



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y
COMUNICACIONES

TEMA:

“PLATAFORMA CLOUD DE MONITOREO, ADQUISICIÓN,
VISUALIZACIÓN Y PREDICCIÓN DE LA CONTAMINACIÓN DEL AIRE,
BASADO EN MODELOS DE REDES NEURONALES ARTIFICIALES”

Trabajo de Graduación. Modalidad: Proyecto de Investigación, presentado previo la obtención del título de Ingeniero en Electrónica y Comunicaciones

SUBLÍNEA DE INVESTIGACIÓN:

Sistemas Embebidos

AUTOR: Juan Carlos Paredes Rodríguez

TUTOR: Dr. Ing. Alberto Ríos Villacorta

Ambato - Ecuador

Julio, 2018

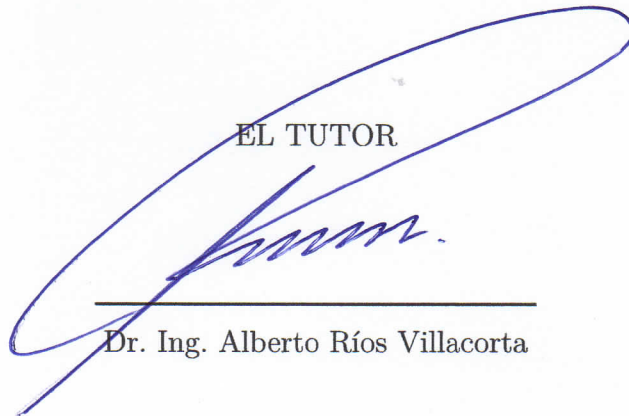
APROBACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Investigación sobre el Tema:

“PLATAFORMA CLOUD DE MONITOREO, ADQUISICIÓN, VISUALIZACIÓN Y PREDICCIÓN DE LA CONTAMINACIÓN DEL AIRE, BASADO EN MODELOS DE REDES NEURONALES ARTIFICIALES”, del señor Juan Carlos Paredes Rodríguez, estudiante de la Carrera de Ingeniería en Electrónica y Comunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato

Ambato, julio de 2018

EL TUTOR



Dr. Ing. Alberto Ríos Villacorta

AUTORÍA

El presente trabajo de investigación titulado: PLATAFORMA CLOUD DE MONITOREO, ADQUISICIÓN, VISUALIZACIÓN Y PREDICCIÓN DE LA CONTAMINACIÓN DEL AIRE, BASADO EN MODELOS DE REDES NEURONALES ARTIFICIALES, es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, julio de 2018



Juan Carlos Paredes Rodríguez

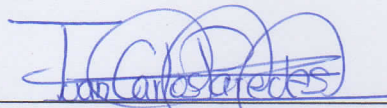
CC:1804798682

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, julio de 2018

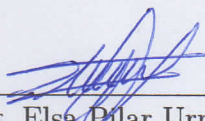


Juan Carlos Paredes Rodríguez

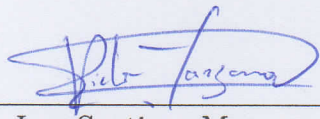
CC:1804798682

APROBACIÓN COMISIÓN CALIFICADORES

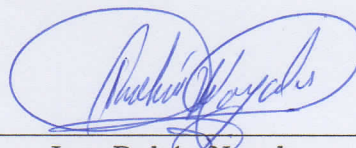
La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Santiago Manzano e Ing. Rubén Nogales, revisó y aprobó el Informe Final del trabajo de graduación titulado “**PLATAFORMA CLOUD DE MONITOREO, ADQUISICIÓN, VISUALIZACIÓN Y PREDICCIÓN DE LA CONTAMINACIÓN DEL AIRE BASADO EN MODELOS DE REDES NEURONALES ARTIFICIALES**”, presentado por el señor Juan Carlos Paredes Rodríguez de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato



Ing. Mg. Elsa Pilar Urrutia Urrutia
PRESIDENTE DEL TRIBUNAL



Ing. Santiago Manzano
DOCENTE CALIFICADOR



Ing. Rubén Nogales
DOCENTE CALIFICADOR

DEDICATORIA

Mi trabajo lo dedico con todo amor y cariño. A ti Dios porque me diste la oportunidad de vivir y darme una familia maravillosa.

Con mucho cariño a mis padres que me dieron la vida y han estado conmigo en todo momento. A mi madre por ser el pilar más importante en mi vida, por darme una carrera para el futuro y por creer en mí, aunque hemos pasado momentos difíciles, siempre has estado dándome tu apoyo y brindándome todo tu amor, por todo te agradezco de todo corazón tu apoyo.

A mis hermanos que me han apoyado en todo momento A mi tía por haberme brindado su apoyo durante toda la carrera universitaria. Gracias a todos ya que sin ustedes no hubiera sido posible lograr esta meta.

Juan Carlos Paredes Rodríguez

AGRADECIMIENTO

A dios por llegar a donde estoy, por darme la fortaleza en los momentos más difíciles.

A mi madre y hermanos por su amor y apoyo incondicional

A mi Tutor de trabajo de titulación Dr. Ing. Alberto Ríos quien me ayudó con su valioso conocimiento para la ejecución de la presente investigación.

A mis amigos por estar en esos momentos de necesidad.

A todos ustedes mi mas profundo agradecimiento.

Juan Carlos Paredes Rodríguez

ÍNDICE

APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
APROBACIÓN COMISIÓN CALIFICADORA	v
Dedicatoria	vi
Agradecimiento	vii
Introducción	xix
CAPÍTULO 1 El problema	1
1.1 Tema de Investigación	1
1.2 Planteamiento del problema	1
1.3 Delimitación	2
1.4 Justificación	3
1.5 Objetivos	4
1.5.1 General	4
1.5.2 Específicos	4
CAPÍTULO 2 Marco Teórico	5
2.1 Antecedentes Investigativos	5
2.2 Fundamentación teórica	7
2.2.1 La atmósfera	7
2.2.1.1 Contaminación del aire	8
2.2.1.2 Contaminantes Atmosféricos	10
2.2.1.3 Material Particulado	11
2.2.1.4 Transporte de contaminantes ambientales	13
2.2.1.5 Contaminación urbana	14
2.2.2 Computación en la Nube	17
2.2.2.1 Historia	17

2.2.2.2	Características	18
2.2.2.3	Arquitectura	19
2.2.2.4	Virtualización	20
2.2.2.5	Modelos de Cloud Computing	21
2.2.2.6	Clasificación del Cloud Computing	22
2.2.2.7	Ventajas del Cloud Computing	23
2.2.2.8	Riesgos del Cloud Computing	24
2.2.3	Historia de las Redes Neuronales	24
2.2.3.1	Neurona Biológica	25
2.2.3.2	Ventajas de una Red Neuronal Artificial	27
2.2.3.3	Aplicaciones de las Redes Neuronales Artificiales	27
2.2.3.4	Red Neuronal Artificial	28
2.2.3.5	Funciones de activación	30
2.2.3.6	Función escalón	30
2.2.3.7	Función Lineal mixta	31
2.2.3.8	Función tangente hiperbólica	32
2.2.3.9	Función sigmoideal	33
2.2.3.10	Función de Gauss	33
2.2.3.11	Elementos de una Red Neuronal Artificial	34
2.2.3.12	Entrenamiento de las Redes Neuronales	34
2.2.3.13	Perceptron	36
2.2.3.14	Red BackPropagation	38
2.2.3.15	Propagación hacia adelante	38
2.2.3.16	Propagación hacia atrás	39
CAPÍTULO 3 Metodología		41
3.1	Modalidad de la investigación	41
3.2	Recolección de información	41
3.3	Procesamiento y análisis de datos	41
3.4	Desarrollo del Proyecto	42
CAPÍTULO 4 Desarrollo de la propuesta		44
4.0.1	Área de estudio	44
4.0.2	Diseño del Prototipo	46
4.0.2.1	Sensor de polvo PPD42	47
4.0.2.2	Modulo GPS	48
4.0.2.3	Sensor de temperatura y Humedad	49
4.0.2.4	Arduino	50

4.0.2.5	Conexión del sistema	51
4.0.2.6	Algoritmo del programa principal	52
4.0.2.7	Algoritmo del módulo WiFi	54
4.0.3	Plataforma Cloud	56
4.0.3.1	Configuración de Google Cloud Computing	57
4.0.3.2	Almacenamiento de Datos	59
4.0.4	Implementación de la plataforma web	60
4.0.4.1	Tablas de valores	62
4.0.4.2	Gráficas	62
4.0.5	Predicción basado en Redes Neuronales Artificiales	64
4.0.5.1	Número de Capas	64
4.0.5.2	Número de Neuronas	65
4.0.5.3	Algoritmo de entrenamiento	65
4.0.5.4	Extracción de datos de entrenamiento	66
4.0.5.5	Preprocesamiento de datos de entrenamiento	66
4.0.5.6	Procesamiento de datos de entrenamiento	66
4.0.5.7	Programa Matlab	67
4.0.5.8	Simulación	68
4.0.6	Resultados	80
CAPÍTULO 5 Conclusiones y Recomendaciones		81
5.1	Conclusiones	81
5.2	Recomendaciones	81
Bibliografía		83
ANEXOS		88

ÍNDICE DE TABLAS

2.1	Composición del aire seco [1].	8
2.2	Comparación entre partículas finas y gruesas [2]	12
2.3	Modelos de redes neuronales artificiales mas usados en la actualidad [3]	36
4.1	Valores de material particulado registrados por el GADMA para el año 2017 [4].	46
4.2	Especificaciones técnicas del sensor PPD42	48
4.3	Características del GPS NEO UBLOX 6M	49
4.4	Características del sensor DHT11	50
4.5	Especificaciones técnicas de la plataforma arduino uno	51
4.6	Tabla de errores al entrenar los distintos modelos de redes neuronales	80

ÍNDICE DE FIGURAS

2.1	Ciclo de los contaminantes atmosféricos [5].	11
2.2	Intercambio de contaminantes liberados por la antroposfera [5]	14
2.3	Generación de contaminantes primarios y secundarios en la atmósfera urbana [5].	16
2.4	Evolución del cloud computing a lo largo del tiempo [6].	18
2.5	Características del cloud computing [6]	19
2.6	Modelo del cloud computing [7].	20
2.7	Elementos de una Red Neuronal Biológica [8].	26
2.8	Esquema de una red neuronal artificial [9]	30
2.9	Función de activación escalón [8].	31
2.10	Función de activación lineal y mixta [8].	32
2.11	Función de activación tangente hiperbólica [8].	32
2.12	Función de activación sigmooidal [8].	33
2.13	Función de Gauss [8].	33
2.14	Esquema básico de una red neuronal artificial [8].	34
2.15	Unidad procesadora básica del perceptron [3].	37
2.16	Unidad de procesamiento básico backpropagation [3].	38
4.1	Mapa de la ciudad de Ambato (Google Earth)	44
4.2	Datos de material particulado PM2.5 obtenidos por la estación de monitoreo del GADMA correspondientes al año 2016 [4].	45
4.3	Datos de material particulado PM2.5 obtenidos por la estación de monitoreo del GADMA correspondientes al año 2017 [4].	46
4.4	Diagrama de bloques del sistema de medición de PM10 y PM2.5	47
4.5	Elementos internos del sensor PPD42	48
4.6	GPS NEO UBLOX 6M	49
4.7	Sensor de Temperatura y Humedad DHT11	50
4.8	Arduino Uno R3	51
4.9	Diagrama de conexión del sistema de medición de PM10 y PM2.5	52
4.10	Diagrama de flujo del programa principal (arduino)	54
4.11	Diagrama de flujo del módulo WiFi	56

4.12	Creación de una instancia en Google Cloud Platform.	58
4.13	Zonas disponibles Google Cloud Platform	59
4.14	Campos de la base de datos MySQL	60
4.15	Página principal para la visualización de los datos obtenidos por la estación de monitoreo de PM10 y PM2.5	61
4.16	Tabla de valores actuales registrados por la estación móvil de monitoreo de las estación móvil	62
4.17	Gráfica del material particulado PM10 registrado por la estación de monitores	63
4.18	Gráfica de material particulado PM2.5 registrado por la estación de monitoreo	63
4.19	Gráfica de la temperatura del ambiente registrado por la estación de monitoreo	64
4.20	Gráfica de la humedad del ambiente registrado por la estación de monitoreo	64
4.21	Representación den número de capas de la red neuronal.	65
4.22	Organización de los datos de entrenamiento de la red neuronal.	67
4.23	Modelo de red neuronal con 16 neuronas en la capa intermedia.	68
4.24	Gráfica de la validación de los datos de entrenamiento del modelo con 16 neuronas en la capa intermedia	69
4.25	Regresión lineal del entrenamiento del modelo con 16 neuronas en la capa intermedia.	70
4.26	Valores reales vs valores obtenidos a la salida del modelo de 16 neuronas en la capa intermedia	71
4.27	Modelo de red neuronal con 16 neuronas en la capa intermedia.	71
4.28	Gráfica de la validación de los datos de entrenamiento del modelo con 32 neuronas en la capa intermedia	72
4.29	Regresión lineal del entrenamiento del modelo con 32 neuronas en la capa intermedia.	73
4.30	Valores reales vs valores obtenidos a la salida del modelo de 32 neuronas en la capa intermedia	74
4.31	Modelo de red neuronal con 16 neuronas en la capa intermedia.	74
4.32	Gráfica de la validación de los datos de entrenamiento del modelo con 32 neuronas en la capa intermedia	75
4.33	Regresión lineal del entrenamiento del modelo con 32 neuronas en la capa intermedia.	76

4.34	Valores reales vs valores obtenidos a la salida del modelo de 32 neuronas en la capa intermedia	77
4.35	Modelo de red neuronal compuesto por 16 neuronas en la capa de entrada, 32 neuronas en la capa intermedia y una neurona en la capa de salida.	77
4.36	Validación de los datos de entrenamiento del modelo de red neuronal compuesto por 16 neuronas en la capa de entrada, 32 neuronas en la capa intermedia y una neurona en la capa de salida.	78
4.37	Regresión lineal del modelo compuesto por 16 neuronas en la capa de entrada, 32 neuronas en la capa intermedia y una neurona en la capa de salida	79
4.38	Gráfico de los valores objetivo vs valores obtenidos a la salida del modelo de red neuronal compuesto por 16 neuronas en la capa intermedia.	80

Resumen

En la presente investigación se propone medir la concentración de las partículas en suspensión que varían entre $2.5 \mu g$ y $10 \mu g$ también conocido como PM10 y las partículas en suspensión menores a $2.5 \mu g$ conocido como PM2.5, en el campus Huachi de la Universidad Técnica de Ambato. El objetivo es determinar los niveles de concentración de estos contaminantes y su probable afección a corto y largo plazo en la salud de los habitantes de la zona. La evaluación consiste en la construcción de una plataforma móvil para la medición de la concentración de los contaminantes PM10 y PM2.5. Dicha plataforma consta de un sensor de polvo PPD42, el que registra las partículas de polvo en un determinado tiempo, el sensor consta de un diodo emisor de luz y un detector de fotodiodos. La plataforma dispone de un sensor DHT11 para la medición de la temperatura y humedad del ambiente y de un módulo GPS Ublox NEO 6M para el registro del posicionamiento de la plataforma móvil. Todos los valores de los sensores son procesados por una tarjeta arduino uno R3 y enviados a una base de datos MySQL en un servidor remoto. La base de datos ha sido instalada y configurada en el sistema operativo Centos 7. Además, se ha configurado un servicio apache para que cualquier persona con acceso a internet pueda visualizar los valores por medio de una página web. Centos 7 está instalado como una instancia sobre Google Cloud Platform. Un servicio PaaS de computación en la nube que ofrece Google. Por último, se realiza la predicción de las concentraciones de material particulado PM10 y PM2.5 utilizando Redes Neuronales artificiales. En el presente proyecto se emplea un perceptrón multicapa que consta de una capa de entrada, una capa intermedia y una capa de salida. La red neuronal artificial ha sido programada utilizando el toolbox de Redes neuronales de Matlab.

Abstract

In the present investigation it is proposed to measure the concentration of suspended particles ranging from 2.5 μg to 10 μg also known as PM10 and particles in suspension less than 2.5 μg known as PM2.5, on the Huachi campus of the Technical University of Ambato. The objective is to determine the levels of concentration of these pollutants and their probable short-term and long-term effects on the health of the inhabitants of the area. The evaluation consists of the construction of a mobile platform for measuring the concentration of PM10 and PM2.5 pollutants. This platform consists of a PPD42 dust sensor, which records the dust particles in a certain time, the sensor consists of a light emitting diode and a photodiode detector. The platform has a DHT11 sensor for measuring the temperature and humidity of the environment and a GPS module Ublox NEO 6M for the registration of the positioning of the mobile platform. All the values of the sensors are processed by an arduino card one R3 and sent to a MySQL database on a remote server. The database has been installed and configured in the operating system Centos 7. In addition, an apache service has been configured so that anyone with Internet access can view the values through a web page. Centos 7 is installed as an instance on Google Cloud Platform. A PaaS cloud computing service offered by Google. Finally, the prediction of PM10 and PM2.5 particulate material concentrations is made using artificial Neural Networks. In the present project, a multilayer perceptron is used consisting of an input layer, an intermediate layer and an output layer. The artificial neural network has been programmed using the toolbox of neuronal networks of Matlab.

Glosario de términos y acrónimos

- **Antropogénica:** Acciones humanas que influyen en el medio ambiente.
- **Amenaza:** Factor de riesgo extremo de un sujeto o sistema, representado por un peligro latente asociado con un fenómeno.
- **Calidad Ambiental:** Capacidad relativa de un medio ambiente para satisfacer las necesidades o los deseos de un individuo o sociedad.
- **Contaminación:** Proceso de entropía causado por la actividad humana en contra de las tendencias que determinan el equilibrio propio de los seres vivos.
- **Ecosistema:** Unidad espacial definida por un complejo de componentes y procesos físicos y bióticos que interactúan en forma independiente.
- **Impacto Ambiental:** Administración integrada del ambiente con criterio de equidad.
- **Morfología:** Parte de la biología que estudia la forma de los seres orgánicos y de las modificaciones o transformaciones que experimenta.
- **Termodinámica:** Disciplina parte de la física que se encarga del estudio de las relaciones que se establece entre el calor y el resto de las formas de energía.
- **Antropogénico:** Acciones humanas que influyen en el medio ambiente.
- **Freón:** Refrigerantes compuestos por clorofluorocarbonos (CFC), productos dañinos para la capa de ozono.
- **Ubicuo:** Que está presente en todas las partes al mismo tiempo.
- **Fusiforme:** Que tiene forma alargada, elipsoide y con extremidades mas estrechas que el centro.
- **Piriforme:** Que tiene forma de pera.
- **Sinapsis:** Aproximación intercelular entre neuronas.
- **Mielina:** Sustancia que envuelve y protege los axones, su función principal es aumentar la velocidad de transmisión del impulso nervioso.
- **Disponibilidad:** Medida que indica cuanto tiempo esta un sistema operativo.

- **Servidor:** Dispositivo final que ofrece varios servicios a otros dispositivos clientes.
- **Centro de Datos:** Infraestructura donde existe servidores, equipos de conmutación, enlaces de comunicaciones, equipo de virtualización, almacenamiento, equipos de seguridad perimetral, equipos de balanceo de carga.
- **Clouding:** Servicios ofrecidos por un proveedor.
- **Ambiente:** Región, alrededores y circunstancias en las que se encuentra un ser o un objeto.
- **Anaerobio:** Proceso químico o condición ambiental que sucede en ausencia de oxígeno.
- **Asentamiento:** Instalación provisional permitida por el gobierno de colonos o agricultores.
- **Atmósfera:** La masa total de aire que circula alrededor de la tierra.
- **Biodiversidad:** Variabilidad de los organismos vivos en cualquier fuente.
- **Biosfera:** Fin de la capa de cobertura de la tierra que contienen el sustento de la vida.
- **Combustibles fósiles:** Aquellos constituidos por restos fósiles de organismos vivos.
- **Contaminación:** Liberación de sustancias de manera directa o indirecta que causan afectos diversos sobre el medio ambiente y los seres vivos,
- **Contaminación del aire:** Presencia en la atmósfera de sustancias resultantes de la actividad humana o de procesos naturales en concentración y tiempo suficiente que afectan el bienestar de las personas y del medio ambiente.
- **Contaminante:** Material o residuo que es agente de riesgo.
- **Contaminante Natural:** Emitido por los diferentes procesos naturales del océano, bosques, volcanes, pantanos, tempestades eléctricas, etc.

INTRODUCCIÓN

La contaminación del aire es un problema a nivel mundial, que afecta a los seres humanos, plantas, animales, etc. Se define como la incorporación a la atmósfera de cualquier sustancia que altere su composición natural, sus propiedades físicas y químicas. Es uno de los problemas más relevantes que se manifiesta en los asentamientos urbanos, industrial y minero. Los contaminantes provienen de los medios de transporte y otras fuentes que hacen uso de motores a combustión. La Organización Mundial de la Salud, (OMS), estima que una de cada nueve muertes en todo el mundo es ocasionado por los altos niveles de contaminantes suspendidos en la atmósfera. Uno de los principales problemas ocasionados por la contaminación del aire son los problemas respiratorios. Los niños y las personas adultas son los mayores afectados por este problema. Las partículas suspendidas en el aire contaminado están relacionadas con la creciente incidencia de cáncer, especialmente cáncer de pulmón, además de cáncer en las vías urinarias y vejiga [10].

Ambato es una ciudad urbana con una alta concentración de industrias que genera muchos contaminantes suspendidos en la atmósfera. Sin embargo, los niveles de contaminación no sobrepasan los $20 \text{ ug}/\text{m}^3$ recomendados por la Organización Mundial de la Salud. Una circunstancia que motiva a implementar sistemas de vigilancia basadas en estaciones de medición de calidad del aire. En la actualidad el GAD Municipal de Ambato vigila continuamente la calidad del aire por medio de 13 estaciones de monitoreo, que registran principalmente los niveles de Material Particulado sedimentable y material particulado suspendido inferior a 2.5 ug (PM2.5).

La mayoría de los sistemas de supervisión de los niveles de contaminación, están formados por una estación fija de recolección de datos. Su principal limitación es el costo excesivo de los equipos y su gran tamaño, que dificulta el traslado a diferentes lugares donde se requiera hacer un mayor control. Por lo expuesto anteriormente se plantea el desarrollo de la presente investigación, cuyo objetivo es diseñar una plataforma móvil para la medición de la concentración de material particulado PM10 y PM2.5. Así mismo se propone realizar predicciones a corto plazo mediante el uso de Redes Neuronales Artificiales. Toda esta información deberá ser accesible para cualquier persona desde una pagina web. Su implementación es de bajo coste debido a la utilización de un sistema de sensores basados en hardware y software libre. La aplicación se enfoca en concientizar los niveles de contaminación y evaluar el impacto

en la salud humana de los habitantes de la ciudad de Ambato.

En el capítulo I de la presente investigación se describe la problemática y las afecciones que tiene la contaminación del aire en la salud de las personas. Además de la factibilidad del desarrollo de un prototipo de plataforma cloud para el monitoreo, adquisición, visualización y predicción de la contaminación del aire basado en modelos de redes neuronales artificiales.

En el capítulo II se hace mención a los trabajos investigativos relacionados con monitoreo y predicción de la contaminación del aire basado en redes neuronales artificiales. Este capítulo incluye la fundamentación teórica del proyecto de investigación y concluye con propuesta de solución del problema.

En el capítulo III se detalla la metodología utilizada en el desarrollo del presente proyecto de investigación.

En el capítulo IV se describe la realización del prototipo para la medición y almacenamiento de las variables de material particulado PM10, PM2.5, temperatura y humedad en la plataforma cloud. Además del desarrollo de una página web para la visualización de dichas variables. También se lleva a cabo el entrenamiento de una red neuronal para la predicción de la contaminación del aire utilizando el software Matlab .

Finalmente, en el capítulo V se redactan las conclusiones y recomendaciones obtenidas del presente trabajo de investigación.

CAPÍTULO 1

El problema

1.1. Tema de Investigación

Plataforma cloud de monitoreo, adquisición, visualización y predicción de la contaminación del aire, basado en modelos de Redes Neuronales Artificiales.

1.2. Planteamiento del problema

El aire es un recurso esencial para la vida y su contaminación constituye una agresión a los seres humanos y otras especies del planeta. En un informe realizado por la Organización Mundial de la Salud, OMS, en 2016, señala que 12.6 millones de personas perdieron la vida en 2012 por vivir y trabajar en ambientes contaminados [11]. Asia Sudoriental sufre unos de los mayores problemas de contaminación del aire, esto se puede evidenciar ya que solo en esta región se producen 3.5 millones de muertes anualmente, 2,2 millones en la región de África, 1,4 millones en la región de Europa, 0,85 millones en la región del mediterráneo Oriental y 0,84 millones en el continente Americano [12].

La contaminación del aire es un fenómeno que tiene mayor afección sobre las personas mayores y niños. En un informe realizado por la Organización Mundial de la Salud, OMS, en 2017, estima que más de una cuarta parte de las defunciones de niños menores de 5 años son consecuencia de la contaminación ambiental [13]. Las condiciones insalubres del entorno, así como el aire contaminado en espacios cerrados y exterior; la exposición al humo de tabaco causa la muerte de 1.7 millones de niños menores de 5 años [13]. El 72 % de las defunciones prematuras relacionadas con la contaminación del aire exterior se deben a cardiopatía isquémica y accidentes cerebrovasculares, mientras que un 14 % se debieron a neumopatía obstructiva crónica o infección agudas de las vías respiratorias, y un 14 % a cáncer de pulmón [10].

En el 2013, el Centro Internacional de Investigadores del cáncer de la Organización Mundial de la Salud, OMS, determinó que la contaminación del aire exterior es cancerígena para el ser humano y que las partículas de aire contaminado están

relacionadas con la creciente incidencia del cáncer, especialmente cáncer de pulmón. También se observó una relación entre la contaminación del aire exterior y el aumento del cáncer de las vías urinarias y vejiga [13].

En 2013, Clear Air Institute, señaló que cada año más de 70 mil personas mueren a consecuencia de la contaminación del aire, además, en América Latina y el Caribe existen al menos 100 millones de personas expuestas a niveles de contaminación atmosférica superiores recomendados por la OMS de $20 \mu\text{g}/\text{m}^3$ [14]. Brasil es el país que lidera el ranking de los países más contaminados de la región, con 24000 muertes por año. Solo en la ciudad de Sao Pablo fallecen anualmente 4600 personas, el triple de las muertes provocadas por accidentes de tráfico. México, con alrededor de 15000 fallecidos, y Argentina, con 10000 muertes ocupan el segundo y tercer lugar respectivamente [14].

En 2015, la Universidad San Francisco de Quito señaló que el tráfico vehicular y las gasolineras constituyen el 95 % de gases producidos que alteran el aire [15]. El 76 % de la contaminación atmosférica proviene de los vehículos, el 5 % de la industria, el 4 % de la generación eléctrica y el 15 % de fuentes naturales. Estas fuentes de contaminación aportan 102.667,4 t/año de dióxido de carbono (CO_2), 20.808 t/año de óxidos de nitrógeno (NO_x), 13.185,6 t/año de compuestos orgánicos volátiles distintos del metano (COVNM), 656,5 t/año de MP_{10} , 1.211,2 t/año de $\text{MP}_{2.5}$ y 451,1 t/año de dióxido de azufre (SO_2) resultando 2,8 millones de toneladas de dióxido de carbono, dando como resultado la ciudad con más contaminación en el país [15].

En 2014, el Ministerio del Ambiente, señaló que en la ciudad de Ambato, una de las mayores problemáticas que sufre el ser humano es la contaminación ambiental [16]. Existen 360 unidades de transporte urbano, de las cuales se estima que un 10 % no debe estar en circulación por haber cumplido el tiempo de vida útil. La combustión incompleta del motor produce problemas en la salud especialmente en la respiración e inflamación de las fosas nasales. El Ministerio del Ambiente afirma que el 88 % de las emisiones atmosféricas son provocadas por el tráfico vehicular, el 1 % por las Industrias y el 11 % por GLP doméstico [16].

1.3. Delimitación

Área académica: Física y Electrónica

Línea de Investigación: Sistemas Electrónicos

Sublínea de Investigación: Sistemas Embebidos

Delimitación Espacial: La presente Investigación de desarrollará en la Universidad Técnica de Ambato, Campus Huachi

Delimitación Temporal: La presente investigación se desarrollará en el periodo septiembre 2017 – febrero 2018 de acuerdo a lo establecido en el Reglamento de Graduación para obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato, aprobado mediante Resolución 0533-CU-P-2015 del 31 de marzo de 2015 (RGTTTNUA).

1.4. Justificación

En la actualidad, los principales problemas de contaminación ambiental se deben a los gases tóxicos emitidos por los vehículos a motor de combustión, así como, al incremento de las actividades industriales, con impacto directo sobre la salud pública y el medio ambiente. La exposición de las personas a contaminantes del aire, dependiendo de factores tales como concentración y tiempo, puede agravar o provocar enfermedades respiratorias especialmente a la población mas vulnerable como los niños y ancianos. La sociedad debe tener conciencia de la importancia de evaluar y contribuir con la reducción del impacto de las actividades industriales en el aire. El empleo de un sistema de medición y predicción de las concentraciones de material particulado PM10 y PM2.5 permitirá conocer la concentración de estos contaminantes; además de generar alertas a corto plazo cuando estos componentes sobrepasen los niveles indicados por la Organización Mundial de la Salud, OMS.

Los beneficiarios de este proyecto son las personas de la Universidad Técnica de Ambato, Campus Huachi, que podrán hacer uso de los avances tecnológicos a bajo costo para contar con datos valiosos de los contaminantes atmosféricos emitidos en un área geográfica. Para conocer las condiciones ambientales a las que se encuentran expuestos a diario es necesario elaborar mecanismos que permitan evaluar el impacto ambiental.

El desarrollo de este proyecto es factible tanto en la parte intelectual, técnica y económica, también al disponer de todos los componentes necesarios para la implementación así como una gran variedad de alternativas en el mercado con precios módicos y convenientes, además del conocimiento necesario para la creación del

prototipo electrónico.

1.5. Objetivos

1.5.1. General

Implementar una plataforma cloud de monitoreo, adquisición, visualización y predicción de la contaminación del aire, basado en modelos de Redes Neuronales Artificiales

1.5.2. Específicos

- Analizar datos históricos de contaminación del aire en la ciudad de Ambato.
- Implementar un sistema de medición de la contaminación del aire basado en hardware libre.
- Entrenar un modelo de Red Neuronal Artificial en la predicción a un día de la contaminación del aire.
- Implementar la plataforma cloud de monitoreo, adquisición, visualización y predicción de la contaminación del aire.

CAPÍTULO 2

Marco Teórico

2.1. Antecedentes Investigativos

El análisis de fuentes especializadas en relación a investigaciones asociadas al monitoreo y contaminación del aire, basado en modelos de Redes Neuronales Artificiales, permite describir los siguientes trabajos:

En 2011, Gabriel Ibarra-Berastegi, Jon Saenz y Agustin Ezcurra en el artículo denominado “A neural network model forecasting for prediction of hourly ozone concentration in Corsica”, se enfocan la predicción de los niveles de SO_2 , CO , NO_2 , NO y O_3 cada 8 horas en seis localidades de la ciudad de Bilbao. Han sido construidos 216 modelos y una base de datos meteorológicos existentes en la zona correspondiente al año 2000. Además se usa Matlab como software para el desarrollo de los modelos de Redes Neuronales Artificiales. Para el caso de zonas urbanas, concluyen que los modelos con mejor rendimiento son las redes Perceptrón Multicapa (MPL), Red de Base Radial (RBF) y General Regression Neural Network (GRNN) [17].

En 2011, Christophe Paoli, Gilles Notton y Marie-Laure Nivet en el artículo denominado “A neural network model forecasting for prediction of hourly ozone concentration in Corsica”, realizan la predicción de la concentración de ozono en la isla de Corsica usando una red neuronal perceptrón multicapa (MLP). El modelo de Red Neuronal es desarrollado en Matlab haciendo uso del toolbox de Redes Neuronales. Como resultado se obtiene un error del 10.5 % al ser evaluados con el índice Mean Absolute Percentage Error (MAPE) [18].

