireBase.

**Fase 3.** Construcción del hardware: se procederá a realizar planos y comparar los componentes para crear la estación con la cual se medirán las variables ambientales, así mismo se programarán los componentes dentro de un IDE que cubra todos los componentes.

**Fase 4.** Pruebas del software/hardware: para completar esta fase, primero se necesitará integrar el hardware y software previamente construido, después de eso se buscará probar la eficiencia, eficacia e integridad de lo que se construyó, junto con las adaptaciones necesarias al código.

**Fase 5.** Despliegue del producto: cuando se tenga el código, se comenzarán los primeros despliegues dentro de la universidad libre junto con el monitoreo de los datos.

## VII. MATERIALES

Este proyecto, aunque ya se han tomado algunos datos, se utilizó un set de datos Excel, en donde se tienen los datos necesarios para desarrollar la lógica del programa, estos datos se pueden descargar en la página RMCAB o red de monitoreo de calidad del aire de Bogotá[22], se tomaron datos desde el primero del 2021 hasta el primero de agosto del 2022 como se muestra en la figura, donde se obtienen aproximadamente 8765 registros, se muestra el Excel en la Fig. 2. Dentro de este Excel se encuentran columnas como Material particulado (PM10 y PM2.5), dióxido de carbono

1 Tipo: Promedio, Base de tiempo: 1 Hora, Anual: 01-08-2021-01-08-2022																
2																
3 Centro de Alto Rendimiento																
4	PM10	CO	OZONO	NO	NO2	NOX	SO2	Vel Vient	Dir Vient	Temperat	HR	Precipitac	Rad Solar	PM2.5	CO2	
5	DateTime	µg/m3	ppm	ppb	ppb	ppb	ppb	m/s	Grados	°C	%	mm	W/M²	µg/m3	ppm	
6	01-08-2021:....		0,31818	2,163	2,229	12,272	14,502	0,641	0,6	249	13,2	70,93	0	0	6	495
7	01-08-2021:....		0,26455	3,18	3,191	8,788	11,979	0,771	0,5	313	12,6	73,43	0	0	4	493
8	01-08-2021:....		0,25779	5,109	3,979	7,99	11,969	0,851	0,6	139	12,5	72,78	0	0	4	496
9	01-08-2021:....	3	0,11343	12,035	0,52	2,791	3,311	0,409	2	107	13,5	64,71	0	0	3	481
10	01-08-2021:....	1,5	0,1064	11,421	0,365	3,557	3,923	0,314	1,1	169	12,4	68,11	0	0	1	486
11	01-08-2021:....	3	0,09962	13,752	0,264	1,815	2,078	0,209	1,4	164	12,9	66,39	0	1	3	484
12	01-08-2021:....	1,8	0,1193	13,598	0,651	2,4	3,051	0,264	2	122	13,7	63,85	0	68	0	478
13	01-08-2021:....	2,1	0,11418	15,952	0,786	1,707	2,493	0,36	1,6	171	16	55,09	0	191	0	472
14	01-08-2021:....	3	0,15734	16,576	1,138	2,32	3,458	0,234	1,7	190	15,2	57,1	0	151	3	467
15	01-08-2021:....	2,2	0,17349	16,847	1,542	2,402	3,945	0,191	2,1	198	16,5	53,99	0	246	1	465
16	01-08-2021:....	3	0,11292	17,65	1,118	1,53	2,648	0,274	2,4	165	16,7	54,46	0	370	3	462
17	01-08-2021:....	2,4	0,15292	17,693	1,648	2,004	3,653	0,339	2,5	188	17,4	51,88	0	356	2	459
18	01-08-2021:....	2,7	0,12612	17,237	1,376	1,711	3,087	0,451	2,3	164	17,5	52,45	0	410	0	457
19	01-08-2021:....	3	0,14311	16,694	1,555	1,904	3,459	0,482	2,2	169	18	50,53	0	359	3	457
20	01-08-2021:....	4	0,1506	16,635	1,686	1,962	3,648	0,232	2,3	188	18,5	49,09	0	431	4	456
21	01-08-2021:....	2,7	0,14742	15,896	1,545	2,152	3,696	0,439	2,2	189	17,2	52,34	0	218	1	458
22	01-08-2021:....	2,7	0,12299	15,974	1,057	2,173	3,23	0,462	2,2	154	15,6	57,19	0	107	0	461
23	01-08-2021:....	2,6	0,13632	14,846	1,047	3,025	4,072	0,028	2	158	15	58,9	0	34	2,6	465
24	01-08-2021:....	2,5	0,13321	14,202	0,553	3,428	3,98	0,323	1,1	170	14	62,35	0	1	0	474
25	01-08-2021:....	2,5	0,13584	13,201	0,636	3,868	4,504	0,285	1,2	186	13,8	64,31	0	0	0	476
26	01-08-2021:....	2,5	0,17173	10,501	0,802	5,721	6,523	0,459	1,1	185	13,1	68,6	0	0	0	480

Fig. 2. Reporte RMCAB[22]

(CO<sub>2</sub>), Óxido nítrico (NO), el estado de oxidación del óxido nítrico (NO<sub>x</sub>), dióxido de azufre (SO<sub>2</sub>), velocidad del viento junto con su dirección en grados, temperatura, humedad relativa etc.

Al mismo tiempo, en la justificación se mencionaron algunos sensores, pero se quiere dar un contexto mucho más amplio sobre algunos requerimientos para el hardware, para este proyecto parcialmente se han utilizado 6 sensores, entre los cuales están:

- Módulo Barómetro Bmp180: este sensor permite medir la altura dependiendo el nivel del mar, por lo cual ofrece datos sobre la longitud de manera exacta[22].
- Módulo Mpu6050 Para Arduino: este sensor brinda datos sobre los 3 ejes (X,Y,Z) utilizando de manera concurrente para navegación de dispositivos etc.[23].
- Modulo Gps Gy-Neo6mv2 con Memoria Eeprom: arroja datos sobre la ubicación[24].
- Modulo Sensor de Calidad del Aire Mq135: este sensor permite el reconocimiento o detección de gases tóxicos los cuales pueden llegar a ser nocivos para la salud en un rango de 10 a 1000 partes por millón[25].
- Módulo Sensor Dht22 Am2302 Temperatura y Humedad: primero toma datos sobre el aire del ambiente, al mismo tiempo, tiene un sensor interno el cual revisa la humedad del ambiente[26].