En 2014, Yuan Chen, Hui Qin y ZhiGang Zhou en el artículo denominado “A comparative study on multi-regression analysis and bp neural network of pm2.5 index”, utilizan datos meteorológicos y concentraciones de PM2.5 de la ciudad de Wuhan para construir un modelo de predicción de la contaminación atmosférica basado en análisis de regresión múltiple y redes neuronales BP. El software Matlab es usado para el proceso de entrenamiento de las Redes Neuronales. Al realizar un estudio comparativo entre los dos métodos abordados en esta investigación, se obtiene que la predicción es más acertada al usar el modelo de redes neuronales BP [19].

En 2015, Chen Xiaojun, Liu Xianpeng y Xu Peng en el artículo denominado “Iot-based air pollution monitoring and forecasting system”, desarrollan el monitoreo de la contaminación del aire en tiempo real usando IoT. Estos sistemas pueden reducir los costos 1/10 y pueden ser usados en grandes áreas como una red de sensores. Además de las funciones tradicionales, este sistema también presenta la predicción de la contaminación atmosférica dentro de un cierto intervalo de tiempo, esto lo consiguen analizando los datos obtenidos por el sistema de acuerdo con la tecnología de redes neuronales artificiales [20].

En 2015, Min Huang, Tao Zhang y Jingyang Wang en el artículo denominado “A new air quality forecasting model using data mining and artificial neural network”, implementan un modelo de predicción de la contaminación atmosférica, basado en técnicas de minado de datos y redes neuronales back propagation. Los datos para el entrenamiento de la red neuronal artificial son obtenidos de estaciones de monitoreo de calidad de aire de la ciudad de Shijiazhuang. Matlab es el software empleado para el desarrollo de esta investigación. Principalmente se usa un modelo de minado de datos para encontrar factores que afectan la calidad del aire, segundo, se usan esos factores para entrenar la red neuronal. Este proceso reduce la carga computacional. [21].

En 2016, Jessica Villa y Fernando Daquilema en el proyecto de investigación, “Diseño e implementación de un sistema para el monitoreo de los niveles de contaminación del aire”, para el proyecto de investigación de la Universidad Nacional de Chimborazo”, realizan la implementación de un sistema de monitoreo de la contaminación del aire, utilizando el estándar GSM conjuntamente con arduino, para el registro y visualización de datos de los sensores de monóxido de carbono y dióxido de carbono en áreas estratégicas de la ciudad de Riobamba. Concluyen que con el desarrollo de la aplicación web se pueden observar los datos de los sensores sin importar la ubicación de los mismos, todo esto en tiempo real. [22].

En 2016, Maja Dedovic, Irfan Turkovic, Tatjana Konjic, Samir Avdakovic, Nediz Dautbasic, en el artículo denominado “Forecasting PM_{10} concentrations using neural networks and system for improving air quality”, entrenan una red neuronal artificial desarrollada en el software Matlab para predecir las concentraciones de PM_{10} en la ciudad de Sarajevo. Los datos de entrada para el modelo propuesto son variables meteorológicas como velocidad del viento, humedad, temperatura y presión. El modelo propuesto es probado en varios casos y se concluye que las aproximaciones son muy satisfactorias para predecir los niveles de concentración de PM_{10} [23].

En 2016, Khaled Shaban, Abdullah Kadri y Eman Rezk, en el artículo titulado “Urban air pollution monitoring system with forecasting models”, implementan un sistema de monitoreo y predicción de la contaminación urbana del aire. El sistema usa sensores de bajo costo, equipados con una cadena de sensores gaseosos y meteorológicos. Estos sensores se comunican con una plataforma responsable de recibir y almacenar los datos, procesar, predecir los contaminantes basados en información histórica y finalmente compartir la información a través de una aplicación móvil, portal web y servicio de mensajes. Como resultado se obtienen que la predicción de los datos son pobres al usar Redes Neuronales Artificiales, debido a que estos necesitan de una gran cantidad de información en la etapa de entrenamiento. [24].

En 2016, Polaiiah Bojja y Vivith Kumar Karumuri en el artículo denominado “Development and evaluation of pollution forecasting model using soft-computing methods for PM_{10} and SO_2 in ambient air”, utilizan una red neuronal artificial y lógica difusa para medir y predecir las concentraciones de material particulado PM_{10} y SO_2 en la ciudad de Tirumala. Para el entrenamiento de la red neuronal Feed-Forward Back Propagation y el modelo de inferencia de Mamdani se usan datos de cinco años atrás, Matlab como software para mejorar la precisión de medición y evaluación del error mínimo de pronóstico. Los resultados obtenidos al usar lógica difusa son similares que al usar redes Neuronales Artificiales. [25].

2.2. Fundamentación teórica

2.2.1. La atmósfera

La atmósfera consiste en una capa de gases que cubren la superficie de la tierra. El aire está constituido por 78.1 % de nitrógeno, 20.9 % de oxígeno, 0.93 % de argón y 0.03 % de dióxido de carbono. El aire contiene 3 % de vapor de agua en volumen, además, contiene una gran variedad de gases en pequeñas porciones por debajo del 0.03 %, como neón, helio, metano, kriptón, óxido nitroso, hidrógeno, xenón, dióxido de azufre, ozono, dióxido de nitrógeno, amoníaco y monóxido de carbono, como se puede observar en el cuadro 2.1. El comportamiento de la atmósfera es consecuencia de los gases que contiene, y son procedentes de fuentes naturales como fuentes antropogénicas y de las fuerzas físicas que actúan sobre ellas [26].

Componente	Fórmula	Proporción en volumen
Nitrógeno	N_2	78.1
Oxígeno	O_2	20.9
Argón	Ar	0.93
Dióxido de carbono	CO_2	0.03
Neón	Ne	1.5×10^{-3}
Helio	He	5×10^{-4}
Metano	CH_4	2×10^{-4}
Hidrógeno	H_2	5×10^{-4}
Xenón	Xe	1×10^{-5}
Ozono	O_3	7×10^{-6}

Tabla 2.1: Composición del aire seco [1].

La atmósfera consiste en una capa protectora que rodea la superficie de la tierra y es la fuente de dióxido de carbono para la fotosíntesis de las plantas y de oxígeno para la respiración. Proporciona el nitrógeno que las bacterias y las plantas productoras de amoníaco emplean para producir el nitrógeno enlazado químicamente, que es un componente esencial de las moléculas de los seres vivos. La atmósfera transporta el agua de los océanos a la tierra, desgraciadamente también es usada como un lugar de descarga de muchos contaminantes, desde el dióxido de azufre al refrigerante freón [26].

2.2.1.1. Contaminación del aire

La contaminación del aire se define como la incorporación a la atmósfera de cualquier sustancia que altere su composición natural y sus propiedades físico-químicas. Ambientalmente, es uno de los problemas más relevantes que se manifiestan en los asentamientos de tipo urbano, industrial y minero. Los contaminantes provienen principalmente de los medios de transporte como vehículos motorizados a combustión y fuentes fijas como las industrias. Además, se originan por fenómenos naturales como erupciones volcánicas y otras reacciones biológicas. El movimiento del aire a través de la superficie de la tierra es un factor crucial para creación y dispersión de los fenómenos de contaminación atmosférica. Al reducirse el movimiento del aire se produce un estancamiento con un aumento de los contaminantes en regiones localizadas [27].

La contaminación antropogénica del aire tiene sus orígenes desde hace unos 500 años, en la época de la industrialización. En América Latina se definen cuatro

rasgos de la industrialización: participación en el mercado internacional; estructura industrial concebida para el mercado interno; aspiración a reproducir el modo de vida de los países avanzados, y limitada valoración social de la función empresarial y precario liderazgo del empresario nacional público y privado [28]. La alteración de la composición atmosférica es un proceso irreversible que tiene gran afección sobre los seres humanos, animales y vegetales. Los primeros casos severos, reconocidos históricamente, se dieron en la ciudad de Londres, en el año 1952, donde los altos niveles de contaminación atmosférica se asociaron con el incremento del número de muertes, 400 casos aproximadamente [29].

La contaminación atmosférica es el principal riesgo ambiental para la salud de las personas. La Organización Mundial de la Salud, OMS, estima que, una de cada nueve muertes en todo el mundo es consecuencia de la contaminación atmosférica. Los contaminantes más relevantes para la salud son el material particulado (PM) con diámetro de 10 micras o inferior, que pueden penetrar profundamente en los pulmones e inducir la reacción de la superficie y las células de defensa. La mayoría de estos contaminantes se originan por la quema de combustibles fósiles. Las directrices de la Organización Mundial de la Salud (OMS), recomienda una exposición máxima de $20\mu\text{g}/\text{m}^3$ para PM_{10} y una exposición máxima de $10\mu\text{g}/\text{m}^3$ de $PM_{2.5}$.

Los riesgos y efectos sobre la salud no están distribuidos equitativamente. Las personas con enfermedades previas, los niños menores de 5 años y las personas adultas son los más afectados. Las personas pobres, las mujeres y sus hijos que utilizan estufas tradicionales de biomasa para cocinar y calentarse, también corren mayor riesgo. Hay efectos sobre la salud a corto y largo plazo, siendo la exposición a largo plazo y de larga duración la más significativa para la salud pública. La mayoría de muertes, relacionada con la contaminación atmosférica, en la población están relacionadas con enfermedades no transmisibles. En efecto, el 36 % de las muertes se deben a cáncer de pulmón, el 35 % de enfermedad pulmonar obstructiva crónica (COPD), el 34 % a por accidentes cerebrovasculares y el 27 % por cardiopatías isquémicas. Sin embargo, el mayor impacto recae sobre los niños menores de 5 años, ya que más de la mitad de muertes son producidas por infecciones de las vías respiratorias inferiores (ALRI) [30].

En una investigación realizada por el Centro de Investigaciones sobre el cáncer de la Organización Mundial de la Salud, OMS, en 2013, determinó que la contaminación del aire exterior es cancerígena para el ser humano y que las partículas ultrafinas están estrechamente relacionadas con la creciente incidencia del cáncer, especialmente cáncer de pulmón. También se ha observado una relación entre la

contaminación exterior y el aumento del cáncer de vías urinarias y vejiga [30].

2.2.1.2. Contaminantes Atmosféricos

Una fuente natural de emisión de componentes atmosféricos son los océanos, los productos disueltos en el agua del mar se encuentran en perfecto equilibrio con la atmósfera, no obstante, la variación de temperatura de la superficie marina provoca que se rompa este equilibrio y se altere las concentraciones de los gases disueltos en los océanos. En este sentido, los océanos emiten una gran variedad de gases como monóxido de carbono CO , dióxido de carbono CO_2 , metano CH_4 , óxido de nitrógeno N_2O , sulfuro de carbono (S_2C), sulfuro de carbonilo (SCO), cloruro de metilo ($CLCH_3$), etc [5].

Los volcanes son otra fuente de contaminación de la atmósfera, cuando se encuentran en erupción emiten gran cantidad de gases y partículas sólidas a elevadas temperaturas capaces de alcanzar grandes distancias. Un volcán emite óxidos de nitrógeno y de azufre, sulfuro de hidrógeno H_2S , sulfuro de carbonilo SCO , cloruro de hidrógeno HCL , ácido fluorhídrico HF , así como cenizas y partículas sólidas procedentes de procesos de combustión [5].

Otras fuentes de contaminación son los incendios forestales, mediante los cuales se emiten grandes cantidades de dióxido de carbono CO_2 , monóxido de carbono CO , óxidos de nitrógeno, SO_2 , clorometano CLH_3 , ácido cianhídrico (HCN), etc. Las descargas eléctricas en tormentas son las responsables de la generación de óxidos de nitrógeno [5].

Otro foco de contaminación es el hombre, que vierte residuos a la atmósfera de forma incontrolada, insistente y en amplias zonas del planeta. El 80 % de los contaminantes que genera el hombre proceden de los componentes residuales formados durante la combustión de combustibles fósiles. Además del CO_2 y H_2O , que son productos finales de la combustión, también se forman productos orgánicos parcialmente oxidados como consecuencia de la combustión ineficiente de los motores. Otro tipo de contaminante que se genera a partir de la combustión de combustibles fósiles son las partículas sólidas, constituidas de material carbonaceo. Otros residuos que el hombre vierte en la naturaleza son los compuestos metálicos, que suelen estar presentes en forma de partículas sólidas. Todos estos compuestos generados por la actividad habitual del hombre generan 800 mil toneladas de compuestos cada año [5].

Una vez vertidos los residuos, estos permanecen suspendidos en la parte baja de la atmósfera. Como se observa en la figura 2.1, los contaminantes emitidos, son mezclados por las masas de aire y transportados a distancias lejanas del centro de emisión. El contaminante puede retornar a la superficie terrestre, ya sea por sedimentación si se encuentra en fase sólida, o bien transportado por el agua de lluvia. Si el tiempo de residencia del residuo es considerable, este puede generar reacciones y transformarse en otras sustancias contaminantes [5].

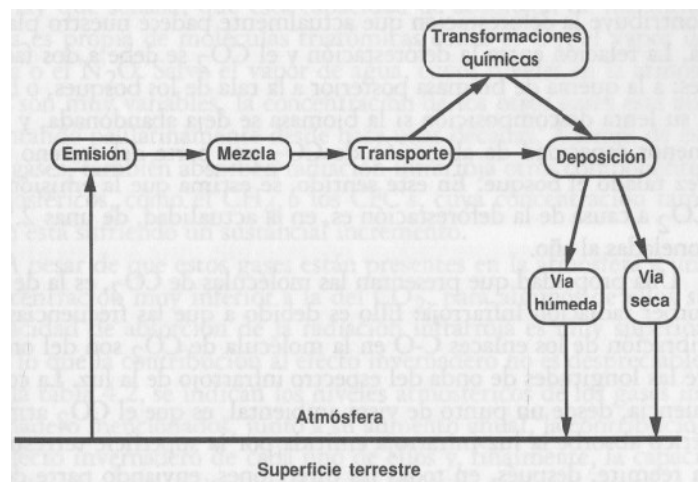


Figura 2.1: Ciclo de los contaminantes atmosféricos [5].

2.2.1.3. Material Particulado

El material particulado (PM), se define como un conjunto de partículas sólidas y/o líquidas de un amplio rango de de tamaños, compuesto de sustancias orgánicas e inorgánicas presentes en suspensión de la atmósfera. El material particulado forma parte de la contaminación del aire. Su composición es muy variada y podemos encontrar entre sus principales componentes sulfatos, nitratos, amoniaco, cloruro de sodio, carbón, polvo de minerales, cenizas metálicas y agua. Dichas partículas además producen reacciones químicas en el aire. Las partículas en la atmósfera se originan de una gran variedad de fuentes y poseen un amplio rango de propiedades morfológicas, físicas, químicas y termodinámicas [31].

El material particulado suspendido en la atmósfera puede ser originado por una gran variedad de fuentes de origen natural o antropogénico. Respecto a los mecanismos de formación, las partículas pueden ser emitidas a la atmósfera o generadas por medio de reacciones químicas, dichas reacciones consisten en la interacción entre gases precursores de la atmósfera para formar una nueva partícula por condensación. De

acuerdo a su tamaño se clasifican en: nanopartículas (<0.1um), ultrafinas (<1um), finas (<2.5um), gruesas (<10um) y sedimentables (>10um). En el cuadro 2.2 se puede observar una comparación entre partículas finas y gruesas [2].

Características	Partículas finas	Partículas gruesas	
Formadas a partir de:	Gases	Sólidos grandes y gotas pequeñas	
Formadas por:	Reacción química	Disrupción mecánica (trituración y abrasión de superficies)	
	Nucleación		
	Condensación	Evaporación de atomizadores	
	Coagulación	Suspensión de polvo	
	Evaporación de niebla		
	Gotas de nubes con gases disueltos		
Composición	Sulfato	Polvos suspendidos	
	Nitrato	Ceniza de carbón	
	Amonio	ceniza de petróleo	
	Ion	óxidos metálicos	
	Hidrógeno	Sal marina	
	carbón elemental	Polen	
	Compuestos orgánicos	Esporas de moho	
	Compuestos metálicos	Fragmentos de animales	
	Agua	Restos de llantas gastadas	
Solubilidad	Muy soluble	Muy insoluble	
	Higroscópico	no higroscópico	
	Delicuescente		
Fuentes	Combustión de carbón	Resuspensión de polvo industrial	
	Hidrocarburos	Suspensión de tierra removida	
	Madera		
	Productos de NOx y SO2	Fuentes biológicas	
	Compuestos orgánicos	Construcción	
	Procesos de altas temperaturas		Demolición
			Combustión de carbón e hidrocarburos
			Brisa marina
Tiempo de Vida	Días a semanas	Minutos a horas	
Distancia de viaje	ciento a miles de kilómetros	Hasta 10 kilómetros	

Tabla 2.2: Comparación entre partículas finas y gruesas [2]

El efecto del material particulado sobre la salud depende en gran medida del tamaño y la concentración, se diferencian de acuerdo a la fuente de emisión y se caracterizan por sus propiedades físicas y químicas. Las propiedades físicas tienen efecto sobre el transporte y depósito sobre el sistema respiratorio, mientras la composición

química tiene impacto directo sobre la salud [2].

El material particulado es considerado como uno de los contaminantes más importantes del aire debido a los efectos que causa sobre la salud del ser humano. Entre los principales problemas destacan: irritación de ojos y nariz, síntomas respiratorios agudos, incremento en los casos de cáncer pulmonar, muertes prematuras, enfermedades cardiovasculares, agravamiento del asma, acumulación de las partículas en los pulmones lo que causa silicosis y asbestosis, etc [2].

2.2.1.4. Transporte de contaminantes ambientales

El transporte de los contaminantes ambientales es un aspecto clave en la determinación de su impacto. En la figura 2.2 se muestra las principales causas del transporte y destino químico. Las sustancias consideradas como contaminantes casi siempre se originan en la antroposfera, estas sustancias pueden pasar al aire, tierra, agua, a los sedimentos y a la biota (plantas y animales). El lugar donde las sustancias van a parar depende de sus propiedades y de las condiciones del medio ambiente. El transporte y destino de las sustancias está controlado por las condiciones físicas y su reactividad, que incluye reacciones químicas y bioquímicas o interacciones físicas. El transporte está en función de tres componentes principales: la atmósfera, las aguas superficiales y la superficie terrestre y el subsuelo [5].

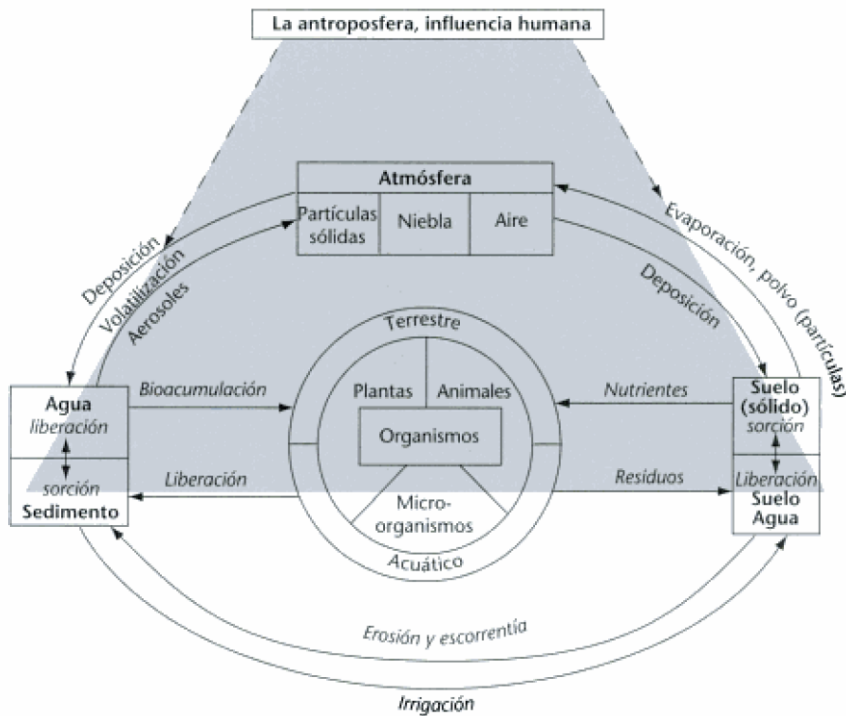


Figura 2.2: Intercambio de contaminantes liberados por la antroposfera [5]

2.2.1.5. Contaminación urbana

La mayor parte de los residuos volátiles que genera el hombre no llega a ascender mas de unas cuantas centenas de metros. El aire en esta zona de la atmósfera está en contacto con la superficie y su movimiento está en función de la rugosidad de esta. Eso da lugar a que se produzcan turbulencias y en consecuencia se produzca una mezcla de los componentes atmosféricos. Es por esta razón que la capa más baja de la atmósfera se conoce con el nombre de capa de mezcla [5].

El principio debido al movimiento constante del aire, los residuos se desplazan mientras se encuentren en la capa de mezcla, en consecuencia, los residuos se diluyen favoreciendo su asimilación por parte de la atmósfera. Solo en caso de que el aporte de residuos sea elevada y constante, y si además las condiciones climáticas lo favorecen, los contaminantes pueden permanecer en una determinada zona en un periodo de tiempo largo, y entonces sus efectos de dejan de notar [32].

Estas son las condiciones que se dan en las ciudades densamente pobladas, así como centros industriales de gran actividad. Debido al clima y a la gran cantidad de residuos que se vierten al aire, propician el desarrollo de intensos episodios de contaminación, cuyos efectos son negativos para la población que habita dichos

lugares [5].

El clima urbano puede llegar a ser muy diferente al de sus alrededores. En primer lugar, hay un relieve diferente formado por los edificios de gran altura, que provoca una mayor resistencia al transporte de los contaminantes. En segundo lugar, el balance energético se encuentra muy perturbado por la urbanización del suelo. Los materiales de construcción absorben la radiación solar, dando lugar a que la temperatura del suelo se incremente durante el día [5].

Los afluentes que circulan por la ciudad lo hacen de forma subterránea a través de la red de alcantarillado, esto hace que se reduzca el tiempo de contacto entre el agua y el aire afectando el intercambio de calor entre estos dos elementos. Este efecto también contribuye al calentamiento de las ciudades [5].

La actividad industrial, el tráfico, las calefacciones domésticas, son aporte de calor extra. Finalmente, la presencia misma de los contaminantes en la atmósfera urbana puede dar lugar al incremento de la temperatura. Las partículas sólidas en suspensión, que se generan por la combustión de carburantes fósiles sólidos y líquidos, actúan absorbiendo la radiación infrarroja que emite el suelo al enfriarse. Ello produce calentamiento de las masas de aire que se encuentran en contacto directo con la superficie urbana [5].

Las partículas que se encuentran en suspensión en la atmósfera urbana, junto con otros contaminantes, absorben la radiación producida por el suelo dando lugar a una inversión térmica, es decir, es una capa de la atmósfera urbana más caliente que las capas inferiores. Una consecuencia de este fenómeno es la posibilidad que se produzca la mezcla vertical de las masas de aire, dificultando que la contaminación urbana se disperse. Además, las masas de aire producidas por la inversión térmica junto con la incidencia de la luz solar, favorece al desarrollo de múltiples reacciones químicas [5].

El estancamiento de las masas de aire hace que se vayan incrementando los niveles de residuos en la atmósfera. La presencia elevada de partículas sólidas, cataliza la nucleación y favorece la formación de niebla sobretodo si la humedad del ambiente es elevada. Esto da lugar a la formación del denominado smog que significa niebla. Además, la presencia de óxidos de azufre y de nitrógeno procedentes de la combustión de carburantes fósiles en motores de explosión, calefacciones y la industria, favorece al desarrollo de reacciones químicas dando lugar a otros contaminantes secundarios como se puede observar en la figura 2.2 [5].

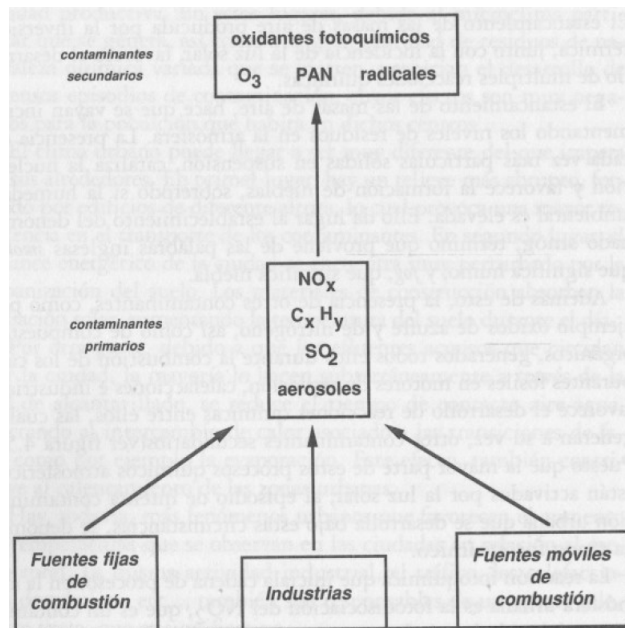


Figura 2.3: Generación de contaminantes primarios y secundarios en la atmósfera urbana [5].

La reacción química que se produce se denomina fotodisociación del NO_2 , que es un contaminante, generado a causa de la fijación del nitrógeno atmosférico en los procesos de combustión a alta temperatura. El átomo de oxígeno es muy reactivo y se combina con oxígeno atmosférico para dar ozono. En la atmósfera también existen una gran variedad de compuestos químicos, emitidos por los tubos de escape de combustión ineficiente. Estos compuestos reaccionan con agentes oxidantes como radicales OH u oxígeno atómico, el resultado final es la acumulación de ozono en la atmósfera urbana [5].

También se generan otras especies químicas de alta toxicidad, una de estas es el nitrato de peroxiacetilo, PAN, un compuesto estable y de elevado poder oxidante. El PAN es el responsable de la irritación de los ojos que ocurre en situaciones de smog intenso. Otro fenómeno químico que se produce, es la reacción del ozono con el dióxido de nitrógeno NO_2 para dar radicales NO_3 muy reactivos. Se combinan con más NO_2 para dar una especie de N_2O_5 que reacciona con el vapor de agua dando como resultado ácido nítrico responsable de la acidez de las nieblas matutinas [5].

Otro de los elementos que constituye el smog es el formaldehído, HCHO, que es un gas irritante de las membranas mucosas. Además, en el smog se encuentran una gran variedad de partículas sólidas de suspensión que generan disminución drástica de la visibilidad, ocasionando accidentes [5].

2.2.2. Computación en la Nube

El Cloud Computing, la computación en la nube o la nube informática, es el conjunto de servidores desplegados en centros de datos a lo largo de todo el mundo, almacenan millones de aplicaciones y enormes cantidades de datos. Según la IEEE, la computación en la nube (cloud computing), se define como un modelo tecnológico ubicuo, que permite el acceso bajo demanda a recursos compartidos de computación a través de internet. La computación en la nube son servidores en internet encargados de atender peticiones en cualquier momento. Sirven a sus usuarios desde varios proveedores de alojamiento repartidos por todo el mundo. Estas medidas ayudan en la reducción de costos y mayor disponibilidad de los recursos [32].

2.2.2.1. Historia

Debido a las elevadas necesidades de cómputo, se han realizando investigaciones en la ejecución de procesos en múltiples computadores. Esta tendencia fue impulsada por la utilización de sistemas abiertos, interoperables y protocolos de comunicación estándar. El primer paso de esta evolución fue propiciado por los sistemas operativos tipo Unix, que permitieron la configuración de clusters, es decir, la agrupación de ordenadores con componentes de hardware comunes que se comportan como un solo elemento [6].

Tras varias décadas de investigación y desarrollo de estas tecnologías, la irrupción del sistema operativo Linux y sus estándares abiertos permitió implementar clusters basados en arquitectura estándar de los PC, consiguiendo instalaciones de cálculo de alto rendimiento y reduciendo los costos de computación. Esto se logró durante la década de los 90. Los clusters funcionaban en centros de investigación y universidades, comenzando a ofrecer servicios a terceros por medio de protocolos de comunicación estándar, constituyendo la arquitectura grid orientada al procesamiento en paralelo y almacenamiento de gran cantidad de datos. Debido a la complejidad para utilizar la infraestructura, la dificultad de utilizar diferentes grids y los problemas de portabilidad entre ellas hicieron que nunca se popularizaran fuera del campo investigativo y educativo [6].

En la primera década del siglo XXI comenzaron a popularizarse las tecnologías de virtualización que permitían implementar máquinas virtuales que desacoplan en hardware del software y permiten replicar el entorno del usuario sin tener que instalar y configurar todo el software que requiere cada aplicación. Esta nueva

arquitectura permitía distribuir la carga de trabajo de forma sencilla, eliminando los problemas que presentaba la arquitectura grid y abriendo una nueva puerta al cálculo distribuido llamado cloud computing [6].

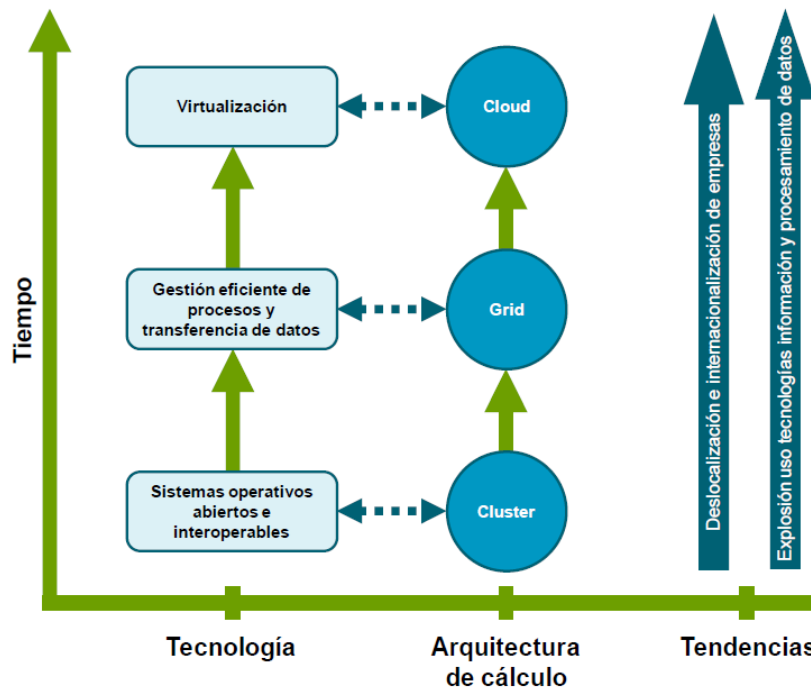


Figura 2.4: Evolución del cloud computing a lo largo del tiempo [6].

2.2.2.2. Características

En la figura 2.5 se puede observar las características asociadas al cloud computing:

- Pago por uso: El costo que debe abonar el cliente varía en función del uso que se realiza del servicio de cloud contratado.
- Abstracción: Capacidad de aislar los recursos informáticos contratados al proveedor de servicios cloud de los equipos informáticos del cliente.
- Agilidad en la escalabilidad: Capacidad de aumentar o disminuir las funcionalidades ofrecidas al cliente en función de sus necesidades. De la misma manera se modifica el coste del servicio.
- Multiusuario: Capacidad que permite a varios usuarios compartir los medios y recursos informáticos, permitiendo la optimización de su uso.
- Autoservicio bajo demanda: Permite al usuario acceder de manera flexible a

las capacidades de computación de forma automática a medida que las vaya requiriendo.

- Acceso sin restricciones: Posibilidad ofrecida a los usuarios para acceder a los servicios en cualquier lugar, en cualquier momento y con cualquier dispositivo con conexión a internet.

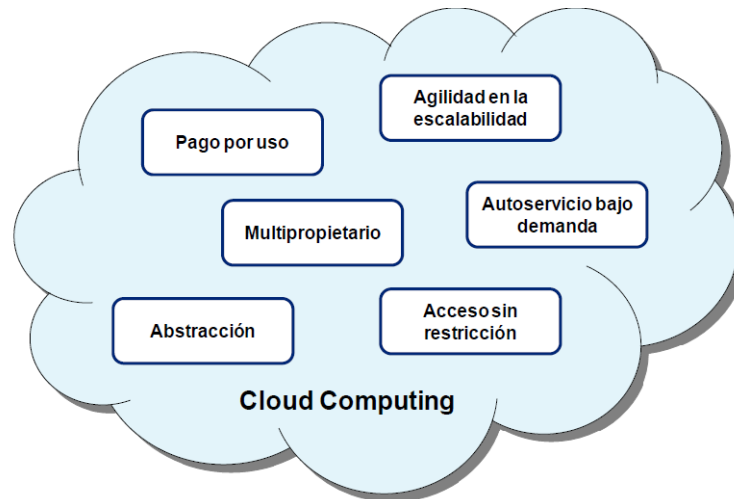


Figura 2.5: Características del cloud computing [6]

2.2.2.3. Arquitectura

El cloud computing se categoriza como un conjunto de modelos de servicios cada uno de ellos orientados a un determinado usuario con necesidades específicas. Así mismo se puede considerar como un conjunto de capas sobre las cuales desplegar aplicaciones. Estas capas son infraestructura (IaaS), plataforma (PaaS), Software (SaaS) con una gran capacidad de virtualización y protocolos de comunicación. En la figura 2.6 se puede observar de manera simplificada la estructura de cómputo en la nube [7].

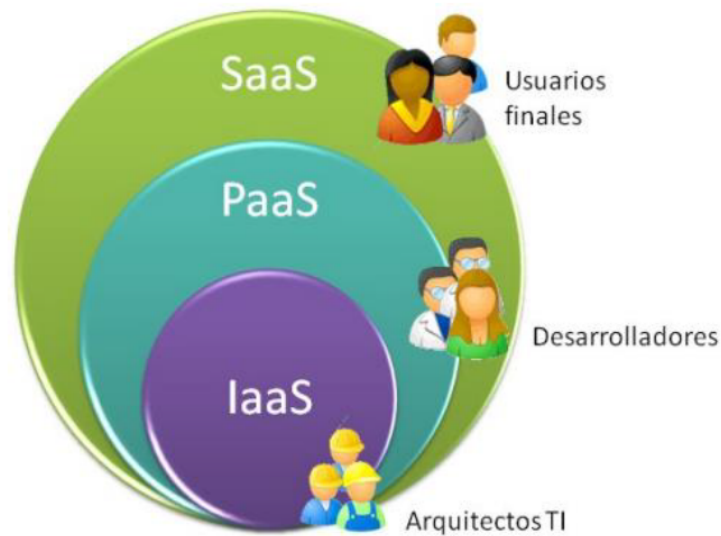


Figura 2.6: Modelo del cloud computing [7].

La capa de infraestructura es el corazón de la nube, es puro almacenamiento y capacidad de cómputo. Con la técnica de virtualización se realizan pequeñas máquinas con sistemas operativos y capacidad de cómputo o almacenamiento que se entregan como el agua o electricidad. La capa de plataforma es un sistema que puede ser programado y personalizado por los desarrolladores para adaptarse a un sin número de necesidades, si se puede programar se trata de una plataforma. La capa de software está formada por aplicaciones y servicios que pueden ser utilizadas por múltiples clientes de forma simultánea sin la necesidad de instalar ni mantener el software [7].

2.2.2.4. Virtualización

La idea de virtualización es poder crear servidores virtuales, almacenamiento virtual, redes virtuales, aplicaciones virtuales, etc. Mediante la virtualización se maximiza el aprovechamiento de los recursos de los servidores, esto implica hacer que un recurso físico, un sistema operativo o un dispositivo de almacenamiento aparezca como si fueran varios recursos lógicos a la vez. O varios recursos físicos como servidores o dispositivos de almacenamiento aparezcan como un único recurso lógico [7].

La virtualización del sistema operativo es el uso de software para permitir que un mismo sistema maneje varias imágenes de varios sistemas operativos a la misma vez. Esta tecnología permite la separación del hardware y del software, permitiendo que varios sistemas operativos se ejecuten de forma simultánea en una sola computadora.

Es un programa que se instala en un sistema operativo (anfitrión) que permite instalar y ejecutar otro sistema operativo como si fuera otro computador. Las tecnologías de virtualización soportan la migración en directo, permitiendo que un sistema operativo y sus aplicaciones se muevan a un nuevo servidor para balancear carga [7].

Características:

- **Particionamiento:** Permite ejecutar múltiples aplicaciones y sistemas operativos en un mismo sistema físico. Los recursos computacionales se distribuye entre las máquinas virtuales de manera controlada.
- **Aislamiento:** Las máquinas virtuales están completamente aislados entre si y del host. Si existen fallos en una máquina virtual, las demás no se ven afectadas. Los datos no se filtran a través de las máquinas virtuales y las aplicaciones solo se pueden comunicar a través de conexiones de red.
- **Encapsulación:** El entorno completo del servidor virtual se guarda en un solo archivo, fácil de mover, copiar y respaldar.
- **Independencia de hardware:** Las máquinas virtuales son totalmente independientes del hardware físico.

2.2.2.5. Modelos de Cloud Computing

- *Infraestructura como Servicio (IaaS).* El modelo de arquitectura como servicio consiste en poner a disposición del cliente el uso de la infraestructura informática (capacidad de computación, espacio en disco y bases de datos). El proveedor de este servicio entrega la infraestructura y se trabaja con la capa de virtualización, donde el cliente debe calcular los recursos que necesita (memoria ram, almacenamiento, maquinas virtuales, procesamiento, diseño de la red, balanceo de carga, enlaces de datos). Los usuarios optan por este servicio en busca de un ahorro de inversión. Los costos del servicio se calculan en base a los recursos consumidos por el cliente, basándose así en un modelo de pago por uso. Este modelo se enfoca en el área de las tecnologías de la información (TI) y administradores de sistemas (todavía se interactúa con la capa de hardware) [33].
- *Software como Servicio (SaaS).* El modelo de software como servicio consiste en la entrega de aplicaciones como servicio donde el proveedor ofrece licencias

de su aplicación a los clientes para su uso bajo demanda. El cliente se despreocupa de la parte de hardware. Los proveedores de los servicios PaaS pueden tener instalado en sus servidores la aplicación a la cual accederán los usuarios por medio de un navegador web u otra herramienta. También se permite la descarga de la aplicación en los sistemas del contratante del servicio, en el cual se desactivará la aplicación una vez terminado el contrato de licencia de uso [6].

- *Plataforma como Servicio (PaaS)*. Este tipo de modelo consiste en la entrega de un conjunto de plataformas informáticas orientadas al desarrollo, testeo, despliegue, hosting y mantenimiento de los sistemas operativos y aplicaciones de clientes. El modelo de plataforma como servicio es más económico que el modelo de infraestructura como servicio. Una de las características de este modelo es que facilita el despliegue de aplicaciones del cliente, sin considerar los costos de la compra y gestión de hardware. Otra de las características es que ofrece a través de redes de servicio IP todos los requisitos necesarios para entregar y crear servicios y aplicaciones web [33, 6].

2.2.2.6. Clasificación del Cloud Computing

- *Nube Pública*. El proveedor del servicio ha realizado toda la inversión para el montaje de la infraestructura cloud, la función del usuario es contratar el servicio que requiere ya sea en modelo de infraestructura como servicio, plataforma como servicio o software como servicio. El proveedor entrega un centro de datos virtual con todas las herramientas que tienen un alto nivel de disponibilidad. está enfocado para empresas que requieren poner un servicio rápidamente en el mercado o carece de infraestructura propia. Una de las características de contratar este servicio es que la tecnología del proveedor es buena, la inversión inicial no es alta, el cliente no tiene la obligación de quedarse con el proveedor, no hay que preocuparse por el mantenimiento y actualizaciones. Una de las desventajas es que no se tiene control total de la plataforma [33].
- *Nube Privada*. En este tipo de cloud el cliente realiza la inversión para disponer de su propia infraestructura de Cloud Computing y tiene las mismas ventajas que una nube pública. Se debe tener personal capacitado para dar soporte a la infraestructura instalada. Entre las principales ventajas es que se tiene mayor seguridad ya que los datos se encuentran dentro de la empresa, pero existe

un gasto de mantenimiento de la infraestructura, tener personal capacitado, el centro de datos debe tener toda la infraestructura eléctrica, aire acondicionado, sistema de control de incendios, el hardware y software está a cargo de la empresa [33].

- *Nube híbrida.* Este tipo de infraestructura se caracteriza por tener dos formas de cloud (privada y pública) coexistiendo entre sí para facilitar el acceso a aplicaciones e información, interconectados mediante tecnología estandarizada o propietaria, tecnología que permite la portabilidad de datos y aplicaciones. Una entidad que opte por este tipo de infraestructura se beneficiará de las ventajas asociadas tanto a la nube pública como a la nube privada. Una de las características es que presenta mayor flexibilidad en la prestación de servicios, rápida prestación de servicios, mayor control de los datos y funciones básicas de la entidad, posibilidad de elección de la infraestructura escalable o flexible y control interno de los servicios cloud desde la propia entidad [6].

2.2.2.7. Ventajas del Cloud Computing

Los servicios de cloud computing presentan una serie de ventajas para las empresas en el ámbito económico, tecnología, seguridad, disponibilidad y movilidad:

- **Económico:** La empresa no tiene que invertir en los servidores, equipos de seguridad perimetral, redes, almacenamiento, energía eléctrica, aire acondicionado, cableado estructurado, respaldos de información, mantenimientos preventivos y correctivos de la infraestructura.
- **Escalabilidad:** Se puede provisionar de recursos adicionales como memoria ram, almacenamiento, CPU, máquinas virtuales para incrementar los recursos en caso de ser necesario o por mal dimensionamiento.
- **Alta disponibilidad:** Los centros de datos de los proveedores brindan los más altos niveles de disponibilidad, disponen de redundancia en la infraestructura y son tolerantes a fallos.
- **Personal altamente calificado:** El proveedor de servicios cloud cuenta con un plantel de personal preparado para solucionar cualquier inconveniente que se presente.
- **Monitoreo del consumo de la infraestructura:** El monitoreo de los servidores, enlaces de datos y equipos de seguridad perimetral se lo realiza de forma

permanente.

2.2.2.8. Riesgos del Cloud Computing

Al lanzar una infraestructura en la nube, el cliente cede el control y administración de una serie de cuestiones a la empresa que provee el servicio, provocando un control restringido de los datos e imposibilitando encontrar sus propios archivos fácilmente dentro de la nube. A menudo los datos deben ser eliminados ya sea por motivos legales o luego de un cierre de oferta, pudiendo ocurrir que los archivos queden parcialmente eliminados de manera insuficiente. En la nube es difícil separar ciertos datos para ciertos clientes, es decir, algunos clientes podrán tener acceso a datos que no debían ser leídos por ellos [6].

El el cloud computing se debe tener en cuenta el cumplimiento de todas las exigencias y demandas legales, Además de revisar el orden jurídico de cada país, ya que muchas veces existen diferencias por país. También existe el peligro de que se irrumpa la privacidad, ya que cualquiera puede acceder a esta nube sin importar el país, el servidor o el centro de cómputo [6].

2.2.3. Historia de las Redes Neuronales

En 1936, Alan Turing fue uno de los primeros en estudiar el cerebro humano como un sistema de computación. Sin embargo, los primeros en concebir conceptos de computación neuronal fueron el neurofisiólogo Warren McCulloch y el matemático Walter Pitts. En 1943, lanzaron una teoría acerca de la forma de trabajar de las neuronas para posteriormente modelar una red neuronal simple basados en circuitos eléctricos. En 1949, Donal Hebb fue unos de los primeros en explicar el proceso de aprendizaje desde un punto de vista psicológico. Hasta el día de hoy este es el fundamento de la mayoría de las funciones de aprendizaje usados en los modelos de redes neuronales artificiales. Su modelo de aprendizaje se basa cuando ciertos cambios activan una determinada neurona. También se propuso a encontrar semejanzas entre la actividad nerviosa y el aprendizaje. Los trabajos de Hebb son las bases de la teoría de Redes Neuronales Artificiales [34].

En 1950, Karl Lashley encontró que la información era almacenada en el cerebro de forma distribuida. En 1957, Frank Rosenblatt comenzó a desarrollar el perceptron, que es la red neuronal más antigua y que es utilizada hoy en

día para el reconocimiento de patrones. En 1959, presentó el libro “Principios de Neurodinámica”, en el cual confirmó que bajo ciertas condiciones, el perceptron convergía hacia un estado finito. En 1960, Bernard Widroff Hoff desarrolló el modelo Adaline (Adaptative Linear Elements). Este fue el primer modelo de red neuronal aplicado a problemas reales como el diseño de filtros adaptativos para eliminar ecos en las líneas telefónicas. En 1969, Matvin Minsky Papert probó matemáticamente que el perceptrón no era capaz de resolver problemas relativamente fáciles, como el aprendizaje de una función no lineal. Esto casi lleva a la muerte de la teoría de redes neuronales artificiales [34].

En 1974, Paul Werbos desarrolló una idea básica del algoritmo de aprendizaje de propagación hacia atrás, que quedaría totalmente aclarado en 1985. En 1977, Stephen Grossberg desarrolla la Teoría de Resonancia Adaptada, que es una arquitectura de red que simula otras habilidades del cerebro como la memoria a corto y largo plazo. En 1985, John Hopfield escribió el libro denominado “Computación neuronal de decisiones en problemas de optimización”, que provocó el renacimiento de las redes neuronales. A partir de 1986 se continúa con el desarrollo e investigación de las redes neuronales. En la actualidad son numerosos los trabajos que se realizan y se publican dentro de este campo [34].

2.2.3.1. Neurona Biológica

El cerebro humano contiene, aproximadamente, mas de cien mil millones de neuronas y sinapsis dentro del sistema nervioso. Cada neurona posee más de 1000 sinapsis en la entrada y hacia la salida. El objetivo principal de las redes neuronales biológicas es verificar las hipótesis que conciernen a los sistemas nerviosos. Las neuronas y sus conexiones entre ellas es lo que constituye la clave para el procesamiento de la información [8].

Una neurona biológica es básicamente una célula especializada en procesar información. Está compuesta por el cuerpo de la célula y dos ramificaciones (axón y las dendritas). Las neuronas reciben las señales de otras neuronas a través de las dendritas y transmite señales procesadas por el cuerpo de la célula a través del axón. Una neurona está compuesta por un cuerpo celular mas o menos esférico de 5 a 10 micras de diámetro, del cual sobresalen una rama principal (axón) y varias ramificaciones mas cortas que corresponden a las dendritas, como se puede observar en la figura 2.2. Las dendritas y el cuerpo celular reciben las señales que provienen de otras neuronas, el cuerpo celular las combina e integra y emite señales de salida.

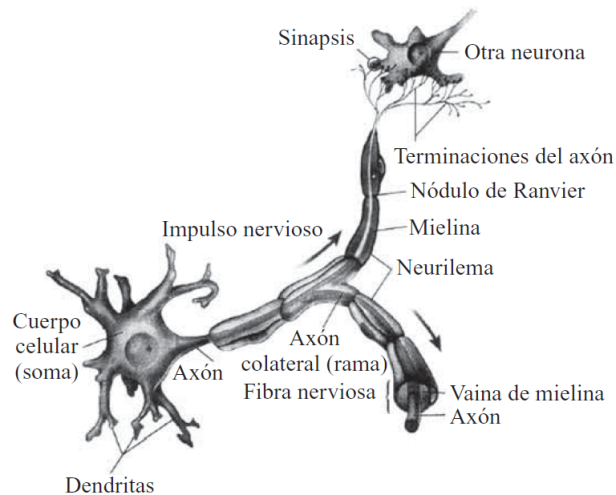


Figura 2.7: Elementos de una Red Neuronal Biológica [8].

El axón transmite dichas señales a los terminales axónicos, los cuales distribuyen la información. Se calcula que en el cerebro humano existen aproximadamente 10^{15} conexiones [8].

Las señales que usa una red neuronal biológica es de dos tipos: eléctricas y químicas. La señal generada por la neurona y que es transmitida a lo largo del axón, es un impulso eléctrico. La señal que se transmite entre los terminales axónicos de una neurona y las dendritas de otra neurona es de origen químico. Las neuronas pueden ser muy variadas en morfología y tamaño. Pueden ser estrelladas, fusiformes, piriformes. También pueden ser muy variadas en cuanto a las prolongaciones. La unión entre dos neuronas se denomina sinapsis. En el tipo de sinapsis mas común no existe contacto físico entre las neuronas, sino que están separadas por un pequeño vacío de 0.2 micras aproximadamente [8].

La forma de comunicación entre dos neuronas es de tipo químico, donde la neurona presináptica libera unas sustancias químicas denominadas neurotransmisores (glutamato o adrenalina), que atraviesan el vacío sináptico. Si la neurona postsináptica es sensible a los neurotransmisores liberados, los fijará y permitirá el paso de determinados iones a través de la membrana. Las corrientes iónicas que se crean provocan pequeños potenciales excitadores (positivos) o inhibidores (negativos), que se integrarán al soma tanto espacial como temporalmente; este es el origen de la existencia de sinapsis excitatoria y sinapsis inhibitorias [8].

Si se produce un número suficiente de excitaciones, donde la suma de los potenciales positivos generados elevan el potencial por encima de los -45mV (umbral de disparo), en este caso, se abre bruscamente los canales de sodio, de modo que los iones entran

masivamente al interior, provocando la despolarización de la neurona, y pasando de un potencial de reposo de -60mV a unos $+50\text{mV}$. A continuación, la neurona vuelve a la situación original de reposo y este proceso genera un potencial de acción que se propaga a lo largo del axón. Tras haber sido provocado el potencia de acción, la neurona sufre un periodo refractario durante el cual no puede generarse de nuevo [8].

2.2.3.2. Ventajas de una Red Neuronal Artificial

El funcionamiento que las Redes Neuronales Artificiales se asemeja al cerebro por lo que presenta una serie de características como por ejemplo aprender de la experiencia, generalizar de casos anteriores nuevos casos, abstraer características esenciales a partir de entradas que presentan información irrelevante, etc. Todo esto hace que presenten ventajas tales como:

- Aprendizaje adaptativo: Capacidad de aprender a realizar tareas basadas en un entrenamiento previo.
- Auto-organización: Toda red neuronal puede crear su propia organización basados en un entrenamiento inicial.
- Tolerancia a fallos: La destrucción parcial de una red neuronal artificial reduce a una degradación de su estructura, sin embargo algunas capacidades de la red pueden seguir funcionando.
- Operación en tiempo real: Los cálculos neuronales pueden ejecutarse en paralelo, para ello se fabrican dispositivos especiales para obtener esta capacidad.
- Fácil inserción dentro de una tecnología existente: Se puede tener chips especializados en redes neuronales capaces de integrarse dentro de los sistemas existentes.

2.2.3.3. Aplicaciones de las Redes Neuronales Artificiales

Algunas áreas donde se aplican las redes neuronales artificiales son:

- Automóviles: Sistemas de piloto automático, detección de fallas por reconocimiento externo de vibraciones.

- Bancos: Lectura de cheques y otros documentos, evaluación de aplicaciones de crédito.
- Electrónica: Predicción de secuencia de código, distribución de elementos en circuitos integrados, control de procesos, análisis de fallos, visión artificial, reconocimiento de voz.
- Finanzas: Tasación real de bienes, asesoría de prestamos, previsión de precios, seguimiento de hipotecas, evaluación de riesgo en créditos, interpretación y reconocimiento de firmas.
- Manufactura: Control de productos y procesos, análisis y diseño de productos, diagnóstico de fallas en procesos y maquinaria, identificación de partículas en tiempo real, identificación de calidad por medio de sistemas visuales, análisis de mantenimiento de maquinaria.
- Medicina: Análisis de células portadoras de cáncer, análisis de electroencefalograma y electrocardiograma, reconocimiento de infartos, diseño de prótesis, optimización de tiempo de trasplante, reducción de gastos hospitalarios.
- Robótica: Control dinámico de trayectorias, robots elevadores, controladores, sistemas ópticos.
- Seguridad: Códigos de seguridad adaptativos, criptografía, reconocimiento de huellas digitales.
- Telecomunicaciones: Comprensión de datos e imágenes, traslación en tiempo real del lenguaje hablado.
- Transporte: Diagnóstico de frenos, sistemas de ruteo y seguimiento de flotas.
- Voz: Reconocimiento de voz, comprensión de voz, transformación de texto a voz.

2.2.3.4. Red Neuronal Artificial

Las redes neuronales artificiales son sistemas de mapeo no lineales cuyos principios se basan en los sistemas nerviosos de los seres humanos y animales. Consta de un número grande de procesadores simples ligados por conexiones con pesos. Las unidades de procesamiento son las neuronas, cada una recibe entradas de otros nodos y genera una salida simple escalar que depende de la información local, registrando

internamente o llega a través de las conexiones con pesos. Una red neuronal se caracteriza por los siguientes elementos:

- Posee un conjunto de unidades de procesamiento conocido como neuronas.
- El estado de activación de cada unidad es equivalente a la salida de la unidad.
- Existe conexiones entre las unidades determinadas por un peso que determina el efecto de la señal de entrada en la unidad.
- Una regla de propagación que determina que determina la entrada efectiva de una unidad a partir de las entradas externas.
- Una función de activación que actualiza el nuevo nivel de activación basándose en la entrada efectiva y la activación anterior.
- Una entrada externa que corresponde a un término determinado como bias para cada unidad.
- Un método para reunir la información correspondiente a la regla de aprendizaje.
- Un ambiente en el que el sistema va a operar.

Muchas de las redes neuronales tienen una respuesta de la forma:

$$y = f\left(\sum_k w_k x_k\right)$$

donde:

x_k : señales de salida de otras neuronas.

w_k : pesos de las conexiones.

f : función no lineal simple.

La función no lineal simple puede ser sigmoideal, tangente hiperbólica, escalón, entre otras. Cada unidad de proceso recibe la entrada de otras unidades y procesa la información para obtener una salida que se propaga hacia otras unidades. Una red neuronal puede tener una estructura arbitraria, pero las capas que contienen esta estructura están definidas de acuerdo con su ubicación en la topología de la red. Las entradas externas son aplicadas en la primera capa, las salidas se considera la última capa. Las capas internas que no se consideran como entradas o salidas se denominan capas ocultas [8].

La entrada total u de una unidad k , es la suma de todos los pesos de las entradas conectadas mas un bias θ :

$$u = \sum_j w_j x_j + \theta$$

Si el peso es positivo se habla de una excitación, y si es negativo se considera una inhibición de la entrada. Se considera a las entradas como una función del tiempo, la expresión queda de la siguiente manera:

$$u(t) = \sum_j w_j(t) x_j(t) + \theta(t)$$

En la figura 2.8 se puede observar un esquema básico de una red neuronal simple, donde w_i son las entradas, w_{j1} son los pesos y y_j es la función de activación.

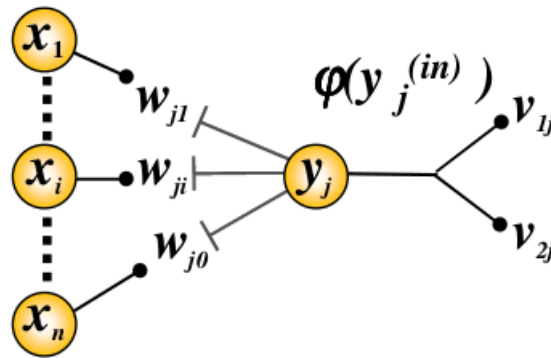


Figura 2.8: Esquema de una red neuronal artificial [9]

2.2.3.5. Funciones de activación

la regla que logra establecer el efecto de las entradas en la activación de la unidad se denomina función de activación F_k .

$$y(t + 1) = F_k(y(t), u(t))$$

$$y(t + 1) = F_k \left(\sum_j w_j(t) x_j(t) + \theta(t) \right)$$

2.2.3.6. Función escalón

La función de activación escalón se asocia a neuronas binarias cuando las sumas de las entradas es mayor o igual al umbral de la neurona, la activación es 1, si es menor la activación es 0 o -1. En gráfico de la izquierda en la figura 2.9 se puede observar

la salida de la neurona cuando se activa es 1 y cuando no, la salida es cero. En el gráfico a la derecha de la figura 2.9 se puede notar que cuando la red neuronal se activa la salida es 1, y cuando no se activa la salida de la neurona es -1 [8].

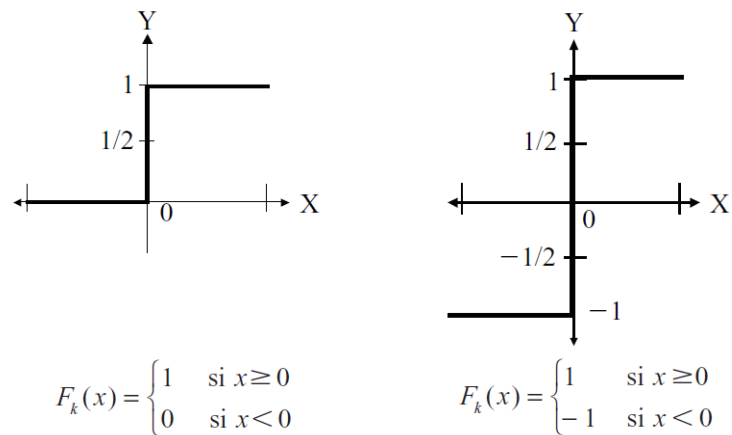


Figura 2.9: Función de activación escalón [8].

2.2.3.7. Función Lineal mixta

En las neuronas con función de activación mixta, si la suma de las señales de entrada es menor que un límite inferior, la función se define con 0 o -1. Si la suma de dichos valores es mayor o igual que el límite superior, entonces la activación es 1. Si la suma de las entradas está comprendida entre ambos límites (superior e inferior), entonces la activación es 1. Si la suma de las entradas está comprendida entre ambos límites (superior e inferior), la función de activación se define como una función lineal de la suma de las entradas como se puede observar en la figura 2.10 [8].

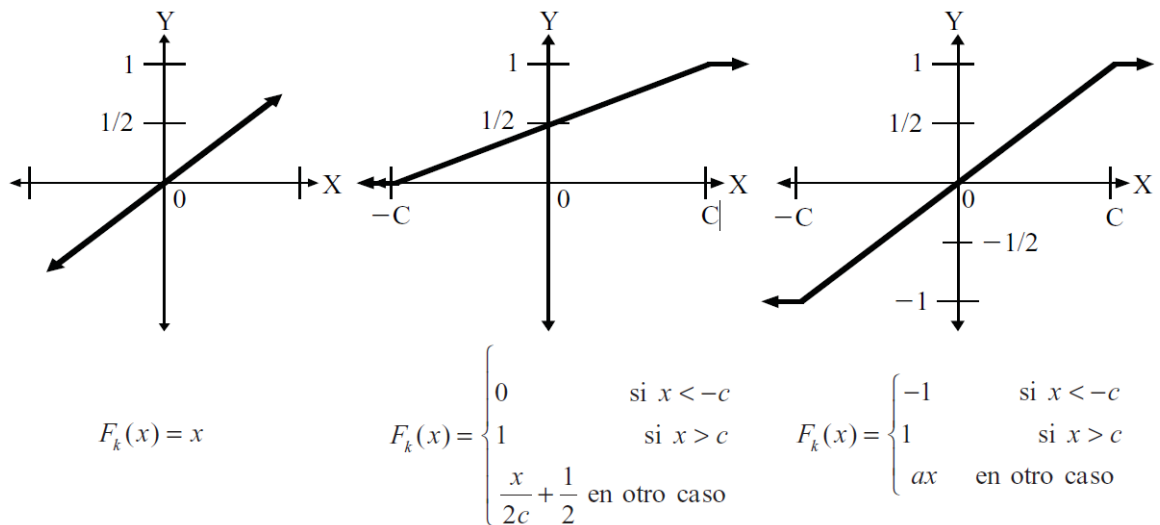


Figura 2.10: Función de activación lineal y mixta [8].

2.2.3.8. Función tangente hiperbólica

Esta función se emplea en casos que se presentan variaciones suaves de valores positivos y negativos de la señal a clasificar, como se puede observar en la figura 2.11. Es una de las funciones más empleadas en entrenamientos supervisados, como el entrenamiento de retropropagación del error. Se debe tener cuidado al momento de utilizar esta función entre los umbrales positivos y negativos antes de la saturación, de otra forma la salida siempre generará valores saturados iguales a 1 y -1 [8].

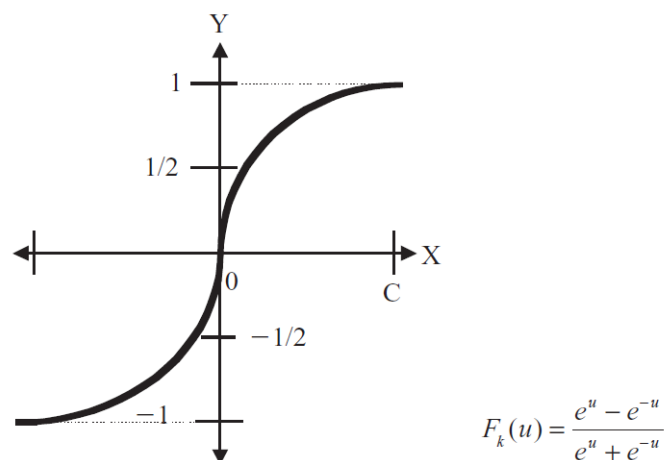


Figura 2.11: Función de activación tangente hiperbólica [8].

2.2.3.9. Función sigmoïdal

Con la función de activación sigmoïdal, el valor dada por la función es cercano a uno de los valores asintóticos. Esto hace que el valor de la salida esté comprendido en la zona alta o baja del sigmoïde, cuando la pendiente es elevada, esta función tiende a la función escalón como se puede observar en la figura 2.12. La importancia de la función sigmoïdal es que su derivada siempre es positiva y cercana a cero para los valores grandes positivos o negativos, además toma su valor máximo cuando $x=0$. Esto hace que se pueda utilizar reglas de aprendizaje definidas para las funciones escalón, con la ventaja de que la derivada está definida en todo el intervalo [8].

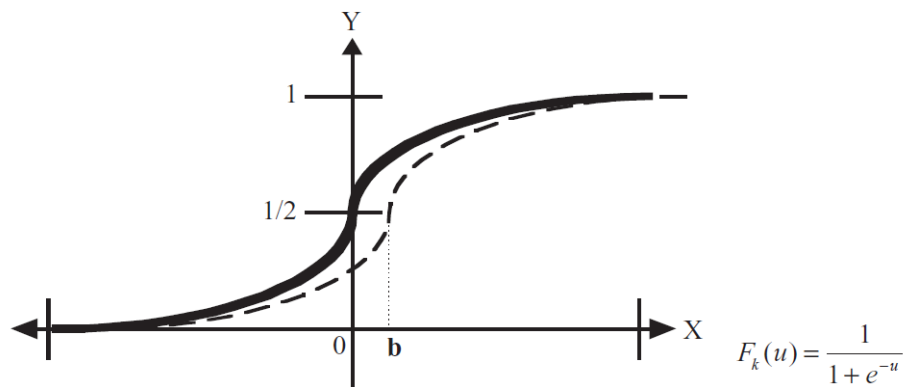


Figura 2.12: Función de activación sigmoïdal [8].

2.2.3.10. Función de Gauss

Los mapeos ocultos algunas veces pueden realizarse con un solo nivel de neuronas mediante el uso de funciones de activación de Gauss en lugar de funciones tipo sigmoïdales como se puede observar en ña figura 2.13.