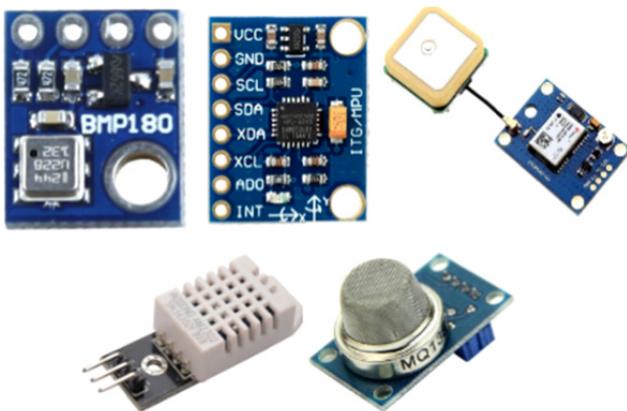


Fig. 3. Sensores.

- Board Nodemcu Wifi Wemos Lolin32 basado en Esp32: es la tarjeta madre del CanSat, cuenta con conectividad Wifi, Bluetooth, además se puede habilitar un servidor web dentro de la misma tarjeta[27].



Fig. 4. Módulo Wemos Lolin32.

Para integrar tanto el Hardware o los componentes con el programa o el software se utilizará el lenguaje de programación Arduino. Este lenguaje es usado porque posee las librerías necesarias para trabajar estos componentes. De la misma forma se espera añadir, eventualmente, nuevos componentes para las siguientes versiones. Entre esas está el sensor MQ-7, el cual se encarga de medir el nivel de monóxido de carbono en el aire.

## VIII. AVANCES

Primero que todo hay que aclarar que, gran parte del software tanto de preprocesamiento como de predicción esta codificado desde un set de datos el cual fue previamente expuesto, este corresponde a una primera versión del código, la idea para las siguientes versiones es que se pueda tomar de una base de datos NoSql como Firebase. Adicionalmente, todo el diseño del software se divide en cuatro carpetas principales, las cuales se subdividen en 4 programas y 2 sets de datos como se observa en la Fig. 5. Si se desea acceder al repositorio, puede acceder por medio del siguiente repositorio Link[28] aclarando que es elaboración de los autores.

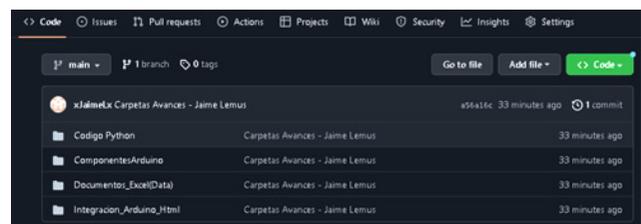


Fig. 5. Rama de carpetas del repositorio. Elaboración propia. Repositorio[23].

Dentro de la carpeta Componentes Arduino, se encuentran la versión la cual hace que todos los componentes o hardware que se habló en materiales logre funcionar, se puede observar en la Fig. 6.

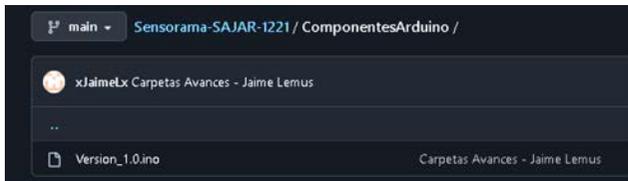


Fig. 6. Carpeta Componentes Arduino. Elaboración propia. Repositorio[28].

Dentro de este código se pueden observar diversas funciones como: definiciones de variables globales, estas corresponden a los pines de salida de la tarjeta Wemos hacia el circuito, se muestra en la Fig. 7 las definiciones de variables globales, en la Fig. 8 se mostraran algunos de los métodos los cuales se utilizaron dentro de la configuración del void o llamado, esta se hace para buscar los componentes y los inicializa, para esa se mostrara el llamado u la inicialización del Bmp180 y Módulo Mpu6050 o acelerómetro. Al mismo tiempo, se configura dentro de la función loop o ciclo repetitivo, lo que se imprimirá posteriormente por consola, en la Fig. 8 y 9 se observa la configuración del Bmp180, y acelerómetro.

Así mismo, en la Fig. 5 se observa una carpeta llamada integración Arduino con HTML, como se mencionó en materiales, el Board Nodemcu Wifi Wemos Lolin32 tiene un servidor web incorporado, lo que se pretende en esta parte del código es crear una página web dentro del servidor con lenguaje Arduino y HTML, como se muestra en la Fig. 10,

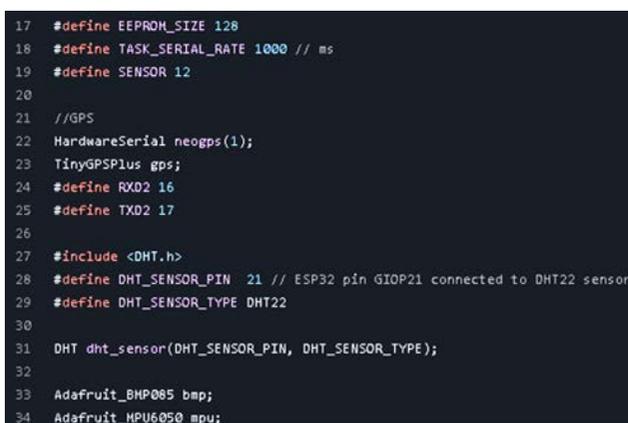


Fig. 7. Variables globales. Elaboración propia. Repositorio[28].



Fig. 8. Configuración del void. Elaboración propia. Repositorio[28].