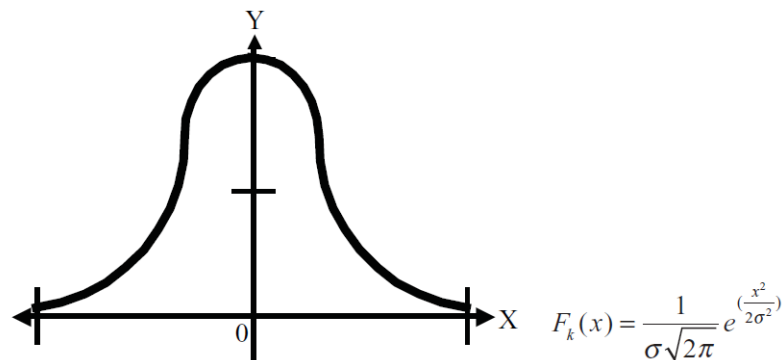


Figura 2.13: Función de Gauss [8].

2.2.3.11. Elementos de una Red Neuronal Artificial

Las redes neuronales artificiales consta de un conjunto de elementos de procesamiento conectados entre si y en los que se envía información a través de conexiones. Un esquema básico de una red neuronal artificial se observa en la figura 2.14 donde se muestra las diferentes capas que tiene la topología. Los elementos básicos de una red neuronal simple son: conjunto de unidades de procesamiento conocido como neuronas, conexiones entre unidades asociadas a cada conexión un peso o valor, y una función de salida o activación para cada unidad de procesamiento. El modelo de neurona y la arquitectura de la red describen como se transforman las entradas en salidas. Los elementos individuales de cálculo que forman la mayoría de los modelos se sistemas neurales artificiales se conoce con el nombre de neuronas artificiales [8].

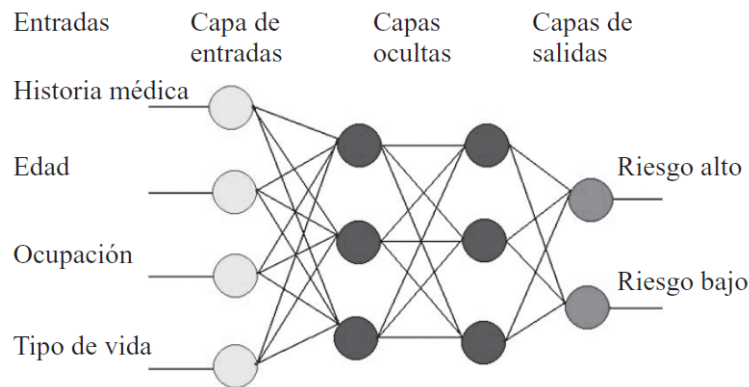


Figura 2.14: Esquema básico de una red neuronal artificial [8].

2.2.3.12. Entrenamiento de las Redes Neuronales

El entrenamiento de una red neuronal artificial es el proceso de configuración para que las entradas produzcan las salidas deseadas a través del fortalecimiento de las conexiones. Una forma de lograr este fin es por medio del establecimiento de pesos conocidos con anterioridad, y otro método implica el uso de técnicas de retroalimentación y patrones de aprendizaje que cambian los pesos hasta encontrar los adecuados [8].

El objetivo del entrenamiento es conseguir que un conjunto de entradas produzcan un conjunto de salidas deseadas. El proceso de entrenamiento consiste en la aplicación de diferentes conjuntos o vectores de entrada para que se ajusten los pesos de las interconexiones. Durante el proceso de entrenamiento los pesos convergen gradualmente hacia los valores que hacen que cada entrada produzca el

vector de salida deseado. Los algoritmos de aprendizaje de las conexiones de las redes neuronales artificiales pueden clasificarse en dos grupos: supervisado y no supervisado [3]:

- **Entrenamiento Supervisado:** Estos algoritmos requieren de un emparejamiento de cada vector de entrada con su respectivo valor de salida. El entrenamiento consiste en presentar un vector a la entrada de la red, calcular la salida de la red, compararla con la salida deseada y el error o diferencia se utiliza para realimentar la red y cambiar los pesos de la red de acuerdo con un algoritmo que minimiza el error. Los vectores del conjunto de entrenamiento se aplica secuencialmente y de forma cíclica. Se calcula el error y el ajuste de los pesos para cada pareja hasta que el conjunto de entrenamiento sea un valor pequeño y aceptable [3].
- **Entrenamiento no Supervisado:** Estos sistemas de entrenamiento han tenido éxito en muchas aplicaciones, sin embargo han tenido muchas críticas debido a que no son muy lógicos. Son modelos de aprendizaje más lógicos en los sistemas biológicos. No requieren de un vector de salidas deseadas y por tanto no se realizan comparaciones entre las salidas reales y salidas esperadas. El conjunto de vectores de entrenamiento consiste unicamente en vectores de entrada. El algoritmo modifica los pesos de la red para que produzcan vectores de salidas consistentes. El proceso extrae las propiedades estadísticas del conjunto de vectores y agrupa las clases en vectores similares [3].

En la tabla 2.3 se hace una descripción de las redes neuronales artificiales usados en los sistemas de computación neuronal.

Nombre	Característica	Tipo
Adaline y Madaline	Técnicas de adaptación para el reconocimiento de patrones	Predicción
Adaptative Resonanc Theory Networks (ART)	Reconocimiento de patrones y modelo del sistema neuronal	Conceptualización
Back-Propagation	Solución a las limitaciones al perceptron	Clasificación
Bi-Directional Associative Memory (BAM)	Inspirada en la red ART	Asociación
The Boltzmann Machine	Similar a la red Hopfield	Asociación
Brain-State-in a Box	Red Asociativa Lineal	Asociación
Cascade-Correlation-Networks	Adición de nuevas capas ocultas en cascada	Asociación
Counter-Propagation	Clasificación adaptativa de patrones	Clasificación
Delta-Bar-Delta (DBD) Networks	Método heurístico para acelerar la convergencia	Clasificación
Digital Neural Network Architecture (DNNA)	Implementación hardware de la función sigmoid	Predicción
Directed Random Search (DRS)	Técnica de valores random de ajuste de pesos	Clasificación
Functional-link Networks (FLN)	Versión mejorada de la red Backpropagation	Clasificación
Hamming Networks	Clasificador de vectores utilizando la distancia de Hamming	Asociación
Hopfield Networks	Conceptos de la red en términos de energía	Optimización
Learning Vector Quantization (LVQ)	Red Clasificadora	Clasificación
Perceptron Networks	Primer modelo de neurona artificial	Predicción
Probabilistic Neural Network (PNN)	Clasificación de patrones utilizando métodos estadísticos	Asociación
Recirculation Networks	Alternativa a la red Backpropagation	Filtrado
Self-Organizing Maps (SOM)	Aprendizaje sin supervisión	Conceptualización
Spatio-Temporal-Pattern Recognition (SPR)	Red clasificadora invariante en el espacio y tiempo	Asociación

Tabla 2.3: Modelos de redes neuronales artificiales mas usados en la actualidad [3]

2.2.3.13. Perceptron

También conocido como mapeo de patrones (pattern-mapping), aprende a supervisar modelos mediante un aprendizaje supervisado. El perceptron presenta dos capas de unidades procesadoras y solo una de ellas presenta la capacidad de adaptar o modificar pesos. La arquitectura admite capas adicionales pero estas no disponen la capacidad de modificar sus propias conexiones. En la figura 2.5 se muestra la unidad básica de procesamiento del perceptron, las entradas a_i a la izquierda, cada conexión tiene asignada un peso w_{ji} . La unidad del perceptron realiza la suma ponderada de las entradas según la ecuación: [3].

$$S_j = \sum a_i w_{ji}$$

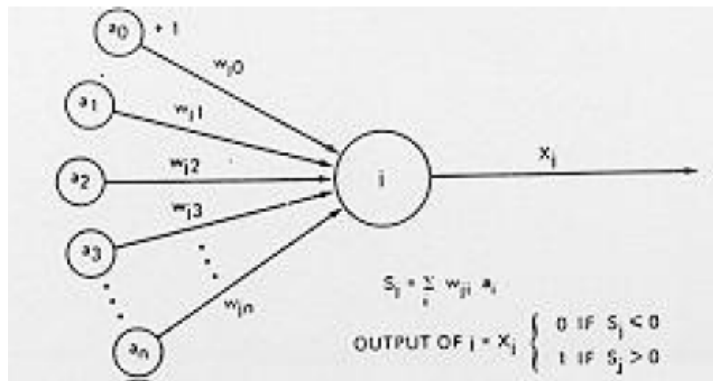


Figura 2.15: Unidad procesadora básica del perceptron [3].

Unas de las características mas comunes de este tipo de modelo es que presenta una entrada adicional conocida con el nombre de bias. Esta entrada siempre representa un valor fijo y funciona como una masa en un circuito donde no varia de valor. El perceptron comprueba si las entradas ponderadas es mayor o menor que un cierto umbral y genera una salida según la ecuación:

si $S_j > 0$ entonces $x_i = 1$

si $S_i \leq 0$ entonces $x_i = 0$

El entrenamiento del perceptron consiste en presentar a la red todo el conjunto de entrenamiento constituido por parejas de vectores (entrada y salida deseada) de forma secuencial. El objetivo es llegar a que los pesos respondan correctamente a todo el conjunto de entrenamiento. Después del entrenamiento los pesos ya no son modificados. La adaptación de pesos se puede realizar mediante a siguiente ecuación [3]:

$$w_{jiviejo} = w_{jiviejo} + C(t_j * x_j)a_i$$

Si t_i el valor de la salida deseada, x_i el valor de la salida producida por la neurona, a_i el valor de la entrada i y C el coeficiente de aprendizaje.

Todo el proceso de entrenamiento de la red va mejorando hasta que se estabiliza y se dice que la red ha convergido. Esta convergencia tiene dos posibilidades, la primera que la red haya aprendido correctamente el conjunto de entrenamiento o que la red no ha aprendido todas las respuestas correctas. Una de las limitaciones del perceptron es que unicamente puede separar patrones linealmente separables [3].

2.2.3.14. Red BackPropagation

Backpropagation es un método de entrenamiento de redes multicapa, su potencia reside en la capacidad de entrenar capas ocultas y de esta manera superar las deficiencias de las redes de una sola capa. El perceptron básico fue presentado en 1974 por Paul Werbos e inventado por David Parker en 1982 [3].

La unidad básica se representa mediante la figura 2.16, las entradas se muestran a la izquierda y a la derecha se encuentran unidades que reciben la salida de la unidad procesadora. Se caracteriza por realizar una suma ponderada llamada S_j , presentar una salida a_j y tener un valor δ_j que se usará en proceso de ajuste de pesos. El peso se representa por w_{ji} . Normalmente, las redes backpropagation utiliza tres o mas capas de unidades procesadoras [3].

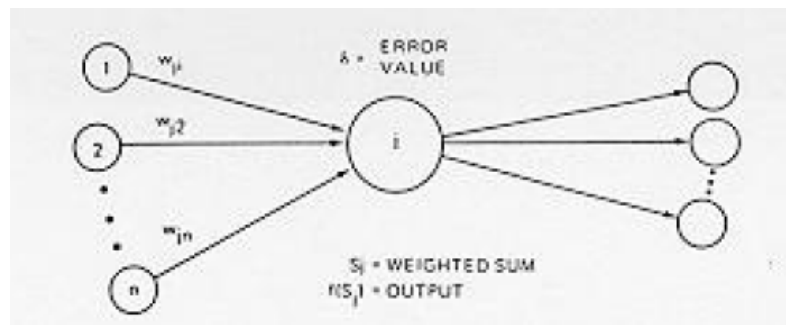


Figura 2.16: Unidad de procesamiento básico backpropagation [3].

Las redes backpropagation tienen un método de entrenamiento supervisado. A la red se le presenta una pareja de patrones (entrada emparejado con un patrón de salida deseada). Por cada presentación los pesos son ajustados de forma que disminuyan el error entre la salida deseada y la respuesta de la red. El algoritmo conlleva a una fase de propagación hacia adelante y otra hacia atrás, ambas fases se realiza por cada patron presentado en el entrenamiento [3].

2.2.3.15. Propagación hacia adelante

Esta fase se inicia cuando se presenta un patrón en la capa de entrada. Las unidades de entrada toman el valor y calcula el valor de activación o nivel de salida de la primera capa. A continuación, las demás capas realizarán la fase de propagación hacia adelante que determinan el nivel de activación de las otras capas. La unidad procesadora j obtiene la cantidad S_j según la ecuación [3]:

$$S_j = \sum_i a_i w_{ji}$$

Se genera una salida dada por la ecuación:

$$Salida = f(S_j)$$

La función f es una función umbral genérica tales como sigmoide o la función hiperbólica. El valor de la salida j es transmitido a lo largo de todas las conexiones de salida. Algunas redes backpropagation utilizan unidades llamadas bias como parte de cualquiera de las capas ocultas y de la capa de salida. Estas unidades presentan un nivel de activación de valor 1, además, esta unidad está conectada a todas las unidades de la capa superior y los pesos de dichas conexiones son ajustables. La utilización de esta unidad sirve para mejorar las propiedades de convergencia y ofrecer un nuevo efecto umbral sobre la unidad que opera [3].

2.2.3.16. Propagación hacia atrás

Una vez completado la propagación hacia delante se inicia la fase de corrección o fase de propagación hacia atrás. Los cálculos de todos los pesos empiezan por la capa de salida y continúan hacia atrás a través de todas las capas de la red hasta la capa de entrada. Dentro de los ajustes se pueden encontrar dos tipos: ajuste de unidades procesadoras de la capa de salida y ajuste de unidades procesadoras de las capas ocultas [3].

En el ajuste de pesos de la capa de salida, el ajuste de pesos es sencillo ya que se conoce el valor deseado para cada unidad de la capa de salida. Cada unidad de la capa de salida produce un número real como salida y se compara con el valor deseado especificado en el patrón del conjunto de entrenamiento. A partir del resultado se calcula el valor de error δ_j , según la ecuación [3]:

$$\delta_j = (t_j - a_j) f'$$

Siendo t_j el valor de salida deseado y f' la derivada de la función umbral f .

En el ajuste de pesos de las capas ocultas, debido a que estas capas no tienen vetos de salida deseada, el valor del error se obtiene a partir de la ecuación:

$$\delta_j = [\sum_k \delta_k w_{kj}] f'$$

El ajuste de los pesos se realiza a partir del valor de cada unidad de proceso, cada peso es ajustado según la ecuación:

$$w_{ji} = \eta \delta_j a_i$$

La variable η es el coeficiente de aprendizaje, normalmente está entre 0.25 y 0.75.

Convergencia: en el proceso de entrenamiento o aprendizaje de la backpropagation es medir cuantitativamente el aprendizaje mediante el valor RMS (Root Mean Square) del error de la red. Esta medida refleja el modo en que la red está logrando respuestas correctas a medida que la red aprende. Debido a que estos valores son reales, es necesario definir un valor umbral del valor RMS del error de la red que permita decir que la red se aproxima a la salida deseada y considerar que la respuesta es correcta [3].

CAPÍTULO 3

Metodología

3.1. Modalidad de la investigación

Investigación aplicada, en la cual se ha puesto en práctica los conocimientos técnicos y científicos adquiridos durante la formación académica, además información sobre la tecnología actual que sirvió para dar solución al problema planteado en la investigación.

Investigación Bibliográfica, por ser una de las herramientas básicas en el inicio de cualquier tipo de investigación, debido a esto se recopiló información para el desarrollo de este trabajo de libros, revistas, artículos científicos, publicaciones existentes, tesis, etc.

Investigación de campo ya que se necesitó realizar mediciones de las concentraciones de los contaminantes del aire.

Investigación Experimental, debido a que se realizó varias pruebas de funcionamiento hasta lograr un sistema óptimo y eficiente.

3.2. Recolección de información

Uno de los factores importantes en la investigación, es la recolección de información, ya que de ello depende la confiabilidad y validez del trabajo. Para la recolección de los datos necesarios, se recurrió al centro de monitoreo de la calidad del aire del Gobierno Administrativo Descentralizado de la Municipalidad de Ambato (GADMA). Además, se diseñó e implementó un sistema de medición de los contaminantes PM10 y PM2.5, integrados sobre una plataforma móvil.

3.3. Procesamiento y análisis de datos

Para llevar a cabo el desarrollo del proyecto se aplicó la siguiente metodología:

- Análisis de la información compilada.

- Depuración de la información.
- Procesamiento de la información.
- Elaboración de una arquitectura esquematizada del diseño.
- Verificación del diseño de prototipo.
- Evaluación del funcionamiento del prototipo y ajustes.
- Presentación de resultados.

3.4. Desarrollo del Proyecto

El desarrollo del proyecto se realizó de acuerdo a los siguientes objetivos:

- Descripción de los contaminantes PM10 y PM2.5 y sus afecciones en la salud humana.
- Recopilación de datos históricos de los niveles de PM10 y PM2.5 en la ciudad de Ambato.
- Análisis de las técnicas y sensores para la medición de los contaminantes PM10 y PM2.5.
- Selección de dispositivos electrónicos utilizados en el sistema de medición de material particulado.
- Implementación del sistema de medición de material particulado PM10 y PM2.5.
- Pruebas de funcionamiento y corrección de errores del sistema móvil de medición de los contaminantes.
- Determinación de los modelos de Redes Neuronales Artificiales utilizados en la predicción de datos.
- Desarrollo del modelo de Red Neuronal Artificial para la predicción a un día de los contaminantes PM10 y PM2.5.
- Entrenamiento del modelo de Red Neuronal Artificial con datos de los niveles de contaminación en la ciudad de Ambato.

- Evaluación del funcionamiento de la Red Neuronal Artificial por medio del índice RMSE.
- Análisis de plataformas cloud compatibles con Redes Neuronales Artificiales.
- Implementación de la plataforma cloud de monitoreo, adquisición, almacenamiento, gestión, visualización y predicción de los contaminantes PM10 y PM2.5.
- Pruebas de funcionamiento y corrección de errores.
- Informe final.

CAPÍTULO 4

Desarrollo de la propuesta

4.0.1. Área de estudio

El estudio se desarrolla en la ciudad de Ambato, campus de la Universidad Técnica de Ambato. La ciudad se encuentra en la provincia de Tungurahua que viene dado por las coordenadas $1^{\circ}15'57.73''$ S y $78^{\circ}36'56.95''$ O a 2570 metros sobre el nivel del mar. Su temperatura varía de entre 14°C a 19°C . La población existente en la ciudad de Ambato es alrededor de 329900 habitantes de acuerdo al INEC 2010. Gran parte de su población se dedica a actividades de comercio, industrias, agricultura y ganadería. El área de estudio se encuentra en la Universidad Técnica de Ambato, campus Huachi, remarcado con un polígono de color verde como se puede observar en la figura 4.1.



Figura 4.1: Mapa de la ciudad de Ambato (Google Earth)

Fuente: Autor

La ciudad de Ambato, en la actualidad cuenta con un plan de calidad del aire realizado por el Gobierno Autónomo Descentralizado Municipal de Ambato (GADMA) que abarca un sistema de monitoreo automático del aire, ubicado en el edificio matriz del GADMA Sur. Se realiza el monitoreo segundo a segundo, las 24 horas del día, los siete días de la semana y los 365 días del año a partir del

2016. En la figura 4.2 se pueden observar los valores de material particulado PM2.5, registrados por la estación de monitoreo de calidad de aire del GADMA referente al año 2016.

Los valores registrados en el mes de diciembre del año 2016 para material particulado PM2.5 son de $22.3 \text{ ug}/\text{m}^3$, lo cual se aproxima al valor máximo permisivo establecido por la Organización Mundial de la Salud (OMS) que es de $25 \text{ ug}/\text{m}^3$. Estos valores son alentadores puesto que en la ciudad de Ambato presenta un aire poco contaminado en el 2016 y por lo tanto no conlleva a ningún riesgo para la salud de las personas.

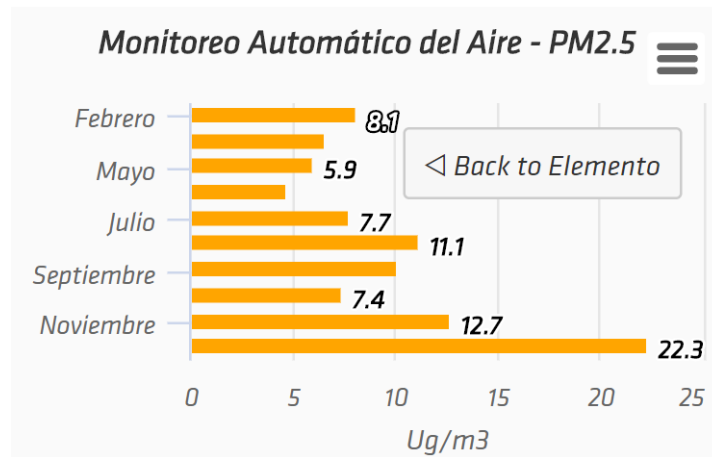


Figura 4.2: Datos de material particulado PM2.5 obtenidos por la estación de monitoreo del GADMA correspondientes al año 2016 [4].

En el año 2017, la estación de monitoreo de la calidad del aire perteneciente al GADMA no registra datos para los meses de Enero, Febrero y Marzo del 2017 por razones desconocidas. Los valores de concentración mas altos corresponde a los meses de Junio y Octubre del 2017 con valores de 10.5 y 10.8, respectivamente, valores que se encuentran muy por debajo de los límites establecidos por la Organización Mundial de la Salud. Una vez mas se puede concluir que el aire para el 2017 no se encontraba contaminado en gran medida por este tipo de material.

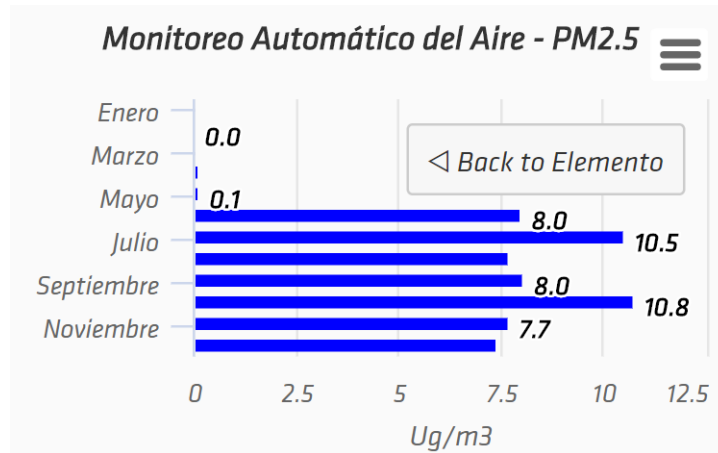


Figura 4.3: Datos de material particulado PM2.5 obtenidos por la estación de monitoreo del GADMA correspondientes al año 2017 [4].

En la tabla 4.1 se pueden observar las concentraciones de material particulado registrados mes a mes del año 2017. Los valores más altos se registran en el mes de Julio y Octubre. En promedio anual, se registran valores de $5.018 \text{ ug}/\text{m}^3$, lo cual no excede el límite establecido por la OMS correspondiente a $10 \text{ ug}/\text{m}^3$.

Mes	Promedio ug/m^3
Enero	0.019
Febrero	0.010
Marzo	0.017
Abril	0.092
Mayo	0.093
Junio	7.99
Julio	10.51
Agosto	7.66
Septiembre	8.02
Octubre	10.76
Noviembre	7.66
Diciembre	7.38
Promedio	5.018

Tabla 4.1: Valores de material particulado registrados por el GADMA para el año 2017 [4].

4.0.2. Diseño del Prototipo

En la figura 4.4 se muestra el diagrama de bloques para la implementación de la propuesta. El sistema consta de un módulo GPS Ublox NEO 6M para

el posicionamiento de la plataforma móvil de monitoreo y adquisición de los contaminantes PM10 y PM25. Un sensor DHT11 para la medición de humedad y temperatura del ambiente y un modulo WiFi ESP8266, que se encarga del envío de datos hacia el servidor remoto, Centos 7 configurado en Google Cloud Computing.

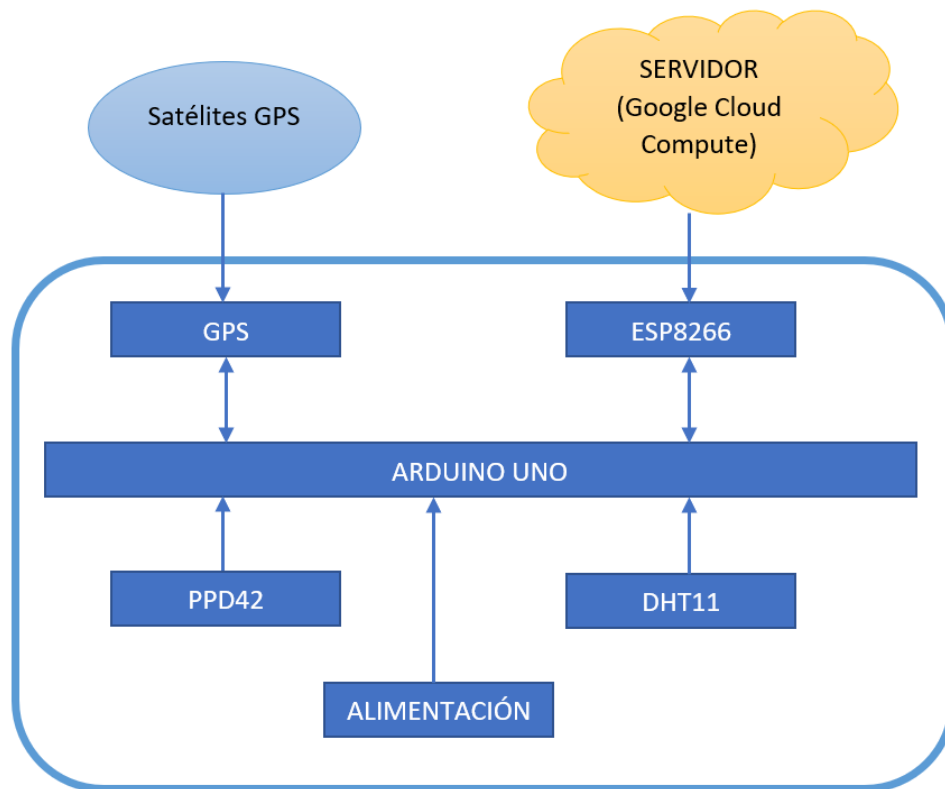


Figura 4.4: Diagrama de bloques del sistema de medición de PM10 y PM2.5
Fuente: Autor

4.0.2.1. Sensor de polvo PPD42

PPD42 es un contador de partículas económico que mide el material particulado en el aire, contando el número de pulsos en un determinado tiempo. El ancho de pulso es proporcional a la concentración de material particulado. Es un sensor que ofrece detección de partículas desde 1 μm hasta 10 μm . Consta de un diodo emisor de luz y un detector de fotodiodos que mide las propiedades de dispersión de partículas cercanas en la corriente de aire. Un calentador resistivo ubicado en la entrada de la cámara de luz ayuda a mover el aire desde la entrada inferior a la salida superior. Los componentes adicionales controlan la detección y procesamiento de la señal, obteniendo como resultado una salida modulada por ancho de pulso. En la figura 4.5 se puede observar los componentes internos del sensor.

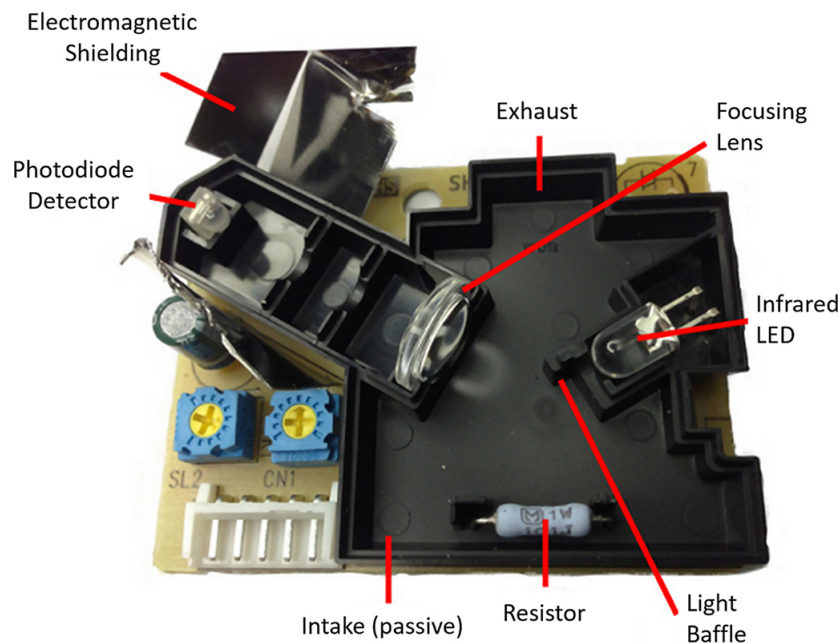


Figura 4.5: Elementos internos del sensor PPD42

Tamaño de partículas detectable	1um a 10um
Rango de concentración detectable	0-28000 pcs/litro
Voltaje de funcionamiento	5V DC
Consumo de energía	90mA
Temperatura de operación	0-45°C
Humedad de funcionamiento	95 % max
Tiempo de estabilización	1 minuto
Dimensiones	59mmx45mmx22mm
Peso	24g

Tabla 4.2: Especificaciones técnicas del sensor PPD42

4.0.2.2. Modulo GPS

La mayoría de los módulos GPS que se encuentran en el mercado están hechos para trabajar con una simple comunicación serial de los datos adquiridos directamente de los satélites. Por lo tanto se escoge el GPS NEO 6M, como se muestra en la figura 4.6, por su gran sensibilidad en la recepción de señales, garantizando que el GPS mantenga su comunicación con los satélites en una variedad de condiciones de operación. Su comunicación con el arduino se realiza por medio de los pines digitales.

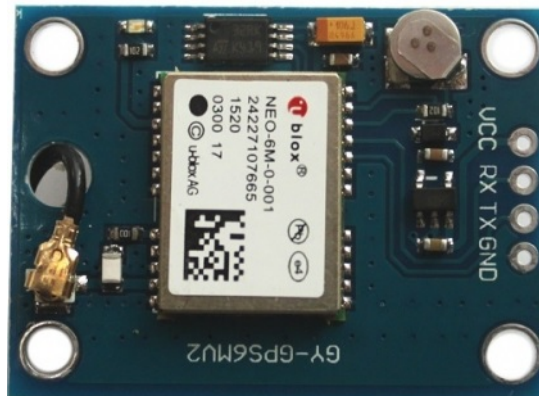


Figura 4.6: GPS NEO UBLOX 6M

Voltaje de alimentación	3-5 VDC
Interfaz	Serial UART 5V
Antena	Cerámica
frecuencia de refresco	5Hz
Tamaño	25mmx35mm
Baud Rate	9600bps
Soporta	SBAS, WAAS, EGNOS, MSAS, GAGAN
Temperatura de funcionamiento	-40 a 85°C
Peso	12g
Sensibilidad de captura	-148dBm
Sensibilidad de rastreo	-161dBm
Sistema de coordenadas	WGS-84
Máxima altura medible	18000
Máxima Velocidad	515 m/s
Frecuencia receptora	1575.42 MHz

Tabla 4.3: Características del GPS NEO UBLOX 6M

4.0.2.3. Sensor de temperatura y Humedad

El sensor que se muestra en la figura 4.7 corresponde a un sensor de temperatura y humedad ambiental. Se caracteriza por tener una señal de salida digital, que asegura fiabilidad y confiabilidad a lo largo del tiempo, además, cuenta con un convertor de 16 bits a la salida. Internamente está constituido por sensores resistivos, calibrados de fábrica ofreciendo una alta precisión. Presenta un tamaño reducido y bajo costo de adquisición, así como bajo consumo de energía y la capacidad de transmitir la señal hasta 20 metros de distancia.

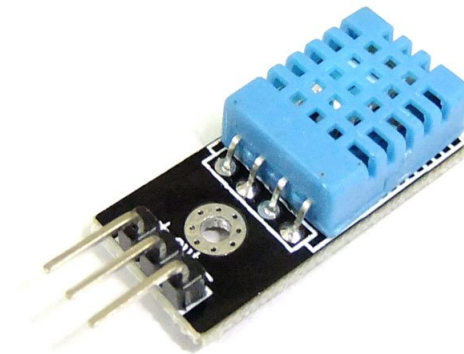


Figura 4.7: Sensor de Temperatura y Humedad DHT11

Rango de temperatura	0°C - 50°C
Rango de humedad	20 a 90 %
Exactitud humedad	±5 %
Exactitud temperatura	±2°C
Periodo de muestreo	2 segundos
Tensión de alimentación	3 a 5 Voltios
Corriente de operación	2.5mA
Tamaño	15mmx12mmx5.5mm

Tabla 4.4: Características del sensor DHT11

4.0.2.4. Arduino

En la actualidad existen muchos dispositivos de hardware libre empleados para el procesamiento de información, que se basan exclusivamente en microcontroladores implementados en PCB. Entre estos dispositivos, Arduino se destaca por ser una de las plataformas más usadas dentro del campo educativo y experimental, además de que su software es abierto y de fácil programación.

Arduino Uno es una placa basada en un microcontrolador ATmega328P, que posee 14 pines digitales de entrada/salida de los cuales se puede usar como entradas analógicas. En la tabla 4.5 se puede observar las características de la placa de desarrollo Arduino Uno.

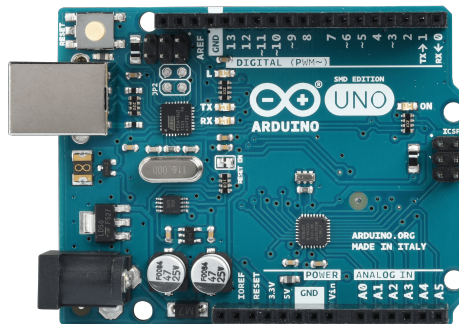


Figura 4.8: Arduino Uno R3

Microcontrolador	ATmega328P
Tensión de funcionamiento	5V
Voltaje de alimentación	7 - 12V
Voltaje de entrada (Límite)	6-20V
Pines E/S digitales	14
Pines E/S PWM	6
Pines de entrada analógicos	6
Corriente DC por Pin	20mA
Corriente DC para 3.3V Pin	50mA
Memoria flash	32Kb
SRAM	2Kb
EEPROM	1Kb
Velocidad de reloj	16MHz
LED_BUILTIN	13
Longitud	68.6mm
Anchura	53.4mm
Peso	25g

Tabla 4.5: Especificaciones técnicas de la plataforma arduino uno

4.0.2.5. Conexión del sistema

Una vez realizada la selección de los equipos se procede a la conexión de los mismos. El diagrama de conexión se muestra en la figura 4.9. La primera etapa es la alimentación de la placa Arduino Uno, que se realiza a través del puerto de alimentación y permite una tensión máxima de hasta 20VDC. La conexión de los demás dispositivos GPS, PPD42 y DHT11, se realizará por medio de los pines de alimentación 5VDC y GND respectivamente. La alimentación del módulo WiFi se realizará por el puerto USB que permite una entrada de 5VDC.

Posteriormente, se requiere comunicar los demás dispositivos. El módulo DHT11 que

permite censar la temperatura y humedad se realiza por medio del pin digital 2 del arduino. A continuación se procede a conectar el dispositivo PPD42 para determinar los valores de PM10 y PM2.5 en los puertos digitales 3 y 4 respectivamente. Para la conexión del GPS Ublox NEO 6M se hace a través de los pines digitales 6 (Rx) y 5 (Tx) conjuntamente con la librería “SoftwareSerial”, que permite establecer comunicación serial por pines digitales a una velocidad máxima de 9600bps. El módulo WiFi ESP8266 se conectan a los pines digitales 7 (Rx) y 8 (Tx) para la transmisión de los datos hacia una base de datos en el servidor Centos 7 alojado en Google Cloud Platform.

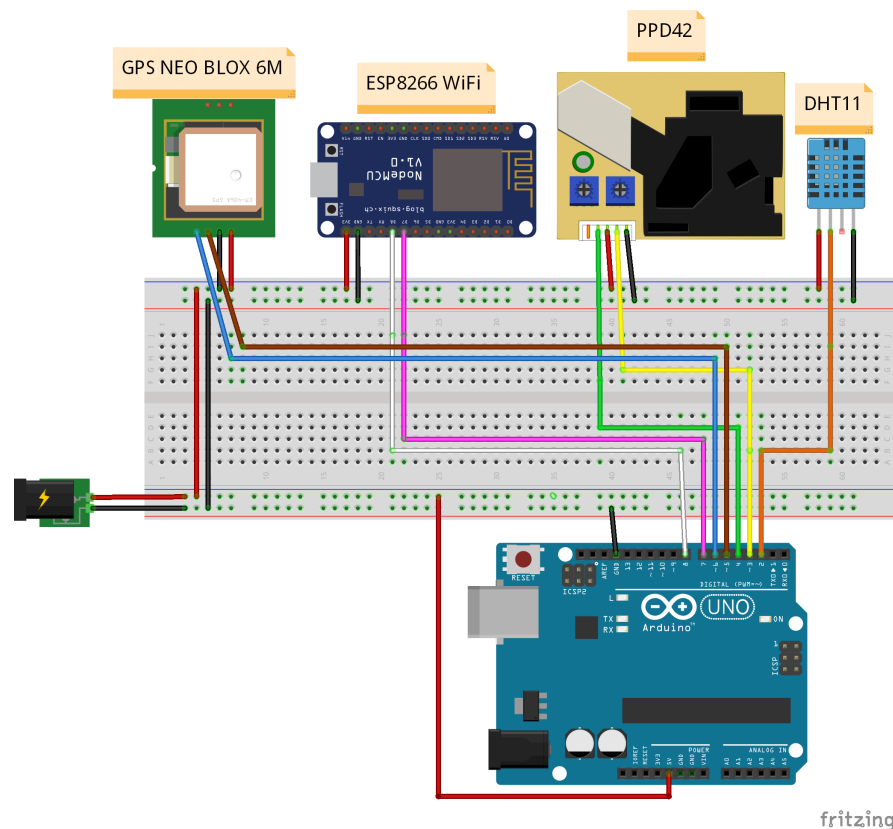


Figura 4.9: Diagrama de conexión del sistema de medición de PM10 y PM2.5
Fuente: Autor

4.0.2.6. Algoritmo del programa principal

El algoritmo de la programación del Arduino Uno se ilustra en la figura 4.10. Inicialmente se incluye la librería “DHT11.h” usado para la lectura de datos del sensor de Temperatura y Humedad, además, se incluye la librería “SoftwareSerial.h” que permite realizar la comunicación serial, utilizando otros puertos digitales del arduino y establecer una comunicación con el módulo GPS NEO 6M y con el modulo

WiFi. También se incluye la librería “TinyGPS.h” que es una librería para analizar flujos de datos NMEA provista por módulos GPS. Esta biblioteca provee métodos compactos y fáciles de usar para extraer la posición, fecha, hora, altitud, velocidad y rumbo de los dispositivos GPS.

A continuación se realiza la declaración de variables globales: “lat” y “lon” donde se almacenan los valores de latitud y longitud, respectivamente obtenidos del GPS Ublox NEO 6M, “pm10” y “pm2.5”, que son las variables que permitirán conservar los valores de material particulado PM10 y PM2.5 obtenidos a través del sensor PPD42, “temp” y “hum” son variables en las cuales se almacenaran los valores de temperatura y humedad del ambiente censadas con el DHT11.

En la subrutina setup se realizan las configuraciones iniciales del sistema. Por lo tanto, lo primero que se realiza es la habilitación de los pines digitales 3 y 4 como entradas de datos. A continuación se habilita la comunicación serial a 9600bd para la comunicación con la computadora e impresión de datos a través de la consola. Posteriormente, se inicializa la lectura de los datos del sensor DHT11, la comunicación serial del GPS y ESP8266 a la velocidad de 9600bd cada uno.

En el cuerpo del programa se llaman a las funciones de censado de material particulado, temperatura y humedad.

- leerDHT() es el método de devuelve la temperatura y humedad registrada por el sensor en grados centígrados y en porcentaje respectivamente. El algoritmo verifica si no existe un error en la adquisición de los datos. Si son correctos se procede a almacenarlos en las variables globales.
- leerPM() devuelve la concentración de material particulado PM10 y PM2.5 registrados por el sensor PPD42 en unidades ug/m^3 , además, verifica que las lecturas no sean erróneas, de no serlo se almacena en las variables globales.
- leerGPS() es el método utilizado para la lectura de los valores de posicionamiento (latitud y longitud). También registra la precisión de las lecturas cuando el valor HDOP es mayor que 70 y se almacena en las variables globales.

Finalmente, se hace una verificación de todas las lecturas. Si son correctas se procede a enviar hacia el módulo WiFi, encargado de subir a la base de datos MySQL en el servidor remoto.

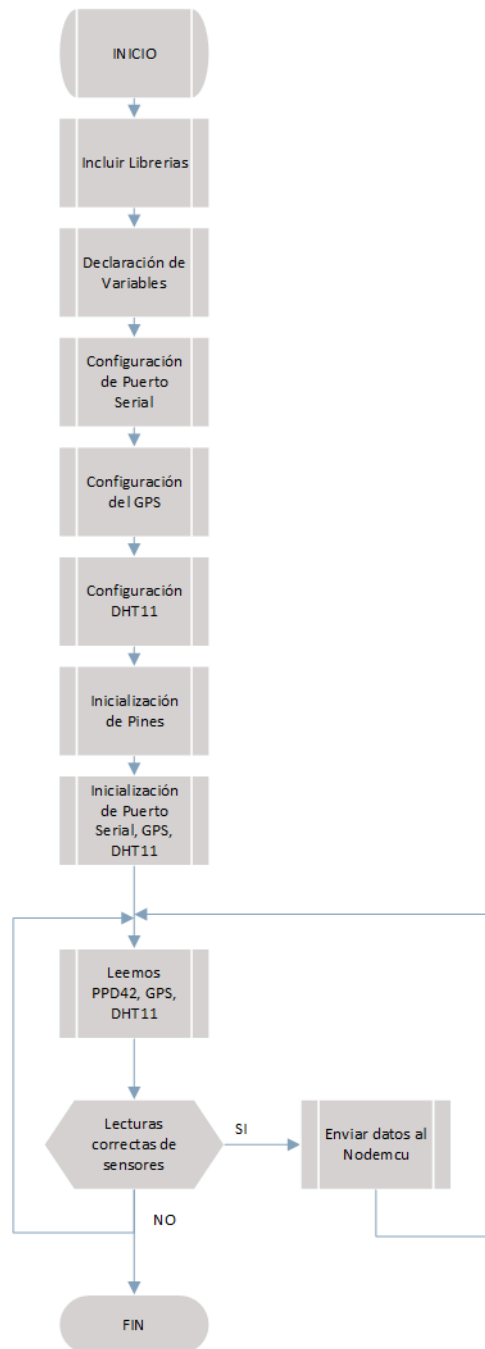


Figura 4.10: Diagrama de flujo del programa principal (arduino)
Fuente: Autor

4.0.2.7. Algoritmo del módulo WiFi

La programación del módulo WiFi Nodemcu basado en el ESP8266 se presenta en la figura 4.11. Inicialmente se procede a incluir la librería “ESP8266WiFi.h”, empleada para realizar la comunicación WiFi con el servidor y la librería “SoftwareSerial.h”

que establece la comunicación serial con el arduino. A continuación se declaran las variables globales ssid donde se almacena el nombre de la red WiFi, el password para almacenar la contraseña del punto de acceso, host donde se almacena la dirección del servidor remoto. Adicionalmente, se declara otras variables globales donde se almacenaran los datos recibidos por comunicación serial del arduino.

En la subrutina setup se realizan todas las configuraciones iniciales del sistema. Se procede con la inicialización de las comunicaciones seriales tanto para la consola como para la comunicación con el arduino. Además, se inicializa la comunicación WiFi con el método “WiFi.begin(ssid, password)”, donde ssid es el nombre de la red y password la contraseña. En el cuerpo del programa se revisa continuamente si hay algún dato en el puerto serial. Si lo hay se procede a leerlo y almacenar en las variables globales. Luego se revisa si los datos se recibieron correctamente para proceder a subirlo a la base de datos MySQL, usando el método “enviar_wifi()” que se conecta al puerto 80 y se realiza la petición usando el método GET seguido de los parámetros y valores.

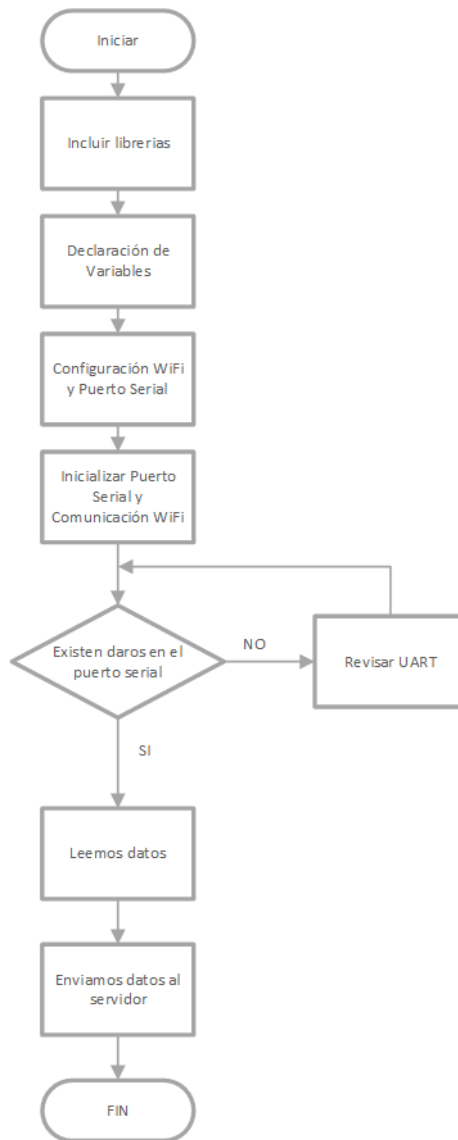


Figura 4.11: Diagrama de flujo del módulo WiFi
Fuente: Autor

4.0.3. Plataforma Cloud

La computación en la nube ofrece optimización de costos operativos, reduce riesgos ofreciendo la posibilidad de una menor inversión inicial, proporciona flexibilidad de acceso y el uso eficaz de los recursos. La oferta de computación en la nube es amplia y en la mayoría de los casos resulta complicado elegir la opción más idónea de acuerdo al proyecto. Dentro del mercado de la computación en la nube se encuentran tres grandes empresas dedicadas a ofrecer estos servicios: Amazon Web Services (AWS), Google Cloud y Microsoft Azure. Para la presente investigación se ha elegido a Google Cloud Platform, debido a las prestaciones que ofrece y sus bajos costos. Además

porque ofrece un año de prueba gratuita ideal para realizar pruebas e investigación dentro de este campo.

La interfaz de la nube de Google es limpia y fácil de navegar. Otra ventaja de google es su rápido despliegue equilibrado de carga. Google App Engine es una PaaS de alto rendimiento que soporta lenguajes de programación Python, Java, PHP y Go. La plataforma trabaja con gran variedad de opciones de almacenamiento incluyendo Cloud SQL, Datastore y Blobstore, además proporciona acceso a la API de búsqueda.

4.0.3.1. Configuración de Google Cloud Computing

Para la creación de una instancia y desplegar un servidor se procede a ingresar en la consola de Google Cloud Platform, en la pestaña Compute Engine, seguido de instancias de VM crearemos nuestra instancia como se puede observar en la figura 4.10. En la sección nombre ubicamos el nombre que le daremos a la instancia, seguido seleccionamos la zona donde se almacenarán los recursos informáticos como se puede observar en la figura 4.13.

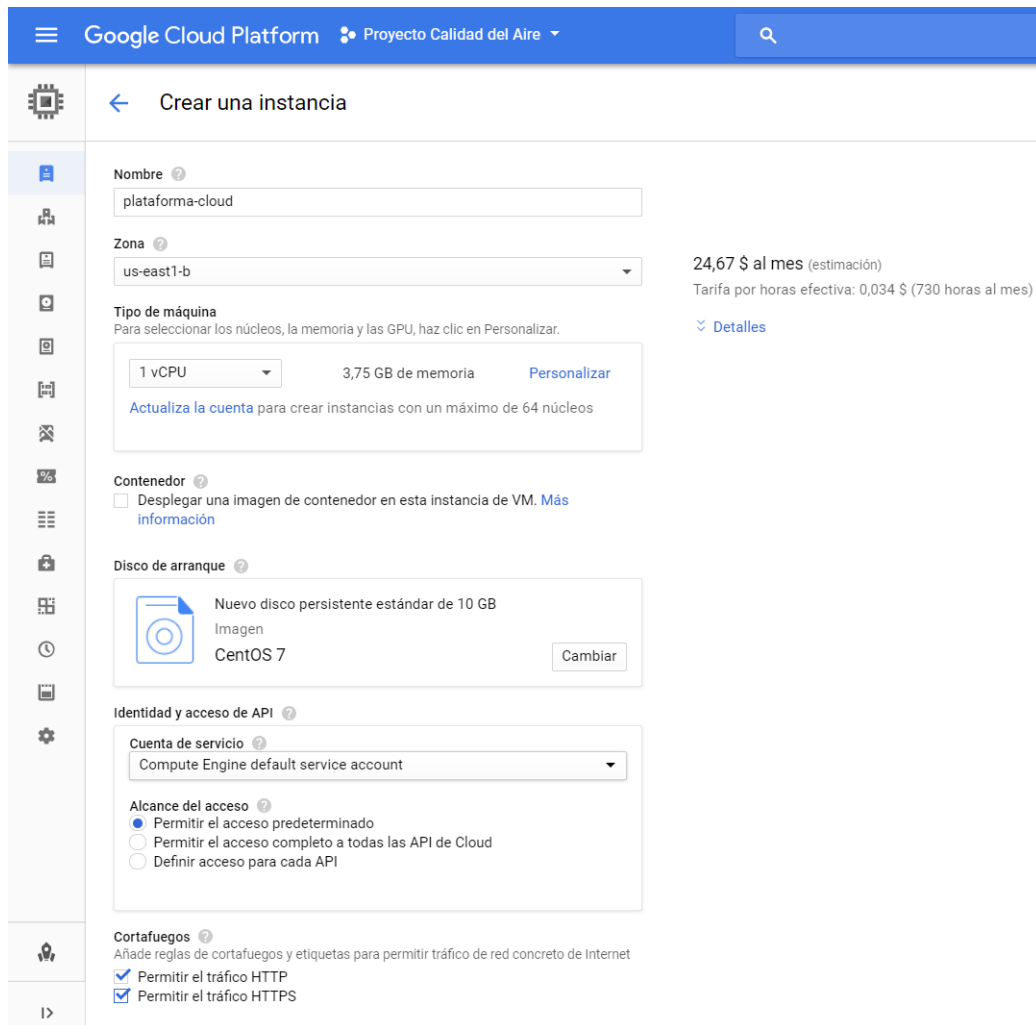


Figura 4.12: Creación de una instancia en Google Cloud Platform.
Fuente: Autor

En la sección tipo de máquina se puede elegir la cantidad de CPUs, desde CPUs compartidos hasta 8 CPUs con 64 núcleos en una cuenta premium. Además de la posibilidad de elegir un disco duro estándar o un disco duro de estado sólido (SSD), con un tamaño máximo de 65536 GB. Entre los sistemas operativos con los que cuenta Google Cloud Platform se tiene: Debian, Centos, CoreOS, Ubuntu, Container Optimized OS, Red Hat, Open Suse, Windows Server, e imágenes personalizadas. Finalmente se procede a habilitar el protocolo HTTP y HTTPS para ejecutar la instancia y acceder a ella por medio de SSH.

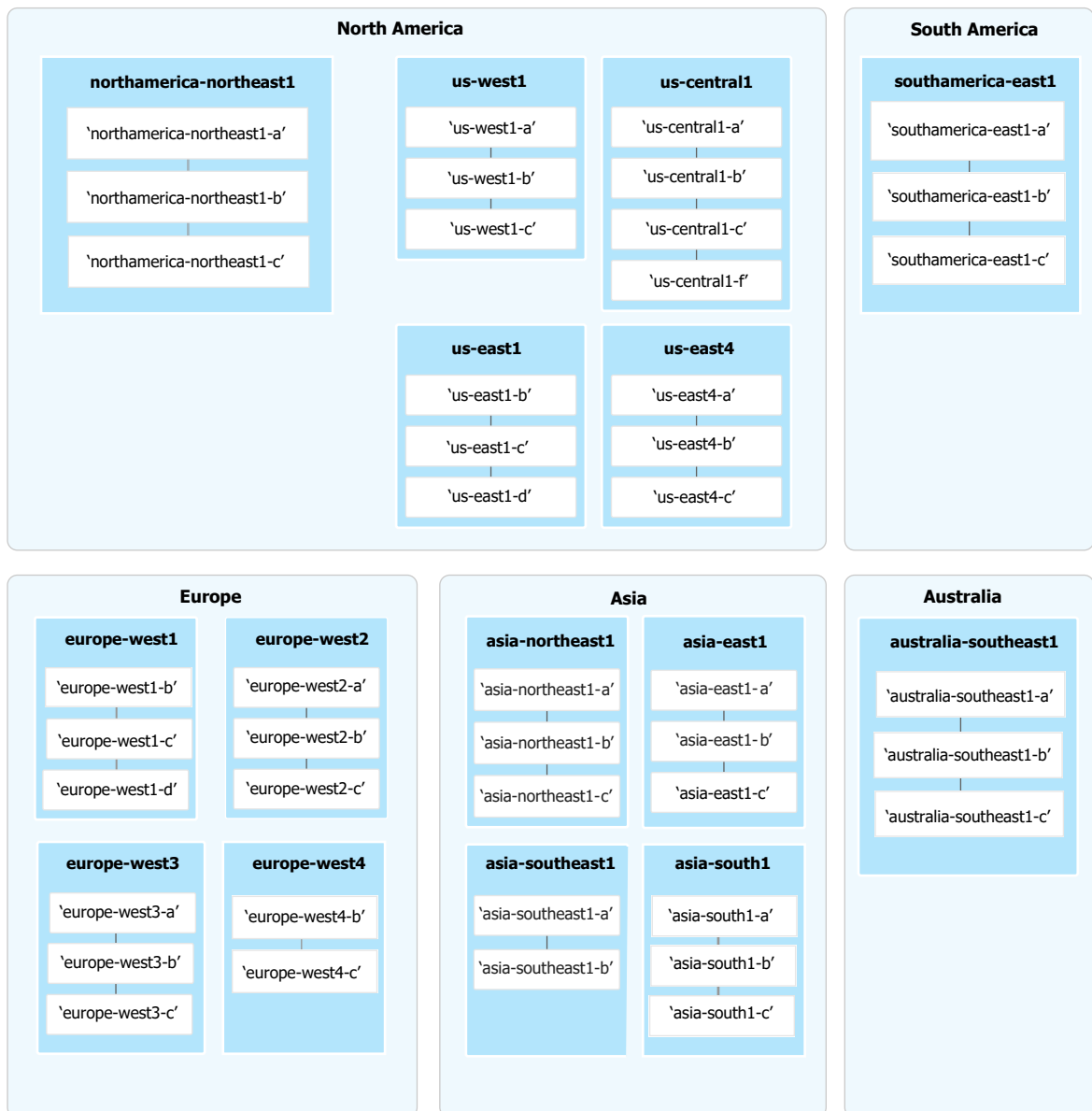


Figura 4.13: Zonas disponibles Google Cloud Platform
Fuente: Autor

4.0.3.2. Almacenamiento de Datos

Para almacenar los datos adquiridos por la plataforma móvil de monitoreo de material particulado, temperatura y humedad, se ha decidido utilizar una base de datos llamada MariaDB, que permite la administración y gestión de forma sencilla y dinámica. La base de datos creada se llama mp dentro la cual se encuentran tablas para el almacenamiento de datos. Una de las tablas de denomina “datos” que es donde se almacena la información recibida de la plataforma de medición de los contaminantes. La estructura se puede observar en la figura 4.14.

A dicha base de datos se puede acceder mediante autenticación, ya que se creó un usuario específico para la lectura y escritura de los datos. En la base de datos se encuentra la tabla de nombre “datos” con los siguientes campos:

- id: Campo de tipo entero autoincremental para almacenar en número de dato.
- fecha: Campo de tipo date para almacenar la fecha en la cual se realiza la medición.
- hora: Campo de tipo time para almacenar al hora de medición de los contaminantes.
- latitud: Campo de tipo double para almacenar la latitud del punto de medición.
- longitud: Campo de tipo double para almacenar los valores de longitud del punto de medición.
- pm10: Campo de tipo double para almacenar los valores de PM10 medidos.
- pm25: Campo de tipo double para almacenar los valores de PM25 medidos.
- temperatura: Campo de tipo double para almacenar los valores de temperatura.
- humedad: Campo de tipo double para almacenar los valores medidos de humedad.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
<input type="checkbox"/>	1 id	int(11)			No	Ninguna	AUTO_INCREMENT	Cambiar Eliminar Primaria
<input type="checkbox"/>	2 fecha	date			No	Ninguna		Cambiar Eliminar Primaria
<input type="checkbox"/>	3 hora	time			No	Ninguna		Cambiar Eliminar Primaria
<input type="checkbox"/>	4 latitud	double			No	Ninguna		Cambiar Eliminar Primaria
<input type="checkbox"/>	5 longitud	double			No	Ninguna		Cambiar Eliminar Primaria
<input type="checkbox"/>	6 pm10	double			No	Ninguna		Cambiar Eliminar Primaria
<input type="checkbox"/>	7 pm25	double			No	Ninguna		Cambiar Eliminar Primaria
<input type="checkbox"/>	8 temperatura	double			No	Ninguna		Cambiar Eliminar Primaria
<input type="checkbox"/>	9 humedad	double			No	Ninguna		Cambiar Eliminar Primaria

Figura 4.14: Campos de la base de datos MySQL
Fuente: Autor

4.0.4. Implementación de la plataforma web

Para la presentación de la información se ha optado por una página web, ya que únicamente se requiere de una conexión a internet para tener acceso. Además es

compatible con la mayoría de dispositivos electrónicos, tanto de escritorio como móviles. Esta plataforma fue desarrollada con lenguaje de programación HTML complementándose con JavaScript para la realización de los diferentes gráficos y la integración del mapa. Así como PHP para la realizar consultas hacia la base de datos MySQL.

En la parte principal se ha colocado un mapa desarrollado con la API de Google Maps, que muestra la ubicación de la estación de monitoreo. Además se presenta un indicador circular de dispersión de color donde el verde es sinónimo de que las concentraciones se encuentran por debajo de los límites establecidos por la OMS. El color rojo indica que se ha excedido del valor máximo para la concentración del contaminante, establecido en $25 \mu\text{g}/\text{m}^3$. Además, se despliega un cuadro que muestra los valores de PM10, PM2.5, temperatura y humedad, registrados por la estación de monitoreo.

En la parte inferior del mapa se ha optado por un diagrama de barras en el cual se muestran el promedio diario correspondiente a los últimos 7 días de PM10, PM2.5, temperatura y humedad.

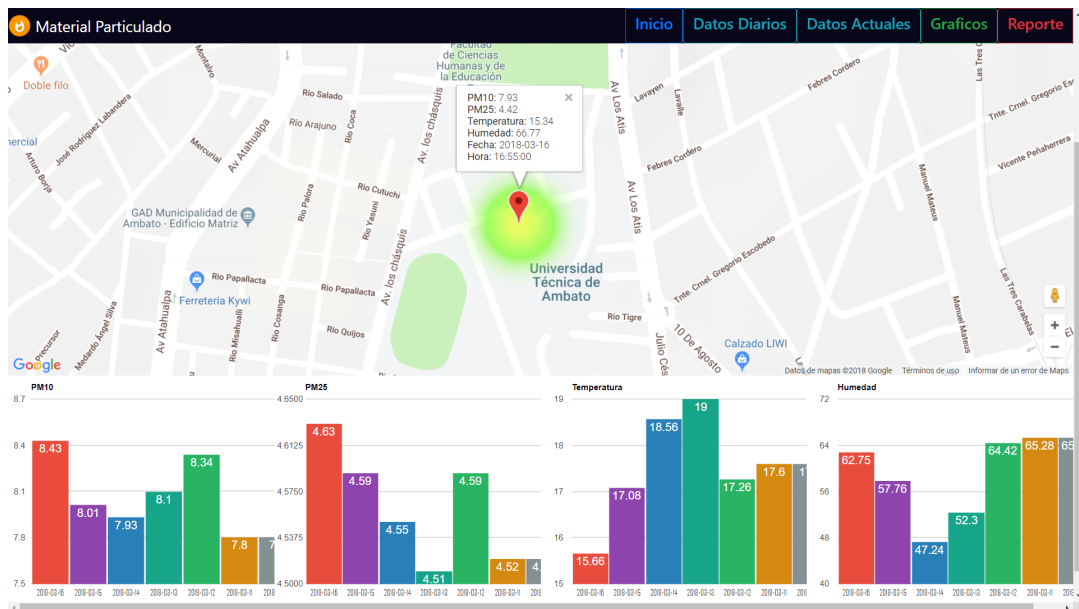


Figura 4.15: Página principal para la visualización de los datos obtenidos por la estación de monitoreo de PM10 y PM2.5

Fuente: Autor

4.0.4.1. Tablas de valores

Otra de las funcionalidades de la plataforma implementada, es que se puede observar los datos registrados minuto a minuto durante las 24 horas del día. Estos valores son mostrados por medio de una tabla. En la columna fecha y hora se puede constatar la temporalidad de la información. Las columnas latitud y longitud indican las coordenadas de la ubicación de la plataforma de monitoreo. Finalmente, las columnas con los nombres PM10, PM2.5, TEMPERATURA y HUMEDAD muestran los valores registrados por los sensores ubicados en la estación de monitoreo.

FECHA	HORA	LATITUD	LONGITUD	PM10	PM2.5	TEMPERATURA	HUMEDAD
2018-03-16	17:44:00	-1.268368	-78.624191	8.8	4.47	14	69
2018-03-16	17:43:00	-1.268355	-78.624214	7.3	6.98	17	69
2018-03-16	17:42:00	-1.268355	-78.624214	9.86	6.51	17	69
2018-03-16	17:41:00	-1.268316	-78.624306	7.93	5.48	16	70
2018-03-16	17:40:00	-1.26832	-78.624313	6.52	5.17	16	70
2018-03-16	17:39:00	-1.268321	-78.624268	12.07	6.01	14	70
2018-03-16	17:38:00	-1.268321	-78.624245	9.02	5.97	16	70
2018-03-16	17:37:00	-1.268289	-78.624237	7.06	4.66	13	69
2018-03-16	17:36:00	-1.268277	-78.624229	9.25	4.04	16	70
2018-03-16	17:35:00	-1.26829	-78.624229	7.8	4.14	17	68

Figura 4.16: Tabla de valores actuales registrados por la estación móvil de monitoreo de las estación móvil

Fuente: Autor

4.0.4.2. Gráficas

Esta es una de las funcionalidades donde se puede observar de mejor manera las variaciones de los valores registrados por la estación de monitoreo. Las gráficas correspondientes a las figuras 4.17, 4.18, 4.19, 4.20 muestran la información de material particulado PM10, PM2.5, temperatura y humedad del ambiente, respectivamente. Han sido desarrolladas utilizando el API de Google Charts, un código abierto que no necesita de ningún pago.

Las gráficas están formadas por un eje horizontal, que muestran la línea de tiempo de los valores registrados. En el eje horizontal se muestra la magnitud del contaminante. Se tiene la ventaja de que al posicionar el puntero sobre la línea se despliega de manera automática un recuadro indicando información de dicho punto señalado.