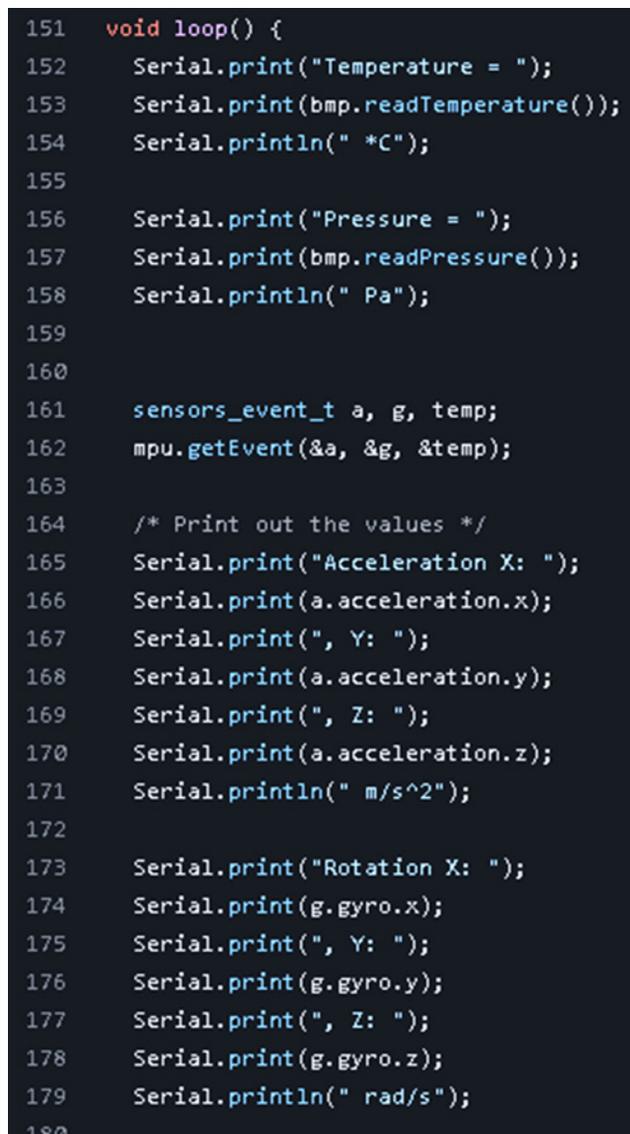


Fig. 9. Configuración del loop e impresión de datos. Elaboración propia. Repositorio[28].

```
44 // Serial
45 uint32_t chipId = 0;
46
47 // WIFI
48 const char* soft_ap_ssid = "SAJR-1221";
49 const char* soft_ap_password = "12345678";
50 const char* host = "SAJR-1221"; // corresponde al nombre del cansat
51
52
53 // Web Server
54 WebServer server(80);
55 WiFiMulti wifiMulti;
56
57 // Insert Firebase project API Key
58 #define API_KEY "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX-X"
59
60 // Insert RTDB URLeFINE the RTDB URL */
61 #define DATABASE_URL "https://ulibreapp-default-rtdb.firebaseio.com"
62
```

Fig. 10. Primera configuración Html y Arduino.  
Elaboración propia. Repositorio[28].

lo primero en configurarse es la tarjeta, por defecto la placa tiene un serial o identificación, para registrar esa información, se crea una variable la cual es la identificación del chip, al mismo tiempo se configura el nombre del CanSat, junto con una contraseña, para el acceso a la base de datos FireBase, así mismo se trae el registro de la llave para acceder a la información y se da el link de la base de datos, en este momento la base de datos es privada, así mismo, se censuró la Api\_Key por razones de seguridad.

De igual forma, en la Fig. 11, se observa un control sobre dos redes Wifi, una local y otra softApip, la local es una ip la cual la tarjeta Wemos habilita dentro del mismo componente para poder acceder y ver la información que se ha recolectado gracias a las diversas funciones, la segunda es similar, la diferencia radica en que si se quiere, se puede crear un punto de acceso dentro de una red específica, allí diferentes dispositivos conectados a la misma red podrán observar la información la cual se contenga.

```
204 void handleIP() {
205   String a = String(Wifi.SSID()) + "\n" + IPAddress2String(Wifi.localIP());
206   String b = String(soft_ap_ssid) + "\n" + IPAddress2String(Wifi.softAPIP());
207   //String adcValue = String(a);
208
209   server.send(200, "text/plain", a+"\n"+b); //Send ADC value only to client ajax request
210 }
```

Fig. 11. Habilitación de puntos de acceso Wifi. Elaboración propia. Repositorio[28].

De este modo, al momento de plantear el código HTML dentro del código Arduino, se planteará un Login primero con usuario y contraseña, lo segundo que se codifica son el obtener direcciones Ip como se mostró en la Fig. 12 y Fig. 13 se mostrara

```
80 const char* loginIndex =
81   "<form name='loginForm'"
82   "<table width='20%' bgcolor='#09F9F' align='center'"
83   "<tr>"
84     "<td colspan=2>"
85     "<center><font size=4>(b)SAJR-1221 v1</b></font></center>"
86     "<br>"
87     "<center><font size=4>(b)WiFi STA - AP</b></font>"
88     "<span id='IP'>0</span></center>"
89     "<br>"
90     "<center><font size=4>(b)GPS</b></font>"
91     "<span id='GPS'>0</span></center>"
92     "<br>"
93     "<center><font size=4>(b)ESP32 Login Page</b></font>"
94   "</td>"
95   "<br>"
96   "<br>"
97   "</tr>"
98   "<tr>"
99     "<td>Username:</td>"
100    "<td><input type='text' size=25 name='userid'><br></td>"
101  "</tr>"
102  "<br>"
103  "<br>"
104  "<tr>"
105    "<td>Password:</td>"
106    "<td><input type='Password' size=25 name='pwd'><br></td>"
107    "<br>"
108    "<br>"
109  "</tr>"
110  "<tr>"
111    "<td><input type='submit' onclick='check(this.form)' value='Login'></td>"
112  "</tr>"
113  "</table>"
114  "</form>"
115  "</script>"
```

Fig. 12. Código HTML dentro de código Arduino. Elaboración propia. Repositorio[28].

```
115 "</script>"
116 "function check(form)"
117 "{"
118 "if(form.userid.value=='admin' && form.pwd.value=='admin')"
```

Fig. 13. Código Script dentro de código Arduino. Elaboración propia. Repositorio[28].

dentro del código HTML y un script, así mismo la obtención de datos del GPS, donde se nos dará latitud, longitud, altura sobre el nivel del mar, adicionalmente se nos dará el DOP sobre los datos, este es el cálculo de la posición con respecto a el lugar donde se esté tomando los datos, entre más alejado este el CanSat con respecto a donde se visualicen los datos habrá mejor precisión. Este es el menú que se tiene parcialmente.

De forma similar, en la Fig. 5 se muestra una carpeta titulada Código Python, dentro de esta carpeta hay dos archivos con la extensión “.py” en la Fig. 14 se mostrará los dos archivos, uno titulado semillero y otro modeloprediccion. El primero hace referencia al preprocesamiento de la data, acá se lee el archivo csv mencionado en los materiales, además se revisa el tamaño de la data, se hace una integración de los datos, eliminamos algunas columnas, se limpian otras columnas y se comienza a realizar un modelo estadístico con respecto a diferentes variables como se muestra en las Fig. del 15 a la 18, todas las datas obtenidas, tanto para el preprocesamiento y la exportación final se encuentran en la carpeta Documentos\_Excel (Data) como se muestran en la Fig. 5.

```

35 #Exportamos la data de xls a csv
36 dfXls = pd.read_excel('StationReport.xlsx')
37 dfXls.to_csv('StationReport.csv', encoding='utf-8', index=False)
38 #Cargamos la data
39 df = pd.read_csv('StationReport.csv')
40
41 #Guardamos el nombre csv para trabajar el Dataframe
42 csv = df
43
44 #Revisamos los registros
45 csv.info()
46
47 #Revisamos el tamaño
48 csv.shape
49
50 #Valores nulos dentro del data frame
51 print(csv.isnull().sum())
52
53 data1 = csv
54 fechas = csv
55
56 ***Para este proyecto separamos totalmente la hora, ya que solo necesitaremos el día por lo cual se eliminan el mes, año y hora.***
57
58 #separamos el día de las fechas
59 data1 = fechas[ 'datetime' ].str .split(pat = '-', n = 1 , expand = True )
60 data1 = data1.drop([ 1 ], axis= 1 )
61 fechas[ 'datetime' ] = data1
62 csv
63

```

Fig. 14. Lectura del Excel, se crea el data frame y separación de la fecha. Elaboración propia. Repositorio[28].