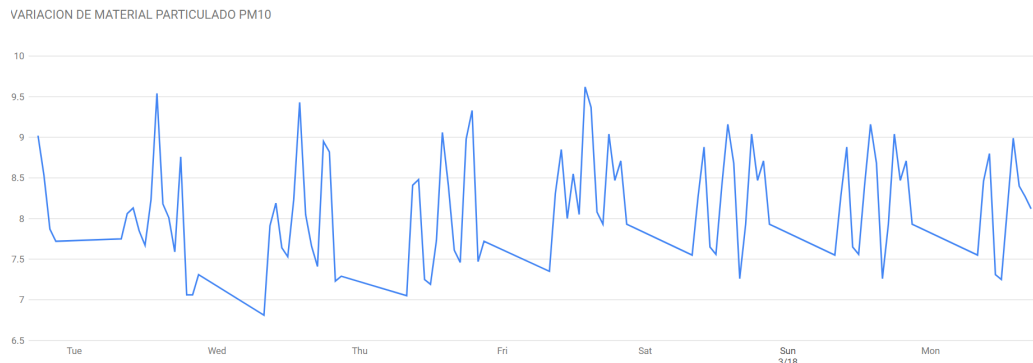


Figura 4.17: Gráfica del material particulado PM10 registrado por la estación de monitores

Fuente: Autor

En la figura 4.17 se observa la variación de la concentración del material particulado suspendido en la atmósfera comprendido entre 2.5 μg y 10 μg (PM10). La presentación de la información corresponde a los últimos 7 días desde la fecha actual. El horario donde se presenta una mayor concentración de este contaminante se registra entre las 12:00 y las 14:00, horario donde también se registra los niveles más altos de temperatura como se puede observar en la figura 4.19.

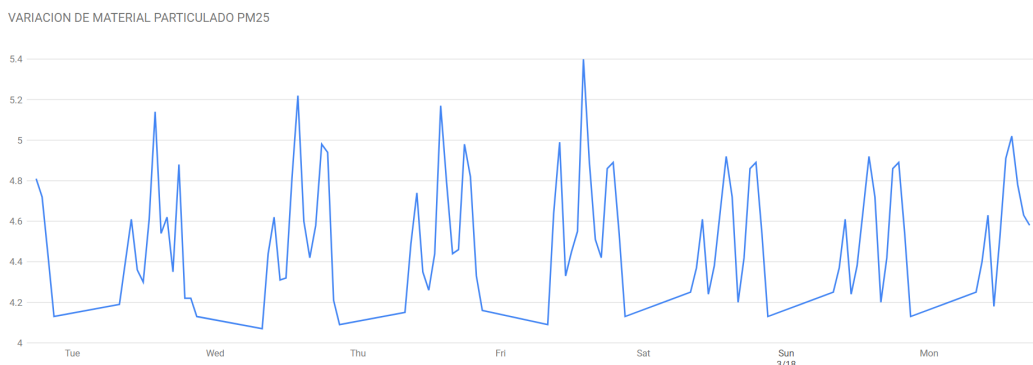


Figura 4.18: Gráfica de material particulado PM2.5 registrado por la estación de monitoreo

Fuente: Autor

En la figura 4.18 se observa la variación de la concentración del material particulado suspendido en la atmósfera inferior a 2.5 μg (PM2.5). Al igual que el material particulado PM10, el horario en que se presenta una mayor concentración de estas partículas suspendidas en la atmósfera está entre las 12:00 y 14:00 horas. Además se observa que el incremento de dichos niveles esta relacionado con el incremento de la temperatura y la reducción de la humedad como se observa en la figura 4.19 y 4.20.

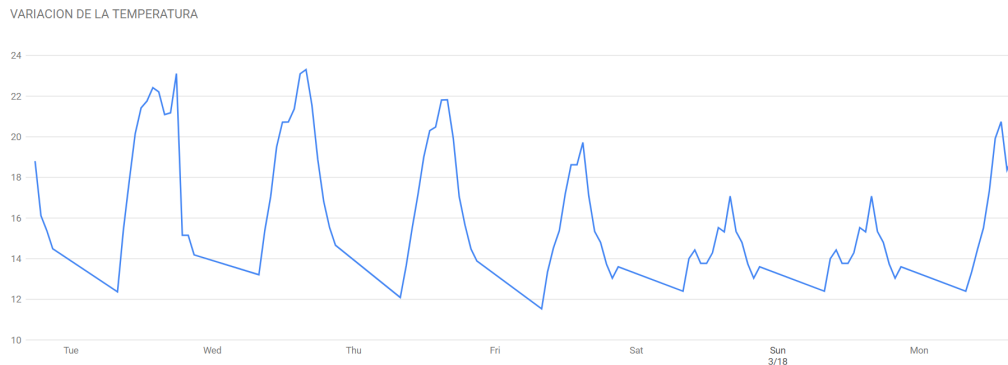


Figura 4.19: Gráfica de la temperatura del ambiente registrado por la estación de monitoreo

Fuente: Autor

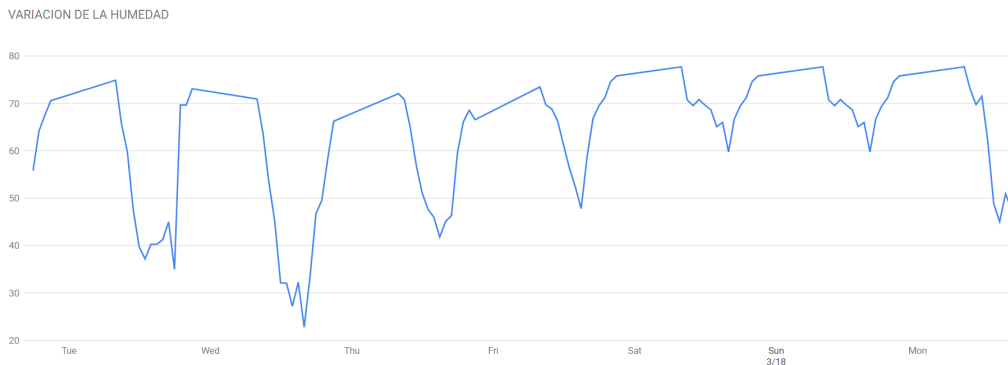


Figura 4.20: Gráfica de la humedad del ambiente registrado por la estación de monitoreo

Fuente: Autor

4.0.5. Predicción basado en Redes Neuronales Artificiales

Para realizar el entrenamiento de la Red Neuronal se hace uso del software Matlab, que permite manejar redes neuronales artificiales con gran facilidad, sin embargo se debe definir una serie de parámetros como el tipo de red neuronal, número de capas, número de neuronas de la capa de entrada, número de neuronas de las capas intermedias y número de neuronas de la capa de salida.

4.0.5.1. Número de Capas

La primera característica en definir es el número de capas que van a formar parte de la red neuronal. La red neuronal a usarse para esta investigación consta de una

capa de entrada, una una capa intermedia y una capa de salida. La red neuronal a implementarse se muestra en la figura 4.21.



Figura 4.21: Representación den número de capas de la red neuronal.
Fuente: Autor

4.0.5.2. Número de Neuronas

La red neuronal propuesta para esta investigación consta de tres capas (entrada, intermedia y salida). El muestreo de los contaminantes se realiza cada minuto y es almacenado en una base de datos MySQL, posteriormente se hace una reducción de los datos para obtener 8 mediciones diarias que determinarán que tan contaminado se encuentra el día, dichos valores son almacenados en otra base de datos. Como se cuenta con 8 datos de material particulado diarios que definen la característica de un día, el número de neuronas de la capa de entrada se establece en 8.

Otro de los parámetros a considerarse es en número de neuronas de la capa intermedia o capa oculta. No existe ningún método de cálculo del número de neuronas para esta capa. Unas de los métodos más empleados es la prueba y error, donde se van incrementando o disminuyendo las neuronas hasta obtener el mejor resultado.

Para esta investigación se plantea conocer el estado del contaminante al siguiente día, por consiguiente se requiere una neurona en la capa de salida.

4.0.5.3. Algoritmo de entrenamiento

El entrenamiento de una red neuronal consiste en el reajuste de los pesos para cada una de las entradas de las neuronas. Los pesos son valores que definen la fuerza de una conexión sináptica entre dos neuronas. Para la presente investigación se emplea el algoritmo de backpropagation, el cual consiste en un método de propagación del error de atrás hacia adelante, es decir, el error se calcula en la capa de salida y se hace un reajuste de los pesos ligados a esta capa para reducir dicho error. Posteriormente

se calcula el error en la capa intermedia y de igual manera se realiza un reajuste de los pesos de esta capa con el objetivo de reducir el error.

4.0.5.4. Extracción de datos de entrenamiento

Se cuenta con una estación de monitoreo de material particulado ubicado en el Campus Huachi de la Universidad Técnica de Ambato. Se realiza el muestreo de los contaminantes Material Particulado PM10 y PM2.5 cada minuto y dichos valores son almacenados en una base de datos MySQL. Se realiza el promedio de cada hora para obtener 8 mediciones diarias, tomando en cuenta las horas pico donde existe mayor movimiento vehicular, ya que de ellos se desprende la mayor cantidad de contaminante. Dichos valores son almacenados en una base de datos diferente y su estructura se la puede observar en la figura 4.16. Se ha realizado mediciones a lo largo de 32 días consecutivos, dichos datos son utilizados en la etapa de entrenamiento de la red neuronal.

4.0.5.5. Preprocesamiento de datos de entrenamiento

La función de activación de la red neuronal admite valores comprendidos entre cero y uno, razón por la cual se procede a realizar la normalización del conjunto de datos de entrenamiento. Se selecciona el mayor valor y se procede a dividir cada uno de los datos para dicho valor, obteniendo como resultado valores que varían entre cero y uno.

4.0.5.6. Procesamiento de datos de entrenamiento

En el post-procesamiento de la información se procede a organizar la matriz de datos para que el software Matlab los interprete. En la figura 4.22 se puede observar cómo queda la matriz. Un día de medición corresponde a ocho datos de material particulado que forman una característica correspondiente a un determinado día.

<i>Dia 1</i>	Dato1	Dato2	Dato3	Dato4	Dato5	Dato6	Dato7	Dato8	Característica 1
<i>Dia 2</i>	Dato1	Dato2	Dato3	Dato4	Dato5	Dato6	Dato7	Dato8	Característica 2
<i>Dia 3</i>	Dato1	Dato2	Dato3	Dato4	Dato5	Dato6	Dato7	Dato8	Característica 3
..
<i>Dia n</i>	Dato n1	Dato n2	Dato n3	Dato n4	Dato n5	Dato n6	Dato n7	Dato n8	Característica n

Figura 4.22: Organización de los datos de entrenamiento de la red neuronal.

Fuente: Autor

4.0.5.7. Programa Matlab

Para el entrenamiento de la red neuronal es necesario conocer las entradas que corresponden a la matriz de datos de entrenamiento. Además, de una vector de valores esperados que corresponden a la salida de la red neuronal. Con el comando `newff` se crea una red de retropropagación de feed-forward con los siguientes parámetros:

- Matriz de valores mínimos y máximos del conjunto de datos de entrenamiento.
- Número de neuronas de la capa intermedia y número de neuronas de la capa de salida.
- Función de transferencia de la capa intermedia, debido a que los valores del conjunto de entrenamiento están acotadas entre 0 y 1, se usa la función `logsig`.
- Función de transferencia de la capa de salida, en este caso se usa la función `purelin` que es una función lineal.
- Algoritmo para la actualización de los pesos y sesgos, se usa la función `trainlm` (Levenberg-Marquardt), que es un algoritmo de backpropagation muy rápido recomendado para entrenamiento supervisado.

```
net=newff(pm2, target2,[64 1], 'logsig', 'purelin', 'trainlm');
```

Posteriormente se procede al entrenamiento del modelo de red neuronal con el comando `train` que admite los parámetros:

- `net`. Es el modelo de red neuronal a entrenarse
- `pm2`. Valores de material particulado presentados a la entrada de la red neuronal.

- target2. Son los valores esperados para el entrenamiento de la red neuronal.

```
[net]=train(net,pm2,target2);
```

4.0.5.8. Simulación

Para la primera simulación se considera una red neuronal con 8 neuronas en la capa de entrada, 16 neuronas en la capa de intermedia y 1 neurona en la capa de salida como se puede observar en la figura 4.23. Posteriormente se procede a entrenar el modelo de red utilizando el conjunto de datos de entrenamiento descritos anteriormente.

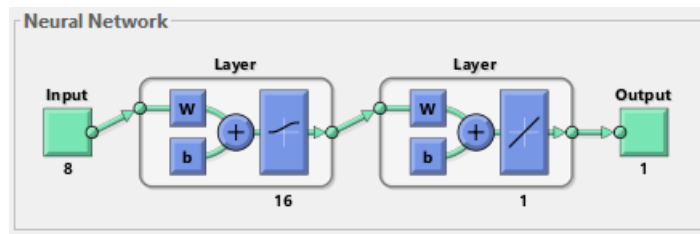


Figura 4.23: Modelo de red neuronal con 16 neuronas en la capa intermedia.
Fuente: Autor

Matlab presenta una gráfica donde se puede observar las curvas de entrenamiento en color azul, validación en color verde y test en color rojo de la figura 4.24. La curva de entrenamiento nos indica como el error de validación va incrementándose o reduciendo conforme la red se va entrenado con cada interacción o época. El entrenamiento finaliza cuando se logra alcanzar mínimo error o el límite de épocas establecidos en la etapa de configuración de la red neuronal. Para el presente caso se observa que en la época 5 se logra reducir el error hasta 10^{-4} , valor establecido en la etapa de configuración. Si la curva de test incrementara significativamente antes que la curva de validación, esto significaría que existe un sobre ajuste, lo que no se da en este caso.

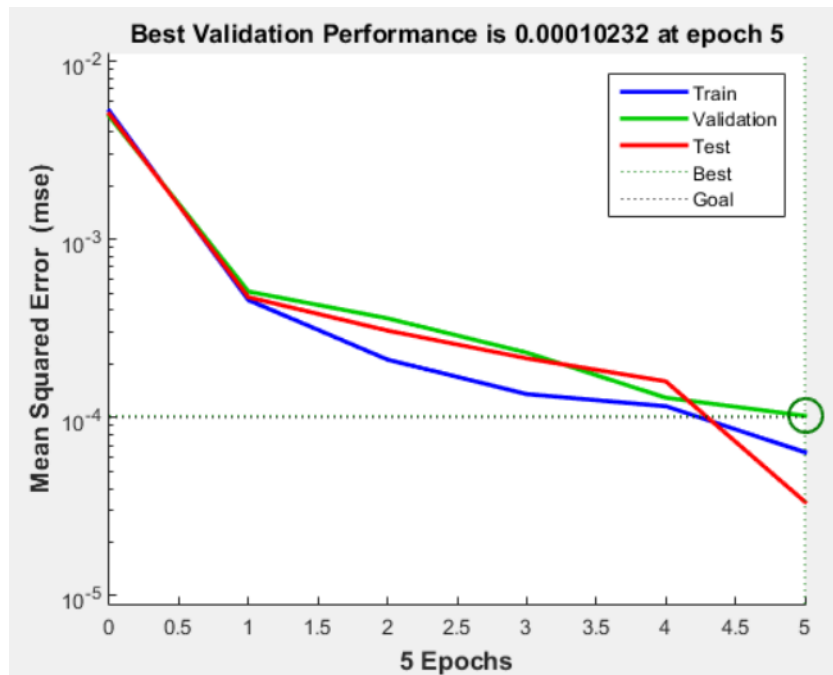


Figura 4.24: Gráfica de la validación de los datos de entrenamiento del modelo con 16 neuronas en la capa intermedia

Fuente: Autor

Para validar el modelo de red neuronal, matlab crea un modelo de regresión que muestra los resultados en relación a los objetivos. En el gráfico 4.25 se muestra los datos de entrenamiento, validación y test. La línea punteada de color negro indica el resultado esperado. La línea continua representa la mejor regresión lineal entre las salidas y los objetivos. El valor R es un valor que indica la relación entre las salidas y los objetivos. Si $R=1$, entonces la relación entre las salidas y los objetivos es exacta. Si R es cero, entonces no existe relación lineal entre las salidas y los objetivos.

Para este ejemplo, los datos de entrenamiento no indican un buen ajuste debido a que la regresión lineal representada con línea continua no se ajusta a los objetivos esperados representada con línea segmentada de color negro. Para los casos de entrenamiento, validación y prueba, el parámetro R muestra valores grandes. El diagrama de dispersión representado con círculos de color negro para todos los casos representa el ajuste inadecuado de los datos.

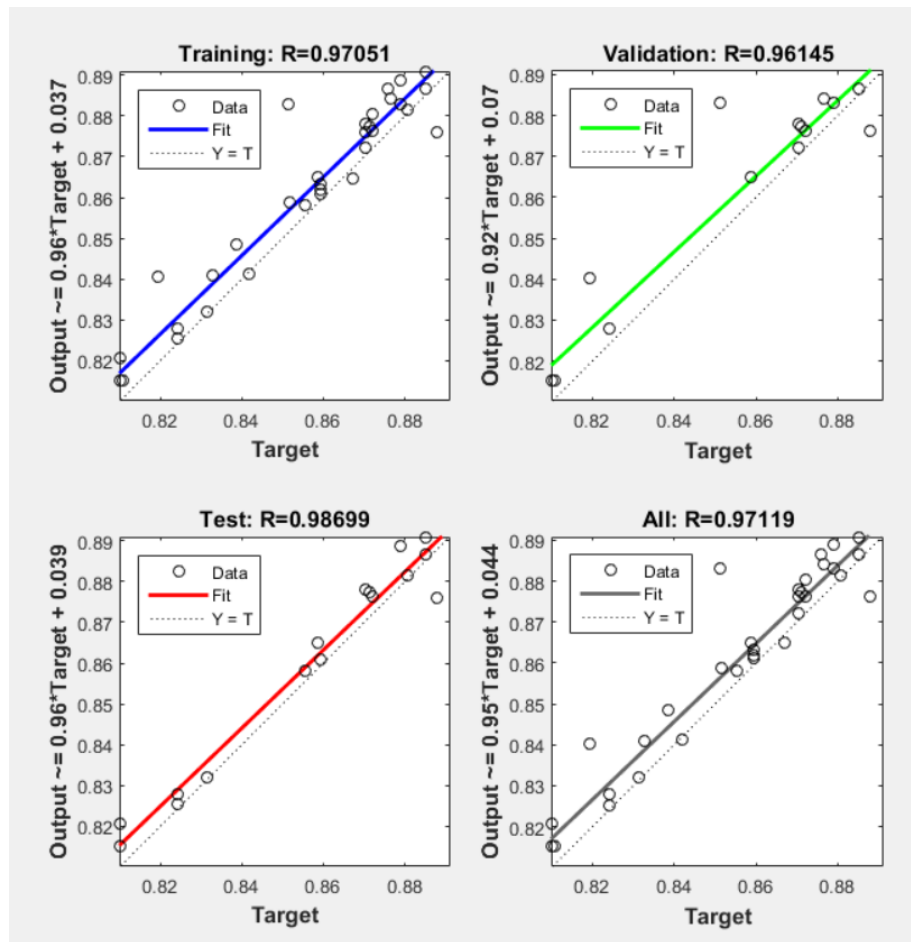


Figura 4.25: Regresión lineal del entrenamiento del modelo con 16 neuronas en la capa intermedia.

Fuente: Autor

En el gráfico 4.26 se representa los valores objetivo representado con línea de color azul y los valores de la salida de la red neuronal representado con línea de color naranja. Por medio de este gráfico se puede deducir que el modelo de red neuronal compuesto por 8 neuronas de entrada, 16 neuronas en la capa intermedia y 1 neurona en la capa de salida, no logra cumplir con los resultados esperados, debido a que las dos líneas no son semejantes.

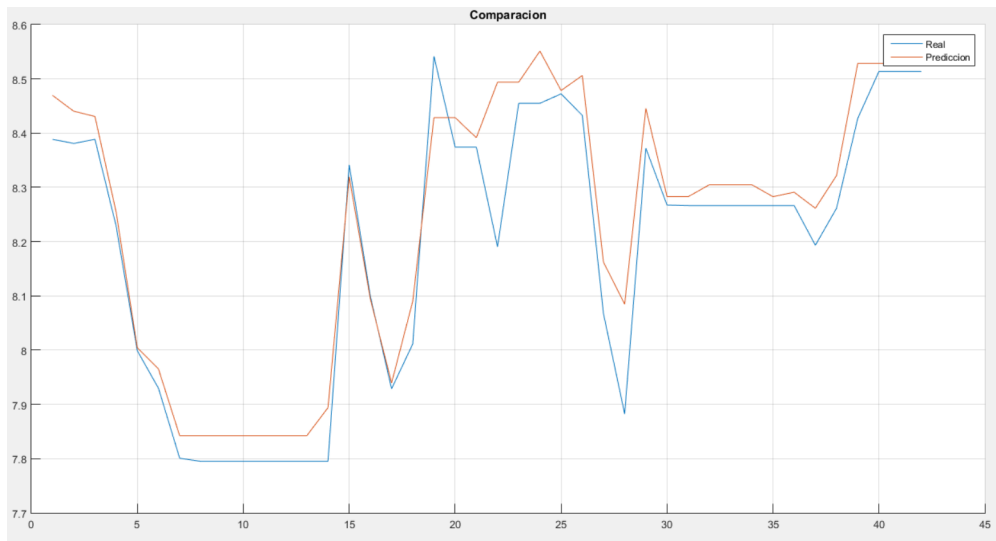


Figura 4.26: Valores reales vs valores obtenidos a la salida del modelo de 16 neuronas en la capa intermedia

Fuente: Autor

Una de las formas de calculo del número de neuronas de la capa intermedia es mediante prueba y error, por ello se plantea un segundo modelo de red neuronal compuesto por 8 neuronas en la capa de entrada, 32 neuronas en la capa intermedia y una neurona en la capa de salida como se representa en la figura 4.27. El error mínimo de entrenamiento se establece en 10^{-4} y el número de épocas en 100.

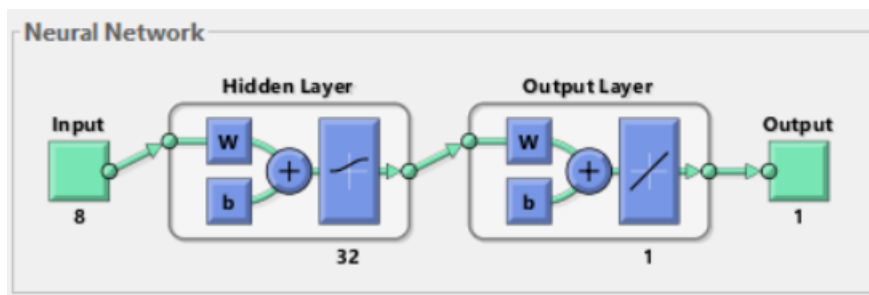


Figura 4.27: Modelo de red neuronal con 16 neuronas en la capa intermedia.

Fuente: Autor

Se entrena el modelo de red neuronal utilizando en algoritmo trainlm, ya que es un algoritmo muy rápido de reajuste de pesos. Luego de entrenar el modelo se obtiene la gráfica 4.28 que representa los errores de entrenamiento. En la época 3 se logra reducir el error hasta 10^{-4} y el entrenamiento se detiene por haber alcanzado con el objetivo establecido. Mientras más similares sean las líneas que representan los errores de entrenamiento, validación y test, el modelo de red neuronal cumple con los

objetivos esperados. La curva de test no se incrementa mayormente sobre la curva de validación, lo que indica que no existe un sobre ajuste de los pesos.

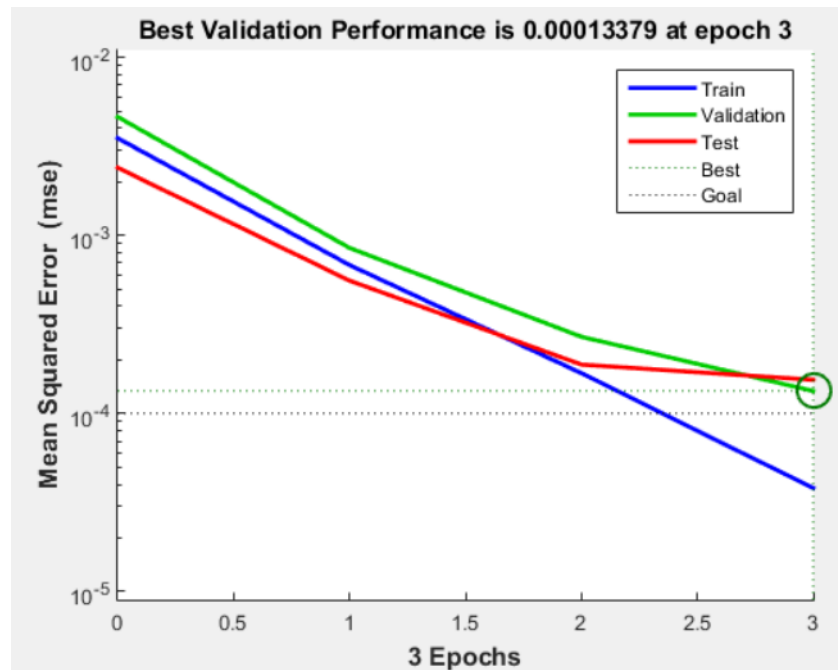


Figura 4.28: Gráfica de la validación de los datos de entrenamiento del modelo con 32 neuronas en la capa intermedia

Fuente: Autor

En la figura 4.29 se muestran los diagramas de regresión lineal de los resultados esperados. La línea segmentada de color negro muestra la regresión lineal de los objetivos esperados, mientras las líneas continuas muestran la regresión lineal de las salidas de la red neuronal. Se puede decir que este modelos compuesto de 32 neuronas en la capa intermedia logra cumplir de mejor manera con los objetivos que el modelo compuesto con 16 neuronas en la capa intermedia. Los círculos de color negro muestra que el ajuste inadecuado de los datos es menor con en modelo planteado para este caso.

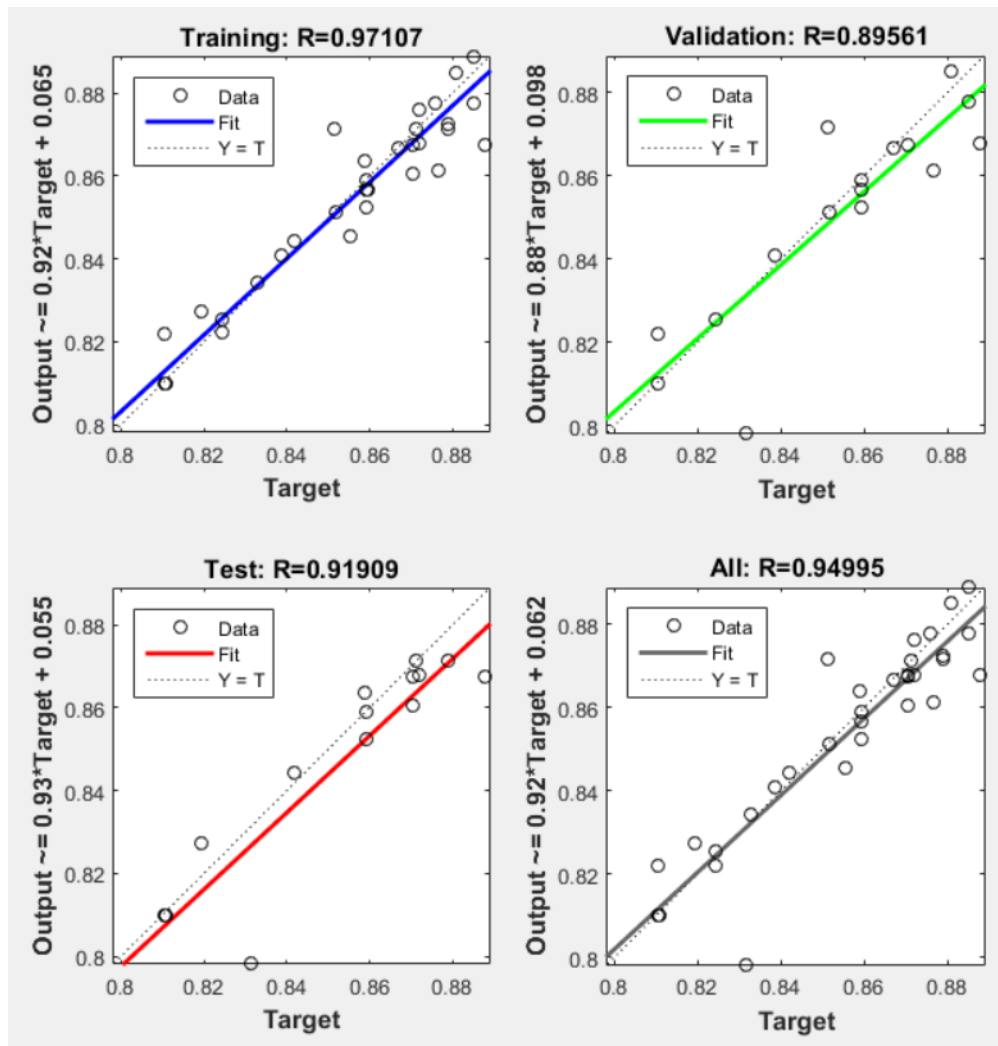


Figura 4.29: Regresión lineal del entrenamiento del modelo con 32 neuronas en la capa intermedia.

Fuente: Autor

En la figura 4.30 se observa de mejor manera el error entre los resultados esperados y los resultados obtenidos a la salida del modelo de red neuronal. La línea de color naranja representa el valor obtenido a la salida de la red neuronal y se ajusta mejor a la línea de color azul que representa los resultados esperados.

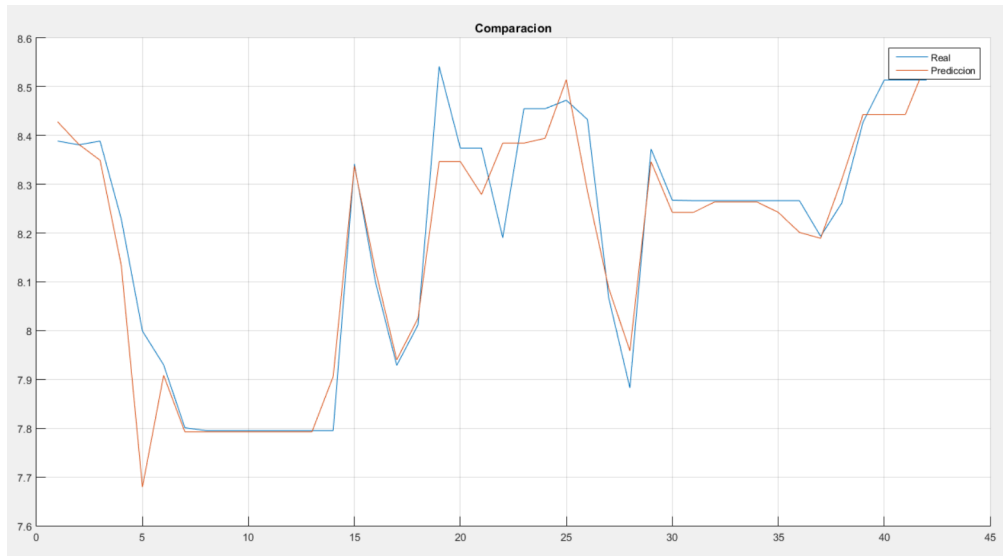


Figura 4.30: Valores reales vs valores obtenidos a la salida del modelo de 32 neuronas en la capa intermedia

Fuente: Autor

Es necesario plantear un tercer caso y realizar la simulación correspondiente del modelo compuesto por 8 neuronas en la capa de entrada, 64 neuronas en la capa intermedia y una neurona en la capa de salida. El número de épocas se establece en 100 y el error mínimo de entrenamiento se establece en 10^{-4} .

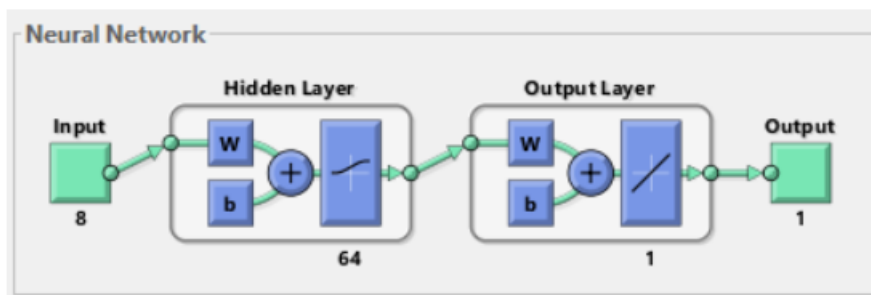


Figura 4.31: Modelo de red neuronal con 16 neuronas en la capa intermedia.