Adicionalmente, dentro de la carpeta de los códigos Python, existe un archivo llamado ModeloPrediccion.py, este es un código el cual, tomando la data completa, correspondiente a 8170 registros, se aplicaron 3 algoritmos correspondientes a regresión lineal, DecisionTreeRegressor y SVR. En las Fig. 19 al 21 se observará la aplicación de los algoritmos de predicción.

```

68
69 # Eliminamos comlumnas
70 csvv=csvv.drop(['PM10 µg/m3'], axis=1)
71 csvv=csvv.drop(['SO2 ppb'], axis=1)
72 csvv=csvv.drop(['CO2 ppm'], axis=1)
73 csvv=csvv.drop(['PM2.5 µg/m3'], axis=1)
74 csvv=csvv.drop(['Precipitacion mm'], axis=1)
75 csvv=csvv.drop(['Rad Solar W/m²'], axis=1)
76
77 ""#Limpieza de datos
78
79 Buscamos datos nulos dentro del DF, esto para eliminar es
80 ""
81
82 #Eliminamos valores nulos
83 csvv = csvv[csvv['CO ppm'].notna()]
84 csvv = csvv[csvv['OZONO ppb'].notna()]
85 csvv = csvv[csvv['NO ppb'].notna()]
86 csvv = csvv[csvv['NO2 ppb'].notna()]
87 csvv = csvv[csvv['NOX ppb'].notna()]
88 csvv = csvv[csvv['Vel viento m/s'].notna()]
89 csvv = csvv[csvv['Dir viento Grados'].notna()]
90 csvv = csvv[csvv['Temperatura °C'].notna()]
91 csvv = csvv[csvv['HR %'].notna()]
92
93 #Opcional para borrar valores nullos
94 csvv = csvv.dropna(how='all')
95 csvv.shape
96
97 #Revisión valores nulos dentro del data frame
98 print(csvv.isnull().sum())

```

Fig. 15. Eliminación de columnas, elaboración valores nulos y revisión de valores nulos. Elaboración propia. Repositorio[28].

```

126 #Varianza
127 print('Varianza= ')
128 print(var(csvv))
129 print("-----")
130 #Desviación Estándar
131 print('Desv Estandar= ')
132 print(std(csvv))
133 #Coeficiente de variación
134 print("-----")
135 print('Coeficiente de variación')
136 print(mean(csvv) / std(csvv))
137 print("-----")
138
139 csvv.CO_ppm.hist()
140
141 csvv.NO_ppb.hist()
142
143 ""Se encuentra que tanto en el CO como en NO, el histograma está
144
145 csvv.NO2_ppb.hist()
146
147 csvv.plot.scatter(x='Datetime', y='CO_ppm')
148
149 csvv.plot.scatter(x='Vel_viento_ms', y='CO_ppm')
150
151 csvv.plot.scatter(x='CO_ppm', y='NOX_ppb')
152
153 csvv.plot.scatter(x='CO_ppm', y='NO_ppb')
154

```

Fig. 16. Gráfica de varianza, desviación estándar y coeficiente de variación, junto con diversos histogramas. Elaboración propia. Repositorio[28].

## IX. RESULTADOS

A continuación, se presentará el avance del hardware, para este se utilizó la herramienta circuit.io[28] la cual nos permite hacer un diagrama del circuito junto con sus componentes, allí se puede editar y mover los componentes. En la Fig. 22 se puede observar el avance del circuito con la herramienta y en la Fig. 23 se nos muestra el circuito en físico. En la Fig. 23 se podrá observar los resultados del programa mencionado en la , sobre la integración de los componentes, estos resultados son únicamente mostrados en consola como se observa en la Fig. 24.

```

172 from scipy import stats
173 # cálculo de correlación y significancia con scipy
174 r, p = stats.pearsonr(csvv['DateTime'], csvv['CO_ppm'])
175 print(f"Correlación Pearson: r={r}, p-value={p}")
176 print("-----")
177 r, p = stats.spearmanr(csvv['DateTime'], csvv['CO_ppm'])
178 print(f"Correlación Spearman: r={r}, p-value={p}")
179 print("-----")
180 r, p = stats.kendalltau(csvv['DateTime'], csvv['CO_ppm'])
181 print(f"Correlación Pearson: r={r}, p-value={p}")
182
183 r, p = stats.pearsonr(csvv['NO_ppb'], csvv['CO_ppm'])
184 print(f"Correlación Pearson: r={r}, p-value={p}")
185 print("-----")
186 r, p = stats.spearmanr(csvv['NO_ppb'], csvv['CO_ppm'])
187 print(f"Correlación Spearman: r={r}, p-value={p}")
188 print("-----")
189 r, p = stats.kendalltau(csvv['NO_ppb'], csvv['CO_ppm'])
190 print(f"Correlación Pearson: r={r}, p-value={p}")
191
    
```

Fig. 17. Análisis de correlaciones con Pearson, Spearman entre otros. Elaboración propia. Repositorio[28].

```

240 #Exportamos la data final
241 Final_Data = csvv
242 Final_Data.to_csv('DataProcesada.csv')
    
```

Fig. 18. Se exporta la data final con el nombre Data Procesada.csv. Elaboración propia. Repositorio[28].

```

28 x = df['CO_ppm'].values.reshape(-1,1)
29 y = df['NOX_ppb'].values.reshape(-1,1)
30 X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=0)
31 regressor = LinearRegression()
32 regressor.fit(X_train, y_train)
33 y_pred = regressor.predict(X_test)
34 RLprecisión = regressor.score(X_train, y_train)
35 RLprecisión = round(RLprecisión, 2)
36 print("La regresión lineal = ", RLprecisión*100)
    
```

Fig. 19. Se aplica el algoritmo de regresión lineal. Elaboración propia. Repositorio[28].

```

38 X = np.array(df[['CO_ppm', 'NO2_ppb']])
39 x = X[:,0]
40 y = X[:,1]
41 x = x.reshape(8170,1)
42 y = y.reshape(8170,1)
43 X_train, X_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 44)
44 model = DecisionTreeRegressor()
45 model.fit(X_train, y_train)
46 predictions = model.predict(X_test)
47 errors = abs(predictions - y_test)
48 aux = 100*(errors / y_test)
49 RLTree = np.mean(aux)
50 RLTree = round(RLTree, 2)
51 print("Precisión del modelo: ", RLTree)
    
```

Fig. 20. Se aplica el algoritmo DecisionTreeRegressor. Elaboración propia. Repositorio[28].

```

53 X = np.array(df[['CO_ppm', 'NO2_ppb']])
54 x = X[:,0]
55 y = X[:,1]
56 x = x.reshape(8170,1)
57 y = y.reshape(8170,1)
58 X_train, X_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 44)
59 regressor = SVR(kernel = "rbf")
60 regressor.fit(X_train,y_train)
61 y_predSVR = regressor.predict(X_test)
62 #calcular la precision del modelo
63 errors = abs(y_predSVR - y_test)
64 aux = 100*(errors / y_test)
65 SVRprecisión = np.mean(aux)
66 SVRprecisión = round(SVRprecisión, 2)
67 print("Precisión del modelo: ", SVRprecisión)
    
```

Fig. 21. Se aplica el algoritmo SVR. Elaboración propia. Repositorio[28].

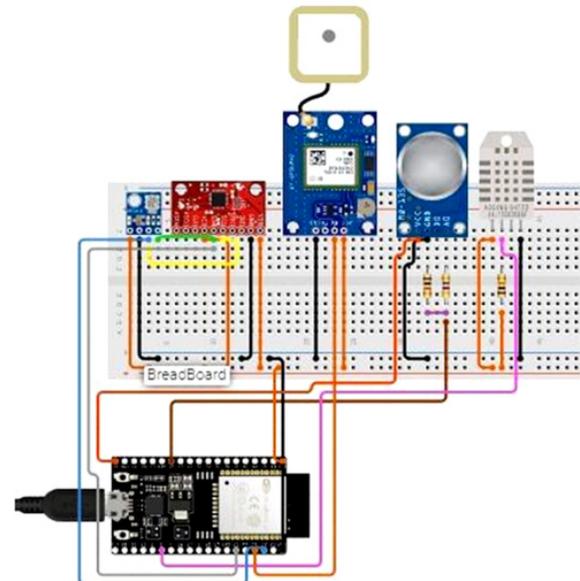


Fig. 22. Diagrama digital. Elaboración propia con la herramienta circuit.io[29].

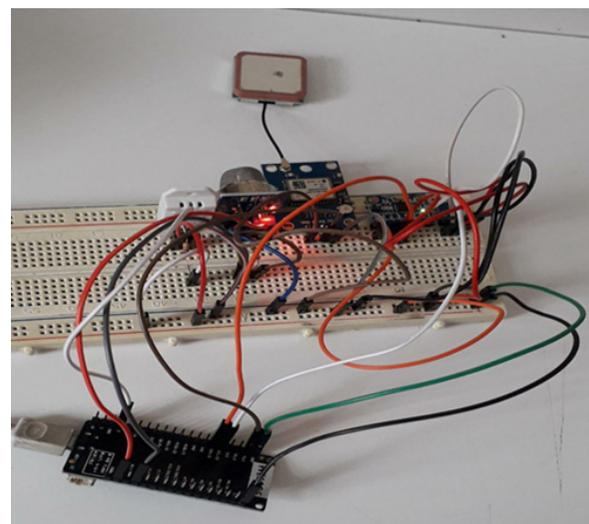
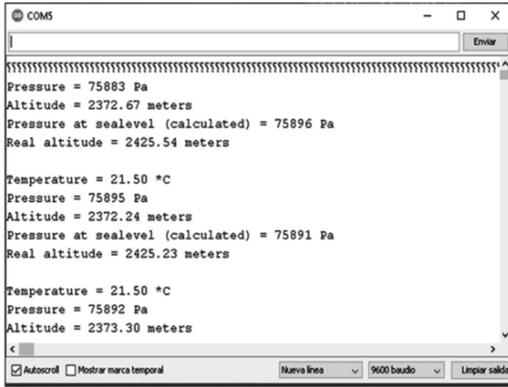


Fig. 23. Circuito físico. Elaboración propia.



```

Lat: 4.686767Lng: -74.103629Speed: 0.46SAT:4ALT:2554
Temperature = 22.70 °C
Pressure = 75594 Pa
Acceleration X: -0.56, Y: 0.33, Z: 9.80 m/s^2
Rotation X: -0.00, Y: 0.01, Z: -0.01 rad/s
Temperature: 22.02 degC
    
```

Fig. 24. Resultado por consola de la carpeta Componentes Arduino. Elaboración propia.

De igual forma, para la integración con HTML y que se muestre el servidor, ya se ha avanzado en la creación de la página, la cual primero ya está publicando datos dentro de la base de datos FireBase, al mismo tiempo, logra mostrar en la página web los datos recolectados del sensor GPS junto con su DOP como se observa en la Fig. 25, este es la demostración de la primera versión de la página funcionando como punto de acceso tanto desde una ip de la Universidad Libre, como una red privada, los datos son traídos desde la Base de datos en firebase, se almacenan de dos formas, en RAW y default. Raw contiene los registros de mediciones on premise, lo que quiere decir que este graba los datos, mientras que default son datos los cuales, la estructura de la base de datos y un registro se puede observar en la Fig. 26 a la 28.

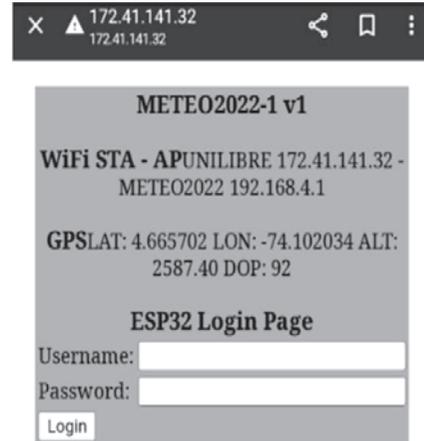


Fig. 25. Primer versión página Web CanSat METEO2022-1. Elaboración propia.

<https://ulibreapp-default-rtdb.firebaseio.com/>



Fig. 26. Base de datos, mediciones de los diferentes CanSat. Elaboración propia.

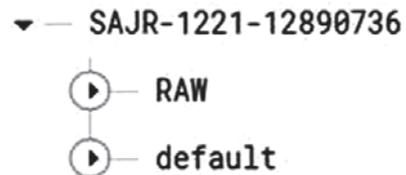


Fig. 27. Base de datos, estructura de los registros en Raw y default. Elaboración propia.