Fuente: Autor

Se realiza el entrenamiento del modelo de red neuronal y los errores de entrenamiento se muestran en la figura 4.32. En la época 3 se logra reducir el error hasta 10^{-4} y se detiene el entrenamiento por haber cumplido con el objetivo. Para que la red neuronal realice una predicción mas acertada, las tres líneas deben ser similares, condición que no se cumple para este caso. La línea de color que representa el test se incrementa en gran medida sobre la línea de color verde que representa los resultados de validación, por lo que se deduce que hubo un sobre ajuste de los pesos.

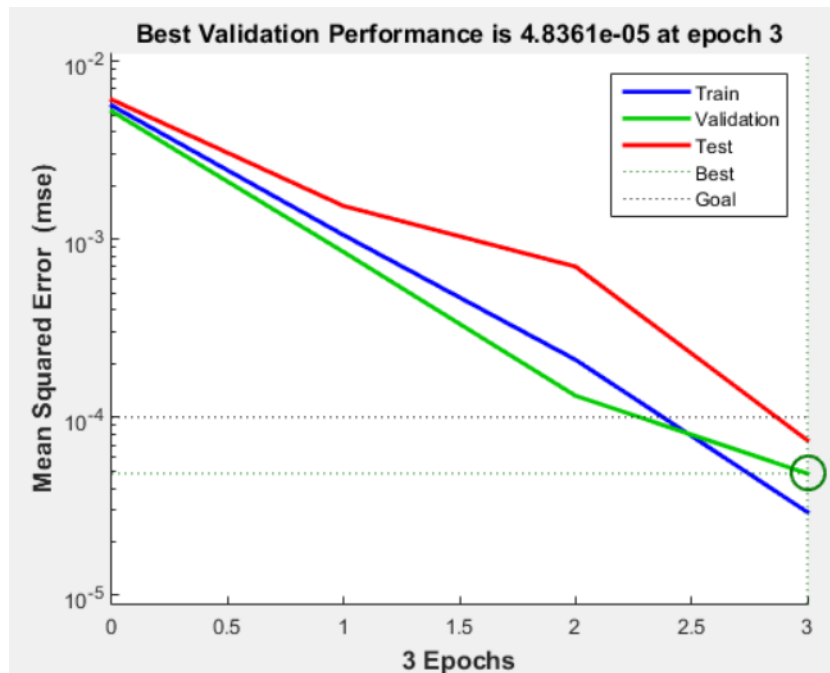


Figura 4.32: Gráfica de la validación de los datos de entrenamiento del modelo con 32 neuronas en la capa intermedia
Fuente: Autor

En la figura 4.33 se representa los diagramas de regresión lineal obtenidos luego del entrenamiento del modelo de red neuronal compuesto por 64 neuronas en la capa intermedia. La línea segmentada de color negro representa la regresión lineal de los valores objetivo y no se ajusta a la línea continua que representa los valores de salida del modelo de red neuronal. Al igual que los modelos probados anteriormente, el valor R es alto. Los círculos de color negro que representan el inadecuado ajuste de los datos, se encuentran muy dispersos, condición que representa que el modelo no realiza una predicción acertada.

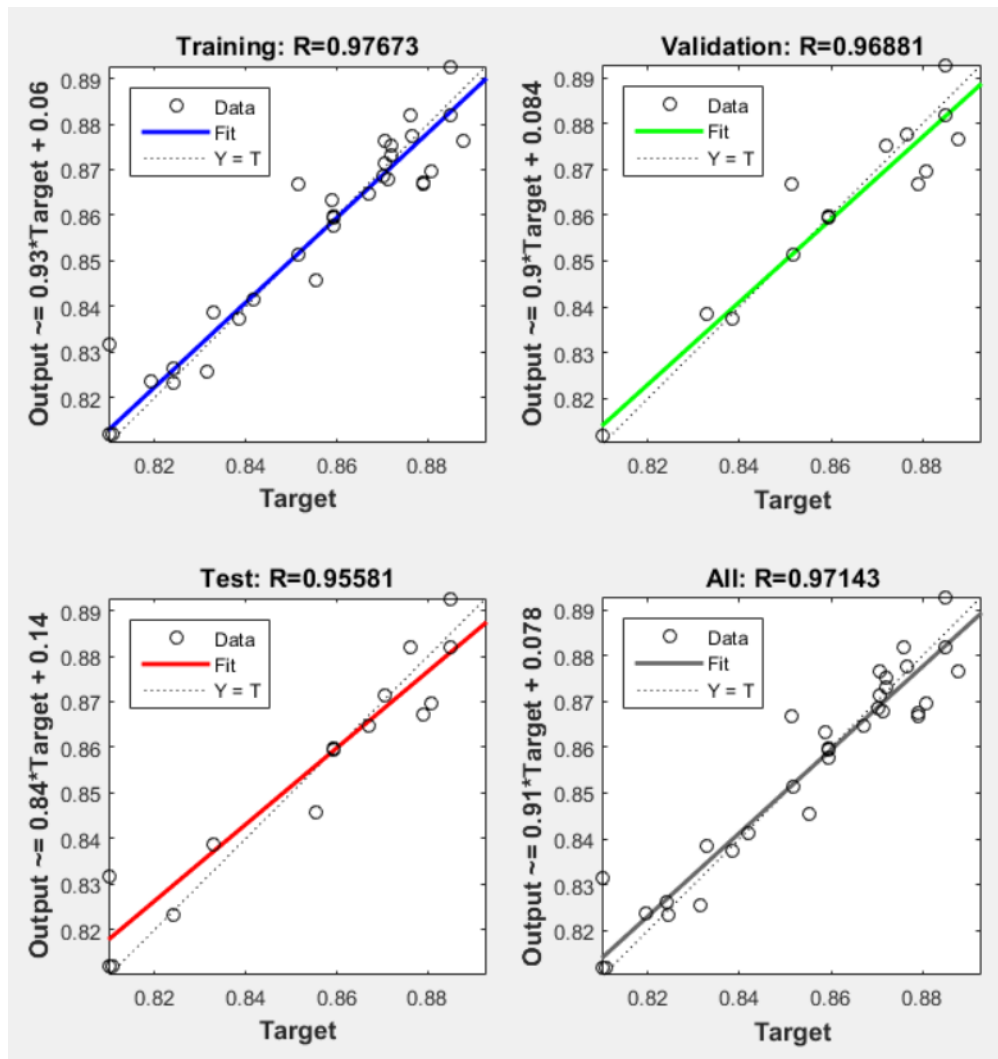


Figura 4.33: Regresión lineal del entrenamiento del modelo con 32 neuronas en la capa intermedia.

Fuente: Autor

En la figura 4.34 se representan las gráficas de los valores objetivos representado con color azul y los valores obtenidos a la salida del modelo de red neuronal con color naranja. Las curvas no se asemejan y se puede deducir que existen errores en la predicción.

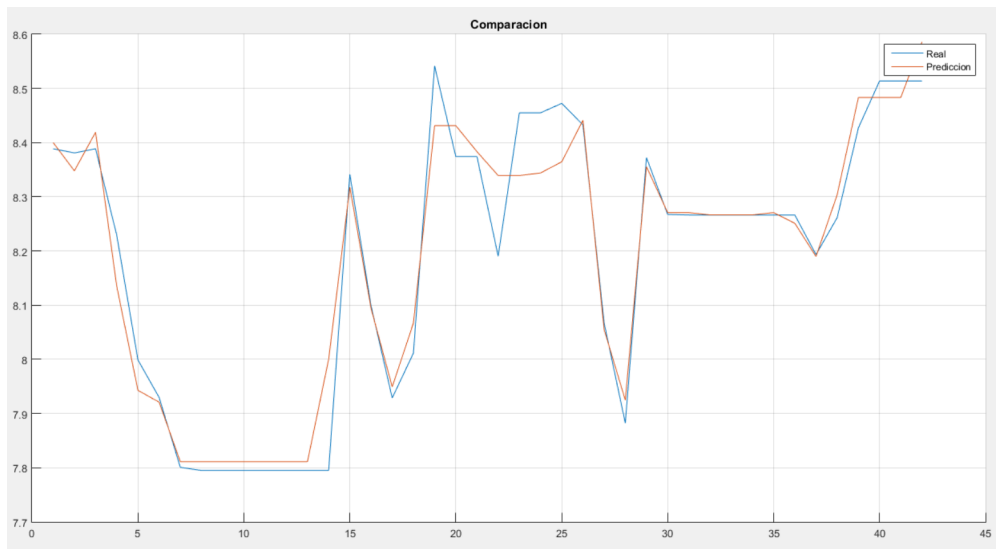


Figura 4.34: Valores reales vs valores obtenidos a la salida del modelo de 32 neuronas en la capa intermedia

Fuente: Autor

Se plantea entrenar un modelo de red neuronal compuesto por 16 neuronas en la capa de entrada que representan los valores de material particulado que varia entra 2.5 *ug* y 10 *ug* conocido como PM10 y valores de material particulado menor a 2.5 *ug*, este modelo cuenta con 32 neuronas en la capa intermedia y una neurona en la capa de salida como se puede observar en la figura 4.35.

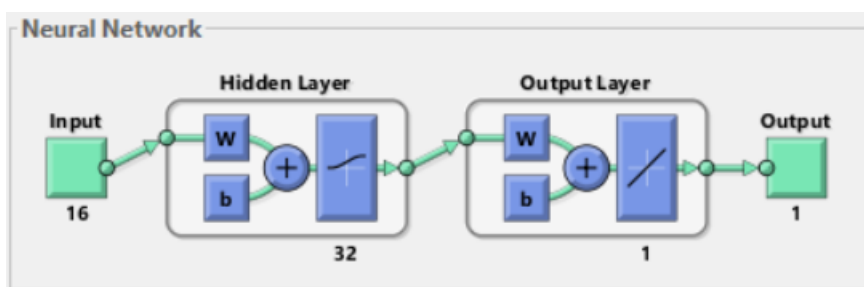


Figura 4.35: Modelo de red neuronal compuesto por 16 neuronas en la capa de entrada, 32 neuronas en la capa intermedia y una neurona en la capa de salida.

Fuente: Autor

Se realiza en entrenamiento del modelo de red neuronal y se obtiene el gráfico de validación presentado en la figura 4.36. Se logra reducir el error de entrenamiento hasta un valor muy cercano a 10^{-4} en la época 3. No existe un sobre ajuste de los pesos ya que los datos de test no sobrepasan a los datos de validación.

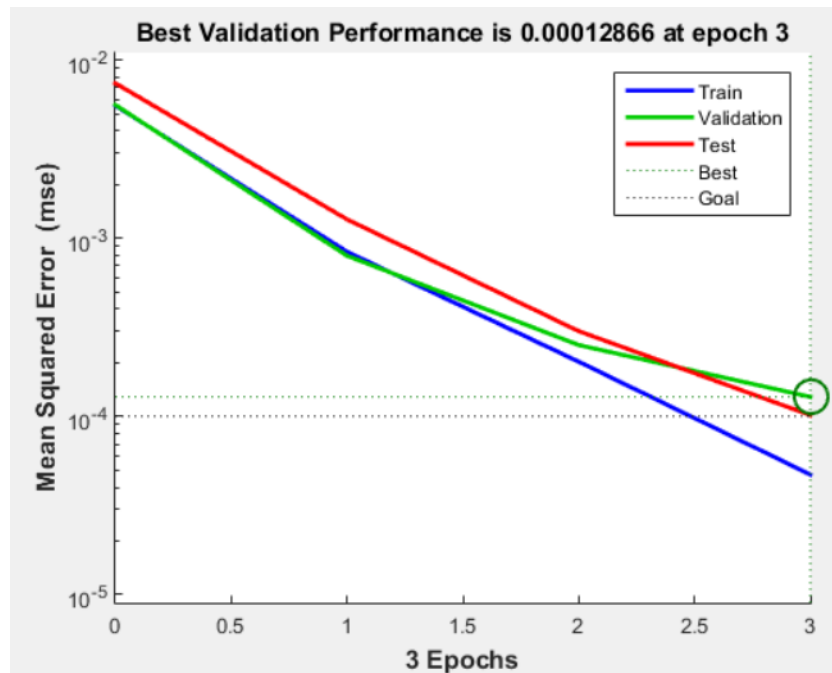


Figura 4.36: Validación de los datos de entrenamiento del modelo de red neuronal compuesto por 16 neuronas en la capa de entrada, 32 neuronas en la capa intermedia y una neurona en la capa de salida.

Fuente: Autor

En la figura 4.37 se muestra la regresión lineal para los datos de entrenamiento, datos de validación y datos de test. La línea continua que representa los valores de la salida no se ajusta a la línea segmentada que representa los valores objetivo. Al igual que los modelos anteriores se tiene un valor R muy alto, lo que significa que se tiene una buena relación entre los datos de salida y los datos objetivo. Se puede decir que existe un ajuste inadecuado de los valores debido a que los círculos de color negro se encuentran muy dispersos.

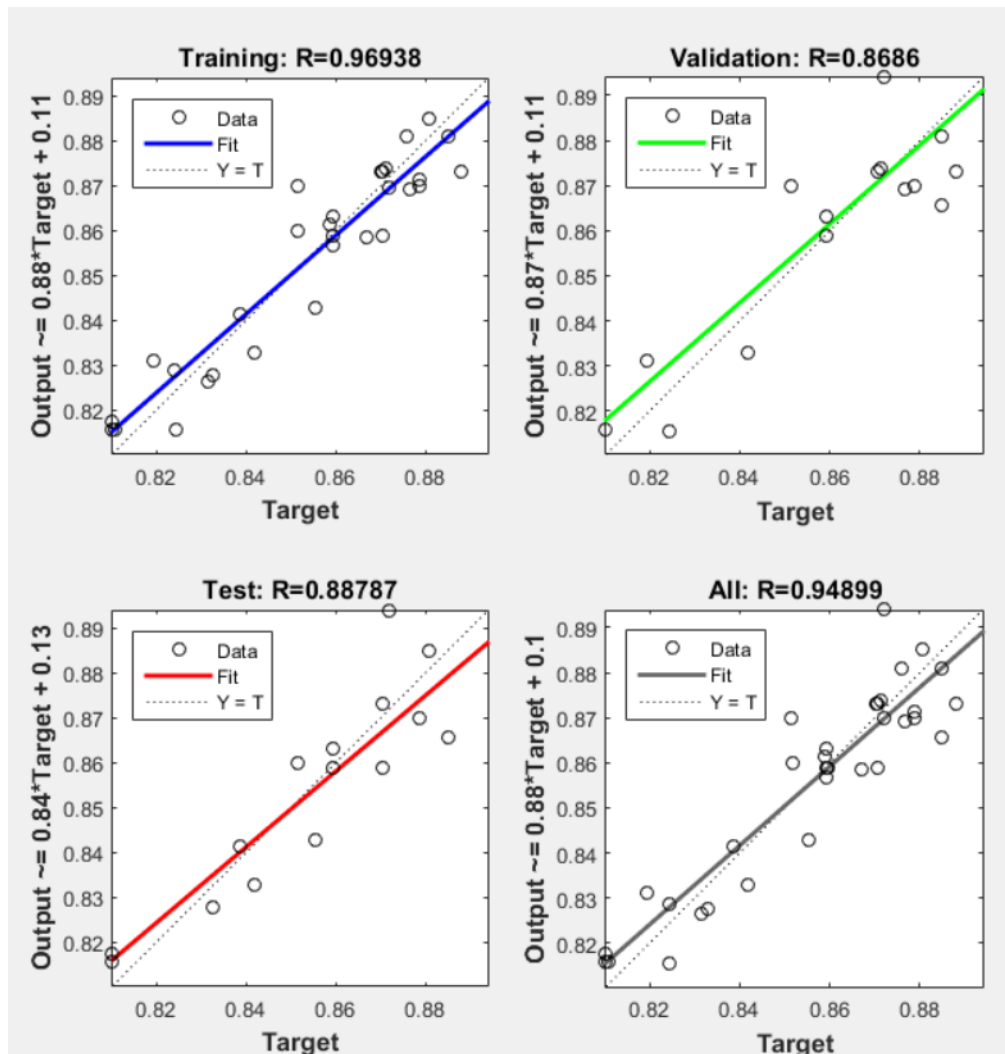


Figura 4.37: Regresión lineal del modelo compuesto por 16 neuronas en la capa de entrada, 32 neuronas en la capa intermedia y una neurona en la capa de salida

Fuente: Autor

En la figura 4.38 se representa una gráfica de los valores objetivos en color azul y los valores obtenidos a la salida del modelo de red neuronal con línea de color naranja. Se puede decir que la predicción no es acertada ya que las dos líneas no son iguales.

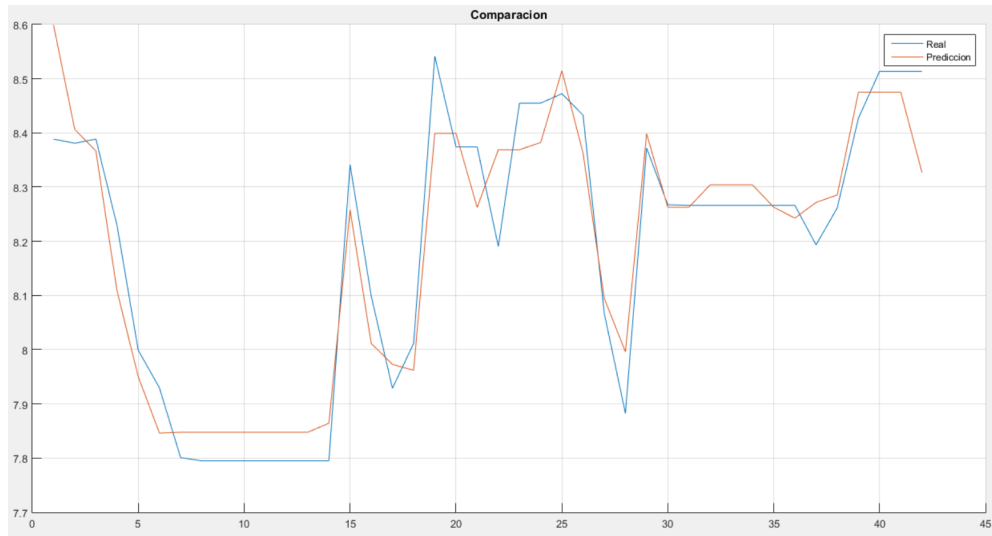


Figura 4.38: Gráfico de los valores objetivo vs valores obtenidos a la salida del modelo de red neuronal compuesto por 16 neuronas en la capa intermedia.

Fuente: Autor

4.0.6. Resultados

Después de realizar el entrenamiento de los modelos de redes neuronales se obtuvieron los siguientes resultados que se muestran en la tabla 4.6. Error cuadrático medio (EMC), Error Absoluto (EA) y Error Porcentual Medio Absoluto (MAPE)

Modelo	Neuronas capa intermedia	EMC	EA	MAPE
Perceptron	16	0.0060	0.0564	0.0069
Perceptron	32	0.0045	0.0458	0.0056
Perceptron	64	0.0036	0.0394	0.0048

Tabla 4.6: Tabla de errores al entrenar los distintos modelos de redes neuronales

Fuente: Autor

CAPÍTULO 5

Conclusiones y Recomendaciones

5.1. Conclusiones

- Los niveles de concentración de material particulado PM10 y PM2.5 registrados en el campo Huachi de la Universidad Técnica de Ambato se encuentra por debajo de los límites establecidos por la organización mundial de la salud que es de $25 \text{ ug}/\text{m}^3$ cada 24 horas.
- Se observa que el incremento de material particulado PM10 y PM2.5 esta ligado con el incremento de la temperatura y descenso de la humedad, registrando los niveles más altos entre las 13:00 y 14:00, horario en el cual se produce un mayor movimiento de personas y vehículos dentro del campus.
- Al analizar los datos obtenidos de la predicción con redes neuronales artificiales, se obtienen mejores resultados con el modelo de red con 24 entradas, 64 neuronas en la capa intermedia y una neurona en la capa de salida.

5.2. Recomendaciones

- Se debe considerar que para tener una mayor precisión del geoposicionamiento, se debe tomar en cuenta el tiempo en que el dispositivo tarda en conectarse a los satélites, ya que el dispositivo utilizado tarda cierto tiempo en conectarse a los satélites y genera valores de posicionamiento inexactos, por lo que es prudente considerar un tiempo de espera.
- Para tener medidas de material particulado se recomienda calibrar el sensor dentro de un ambiente controlado o donde se conozca con exactitud la concentración de estos contaminantes a medir. Además tomar en cuenta que el sensor cuenta con una resistencia para generar calor y provocar un flujo de aire de la parte inferior hacia la parte superior, por lo que es necesario colocar el sensor adecuadamente.
- Es recomendable tener gran cantidad de datos para el entrenamiento de la red

neuronal para que de esta manera las predicciones sean más exactas. En este estudio se tubo mediciones de un mes por lo que los errores de la predicción no tienden a cero como se esperaba. Además se recomienda usar más variables que influyen en la concentración de material particulado como la velocidad del viento, estación, etc.

Bibliografía

- [1] C. L. B. Arcos, “Diseño de la primera red permanente de monitoreo de la calidad del aire en la ciudad de ambato,” 205. [Online]. Available: <http://repositorio.espe.edu.ec/xmlui/handle/21000/13007>
- [2] M. P. V. Chachapoya, “Valoración de la campaña de monitoreo pasivo de gases en la ciudad de ambato para el diseño del sistema de calidad del aire de la ciudad 2013-2014,” 2015. [Online]. Available: <http://repositorio.uisek.edu.ec/handle/123456789/1092>
- [3] X. B. Olabe, “Redes neuronales artificiales y sus aplicaciones.”
- [4] G. A. D. M. de AMbato. (2018). [Online]. Available: <http://gobiernoabierto.ambato.gob.ec/asentamientoshumanos.html>
- [5] X. Domenech, *Química Ambiental: El impacto ambiental de los residuos*, 2000.
- [6] A. Urueña, A. Ferrari, D. Blanco, and E. Valdecasa, “Cloud computing. retos y oportunidades,” Observatorio Nacional de las Telecomunicaciones y de la SI, Tech. Rep., 2012.
- [7] V. Z. Yustres, “Estudio del cloud computing y su interoperabilidad,” 2012.
- [8] P. P. Cruz, *Inteligencia Artificial con aplicaciones a la ingeniería*, 2010.
- [9] S. C. Izaurieta Fernando, “Redes neuronales artificiales,” *Departamento de Física, Universidad de Concepción*, 2010.
- [10] W. H. Organization, “Inheriting a sustainable world: Atlas on children’s health and the environment,” World Health Organization, Tech. Rep., 2017. [Online]. Available: <http://www.who.int/ceh/publications/inheriting-a-sustainable-world/en/>
- [11] A. Pruss-Ustun, J. Wolf, C. Corvalán, R. Boss, and M. Neira, “Preventing disease through healthy environments. a global assessment of the burden of disease from environmental risks,” World Health Organization, Tech. Rep., 2016. [Online]. Available: http://www.who.int/quantifying_ehimpacts/publications/preventing-disease/en/
- [12] O. M. de la Salud. (2016) Cada año mueren 12,6 millones

- de personas a causa de la insalubridad del medio ambiente. [Online]. Available: <http://www.who.int/mediacentre/news/releases/2016/deaths-attributable-to-unhealthy-environments/es/>
- [13] ——. (Septiembre 2016) Calidad del aire ambiente (exterior) y salud. [Online]. Available: <http://www.who.int/mediacentre/factsheets/fs313/es/>
- [14] J. Green and S. Sánchez, “La calidad del aire en américa latina,” *Clean Air Institute*, 2013.
- [15] D. Vega, L. Ocaña, and R. Parra, “Inventario de emisiones atmosféricas del trafico vehicular en el distrito metropolitano de quito,” *Avances en Ciencias e Ingeniería*, 2012. [Online]. Available: https://www.usfq.edu.ec/publicaciones/avances/archivo_de_contenidos/Documents/volumen_7_numero_2/C86-7-2-2015.pdf
- [16] V. G. Cueva, “Inventario preliminar de las emisiones de contaminantes del aire, de los cantones ambato, riobamba, santo domingo de los colorados, latacunga, ibarra, manta, portoviejo, esmeraldas y milagro,” Ministerio del Ambiente, Tech. Rep., 2014.
- [17] G. Ibarra-Berastegi, J. Saenz, and A. Ezcurra, “Using neural networks for short-term prediction of air pollution levels,” *Advances in Computational Tools for Engineering Applications, 2009. ACTEA '09. International Conference on*, 2009. [Online]. Available: <http://ieeexplore.ieee.org/document/5227910/>
- [18] C. Paoli, G. Notton, and M.-L. Nivet, “A neural network model forecasting for prediction of hourly ozone concentration in corsica,” *Environment and Electrical Engineering (EEEIC), 2011 10th International Conference on*, 2011. [Online]. Available: <http://ieeexplore.ieee.org/document/5874661/>
- [19] Y. Chen, H. Qin, and Z. Zhou, “A comparative study on multi-regression analysis and bp neural network of pm2.5 index,” *Natural Computation (ICNC), 2014 10th International Conference on*, 2014. [Online]. Available: <http://ieeexplore.ieee.org/document/6975827/>
- [20] C. Xiaojun, L. Xianpeng, and X. Peng, “Iot-based air pollution monitoring and forecasting system,” *Computer and Computational Sciences (ICCCS), 2015 International Conference on*, 2015. [Online]. Available: <http://ieeexplore.ieee.org/document/7361361/>

- [21] M. Huang, T. Zhang, and J. Wang, "A new air quality forecasting model using data mining and artificial neural network," *Software Engineering and Service Science (ICSESS), 2015 6th IEEE International Conference on*, 2015. [Online]. Available: <http://ieeexplore.ieee.org/document/7339050/>
- [22] J. Villa and F. Daquilema, "Diseño e implementación de un sistema para el monitoreo de los niveles de contaminación del aire para el proyecto de investigación de la universidad nacional de chimborazo," Master's thesis, 2016. [Online]. Available: <http://dspace.unach.edu.ec/handle/51000/2736>
- [23] M. Dedovic, I. Turkovic, T. Konjic, S. Avdakovic, and N. Dautbasic, "Forecasting pm10 concentrations using neural networks and system for improving air quality," *Telecommunications (BIHTEL), 2016 XI International Symposium on*, 2016. [Online]. Available: <http://ieeexplore.ieee.org/document/7775721/>
- [24] K. Shaban, A. Kadri, and E. Rezk, "Urban air pollution monitoring system with forecasting models," *IEEE Sensors Journal*, 2016. [Online]. Available: <http://ieeexplore.ieee.org/document/7370876/>
- [25] P. Bojja and V. K. Karumuri, "Development and evaluation of pollution forecasting model using soft-computing methods for pm10 and so2 in ambient air," *Wireless Communications, Signal Processing and Networking (WISPNET), International Conference on*, 2016. [Online]. Available: <http://ieeexplore.ieee.org/document/7566323/>
- [26] S. Manahan, *Introducción a la Química Ambiental*, primera ed., M. del Carmen Durán Domínguez de Bazúa, Ed. REVERTÉ, 2007.
- [27] C. J. D. Argumedo and J. R. F. Castillo, "Caracterización química de material particulado pm10 en la atmósfera de la guajira colombiana," *Revista Colombiana de Química*, 2016. [Online]. Available: <https://revistas.unal.edu.co/index.php/rcolquim/article/view/56991>
- [28] F. Fajnzylber, *Industrialización en América Latina : de la caja negra al casillero vacío*, 1990. [Online]. Available: http://repositorio.cepal.org/bitstream/handle/11362/27955/S9000502_es.pdf?sequence=1&isAllowed=y
- [29] J. A. R. Castillo, V. M. T. Meza, G. O. Fernández, and V. B. Aburto, "Los efectos agudos de la contaminación del aire en la salud de la población: evidencias de estudios epidemiológicos," *Salud pública de México*, 2001. [Online].

Available: http://scielo.unam.mx/scielo.php?pid=S0036-36342001000600005&script=sci_arttext

- [30] O. M. de la Salud. (2016) Contaminación del aire ambiental. [Online]. Available: http://www.paho.org/hq/index.php?option=com_content&view=article&id=12918%3A2017-ambient-air-pollution&catid=2619%3Aenvironmental-health&Itemid=42246&lang=es
- [31] F. García, R. Agudelo, and K. Jiménez, “Distribución espacial y temporal de la concentración de material particulado,” *Revista Facultad Nacional de Salud Pública*, 2006. [Online]. Available: <http://aprendeonline.udea.edu.co/revistas/index.php/fnsp/article/view/247>
- [32] P. M. A. Fernando, “Implementación de una plataforma de nube privada opensource en la carrera de ingeniería en sistemas computacionales de la universidad católica de santiago de guayaquil, para la gestión de recursos de infraestructura bajo demanda,” 2017. [Online]. Available: <http://repositorio.ucsg.edu.ec/handle/3317/7659>
- [33] W. F. E. Espín, “Estudio para el servicio de arrendamiento de cloud computing infraestructura como servicio para la empresa angos e hijos construcciones,” 2015.
- [34] C. A. Ruiz and M. S. Basualdo, “Redes neuronales: Conceptos básicos y aplicaciones,” 2001.