GD <https://ulibreapp-default-rtdb.firebaseio.com/> > SAJR-1221-128... > RAW > 2022-10-20 22:4



Fig. 28. Base de datos, Registro de datos en la BD. Elaboración propia.

Con respecto al preprocesamiento de la data, en primera instancia se mostraron cuantos datos eran nulos dentro del set de datos cargados, como se observa en la Fig. 29, existen una cantidad muy grande de datos las cuales son nulos, así que antes de eliminar los registros nulos, se procedió a eliminar columnas las cuales tengan mayores nulos y después limpiar la data de valores nulos, esto como se muestra en la Fig. 30 nos deja un total de 8170 registros por 10 columnas, con 0 valores nulos.

```

1 #Valores nulos dentro del data frame
2 print(csvv.isnull().sum())

DateTime          0
PM10 µg/m3       411
CO ppm            185
OZONO ppb        198
NO ppb           349
NO2 ppb          349
NOX ppb          349
SO2 ppb          2244
Vel Viento m/s   2
Dir Viento Grados 2
Temperatura °C   1
HR %             12
Precipitacion mm 28
Rad Solar W/M²   99
PM2.5 µg/m3     186
CO2 ppm          3745
dtype: int64
    
```

Fig. 29. Ejecución código, preprocesamiento de la data, revisión de los valores nulos antes de la limpieza de los datos. Elaboración propia.

```

[] 1 #Eliminamos valores nulos
2 csvv = csvv[csvv['CO ppm'].notna()]
3 csvv = csvv[csvv['OZONO ppb'].notna()]
4 csvv = csvv[csvv['NO ppb'].notna()]
5 csvv = csvv[csvv['NO2 ppb'].notna()]
6 csvv = csvv[csvv['NOX ppb'].notna()]
7 csvv = csvv[csvv['Vel Viento m/s'].notna()]
8 csvv = csvv[csvv['Dir Viento Grados'].notna()]
9 csvv = csvv[csvv['Temperatura °C'].notna()]
10 csvv = csvv[csvv['HR %'].notna()]

[] 1 #Opcional para borrar valores nullos
2 csvv = csvv.dropna(how='all')
3 csvv.shape

(8170, 10)

[] 1 #Revisión valores nulos dentro del data frame
2 print(csvv.isnull().sum())

DateTime          0
CO ppm            0
OZONO ppb         0
NO ppb            0
NO2 ppb           0
NOX ppb           0
Vel Viento m/s   0
Dir Viento Grados 0
Temperatura °C   0
HR %              0
dtype: int64
    
```

Fig. 30. Ejecución código, preprocesamiento de la data, eliminación de los valores nulos. Elaboración propia.

Para la revisión estadística de los datos, primero se encuentra la varianza, desviación estándar y coeficiente de variación con respecto a toda las data. La varianza se utiliza para conocer cuanta dispersión hay con respecto a las variables que se miden, la desviación estándar consiste en calcular la dispersión del conjunto de datos, por último, el coeficiente de variación es donde se hace relación a el tamaño de la media y como varia con respecto a las demás variables o datos. Como se observa en la Fig. 31, hay datos los cuales varían bastante en cuestión de dispersión con respecto a otras variables como la dirección del viento o la humedad relativa, mientras que el monóxido "Co\_ppm" no varía mucho con respecto a otras variables. Esto se presenta

```

Varianza=
DateTime          74.882717
CO_ppm            0.178131
OZONO_ppb        149.952416
NO_ppb           474.023669
NO2_ppb          57.589482
NOX_ppb          642.191901
Vel_viento_ms    0.826634
Dir_Viento_Grados 12419.011990
Temperatura_C    16.956417
HR               253.291577
dtype: float64

-----
Desv Estandar=
DateTime          8.653480
CO_ppm            0.422056
OZONO_ppb        12.245506
NO_ppb           21.772085
NO2_ppb          7.588773
NOX_ppb          25.341506
Vel_viento_ms    0.909194
Dir_Viento_Grados 111.440621
Temperatura_C    4.117817
HR               15.915137
dtype: float64

-----
Coeficiente de variación
DateTime          1.774581
CO_ppm            1.367356
OZONO_ppb         0.941402
NO_ppb            0.732941
NO2_ppb          1.974694
NOX_ppb          1.221047
Vel_viento_ms    1.307990
Dir_Viento_Grados 1.937392
Temperatura_C    3.582057
HR               4.325501
dtype: float64

-----
    
```

Fig. 31. Ejecución código, valores estadísticos de la data. Elaboración propia.

en el histograma de la Fig. 32, en donde se observa que la mayor cantidad de datos se encuentran en valores desde 0 a 1, donde se concentran aproximadamente el 75% a 80% de sus datos.

A continuación, usamos la descripción estadística de los datos para buscar correlaciones con la data que se trabaja, la primera fue sobre el tiempo o DateTime como se observa en la Fig. 33.

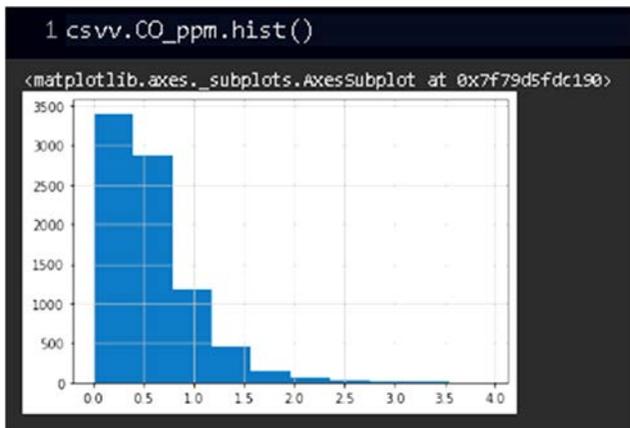


Fig. 32. Ejecución código, valores estadísticos de la data. Elaboración propia.

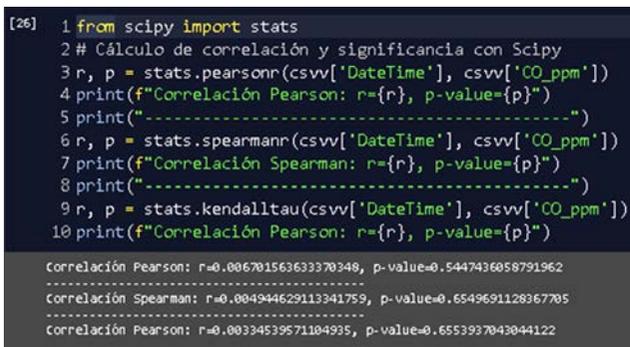


Fig. 33. Ejecución código, correlación DateTime y Monóxido de carbono. Elaboración propia.

Aunque el primer análisis es bueno, cuando se halla el intervalo de confianza CI95%, se encuentra que no se puede confiar en la variable sobre el tiempo, pero se encuentra una excelente relación con respecto a óxido de nitrógeno (No) como se nos muestra en la Fig. 34. A fin de hallar este intervalo de confianza fue necesario instalar la librería pingouin, la cual solo añade algunas funciones para análisis estadístico. Para entender los resultados de las ejecuciones que se muestran a continuación, primero necesitamos entender qué significan las variables las cuales se están midiendo:

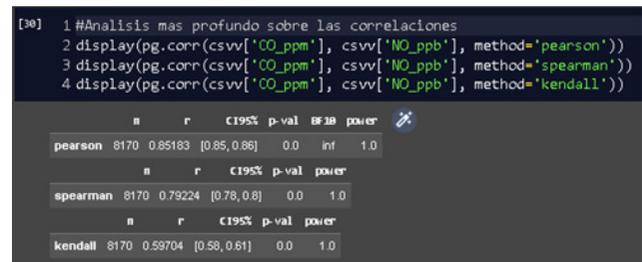


Fig. 34. Ejecución código, correlación Óxido de nitrógeno y Monóxido de carbono. Elaboración propia.

- $n$  = # de muestras dentro del Data Frame.
- $r$  = El coeficiente de correlación sobre la intensidad de la relación lineal entre dos variables.
- CI95% = También llamado intervalo de confianza es donde está concentrado la mayor cantidad de datos (95%), los datos que se encuentran en los extremos son valores atípicos, corresponden al (2,5%) en ambos extremos, dando como resultado un 5%.
- p-val = Es el 5% de los datos atípicos de los cuales se habló en el concepto anterior, esto es para demostrar las hipótesis dentro de las probabilidades.
- BF10 = También llamado análisis o estadística Bayesiana es para medir que tan compatible son las variables (Aunque para este no se nos muestra más información).
- Power = Incrementa junto con el tamaño de la muestra, por lo general se relaciona con el p-val.

Se aprecia que el óxido de nitrógeno es una de las relaciones las cuales tiene mayor cobertura del intervalo de confianza, en promedio está en el 92% en relaciones a todos los datos, de la misma forma el valor de  $r$  está muy bien. Lo que nos quiere decir que hay una buena correlación entre esas variables. Al mismo tiempo, se analizó fue el monóxido con respecto a la velocidad del viento como se muestra en la Fig. 35.

Lo primero que podemos notar de la Fig. 35 es que las  $r$  son negativas, por lo cual la relación entre esas dos variables es poco probable. Al mismo tiempo nuestro intervalo de confianza es negativo, por lo cual se puede afirmar que los datos se encuentran en el 5%, pero no dentro del 95%, esto significa que no es posible relacionar estas variables.

```

2 #Análisis más profundo sobre las correlaciones
3 display(pg.corr(csvw['CO_ppm'], csvw['Vel_Viento_ms'], method='pearson'))
4 display(pg.corr(csvw['CO_ppm'], csvw['Vel_Viento_ms'], method='spearman'))
5 display(pg.corr(csvw['CO_ppm'], csvw['Vel_Viento_ms'], method='kendall'))

```

	n	r	c195%	p-val	BF10	power
pearson	8170	-0.344004	[-0.38, -0.32]	1.051468e-225	2.981e+221	1.0
spearman	8170	-0.388496	[-0.41, -0.37]	1.458818e-292		1.0
kendall	8170	-0.263195	[-0.28, -0.24]	6.621879e-268		1.0

Fig. 35. Ejecución código, correlación de la Velocidad del y Monóxido de carbono. Elaboración propia.

Esto probablemente quiera decir que a mayor velocidad del viento menor concentración de monóxido de carbono.

Por otra parte, para el modelo de predicción del algoritmo, existen dos diferentes resultados, lo único que varía es la forma de tomar los datos, lo explicado en este trabajo es utilizando la data completa, quiere decir los 8170 registros los cuales posee el algoritmo, pero existe una versión alternativa la cual se realizó con 3001 registros, aplicándose los mismos algoritmos de predicción, los cuales son regresión lineal, los resultados corresponden a las Fig. 36 y 37, DecisionTreeRegressor observar la Fig. 38 y 39, SVR los resultados se encuentran en la Fig. 40 y 41.

La regresión lineal = 86.0

Fig. 36. Resultado regresión lineal, registros completos. Elaboración propia.

Regresión lineal 87.9 %

Fig. 37. Resultado regresión lineal, data reducida. Elaboración propia.

Precisión del modelo: 82.27

Fig. 38. Resultado DecisionTreeRegressor, registros completos. Elaboración propia.

Decision Tree Regressor 98.8 %

Fig. 39. Resultado DecisionTreeRegressor, data reducida. Elaboración propia.

Precisión del modelo: 76.28

Fig. 40. Resultado SVR, registros completos. Elaboración propia.

SVR 92.1 %

Fig. 41. Resultado SVR, data reducida. Elaboración propia.

## X. CONCLUSIONES

El presente trabajo mostró la aplicación de algoritmos de clustering, predicción junto a un alto nivel de precisión dentro de estos modelos, dando como resultado un buen grado de eficacia. Por lo cual encontrar el nivel de monóxido sobre ciertas áreas es acertado. Así mismo se encuentra que se puede encontrar variables de las cuales depende el monóxido de Carbono para estar presentes en el aire como el NO<sub>2</sub> mejor conocido como dióxido de nitrógeno, óxido nítrico (NO), etc. Además de ello se concluye que para el algoritmo de predicción el mejor algoritmo con la data completa es la regresión lineal y con la data parcial se ve que es Decision Tree regresor. Al mismo tiempo se mostró un avance en el software para la publicación de resultados con el lenguaje Arduino, así mismo el avance del hardware para la toma de datos.