Anexos y Apéndices

Anexo A

Código de la programación del arduino para la adquisición de datos

```
//LIBRERIA PARA LECTURA DE DHT11
#include "DHT.h"
#define DHTTYPE DHT11
const int DHTPin = 2;
DHT dht(DHTPin, DHTTYPE);

//LIBRERIA PARA COMUNICACION SERIAL NODEMCU
#include <SoftwareSerial.h>
SoftwareSerial MySerial(7, 8); // 0 Rx, 1 Tx
#define pinpm10 3
#define pinpm25 4

//LIBRERIA PARA LECTURA GPS
#include <TinyGPS.h>
SoftwareSerial SerialGPS(6, 5); // 6 Rx, 5 Tx
TinyGPS gps; char dato = ' ';

//VARIABLES GLOBALES LECTURA PM
unsigned long inicio;
unsigned long fin;
unsigned long duracion;
unsigned long pulsoBajo;
unsigned long tiempoMuestreo = 30000;
float ratio = 0;
unsigned long concentracionpm25 = 0;
unsigned long concentracionpm10 = 0;

//DATOS GLOBALES
float lastpm10 = 0;
float lastpm25 = 0;
float lat = 0;
float lon = 0;
```

```

float pm10 = 0.0;
float pm25 = 0.0;
float temp = 0.0;
float hum = 0.0;

//ESTADO DE LAS LECTURAS
bool estadoPM = false;
bool estadoDHT = false;
bool estadoGPS = false;

void setup() {
  pinMode(pinpm25, INPUT); //PM2.5
  pinMode(pinpm10, INPUT); // PM10

  //INICIAR COMUNICACION SERIAL CONSOLA
  Serial.begin(9600);

  //INICIAR DHT11
  dht.begin();

  //INICIAR COMUNICACION SERIAL NODEMCU
  MySerial.begin(9600);

  //INICIAR COMUNICACION SERIAL GPS
  SerialGPS.begin(9600);
  Serial.println("LISTO");
}

void loop() {
  //Leemos los sensores
  estadoDHT = leerDHT();
  estadoPM = leerPM();
  estadoGPS = leerGPS();

  //COMPROBAMOS LOS DATOS Y ENVIAMOS AL NODEMCU
  if ((pm10 > 0.00) && (pm25 > 0.00) && (temp > 0.0) && (hum > 0.0)){
    Serial.println("Datos enviados por wifi");
    enviar_nodemcu();
  }
}

```

```

    } else {
        Serial.println("Datos no disponibles");
        imprimir_datos();
    }
}

```

```

bool enviar_nodemcu() {
    bool estado_uart = false;
    String dato = "";
    static char vlat[10];
    dtostrf(lat, 10, 7, vlat);
    dato += vlat;
    dato += "a";
    static char vlon[11];
    dtostrf(lon, 11, 7, vlon);
    dato += vlon; dato += "b";
    dato += pm10; dato += "c";
    dato += pm25; dato += "d";
    dato += temp; dato += "e";
    dato += hum; dato += "f";
    Serial.println(dato);
    Serial.println("");
    if (MySerial) {
        MySerial.println(dato);
        return true;
    } else {
        return false;
    }
}

```

```

bool leerDHT() {
    float h = dht.readHumidity();
    float t = dht.readTemperature();

    if (isnan(h) || isnan(t)) {
        return false;
    } else {

```

```
    temp = t;
    hum = h;
    return true;
}
}
```

```
bool leerPM() {
    //Leemos los datos
    concentracionpm10 = leer_pinPM(pinpm10);
    concentracionpm25 = leer_pinPM(pinpm25);

    //Convertimos a ug/m3
    lastpm10 = convertirPM10(concentracionpm10);
    lastpm25 = convertirPM25(concentracionpm25);

    //Comprobar si la lectura es correcta y guardar en las variables
    globales
    if (lastpm10>0.00) {
        pm10 = lastpm10;
    }
    if (lastpm25>0.00) {
        pm25 = lastpm25;
    }

    //Valores de retorno
    if (lastpm10>0 && lastpm25>0) {
        return true;
    } else {
        return false;
    }
}
```

```
bool leerGPS() {
    bool newData = false;
    unsigned long chars;
    unsigned short sentences, failed;
```

```

for (unsigned long start = millis(); millis() - start < 1000;) {
    while (SerialGPS.available()) {
        char c = SerialGPS.read();
        if (gps.encode(c)) {
            newData = true;
        }
    }
}
if (newData) {
    float flat, flon;
    unsigned long age;
    gps.f_get_position(&flat, &flon, &age);
    if (gps.hdop() >= 70) {
        lat = flat;
        lon = flon;
        return true;
    } else {
        return false;
    }
}
//gps.stats(&chars, &sentences, &failed);
}

long leer_pinPM(int pin) {
    inicio = millis();
    while (1) {
        duracion = pulseIn(pin, LOW);
        pulsoBajo += duracion; fin = millis();
        if ((fin - inicio) > tiempoMuestreo) {
            //ratio = (pulsoBajo - fin + inicio) / (tiempoMuestreo*10.0);
            ratio = pulsoBajo / (tiempoMuestreo*10.0);
            long concentracion = 1.1*pow(ratio, 3) - 3.8*pow(ratio, 2) + 520 *
                ratio + 0.62;
            pulsoBajo = 0;
            return(concentracion);
        }
    }
}

```



```
}
```

```
float convertirPM10(long concentracionPM10) {  
    double pi = 3.14159;  
    double density = 1.65 * pow(10, 12);  
    double r10 = 0.44 * pow(10, -6);  
    double vol10 = (4 / 3) * pi * pow(r10, 3);  
    double mass10 = density * vol10;  
    double K = 3531.5;  
    return (concentracionPM10)* K * mass10;  
}
```

```
float convertirPM25(long concentrationPM25) {  
    double pi = 3.14159;  
    double density = 1.65 * pow(10, 12);  
    double r25 = 0.44 * pow(10, -6);  
    double vol25 = (4 / 3) * pi * pow(r25, 3);  
    double mass25 = density * vol25;  
    double K = 3531.5;  
    return (concentrationPM25)* K * mass25;  
}
```

```
void imprimir_datos() {  
    Serial.print("Latitud:");  
    Serial.print(lat, 7);  
    Serial.print(" , Longitud:");  
    Serial.print(lon, 7);  
    Serial.print(" , PM10:");  
    Serial.print(pm10);  
    Serial.print(" , PM2.5:");  
    Serial.print(pm25);  
    Serial.print(" , temp:");  
    Serial.print(temp);  
    Serial.print(" , hum:");  
    Serial.print(hum);  
    Serial.print(" , DHT:");  
    Serial.print(estadoDHT);  
}
```

```
Serial.print(" , PM: ");  
Serial.print(estadoPM);  
Serial.print(" , gps: ");  
Serial.println(estadoGPS);  
//Serial.println("\n");  
}
```

Anexo B

Código del modulo Nodemcu para el envío de datos a la plataforma cloud

```
#include <ESP8266WiFi.h>
#include <SoftwareSerial.h>

const char* ssid = "UTA-WiFi";
const char* password = " ";
const char* host = "104.198.195.161";

//Objeto cliente
WiFiClient client;

//Objeto Serial SoftwareSerial
MySerial(D7,D8); // 7 Rx y 8 Tx

char inChar; String string = "";
String lat = ""; String lon = "";
String pm10 = ""; String pm25 = ""; String temp = ""; String hum = "";
bool estado_dato_recibido = false;

void setup(){
  pinMode(LED_BUILTIN, OUTPUT);
  digitalWrite(LED_BUILTIN, HIGH);
  string.reserve(150);

  //Iniciamos el Serial
  Serial.begin(9600);

  //Iniciamos comunicacion UART
  MySerial.begin(9600); //Velocidad maxima 9600

  //Iniciamos la conexion wifi
  WiFi.begin(ssid, password);
```

```

while (Serial.available() > 0)Serial.read();
Serial.print("Conectando a ");
Serial.print(ssid);
while (WiFi.status() != WL_CONNECTED){
    delay(1000);
    Serial.print(".");
}
Serial.println("");
Serial.println("Conectado");
Serial.print("IP: ");
Serial.println(WiFi.localIP());
}

```

```

void loop() {

    while (MySerial.available()>0) {

        //Lectura de caracteres
        inChar = MySerial.read();

        if (inChar == 'a') {
            lat = string;
            string = "";
        }
        else if (inChar == 'b') {
            lon = string;
            string = "";
        }
        else if (inChar == 'c') {
            pm10 = string;
            string = "";
        }
        else if (inChar == 'd') {
            pm25 = string;
            string = "";
        }
        else if (inChar == 'e') {

```

```

    temp = string;
    string = "";
}
else if (inChar == 'f') {
    hum = string;
    string = "";
    estado_dato_recibido = true;
}
else if(inChar == '\n'){
    string = "";
}
else {
    string += inChar;
}
}
if (estado_dato_recibido) {
    if (enviar_wifi()) {
        Serial.println("Dato enviado");
        digitalWrite(LED_BUILTIN, LOW);
        estado_dato_recibido = false;
    }
    else {
        Serial.println("Dato no enviado");
        digitalWrite(LED_BUILTIN, HIGH);
        estado_dato_recibido = false;
    }
    estado_dato_recibido = false;

}
else {
    estado_dato_recibido = false;
}
}

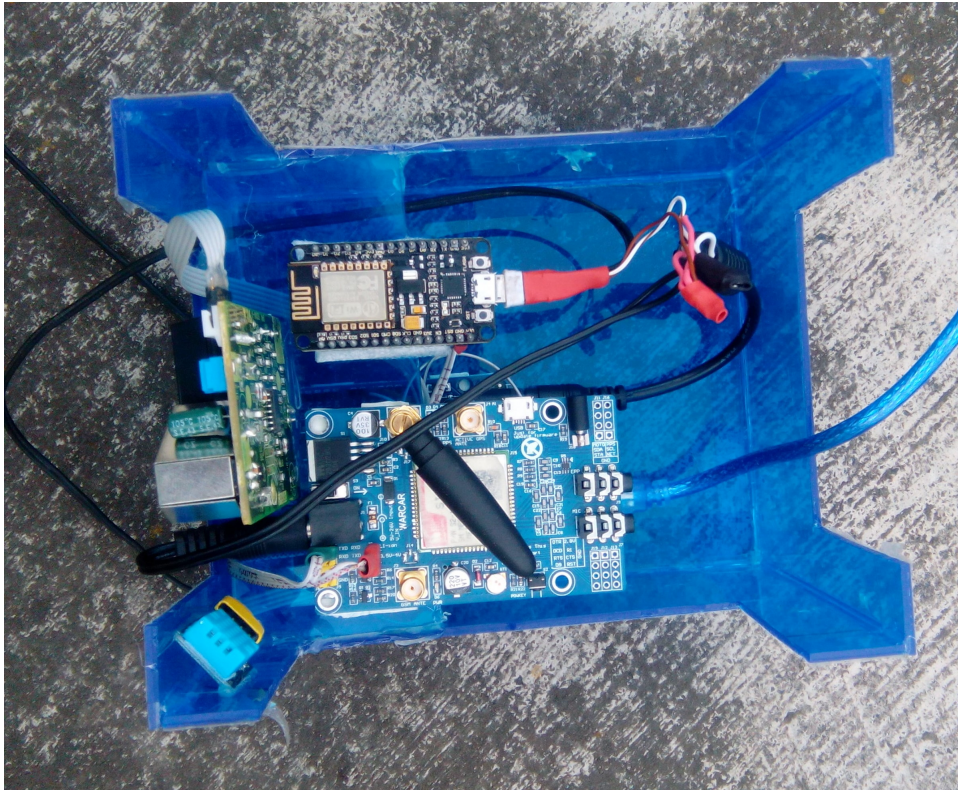
bool enviar_wifi() {
    if (client.connect(host, 80)) {
        //Creamos el string para hacer la peticion

```

```
String peticion = "GET /php/insertar_dato.php?lat=";  
peticion += lat;  
peticion += "&lon=";  
peticion += lon;  
peticion += "&pm10=";  
peticion += pm10;  
peticion += "&pm25=";  
peticion += pm25;  
peticion += "&temp=";  
peticion += temp;  
peticion += "&hum=";  
peticion += hum;  
peticion += "HTTP/1.1\r\n";  
client.println(peticion);  
Serial.println(peticion);  
return true;  
}else {  
return false;  
}  
}
```

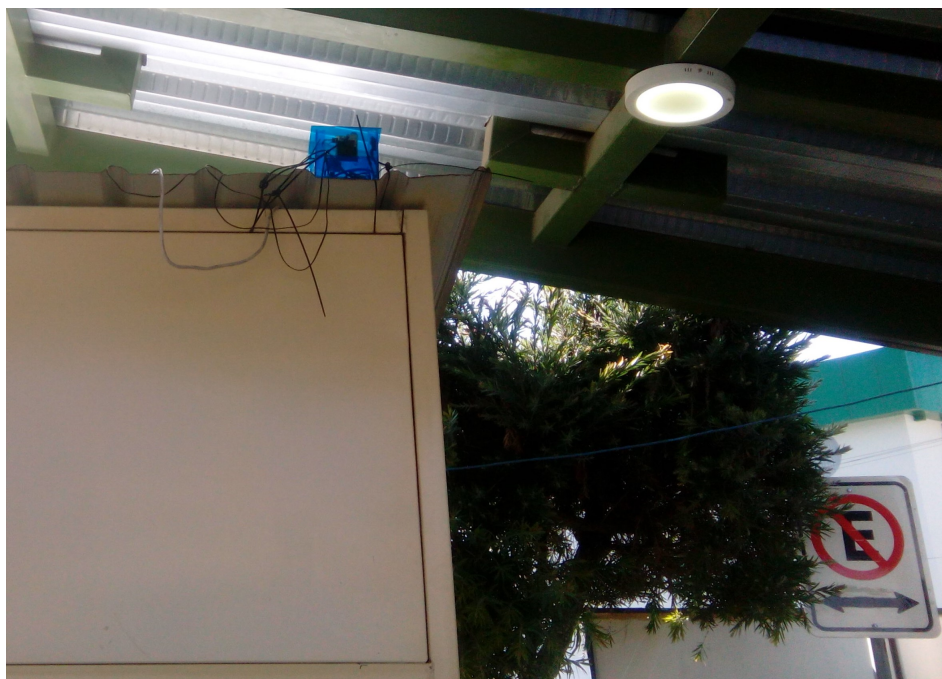
Anexo C

Plataforma móvil de monitoreo de material particulado PM10 y PM2.5 completamente armado



Anexo D

Plataforma de monitoreo de material particulado PM10 y PM2.5 instalada en el campo Huachi de la Universidad Técnica de Ambato



Anexo E

Código de la pagina web para de monitoreo, adquisición, visualización y predicción de la contaminación del aire

E.1. Página principal

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>MR</title>
  <meta name="viewport" content="width=device-width, user-scalable=no,
    initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <link rel="stylesheet" href="css/estilo_index.css">
  <script src="js/jquery.min.js"></script>
  <script src="js/bootstrap.min.js"></script>
</head>
<body>
  <!--BARRA DE NAVEGACION-->
  <nav class="navbar navbar-dark navbar-expand-md sticky-top" style="
    background-color: #070719;">
    <a class="navbar-brand" href="index.html">
      
      Material Particulado
    </a>
    <button class="navbar-toggler navbar-toggler-right" type="button" data-
      -toggle="collapse" data-target="#navbar-1">
      <span class="navbar-toggler-icon" ></span>
    </button>
    <div class="collapse navbar-collapse" id="navbar-1">
      <div class="navbar-nav mr-auto">
        </div>
      <div class="navbar-nav">
```

```

<a class="btn btn-outline-primary" href="index.html">Inicio</a>
<a class="btn btn-outline-info" href="ver_datos.html">
  Datos Diarios</a>
<a class="btn btn-outline-info" href="datos_actuales.html">
  Datos Actuales</a>
<a class="btn btn-outline-success" href="grafico_pm10.html">
  Graficos</a>
<a class="btn btn-outline-danger" href="reporte.html">
  Reporte</a>
</div>
</div>
</nav>
<!--SECCION MAPA-->
<div class="container-fluid">
  <!--mAPA-->
  <section class="main row">
    <div id="map"></div>
  </section>
</div>
<div class="container"></div>
<!--SECCION CUADROS-->
<div class="container-fluid">
  <div class="row">
    <div class="col-sx-12 col-sm-6 col-md-3" id="cuadro1" style="
      padding-left: 0px; padding-right: 0px;">
      <div id="chart1"></div>
    <div class="col-sx-12 col-sm-6 col-md-3" id="cuadro1" style="
      padding-left: 0px; padding-right: 0px;">
      <div id="chart2"></div>
    <div class="col-sx-12 col-sm-6 col-md-3" id="cuadro1" style="
      padding-left: 0px; padding-right: 0px;">
      <div id="chart3"></div>
    <div class="col-sx-12 col-sm-6 col-md-3" id="cuadro1" style="
      padding-left: 0px; padding-right: 0px;">
      <div id="chart4"></div>
  </div>
</div>

```

```

</div>
</div>
<!--PIE DE PAGINA-->
<footer>
  <div class="container centrar">
    <h6>By Juan</h6>
  </div>
</footer>
<script>
  var latitud_m = Array();
  var longitud_m = Array();
  var fecha_m = Array();
  var hora_m = Array();
  var pm10_m = Array();
  var pm25_m = Array();
  var temp_m = Array();
  var hum_m = Array();
  function consultar_datos(){
    $.getJSON("/php/consulta_ultimo_diarios.php",
function(datos) {
      $.each(datos, function(i, dato) {
        latitud_m[i] = dato.latitud;
        longitud_m[i] = dato.longitud;
        fecha_m[i] = dato.fecha;
        hora_m[i] = dato.hora;
        pm10_m[i] = dato.pm10;
        pm25_m[i] = dato.pm25;
        temp_m[i] = dato.temperatura;
        hum_m[i] = dato.humedad;
      });
    });
  }
  function initMap() {
    //alert(pm10_m[0]);
    var map, heatmap;
    var latitud = parseFloat(latitud_m[0]);
    var longitud = parseFloat(longitud_m[0]);

```

```

var fecha = fecha_m[0];
var hora = hora_m[0];
var peso = pm10_m[0]/20;
var pm10 = pm10_m[0];
var pm25 = pm25_m[0];
var temperatura = temp_m[0];
var humedad = hum_m[0];
var posicion= {lat: latitud , lng: longitud};
    var map = new google.maps.Map(document.getElementById
        ('map') , {
        zoom: 17,
        center: posicion ,
        mapTypeId: 'terrain'
    });
    var marker = new google.maps.Marker({
        position: posicion ,
        map: map
    });
var heatmapData = [
    {location: new google.maps.LatLng(latitud , longitud) , weight
        : peso}
];
heatmap = new google.maps.visualization.HeatmapLayer({
    data: heatmapData,
    radius: 80,
    dissipating: true,
    map: map
});
//Marcador
var contentString = '<div>'+<strong>PM10: </strong>'+pm10+'</
    div>'+
    '<div>'+<strong>PM25: </strong>'+pm25+'</div>'+
    '<div>'+<strong>Temperatura: </strong>'+temperatura+'</div>'+
    '<div>'+<strong>Humedad: </strong>'+humedad+'</div>'+
    '<div>'+<strong>Fecha: </strong>'+fecha+'</div>'+
    '<div>'+<strong>Hora: </strong>'+hora+'</div>';
var infowindow = new google.maps.InfoWindow({

```

```

        content: contentString,
        maxWidth: 120
    });
    var marker = new google.maps.Marker({
        position: posicion,
        map: map,
        title: 'marcador'
    });
    marker.addListener('click', function() {
        infowindow.open(map, marker);
    });
}
consultar_datos();
var intervalo=setTimeout('initMap()',1000);
</script>

<style>
    #map {
        width: 100%;
        height: 650px;
    }
</style>
<!--SCRIPT PARA REALIZAR LAS BARRAS-->
<script type="text/javascript" src="https://www.gstatic.com/charts/
    loader.js"></script>
<script type="text/javascript">
    var fecha = Array();
    var pm10 = Array();
    var pm25 = Array();
    var temp = Array();
    var hum = Array();
    function consultar(){
        var ancho=$("#cuadro1").width();
        $.getJSON("/php/consulta_siete.php", function(datos) {
            $.each(datos, function(i, dato) {
                fecha[i] = dato.fecha;
                pm10[i] = dato.pm10;
            });
        });
    }

```

```

        pm25[i] = dato.pm25;
        temp[i] = dato.temperatura;
        hum[i] = dato.humedad;
    });
});

        google.charts.load("current", {packages
            :['corechart']});
google.charts.setOnLoadCallback(drawChart);
function drawChart() {
    var data = google.visualization.arrayToDataTable([
        ["Element", "PM10", { role: "style" } ],
        [fecha[0], pm10[0], "#E74C3C"],
        [fecha[1], pm10[1], "#E44AD"],
        [fecha[2], pm10[2], "#2980B9"],
        [fecha[3], pm10[3], "#17A589"],
        [fecha[4], pm10[4], "#28B463"],
        [fecha[5], pm10[5], "#D68910"],
        [fecha[6], pm10[6], "#839192"]
    ]);
    var view = new google.visualization.DataView(data);
    view.setColumns([0, 1,
        { calc: "stringify",
            sourceColumn: 1,
            type: "string",
            role: "annotation" },
        2]);
    var options = {
        title: "PM10",
        width: ancho,
        height: 300,
        bar: {groupWidth: "95%"},
        legend: { position: "none" },
        hAxis: {
            textStyle: {
                fontName: "Agency FB",
                fontSize: 10
            }
        }
    }
}

```

```

    },
    annotations: {
      textStyle: {
        fontSize:15
      }
    },
    chartArea:{ left:30,top:30,right:10,width:'100%',height:'80%' }
  };
  var chart = new google.visualization.ColumnChart(document.
    getElementById("chart1"));
  chart.draw(view, options);

```

```
//Grafico 2
```

```

var data2 = google.visualization.arrayToDataTable([
  ["Element", "PM25", { role: "style" } ],
  [fecha[0], pm25[0], "#E74C3C"],
  [fecha[1], pm25[1], "#8E44AD"],
  [fecha[2], pm25[2], "#2980B9"],
  [fecha[3], pm25[3], "#17A589"],
  [fecha[4], pm25[4], "#28B463"],
  [fecha[5], pm25[5], "#D68910"],
  [fecha[6], pm25[6], "#839192"]
]);
var view2 = new google.visualization.DataView(data2);
view2.setColumns([0, 1,
  { calc: "stringify",
    sourceColumn: 1,
    type: "string",
    role: "annotation" },
  2]);
var options2 = {
  title: "PM25",
  width: ancho,
  height: 300,
  bar: {groupWidth: "95%"},
  legend: { position: "none" },
  hAxis: {

```

```

    textStyle:{
      fontName: "Agency FB",
      fontSize: 10
    }
  },
  annotations: {
    textStyle: {
      fontSize:15
    }
  },
  chartArea:{ left:40,top:30,right:10,width:'100%',height:'80%' }
};
var chart = new google.visualization.ColumnChart(document.
  getElementById("chart2"));
chart.draw(view2, options2);
//Grafico 3
var data3 = google.visualization.arrayToDataTable([
  ["Element", "Temperatura", { role: "style" } ],
  [fecha[0], temp[0], "#E74C3C"],
  [fecha[1], temp[1], "#E44AD"],
  [fecha[2], temp[2], "#2980B9"],
  [fecha[3], temp[3], "#17A589"],
  [fecha[4], temp[4], "#28B463"],
  [fecha[5], temp[5], "#D68910"],
  [fecha[6], temp[6], "#839192"]
]);
var view3 = new google.visualization.DataView(data3);
view3.setColumns([0, 1,
  { calc: "stringify",
    sourceColumn: 1,
    type: "string",
    role: "annotation" },
  2]);
var options3 = {
  title: "Temperatura",
  width: ancho,
  height: 300,

```



```

    bar: {groupWidth: "95%"},
    legend: { position: "none" },
    hAxis: {
    textStyle:{
        fontName: "Agency FB",
        fontSize: 10
    }
},
    annotations: {
    textStyle: {
        fontSize:15
    }
},
chartArea:{left:40,top:30,right:10,width:'100%',height:'80%'}
};
var chart = new google.visualization.ColumnChart(document.
    getElementById("chart3"));
chart.draw(view3, options3);
//Grafico 4
var data4 = google.visualization.arrayToDataTable([
    ["Element", "Humedad", { role: "style" } ],
    [fecha[0], hum[0], "#E74C3C"],
    [fecha[1], hum[1], "#E44AD"],
    [fecha[2], hum[2], "#2980B9"],
    [fecha[3], hum[3], "#17A589"],
    [fecha[4], hum[4], "#28B463"],
    [fecha[5], hum[5], "#D68910"],
    [fecha[6], hum[6], "#839192"]
]);
var view4 = new google.visualization.DataView(data4);
view4.setColumns([0, 1,
    { calc: "stringify",
    sourceColumn: 1,
    type: "string",
    role: "annotation" },
    2]);
var options4 = {

```

```

        title: "Humedad",
        width: ancho,
        height: 300,
        bar: {groupWidth: "95%"},
        legend: { position: "none" },
        hAxis: {
            textStyle:{
                fontName: "Agency FB",
                fontSize: 10
            }
        },
        annotations: {
            textStyle: {
                fontSize:15
            }
        },
        chartArea:{ left:40,top:30,right:10,width:'100%',height:'80%' }
    };
    var chart = new google.visualization.ColumnChart(document.
        getElementById("chart4"));
    chart.draw(view4, options4);
    }
}
consultar();
</script>
<script async defer src="https://maps.googleapis.com/maps/api/js?key
    =AlzaSyDT4nQ_ZEN4oTfkSEkiSm92_4UK-uchY0&libraries=visualization"
    >
</script>

</body>
</html>

```

Anexo F

Código de la Red Neuronal en Matlab

```
close all;
clear all;
pack;
clc;
pm=[0.7963  0.8074  0.8259  0.7815  0.7815  0.7815  0.7815  0.7815
    0.7815  0.7815  0.7815  0.7815  0.7815  0.7815  0.7759  0.7537
    0.7685  0.7574  0.7870  0.7870  0.7870  0.8093  0.8093  0.8093
    0.7889  0.7889  0.7889  0.8370  0.8056  0.8148  0.8148  0.8148
    0.8148  0.8148  0.8148  0.8315  0.8370  0.8370  0.8204  0.8204
    0.8204  0.8204  0.8241  0.8444  0.8222  0.8185  0.8185  0.8185  0.8185
    0.8185  0.8185  0.8185  0.8185  0.8185  0.8185  0.8185  0.8148
    0.8222  0.8315  0.8593  0.8093  0.8093  0.8148  0.8296  0.8296
    0.8296  0.8537  0.8537  0.8537  0.8778  0.8870  0.8907  0.8907
    0.8907  0.8907  0.8907  0.8907  0.8352  0.8222  0.8222  0.8593
    0.8593  0.8593  0.8593  0.8056  0.8241  0.8685  0.8352  0.7741  0.7741
    0.7759  0.7759  0.7759  0.7759  0.7759  0.7759  0.7759  0.8870
    0.8074  0.7981  0.8056  0.8019  0.7852  0.7852  0.7741  0.8370
    0.8370  0.8370  0.7574  0.8167  0.7648  0.8056  0.8204  0.8204
    0.8204  0.8204  0.8204  0.8204  0.8204  0.8148  0.8278  0.8278
    0.7889  0.7889  0.7889  0.7889  0.8315  0.8444  0.8759  0.8481  0.8907
    0.8481  0.8741  0.8741  0.8741  0.8741  0.8741  0.8741  0.8741
    0.9537  0.9519  0.9667  0.9574  1.0000  0.9111  0.9111  0.9296
    0.9296  0.9296  0.9296  0.8852  0.8963  0.8963  0.8815  0.8426
    0.8833  0.8833  0.8833  0.8833  0.8833  0.8833  0.8296  0.8370
    0.8370  0.9278  0.9278  0.9278  0.9278  0.8611  0.9056  0.9444  0.9352
    0.9481  0.9574  0.9389  0.9389  0.9389  0.9389  0.9389  0.9389
    0.9389  0.8630  0.8407  0.8519  0.8870  0.9056  0.8741  0.8741
    0.8852  0.8852  0.8852  0.8852  0.9130  0.8815  0.8815  0.9056
    0.8407  0.8833  0.8833  0.8833  0.8833  0.8833  0.8833  0.8093
    0.8093  0.8093  0.8093  0.8093  0.8093  0.8278  0.9963  0.8130  0.8204
    0.7944  0.8630  0.8574  0.8407  0.8407  0.8407  0.8407  0.8407
    0.8407  0.8407  0.7870  0.8056  0.8481  0.8259  0.8185  0.8185
```

```

0.8185 0.8481 0.8481 0.8481 0.8481 0.9130 0.8685 0.8685
0.9056 0.9389 0.9574 0.9574 0.9574 0.9574 0.9574 0.9574
0.8796 0.8796 0.9537 0.9537 0.9537 0.9537 0.9537 0.8056 0.8037
0.8370 0.8370 0.7907 0.8167 0.8259 0.8259 0.8259 0.8259
0.8259 0.8259 0.8259 0.8907 0.9037 0.9222 0.9222 0.9000
0.9000 0.9000 0.9278 0.9278 0.9278 0.9278 0.8648 0.8463
0.8463 0.8963 0.9222 0.9537 0.9537 0.9537 0.9537 0.9537
0.9537 0.9056 0.9056 0.8685 0.8685 0.8685 0.8685 0.8796 0.8574
0.8500 0.8519 0.8574 0.8093 0.8241 0.8463 0.8463 0.8463
0.8463 0.8463 0.8463 0.8463 0.8204 0.7815 0.7796 0.8019
0.8407 0.8407 0.8407 0.8000 0.8000 0.8000 0.7778 0.7889
0.7611 0.7611 0.7796 0.7685 0.7500 0.7500 0.7500 0.7500
0.7500 0.7500 0.8704 0.8704 0.8296 0.8296 0.8296 0.8296
0.9111];

```

```

target=[0.8720 0.8472 0.8366 0.8558 0.8384 0.8345 0.8347 0.8377
0.8377 0.8377 0.8377 0.8377 0.8377 0.8377 0.8502 0.8352
0.8428 0.8500 0.8604 0.8407 0.8407 0.8458 0.8583 0.8583
0.8556 0.8456 0.8391 0.8326 0.8611 0.8532 0.8692 0.8692
0.8692 0.8692 0.8692 0.8692 0.8470 0.8486 0.8481 0.8572
0.8572 0.8572];

```

```

pm2=[pm pm pm];
target2=[target target target];
net=newff(pm2, target2,[16],{'logsig','purelin'},'trainlm');
net.trainParam.show = 50;
net.trainParam.lr = 0.01;
net.trainParam.epochs = 100;
net.trainParam.goal = 1e-4;
[net]=train(net,pm2,target2);
real=target.*9.62;
pred=sim(net,pm).*9.62; x=1:42;
hold on;
title('Comparacion');
plot(x,real);
plot(x,pred);
legend('Real','Prediccion');
grid on;

```

Anexo G

Datasheet del Sensor de polvo PPD42

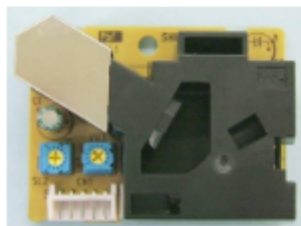
Particle Sensor Model PPD42NS

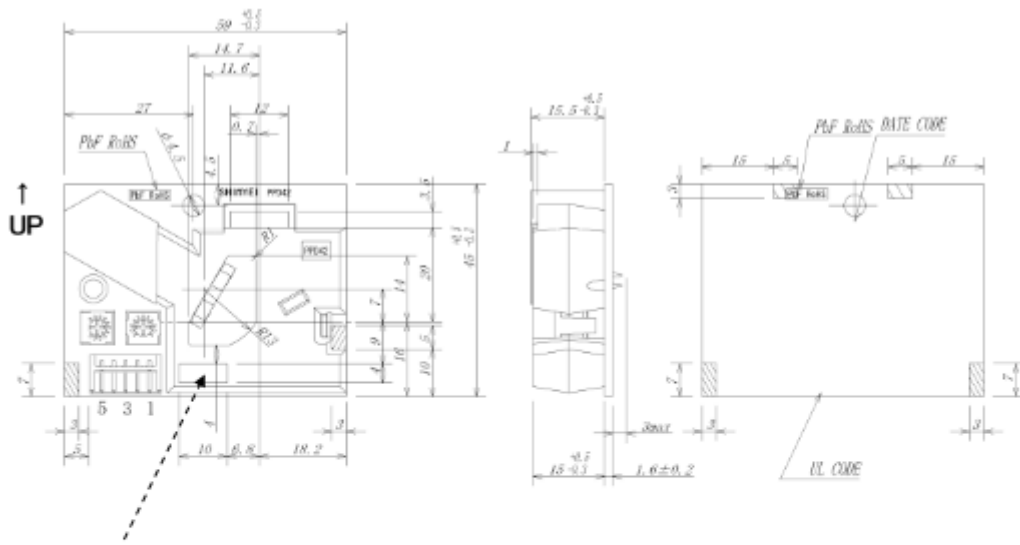
This sensor is to create Digital (Lo Pulse) output to Particulate Matters(PM). Lo Pulse Occupancy time (LPO time) is in proportion to PM concentration. The output from "P1" is for PM whose size is around 1 micro meter or larger. "P1" Output characteristics is shown in the attached drawing Fig2.,when tested in standard condition stipulated below.

Specification

Model	PPD42NS
Detectable particle size	approx. 1 μ m (minimum.)
Detectable range of concentration	0~28,000 pcs/liter (0~8,000pcs/0.01 CF=283ml)
Supply Voltage	DC5V +/- 10% (CN1:Pin1=GND, Pin3=+5V) Ripple Voltage within 30mV
Operating Temperature Range	0~45°C
Operating Humidity Range	95%rh or less (without dew condensation)
Power consumption	90mA
Storage temperature	-30~60°C
Time for stabilization	1 minute after power turned on
Dimensions	59(W) × 45(H) × 22(D) [mm]
Weight	24g(approx.)
Output Method	Negative Logic, Digital output, Hi : over 4.0V(Rev.2) Lo : under 0.7V (As Input impedance : 200k Ω) OP-Amp output, Pull-up resistor : 10k Ω

Configuration and connector pin assignment





Rib (inside of housing)

An additional rib structured to inside of housing to hide RH1 lead wire from view. Following components replaced with equivalent ones; OP-Amp, Volume and AI Electrolytic Cap.

Connector

- CN : S5B-EH(JST)
- 1 : COMMON(GND)
- 2 : OUTPUT(P2)
- 3 : INPUT(5VDC 90mA)
- 4 : OUTPUT(P1)
- 5 : INPUT(T1) ... FOR THRESHOLD FOR [P2]