## REFERENCIAS

- [1] Noticias ONU. «Respirar aire contaminado causa cerca de siete millones de muertes prematuras al año». Noticias ONU. <https://news.un.org/es/story/2021/09/1496382>.
- [2] A. Giddens, En El Límite. La Vida En El Capitalismo Global. Tusquets, 2002.
- [3] F. Tablado. «Python en proyectos Big data ¿por qué elegir lenguaje de programación?» Ayuda Ley Protección Datos. <https://ayudaleyprotecciondatos.es/big-data/python/#:~:text=Una%20de%20las%20principales%20ventajas,que%20otros%20lenguajes%20de%20programación>.
- [4] Tokio. «Librerías de Python para Machine Learning | Tokio». Tokio School. <https://www.tokioschool.com/noticias/librerias-python-machine-learning/#:~:text=Pues%20porque%20se%20trata%20de,de%20Python%20para%20machine%20learning>.
- [5] F. Ballester, Contaminación atmosférica, cambio climático y salud, Rev Esp Salud Pública, vol. 80, n.º 2, p. 17, 2005.
- [6] Naciones Unidas. «Respirar aire contaminado causa cerca de siete millones de muertes prematuras al año». Noticias ONU. <https://news.un.org/es/story/2021/09/1496382>
- [7] E. Boldo. «La contaminación del aire». Repisalud Principal. <https://repisalud.isciii.es/handle/20.500.12105/7274>
- [8] A. Gallego et al. «Contaminación atmosférica». Google Books.

- [9] C. Ubila y K. Yohannessen. «Redirecting». Digital Object Identifier System. <https://doi.org/10.1016/j.rmcl.2016.12.003>
- [10] M. Gaitán, J. Cancino y E. Behrentz, «Análisis del estado de la calidad del aire en Bogotá», *Revista de Ingeniería*, n.º 26, pp. 81-92, noviembre de 2007. [En línea]. Disponible: <https://doi.org/10.16924/revinge.26.10>
- [11] Ministerio Salud Publica Argentina. «Recomendaciones para evitar intoxicaciones por monóxido de carbono». Ministerio de Salud Pública | Misiones. <https://salud.misiones.gob.ar/recomendaciones-para-evitar-intoxicaciones-por-monoxido-de-carbono-2/>
- [12] A. Colin. «Pico satélites cansat: Una herramienta para la educación en ciencias del espacio». Research Gate. [https://www.researchgate.net/publication/317837257\\_Picosatelites\\_cansat\\_Una\\_herramienta\\_para\\_la\\_educacion\\_en\\_ciencias\\_del\\_espacio](https://www.researchgate.net/publication/317837257_Picosatelites_cansat_Una_herramienta_para_la_educacion_en_ciencias_del_espacio)
- [13] BBVA. «Análítica de datos, Inteligencia Artificial y Big Data en la banca | OpenMind». OpenMind BBVA. <https://www.bbvaopenmind.com/economia/finanzas/analitica-de-datos-inteligencia-artificial-y-big-data-en-la-banca/>
- [14] Tokio School. «Librerías de python para machine learning | Tokyo». Tokio School. <https://www.tokioschool.com/noticias/librerias-python-machine-learning/#:~:text=Pues%20porque%20se%20trata%20de,de%20Python%20para%20machine%20learning.>
- [15] C. Moncayo. Las empresas colombianas invierten en IT, la suma llega a más de 40 mil millones de pesos - Instituto Nacional de Contadores Públicos de Colombia». Instituto Nacional de Contadores Públicos de Colombia. <https://incp.org.co/las-empresas-colombianas-invierten-en-it-la-suma-llega-a-mas-de-40-mil-millones-de-pesos/>
- [16] SAS Industry Solutions. «SAS Industry Solutions». Industry Solutions. [https://www.sas.com/es\\_co/industry.html](https://www.sas.com/es_co/industry.html)
- [17] A. Rayon. «Por qué hablamos del business analytics y no solo de business intelligence - deusto data». Deusto Data. <https://blogs.deusto.es/bigdata/por-que-hablamos-del-business-analytics-y-no-solo-de-business-intelligence/> (accedido el 22 de noviembre de 2022).
- [18] N. Cortez, W. Medina y O. Arturo. Análítica de datos no estructurados para dar soporte a la toma decisiones en el área de comercialización de la Empresa Representaciones Batericar S.A.C. utilizando la metodología ICAV y la plataforma de Microsoft». Universidad Privada Antenor Orrego: Repositorio de Tesis UPAO. <https://hdl.handle.net/20.500.12759/6628>
- [19] C. Mejía y J. Rincón. «Comportamiento de la producción de café en Colombia usando técnicas de analítica de datos.» Repositorio de la Universidad Tecnológica de Pereira: Inicio. <https://repositorio.utp.edu.co/server/api/core/bitstreams/a9684a37-eabd-4008-befc-089068088f12/content>
- [20] ArcGIS. «Análisis espacial en ArcGIS Pro – ArcGIS Pro | Documentación». ArcGIS. <https://pro.arcgis.com/es/pro-app/latest/help/analysis/introduction/spatial-analysis-in-arcgis-pro.htm#:~:text=El%20análisis%20espacial%20permite%20resolver,realizar%20predicciones%20y%20tomar%20decisiones.>
- [21] J. González, A. Sánchez y J. Alonso. Optimización de la predicción de problemas financieros en empresas sanitarias privadas españolas aplicando algoritmos genéticos. <https://www.scielo.org/article/gS/2019.v33n5/462-467/es>
- [22] Naylamp mechatronics. Sensor de presión BMP 180. Naylamp Mechatronics - Perú. <https://naylampmechatronics.com/sensores-posicion-inerciales-gps/75-sensor-de-presion-bmp180.html>
- [23] Naylamp Mechatronics. Tutorial MPU6050, acelerómetro y giroscopio. Naylamp Mechatronics - Perú. [https://naylampmechatronics.com/blog/45\\_tutorial-mpu6050-acelerometro-y-giroscopio.html](https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html)
- [24] Naylamp Mechatronics. Tutorial módulo GPS con arduino. Naylamp Mechatronics - Perú. [https://naylampmechatronics.com/blog/18\\_tutorial-modulo-gps-con-arduino.html](https://naylampmechatronics.com/blog/18_tutorial-modulo-gps-con-arduino.html)
- [25] Unit Electronics. MQ-135 detector de calidad de aire. Unit Electronics. <https://uelectronics.com/producto/mq-135-modulo-detector-de-calidad-de-aire/>
- [26] VISTRONICA. Módulo sensor DHT22 AM2302 temperatura y humedad. VISTRONICA S.A.S. <https://www.vistronica.com/sensores/humedad/modulo-sensor-dht22-am2302-temperatura-y-humedad-detail.html>
- [27] VISTRONICA. «Board NodeMcu WIFI Wemos LOLIN32 4MB Basado En ESP32». <https://www.vistronica.com/board-de-desarrollo/board-nodemcu-wifi-wemos-lolin32-basado-en-esp32-detail.html>
- [28] Repositorio Sensorama SAJR-1221, <https://github.com/WorkJaime1221/Sensorama-SAJR-1221>
- [29] circuito.io. «Circuit Design App for Makers-circuito.io». Circuit Design App for Makers-circuito.io. <https://www.circuito.io/> (accedido el 23 de noviembre de 2022).